

POLITECHNIKA WARSZAWSKA

Rozprawa doktorska

Inżynieria Lądowa, Geodezja i Transport
Dziedzina Nauk Inżynieryjno-Technicznych

mgr inż.

Jakub Łobodecki

Koncepcja i realizacja symulacyjnego środowiska wirtualnego jako
narzędzia do badań percepcji i skuteczności przekazu kartograficznego
w aplikacjach nawigacyjnych wewnątrz budynków

Promotor

dr hab. inż. **Dariusz Gotlib**, prof. uczelni

Warszawa 2026

Podziękowania

Żonie Alicji, za nieograniczoną cierpliwość, wyrozumiałość i wsparcie.

Rodzicom, za cały trud włożony w mój rozwój i edukację.

Dariuszowi Gotlibowi, za niezliczone inspiracje i pomoc w pokonywaniu przeszkód stojących na drodze młodego doktoranta.

Kamilowi Choromańskiemu, za niezastąpione towarzystwo podczas całej mojej naukowej podróży.

Miłoszowi Gnatowi, za entuzjazm i wsparcie przy tworzeniu wirtualnego świata.

Streszczenie

Głównym celem pracy było opracowanie oraz weryfikacja nowatorskiej platformy badawczej wykorzystującej technologie symulacyjne (silniki gier komputerowych) i cyfrowe bliźniaki budynków (digital twins) do wieloaspektowego testowania metod przekazu kartograficznego oraz funkcjonalności aplikacji nawigacyjnych.

W części teoretycznej przeanalizowano m.in. istniejące rozwiązania w zakresie pozycjonowania wewnątrz budynków, wskazując na trudności w prowadzeniu badań w warunkach rzeczywistych. W odpowiedzi na te wyzwania postawiono tezę, że środowisko wirtualne może w istotnym stopniu ograniczyć ten problem i umożliwić testowanie wybranych funkcji aplikacji nawigacyjnych przeznaczonych do nawigacji we wnętrzach budynków.

Metodyka badawcza zakładała wykorzystanie modeli cyfrowych bliźniaków prawdziwych budynków oraz technologii gier. Istotą pracy było zaprojektowanie i implementacja takiego środowiska z wykorzystaniem silnika Unreal Engine oraz zintegrowanie go z funkcjonującą aplikacją nawigacyjną *Mapa PW*. Utworzone środowisko pozwala na wielokrotne powtarzanie eksperymentów w kontrolowanych warunkach oraz umożliwia analizę zachowań i percepcji użytkowników aplikacji nawigacyjnych. Kluczowym elementem pracy było przeprowadzenie kompleksowych badań na grupie ponad 200 uczestników. Zrealizowano 7 eksperymentów sprawdzających m.in. wpływ skali mapy, sposobu wizualizacji trasy, czy rotacji widoku mapy na efektywność poruszania się po budynkach. Przetestowano również scenariusze trudne do realizacji w rzeczywistości, takie jak ewakuacja w sytuacji zagrożenia.

Praca dowodzi, że zastosowanie symulacji stanowi ekonomiczną i efektywną alternatywę dla testów terenowych, umożliwiając optymalizację przekazu kartograficznego i tym samym wpływ na jakość tworzonych aplikacji nawigacyjnych.

Dodatkowym efektem pracy było sformułowanie na podstawie przeprowadzonych badań rekomendacji dla projektantów systemów nawigacji wewnątrz budynków w zakresie sposobu definiowania przekazu kartograficznego.

Słowa kluczowe: nawigacja wewnątrz budynków, kartografia wnętrz budynków, środowisko wirtualne, symulator nawigacji, trening przestrzenny, projektowanie aplikacji geoinformacyjnych, cyfrowe bliźniaki, serious games, Unreal Engine

Abstract

The primary objective of this study was to develop and verify an innovative research platform utilising simulation technologies (game engines) and digital twins of buildings for multi-faceted testing of cartographic communication methods and navigation application functionalities.

The theoretical section analysed existing solutions for indoor positioning, highlighting the difficulties in conducting research under real-world conditions. In response to these challenges, a thesis was formulated that a virtual environment could significantly mitigate this problem and enable the testing of selected functions of navigation applications designed for indoor navigation.

The research methodology assumed the use of digital twin models of real buildings and gaming technologies. The essence of the work was the design and implementation of such an environment using the Unreal Engine and its integration with the operational navigation application, Mapa PW. The developed environment allows for multiple repetitions of experiments under controlled conditions and enables the analysis of user behaviour and perception regarding navigation applications. A key element of the work was conducting comprehensive research involving over 200 study participants. Seven experiments were carried out examining, amongst other factors, the influence of map scale, route visualisation methods, and map view rotation on the efficiency of movement within buildings. Scenarios difficult to implement in reality were also tested, such as evacuation in emergency situations.

The study demonstrates that the application of simulation constitutes an economical and effective alternative to field testing, enabling the optimisation of cartographic communication and thereby influencing the quality of developed navigation applications.

An additional outcome of the work was the formulation of recommendations, based on the conducted research, for designers of indoor navigation systems regarding the method of defining cartographic communication.

Keywords: indoor navigation, indoor cartography, virtual environment, navigation simulator, spatial training, map user experience, digital twins, serious games, Unreal Engine

Spis treści

Streszczenie	5
Abstract.....	6
Wykaz wykorzystanych skrótów.....	11
Informacja o wykorzystaniu sztucznej inteligencji.....	12
1. Wprowadzenie	13
1.1. Uzasadnienie wyboru tematyki rozprawy i identyfikacja luki badawczej	13
1.2. Cel, teza pracy oraz pytania badawcze.....	16
1.3. Struktura rozprawy.....	18
2. Analiza istniejących rozwiązań w zakresie nawigacji i lokalizacji wewnątrzbudynkowej...20	
2.1. Kartografia wnętrz budynków.....	20
2.2. Technologie i systemy pozycjonowania w środowiskach zamkniętych	23
2.3. Analiza rynku aplikacji przeznaczonych do nawigacji wewnątrz budynków.....	26
3. Przegląd metod testowania i badania map oraz aplikacji mapowych	33
3.1. Badania oparte o metody ankietowe i analizę opinii użytkowników.....	36
3.2. Badania z wykorzystaniem śledzenia ruchu gałek ocznych.....	38
3.3. Testy terenowe w warunkach rzeczywistych.....	40
4. Zastosowanie symulatorów w badaniach naukowych	42
4.1. Cyfrowe bliźniaki (Digital Twins).....	42
4.1.1. Cyfrowy bliźniak budynku – definicja	42
4.1.2. Digital Twins w różnych branżach - zastosowania praktyczne	44
4.2. Badania kartograficzne i nawigacyjne wykorzystujące symulatory i środowiska wirtualne.....	45
4.3. Symulatory środowisk wewnątrzbudynkowych	49
5. Metodyka badawcza oraz koncepcja proponowanego narzędzia badawczego	55

5.1.	Założenia główne	55
5.2.	Opis procesu badawczego.....	59
5.3.	Charakterystyka zmiennych niezależnych kontrolowanych w procesie badawczym	63
5.4.	Opis danych zbieranych podczas procesu badawczego	66
6.	Techniczne aspekty implementacji symulatora	69
6.1.	Silnik gier Unreal Engine jako podstawa platformy badawczej	69
6.2.	Architektura informatyczna systemu	73
6.3.	Integracja Cesium i Unreal.....	79
6.4.	Modele budynków – proces pozyskiwania i aspekty wydajnościowe	80
6.5.	Wybór technologii renderowania grafiki trójwymiarowej	86
6.5.1.	Wyświetlanie złożonej geometrii.....	86
6.5.2.	Renderowanie globalnego oświetlenia.....	88
6.6.	Implementacja mechanizmów symulacji ruchu użytkownika	90
6.7.	Warianty dostępu do środowiska symulacyjnego – aspekty techniczne i użytkowe	94
6.7.1.	Dostęp do systemu badawczego.....	94
6.7.2.	Wariant A – przesyłanie obrazu z chmury.....	97
6.7.3.	Wariant B – uruchamianie aplikacji na komputerze desktop.....	102
6.8.	Wybór metody wyświetlania mapy wewnątrz budynków	103
6.8.1.	Metoda nr 1 (odrzucona).....	104
6.8.2.	Metoda nr 2 (wybrana)	107
6.9.	Komunikacja pomiędzy aplikacją nawigacyjną a symulatorem	110
6.9.1.	Opis wybranej technologii komunikacji	110
6.9.2.	Implementacja serwera komunikacyjnego	111
6.9.3.	Dokumentacja interfejsu API.....	112
6.10.	Integracja aplikacji „Mapa PW” z symulatorem.....	114

6.11.	Mechanizm rejestrowania i zbierania danych badawczych	122
6.12.	Badanie pilotażowe - wstępna ewaluacja narzędzia badawczego	124
6.13.	Profilowanie i optymalizacja środowiska wirtualnego	131
6.14.	Aplikacja wspomagająca analizę wyników badań	136
7.	Badanie na użytkownikach z wykorzystaniem symulatora	138
7.1.	Cele i ogólna charakterystyka badania	138
7.2.	Poziom treningowy oraz scenariusze testowe	139
7.3.	Narzędziach wykorzystane do analizy danych	142
7.4.	Analiza grupy badawczej.....	148
7.5.	Przeprowadzone eksperymenty – założenia i analiza wyników.....	156
7.5.1.	Eksperyment 1 – wpływ skali wyświetlanej mapy na efektywność nawigacji.	161
7.5.2.	Eksperyment 2a – wpływ prezentacji wyznaczonej trasy lub aktualnej lokalizacji na efektywność nawigacji	168
7.5.3.	Eksperyment 2b – wpływ prezentacji wyznaczonej trasy lub aktualnej lokalizacji na zapamiętywanie przebytej drogi	177
7.5.4.	Eksperyment 3 – wpływ sposobu prezentacji kierunku patrzenia użytkownika na efektywność nawigacji.....	180
7.5.5.	Eksperyment 4 – wpływ prezentacji wyznaczonej trasy na efektywność nawigacji	186
7.5.6.	Eksperyment 5 – wpływ wyświetlenia na ekranie dodatkowych wskazówek nawigacyjnych na efektywność nawigacji oraz sposób korzystania z mapy	192
7.5.7.	Eksperyment 6 – sprawdzenie potrzeby korzystania z mapy w dobrze oznakowanym budynku.....	198
7.5.8.	Eksperyment 7 – wpływ wyświetlania trasy podczas poszukiwania pomieszczenia w budynku na obraną później trasę ewakuacji	206
7.6.	Analiza dodatkowych swobodnych odpowiedzi użytkowników w badaniu ankietowym.....	216

7.6.1.	Ocena ogólna działania symulatora	217
7.6.2.	Ocena map wewnątrz budynków	221
7.6.3.	Sugestie ulepszeń.....	222
7.7.	Podsumowanie wniosków oraz rekomendacje dla twórców aplikacji nawigacyjnych wynikające z przeprowadzonych testów	225
8.	Ewaluacja opracowanego narzędzia	231
9.	Podsumowanie i wnioski końcowe.....	234
	Bibliografia.....	238
	Spis rysunków	250
	Spis tabel.....	260
	Spis załączników.....	261
	Załącznik 1. Opis architektury i funkcjonalności aplikacji „Mapa PW”	262
	Załącznik 2. Statystyki oraz wizualizacja wszystkich modeli budynków wykorzystanych w symulatorze badawczym	276

Wykaz wykorzystanych skrótów

API	(ang. Application Programming Interface)
AR	(ang. Augmented Reality)
BLE	(ang. Bluetooth Low Energy)
FPS	(ang. First Person Shooter)
GNSS	(ang. Global Navigation Satellite Systems)
IMU	(ang. Inertial Measurement Unit)
IPS	(ang. Indoor Positioning System)
LBS	(ang. Location Based Services)
NFC	(ang. Near Field Communication)
PDR	(ang. Pedestrian Dead Reckoning)
POI	(ang. Point of Interest)
PW	Politechnika Warszawska
QR	(ang. Quick Response)
RFID	(ang. Radio-Frequency Identification)
SaaS	(ang. Software as a Service)
SION	System Informacji o Nieruchomościach Politechniki Warszawskiej
Testy A/B	(ang. A/B testing)
UE	(ang. Unreal Engine)
UWB	(ang. Ultra Wideband)
VR	(ang. Virtual Reality)

Informacja o wykorzystaniu sztucznej inteligencji

Duże modele językowe zostały wykorzystane w niektórych pracach programistycznych oraz na etapie opracowywania danych zebranych podczas badań, a także wspomagały wyszukiwanie i selekcję źródeł bibliograficznych, tłumaczenia fragmentów literatury naukowej z języków obcych oraz korekty językowe, gramatyczne i stylistyczne.

Oświadczam, że jestem autorem niniejszej pracy, a jej treść ma charakter oryginalny w rozumieniu przepisów prawa autorskiego. Ponoszę pełną odpowiedzialność za całość pracy, w tym za rzetelność naukową przedstawionych treści, wyników i wniosków.

1. Wprowadzenie

1.1. Uzasadnienie wyboru tematyki rozprawy i identyfikacja luki badawczej

W ostatnich dwóch dekadach korzystanie z aplikacji nawigacyjnych należących do kategorii aplikacji i usług opartych o lokalizację (ang. *Location Based Services*) stało się integralną częścią codziennego życia użytkowników smartfonów i innych urządzeń mobilnych. Rozwój technologii GNSS oraz powszechność dostępu do Internetu umożliwiły szybkie i precyzyjne wyznaczanie pozycji użytkownika na zewnątrz budynków, co przełożyło się na masowe wykorzystanie aplikacji nawigacyjnych w transporcie, turystyce czy logistyce. Według agendy badawczej dla LBS (H. Huang i in. 2018), usługi tego typu są obecnie jednym z kluczowych obszarów rozwoju technologii mobilnych, a ich dostępność na każdym telefonie komórkowym sprawiła, że korzystanie z nawigacji stało się powszechne i oczekiwane przez użytkowników.

Pomimo dynamicznego rozwoju nawigacji w przestrzeni otwartej (Laor i Galily 2022), aplikacje przeznaczone do nawigacji wewnątrz budynków (ang. *indoor navigation*) pozostają wciąż rzadkością i nie są rozwiązaniami w pełni dojrzałymi technologicznie (Simões i in. 2020a). Wynika to z szeregu wyzwań technicznych, takich jak brak wystarczająco dokładnych metod pozycjonowania (Zhang i in. 2025) oraz ograniczona dostępność szczegółowych danych przestrzennych dotyczących wnętrza budynków (Sarot, Delazari, i Camboim 2021). W efekcie użytkownicy, którzy oczekują podobnego, względem przestrzeni otwartych, poziomu wsparcia nawigacyjnego w galeriach handlowych, kampusach uczelni, szpitalach czy na lotniskach, napotykają na znaczne ograniczenia funkcjonalne obecnych rozwiązań.

W literaturze podkreśla się, że w nadchodzących latach można spodziewać się dalszego rozwoju technologii lokalizacyjnych dla przestrzeni wewnętrznych, zarówno w zakresie dokładności pozycjonowania, jak i dostępności danych przestrzennych (Cheema 2018). W związku z tym już teraz konieczne jest prowadzenie badań praktycznych nad skutecznością przekazu kartograficznego oraz funkcjonalnością aplikacji nawigacyjnych, które muszą być dostosowane do specyficznych warunków panujących wewnątrz budynków, takich jak

wielopoziomowość, ograniczona widoczność, nieuporządkowany ruch w niektórych częściach budynków (np. hole, aule) czy dynamiczne zmiany organizacji przestrzeni. Badania te są niezbędne, aby wypracować nowe metody prezentacji informacji przestrzennej, odpowiednie dla użytkowników poruszających się w środowiskach zamkniętych.

Współczesny użytkownik, przyzwyczajony do wsparcia, jakie oferują aplikacje nawigacyjne, często doświadcza trudności podczas poruszania się w nieznanych miejscach bez ich pomocy. Dotyczy to zarówno pieszych, jak i kierowców, dla których brak dostępu do nawigacji może prowadzić do dezorientacji, wydłużenia czasu podróży, a nawet zwiększenia ryzyka popełnienia błędów na drodze. Zjawisko to zostało opisane w literaturze (W. Yan i in. 2022) jako tzw. „uzależnienie od nawigacji cyfrowej”, które wpływa na spadek umiejętności orientacji przestrzennej u użytkowników. Wspomniane badania wskazują, że coraz większa liczba osób deklaruje trudności w samodzielnym planowaniu tras bez wsparcia aplikacji mobilnych. Dodatkowo, długotrwałe korzystanie z cyfrowych systemów nawigacyjnych prowadzi do osłabienia zdolności uczenia się przestrzennego nawet u ekspertów, takich jak zawodowi kierowcy, co potwierdzają wyniki badań kognitywnych i behawioralnych (Ying, Dong, i Fabrikant 2024). Powyższe prowadzi do wniosku iż w badaniach nad skutecznością aplikacji nawigacyjnych należy brać pod uwagę również to, jak korzystanie z aplikacji nawigacyjnych wewnątrz budynków wpływa na zdolność użytkowników do samodzielnego zapamiętywania i rekonstruowania układu przestrzennego obiektów. Negatywny wpływ rozwiązań nawigacyjnych znajduje potwierdzenie w dotychczasowych badaniach, natomiast luka badawcza ujawnia się w kontekście metod projektowania aplikacji z uwzględnieniem percepcji przestrzeni wewnętrznych przez ich użytkowników i minimalizacji szkodliwości wpływu ich użycia na orientację przestrzenną człowieka.

Testowanie systemów nawigacyjnych wewnątrz budynków w warunkach rzeczywistych jest utrudnione nie tylko ze względu na brak dostępnych powszechnie dokładnych technologii lokalizacyjnych, ale także z powodu trudności organizacyjnych, takich jak ograniczony dostęp lub konieczność dezorganizacji pracy w budynku. W związku z tym w ramach niniejszej rozprawy doktorskiej zaproponowano wykorzystanie środowiska symulacyjnego, które pozwala na wielokrotne powtarzanie eksperymentów w kontrolowanych warunkach oraz umożliwia analizę zachowań i percepcji użytkowników aplikacji nawigacyjnych bez

konieczności wdrażania kosztownych rozwiązań technologicznych w rzeczywistych budynkach i wprowadzania utrudnień w pracy osób w nich przebywających. Dodatkowo w środowisku symulacyjnym można przeprowadzać eksperymenty, których sprawdzenie nie byłoby możliwe lub bardzo ograniczone w rzeczywistości ze względów bezpieczeństwa (np. ewakuacja w czasie pożaru). Środowiska wirtualne z powodzeniem były już wykorzystywane w pojedynczych badaniach poszczególnych funkcji aplikacji, a ich autorzy zgodnie zauważają potencjał tej metody i konieczność dalszej pracy nad jej rozwojem (Kinaterer, Ronchi, Gromer, i in. 2014; Kinaterer, Ronchi, Nilsson, i in. 2014; Łobodecki i Gotlib 2022; Stachoń i in. 2025). Opracowanie metodyki badawczej wykorzystującej wirtualne poruszanie się po budynku jest możliwe dzięki zastosowaniu technologii gier komputerowych oraz cyfrowych kopii rzeczywistych budynków (ang. *digital twin*). Dodatkowym atutem tego podejścia jest potencjał do wykorzystania treningu w świecie cyfrowym jako ćwiczenia poprawiającego orientację w przestrzeni (Amedi i in. 2024).

Osoba poruszająca się po budynku rozgląda się, zazwyczaj spoglądając na numery pomieszczeń i domyślając się w którym kierunku należy podążać, aby dojść do wskazanego pomieszczenia. Nie zawsze jednak znany jest numer pomieszczenia. Z doświadczeń osobistych praktycznie każdego z nas (w tym autora), można stwierdzić, że pewne układy korytarzy sprzyjają zagubieniu się i utrudniają dojście do celu, a inne nie. Czy w rozwiązaniu tych problemów może pomóc mapa budynku, a jeżeli tak to w jakich momentach jest najbardziej użyteczna? Odpowiedź na to pytanie jest istotna, ale nie można znaleźć na nią pełnej odpowiedzi w obecnej literaturze naukowej. Przeprowadzona analiza literatury wykazała brak kompleksowych badań nad optymalnym przekazem kartograficznym i interfejsem użytkownika w aplikacjach umożliwiających nawigację w przestrzeniach wewnętrznych, jak i niedostateczne wykorzystanie nowoczesnych środowisk symulacyjnych do testowania i ewaluacji tych rozwiązań w warunkach zbliżonych do rzeczywistych. Rozwój tego obszaru wymaga interdyscyplinarnego podejścia łączącego perspektywę technologiczną, kartograficzną oraz psychologiczną, co jest postulowane w agendach badawczych (H. Huang i in. 2018).

Opracowane w ramach niniejszej rozprawy doktorskiej rozwiązanie ma potencjał, by stać się istotną pomocą podczas projektowania aplikacji nawigacyjnych, dostosowanych do potrzeb użytkowników i specyfiki środowisk zamkniętych. Może wnieść również cenne doświadczenia i wskazówki do rozwoju kartografii wewnątrz budynków.

1.2. Cel, teza pracy oraz pytania badawcze

Podstawowym celem rozprawy jest opracowanie platformy badawczej, która będzie wykorzystywała współczesne technologie symulacyjne oraz cyfrowe modele budynków, co pozwoli na wygodne i wieloaspektowe testowanie różnych metod przekazu kartograficznego oraz różnych funkcjonalności aplikacji nawigacyjnych lub lokalizacyjnych.

Teza postawiona w niniejszej rozprawie doktorskiej brzmi:

Wykorzystanie środowiska wirtualnego umożliwi systematyczne, powtarzalne i kontrolowalne testowanie aplikacji przeznaczonych do nawigacji i lokalizacji wewnątrz budynków, a oparta o symulację metodyka pozwoli na wieloaspektową analizę percepcji użytkowników oraz doskonalenie różnorodnych metod przekazu kartograficznego.

W kontekście przedstawionej tezy postawione zostały następujące pytania badawcze:

- 1) Czy na podstawie testów w środowisku wirtualnym można zidentyfikować wpływ różnych rodzajów wskazówek nawigacyjnych na świadomość sytuacyjną w budynku?
- 2) W jaki sposób różne metody prezentacji informacji kartograficznej wpływają na skuteczność nawigacji wewnątrz budynków?
- 3) Czy środowisko wirtualne umożliwia wykrycie:
 - a. szczególnie trudnych miejsc dla osób poruszających się w budynku wynikających z jego architektury?
 - b. niedostatków analizowanych lub projektowanych aplikacji nawigacyjnych?
- 4) W jakim stopniu środowisko wirtualne pozwala na testowanie sytuacji niemożliwych lub trudnych do odtworzenia w rzeczywistości (np. ewakuacja)?
- 5) Jakie są ograniczenia i potencjalne źródła błędów w tego typu badaniach prowadzonych w środowisku wirtualnym?

W ramach realizacji przedstawionych założeń wyznaczono następujące cele szczegółowe, które precyzują kierunki prowadzonych badań i określają kluczowe etapy pracy nad rozprawą:

- 1) Zaprojektowanie i wdrożenie uniwersalnego środowiska symulacyjnego umożliwiającego realizację zróżnicowanych scenariuszy badawczych.
- 2) Integracja cyfrowych modeli rzeczywistych budynków z środowiskiem wirtualnym w celu umożliwienia przeprowadzania eksperymentów w przestrzeniach odpowiadających rzeczywistym warunkom architektonicznym.
- 3) Przygotowanie interfejsu komunikacyjnego pomiędzy symulatorem, a aplikacją nawigacyjną.
- 4) Opracowanie i wdrożenie systemu automatycznego zbierania danych eksperymentalnych, umożliwiających pomiar czasu realizacji zadań, ścieżek poruszania osób badanych oraz innych wskaźników efektywności nawigacji.
- 5) Stworzenie zestawu scenariuszy badawczych odzwierciedlających typowe oraz nietypowe sytuacje nawigacyjne przeznaczonych do badania konkretnych elementów procesu nawigacji.
- 6) Przeprowadzenie badania na użytkownikach z wykorzystaniem przygotowanego systemu badawczego.
- 7) Przeprowadzenie kompleksowych analiz danych zebranych dla każdego ze scenariuszy, zarówno metodami jakościowymi jak i ilościowymi.
- 8) Zidentyfikowanie kluczowych ograniczeń zaproponowanej metodyki, a także określenie priorytetowych obszarów wymagających dalszej optymalizacji i rozwoju w kontekście efektywności środowiska badawczego oraz praktycznego zastosowania wyników.
- 9) Sformułowanie wytycznych dla twórców aplikacji przeznaczonych do nawigacji we wnętrzach budynków, na podstawie wniosków uzyskanych z przeprowadzonego badania.

1.3. Struktura rozprawy

Rozprawa została podzielona na jedenaście rozdziałów, z których rozdziały 2–4 tworzą część teoretyczną, obejmującą pogłębiony przegląd literatury oraz analizę aktualnego stanu wiedzy i technologii w zakresie pozycjonowania i nawigacji wewnątrz budynków. W rozdziałach 5–6 szczegółowo omówiono zaproponowaną metodykę badawczą oraz sposób budowy platformy badawczej, natomiast rozdziały 7–9 poświęcono opisowi przeprowadzonych badań, w tym procedury testowania metodyki oraz analizie uzyskanych wyników i sformułowanych na ich podstawie wniosków. Poniżej przedstawiono zawartość poszczególnych rozdziałów:

Rozdział 2 zawiera analizę istniejących rozwiązań w zakresie nawigacji i lokalizacji wewnątrzbudynkowej. W pierwszej części omówiono podstawy kartografii wnętrz budynków. Następnie przedstawiono przegląd technologii oraz systemów pozycjonowania wykorzystywanych w środowiskach zamkniętych, wraz z ich ograniczeniami dokładnościowymi i infrastrukturalnymi. Rozdział kończy analiza rynku aplikacji przeznaczonych do nawigacji wewnątrz budynków.

Rozdział 3 poświęcony jest przeglądowi metod testowania i badania map oraz aplikacji mapowych. Opisano w nim podejścia oparte na metodach ankietowych i analizie opinii użytkowników, badania wykorzystujące śledzenie ruchu gałek ocznych, a także testy terenowe prowadzone w warunkach rzeczywistych. Przedstawiono zalety i ograniczenia poszczególnych metod. Na tym tle zarysowano potencjał środowisk wirtualnych jako alternatywy pozwalającej na powtarzalne eksperymenty i precyzyjną kontrolę warunków badania.

Rozdział 4 omawia zastosowanie symulatorów i środowisk wirtualnych w badaniach naukowych, ze szczególnym uwzględnieniem koncepcji cyfrowych bliźniaków (ang. *Digital Twins*). Zaprezentowano przegląd badań kartograficznych i nawigacyjnych wykorzystujących symulatory oraz środowiska wirtualne, w tym prace poświęcone symulacji ewakuacji, treningom w VR oraz wcześniejszym prototypom symulatorów poruszania się po wnętrzach budynków.

Rozdział 5 prezentuje metodykę badawczą oraz koncepcję proponowanego narzędzia badawczego. Szczegółowo opisano założenia główne metodyki, w tym wykorzystanie środowiska wirtualnego do realizacji zróżnicowanych scenariuszy nawigacyjnych oraz

integrację cyfrowych modeli rzeczywistych budynków. Przedstawiono strukturę procesu badawczego, opisano zmienne niezależne kontrolowane w procesie badawczym oraz rodzaje danych zbieranych podczas eksperymentów.

Rozdział 6 skupia się na technicznych aspektach implementacji symulatora badawczego. Omówiono wybór silnika gier Unreal Engine jako podstawy platformy badawczej oraz architekturę informatyczną systemu, w tym integrację z infrastrukturą chmurową CENAGIS i wykorzystanie akceleratorów GPU. Istotną część rozdziału stanowi opis komunikacji między aplikacją nawigacyjną, a symulatorem oraz mechanizmu rejestracji danych badawczych.

Rozdział 7 opisuje badanie na użytkownikach przeprowadzone z wykorzystaniem opracowanego symulatora. Przedstawiono cele badania, charakterystykę grupy badawczej oraz zastosowane narzędzia analizy danych. Rozdział zawiera szczegółowy opis przeprowadzonych eksperymentów (siedem eksperymentów obejmujących łącznie szesnaście scenariuszy badawczych). W każdym z eksperymentów przedstawiono założenia, zastosowane warunki, opis przebiegu oraz wyniki ilościowe.

Rozdział 8 poświęcony jest ewaluacji opracowanego narzędzia badawczego. Oceniono w nim działanie symulatora oraz zintegrowanej aplikacji nawigacyjnej zarówno z punktu widzenia parametrów technicznych (wydajność, stabilność, skalowalność), jak i użyteczności w kontekście planowanych badań. Uwzględniono wyniki przeprowadzonego badania oraz wnioski z ankiet jakościowych, w których uczestnicy oceniali m.in. działanie symulatora, czytelność map wewnątrz budynków i zgłaszali sugestie ulepszeń. Na tej podstawie sformułowano rekomendacje dotyczące dalszej rozbudowy środowiska badawczego i potencjalnych obszarów jego zastosowania.

Rozdział 9 zawiera podsumowanie oraz wnioski końcowe zrealizowanego projektu badawczego. Sformułowano kluczowe wnioski dotyczące wpływu różnych metod prezentacji informacji kartograficznej i wskazówek nawigacyjnych na efektywność nawigacji oraz percepcję użytkowników, a także omówiono ograniczenia zaproponowanej metodyki i wskazano kierunki dalszych badań, w tym rozwój rozwiązań wykorzystujących VR oraz rozszerzenie zakresu analiz na inne typy budynków i grup użytkowników.

2. Analiza istniejących rozwiązań w zakresie nawigacji i lokalizacji wewnątrzbudynkowej

2.1. Kartografia wnętrz budynków

Rozpoczynając analizę zagadnień związanych z kartografią wnętrz budynków i aplikacjami LBS, warto najpierw przyjrzeć się definicji mapy. Naturalnie to pojęcie zmienia się wraz z rozwojem nauki i technologii, dlatego zrozumienie jego ewolucji pozwala lepiej uchwycić współczesne podejście do przedstawiania przestrzeni.

Tradycyjne rozumienie mapy koncentruje się na jej funkcji jako graficznej reprezentacji wybranych elementów rzeczywistości geograficznej (Kraak i Fabrikant 2017). Mapa była postrzegana przede wszystkim jako odwzorowanie przestrzeni – narzędzie do przedstawiania rozmieszczenia obiektów, ich wzajemnych relacji. Klasyczne definicje, takie jak te przyjęte przez Międzynarodową Asocjację Kartograficzną (ICA)¹, podkreślają, że mapa jest symbolizowanym obrazem rzeczywistości geograficznej, w którym istotne są relacje przestrzenne i zależności między przedstawianymi elementami.

Współczesne dyskusje naukowe rozszerzają jednak dotychczasowe interpretacje, wskazując, że mapa nie musi być ograniczona jedynie do odwzorowania przestrzeni geograficznej. Coraz częściej akcentuje się rolę mapy jako narzędzia do modelowania i przedstawiania różnorodnych danych – nie tylko przestrzennych, ale także tematycznych czy nawet abstrakcyjnych. Przykładem mogą być mapy wykorzystywane w medycynie do modelowania ciała ludzkiego lub w naukach społecznych do wizualizacji powiązań i relacji. W literaturze naukowej mapa bywa definiowana jako intelektualna konstrukcja służąca komunikacji i analizie informacji, niezależnie od jej bezpośredniego związku z przestrzenią geograficzną. Oczywiście warto przy tym zauważyć, że pojęcia mapa i mapowanie nie są związane tylko z dziedziną kartografii. Mapa i mapowanie to terminy znacznie szersze (mapa myśli, mapowanie atrybutów, mapowanie procesów, mapowanie pamięci), choć prawdopodobnie najczęściej

¹ International Cartographic Association – <https://icaci.org/mission/> [dostęp: 10.01.2026]

kojarzone z kartografią. Najogólniej zdefiniowane są prawdopodobnie w matematyce, gdzie pod pojęciem mapowania rozumie się przypisanie elementów jednego zbioru do elementów drugiego zbioru. W tym kontekście warto przeanalizować współczesne spojrzenie na cechy modelowania kartograficznego, jako jednego z wielu sposobów modelowania rzeczywistości (Baranowski, Gotlib, i Olszewski 2016).

Znaczenie tego zagadnienia dla niniejszej rozprawy wynika z faktu, iż w ramach prowadzonych badań zastosowano zarówno tradycyjną, dwuwymiarową mapę (prezentowaną w nowoczesnej, interaktywnej formie z wykorzystaniem profesjonalnej bazy danych) jak i trójwymiarowe modele budynków, które stanowią specyficzną i stosunkowo nową formę mapowania rzeczywistości. W opracowanym symulatorze modele 3D pełnią rolę nie tylko przekazywania informacji o przestrzeni, ale przede wszystkim stanowią podstawę przestrzeni wirtualnej, tak by na możliwie najwyższym poziomie immersji przenieść uczestnika badania we wskazane miejsce.

Dlatego warto traktować mapę jako model przestrzeni, w szczególności przestrzeni geograficznej. Mapa to model rzeczywistości, przedstawiający lokalizację oraz wybrane cechy obiektów i zjawisk w odniesieniu do powierzchni Ziemi, innego ciała niebieskiego lub innych obiektów, np. budynku, wnętrza ciała ludzkiego, a także wzajemne relacje przestrzenne między obiektami i zjawiskami (Gotlib 2011).

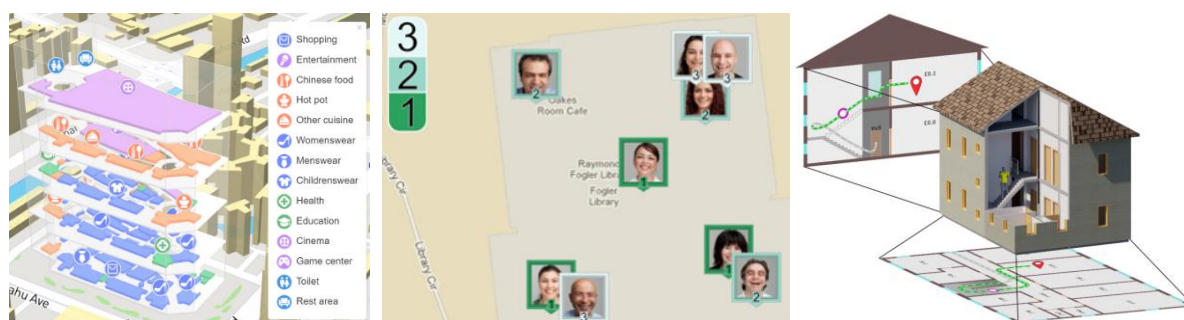
Kartografia z kolei jest najczęściej rozumiana, jako sztuka, nauka i technologia tworzenia i używania map (Kraak i Fabrikant 2017). Kartografia wewnątrz budynków stanowi wyspecjalizowaną dziedzinę kartografii, która zajmuje się tworzeniem, przetwarzaniem i prezentacją map przestrzeni zamkniętych, charakteryzujących się obecnością ścian, drzwi i struktur wielopoziomowych (Chen i Clarke 2020).

Przestrzeń wewnętrzna różni się fundamentalnie od zewnętrznej pod wieloma względami. Podstawowym wyzwaniem jest fakt, że przestrzeń ta jest o wiele częściej podzielona na jednostki przestrzenne (takie jak pomieszczenia), gdzie każda jednostka jest otoczona przez fizyczne komponenty architektoniczne lub oddzielona przez wirtualne granice (Zlatanova i in. 2020).

Reprezentacja kartograficzna wnętrz budynków może przyjmować różne formy, od prostych map 2D, poprzez wizualizacje 2,5D, 2D+ lub wizualizacje hybrydowe, aż do zaawansowanych modeli 3D. Geowizualizacja wnętrz powinna zapewnić: wysoką czytelność prezentacji w ruchu na małym ekranie, prezentację budynku na wielu poziomach szczegółowości, prawidłową wizualizację trasy oraz pozycji użytkownika na tle topografii budynku (Gotlib, Wyszomirski, i Gnat 2020).

W kartografii wnętrz jest również istotne odpowiednie zarządzanie skalą i poziomem szczegółowości. W zależności od rozmiaru budynku, plany pięter są zazwyczaj wyświetlane w skalach między 1:200, a 1:50 (Brumana i in. 2019). Ponadto precyzyjne oddanie szczegółowości układu pomieszczeń skomplikowanego budynku wymaga przesyłania dużej ilości danych, a ich renderowanie jest obciążające dla urządzenia klienta (Richter i in. 2022).

Wizualizacja struktur wielopoziomowych to kolejne wyzwanie w kartografii wnętrz budynków. Nawigacja w wielopiętrowych budynkach sprawia większe trudności w utrzymaniu orientacji przestrzennej i w tworzeniu map poznawczych (Nossum, Li, i Giudice 2013). Interesujące i trudne zadanie dla kartografów stanowi niemożliwość wyświetlania wszystkich pięter jednocześnie w widoku 2D (Gotlib, Wyszomirski, i Gnat 2020). Warto zwrócić uwagę na wizualizację w uproszczonym widoku 3D (Xiao i in. 2020), próbę przekazania informacji o piętrze z wykorzystaniem koloru, czy innowacyjną koncepcję mapy 2D+ (Rysunek 2.1).



Rysunek 2.1. a) Uproszczony widok 3D (Xiao i in. 2020)

b) Lokalizacja osób na piętrach oznaczona kolorem ramki (Nossum, Li, i Giudice 2013)

c) Koncepcja mapy 2D+ dodająca przekrój pionowy do prezentacji kartograficznej (Gotlib, Wyszomirski, i Gnat 2020)

Kolejnym zagadnieniem jest kwestia dostępu do danych lokalizacyjnych. Z uwagi na powszechność wykorzystania systemów GNSS problem braku znajomości lokalizacji użytkownika właściwe nie występuje w aplikacjach do nawigacji w przestrzeniach zewnętrznych (Ahmed i in. 2025). W aplikacjach przeznaczonych do nawigacji w przestrzeniach

wewnętrznych implementacja integracji z systemami pozycjonowania pozostaje po stronie dewelopera, a precyzja i skuteczność pozycjonowania wciąż nie jest wystarczająca, o czym więcej w kolejnym podrozdziale.

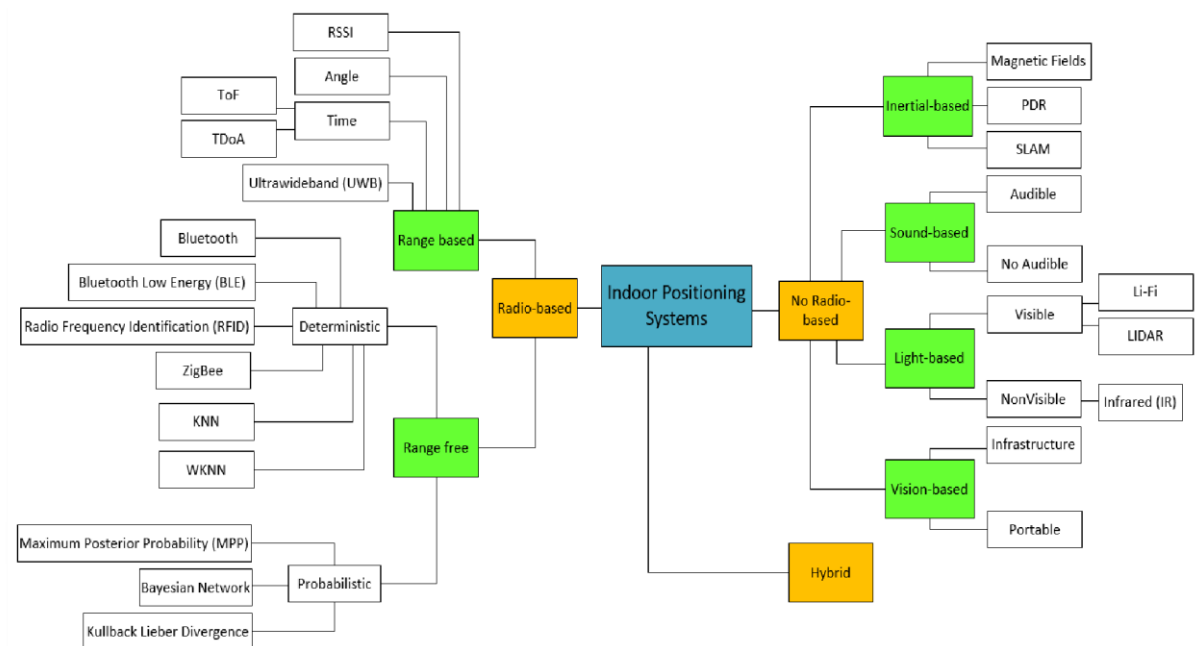
2.2. Technologie i systemy pozycjonowania w środowiskach zamkniętych

Dostarczenie informacji o lokalizacji jest niezwykle istotne z perspektywy wspomagania orientacji użytkownika w przestrzeni i działania aplikacji nawigacyjnych. Warto, więc przyjrzeć się problematyce ustalania lokalizacji wewnątrz budynku, aby lepiej zrozumieć powody napotykaných trudności technicznych.

Rozważając zagadnienie lokalizacji wewnątrz budynków, szczególną uwagę należy poświęcić różnorodności technologii oraz systemów, które mogą być wykorzystane w tym celu. Podstawowy podział systemów pozycjonowania opiera się na rozróżnieniu między rozwiązaniami bazującymi na falach radiowych (np. Wi-Fi, Bluetooth, RFID, UWB), a systemami wykorzystującymi inne sensory, takie jak sensory inercyjne, światło, dźwięk bądź wizję komputerową (Rysunek 2.2). Coraz częściej pojawiają się rozwiązania hybrydowe, integrujące wiele modalności sensorycznych, co pozwala na zwiększenie elastyczności i odporności systemów (Simões i in. 2020b).

Kluczowe parametry oceniane w literaturze to przede wszystkim:

- **Dokładność** (ang. *accuracy*) – rozumiana jako średni błąd lokalizacyjny użytkownika/obiektu w przestrzeni.
- **Skalowalność** – zdolność systemu do efektywnego działania w złożonych lub dużych obiektach i obsługi dużej liczby użytkowników.
- **Koszt wdrożenia** – zarówno infrastrukturalny, jak i użytkowy.



Rysunek 2.2. Klasyfikacja systemów pozycjonowania w przestrzeniach zamkniętych (Simões i in. 2020b)

Ze względu na zróżnicowanie środowisk wewnętrznych, te kryteria rzadko są spełniane równocześnie na wysokim poziomie, co oznacza konieczność szukania rozwiązań kompromisowych na etapie projektowania (Sakpere, Adeyeye Oshin, i Mlitwa 2017).

W literaturze konsekwentnie opisuje się trend polegający na praktycznym wykorzystaniu czujników i modułów komunikacyjnych, będących standardową funkcjonalnością współczesnych smartfonów (Wi-Fi, Bluetooth, BLE, IMU, kamery) (Liu i in. 2012). Takie podejście minimalizuje koszty wdrożenia – eliminuje konieczność angażowania kosztownej infrastruktury specjalistycznej. Ponadto, powszechność smartfonów zwiększa potencjał adaptacyjny technologii w różnych populacjach użytkowników i dla budynków różnego typu.

Podkreśla się również, że systemy hybrydowe, łączące wiele technologii pozycjonowania, oferują największy potencjał w zakresie precyzji i ciągłości nawigowania (Marciniak 2018). Takie podejście pozwala na wzajemne kompensowanie ograniczeń poszczególnych technologii – np. łączenie sygnatur radiowych (ang. *fingerprint*) zmierzonych wcześniej w budynku, z danymi inercyjnymi czy obrazowaniem wizualnym może znacząco ograniczyć błędy dryfu oraz poprawić ciągłość śledzenia użytkownika (Hailu i in. 2024).

Rozwiązania hybrydowe zwiększają niezawodność i odporność systemu na awarie pojedynczych sensorów. Kluczowe dla ich wdrożenia są strategie fuzji danych i dynamiczne

łączenie źródeł informacji, z czym wiąże się wysoka złożoność obliczeniowa i potrzeba wydajnego przetwarzania danych w czasie rzeczywistym (Manap i in. 2025). Ciekawym przykładem realizacji tej strategii jest opracowanie rozwiązania niskokosztowego, wykorzystującego metodę PDR (ang. *Pedestrian Dead Reckoning*) opartą o zliczanie kroków i inercyjny pomiar przemieszczeń. Dzięki połączeniu jej z regularnym nawiązywaniem lokalizacji w kluczowych miejscach w budynku możliwe było uzyskanie wystarczającej dokładności w testowanym obszarze budynku (Y.-S. Li i Ning 2018).

Systemy wymagające rozbudowanej infrastruktury (czytniki RFID, beacons BLE na dużą skalę, specjalistyczne nadajniki i sensory UWB, sieci LIDAR) są w stanie zapewnić bardzo wysoką precyzję, często poniżej 1 m (Sartayeva i in. 2023). Jednakże podkreśla się znaczące bariery wejścia – koszty zakupu, instalacji oraz utrzymania infrastruktury, jak również problemy związane z czasochłonną kalibracją i konserwacją. Takie rozwiązania sprawdzają się w specjalistycznych lub przemysłowych zastosowaniach, natomiast ich masowe wdrożenie w miejscach publicznych jest zazwyczaj nieopłacalne ekonomicznie (Basiri i in. 2017).

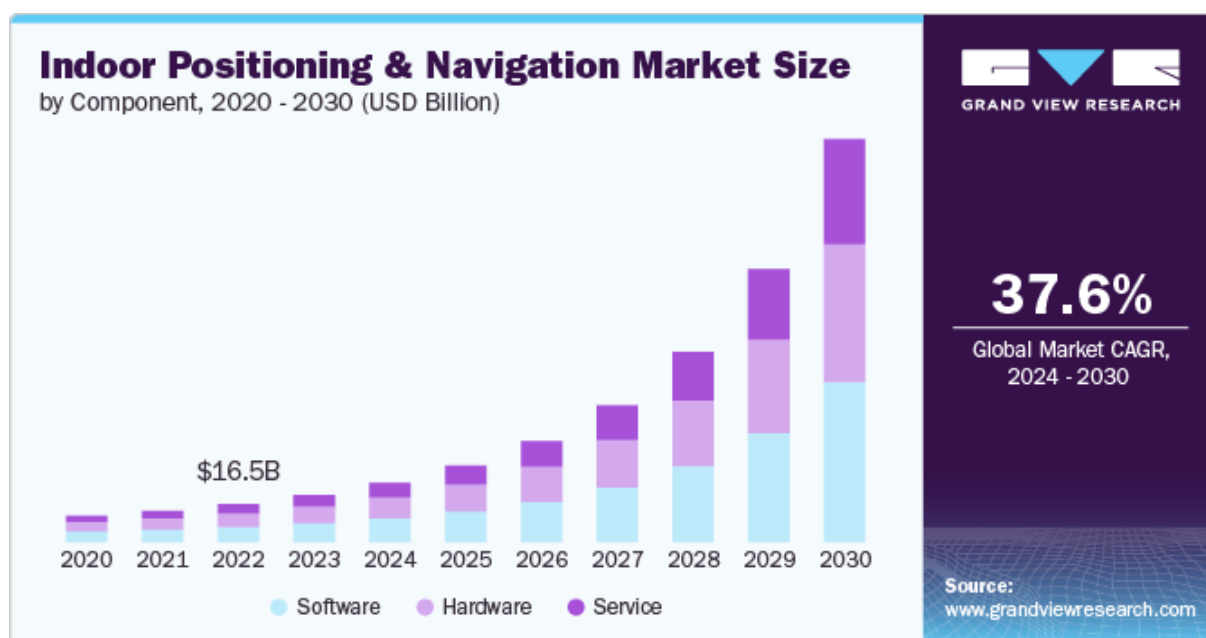
W ostatnich przeglądach literatury badacze zgodnie wskazują, że nie istnieje jeszcze system IPS spełniający wszystkie wymagania dotyczące dokładności, natychmiastowości, dostępności i kosztów implementacji w pełnym spektrum scenariuszy użytkowania. Dodatkowo podkreślane są wyzwania skalowalności, prywatności (związane z rejestracją i przetwarzaniem danych lokalizacyjnych), energochłonności, konserwacji (np. wymiana baterii beaconów BLE), oraz problemów z aktualizacją baz danych sygnatur radiowych w dynamicznie zmieniającym się środowisku wewnętrznym (Hailu i in. 2024; J i in. 2022; Manap i in. 2025; Sartayeva i in. 2023).

Omawiane wyzwania są wciąż przedmiotem aktywnej, interdyscyplinarnej debaty naukowej. Dla niniejszej rozprawy kluczowy jest jednak wniosek, iż wciąż nie dysponujemy technologią wystarczająco dokładną, a jednocześnie na tyle tanią do wdrożenia i powszechną, aby móc ją uznać za ogólnodostępną. Tym samym istnieją obiektywne trudności i bardzo duże ograniczenia w testowaniu aplikacji do nawigacji wewnątrz budynków. Jednocześnie mnogość badań i nieustanny postęp w uzyskiwanych wynikach nowych rozwiązań i algorytmów sugeruje, że w przyszłości możemy spodziewać się ukazania się rozwiązań spełniających wszystkie kryteria. By uprzedzić ten fakt badania nad sposobem prezentacji kartograficznej i metod

przekazu informacji nawigacyjnej można rozpocząć już teraz, wykorzystując środowiska wirtualne, takie jak proponowane w tej pracy.

2.3. Analiza rynku aplikacji przeznaczonych do nawigacji wewnątrz budynków

Podczas gdy nawigacja w przestrzeniach zewnętrznych oparta na systemach GNSS osiągnęła wysoką dojrzałość technologiczną, mnogość aplikacji i powszechność wykorzystania, systemy pozycjonowania wewnętrznego nadal borykają się z fundamentalnymi wyzwaniami technicznymi. Mimo to prognozy dla rynku pozycjonowania wewnętrznego są bardzo optymistyczne. Globalne badania wskazują na dynamiczny rozwój i wzrost ogólnej wartości rynku IPS z 20,38 miliarda USD w 2023 roku, do 174,02 miliarda USD w roku 2030. Daje to przewidywaną roczną stopę wzrostu na poziomie 37,6% (Grand Review Research 2024). Jednakże znaczna część tego rynku koncentruje się na rozwiązaniach B2B i infrastrukturalnych, podczas gdy aplikacje mobilne dostępne dla końcowych użytkowników stanowią relatywnie niewielką część tej wartości.



Rysunek 2.3. Prognozowany rozmiar rynku pozycjonowania i nawigacji wewnątrz budynków w latach 2020-2030 (Grand Review Research 2024)

Choć rozwiązania dla użytkowników końcowych nie są jeszcze bardzo powszechne analiza istniejących implementacji pozwala na wyodrębnienie kilku charakterystycznych kategorii aplikacji do nawigacji wewnątrz budynków, różniących się przeznaczeniem, technologią oraz grupami docelowymi.

Poniżej przedstawiono krótką charakterystykę tych kategorii w oparciu o analizę H. Świecha (Świech 2024).

Aplikacje dla instytucji szkolnictwa wyższego

Uczelnie wyższe stanowią jeden z najprężniej rozwijających się segmentów rynku aplikacji do nawigacji wewnątrz budynków. Specyfika środowiska akademickiego, charakteryzująca się wysoką liczbą nowych użytkowników (studenci pierwszego roku), rozległymi kampusami oraz kompleksowymi układami budynków, tworzy naturalne zapotrzebowanie na rozwiązania nawigacyjne. Polskie uczelnie wykazują szczególną aktywność w tym obszarze, rozwijając aplikacje takie jak PW Navi (Politechnika Warszawska), Navi AGH (Akademia Górniczo-Hutnicza), naviPŁ (Politechnika Łódzka) czy UAM GO (Uniwersytet im. Adama Mickiewicza). Aplikacje uniwersyteckie charakteryzują się integracją z systemami rezerwacji sal, kalendarzami akademickimi oraz funkcjonalnościami umożliwiającymi nawigację między budynkami w ramach kampusu.

Badania porównawcze skuteczności systemów nawigacyjnych przeprowadzone na Norweskim kampusie Uniwersytetu Nauki i Technologii w Trondheim wykazały, że dla aplikacji wyznaczających trasę w obrębie kampusu krytyczna jest integracja pracy w zarówno w przestrzeni zamkniętej jak i otwartej (Tahir i Krogstie 2023).

Aplikacje dla portów lotniczych

Lotniska, ze względu na swoją złożoną i dynamiczną strukturę, stanowią jedno z kluczowych środowisk, w których systemy nawigacji w przestrzeni zamkniętej znajdują praktyczne zastosowanie, przyczyniając się do redukcji stresu pasażerów i optymalizacji operacyjnej. Wdrożenia te często polegają na hybrydowym wykorzystaniu technologii, takich jak Wi-Fi, beaconów BLE oraz coraz częściej, rozszerzonej rzeczywistości (AR). Te aplikacje oferują podróżnym nie tylko wyznaczanie trasy do bramek, punktów usługowych czy sklepów, ale

również integrację z informacjami o lotach w czasie rzeczywistym, co podnosi ogólną jakość doświadczeń pasażerskich i efektywność zarządzania portem lotniczym.

Dobrym przykładem jest hiszpańska sieć lotnisk Aena, we współpracy z firmą Situm, która pozycjonuje się na jednego ze światowych liderów we wdrażaniu rozwiązań nawigacyjnych dla pasażerów². Precyzja lokalizacji deklarowana przez producenta wynosi „mniej niż 5 metrów”. Zarówno literatura naukowa (Marciniak 2018) jak i indywidualne testy autora rozprawy potwierdzają, że w sytuacji faktycznego używania nawigacji taka dokładność nie jest wystarczająca.

Aplikacje dla centrów handlowych

Rozwiązania technologiczne w centrach handlowych często opierają się na hybrydowych systemach łączących interaktywne kioski z aplikacjami mobilnymi. Zazwyczaj możemy spotkać pozycjonowanie w oparciu o Wi-Fi, które stanowi dominującą technologię w tym segmencie ze względu na stosunkowo niskie koszty implementacji i możliwość wykorzystania istniejącej infrastruktury. Aplikacje takie jak systemy wdrożone w galeriach Westfield Arkadia czy Manufaktura w Łodzi demonstrują możliwości tej technologii na smartfonach, ale większość galerii w Polsce ogranicza się do kiosków informacyjnych, czyli ekranów dotykowych umożliwiających odnalezienie sklepu.

Często podnoszonym tematem jest kwestia świadomego organizowania przestrzeni w centrum handlowym, tak aby utrudnić nawigację i zatrzymać klienta w centrum handlowym na dłużej (czego przeciwieństwem są rozwiązania ułatwiające nawigację). Choć strategia ta może być skuteczna w krótkim terminie, badania wskazują, że wygoda i łatwość poruszania się są ważne dla pozytywnych doświadczeń zakupowych i istotne dla lojalności klientów (Gilboa, Vilnai-Yavetz, i Mitchell 2024).

Rozwiązania komercyjne

Segment komercyjnych aplikacji wieloplatformowych reprezentują zaawansowane technologicznie rozwiązania opracowane przez specjalistyczne firmy technologiczne.

² Witryna internetowa rozwiązania firmy Situm przeznaczonego do nawigacji po lotniskach: <https://situm.com/en/solutions/airport-indoor-navigation/> [dostęp: 10.01.2026]

Platformy takie jak MazeMap³, Mapsindoors⁴, Situm Wayfinding⁵, ArcGIS Indoors⁶ czy Mappedin⁷ charakteryzują się wysoką dojrzałością technologiczną i możliwością wdrożenia w różnorodnych środowiskach. Zazwyczaj systemy te wykorzystują algorytmy pozycjonowania integrujące dane z wielu źródeł próbując uzyskać najlepsze możliwe rozwiązanie w dostępnych ramach, opisanych w poprzednim podrozdziale.

Często oprócz kompletnego wdrożenia oferowane jest udostępnienie SDK, które umożliwia tworzenie nowych aplikacji w oparciu o technologię danej firmy.

Aplikacje gigantów technologicznych

Największe firmy technologiczne, takie jak Google, Apple czy Microsoft, oferują ograniczone rozwiązania do nawigacji wewnątrz budynków jako rozszerzenie swoich platform mapowych. Google Maps udostępnia funkcjonalność map wybranych budynków, umożliwiając przeglądanie planów pięter i podstawową nawigację w przestrzeniach takich jak lotniska, centra handlowe czy dworce. Rozwiązania te charakteryzują się ograniczoną dostępnością i brakiem precyzyjnego pozycjonowania wewnątrz budynku, co znacząco ogranicza ich praktyczność.

Aplikacje dla osób z niepełnosprawnościami

Aplikacje przeznaczone osobom z niepełnosprawnościami stanowią wysoko wyspecjalizowany segment rynku, charakteryzujący się szczególnymi wymaganiami. Aplikacje takie muszą cechować się ponadprzeciętną dokładnością pozycjonowania, a także maksymalną pewnością wyznaczonej trasy czy pozycji. Funkcjonalne rozwiązania byłyby wyjątkową pomocą dla osób z niepełnosprawnościami, a szczególnie osób z dysfunkcją wzroku. Można znaleźć wiele opracowań dopasowanych do potrzeb osób niedowidzących, rozwiązań do lokalizowania, nawigowania czy wykrywania przeszkód (Bai i in. 2018; Kubota i in. 2024; B. Li i in. 2019; Simões i in. 2020b) i sposobów przekazu informacji nawigacyjnej (Khan, Nazir, i Khan 2021), a nawet prototyp robota prowadzącego do celu po korytarzach budynku (Kuribayashi i in. 2023).

³ <https://www.mazemap.com/> [dostęp: 10.01.2026]

⁴ <https://www.mapspeople.com/mapsindoors> [dostęp: 10.01.2026]

⁵ <https://situm.com/en/technology/indoor-positioning-system/> [dostęp: 10.01.2026]

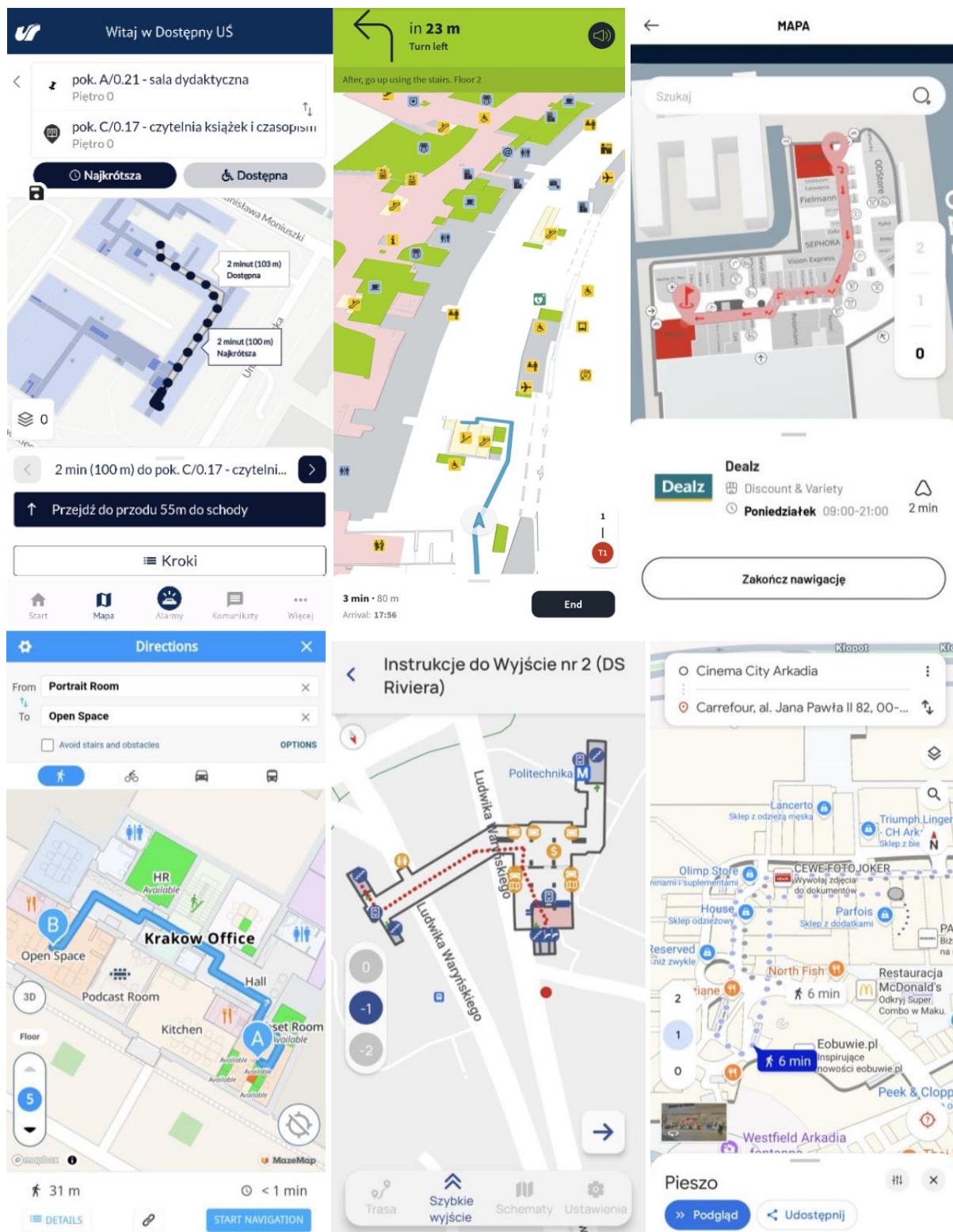
⁶ <https://www.esri.com/pl-pl/arcgis/products/arcgis-indoors/overview> [dostęp: 10.01.2026]

⁷ <https://www.mappedin.com/> [dostęp: 10.01.2026]

Innowacyjne aplikacje z elementami 3D i AR

Innowacje, jak aplikacja Insiteo wykorzystująca modele 3D i AR, ułatwiają orientację poprzez wirtualne porównanie z rzeczywistością. Literatura wskazuje, że technologie VR/AR reprezentują kolejny etap rozwoju, oferując przewagi nad 2D, takie jak lepsza immersja, choć wciąż są one w fazie prototypowej (B.-C. Huang i in. 2020). Mapa 2D z kolei zyskuje dzięki dotychczasowym przyzwyczajeniom użytkowników, dlatego najlepszym rozwiązaniem może być łączenie rozwiązań nowatorskich z klasycznymi (Gotlib, Wyszomirski, i Gnat 2020).

Przykłady wyglądu aplikacji z różnych kategorii przedstawiono na rysunku 2.4.



Rysunek 2.4. Zrzuty ekranów przykładowych aplikacji z różnych kategorii. Kolejno: Dostępny UŚ, Aena, Galeria Wileńska w Warszawie, MazeMap, GdzieToMetro, Google Maps nawigacja wewnątrz budynku Westfield Arkadia (na podstawie Świech 2024)

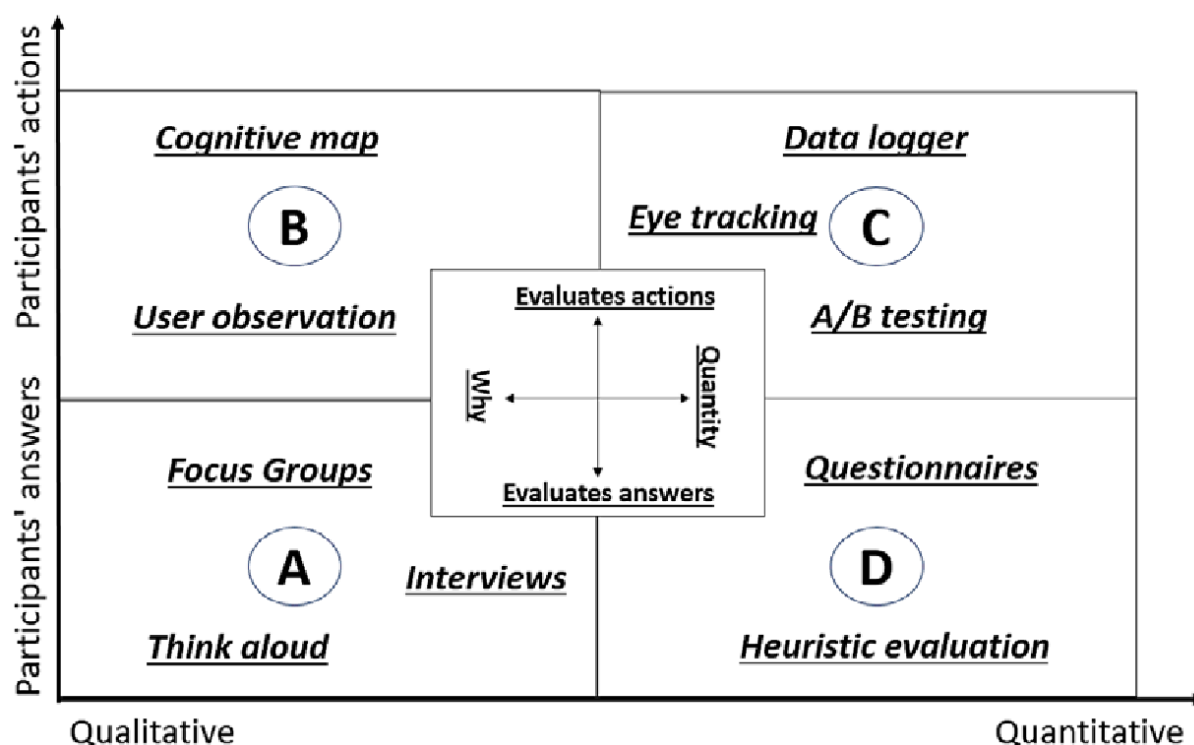
Podsumowując, choć rynek aplikacji nawigujących po przestrzeniach zamkniętych nie jest jeszcze w pełni dojrzały, obserwuje się wzrost zainteresowania, napędzany przez zarówno przez środowisko badawcze jak i komercyjne. Wyzwania, takie jak precyzja lokalizowania czy koszty wdrożenia i utrzymania infrastruktury lokalizującej, pozostają aktualne, ale prognozy dla tego rynku są bardzo optymistyczne. Rozwój systemów pozycjonowania wewnątrz budynków nie tylko może rozwiązać problemy praktyczne końcowych użytkowników, ale także przyczynić się do usprawnienia procesów logistycznych czy biznesowych.

3. Przegląd metod testowania i badania map oraz aplikacji mapowych

Niniejszy przegląd przedstawia metody testowania i oceny map oraz aplikacji mapowych spotykane w literaturze, ze szczególnym uwzględnieniem ich zastosowania w porównaniu do proponowanej metody symulacyjnej.

Podział metod ewaluacji map

Jedną z klasyfikacji metod ewaluacji map przedstawiono na rysunku 3.1 (Martins i in. 2024). W tym podejściu autorzy wyróżniają cztery kategorie metodologiczne różnicujące się charakterem zbieranych danych oraz źródeł informacji o zachowaniach użytkowników.



Rysunek 3.1. Podział metod ewaluacji map (Martins i in. 2024)

Dwuwymiarowy system kategoryzacji opiera się na dwóch osiach: rozróżnieniu metod jakościowych od ilościowych oraz rozróżnieniu technik analizujących działania użytkowników od tych koncentrujących się na ich odpowiedziach.

Kategoria A reprezentuje podejścia badawcze charakteryzujące się pogłębioną analizą jakościową odpowiedzi uczestników badań. Dominującymi metodami w tej kategorii są

wywiady fokusowe, które umożliwiają eksplorację kolektywnych postaw i percepcji grupy użytkowników w kontrolowanym środowisku. Metoda ta pozwala na identyfikację nie tylko indywidualnych preferencji, ale także dynamiki grupowej i wzajemnego wpływu uczestników na kształtowanie opinii. Wyróżnia się niewielką liczbą uczestników przy szczegółowej analizie każdej odpowiedzi, pozwalając na głębokie zrozumienie motywacji użytkowników.

Kategoria B obejmuje techniki badawcze koncentrujące się na bezpośredniej obserwacji zachowań użytkowników, przy wykorzystaniu podejścia jakościowego do analizy zebranych danych. Koncentruje się na bezpośredniej obserwacji zachowań przez mapy kognitywne umożliwiające wizualizację sposobu, w jaki użytkownicy organizują informacje przestrzenne w kontekście nawigacji oraz obserwację użytkowników w naturalnym środowisku. Każde zachowanie podlega szczegółowej analizie jakościowej celem identyfikacji wzorców behawioralnych i zrozumienia kontekstu działań.

Kategoria C reprezentuje podejścia metodologiczne wykorzystujące zaawansowane techniki rejestracji i analizy ilościowej zachowań użytkowników. Śledzenie ruchu gałek ocznych dla analizy wzorców uwagi wizualnej podczas interakcji z interfejsem lub przeglądania mapy, testy A/B umożliwiające porównanie różnych wariantów oraz automatyczne zbieranie danych o wzorcach użytkowania. Charakterystyczną cechą metod z kategorii C jest analiza faktycznie zarejestrowanych danych behawioralnych od znacznej liczby uczestników, co umożliwia identyfikację statystycznie istotnych wzorców oraz uogólnienie wyników na szerszą populację użytkowników.

Kategoria D to przede wszystkim kwestionariusze do standaryzowanego zbierania opinii. Zbierane są informacje o preferencjach, satysfakcji oraz postrzeganej użyteczności systemu. Stosowanie skal psychometrycznych pozwala na kwantyfikację subiektywnych ocen użytkowników oraz porównywanie wyników między różnymi grupami badawczymi. Znaczna liczba respondentów pozwala na zastosowanie zaawansowanych technik analizy statystycznej oraz zwiększa wiarygodność uogólnień na całą populację docelową.

Przedstawiona klasyfikacja posiada szczególne znaczenie dla projektowania i oceny interfejsów aplikacji nawigacyjnych. Metody jakościowe pozwalają na zrozumienie procesów poznawczych

związanych z interpretacją informacji przestrzennych, podczas gdy podejścia ilościowe umożliwiają obiektywną ocenę skuteczności rozwiązań projektowych.

Utworzony w ramach niniejszej rozprawy symulator badawczy ma potencjał do wykorzystania w każdym z opisanych sposobów ewaluacji map. W ramach pracy wykorzystano go do badań w kategorii C tj. w poprzez prowadzenie rozbudowanych testów A/B i zbieranie szczegółowych danych o trasie przebytej przez uczestnika, a także w kategorii A, poprzez notowanie swobodnej wypowiedzi na temat percepcji używanej mapy.

Szczegóły wykorzystanych metod opisano w rozdziale 5.

Liczba uczestników w badaniach jakości map

W przeprowadzonej przez Martinsa i in. (Bergmann Martins i in. 2023) kolejnej analizie, bazując na tym samym podziale metod ewaluacji map na 4 grupy, ustalono najczęściej występującą w badaniach wielkość grupy badawczej. Co istotne, żadne spośród 85 przeanalizowanych badań nie zostało przypisane do kategorii C, co potwierdza potrzebę badań takich jak przedstawione w niniejszej rozprawie, które koncentrują się właśnie wokół tej kategorii.

W Tabeli 1 przedstawiono zakresy liczby uczestników, którzy wzięli udział w badaniach w każdej z kategorii oraz uzyskaną na podstawie ich analizy rekomendowaną wielkość grupy badawczej.

Tabela 1. Liczba uczestników w badaniach map (na podstawie (Bergmann Martins i in. 2023))

Typ badania	Liczba uczestników	Rekomendowana liczba
Kategoria A	8–40	20–30
Kategoria B	25–65	30–40
Kategoria C	–	–
Kategoria D	5–250	≥50, najlepiej ~100

W przeprowadzonym w ramach pracy badaniu na użytkownikach wzięło udział

205 uczestników, z czego 127 ukończyło je w 100%.

Dzięki temu można uznać tę próbę badawczą za rozbudowaną na tle innych badań oraz w zupełności wystarczającą do wyciągania wniosków ilościowych.

Charakterystykę grupy badawczej opisano w rozdziale 7.4.

W kolejnych podrozdziałach omówiono wybrane badania przeprowadzone w celu ewaluacji map cyfrowych i aplikacji kartograficznych, zwracając uwagę na różnorodność podejść i aspektów badawczych.

3.1. Badania oparte o metody ankietowe i analizę opinii użytkowników

Świetnym przykładem badania przedstawiającego zastosowanie ankiety ewaluacyjnej w praktyce jest artykuł opisujący system nawigacji po kampusie bazujący na technologii ARCore, który wykorzystuje technologię rozszerzonej rzeczywistości (AR) do wyświetlania trójwymiarowych informacji na obrazie z kamery. W badaniu wzięło udział 120 uczestników podzielonych na 4 grupy, uwzględniając m.in. zastosowanie wewnątrz budynku oraz na otwartej przestrzeni. Ewaluację przygotowanego rozwiązania nawigacyjnego przeprowadzono na podstawie kwestionariusza obejmującego pytania obiektywne, pozwalające porównać system z alternatywnymi metodami zdobywania informacji o kampusie, oraz pytania subiektywne zachęcające uczestników do przedstawiania własnych sugestii, zalet i ograniczeń dotyczących tego systemu (Lu i in. 2021). Porównanie statystycznej różnicy w odpowiedziach pomiędzy uczestnikami różnych grup pozwala wyciągać wnioski o szczególnie istotnych aspektach testowanego rozwiązania.

Innowacyjnym rozszerzeniem standardowego rozumienia pojęcia ankiety jest narzędzie Geo-Survey (Kulawiak i in. 2023), które umożliwia prowadzenie zindywidualizowanych geoankiet dotyczących orientacji przestrzennej. Narzędzie to wprowadza interaktywny sposób odpowiadania poprzez oznaczanie odpowiedzi bezpośrednio na mapie, co pozwala na bardziej

precyzyjne zbieranie danych oraz automatyczną analizę wyników. Testy przeprowadzono na międzynarodowej grupie studentów informatyki, łącząc wywiady z ankietą online w celu oceny jakości różnych typów map cyfrowych. Wyniki badania wykazały, że takie podejście zwiększa zaangażowanie uczestników oraz pozwala na bardziej szczegółową diagnozę problemów z orientacją przestrzenną, a także lepsze zrozumienie indywidualnych potrzeb użytkowników map cyfrowych w odniesieniu do ich funkcjonalności i użyteczności.

Kolejnym ciekawym przykładem badań opartych na wywiadzie z uczestnikami są badania dotyczące potrzeb emocjonalnych, które prowadzono z użyciem aplikacji przeznaczonej dla turystów – analizowano w nich emocje takie jak stres, podekscytowanie czy zrelaksowanie, z naciskiem na rolę stałości i niezawodności dostępu do Internetu podczas korzystania z aplikacji w różnych kontekstach podróży. Autorzy wyodrębnili cztery kluczowe czynniki wpływające na emocjonalne doświadczenie użytkownika mapy cyfrowej: poczucie bezpieczeństwa, zaufanie do funkcjonalności aplikacji, możliwość personalizacji oraz niezawodność działania nawet w warunkach ograniczonego dostępu do sieci. Badania wykazały, że odpowiednia konstrukcja mapy oraz aplikacji turystycznej może znacząco wpłynąć na komfort psychiczny użytkowników, redukując niepewność i sprzyjając pozytywnemu odbiorowi całego procesu nawigacji (Dirin, Laine, i Alamäki 2018).

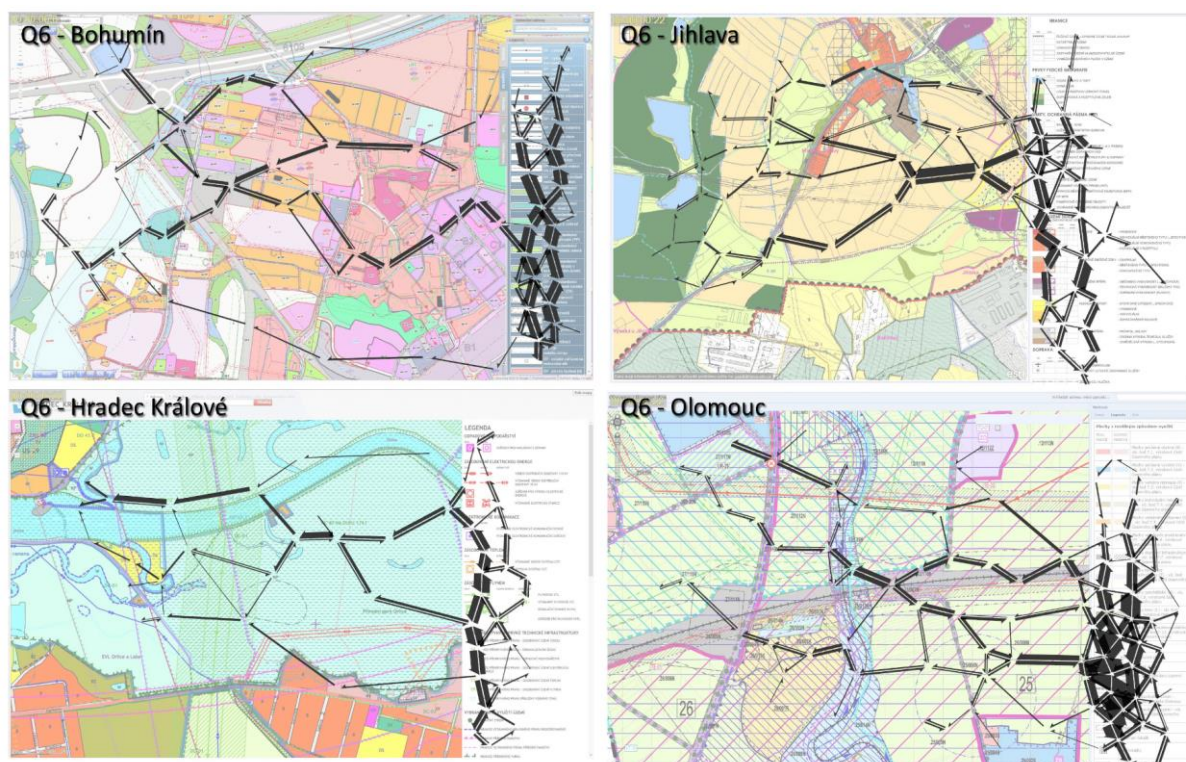
Warto również zwrócić uwagę na niestandardowe podejścia w wykorzystaniu badań ankietowych. Przykładem tego typu jest model „User-driven usability assessment” (Balciunas 2013). Model ten polega na aktywnym zaangażowaniu użytkowników w proces projektowania, dając im możliwość samodzielnego tworzenia map o różnych funkcjach i konfiguracjach interfejsu. Dzięki temu można dokładniej poznać i uwzględnić ich potrzeby oraz oczekiwania, a także zidentyfikować elementy interfejsu i funkcjonalności, które są dla nich najbardziej przydatne.

Z kolei analiza artykułu podsumowującego badania nad mapami holograficznymi i dotykowymi (Campos i in. 2025) pozwala zauważyć, jak powszechne jest łączenie różnych metod ewaluacji, takich jak wywiady, ankietowe badania użyteczności oraz obserwacje, szczególnie w kontekście aplikacji kartograficznych wykorzystujących nowoczesne technologie, np. dotykowe lub holograficzne mapy. Systematyczny przegląd literatury przedstawiony w tym badaniu podkreśla, że łączenie metod jakościowych i ilościowych pozwala na kompleksową ocenę

efektywności, ergonomii i satysfakcji użytkowników, co jest kluczowe dla sukcesu interaktywnych systemów mapowych. Takie holistyczne podejście umożliwia nie tylko identyfikację problemów technicznych, ale także zrozumienie emocjonalnych i behawioralnych aspektów interakcji z mapą.

3.2. Badania z wykorzystaniem śledzenia ruchu gałek ocznych

Ciekawym wykorzystaniem technologii śledzenia ruchów gałek ocznych (ang. *eyetracking*) jest analiza sposobu, w jaki użytkownicy wchodzi w interakcję z klasycznymi planami miejskimi (Burian, Popelka, i Beitlova 2018). Szczególnie interesująca jest tzw. mapa kolejności interakcji, która wizualizuje sekwencję oraz ścieżki wzrokowe użytkowników podczas eksploracji mapy (Rysunek 3.2).



Rysunek 3.2. Mapy pokazujące zagregowane ruchy oczu uczestników podczas rozwiązywania wskazanego podczas badań zadania związanego z mapą topograficzną (Burian, Popelka, i Beitlova 2018)

Badanie porównywało zachowania ekspertów i studentów, co pozwoliło zidentyfikować różnice w sposobie poszukiwania informacji i nawigacji wzrokowej – eksperci wykazywali

bardziej uporządkowane i celowe ruchy oczu, podczas gdy studenci mieli bardziej rozproszone i mniej skoordynowane wzorce. Dzięki temu badanie przyczyniło się do lepszego zrozumienia sposobów korzystania z map topograficznych oraz zidentyfikowało potencjalne obszary poprawy w projektowaniu map pod kątem różnych grup użytkowników.

Interesującym zastosowaniem przenośnego czujnika ruchów gałek ocznych oraz zestawu czujników inercyjnych jest monitorowanie zachowań pieszych. Celem badań przedstawionych w (Kattenbeck i in. 2025) było zbadanie związku między ruchem oczu i ciała badanego, a znajomością przemierzanej przestrzeni. Co ciekawe wbrew wstępnym przypuszczeniom ruch gałek ocznych miał dużo mniejsze znaczenie od ruchów głowy. Chociaż badanie nie koncentruje się bezpośrednio na ewaluacji kartograficznej map, dostarcza istotnych informacji na temat map mentalnych oraz wykrywania stopnia znajomości danej przestrzeni przez użytkownika. Wyniki mogą posłużyć do dynamicznego dopasowywania komunikatów nawigacyjnych do indywidualnego poziomu orientacji w terenie, co stanowi istotny krok w kierunku bardziej spersonalizowanych i efektywnych systemów wspomagania nawigacji (Kattenbeck i in. 2025).

Na podstawie analizy danych z eyetrackingu oraz technik uczenia maszynowego wykazano, że możliwe jest przewidzenie, czy symbol kartograficzny znajduje się w centrum widoku mapy, czy bliżej jego krawędzi. W badaniu wykorzystano różnorodne wskaźniki ruchu gałek ocznych, takie jak czas fiksacji czy rozproszenie wzroku, które posłużyły do skutecznej klasyfikacji lokalizacji symboli. Najważniejszym wskaźnikiem okazało się rozproszenie fiksacji w osi pionowej, które silnie koreluje z położeniem punktów zainteresowania na mapie. To badanie otwiera drogę do przyszłych aplikacji map adaptacyjnych, które w czasie rzeczywistym będą dostosowywać się do użytkownika dzięki analizie ruchów jego oczu, dynamicznie modyfikując interfejs i podkreślając istotne informacje, co może znacząco usprawnić interakcję z mapami cyfrowymi (Cybulski 2025).

Niektórzy badacze rozwijają dalej koncepcję adaptacyjnych map, prezentując i analizując interfejsy sterowane wzrokiem. W jednym z badań porównywano interfejsy całkowicie sterowane wzrokiem z tymi, które łączą sterowanie wzrokiem z użyciem klawiatury (Murphy, Kurumbayeva, i Serrao 2023). Takie badania dostarczają cennych danych na temat przydatności i użyteczności rozwiązań opartych na śledzeniu ruchów gałek ocznych w interakcji z mapami,

wskazując możliwości usprawnienia nawigacji i poprawy komfortu użytkowania za pomocą naturalnych, intuicyjnych gestów wzrokowych.

Warto również przywołać przegląd literatury podsumowujący dotychczasowe zastosowania technologii eyetrackingu w badaniu procesów czytania map. Przeanalizowano 76 artykułów z okresu dziesięciu lat (2009-2019), co świadczy o dużym zainteresowaniu wykorzystaniem tej metody w środowisku kartograficznym (Krassanakis i Cybulski 2019). Główne obszary zastosowań obejmują analizę kartograficznej symbolizacji i zasad projektowania map, badanie sposobu, w jaki użytkownicy czytają i interpretują mapy, a także ocenę samych metod śledzenia ruchu gałek ocznych. Przegląd omawia teoretyczne podstawy, wykorzystywane techniki analizy danych wzrokowych oraz stojące przed badaczami wyzwania związane z interpretacją i wizualizacją wyników, które mają kluczowe znaczenie dla efektywnego projektowania map i ich interfejsów użytkownika.

3.3. Testy terenowe w warunkach rzeczywistych

Testy terenowe stanowią zawsze bardzo istotny wkład w stan wiedzy na dany temat, gdyż są testem najbardziej odzwierciedlającym zachowania użytkowników aplikacji nawigacyjnych w warunków rzeczywistych. Badania opisane w artykule Dong i in. (2021) stanowią doskonały przykład przeprowadzenia testów terenowych w dziedzinie nawigacji pieszej i innowacyjnej kartografii. Autorzy nie ograniczyli się do analizy teoretycznej, lecz zaprojektowali i zrealizowali eksperymenty w rzeczywistych warunkach, co pozwoliło na praktyczną ocenę skuteczności rozszerzonej rzeczywistości (AR) w porównaniu do tradycyjnych, dwuwymiarowych map elektronicznych. Zastosowanie technologii śledzenia wzroku oraz dokładne obserwacje zachowań użytkowników w terenie ukazały, jak różne formy prezentacji informacji wpływają na orientację przestrzenną i efektywność poruszania się. Co ciekawe ostatecznym wnioskiem z przeprowadzonych badań jest porównywalna skuteczność przekazywania wskazówek na mapie 2D i AR (Dong i in. 2021).

Do podobnych wniosków dotarli badacze przedstawiający wyniki badania terenowego porównującego trzy różne interfejsy nawigacyjne dla pieszych: rozszerzoną rzeczywistość (AR), głosowe wskazówki oraz tradycyjną mapę cyfrową (Rehrl i in. 2014). Badanie zostało

przeprowadzone w Salzburgu z udziałem 48 osób o różnym doświadczeniu w korzystaniu z mobilnych aplikacji nawigacyjnych. W eksperymencie mierzono efektywność nawigacji (liczbę i przyczyny zatrzymań, dokładność GPS), efektywność czasową (czas chodzenia i realizacji zadania) oraz satysfakcję użytkowników za pomocą wskaźników NASA Task Load Index⁸ i System Usability Scale⁹. Po pierwszym etapie wprowadzono poprawki w aplikacji na podstawie opinii użytkowników. Wyniki wykazały, że nawigacja z wykorzystaniem klasycznej geowizualizacji i wskazówek głosowych zapewniała lepszą efektywność i doświadczenie użytkownika niż system z AR. Problemy z AR wynikały głównie z ograniczeń sprzętowych i niskiej użyteczności, skutkując większą niepewnością podczas nawigacji. Autorzy podkreślają, że najlepsze wyniki osiąga się dzięki połączeniu dokładnej mapy cyfrowej z precyzyjnymi instrukcjami głosowymi, podczas gdy AR wymaga dalszego udoskonalenia. Po upływie ponad 10 lat warto by było wykonać analogiczne badanie ponownie, gdy poprawiono wiele niedoskonałości technicznych, a technologia AR stała się dużo bardziej dojrzała.

Rozwiązanie ScrollingHome przedstawia innowacyjne podejście do nawigacji wewnątrz budynków za pomocą obrazów wyświetlanych na zegarkach (smartwatch) z kolorowym ekranem dotykowym. Ze względu na ograniczoną przestrzeń ekranu tych urządzeń, autorzy zaproponowali system nawigacji opierający się na sekwencji zdjęć przechwyconych z nagrania wideo wykonanego z perspektywy pierwszej osoby podczas przejścia trasy. Użytkownik przewijał obrazy prostym gestem przewijania pionowego na zegarku, co umożliwiało intuicyjne i swobodne poruszanie się bez konieczności skomplikowanej nawigacji czy lokalizowania się za pomocą infrastruktury. Koncepcję od razu przetestowano w badaniu terenowym, a użytkownicy szybko przyswoili sposób działania systemu i potrafili samodzielnie korygować błędy w nawigacji, przewijając do poprzednich obrazów i porównując je z otoczeniem (Wenig i in. 2016).

⁸ NASA Task Load Index (NASA-TLX) to subiektywna, wielowymiarowa skala oceny obciążenia poznawczego, opracowana przez NASA, mierząca wysiłek umysłowy, fizyczny, czasowy, wydajność, nakład pracy i frustrację podczas zadań (Wikipedia 2025b).

⁹ System Usability Scale (SUS) to standaryzowana, 10-punktowa ankieta oceniająca użyteczność interfejsów użytkownika poprzez pytania dotyczące łatwości użycia, spójności i nauki, z wynikiem od 0 do 100 (Wikipedia 2025c).

4. Zastosowanie symulatorów w badaniach naukowych

Współczesne badania naukowe coraz częściej sięgają po zaawansowane symulatory, których funkcjonalność oraz możliwości odzwierciedlają złożoność interakcji zachodzących w środowiskach rzeczywistych. Możliwości tych narzędzi wykraczają daleko poza proste modelowanie procesów, pozwalając na wielopoziomą analizę zjawisk, projektowanie eksperymentów oraz kształtowanie nowych metod badawczych, które niejednokrotnie redefiniują klasyczne podejścia empiryczne. Symulatory pozwalają też na szybkie prototypowanie nowych rozwiązań bez narażania uczestników na realne ryzyko czy wysokie koszty terenowe.

Opracowany w ramach pracy nad rozprawą symulator przeznaczony jest do badań związanych z percepcją osób poruszających się wewnątrz budynków, a technologicznie łączy wykorzystanie cyfrowych bliźniaków, aplikacji nawigacyjnej oraz technologii gier. W tym rozdziale przeanalizowano wybrane zastosowania i badania opisane w literaturze dotyczące tych technologii.

4.1. Cyfrowe bliźniaki (Digital Twins)

Cyfrowy bliźniak (ang. *Digital Twin*) to pojęcie technologiczne o fundamentalnym znaczeniu dla współczesnych cyfrowych systemów symulacyjnych. Jest ono używane w wielu różnych branżach. Zależnie od kontekstu zastosowania tej technologii można spotkać jej różne definicje.

4.1.1. Cyfrowy bliźniak budynku – definicja

By zdefiniować to pojęcie w sposób jak najbardziej aktualny warto sięgnąć do pracy przeglądowej z 2025 roku, analizującej przypadki jego zastosowania w 15 000 publikacji naukowych dotyczących branży budowlanej (Abdelrahman i in. 2025). Autorzy definiują cyfrowe bliźniaki budynków (ang. *Building Digital Twin – BDT*) jako przestrzenno-czasową wirtualną reprezentację fizycznego zasobu lub systemu, która zarządza wszystkimi istotnymi

informacjami statycznymi i dynamicznymi. Umożliwia współpracę, dostosowuje się do zmian operacyjnych w odpowiednich odstępach czasu oraz potrafi prognozować przyszłe scenariusze w celu optymalizacji wydajności i podejmowania decyzji.

Wskazują też na następujące kluczowe komponenty cyfrowych bliźniaków:

- Dokładna geometria 3D budynków,
- Rozbudowana informacja semantyczna,
- Regularne aktualizacje zmian, jakie zachodzą w świecie rzeczywistym.

Autorzy wskazują na szereg ograniczeń w obecnym rozumieniu i wdrażaniu koncepcji digital twin w sektorze budowlanym. Jednym z kluczowych problemów jest niski poziom integracji danych w czasie rzeczywistym oraz ograniczone wykorzystanie zaawansowanych modeli sztucznej inteligencji, mimo że większość ekspertów uznaje je za fundamentalne elementy tej technologii. Zidentyfikowane luki wynikają w dużej mierze ze złożoności integracji heterogenicznych źródeł danych oraz wysokich wymagań obliczeniowych niezbędnych do prowadzenia symulacji i przetwarzania danych w czasie rzeczywistym w skali budynków i całych miast. Autorzy podkreślają również, iż definicja ewoluuje wraz z pojawianiem się nowych technologii i potrzeb praktycznych.

W kontekście prowadzonych badań nad nawigacją wewnątrz budynków należy odnotować, że wykorzystywane w opracowanym symulatorze modele cyfrowe budynków nie spełniają definicji cyfrowego bliźniaka w jej pełnym rozumieniu. Głównym ograniczeniem jest brak regularnych aktualizacji danych, która stanowi fundamentalną cechę prawdziwego digital twin. Wykorzystywane modele reprezentują statyczne odwzorowania przestrzeni wewnętrznych, oparte na danych zebranych w określonym momencie czasowym, bez ciągłej wymiany informacji między obiektem fizycznym, a jego cyfrową reprezentacją.

Niemniej jednak, dla celów badawczych związanych z nawigacją wewnątrz budynków, tego typu uproszczone modele cyfrowe okazały się wystarczające do przeprowadzenia badań prowadzonych w ramach rozprawy. Umożliwiły one testowanie wskazówek nawigacyjnych, rejestrowanie i analizę ścieżek przemieszczania się oraz symulację różnych scenariuszy poruszania się w przestrzeni wewnętrznej. Statyczne modele 3D budynków pozwoliły na dokładne odwzorowanie geometrii przestrzennej oraz topologii połączeń między

pomieszczeniami, co zdaniem autora w oparciu o przeprowadzone eksperymenty, stanowi wystarczającą podstawę dla rozwoju i testowania systemów nawigacji wewnątrz budynków.

4.1.2. Digital Twins w różnych branżach - zastosowania praktyczne

Digital Twins są stosowane w wielu zróżnicowanych branżach, poniższa lista (Saback i in. 2024) prezentuje najistotniejsze z nich:

Przemysł produkcyjny: planowanie produkcji, kontrola jakości, predykcja konserwacja maszyn, optymalizacja procesów na hali produkcyjnej.

Budownictwo i inżynieria: monitorowanie budynków, optymalizacja zużycia energii, projektowanie i zarządzanie infrastrukturą.

Motoryzacja: projektowanie pojazdów, symulacje i testy samochodów, rozwój autonomicznych pojazdów.

Lotnictwo i kosmonautyka: symulacje działania statków powietrznych, monitorowanie stanu technicznego, wspieranie certyfikacji.

Opieka zdrowotna: personalizowana diagnostyka i leczenie, rozwój leków i urządzeń medycznych, monitorowanie pacjentów.

Rolnictwo: zarządzanie uprawami i hodowlą, optymalizacja nawadniania i zasobów, automatyzacja prac polowych.

Logistyka i łańcuch dostaw: optymalizacja tras transportu, projektowanie magazynów, ochrona przesyłek.

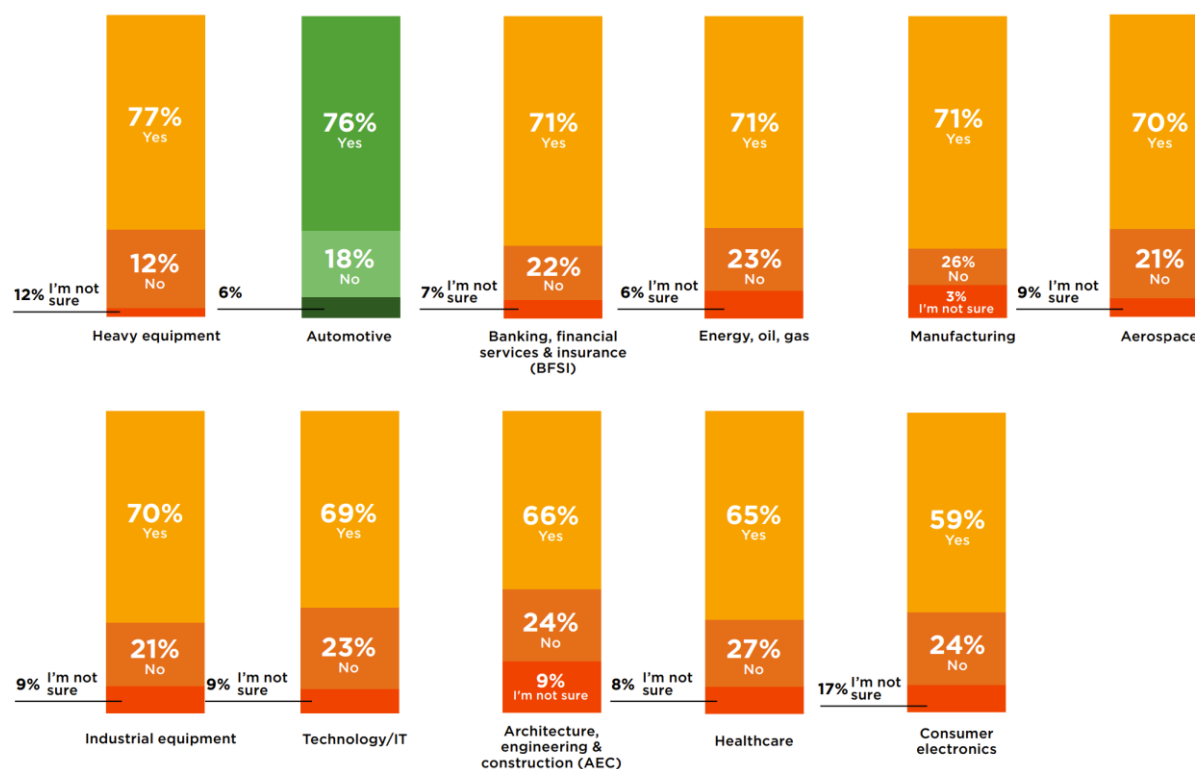
Energetyka: zarządzanie sieciami energii, optymalizacja pracy elektrowni, wsparcie odnawialnych źródeł energii.

Handel detaliczny: wirtualne sklepy, optymalizacja rozmieszczenia produktów, doświadczenia zakupowe z użyciem rozszerzonej rzeczywistości.

Elektronika użytkowa: produkcja i testowanie urządzeń, showroomy wirtualne, współpraca w projektowaniu.

Smart Cities: planowanie miejskie, zarządzanie infrastrukturą i ruchem ulicznym.

Wykresy przedstawione na rysunku 4.1 prezentują odpowiedzi ponad 2000 pracowników różnych branż, zapytanych „czy organizacja, w której pracujesz wykorzystuje już technologię cyfrowych bliźniaków” (Altair 2023).



Rysunek 4.1. Powszechność wykorzystywanie cyfrowych bliźniaków w różnych branżach (Altair 2023)

Warto zwrócić uwagę na fakt, iż termin „Digital Twins” w architekturze i konstrukcji budynków jest wciąż stosunkowo mało znany na tle innych branż.

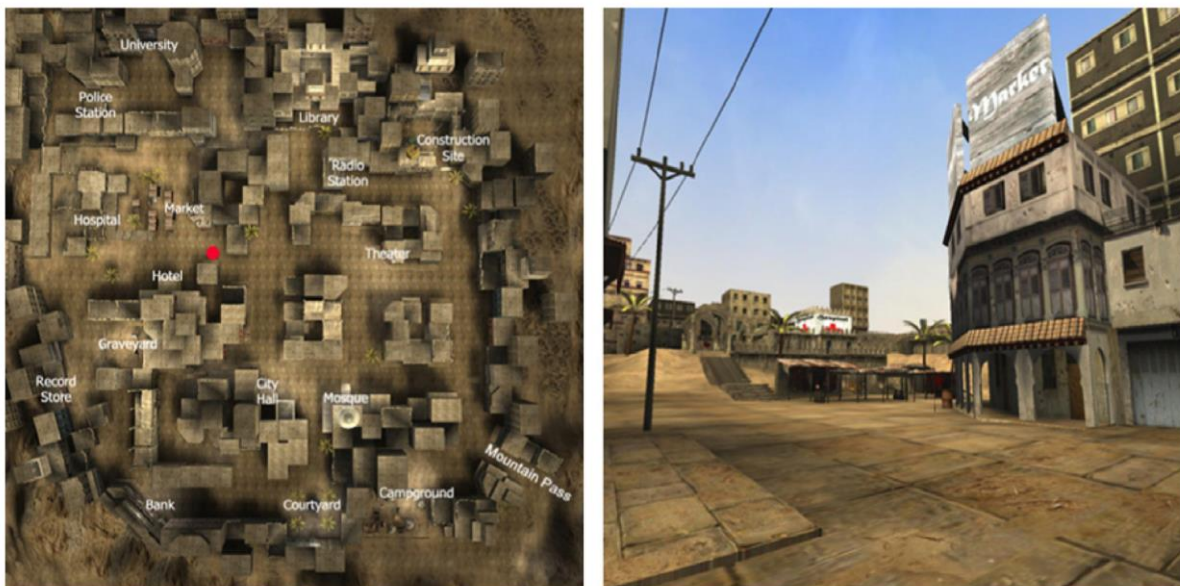
4.2. Badania kartograficzne i nawigacyjne wykorzystujące symulatory i środowiska wirtualne

Jedno z najczęściej cytowanych badań z zakresu badań kartograficznych z wykorzystaniem symulatorów dotyczy wykorzystania środowisk wirtualnych do oceny pamięci drogi i strategii nawigacyjnych u ludzi (van der Ham i in. 2015). Autorzy wskazują, że środowiska VR pozwalają na precyzyjną kontrolę zmiennych eksperymentalnych — w prowadzonym badaniu analizowano, jak różne typy środowisk (rzeczywiste, wirtualne, hybrydowe) wpływają na

skuteczność zapamiętywania tras. Wyniki dowodzą, że środowiska wirtualne zapewniają wysoką powtarzalność i umożliwiają łatwe zbieranie dokładnych danych ruchu, co czyni je wartościowym narzędziem w badaniach nawigacyjnych, choć czasem brakuje im realizmu kontekstowego rzeczywistej przestrzeni.

Badanie „Comparative Study on Simulated Outdoor Navigation for Agricultural Robots” (Khanzada i in. 2024) prezentuje porównanie automatycznego nawigowania robotem w wirtualnym sadzie z wykorzystaniem klasycznych algorytmów SLAM, a także w oparciu o głęboką sieć neuronową wytrenowaną przez naśladowanie (ang. *behavior cloning*). Wyniki pokazują, że realistyczna symulacja umożliwia powtarzalne testy pod względem dokładności jazdy, czasu przejazdu, czy poziomu autonomii jeszcze przed kosztownymi eksperymentami w terenie.

Z analizy literatury naukowej wynika, że dużo uwagi autorzy poświęcają na analizę zdolności człowieka do zapamiętywania układu przestrzennego, m.in. zbadano jak różne perspektywy wizualne wpływają na efektywność uczenia się tras oraz nawigacji w wirtualnych miastach (Brunyé i in. 2012). Poprzez szereg eksperymentów wykazano, że sposób prezentacji (widok z perspektywy osoby poruszającej się, w porównaniu do widoku „z lotu ptaka” (Rysunek 4.2)) istotnie wpływa na efektywność orientacji przestrzennej z korzyścią dla widoku „z lotu ptaka”. Wyniki podkreślają znaczenie typu i jakości wizualizacji oraz roli kartografii w projektowaniu systemów nawigacyjnych przeznaczonych do nawigowania osób. Wpływ perspektywy, z jakiej zapoznawano się z układem przestrzeni, na zapamiętaną reprezentację przestrzeni potwierdzają również badania porównujące symulację widoku z pierwszej osoby oraz mapy dynamicznej (van der Kuil i in. 2021).



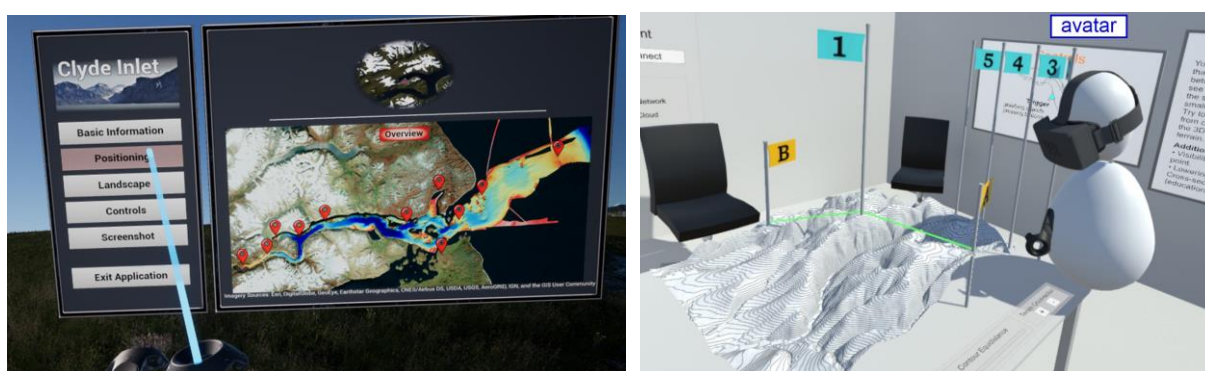
Rysunek 4.2. Porównanie dwóch sposobów prezentacji wirtualnego miasta: widok "z lotu ptaka" po lewej oraz widok z pierwszej osoby po prawej (Brunyé i in. 2012)

Na uwagę zasługuje symulator poruszania się po obszarze kampusu Uniwersytetu w Tajpej (Tajwan), stworzony w programach Blender i Unity (Hsu, I Chen, i Zheng 2023). Trzydziestu uczestników, podzielonych na trzy grupy, wykonywało zadania nawigacyjne w wirtualnym środowisku z perspektywy pierwszej osoby, poruszając się po kampusie za pomocą klawiszy WASD i myszy. Każda grupa otrzymała inny scenariusz z sześcioma trasami, a uczestnicy musieli odnaleźć określone cele na kampusie, korzystając z istniejącego systemu map. Badanie ujawniło dwa główne problemy projektowe: po pierwsze, brakujące informacje – ważne miejsca nie były zaznaczone na mapie, zmuszając uczestników do ślepego poszukiwania. Po drugie, wady projektowe – mapy wzdłuż tej samej drogi miały niespójne wskazania kierunków (trzy mapy na jednej ulicy prezentowały informacje w trzech różnych orientacjach). Dodatkowo, kodowanie kolorami i numeracją budynków okazały się mało pomocne (25 na 30 uczestników nie uznało ich za użyteczne), a niekompletne oznaczenia "Jesteś tutaj" prowadziły do błędnych decyzji nawigacyjnych. Badanie wykazało również, że doświadczenie w grach 3D z perspektywy pierwszej osoby znacząco wpływało na skuteczność poruszania się w symulatorze.

W badaniach przeanalizowano także minimapy w 100 popularnych grach wideo, pod kątem ośmiu cech projektowych, takich jak zastosowana projekcja, sposób centrowania czy kształt. Zidentyfikowano najczęściej wykorzystywane parametry tych cech, np. minimapa w kształcie koła, centrowana na gracz z projekcją ortograficzną i strzałką północy, co potwierdza

możliwość projektowania minimap zgodnie z tradycyjnymi regułami kartografii (Zagata i Medyńska-Gulij 2023).

Kolejny nurt badań koncentruje się na wykorzystaniu VR do prezentacji klasycznych danych przestrzennych, takich jak numeryczny model terenu. Dzięki wysokiej immersji, te rozwiązania znacząco podnoszą jakość edukacji geograficznej. Użytkownicy mogą swobodnie eksplorować szczegółowe modele terenu, zmieniać tekstury, kontrolować poziom wody czy wykonywać pomiary bezpośrednio w przestrzeni 3D (Lütjens i in. 2019; Sedlák i in. 2022). Rysunek 4.3 przedstawia przykłady aplikacji badawczych tego typu.

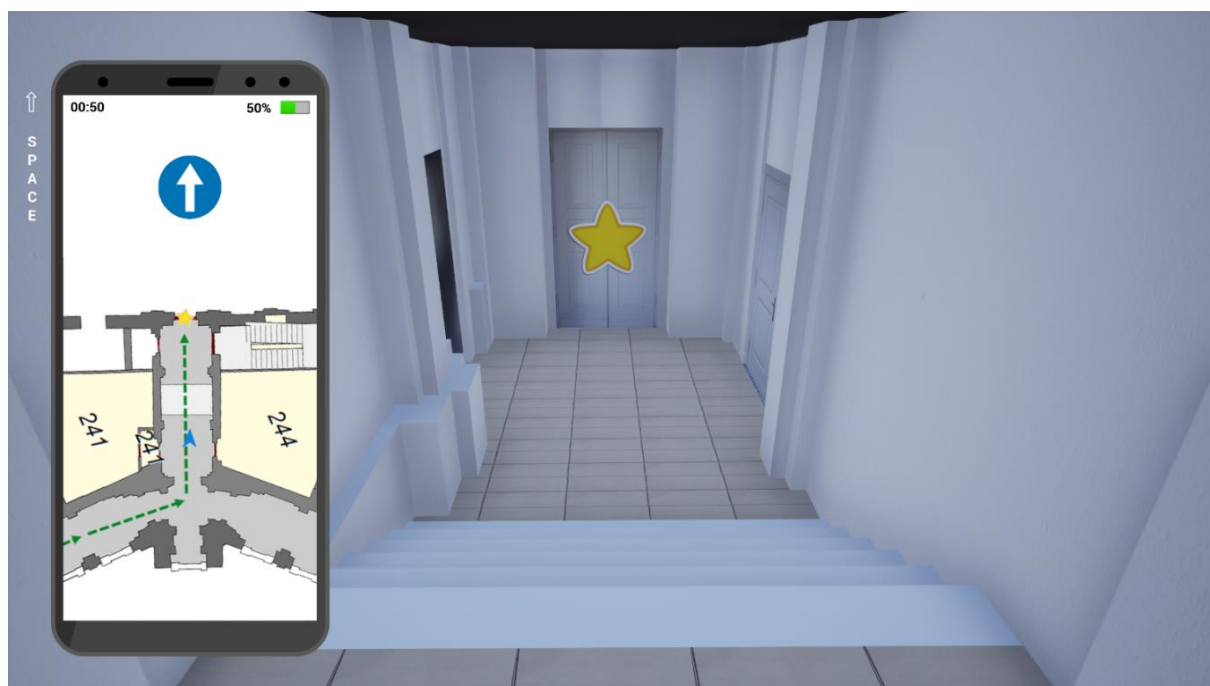


Rysunek 4.3. Przykłady aplikacji badawczych wykorzystujących środowiska wirtualne do pracy z danymi przestrzennymi (Lütjens i in. 2019; Sedlák i in. 2022)

Dużym zainteresowaniem badaczy cieszą się również symulatory nawigacji morskiej. Przykładowo przeanalizowane zostały aplikacje zaawansowanych symulatorów mostka nawigacyjnego w przemyśle morskim oraz badaniach naukowych (Zghyer i in. 2022). Autor wskazuje, że symulatory pozwalają nie tylko szkolić załogi w bezpiecznych warunkach, ale również rozwijać i testować nowe systemy kontroli, oceniać interfejsy czy analizować zachowania w sytuacjach awaryjnych. W innym badaniu przedstawiono metodykę implementacji symulatora nawigacji morskiej z wykorzystaniem wirtualnej rzeczywistości (Kong i Roh 2024). System umożliwia generowanie realistycznych scenariuszy nawigacyjnych oraz analizowanie zachowań użytkowników w złożonych warunkach hydrometeorologicznych i ruchu statków. Artykuł podkreśla skuteczność środowiska VR do zdobywania precyzyjnych danych dotyczących trajektorii statków wynikających z decyzji nawigacyjnych.

Warto również zwrócić uwagę na wcześniejsze prace autora (Łobodecki i Gotlib 2022), które koncentrowały się na pomiarze efektywności pojedynczych komunikatów nawigacyjnych

podczas nawigacji w budynku (Rysunek 4.4). Wyniki pilotażowych badań zrealizowanych w prototypowym środowisku wirtualnym wewnątrz budynku wykazały, iż tego typu symulator pozwala na efektywne ocenianie różnych typów wskazówek nawigacyjnych oraz preferencji użytkowników. Badanie potwierdziło także przydatność silnika Unreal Engine jako narzędzia służącego implementacji zaawansowanego systemu przeznaczonego do badań kartograficznych.



Rysunek 4.4. Symulator poruszania po wnętrzu budynku, przeznaczony do testowania wskazówek nawigacyjnych (Łobodecki i Gotlib 2022)

4.3. Symulatory środowisk wewnątrzbudynkowych

Na podstawie analizy literatury można wyróżnić kilka głównych branż, w których istotne jest wykorzystanie modeli wnętrz budynków. Niewiele z prowadzonych badań dotyka jednak kwestii ludzkiej percepcji takich przestrzeni.

Główną branżą, która zwraca uwagę na wrażenia ludzi w kontakcie z modelami wnętrz jest branża handlu nieruchomościami. W 2025 roku trudno znaleźć ofertę nowobudowanych mieszkań, dla których deweloper nie przygotowałby atrakcyjnych trójwymiarowych wizualizacji, co dowodzi ich istotnego wpływu na decyzje zakupowe potencjalnych nabywców. Wpływ ten został również wykazany w publikacjach naukowych (Prema i in. 2024), choć wyniki

nie są jednoznaczne. Badanie (Z. Yan, Meng, i Tan 2025), bazujące na danych z dużej platformy sprzedażowej, potwierdza, że wykorzystanie wirtualnej rzeczywistości znacząco skraca czas ekspozycji nieruchomości na rynku, lecz nie wykazuje statystycznie istotnego wpływu na ostateczną cenę sprzedaży. Z kolei analiza z wykorzystaniem metod uczenia maszynowego wykazała, że wdrożenie modeli 3D w prezentacji nieruchomości prowadzi do podniesienia średniej ceny sprzedaży o 7%, z mniej istotnym wpływem na czas obecności nieruchomości na rynku (AlDhaheri 2023). Z perspektywy deweloperów istotne mogą być również badania dotyczące ulepszania projektowania. Przykładem może być metodyka integracji technologii CAD z analizą dużych zbiorów danych o zachowaniach użytkowników w celu optymalizacji układu przestrzeni wewnętrznych budynków (S. Huang i Kang 2024). Dzięki analizie rzeczywistych wzorców zachowań użytkowników możliwe jest inteligentne planowanie rozmieszczenia elementów wnętrza, co poprawia funkcjonalność i ergonomię przestrzeni. Taka synergia technologii wspiera efektywne projektowanie wnętrz, co ma istotne znaczenie dla deweloperów planujących inwestycje, umożliwiając zwiększenie komfortu użytkowników oraz optymalizację kosztów i czasu realizacji.

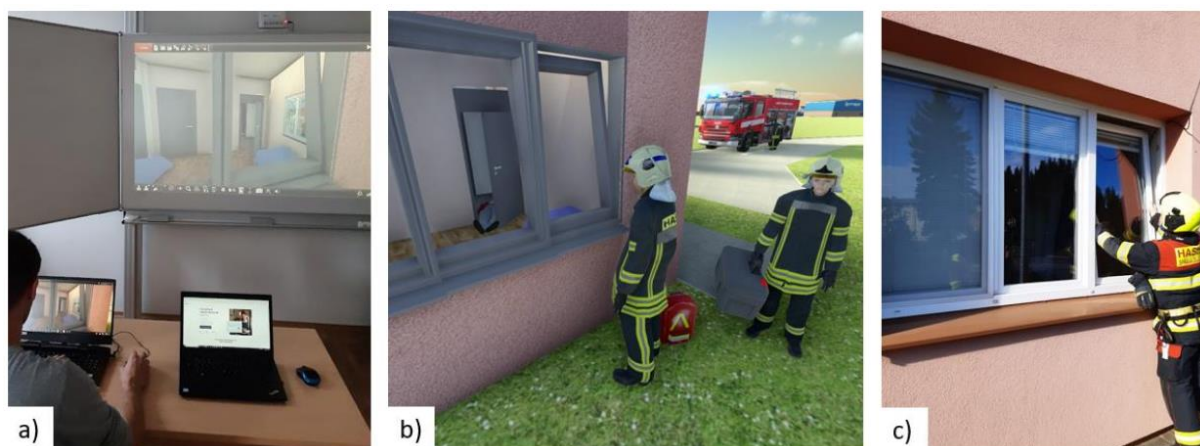
Drugim obszarem, w którym symulacje środowiska wewnątrz budynków są wykorzystywane szczególnie często, jest dziedzina bezpieczeństwa pożarowego. Modele budynków wykorzystywane są do modelowania rozprzestrzeniania się ognia oraz dymu, co pozwala na ocenę dynamiki pożaru i identyfikację najbardziej zagrożonych obszarów (Charalampous, Besharat, i Stylios 2021; Yi i in. 2024). Ponadto, symulacje te są istotnym narzędziem w zarządzaniu procesami ewakuacji osób podczas sytuacji kryzysowych. Umożliwiają one optymalizację ścieżek ewakuacyjnych poprzez analizę różnych scenariuszy przemieszczania się ludzi w warunkach zagrożenia, co pozwala na wskazanie najbezpieczniejszych tras ucieczki z budynku (Shih i Tsai 2023; Tan i in. 2025). Dzięki wykorzystaniu danych GIS oraz symulacji trójwymiarowych, możliwe jest także modelowanie zachowań osób przebywających w budynku, uwzględniając dynamiczne zmiany środowiska, na przykład powstawanie stref zagrożenia czy zadymienie (Tang i Ren 2012).



Rysunek 4.5. Symulacja ewakuacji z budynku podczas pożaru (Tang i Ren 2012)

Takie podejście pozwala na lepsze planowanie ścieżek ewakuacji, minimalizując ryzyko paniki i zwiększając efektywność działań ratunkowych. Dodatkowo, symulacje te umożliwiają ocenę bezpieczeństwa ewakuacji w specyficznych typach konstrukcji, na przykład w obiektach zabytkowych o drewnianej strukturze, gdzie warunki pożarowe mogą się istotnie różnić, co wpływa na dynamikę ewakuacji i wymaga indywidualnego podejścia do planowania ścieżek ewakuacyjnych (Yi i in. 2024).

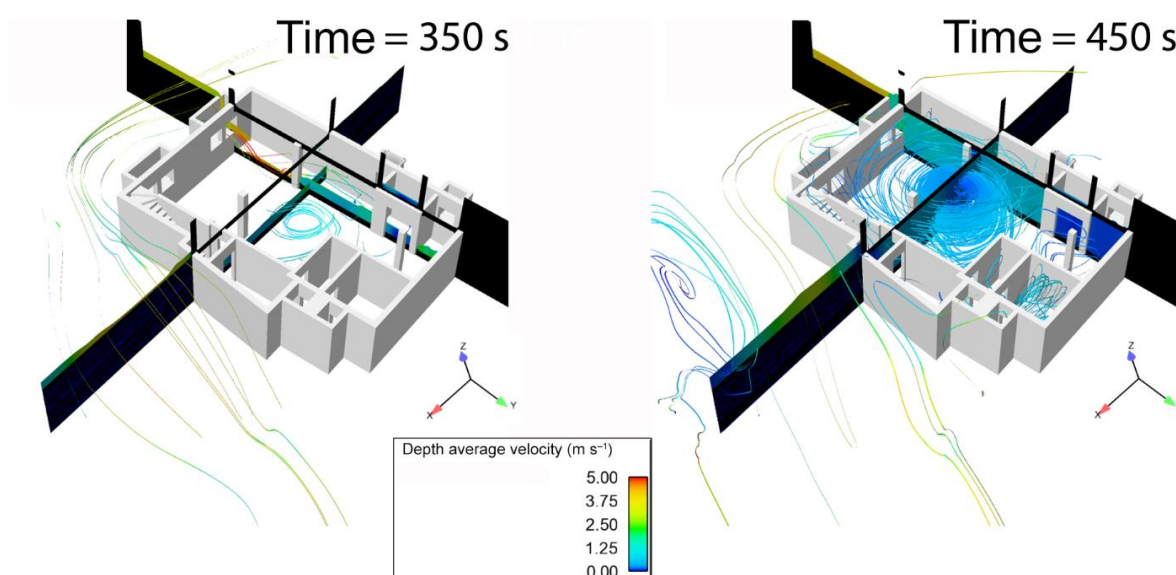
Równie ważnym zastosowaniem jest trening służb ratowniczych, zwłaszcza strażaków, którzy dzięki realistycznym symulacjom mogą doskonalić swoje umiejętności walki z żywiołem i mieć okazję na ćwiczenie różnych scenariuszy akcji ratunkowej w różnych typach budynków. Badania potwierdzają istotną poprawę efektywności członków oddziałów straży podczas treningu rzeczywistego, gdy wcześniej przeszli oni już trening w środowisku wirtualnym (Antonin Miroslav 2024; Hady Sassi, Battisti, i Carli 2022).



Rysunek 4.6. Kolejne fazy badania potwierdzającego wpływ treningu w VR na sytuacje w świecie rzeczywistym (Antonin Miroslav 2024)

Środowisko VR pozwala również w bezpiecznych warunkach sprawdzać oraz trenować odpowiedź organizmu na stres i presję związane z akcją gaśniczą (Mehta i in. 2025). Wszystko to przekłada się na faktyczną skuteczność działań ratowniczych w rzeczywistości.

Innym przykładem jest modelowanie numeryczne powodzi, ze szczególnym uwzględnieniem trójwymiarowych symulacji hydrodynamicznych, umożliwiające przewidywanie przepływu wód powodziowych wokół budynków oraz ich infiltracji do wnętrza, co jest kluczowe dla oceny zagrożeń i uszkodzeń infrastruktury (Gems i in. 2016). Takie symulacje pozwalają na uwzględnienie warunków terenowych i konstrukcyjnych, wspierając planowanie działań ochronnych oraz operacje ratunkowe.



Rysunek 4.7. Ilustracja ścieżki przepływu wody do oraz z budynku (Gems i in. 2016)

Artykuł Stachoňa i współautorów (Stachoň i in. 2025) jest szczególnie interesujący ze względu na wysoką zgodność metodologiczną z badaniami prowadzonymi w ramach niniejszej rozprawy doktorskiej, szczególnie w zakresie wykorzystania technologii wirtualnego środowiska wnętrza budynku do analizy zachowań nawigacyjnych w środowiskach zamkniętych. Badanie to bezpośrednio porównuje zachowania ludzi podczas ewakuacji w rzeczywistym i wirtualnym środowisku, wykorzystując zaawansowane narzędzia takie jak eyetracking i precyzyjne modele 3D.



Rysunek 4.8. Porównanie różnic pomiędzy środowiskiem rzeczywistym a wirtualnym (Stachoň i in. 2025)

W eksperymencie wzięło udział 73 uczestników, a badacze analizowali trzy typy danych: trajektorie ruchu, zarejestrowane spojrzenia na kluczowe obiekty (np. znaki ewakuacyjne), oraz dodatkowe parametry (np. odległość od obiektu w momencie pierwszego spojrzenia). Nie stwierdzono istotnych różnic statystycznych w wyborze trasy ewakuacji między środowiskiem rzeczywistym, a wirtualnym. Większość uczestników w obu grupach podążała za znakami ewakuacyjnymi (74% w środowisku rzeczywistym, 60,5% w środowisku wirtualnym).

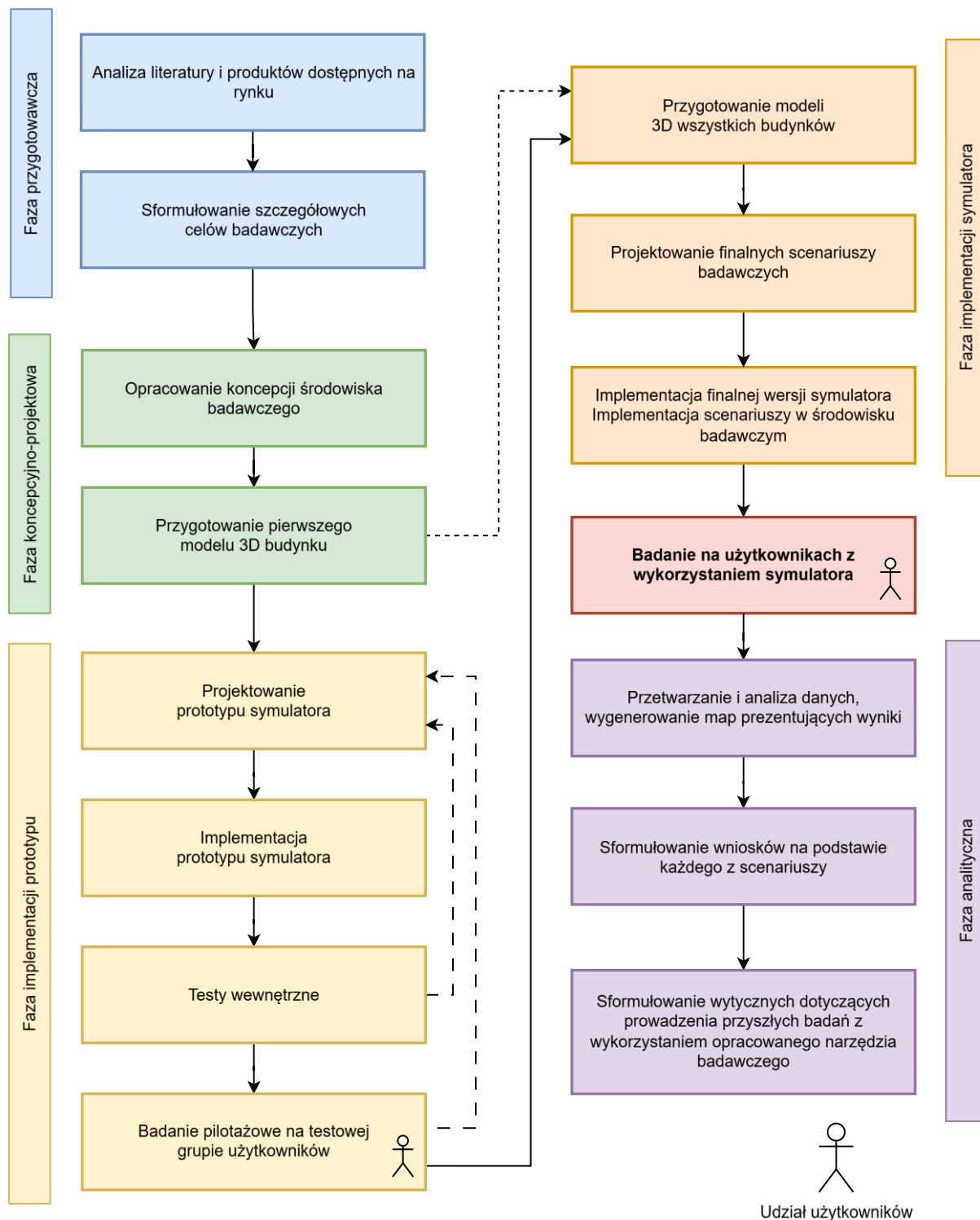
Badanie to potwierdza, że rzeczywistość wirtualna może być wiarygodnym narzędziem do badań nad orientacją przestrzenną i ewakuacją, szczególnie w kontekście oceny skuteczności znakowania nawigacyjnego.

W przeprowadzonych w ramach rozprawy badaniach jeden ze scenariuszy badawczych również dotyczył analizy wyboru drogi ewakuacji z budynku. Został on szczegółowo opisany w rozdziale 7.5.8.

5. Metodyka badawcza oraz koncepcja proponowanego narzędzia badawczego

5.1. Założenia główne

Zaproponowane podejście metodyczne zakłada łączne wykorzystanie modeli cyfrowych bliźniaków prawdziwych budynków, innowacyjnego środowiska symulacyjnego, iteracyjnego podejścia do procesu jego implementacji oraz wykorzystanie kilku metod badawczych. Proces badań podzielony został na sześć wzajemnie powiązanych faz, z których każda służy realizacji specyficznych celów poznawczych i metodologicznych, przedstawionych na rysunku 5.1.



Rysunek 5.1. Diagram metodyki badawczej (opracowanie własne)

Faza przygotowawcza

Proces badawczy zainicjowała krytyczna analiza literatury przedmiotu oraz produktów dostępnych na rynku. W obliczu rzadkości aplikacji przeznaczonych do nawigacji wewnątrz budynków oraz ich niepełnej dojrzałości technologicznej, systematyczny przegląd istniejących rozwiązań umożliwił identyfikację kluczowych obszarów problemowych i określenie ram teoretycznych dla dalszych działań badawczych.

Na tym etapie sformułowano również szczegółowe cele badawcze, które stanowią uzupełnienie głównej tezy rozprawy. Proces ten wymagał z jednej strony wzięcia pod uwagę założonych możliwości i ograniczeń środowiska wirtualnego, a z drugiej strony uwzględnienia specyfiki środowisk zamkniętych oraz związanych z nimi wyzwań technicznych, takich jak brak wystarczająco dokładnych metod pozycjonowania czy ograniczona dostępność szczegółowych danych przestrzennych.

Faza koncepcyjno-projektowa

W tej fazie opracowano koncepcję systemu oraz prototypu symulatora jako głównego systemu do prowadzenia badań. Kluczowym elementem tej fazy było przygotowanie pierwszych modeli 3D budynków osadzonych w wirtualnej przestrzeni posiadającej georeferencję, co stanowiło podstawę dla późniejszych eksperymentów symulacyjnych.

Wykorzystanie cyfrowych kopii rzeczywistych budynków oraz silników gier komputerowych pozwoliło na stworzenie kontrolowanego środowiska badawczego, które eliminuje ograniczenia związane z trudnościami organizacyjnymi testowania w warunkach rzeczywistych. Podejście to umożliwia wielokrotne powtarzanie eksperymentów oraz szczegółową analizę zachowań użytkowników bez konieczności organizacji badań w rzeczywistych budynkach.

Faza implementacji prototypu

Etap implementacyjny charakteryzował się iteracyjnym procesem rozwoju prototypu symulatora. Każda iteracja składała się z procesów projektowania, implementacji i walidacji. Walidacja systemu przebiegała w dwóch etapach: testy wewnętrzne, badanie pilotażowe na małej grupie użytkowników. Uzyskana informacja zwrotna pozwalała na stopniowe

doskonalenie narzędzia badawczego oraz minimalizację ryzyka wystąpienia błędów krytycznych w głównej fazie eksperymentalnej.

Modularność architektury systemu zapewniła elastyczność implementacji i możliwość przyszłych modyfikacji, co jest istotne wobec dynamicznie rozwijających się technologii, potencjalnych zmian w założeniach badawczych oraz ewentualnym wykorzystaniu tego samego systemu w kolejnych badaniach.

Faza implementacji badania na użytkownikach

W tej fazie na podstawie wcześniejszych doświadczeń przygotowywano badanie przeznaczone dla szerokiej grupy uczestników.

Kluczowym elementem tej fazy było przygotowanie i implementacja scenariuszy badawczych oraz rozszerzenie repozytorium o modele 3D kolejnych budynków. Scenariusze rozumiane są jako zaprojektowana wcześniej sytuacja nawigacyjna, odbywająca się w przestrzeni określonego budynku, z wyznaczonym konkretnym zadaniem nawigacyjnym (np. dotarcia do wskazanego pomieszczenia). Scenariusze różnią się między sobą rodzajem pomocy i wskazówek dostępnych podczas wykonywania zadania (np. mapa z wyznaczoną trasą), a wewnątrz pojedynczego scenariusza uczestnicy są losowo przydzielani do grupy eksperymentalnej i kontrolnej (np. grupa eksperymentalna widzi mapę z trasą, a grupa kontrolna tylko mapę). Wszyscy uczestnicy badania realizują wszystkie scenariusze w tej samej kolejności.

Każdy scenariusz został zaprojektowany jako wyodrębniony eksperyment, precyzyjnie ukierunkowany na realizację celu badawczego. Scenariusze te różnicuje nie tylko różna wirtualna przestrzeń, przez którą porusza się uczestnik, ale również typ oraz sposób dostarczania informacji nawigacyjnych czy poziom skomplikowania zadań.

W finalnej wersji badania wdrożonych zostało 16 scenariuszy wykorzystujących 5 modeli budynków 3D. Podczas symulacji poruszania się w budynku rejestrowany był nie tylko czas realizacji zadania, ale również współrzędne trasy, jaką pokonał uczestnik. Tym samym, uzyskiwane dane pozwalają zarówno na weryfikację hipotez dotyczących efektywności i użyteczności testowanych aplikacji, jak i na sformułowanie uogólnionych wniosków na temat wpływu specyfiki i struktury przestrzeni na percepcję i zachowania użytkowników.

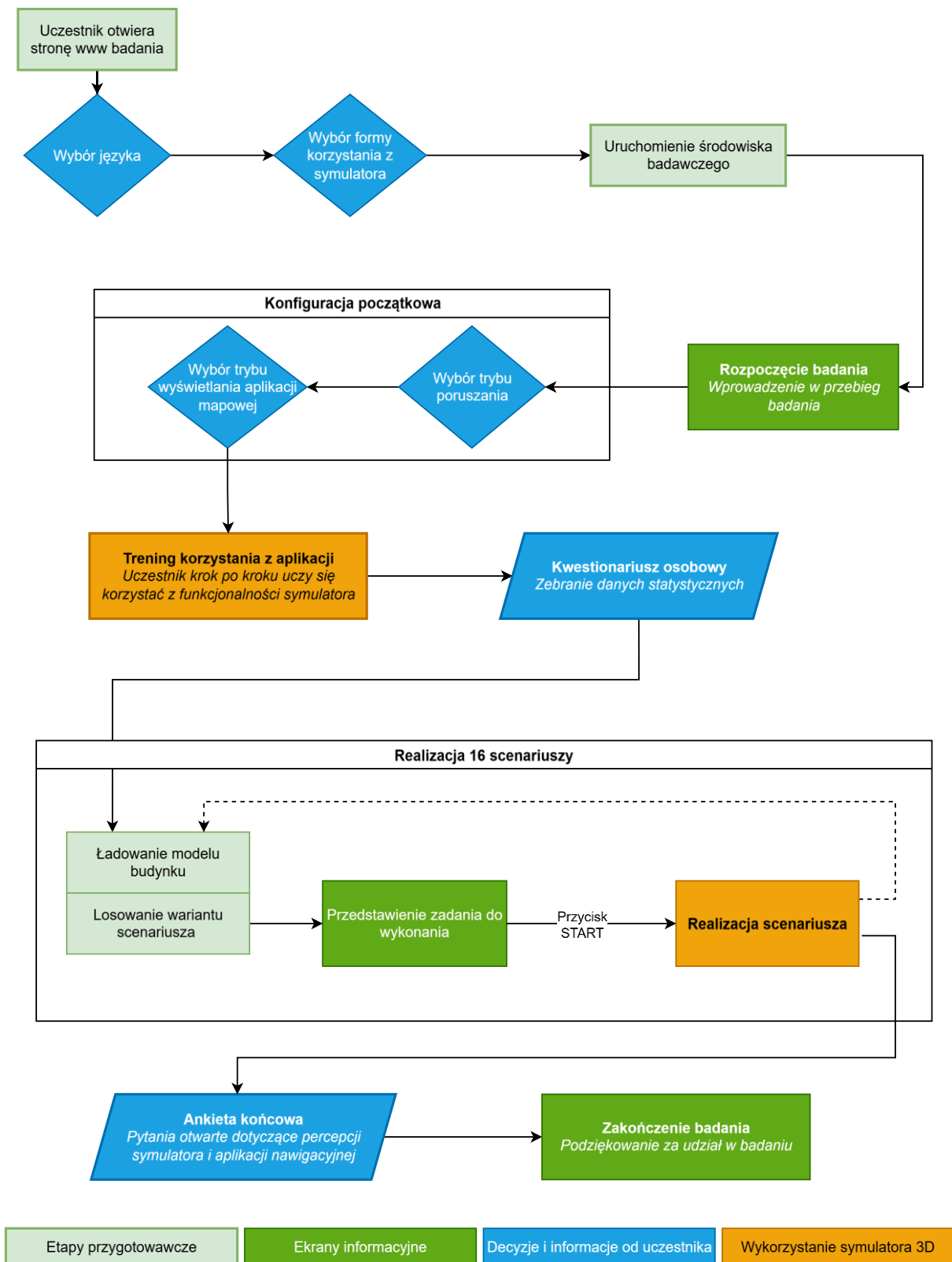
Faza analityczna

Analiza obejmowała przetwarzanie danych uzyskanych w ramach realizacji poszczególnych scenariuszy badawczych względem zakładanych celów badawczych oraz generowanie map prezentujących wyniki. Wieloaspektowy charakter analizy obejmujący zarówno metody ilościowe, jak i jakościowe umożliwia zrozumienie mechanizmów wpływu aplikacji nawigacyjnych na zdolności uczenia się przestrzennego oraz orientację przestrzenną użytkowników. Z kolei zebrane dane o lokalizacji uczestników w przestrzeni w każdym momencie badania pozwala lepiej zrozumieć charakter wyzwań z jakimi mierzą się osoby odwiedzające dany budynek.

Ostatni etap badań koncentrował się na sformułowaniu wniosków i wytycznych dotyczących prowadzenia przyszłych badań z wykorzystaniem opracowanego narzędzia badawczego. Wyniki tego procesu mają potencjał, by stać się istotną pomocą podczas projektowania aplikacji nawigacyjnych, dostosowanych do potrzeb użytkowników i specyfiki środowisk zamkniętych.

5.2. Opis procesu badawczego

Przebieg badania z udziałem opracowanego symulatora przedstawiono na diagramie pokazanym na rysunku 5.2. Struktura diagramu odzwierciedla sekwencyjny przebieg eksperymentu z jasno zdefiniowanymi punktami decyzyjnymi. Kluczowym elementem procesu badawczego jest wykorzystanie 16 zaplanowanych scenariuszy badawczych. Na ich podstawie zaplanowano przeprowadzenie 7 eksperymentów. Zakładany czas realizacji badania dla każdego z uczestników to 20 do 40 minut.



Rysunek 5.2. Diagram procesu badawczego z użyciem projektowanego symulatora (opracowanie własne)

Badanie rozpoczyna się, gdy uczestnik otwiera stronę www z aplikacją badawczą. W tym momencie uczestnik dokonuje wyboru języka, co pozwala na personalizację interfejsu (dostępne języki to polski oraz angielski).

Kolejnym krokiem jest wybór technologii korzystania z aplikacji. Uczestnik badania ma dwie alternatywne ścieżki: uczestnictwo zdalne poprzez przesyłanie obrazu z chmury obliczeniowej (model SaaS -Software as a Service) lub korzystanie z aplikacji na komputerze personalnym po pobraniu odpowiednich plików instalacyjnych. Umożliwia to dotarcie do szerszej grupy badanych poprzez uwzględnienie różnych preferencji i możliwości technicznych. Szczególnie istotna jest możliwość uruchomienia aplikacji w chmurze za pośrednictwem przeglądarki internetowej, co zapewnia łatwość rozpoczęcia badania.

W finalnym badaniu ponad 87% uczestników wybrało opcję przesyłania obrazu z chmury.

Szczegółowa analiza wyborów uczestników została przeprowadzona w rozdziale 7.4.

Na początku wyświetlany jest ekran powitalny zawierający ogólne informacje o przebiegu badania. Ten element służy nie tylko edukacji uczestnika, ale również stanowi część protokołu etycznego, informując o celach badania oraz procedurach.

Później następuje konfiguracja początkowa, która obejmuje wybór trybu wyświetlania map budynków oraz wybór trybu poruszania w symulatorze badawczym. Ta faza ma znaczenie z perspektywy uniwersalności przeprowadzanego badania, gdyż zapewnia dostosowanie sposobu korzystania z symulatora do preferencji i doświadczenia uczestnika.

Przed właściwym eksperymentem każdy uczestnik musi przejść przez poziom treningowy, w którym stopniowo poznaje funkcjonalność symulatora. Poziom treningowy pełni w badaniach podwójną funkcję. Z jednej strony pozwala na wyeliminowanie z udziału w badaniu osób, które nie posiadają wymaganych minimalnych umiejętności technicznych do obsługi symulatora. To podnosi jakość zebranych danych i oszczędza czas osób, które i tak nie dałyby rady wziąć udziału w badaniu tego typu, które wymaga umiejętności manualno-technicznych). Z drugiej

strony, poziom treningowy dzięki swojej prostocie, umożliwia uczestnikom zapoznanie się z symulatorem w komfortowych, bezstresowych warunkach. Dzięki temu uczestnicy chętniej angażują się już na początku procesu i nie obawiają się kolejnych etapów.

Po ukończeniu treningu konieczne jest wypełnienie kwestionariusza osobowego służącego zebraniu danych statystycznych uczestnika. Ten komponent umożliwia późniejszą analizę statystyczną oraz identyfikację potencjalnych czynników osobowych mających istotny wpływ na wyniki badań.

Rdzeniem procesu badawczego jest iteracyjna pętla obejmująca realizację 16 scenariuszy przejścia po budynkach. System dynamicznie konfiguruje środowisko wirtualne zgodnie z parametrami danego scenariusza oraz losuje wartości zmiennych niezależnych, które będą testowane podczas jego realizacji. Po ukończonym ładowaniu danych do symulatora prezentowany jest widok całego budynku z góry, co pozwala uczestnikowi na orientację przestrzenną w szerszym kontekście oraz przygotowanie do kolejnego zadania. Realizacja scenariusza stanowi właściwą fazę eksperymentalną, w której uczestnik wykonuje w środowisku wirtualnym zadanie nawigacyjne polegające na dotarciu do wskazanych drzwi zgodnie z instrukcjami aplikacji badawczej. Informacja o punkcie docelowym scenariusza poprzedza każde zadanie, a także jest wyświetlana przez cały czas trwania scenariusza zapewniając jasność instrukcji oraz standaryzację warunków eksperymentalnych.

Po zakończeniu wszystkich scenariuszy uczestnik ma możliwość wzięcia udziału w ankiecie końcowej, która obejmuje pytania otwarte dotyczące sposobu korzystania z symulatora, percepcji aplikacji nawigacyjnej i wykorzystanej mapy. Jest to istotny element proponowanej metody, uzupełniający dane ilościowe zebrane podczas realizacji scenariuszy, o komponent jakościowy pozwalający na głębsze zrozumienie doświadczeń uczestników oraz identyfikację aspektów, które mogą być niedostępne poprzez obserwację behawioralną.

Szczegółowy przebieg badania można sprawdzić i przeanalizować na dwa sposoby:

- 1) Uruchomienie symulatora analogicznie jak uczestnicy badania, poprzez wejście na stronę:

<https://mapa.pw.edu.pl/indoorlab/>

(dostępność przynajmniej do końca roku 2026)

Potrzebne pliki instalacyjne dołączono także do rozprawy w formie archiwum .zip

- 2) Obejrzenie nagrania ekranu przechwyconego podczas przykładowej realizacji całości badania.

Nagranie dołączono do rozprawy w formie pliku .mp4 (16m 45s)

oraz udostępniono w serwisie YouTube pod adresem:

<https://youtu.be/INZMCPS8JNQ>



5.3. Charakterystyka zmiennych niezależnych kontrolowanych w procesie badawczym

W tym podrozdziale przedstawiona zostanie logika eksperymentów, idea przeprowadzonych testów A/B oraz lista zmiennych niezależnych, które w opracowanej metodyce mogą być zmieniane na potrzeby poszczególnych scenariuszy badawczych.

W literaturze naukowej zmienna niezależna — często określana również mianem zmiennej objaśniającej lub predyktora — oznacza taką cechę, którą badacz w sposób zamierzony kontroluje lub modyfikuje w przebiegu eksperymentu bądź obserwacji (Andrade 2021). Jej główną funkcją jest umożliwienie badania wpływu tej cechy na inne zmienne, zwane zależnymi, których wartości mają odzwierciedlać efekt owej manipulacji lub różnicowania.

W Tabeli 2 przedstawiono wykaz wszystkich zmiennych niezależnych, których randomizacja została zaimplementowana w symulatorze i wykorzystana w badaniu na użytkownikach, opisanym w rozdziale 7.

Tabela 2. Wykaz zmiennych niezależnych, których randomizacja została zaimplementowana w symulatorze i wykorzystana w badaniu na użytkownikach (opracowanie własne)

Nazwa zmiennej	Możliwe opcje	Uzasadnienie badawcze
Wyświetlanie mapy <i>mapVisibilityMode</i>	a. alwaysVisible <i>mapa zawsze widoczna</i> b. onRequest <i>mapa wyświetlana na żądanie</i> c. notVisible <i>mapa niedostępna</i>	Potwierdzenie w jakich sytuacjach wyświetlenie mapy przynosi faktyczną korzyść
Wyświetlanie trasy <i>routeMode</i>	a. showRoute <i>wyświetlanie na mapie trasy prowadzącej do celu</i> b. hideRoute <i>trasa niewyświetlana</i>	Potwierdzenie w jakim stopniu wyświetlenie trasy skraca czas dotarcia do celu
Wyświetlanie lokalizacji <i>locationMode</i>	a. showLocation <i>wyświetlanie na mapie aktualnej pozycji</i> b. hideLocation <i>pozycja niewyświetlana</i>	Potwierdzenie jak bardzo istotna jest precyzyjna lokalizacja wewnątrz budynków z perspektywy nawigowania osób w budynkach
Tryb prezentacji kierunku ruchu <i>rotationMode</i>	a. mapRotation <i>podczas obrotu obracanie całej mapy zgodnie z ruchem użytkownika</i> b. blueDotRotation <i>podczas obrotu obracanie wskaźnika lokalizacji</i> c. noRotation <i>kierunek niewyświetlany</i>	Stwierdzenie jak dużą stratą dla skuteczności nawigowania będzie rezygnacja z oznaczania kierunku patrzenia użytkownika.
Wskazówki w formie strzałek <i>hintsMode</i>	a. showHints <i>wyświetlanie strzałek nawigacyjnych podpowiadających kierunek ruchu</i>	Sprawdzenie, czy mając do dyspozycji strzałkę nawigacyjną

	b. hideHints <i>strzałki niewyświetlane</i>	użytkownicy rzadziej będą korzystać z funkcji oglądania mapy.
Poziom przybliżenia <i>zoomMode</i>	a. 19.5 <i>pełen korytarz, ~1:850</i> b. 20 <i>część korytarza, ~1:550</i> c. 21 <i>najbliższe pomieszczenia, ~1:300</i>	Sprawdzenie, czy wyświetlenie mapy w większej skali przekłada się na mniejszą liczbę spojrzeń na mapę

Na liście zmiennych zależnych obserwowanych w opisywanym badaniu znalazły się: czas realizacji scenariusza, przebyty dystans i uzyskana średnia prędkość, liczba spojrzeń na mapę, lokalizacja w których wystąpiły spojrzenia na mapę oraz czas kontaktu z mapą.

Zmienne niezależne uboczne to zmienne, z których istnienia badacz zdaje sobie sprawę i które stara się kontrolować.

Na liście tego typu zmiennych znalazły się: różnice demograficzne (np. wiek, płeć), różnice indywidualne (np. wcześniejsze doświadczenie, znajomość budynków), jakość wykorzystywanego sprzętu.

Z kolei zmienne niezależne zakłócające to te zmienne, o których istnieniu badacz nie wie, a które wpływają na zachowanie badanych. W przypadku tego badania może to być m.in. sytuacja eksperymentalna (np. pora dnia podczas wykonywania eksperymentu, aktualny poziom zmęczenia i stresu).

Istotą proponowanej metodyki jest manipulacja jedną lub kilkoma, określonymi zmiennymi niezależnymi w ramach pojedynczego scenariusza lub grupy powiązanych scenariuszy. Wartość tej zmiennej zostaje wylosowana podczas ładowania w symulatorze każdego z scenariuszy. Dzięki odpowiednio licznej próbie możliwe staje się obliczanie średniej sprawności realizacji zadań (np. średniego wyniku czasowego) w zależności od ustawień kontrolowanej zmiennej.

Test A/B (inaczej test dzielony – ang. *split test*) lub randomizowany test kontrolny) polega na równoległym porównaniu co najmniej dwóch wariantów interfejsu / zachowania mapy, z losowym przydziałem uczestników do wariantu kontrolnego (A) oraz co najmniej jednego wariantu eksperymentalnego (B). Ta metoda prowadzenia badań jest szczególnie popularna w zagadnieniach związanych z testowaniem oprogramowania komputerowego czy analizie skuteczności procesów sprzedażowych (Quin i in. 2024). Kluczowe elementy testów A/B to:

- Losowa alokacja użytkowników – minimalizuje wpływ zmiennych zakłócających
- Metryka nadrzędna – zdefiniowany wskaźnik sukcesu, np. czas ukończenia zadania
- Analiza istotności – porównanie rozkładów metryki w grupach oraz testowanie hipotezy zerowej o braku różnicy z użyciem testów proporcji lub testów t-Studenta

W kontekście nawigacji w przestrzeniach zamkniętych taka procedura umożliwia rzetelne porównanie skutków różnych konfiguracji widoku mapy czy różnych typów wskazówek nawigacyjnych, co bywa trudne do osiągnięcia w badaniach obserwacyjnych.

5.4. Opis danych zbieranych podczas procesu badawczego

Po każdym ukończonym scenariuszu wszystkie zebrane dane są natychmiast zapisywane na serwerze (w chmurze). Dzięki temu możliwe jest określenie w ilu scenariuszach badawczych wzięł udział każdy z uczestników.

Przesyłane są wszystkie zebrane dotychczas dane, w tym dane demograficzne, konfiguracja wybrana przez uczestnika oraz statystyki dotyczące danego scenariusza. W Tabeli 3 przedstawiono kompletną listę danych rejestrowanych i przesyłanych przez system badawczy.

Tabela 3. Listę danych rejestrowanych i przesyłanych przez system badawczy (opracowanie własne)

Kategoria	Zbierana statystyka	Możliwe opcje lub typ danych
Dane demograficzne	Płeć	Kobieta, Mężczyzna, Nie chcę podawać
	Wiek	<16, 17-24, 25-34, 35-44, 45-54, 55-64, 65+
Dane kompetencyjne	Umiejętność grania w gry komputerowe	Nie, Tak, Trochę
	Częstotliwość używania aplikacji nawigacyjnych	Codziennie, Często, Nigdy, Rzadko
	Samooocena orientacji w przestrzeni	Dobrze, Słabo, Średnio
	Deklaracja znajomości budynków, w modelach których przebiega badanie	Gmach Architektury, Gmach Chemii, Gmach Elektroniki, Gmach Elektrotechniki, Gmach Główny
Wybrana konfiguracja	Tryb poruszania (opisane szczegółowo w rozdziale 6.6)	Rozglądanie myszą, a poruszanie klawiaturą tak jak w grach FPS ¹⁰ . Rozglądanie klawiaturą, a przemieszczenia kliknięciem myszy, podobnie jak w grach Point-and-click ¹¹
	Tryb wyświetlania mapy	Mapa osadzona jako element interfejsu na ekranie komputera. Mapa wyświetlana na smartfonie, używanym równolegle do komputera.

¹⁰ FPS (First Person Shooter) - gatunek gier komputerowych, których akcja obserwowana jest z perspektywy pierwszej osoby.

¹¹ Point-and-click – ang. *wskaż i kliknij* – gatunek gier komputerowych, w którym postacią steruje się za pomocą wskaźnika. Wskazuje się nim miejsce, do którego ma iść bohater.

	Sposób uczestnictwa	Pixel Streaming, Pobranie aplikacji na własny komputer
Wylosowana konfiguracja zmiennych niezależnych	Listę zmiennych przedstawia Tabela 2	Lista zmiennych z wylosowanymi wartościami
Statystyki ukończonego scenariusza	Czas ukończenia scenariusza	Liczba sekund
	Czas kontaktu z mapą	Liczba sekund
	Dotychczasowy łączny czas trwania badania	Liczba sekund
	Wyjście, które wybrał uczestnik (dotyczy scenariuszy z więcej niż jednym możliwym punktem końcowym)	Nazwa punktu końcowego
	Przebyta trasa	Lista współrzędnych
	Lokalizacje spojrzeń na mapę	Lista współrzędnych
Odpowiedzi udzielone w ankiecie końcowej	Opisowa ocena działania symulatora	Tekst
	Opisowa ocena percepcji mapy	Tekst
	Opisowe sugestie ulepszeń mapy	Tekst

Po odebraniu danych są one automatycznie uzupełniane o dodatkowe statystyki pochodne, takie jak przebyty dystans oraz obliczona średnia prędkość przemieszczania.

Szczegóły dotyczące technicznych aspektów zbierania danych opisano w rozdziale 6.10.

6. Techniczne aspekty implementacji symulatora

Przedmiotem niniejszej rozprawy jest nie tylko zaproponowanie metodyki badawczej oraz przeprowadzenie konkretnych badań na użytkownikach, ale również zaproponowanie konkretnego, autorskiego rozwiązania technologicznego. Autor podczas swojej pracy nad rozprawą nie znalazł żadnego środowiska umożliwiającego mu przeprowadzenia zaplanowanych badań zgodnie z opisaną metodyką. Opracował więc własne rozwiązanie (platformę badawczą), od etapu koncepcji (przy współpracy promotora), przez etap doboru technologii, projektowania aż po etap implementacji. W kolejnych podrozdziałach opisane zostały kluczowe aspekty techniczne budowy platformy badawczej, która powstała w oparciu o silnik Unreal Engine. Utworzony projekt otrzymał nazwę *IndoorLab*.

6.1. Silnik gier Unreal Engine jako podstawa platformy badawczej

Unreal Engine stanowi kompleksowy silnik gier komputerowych opracowany przez przedsiębiorstwo Epic Games, będący jedną z najważniejszych platform rozwojowych w branży interaktywnych aplikacji multimedialnych. Narzędzie to definiuje się jako zintegrowane środowisko programistyczne umożliwiające renderowanie grafiki dwu- i trójwymiarowej, implementację systemów fizycznych, zarządzanie zasobami audio oraz programowanie algorytmów sztucznej inteligencji. Architektura systemu oparta jest na języku C++, co zapewnia optymalną wydajność obliczeniową oraz wieloplatformowość.

Geneza Unreal Engine sięga roku 1995, kiedy Tim Sweeney, założyciel Epic Games, zainicjował prace nad technologią silnika. Komercyjna implementacja nastąpiła w 1998 roku wraz z debiutem tytułu Unreal (Rysunek 6.1), który stanowił pierwszą praktyczną aplikację tej technologii. Pierwotne założenia projektowe koncentrowały się na obsłudze gier gatunku First person shooter (pol. strzelanka pierwszoosobowa), jednak już w fazie rozwojowej w 1996 roku technologia została udostępniona zewnętrznym studiom deweloperskim jako Unreal Engine, co umożliwiło pozyskanie środków finansowych na dalszy rozwój platformy (Wikipedia 2025d).



Rysunek 6.1. Okładka gry Unreal z 1998 roku (archive.org).
 Hasło reklamowe: „Najlepiej wyglądająca gra wszechczasów.”

Ewolucja technologiczna Unreal Engine charakteryzuje się progresywnym wprowadzaniem zaawansowanych rozwiązań algorytmicznych - od pierwszej iteracji wykorzystującej programowe renderowanie, poprzez Unreal Engine 2 z implementacją obsługi konsol siódmej generacji, Unreal Engine 3 ze wsparciem dla architektury wielowątkowej, aż po Unreal Engine 4 wprowadzający wizualny system programowania "Blueprints". Najnowsza wersja Unreal Engine 5, wydana w kwietniu 2022 roku, implementuje technologie Nanite (system zwirtualizowanej geometrii) oraz Lumen (dynamiczne globalne oświetlenie w czasie rzeczywistym).

Współczesne zastosowania Unreal Engine wykraczają znacząco poza pierwotne przeznaczenie jako narzędzie do tworzenia gier komputerowych. Platforma znajduje szerokie zastosowanie w badaniach naukowych, szczególnie w dziedzinie sztucznej inteligencji i robotyki jako środowisko symulacyjne. W sektorze przemysłu filmowego wykorzystywana jest do tworzenia wirtualnych planów zdjęciowych oraz efektów wizualnych w czasie rzeczywistym. Dodatkowo, silnik służy jako narzędzie w wizualizacji architektonicznej, aplikacjach rzeczywistości wirtualnej i rozszerzonej, oraz jako platforma badawcza w obszarze wizji komputerowej i uczenia maszynowego. Dzięki zaawansowanym możliwościom technicznym oraz elastycznej architekturze, Unreal Engine stanowi uniwersalną platformę do tworzenia fotorealistycznych

środowisk wirtualnych, służących zarówno celom komercyjnym, edukacyjnym, jak i badawczym w obszarze nauk komputerowych.

Zastosowanie Unreal Engine w środowisku badawczym zyskało znaczące uznanie w społeczności naukowej, co potwierdzają liczne publikacje akademickie dokumentujące wykorzystanie tej platformy w różnorodnych projektach badawczych. Analiza literatury przedmiotu przeprowadzona przez projekt UE4Research wskazuje na kilka kluczowych cech technicznych, które decydują o przydatności tego silnika jako narzędzia badawczego (Rehn 2021).

Generowanie fotorealistycznej grafiki stanowi jedną z najczęściej wymienianych zalet platformy w kontekście naukowym. Zaawansowane algorytmy renderowania Unreal Engine umożliwiają tworzenie wysokiej jakości wizualizacji, które są niezbędne w badaniach z zakresu wizji komputerowej, symulacji środowiskowych oraz projektów wymagających precyzyjnej reprodukcji warunków wizualnych rzeczywistego świata.

Wysokiej jakości symulacja fizyki i kolizji stanowi kolejny istotny aspekt wykorzystywany w badaniach naukowych. Zintegrowany system fizyczny Unreal Engine pozwala na realistyczne modelowanie interakcji między obiektami, co jest szczególnie wartościowe w robotyce, symulacjach behawioralnych oraz badaniach nad sztuczną inteligencją wymagających precyzyjnego odwzorowania praw fizyki.

System programowania wizualnego Blueprint znacząco obniża próg wejścia dla badaczy nieposiadających zaawansowanej wiedzy programistycznej. Ta cecha umożliwia szybkie prototypowanie eksperymentów oraz implementację złożonych scenariuszy badawczych bez konieczności pisania kodu w języku C++, co czyni platformę dostępną dla szerszego grona naukowców.

Otwarty kod źródłowy Unreal Engine zapewnia badaczom pełną kontrolę nad środowiskiem symulacyjnym oraz możliwość modyfikacji funkcjonalności zgodnie z wymaganiami konkretnych projektów badawczych. Ta charakterystyka jest szczególnie istotna w kontekście reprodukowalności badań oraz konieczności dostosowania narzędzi do specyficznych potrzeb eksperymentalnych.

Dostępność gotowych zasobów w postaci modeli 3D, tekstur, materiałów oraz prefabrykatów znacznie przyspiesza proces tworzenia środowisk badawczych. Bogata biblioteka fab.com (wcześniej nazywana Unreal Marketplace) oraz społeczność deweloperska dostarczają rozbudowany ekosystem komponentów, które mogą być bezpośrednio wykorzystane w projektach naukowych.

Interoperacyjność z oprogramowaniem zewnętrznym umożliwia integrację Unreal Engine z istniejącymi narzędziami badawczymi, bibliotekami uczenia maszynowego oraz platformami analitycznymi. Ta cecha jest kluczowa dla projektów wymagających złożonych procesów przetwarzania danych oraz współpracy z wyspecjalizowanym oprogramowaniem naukowym.

Wielopatformowość Unreal Engine stanowi szczególnie istotny aspekt jego wykorzystania w środowisku badawczym. Silnik funkcjonuje natywnie w trzech głównych systemach operacyjnych: Windows, macOS oraz Linux, zapewniając badaczom elastyczność w wyborze środowiska deweloperskiego. Projekty tworzone w Unreal Engine mogą być eksportowane nie tylko na platformy desktopowe, ale również na urządzenia mobilne (Android, iOS), konsole do gier, a także przesyłane bezpośrednio do przeglądarki użytkownika za pośrednictwem technologii Pixel Streaming (zastosowane w niniejszej pracy, więcej w rozdziale 6.7.2). Wsparcie dla systemu Linux ma szczególne znaczenie w kontekście badań naukowych, ponieważ ten system operacyjny jest szeroko wykorzystywany w infrastrukturze obliczeniowej ośrodków badawczych ze względu na swoją stabilność, bezpieczeństwo oraz możliwość precyzyjnego konfigurowania środowiska systemowego. Analiza publikacji naukowych wskazuje, że badacze cytujący wielopatformowość jako kryterium wyboru Unreal Engine najczęściej wykorzystywali właśnie dystrybucje Linux do przeprowadzania eksperymentów.

Powszechne wykorzystanie w branży oraz wsparcie techniczne ze strony Epic Games zapewniają stabilność oraz długoterminową dostępność platformy, co jest istotne dla projektów badawczych o długim horyzoncie czasowym.

W oparciu o przeprowadzoną analizę cech technicznych oraz przegląd literatury naukowej, autor zdecydował się na wybór Unreal Engine w wersji 5.4 jako platformy badawczej do realizacji niniejszej pracy. Decyzja ta wynika z kilku kluczowych argumentów technicznych i metodologicznych. Przede wszystkim, udokumentowane w literaturze zalety platformy - w

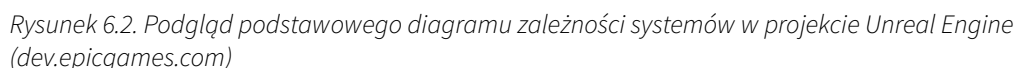
szczegółności zdolność do generowania fotorealistycznej grafiki, zaawansowana symulacja fizyki oraz system programowania wizualnego Blueprint - korespondują bezpośrednio z wymaganiami projektów badawczych wymagających wysokiej precyzji wizualnej i symulacyjnej. Wieloplatformowość silnika, szczególnie natywne wsparcie dla systemów Linux, zapewnia kompatybilność z istniejącymi środowiskami obliczeniowymi i pozwala na skorzystanie z możliwości technologii Pixel Streaming. Dodatkowo, otwarty kod źródłowy oraz rozbudowany ekosystem gotowych zasobów znacząco redukują czas potrzebny na implementację środowiska eksperymentalnego, co jest szczególnie istotne w kontekście efektywności metodologicznej badań naukowych. Wybór ten jest również wspierany przez dostępność obszernej dokumentacji technicznej, a także społeczności aktywnych użytkowników silnika.

6.2. Architektura informatyczna systemu

Ze względu na wysoką uniwersalność oraz rozległą funkcjonalność silnika Unreal Engine, architektura realizowanego projektu *IndoorLab* charakteryzuje się dużą złożonością strukturalną. Niniejszy rozdział zawiera syntetyczny opis architektury projektowej, obejmujący wybrane komponenty strukturalne o kluczowym znaczeniu dla funkcjonowania systemu, wyselekcjonowane według kryteriów istotności określonych przez autora opracowania.

Podstawowym pojęciem Unreal Engine jest tzw. Projekt. Każdy projekt realizowany w silniku Unreal Engine implementuje złożoną hierarchię klas i komponentów, które tworzą podstawową strukturę do zarządzania logiką gry. Proces inicjalizacji silnika charakteryzuje się deterministyczną sekwencją uruchamiania poszczególnych modułów, gdzie każdy z nich pełni specyficzną rolę w architekturze systemu. Mechanizm ten determinuje tworzenie instancji obiektów ważnych dla funkcjonowania aplikacji, choć nie w każdym projekcie wszystkie są wykorzystywane, z uwagi na przeznaczenie dla projektów wieloosobowych (ang. *multiplayer*).

<https://dev.epicgames.com/documentation/en-us/unreal-engine/gameplay-framework-in-unreal-engine>



1. Konfiguracja Projektu

74

2. Instancja Game Instance

W kolejnym kroku realizacji projektu tworzona jest klasa Game Instance - najbardziej globalny komponent w architekturze silnika, charakteryzujący się dostępnością między różnymi poziomami gry. W opracowanym projekcie klasa ta nie jest intensywnie wykorzystywana, ponieważ całość symulacji opiera się o jeden główny poziom (ang. *MainLevel*) i nie jest przewidziana jego zmiana, a jedynie doładowywanie dodatkowych podpoziomów (ang. *SubLevel*).

3. Ładowanie Poziomu

Po ustanowieniu Game Instance następuje proces ładowania poziomu (ang. *level*), który stanowi kontekst przestrzenny dla wszystkich obiektów istniejących w wirtualnym świecie.

4. Aktywacja Game Mode

Game Mode reprezentuje centralny mechanizm zarządzania regułami gry. Odpowiada za definiowanie domyślnych klas dla kluczowych komponentów systemu (np. Default Pawn Class, Player Controller Class, HUD Class), a także koordynuje ich inicjalizację.

5. Tworzenie obiektu gracza (Player object), na który składa się:

- a. Pawn/Character – reprezentacja fizyczna gracza w świecie gry
- b. Player Controller – logiczna warstwa zarządzająca zachowaniem obiektu gracza

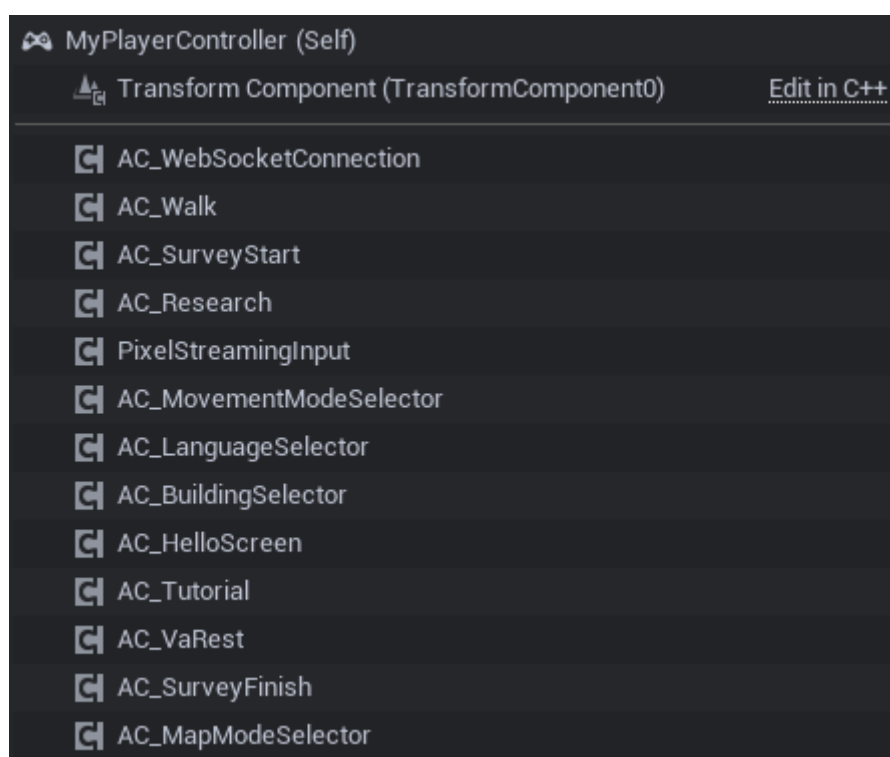
6. Uruchamianie kolejnych podsystemów, w szczególności:

- a. Input System – zbiór reguł definiujących sposób manipulacji postacią przez użytkownika. System ten umożliwia mapowanie sygnałów z urządzeń wejściowych na funkcje akcji gracza poprzez współpracę z PlayerController.
- b. Navigation System – zestaw klas odpowiedzialnych za nawigację implementuje algorytmy wyznaczania ścieżek, umożliwiając przemieszczanie się gracza w przestrzeni 3D (system ten wykorzystywany jest w uproszczonym trybie poruszania, patrz rozdział 6.6)
- c. HUD System (ang. *Head-Up Display*) – zarządza warstwą prezentacji informacji dla użytkownika, implementując wzorzec MVC poprzez separację logiki prezentacji od logiki biznesowej

Znaczna część tych komponentów pozostaje dostępna dla większości encji funkcjonujących w środowisku symulacyjnym, co czyni je optymalnymi kontenerami do przechowywania bieżących parametrów stanu systemu symulacyjnego.

W opracowanym projekcie liczne komponenty architektury systemowej zostały zaimplementowane w rozszerzonej klasie **Player Controller** (Rysunek 6.3). Instancja tej klasy inicjalizowana jest w jednej z końcowych faz procesu konstrukcji środowiska wirtualnego i w prezentowanym przypadku pozostaje aktywna przez całą długość cyklu wykonawczego symulacji. Architektura systemu nie przewiduje scenariuszy destrukcji obiektu reprezentującego użytkownika w konfiguracjach innych niż standardowe zakończenie procesu symulacyjnego.

Kluczowe dane przechowywane w klasie reprezentującej użytkownika obejmują między innymi statyczne parametry konfiguracyjne projektu oraz obiekty enkapsulujące bieżące zmienne stanu systemu (przykładowo: identyfikator aktualnie wykonywanego scenariusza, status interfejsu użytkownika, parametry połączenia sieciowego). Dodatkowo klasa ta jest rozbudowywana poprzez wzorzec kompozycji o dodatkowe komponenty funkcjonalne, które enkapsulują wyspecjalizowane funkcjonalności środowiska symulacyjnego.



Rysunek 6.3. Lista komponentów rozszerzających funkcjonalność klasy *PlayerController* (opracowanie własne)

Zgodnie z przyjętą konwencją nazwy komponentów zaczynają się od przedrostka „AC”, będącego skrótem od Actor Component (pol. *komponent aktora*).

Na szczególną uwagę zasługuje komponent *AC_WebSocketConnection* odpowiadający za połączenie z serwerem służącym komunikacji z aplikacją mapową (więcej w rozdziale 6.9), komponent *AC_MapModeSelector* tworzący obiekt mapy w trybie embedded oraz *AC_Research* – komponent odpowiadający za całość procesu przebiegu badania głównego.

Wiele komponentów ma dodatkowe zadanie polegające na obsługiwaniu interfejsu użytkownika widocznego na ekranie poprzez komunikację z klasą HUD. Jednym z podstawowych zadań tych komponentów jest utworzenie instancji WBP (ang. *Widget Blueprint*) czyli tzw. widżetu i wyświetlenie go użytkownikowi. Instancje widżetów są natomiast przechowywane w klasie MyHUD, za pośrednictwem której mogą być pokazywane i ukrywane z ekranu.

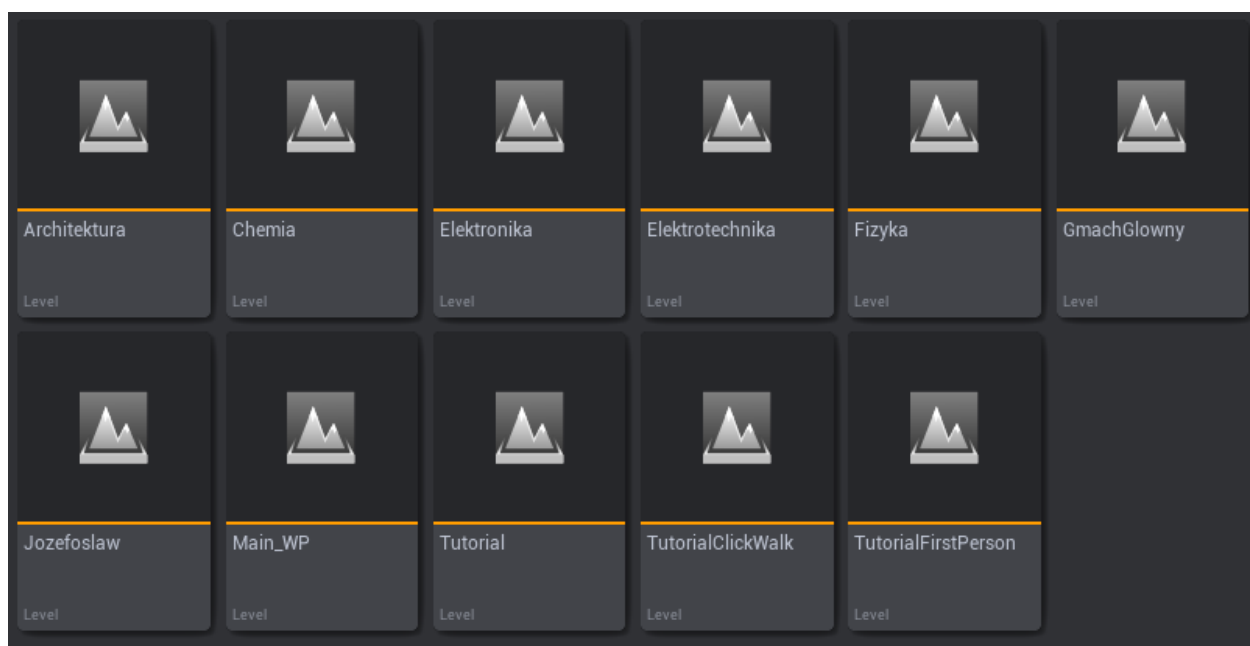
WBP_ControlsCore	WBP Controls Core	
WBP_HelloScreen	WBP Hello Screen	
WBP_MovementModeSelect	WBP Movement Mode Select	
WBP_MapModeSelect	WBP Map Mode Selector	
WBP_LanguageSelector	WBP Language Selector	
WBP_Tutorial	WBP Tutorial	
WBP_BetweenNavTasks	WBP Between Nav Tasks	
WBP>LoadingScreen	WBP Loading Screen	
WBP_SurveyStart	WBP Survey Start	
WBP_SurveyFinish	WBP Survey Finish	
WBP_QRWebView	WBP QRWeb View	
WBP_MapWebView	WBP Map Web View	
WBP_Menu	WBP Menu	
WBP_Timer	WBP Timer	
WBP_VersionCheck	WBP Version Check	
WBP_ElevatorButtons	WBP Elevator Buttons 2	
WBP_Hint	WBP Hint	
WBP_Destination	WBP Destination	
WBP_FinalScreen	WBP Final Screen	

Rysunek 6.4. Lista widżetów składających się na interfejs użytkownika symulatora IndoorLab (opracowanie własne)

Poziomy, czyli wycinki wirtualnego świata

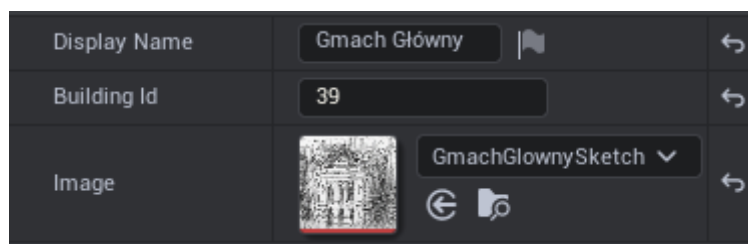
Jedną z wykorzystanych w projekcie najnowszych funkcji silnika Unreal jest system World Partition, umożliwiający konstrukcję środowisk wirtualnych o nieskończonych wymiarach przestrzennych poprzez segmentację na mniejsze poziomy / mapy (ang. *Levels / Maps*).

Dodatkowo geometria modeli poszczególnych struktur budynków została zapisana w odrębnych mapach i jest dynamicznie ładowana do pamięci operacyjnej na żądanie. Ze względu na wyjątkowo wysoką złożoność geometryczną modeli budynków skutkuje to istotną optymalizacją pod kątem wydajności i szybkości renderowania. Za zarządzanie aktualnie załadowanymi poziomami budynków odpowiada plugin CesiumForUnreal opisany w rozdziale 6.3. Listę wszystkich poziomów używanych w projekcie, tj. 1 poziom główny, 3 poziomy samouczka oraz 7 podpoziomów z modelami różnych budynków Politechniki Warszawskiej, zaprezentowano poniżej (Rysunek 6.5).



Rysunek 6.5. Poziomy wykorzystywane w projekcie IndoorLab (opracowanie własne)

Ponadto klasa zawierająca reprezentację budynku została rozszerzona o komponent informacyjny zawierający metadane, takie jak: identyfikator budynku w bazie danych, pełną nazwę obiektu oraz nawiązani do grafiki reprezentującej dany budynek (Rysunek 6.6).



Rysunek 6.6. Konfiguracja komponentu przechowującego metadane budynku (opracowanie własne)

6.3. Integracja Cesium i Unreal

Cesium to kompleksowa platforma technologiczna przeznaczona do wizualizacji danych 3D, która początkowo została opracowana w 2011 roku przez zespół deweloperów z firmy Analytical Graphics Inc. (AGI) w celu obsługi wizualizacji lotniczych w oparciu o wirtualny globus. Następnie na bazie rozwoju produktu w 2019 roku powstała firma Cesium.

Pierwotnie Cesium było przede wszystkim biblioteką przeznaczoną dla środowisk webowych, umożliwiając wyświetlanie Ziemi jako trójwymiarowego globu z wysokim poziomem szczegółowości. Jej modułowa architektura i rozbudowane API pozwalają na efektywne prezentowanie danych przestrzennych w sposób atrakcyjny wizualnie oraz użyteczny funkcjonalnie. Elastyczna struktura frameworku umożliwia jego wykorzystanie jako solidnej podstawy dla aplikacji analitycznych różnego rodzaju, wykraczając poza podstawową funkcjonalność wizualizacyjną.

Jedną z najistotniejszych innowacji wprowadzonych przez platformę Cesium jest format **Cesium 3D Tiles** – otwarta specyfikacja umożliwiająca optymalne strumieniowanie dużych zbiorów danych 3D. Format ten został zaprojektowany w celu efektywnej transmisji i wizualizacji różnorodnych typów treści trójwymiarowych, obejmujących modele fotogrametryczne, dane pochodzące z systemów CAD/BIM oraz chmur punktów.

Możliwości funkcjonalne Cesium 3D Tiles obejmują m.in.:

- Hierarchiczne organizowanie danych 3D w strukturę wielopoziomową zapewniającą adaptacyjne ładowanie treści (analogicznie do kafelków w serwisach mapowych)
- Wsparcie dla metadanych pozwalających na zaawansowane stylizowanie i filtrowanie obiektów 3D

Podczas realizacji projektu *IndoorLab* wykorzystano dwa źródła 3D Tiles:

- Google Photorealistic 3D Tiles¹²
- 3D Tiles okolicy budynków kampusu głównego Politechniki Warszawskiej przygotowane w Zakładzie Fotogrametrii, Teledetekcji i Systemów Informacji Przestrzennej Wydziału Geodezji i Kartografii PW

¹² Strona www usługi Google Photorealistic 3D Tiles:

<https://developers.google.com/maps/documentation/tile/3d-tiles> [dostęp: 10.01.2026]

Integracja z silnikiem Unreal Engine

W marcu 2021 roku Cesium udostępniło plugin dla silnika Unreal Engine, rozszerzając znacząco zakres zastosowań platformy poza środowiska webowe. Dodatek ten, dostępny jako oprogramowanie open source na licencji Apache 2, umożliwia wykorzystanie zaawansowanych możliwości wizualizacji Cesium w projektach opartych na tym silniku gier (Cesium 2021).

W kontekście przygotowanego symulatora, plugin Cesium dla Unreal Engine realizuje następujące kluczowe funkcjonalności:

Renderowanie trójwymiarowego otoczenia budynków: Plugin zapewnia wyświetlanie realistycznego tła, wykorzystując dane w formacie 3D Tiles.

Georeferencja modeli: Umożliwia precyzyjne osadzenie modeli budynków w globalnym układzie współrzędnych.

Transformacje układów współrzędnych: System wykonuje w czasie rzeczywistym przeliczenia pozycji uczestnika w świecie wirtualnym z lokalnego układu współrzędnych silnika na globalne współrzędne geograficzne.

Dynamiczne ładowanie treści: Implementuje algorytm adaptacyjnego ładowania modeli budynków w oparciu o aktualne położenie uczestnika, optymalizując w ten sposób wykorzystanie zasobów systemowych i zapewniając płynność działania aplikacji.

Zastosowanie Cesium for Unreal w znaczący sposób udoskonala powstały symulator, łącząc fotorealistyczną wizualizację Unreal Engine z danymi przestrzennymi platformy Cesium.

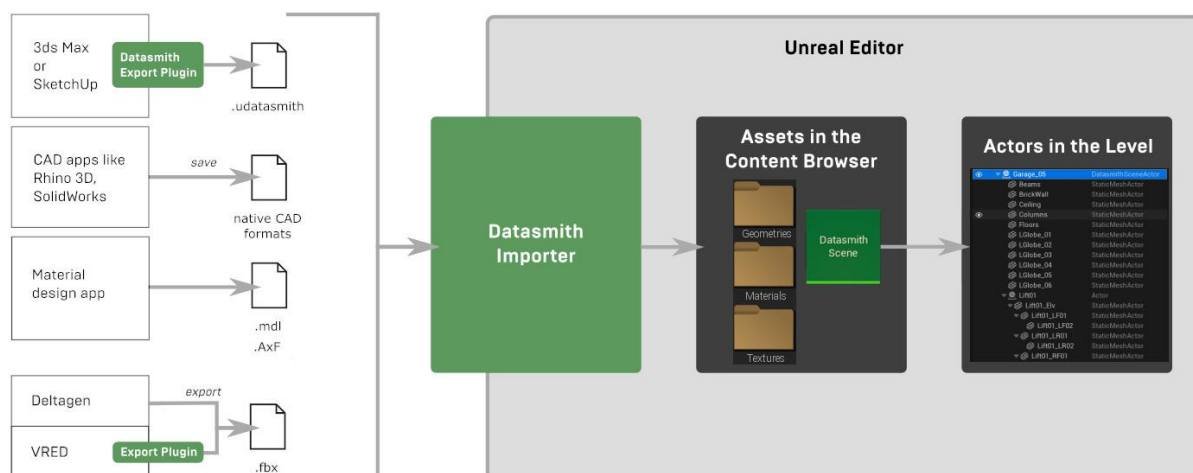
6.4. Modele budynków – proces pozyskiwania i aspekty wydajnościowe

Proces pozyskiwania trójwymiarowych modeli budynków stanowi zagadnienie złożone, wymagające integracji różnorodnych źródeł danych i zaawansowanych narzędzi cyfrowych. Pozyskanie wysokiej jakości modeli wymaga przejścia przez szereg etapów, począwszy od gromadzenia i standaryzacji danych, przez ich przetworzenie, aż po końcową implementację w środowisku symulacyjnym. W ramach pracy nad rozprawą skupiono się na tym końcowym

etapie, czyli wykorzystaniu gotowych już danych 3D w środowisku silnika gier. Opracowanie samych modeli trójwymiarowych zostało zrealizowane przez zespół pracowników Politechniki Warszawskiej w ramach projektu "Mapy Dostępności Budynków PW"¹³ będącego elementem ogólnouczelnianego projektu pt. "Politechnika Warszawska ambasadorem innowacji na rzecz dostępności". Mimo iż można w literaturze znaleźć propozycje algorytmów automatyzujących proces tworzenia modeli budynków (Abdollahi i in. 2023; Fan, Kong, i Zhang 2021), to nie są to jeszcze technologie na tyle dojrzałe by można ich było użyć na szeroką skalę. W związku z tym modele 3D zostały utworzone w procesie ręcznego modelowania klasycznymi narzędziami przeznaczonymi do tego celu.

W celu integracji otrzymanych modeli budynków z silnikiem Unreal Engine wykorzystano plugin DataSmith, umożliwiający import modeli trójwymiarowych z różnorodnych formatów wraz z ich podstawowymi właściwościami oraz teksturami. DataSmith obsługuje szeroką gamę formatów źródłowych, w tym natywne pliki CAD, pliki SketchUp, oraz modele Revit. DataSmith pozwala na przeniesienie złożonych struktur z narzędzi CAD/BIM bez konieczności ręcznego eksportu i importu geometrii, zapewniając tym samym zgodność i spójność modeli w środowisku symulacyjnym. Historia rozwoju narzędzia DataSmith sięga roku 2017, kiedy to Epic Games udostępniło ten plugin w odpowiedzi na rosnące potrzeby rynku w zakresie transferu modeli architektonicznych i projektów CAD do silnika Unreal Engine (Epic Games 2024a). W ramach procesu integracji wykorzystywano jest zarówno DataSmith Exporter po stronie aplikacji, w której zostały utworzone modele 3D, jaki i DataSmith Importer po stronie silnika Unreal.

¹³ Szczegółowy opis projektu oraz lista twórców są dostępne na stronie [www: https://www.gik.pw.edu.pl/Badania-i-nauka/Mapy-PW](https://www.gik.pw.edu.pl/Badania-i-nauka/Mapy-PW) [dostęp 10.01.2026]



Rysunek 6.7. Proces importu danych z wykorzystaniem pluginu DataSmith (Epic Games 2024a)

Po załadowaniu modeli budynków do Unreal Engine wzbogacono je o niezwykle istotny dla symulacji poruszania się wewnątrz budynku element – numery pomieszczeń (Rysunek 6.8). Wymagało to integracji nie zawsze idealnie spójnych danych źródłowych GIS i BIM. Opracowano więc w ramach rozprawy doktorskiej autorski komponent „RoomNumbersGenerator” dziedziczący po dostępnej w silniku Unreal klasie „EditorUtilityActorComponent”, który na podstawie współrzędnych punktu określającego numer pomieszczenia (pozyskanego z modelu budynku GIS) wyszukiwał najbliższe drzwi i dodawał do nich obiekt typu TextRenderer na odpowiedniej wysokości i z odpowiednią rotacją względem płaszczyzny drzwi. Ta automatyzacja była kluczowa dla prowadzonych badań oraz umożliwiła automatyczne oznaczenie numerem ponad siedmiu tysięcy drzwi.

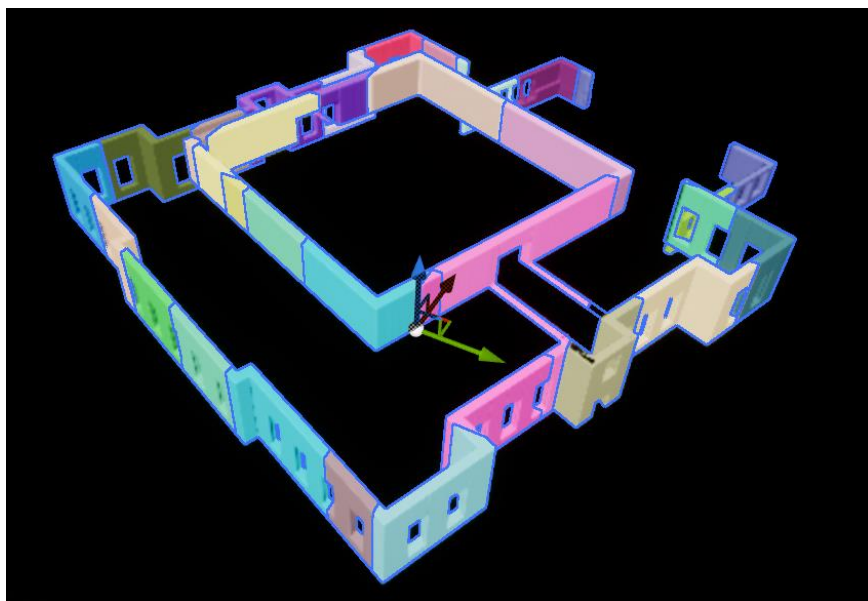


Rysunek 6.8. Numery pomieszczeń wyświetlane na drzwiach w modelu budynku Gmachu Elektroniki (opracowanie własne)

Analiza wydajności wyświetlania modeli budynków

Analiza wydajności wyświetlania modeli trójwymiarowych stanowiła jeden z kluczowych etapów procesu importu budynków do budowanego symulatora *IndoorLab*. Testowano trzy strategie organizacji geometrii:

- Reprezentacja całego budynku jako jednej, złożonej bryły.
- Automatyczny podział bryły na nieregularne, mniejsze fragmenty (Rysunek 6.9).
- Budowanie budynku z pojedynczych, prostych obiektów geometrycznych, osobno dla ścian, podłóg, drzwi, okien itp.



Rysunek 6.9. Przykład Gmachu Fizyki wymodelowanego jako jeden pojedynczy obiekt, który po zaimportowaniu automatycznie podzielono na mniejsze (opracowanie własne)

Strategie te porównywano pod kątem liczby rysowanych trójkątów, efektywności zarządzania pamięcią GPU oraz płynności renderowania w czasie rzeczywistym, kierując się literaturą z zakresu optymalizacji grafiki 3D, własnymi testami oraz rekomendacjami Epic Games. Ostatecznie w symulatorze wykorzystano tylko modele złożone z wielu prostych obiektów geometrycznych, a powody i wyjaśnienie tej decyzji opisano szczegółowo w rozdziale 6.5.

Ponadto autor na podstawie badań uznał, aby tam, gdzie to możliwe, wykorzystywać istniejące i znane metody optymalizacji renderowania 3D. Jedną ze szczególnie przydatnych technik w kontekście dużych i skomplikowanych budynków jest Instanced Static Mesh (ISM) (Epic Games 2024c). Technika ISM polega na tworzeniu wielu kopii tego samego obiektu geometrycznego, które są renderowane jako pojedyncza instancja w pamięci graficznej, co znacząco zmniejsza obciążenie GPU poprzez redukcję liczby wywołań rysowania sceny (ang. *draw calls*). ISM jest szczególnie efektywna w przypadku powtarzalnych obiektów, takich jak elementy konstrukcyjne, okna, drzwi czy kolumny, które występują wielokrotnie w obrębie modelu budynku. Dzięki temu technika ta pozwala na utrzymanie wysokiej wydajności grafiki w scenach o dużej złożoności geometrycznej, bez utraty jakości wizualnej (Tariq i in. 2021). Najlepiej stosować ISM tam, gdzie występuje duża liczba identycznych obiektów rozmieszczonych w różnych miejscach przestrzeni symulacji.

W procesie teksturowania najczęściej wykorzystywano bazowe tekstury zaimportowane z narzędzi SketchUp/Revit przez plugin DataSmith, co zapewniało szybki import modeli. Dodatkowo, w wybranych przypadkach testowano wykorzystanie biblioteki tekstur Quixel Megascans¹⁴, która oferuje darmowy dostęp do zaawansowanych, fotorealistycznych tekstur dla użytkowników Unreal Engine. Rozwiązanie to wyraźnie podnosiło jakość wyświetlanych modeli, ale z uwagi na czasochłonność implementacji nie zostało ostatecznie użyte dla wszystkich budynków.

Opracowane modele budynków poddano analizie pod kątem liczby obiektów geometrycznych oraz liczby trójkątów, co umożliwia ilościowe porównanie stopnia szczegółowości i skali projektu. Wszystkie wykorzystane w projekcie modele budynków wraz z dotyczącymi ich statystykami przedstawiono w załączniku nr 2.

¹⁴ Strona www biblioteki Quixel Megascans:
<https://quixel.com/megascans/> [dostęp 10.01.2026]

6.5. Wybór technologii renderowania grafiki trójwymiarowej

Z perspektywy efektywności renderowania grafiki trójwymiarowej należy wyodrębnić dwa fundamentalne aspekty techniczne: po pierwsze, strategie optymalizacji wyświetlania geometrii, a po drugie, metody obliczeniowe globalnego oświetlenia (ang. *Global Illumination*). Te komponenty stanowią podstawę współczesnych silników graficznych, determinując zarówno jakość wizualną, jak i wydajność tworzonej symulacji. W niniejszej pracy analizy związane z wyborem technologii przeprowadzono wyłącznie w kontekście wizualizacji modeli budynków.

6.5.1. Wyświetlanie złożonej geometrii

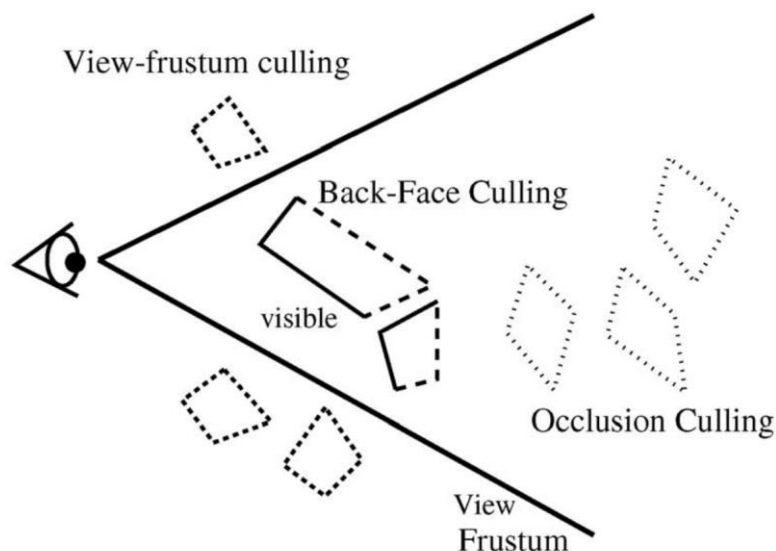
Model każdego budynku konstruowany jest z agregacji licznych, stosunkowo prostych elementów geometrycznych, które składają się na ściany, podłogi oraz sufity. Kluczowa decyzja projektowa polegała na przyjęciu zasady dekompozycji strukturalnej modelu budynku na możliwie najprostsze komponenty geometryczne i rezygnacji z reprezentacji budynku w postaci złożonego, pojedynczego modelu.

Takie podejście eliminuje znaczące problemy związane z procesem teksturowania oraz obliczeniami oświetleniowymi. Zbyt skomplikowane obiekty geometryczne utrudniają mapowanie tekstur na powierzchniach o złożonej topologii, a także komplikują obliczenia w algorytmach śledzenia promieni świetlnych i kalkulacji odbić.

Dekompozycja modeli architektonicznych na elementy składowe umożliwia pełne wykorzystanie zaawansowanych algorytmów optymalizacyjnych implementowanych w nowoczesnych silnikach graficznych. Wśród najistotniejszych technik optymalizacyjnych można wymienić (Cohen-Or i in. 2003):

Occlusion Culling - algorytm eliminujący z procesu renderowania obiekty zastonięte przez inne elementy geometryczne, co znacząco redukuje obciążenie obliczeniowe.

Frustum Culling - mechanizm odrzucający obiekty znajdujące się poza polem widzenia kamery, ograniczając obliczenia wyłącznie do geometrii potencjalnie widocznej dla obserwatora.



Rysunek 6.10. Ilustracja zasady działania technik Occlusion Culling oraz Frustum Culling (Cohen-Or i in. 2003)

Level of Detail (LOD) Management - system dynamicznego dostosowywania złożoności geometrycznej w zależności od odległości od kamery, gdzie obiekty odległe reprezentowane są przez modele o zredukowanej liczbie wielokątów.

Distance Culling - eliminacja obiektów przekraczających określony próg odległości, co zapobiega niepotrzebnemu przetwarzaniu elementów geometrycznych niewidocznych dla użytkownika.

W trakcie procesu projektowego rozważano wykorzystanie technologii **Nanite** (Rysunek 6.11), stanowiącej jedną z najnowocześniejszych innowacji w silniku Unreal Engine. Technologia ta umożliwia wirtualizację geometrii, pozwalając na efektywne wyświetlanie niezwykle szczegółowych modeli składających się z milionów trójkątów bez konieczności stosowania uproszczonych reprezentacji modeli o niższym LOD.



Rysunek 6.11. Przykłady różnych trybów wizualizacji w technologii Nanite w silniku Unreal Engine (Epic Games 2024d)

Przeprowadzone testy wydajnościowe wykazały, że w kontekście projektu *IndoorLab* będącego przedmiotem niniejszej rozprawy, charakteryzującego się renderowaniem dużej liczby stosunkowo prostych obiektów geometrycznych, technologia Nanite nie przynosiła oczekiwanej poprawy parametrów wydajnościowych. Ponadto, natura modeli budynków, składających się ze stosunkowo prostych brył geometrycznych, nie uzasadniała potrzeby tworzenia hierarchii LOD, co czyni główne zalety tej technologii nieistotnymi w tym przypadku użycia.

6.5.2. Renderowanie globalnego oświetlenia

Druga fundamentalna decyzja projektowa dotyczyła wyboru technologii generowania globalnego oświetlenia. Dostępne opcje obejmowały:

Oświetlenie statyczne (ang. *baked lighting*) – metoda polegająca na prekalkulacji map oświetleniowych (ang. *light maps*). Główną wadą tego rozwiązania jest znaczące zapotrzebowanie na zasoby czasowe i pamięciowe, ponieważ każdy obiekt geometryczny wymaga własnej mapy oświetleniowej, co prowadzi do nadmiernego wykorzystania pamięci tekstur. Praca z warstwą wizualną symulacji jest w tym przypadku spowolniona, ponieważ każda zmiana pozycji lub parametrów dowolnego obiektu lub źródła światła umieszczonego w

scenie wymaga ponownego wygenerowania map oświetlenia. Zaletą tego rozwiązania jest to, iż po załadowaniu tekstur do pamięci renderowanie obrazu jest najszybsze i najmniej obciążające wydajnościowo, przy zachowaniu wysokiej jakości obrazu. Metoda ta jest polecana np. w przypadku uruchamiania wizualizacji z wykorzystaniem gogli VR, które wymagają niezachwianego utrzymywania dużej liczby klatek na sekundę (Epic Games 2024e).

NVIDIA RTXGI – technologia będąca to implementacją tzw. DDGI (ang. *Dynamic Diffuse Global Illumination*), które polega na aktualizowaniu w czasie rzeczywistym siatki próbek świetlnych (ang. *irradiance probes*) za pomocą śledzenia promieni, które „zbierają” informację o świetle odbitym, aby następnie płynnie interpolować te dane na powierzchnie obiektów w scenie (Rysunek 6.12). Kluczem do wydajności tej technologii jest oderwanie procesu śledzenia promieni od każdego pojedynczego piksela na ekranie i przeniesienie go na rzadszą siatkę próbek. Zamiast obliczać oświetlenie globalne dla każdego fragmentu geometrii w każdej klatce (co byłoby niezwykle obciążające), silnik graficzny pobiera gotowe, uśrednione informacje z najbliższych sond. Dzięki temu uzyskuje się efekt wielokrotnego odbicia światła i dynamicznej reakcji na zmiany otoczenia przy zachowaniu wysokiej płynności animacji.



Rysunek 6.12. Wizualizacja technologii RTXGI pozwalającej na generowanie globalnego oświetlenia w oparciu o regularnie rozmieszczone próbki (opracowanie własne)

Wstępne implementacje tej technologii zostały przeprowadzone w ramach przygotowania symulatora *IndoorLab*, wykazując pozytywne rezultaty w zakresie wydajności i jakości wizualnej. Mimo to zrezygnowano z jej wykorzystania z uwagi na fakt, iż rozwój pluginu implementującego tę technologię dla silnika Unreal Engine został przerwany (NVIDIA 2023), a ostatnią dostępną jest wersja dla silnika w wersji 5.0.

Lumen – to technologia globalnego oświetlenia zaimplementowana bezpośrednio w silniku Unreal Engine, stanowiąca integralną część jego systemu renderowania. Ostatecznie to właśnie ta technologia została wybrana jako optymalne rozwiązanie dla omawianego projektu *IndoorLab*. System Lumen funkcjonuje poprzez kombinację dynamicznego śledzenia promieni oraz wykorzystania techniki Surface Cache (pamięci podręcznej powierzchni), która przechowuje informacje o oświetleniu bezpośrednim na powierzchniach geometrycznych w czasie rzeczywistym. Technologia ta implementuje algorytm wieloskokowego odbijania światła, gdzie każdy promień może wykonywać określoną liczbę odbić od powierzchni, umożliwiając realistyczne symulowanie zjawisk takich jak color bleeding (przenikanie kolorów) oraz soft shadows (miękkie cienie) bez konieczności pre-kalkulacji map oświetleniowych.

Zastosowanie systemu Lumen zapewniło wysoką jakość oświetlenia przestrzeni wirtualnej, jednocześnie oferując znaczną łatwość w procesie rozwoju projektu *IndoorLab* i modelowania środowiska wirtualnego. Technologia ta zapewniła wystarczającą wydajność zarówno dla wersji uruchamianej na komputerze uczestnika, jak i wersji renderowanej na serwerach chmury obliczeniowej (więcej w rozdziale 6.7).

6.6. Implementacja mechanizmów symulacji ruchu użytkownika

W ramach realizowanego środowiska badawczego opracowano dwa odrębne tryby symulacji ruchu użytkownika (poruszania w przestrzeni wirtualnej), dostosowane do zróżnicowanych kompetencji cyfrowych uczestników eksperymentu (Rysunek 6.13). Przyjęte rozwiązanie zaprojektowano tak, aby zminimalizować wpływ czynników związanych z poziomem doświadczenia w korzystaniu z gier komputerowych na jakość danych pozyskiwanych w trakcie zaplanowanych badań.



Rysunek 6.13. Ekran wyboru trybu sterowania (opracowanie własne)

Tryb standardowy

Pierwszy z zaimplementowanych trybów opiera się na konwencjach sterowania charakterystycznych dla gier z gatunku tzw. pierwszoosobowych strzelanek (ang. *First Person Shooter*, FPS). Mechanizm ten zakłada kontrolę orientacji kamery poprzez wykorzystanie urządzenia wskazującego (myszy komputerowej), natomiast przemieszczanie uczestnika realizowane jest za pośrednictwem naciskania klawiszy WASD na klawiaturze. Powyższe rozwiązanie bazuje na domyślnej implementacji systemu sterowania dostępnej w ramach przykładów silnika Unreal Engine, która została zaadaptowana do specyficznych wymagań środowiska badawczego. Udostępnienie tego trybu naturalnie wynikało z jego powszechnego stosowania w grach komputerowych oraz intuicyjności dla użytkowników posiadających doświadczenie w tym zakresie.

Tryb uproszczony

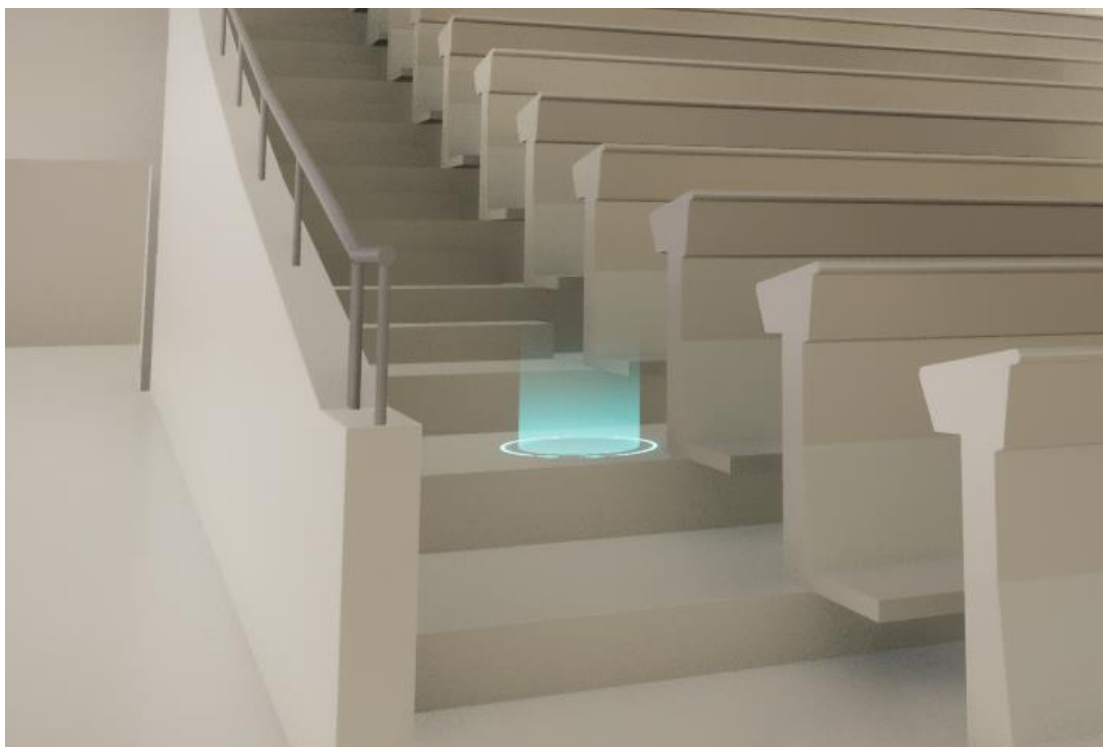
Drugi tryb został zaprojektowany z myślą o uczestnikach nieposiadających wcześniejszego doświadczenia z grami komputerowymi (lub bardzo ograniczone), mających trudności z opanowaniem konwencjonalnych mechanizmów sterowania. Tryb uproszczony bazuje na mechanice analogicznej do tej znanej z gier typu wskaż i kliknij (ang. *point-and-click*), gdzie

użytkownik wskazuje kursorem myszy lokalizację docelową w przestrzeni wirtualnej, a następnie poprzez kliknięcie inicjuje automatyczne przemieszczenie do wybranego punktu.

Implementacja tego rozwiązania wiązała się z koniecznością rozwiązania szeregu problemów technicznych. Pierwszym wyzwaniem okazało się precyzyjne przeniesienie pozycji kursora z dwuwymiarowego obszaru ekranu na współrzędne w trójwymiarowej przestrzeni wirtualnej. Proces ten wymagał zastosowania technik rzutowanie promienia (ang. *raycasting*) w celu określenia punktu przecięcia promienia rzuconego z pozycji kamery w kierunku wskazanym przez kursor z geometrią sceny. Następnie niezbędne było zrzutowanie uzyskanej pozycji na powierzchnię podłogi, a także odsunięcie uzyskanej lokalizacji od najbliższych przeszkód co zapewniało prawidłowe umiejscowienie punktu docelowego na płaszczyźnie poruszania się użytkownika.

Drugim istotnym zagadnieniem było wyznaczenie optymalnej ścieżki przemieszczenia do wskazanego punktu z uwzględnieniem złożoności architektonicznej budynków, w których odbywało się badanie. Proces ten wykorzystywał wbudowany w silnik Unreal Engine system nawigacji oparty na algorytmach pathfindingu, który jednak wymagał wcześniejszego przygotowania tzw. Navigation Mesh (NavMesh). **Generowanie NavMesh dla tak skomplikowanych struktur architektonicznych, jakimi charakteryzowały się modelowane budynki, stanowiło znaczące wyzwanie techniczne.** Konieczne było zapewnienie odpowiedniej granularności siatki nawigacyjnej, która z jednej strony umożliwiałaby płynne przemieszczanie się po wszystkich dostępnych powierzchniach, a z drugiej nie generowała nadmiernego obciążenia obliczeniowego systemu.

Dodatkowo implementacja trybu uproszczonego wymagała opracowania mechanizmów wizualnej informacji zwrotnej (ang. *visual feedback*), które informowały użytkownika o dostępności wybranego punktu docelowego. Zastosowano system podświetlania spodziewanego punktu docelowego (Rysunek 6.14).



Rysunek 6.14. Zrzut ekranu prezentujący podświetlenie spodziewanego punktu docelowego w uproszczonym trybie sterowania (opracowanie własne)

Oba tryby symulacji ruchu zostały poddane testom pilotażowym (rozdział 6.12) w celu weryfikacji ich funkcjonalności oraz przydatności w kontekście planowanych badań percepcji nawigacji wewnątrz budynków. Wyniki tych testów potwierdziły zasadność implementacji dwóch odrębnych modalności sterowania jako skutecznego sposobu na neutralizację wpływu wcześniejszego doświadczenia użytkowników z grami komputerowymi na rezultaty badań.

W finalnym badaniu ponad 68% uczestników wybrało standardowy tryb sterowania.

Szczegółowa analiza wyborów uczestników została przeprowadzona w rozdziale 7.4.

6.7. Warianty dostępu do środowiska symulacyjnego – aspekty techniczne i użytkowe

Opierając się na doświadczeniach zdobytych podczas pierwszej próby wykorzystania środowiska symulacyjnego w badaniach nad nawigacją wewnątrz budynków (Łobodecki 2020), zidentyfikowano istotne ograniczenie związane z dostępnością badania. Konfiguracja wymagająca instalacji i uruchomienia aplikacji na urządzeniu uczestnika znacząco ograniczała liczbę potencjalnych osób badanych. Bariery okazały się: po pierwsze, wymogi sprzętowe, które obejmowały konieczność posiadania wydajnego komputera wyposażonego w wysokiej klasy kartę graficzną, a po drugie, wysokie bariery psychologiczne związane z naturalną niechęcią użytkowników do pobierania dużych plików oraz instalowania nieznanych aplikacji na własnych urządzeniach. **W odpowiedzi na te ograniczenia zaproponowano alternatywną metodę dostępu do środowiska symulacyjnego, wykorzystującą technologię Pixel Streaming.** Technologia Pixel Streaming stanowi zaawansowane rozwiązanie oferowane przez Epic Games w ramach silnika Unreal Engine, umożliwiające przekazywanie interaktywnych aplikacji 3D z serwera obliczeniowego bezpośrednio do przeglądarek internetowych użytkowników. Konceptyjnie, system ten reprezentuje ewolucję w podejściu do dystrybucji treści graficznych o wysokiej jakości, eliminując konieczność instalacji aplikacji na urządzeniach końcowych (Feng i in. 2024).

6.7.1. Dostęp do systemu badawczego

Wraz z informacją i zaproszeniem do udziału w badaniu uczestnik dostawał link, prowadzący do strony internetowej umożliwiającej udział w badaniu dostępnej pod adresem:

<https://mapa.pw.edu.pl/indoorlab/>

(dostępność przynajmniej do końca roku 2026)

Na stronie startowej podano informacje dotyczące przebiegu badania, takie jak: cel badania, zastosowanie, czas trwania badania, metodyka oraz informacja o autorze badania. Zamieszczono także zachętę do zaproszenia kolejnych osób do udziału, a także zrzuty ekranu przedstawiające przebieg badania (Rysunek 6.15).

Symulator nawigacji wewnątrz budynków

English

O Projekcie

Cel badania: Analiza użyteczności aplikacji nawigacyjno-lokalizacyjnych w czasie poszukiwania pomieszczeń w skomplikowanych budynkach.

Zastosowanie: Wnioski z badań pomogą w opracowaniu bardziej intuicyjnego oznakowania oraz systemów wspomagających nawigację wewnątrz budynków publicznych (centra handlowe, szpitale, lotniska).

Czas trwania badania: 15-30 minut

Metodyka: Uczestnik badania wykorzystuje symulator, w którym porusza się po cyfrowym bliźniaku istniejącego budynku. Celem jest jak najszybsze dotarcie do wskazanego pomieszczenia z użyciem aplikacji nawigacyjnej. System rejestruje ścieżkę poruszania się uczestnika, czas dotarcia do celu oraz wykorzystane wskazówki nawigacyjne, dzięki czemu możliwe będzie porównanie skuteczności różnych metod nawigowania.

Autor: Jakub Łobodecki, Politechnika Warszawska
jakub.lobodecki@pw.edu.pl

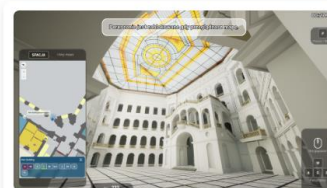
Aby uzyskać wiarygodne wyniki, potrzebujemy kilkuset uczestników. Będę ogromnie wdzięczny, jeśli przekażesz link rodzinie i znajomym – każda osoba ma znaczenie!

<https://mapa.pw.edu.pl/indoorlab/>

Kopiuj Link

Dziękuję za pomoc i wsparcie projektu!

Zapraszam do symulacji poniżej!



Rysunek 6.15. Pierwsza część strony dostępnej systemu badawczego, zawierająca charakterystykę badania (opracowanie własne)

W drugiej części strony dostępnej (Rysunek 6.16) wyświetlono login i hasło potrzebne do logowania w dalszym etapie, a także przycisk przenoszący na stronę logowania. Takie rozwiązanie kwestii autoryzacji było konieczne z uwagi na charakterystykę zabezpieczeń chmury obliczeniowej CENAGIS, używanej do serwowania symulatora *IndoorLab*. Jednocześnie konieczność zalogowania zmniejsza ryzyko uruchamiania symulacji przez roboty indeksujące strony internetowe, co mogłoby generować niepotrzebne obciążenie i dodatkowe koszty.

Logowanie do symulatora ruchu w budynkach

Dane Logowania

Nazwa użytkownika:	Hasło:
showroom	BUb\$4ceWvPC#
Kopiuj	Kopiuj

Bezpośredni Link do Symulacji:

*Symulacja działa tylko na komputerach PC.
Nie jest kompatybilna ze smartfonami i tabletami.*

[Otwórz Symulację w Nowej Karcie](#)

*Uwaga: Ze względu na ograniczenia bezpieczeństwa,
symulacja musi zostać otwarta w nowej karcie. Użyj
powyższych danych logowania.*

Rysunek 6.16. Druga część strony dostępowej systemu badawczego, zawierająca dane dostępowe oraz przycisk przenoszący na stronę logowania (opracowanie własne)

Po zalogowaniu uczestnikom prezentowany jest ekran wyboru wariantu dostępu do środowiska symulacyjnego, z przedstawieniem zalet i wad każdego z nich (Rysunek 6.17). W dwóch kolejnych podrozdziałach opisano szczegółowo specyfikę każdego z wariantów.



Rysunek 6.17. Ekran wyboru wariantu A i B dostępu do aplikacji badawczej (opracowanie własne)

W finalnym badaniu prawie 88% uczestników wybrało wariant przesyłania obrazu z chmury.

Szczegółowa analiza wyborów uczestników została przeprowadzona w rozdziale 7.4.

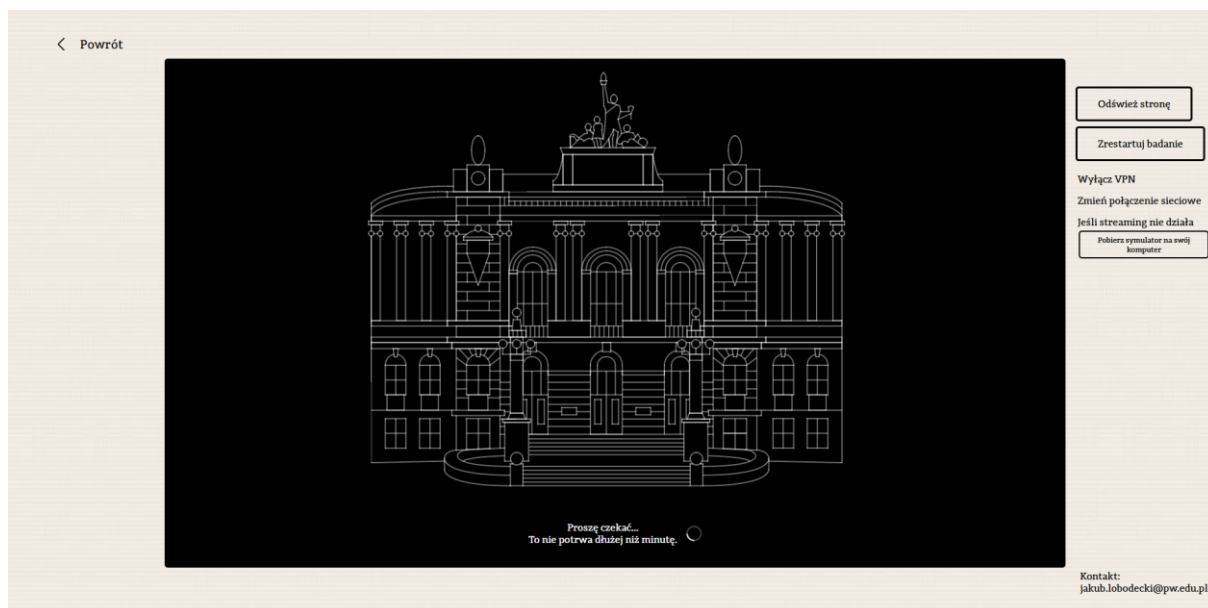
6.7.2. Wariant A – przesyłanie obrazu z chmury

Podstawowym założeniem wariantu A jest to, że obraz renderowany jest w chmurze obliczeniowej. Aby uzyskać do niego dostęp wystarczy przeglądarka internetowa, a uzyskany rezultat przypomina interaktywne wideo. Rozwiązanie to jednak ma też szereg wad (Tabela 4), takich jak opóźnienie interakcji użytkownika względem transmisji czy niższa jakość będąca wynikiem koniecznej kompresji obrazu.

Tabela 4. Wady i zalety przesyłania obrazu z chmury (opracowanie własne)

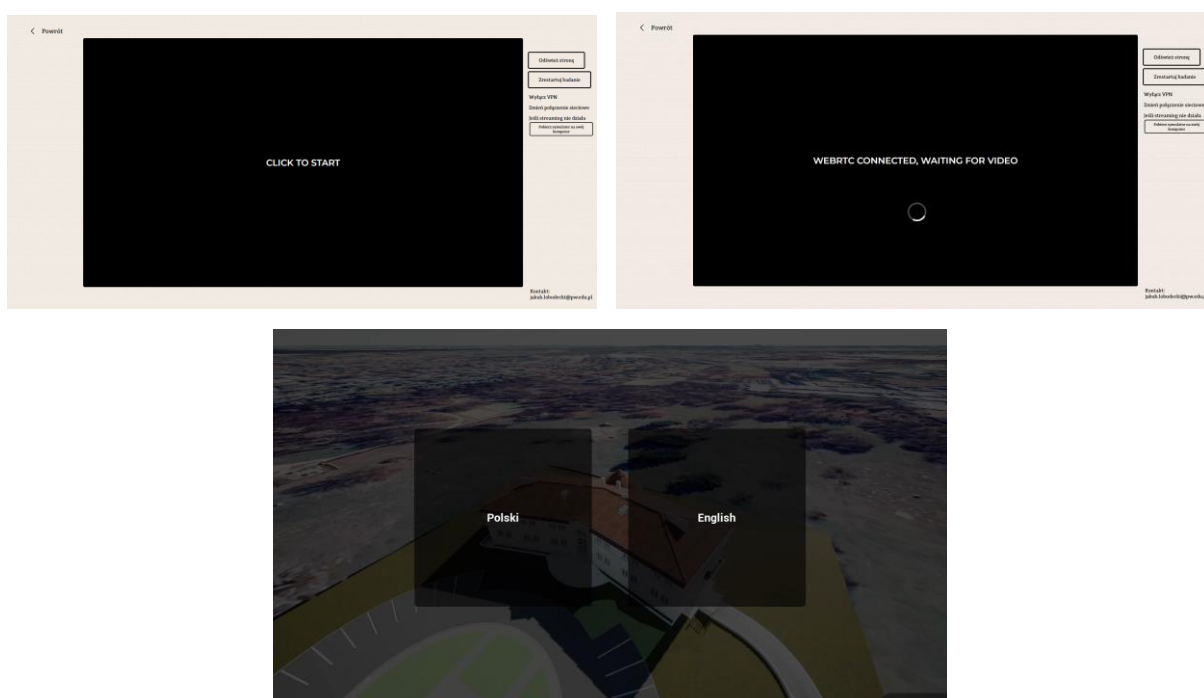
Zalety	Wady
• Uruchomienie symulacji i nawiązanie połączenia trwa tylko kilka sekund • Symulacja działa również na komputerach o mniej wydajnej specyfikacji sprzętowej, niewyposażonych w kartę graficzną • Nie jest wymagane pobieranie ani instalowanie dodatkowych plików lub aplikacji	• Wymagane jest szerokopasmowe połączenie internetowe o możliwie najniższym opóźnieniu (ang. <i>latency</i>) • Obniżona jakość obrazu ze względu na kompresję przesyłanego wideo • Występowanie opóźnień pomiędzy interakcją uczestnika, a zaobserwowaniem efektu działania na ekranie • Ustanowienie połączenia może być niemożliwe w niektórych konfiguracjach sieciowych (np. zabezpieczone sieci firmowe)

Interfejs strony dostępowej składa się przede wszystkim z umieszczonej centralnie ramki ładowania symulacji, w której później w obiekcie typu *iframe* osadzony jest właściwy streaming (Rysunek 6.18). Po jego lewej stronie dostępny jest przycisk powrotu do ekranu wyboru preferowanego wariantu. Z uwagi na ewentualność wystąpienia problemów z nawiązaniem połączenia, po prawej stronie umieszczono dostęp do dodatkowych funkcji, które mogą być pomocne w rozwiązaniu napotkanych problemów.



Rysunek 6.18. Ekran ładowania trybu "Przesyłanie z chmury" (opracowanie własne)

Po załadowaniu wyświetlana jest zachęta do interakcji o treści „Click to start”. Kliknięcie w umieszczoną centralnie ramkę symulacji powoduje nawiązanie połączenia z serwerem i uruchomienie streamingu w widoku pełnoekranowym (Rysunek 6.19).



Rysunek 6.19. Proces nawiązywania połączenia z serwerem streamingu (opracowanie własne)

Po połączeniu, zakładając jego stabilne i poprawne działanie symulacja zachowuje się bardzo podobnie do wariantu uruchomionego na komputerze uczestnika. Istotnym elementem, który

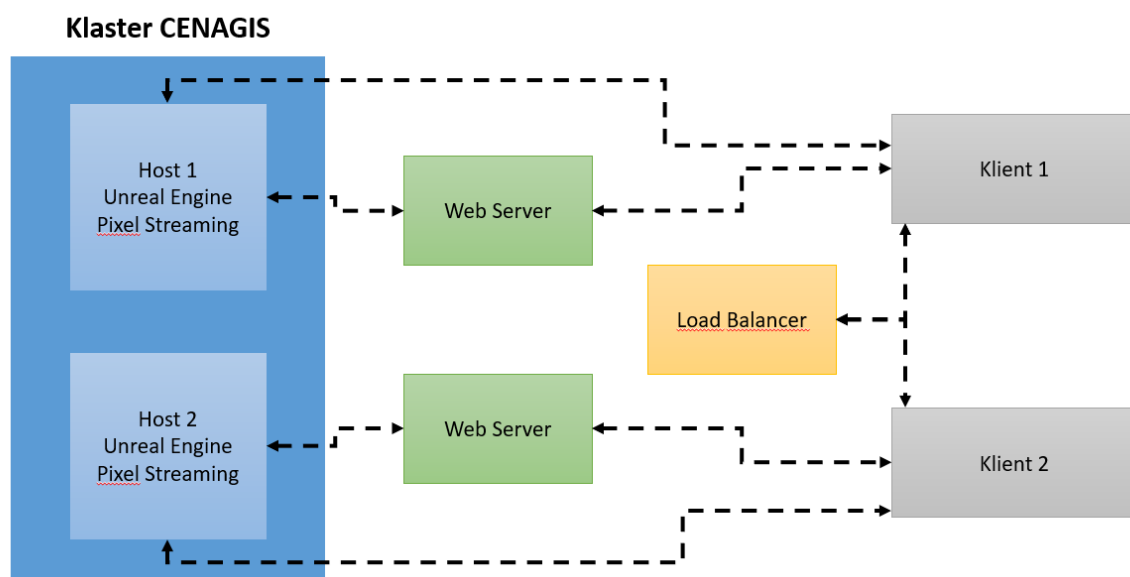
wymagał dodatkowego oprogramowania był aspekt przechwytywania i uwalniania kursora myszy przez przeglądarkę internetową, który powinien odpowiadać aktualnemu stanowi interfejsu symulacji. Ponadto zaimplementowano dodatkowe mechanizmy kontroli połączenia, których zadaniem było automatyczne ponowne nawiązanie połączenia w rzadkich przypadkach jego zerwania, spowodowanego najczęściej niestabilnością sieci po stronie uczestnika.

Techniczne aspekty technologii Pixel Streaming

Architektura Pixel Streaming opiera się na kilku kluczowych komponentach technologicznych. Centralnym elementem jest Rendering Server, który wykorzystuje wysokowydajne karty graficzne do przetwarzania i renderowania treści 3D w czasie rzeczywistym. Protokół WebRTC pełni funkcję protokołu komunikacyjnego, zapewniając transmisję renderowanych klatek oraz obsługę sygnałów sterujących pochodzących od użytkowników. Dodatkowo, Signaling Server zarządza procesem ustanawiania połączeń między klientem a serwerem renderującym, koordynując wymianę informacji niezbędnych do nawiązania sesji WebRTC.

Ze względu na fakt, że dokumentacja Unreal Engine nie precyzuje sposobu skalowania aplikacji wykorzystujących Pixel Streaming, **w ramach rozprawy doktorskiej opracowano własny system dynamicznego uruchamiania instancji serwerów renderujących w zależności od zapotrzebowania.**

Opracowane rozwiązanie jest zbliżone do przedstawionego w artykule „An open architecture for scalable Pixel Streaming” w sekcji 2.3.1. „Scenario 1: Controlling Peer only” (Rehn i Birmingham 2021). Poniżej przedstawiono diagram architektury sieciowej komponentów biorących udział w uruchamianiu Pixel Streamingu (Rysunek 6.20).



Rysunek 6.20. Diagram architektury konfiguracji sieciowej wdrożonej na potrzeby technologii Pixel Streaming (opracowanie własne)

Na potrzeby tego rozwiązania wykorzystano podsystem Big Data infrastruktury CENAGIS (Gotlib, Choromański, i Kaczalek 2022), który pozwala na uruchamianie usług w skonteneryzowanym¹⁵ środowisku.

Kluczowym elementem architektury konteneryzacji jest DC/OS (Data Center Operating System), funkcjonujący jako open-source'owy system operacyjny bazujący na Apache Mesos. W przypadku Platformy IT CENAGIS umożliwia on autonomiczne zarządzanie klastrem składającym się z 30 serwerów (łącznie 1800 vCPU, 11 TB RAM oraz 60 akceleratorów GPU NVIDIA Tesla T4) zapewniając zaawansowaną orkiestrację kontenerów Docker, w których mogą być uruchamiane instancje aplikacji wykorzystującej Pixel Streaming. Dzięki temu każde żądanie użytkownika może być dynamicznie mapowane na odpowiednią instancję kontenerową, przydzielając zasoby obliczeniowe oraz GPU zgodnie z aktualnym obciążeniem, co pozwala na elastyczne skalowanie usługi oraz optymalizację wydajności i kosztów eksploatacyjnych. W tym środowisku możliwe jest równoczesne obsługiwane 60 użytkowników (1 karta graficzna na każdego użytkownika) zakładając, że klaster nie jest obciążony innymi zadaniami.

¹⁵ Środowisko skonteneryzowane to odizolowane środowisko uruchomieniowe, które zawiera kod aplikacji wraz z jej zależnościami (biblioteki, konfiguracje), współdzieląc jądro systemu operacyjnego hosta, ale izolując procesy aplikacji (Wikipedia 2025a).

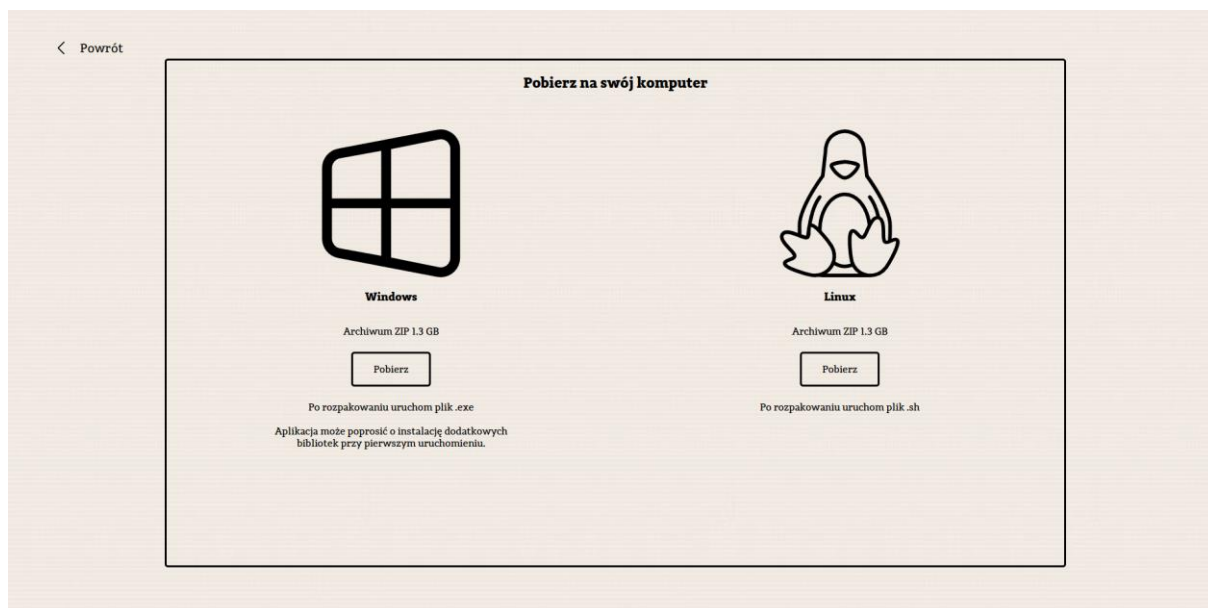
6.7.3. Wariant B – uruchamianie aplikacji na komputerze desktop

Ten wariant zakłada, że uczestnik jest wyposażony w komputer o odpowiedniej wymaganej specyfikacji, która umożliwi płynne działanie systemu symulacyjnego. Tabela 5 przedstawia wady i zalety tego rozwiązania.

Tabela 5. Wady i zalety uruchomienia aplikacji badawczej na komputerze uczestnika (opracowanie własne)

Zalety	Wady
• Symulacja prezentuje wyższą jakość renderowanej grafiki • Działania uczestnika mają natychmiastowe odwzorowanie w wirtualnym świecie – nie występuje opóźnienie • Wystarczające jest połączenie internetowe o ograniczonej prędkości	• Wymagany jest komputer o odpowiedniej specyfikacji technicznej, posiadający wydajną kartę graficzną • Przed uruchomieniem aplikacji konieczne jest pobranie potrzebnych plików oraz doinstalowanie brakujących bibliotek

Aplikacja badawcza została skompilowana dla dwóch systemów operacyjnych tj. Windows oraz Linux. W obu przypadkach formą dostarczenia jest link prowadzący do archiwum .zip zawierającego pliki potrzebne do uruchomienia symulatora *IndoorLab*. Rozmiar archiwów to około 1,3 GB.



Rysunek 6.21. Ekran prezentujący przyciski prowadzące do pobierania potrzebnych plików (opracowanie własne)

Aby uruchomić pobrany projekt komputer powinien spełniać następujące minimalne wymagania sprzętowe (Epic Games 2024b):

- System operacyjny: Windows 10 64-bit
- Procesor: Quad-core Intel lub AMD 2.5 GHz lub szybszy
- Pamięć RAM: 8 GB
- Karta graficzna: zgodna z DirectX 11 lub 12

6.8. Wybór metody wyświetlania mapy wewnątrz budynków

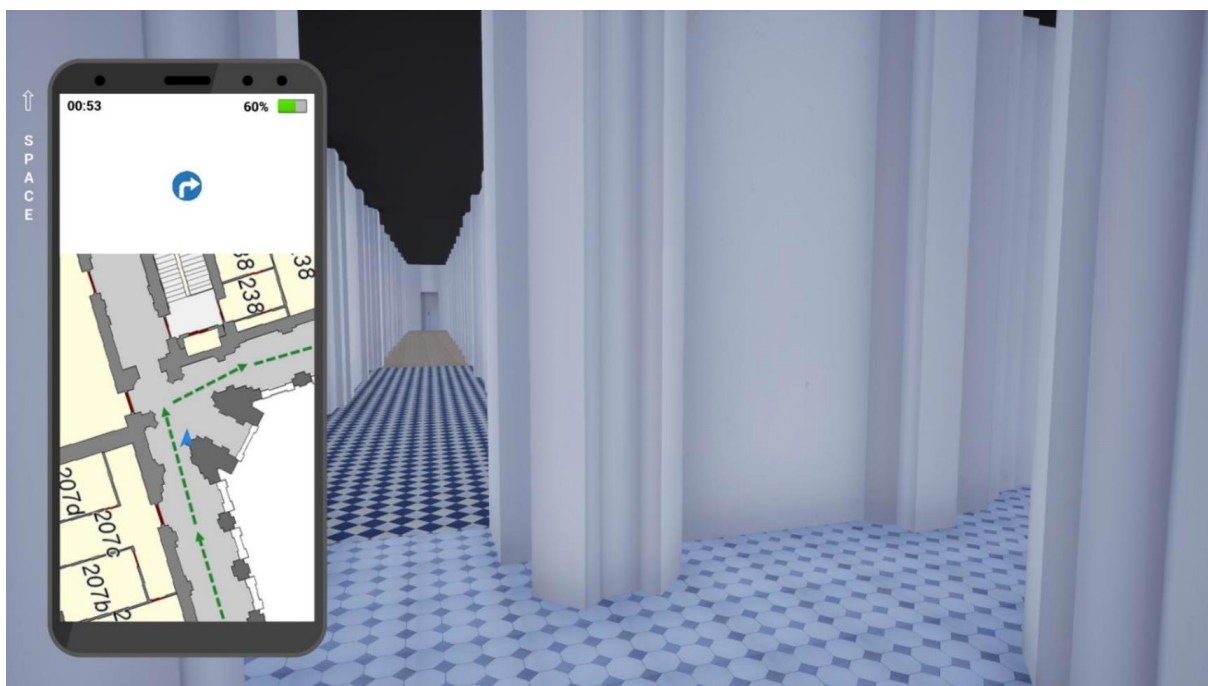
W celu realizacji założonych celów badawczych, już na etapie wstępnego planowania badań przewidziano konieczność implementacji systemu prezentacji mapy nawigacyjnej podczas przemieszczania się użytkownika w wirtualnym budynku. **Koncepcja technicznej realizacji tego zadania przechodziła znaczące transformacje na przestrzeni rozwoju symulatora *IndoorLab*, odzwierciedlając ewolucję myśli badawczej oraz postęp w zakresie dostępnych rozwiązań technologicznych.**

W niniejszym rozdziale przedstawiono przegląd rozważanych mechanik prezentacji mapy wraz z analizą ich potencjału badawczego, a także omówiono kluczowe aspekty techniczne integracji

silnika gry z aplikacją nawigacyjną. Szczegółowa charakterystyka samej aplikacji oraz jej funkcjonalności została zaprezentowana w rozdziale 0.

6.8.1. Metoda nr 1 (odrzucona)

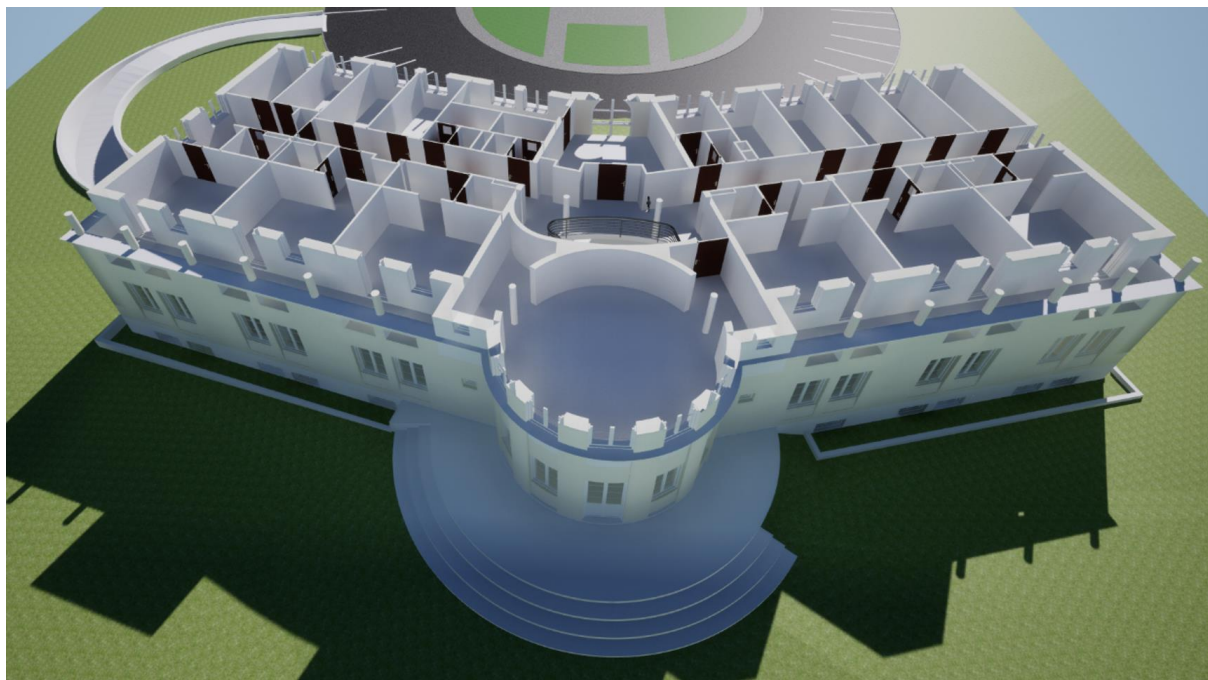
Zastosowane w początkowym etapie badań podejście opierało się na implementacji statycznych wizualizacji kartograficznych, na które nakładano informację o zdefiniowanej wcześniej trasie oraz aktualnym położeniu uczestnika badania (Rysunek 6.22). Rozwiązanie to, choć relatywnie proste w realizacji technicznej, okazało się niewystarczające z perspektywy badawczej, głównie ze względu na ograniczoną interaktywność oraz konieczność ręcznego opracowania szeregu wizualizacji i tras, co ograniczałoby możliwości ponownego wykorzystania symulatora w przyszłości. Tego typu rozwiązanie zastosowano i szczegółowo opisano w pracy magisterskiej autora (Łobodecki 2020), w ramach której powstał prototyp narzędzia badawczego.



Rysunek 6.22. Ilustracja wyglądu mapy w prototypie systemu (Łobodecki 2020)

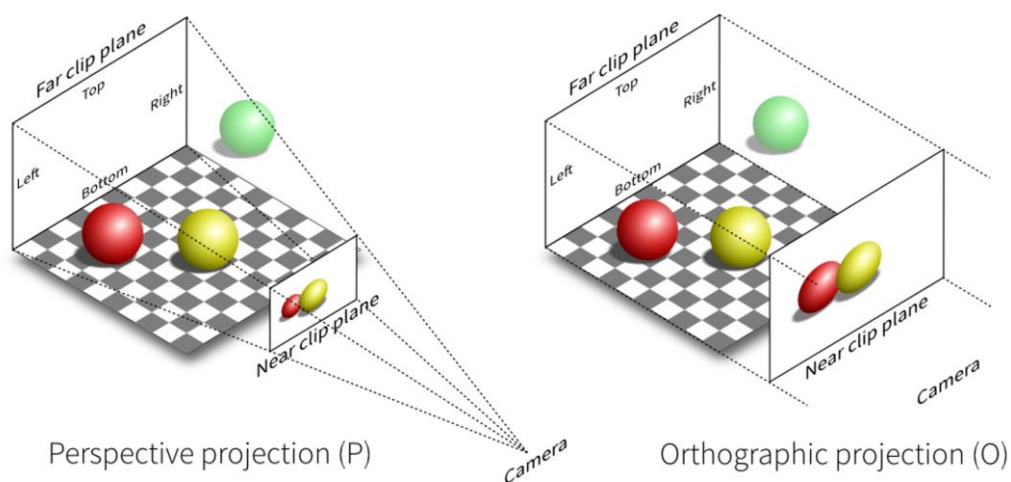
W kolejnej fazie badań podjęto próbę generowania map w czasie rzeczywistym na podstawie trójwymiarowych modeli budynków osadzonych już w silniku gry, wykorzystując podejście analogiczne do mechanizmów implementowanych zazwyczaj w tzw. „minimapach” stosowanych w grach komputerowych. Wykorzystywano w tym celu mechanizm wyświetlania

przekroju budynku, co jest możliwe poprzez ukrycie (maskowanie) jego części. Technicznie operacja ta odbywa się poprzez manipulację danymi wykorzystywanymi przez silnik renderujący do wyświetlania wszystkich elementów budynku.

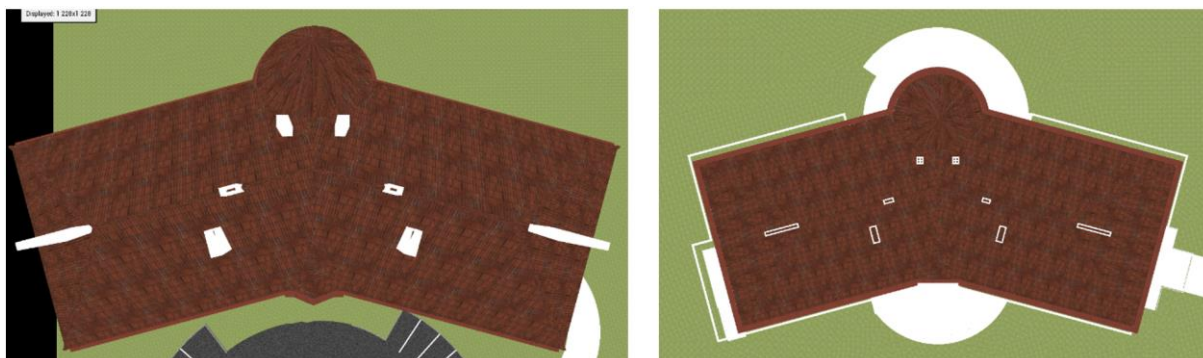


Rysunek 6.23. Widok przekroju budynku (opracowanie własne)

Przechwycenie widoku przekroju poziomego celem wygenerowania na tej podstawie mapy 2D odbywa się poprzez odpowiednio skonfigurowany tzw. obiekt kamery. Przede wszystkim dla uzyskania wiernego odwzorowania należy skorzystać z widoku ortogonalnego, a nie perspektywicznego, używanego standardowo w grafice 3D (Rysunek 6.24, Rysunek 6.25).

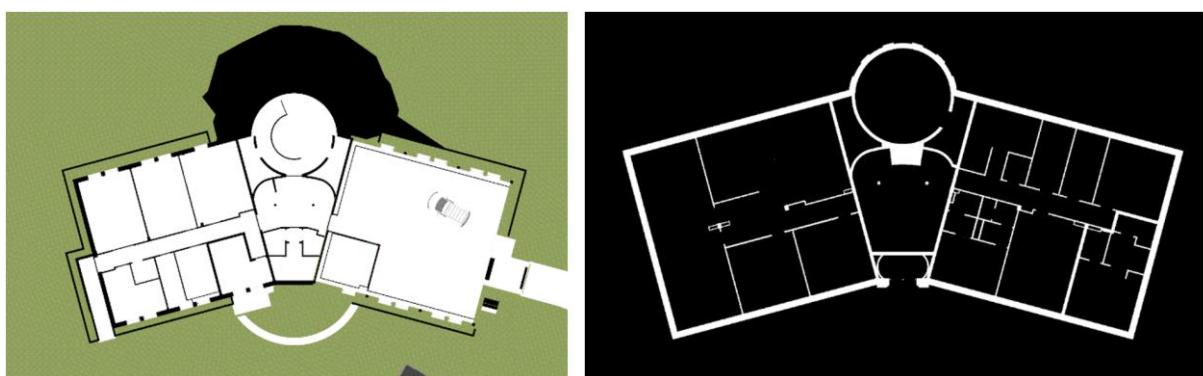


Rysunek 6.24. Różnica pomiędzy projekcją perspektywiczną, a ortograficzną (Schim 2019)



Rysunek 6.25. Porównanie projekcji perspektywicznej (obraz po lewej) i ortograficznej (obraz po prawej) (opracowanie własne)

Na kadr przechwycony wirtualną kamerą można nałożyć filtry i kontynuować przetwarzanie stosując klasyczne operacje przetwarzania obrazów. Rysunek 6.26 prezentuje zastosowanie filtrów w taki sposób, aby otrzymać zarys ścian budynku, na podstawie którego można utworzyć schematyczną mapę wnętrza.



Rysunek 6.26. Po lewej obraz przekroju poziomego przez budynek o wysokim poziomie ekspozycji. Po prawej efekt działania inwersji kolorów na przechwyconym obrazie (opracowanie własne)

Utworzona w taki sposób mapa 2D budynku została następnie osadzona w interfejsie użytkownika z wykorzystaniem dodatkowego pluginu Journeyman's Minimap (Shao 2022). Odpowiada on za aktualizowanie wyświetlanego fragmentu mapy w czasie rzeczywistym, a także nakładanie na mapę dodatkowych informacji, takich jak aktualna lokalizacja czy oznaczenie punktów charakterystycznych.

Główną zaletą takiego rozwiązania jest fakt, że raz przygotowane nie wymagałyby dalszej ingerencji wraz z opracowywaniem kolejnych scenariuszy badawczych. Wyzwaniami pozostawała jednak kwestia wyświetlania trasy na przygotowanej mapie oraz ograniczone możliwości interakcji i stylizacji wizualizacji kartograficznej. Wyświetlona mapa wciąż byłaby

zbytнім uproszczeniem względem mapy wyświetlanej w rzeczywistej aplikacji nawigacyjnej. W związku z powyższym ta metoda została w całości odrzucona, na rzecz metody nr 2.

6.8.2. Metoda nr 2 (wybrana)

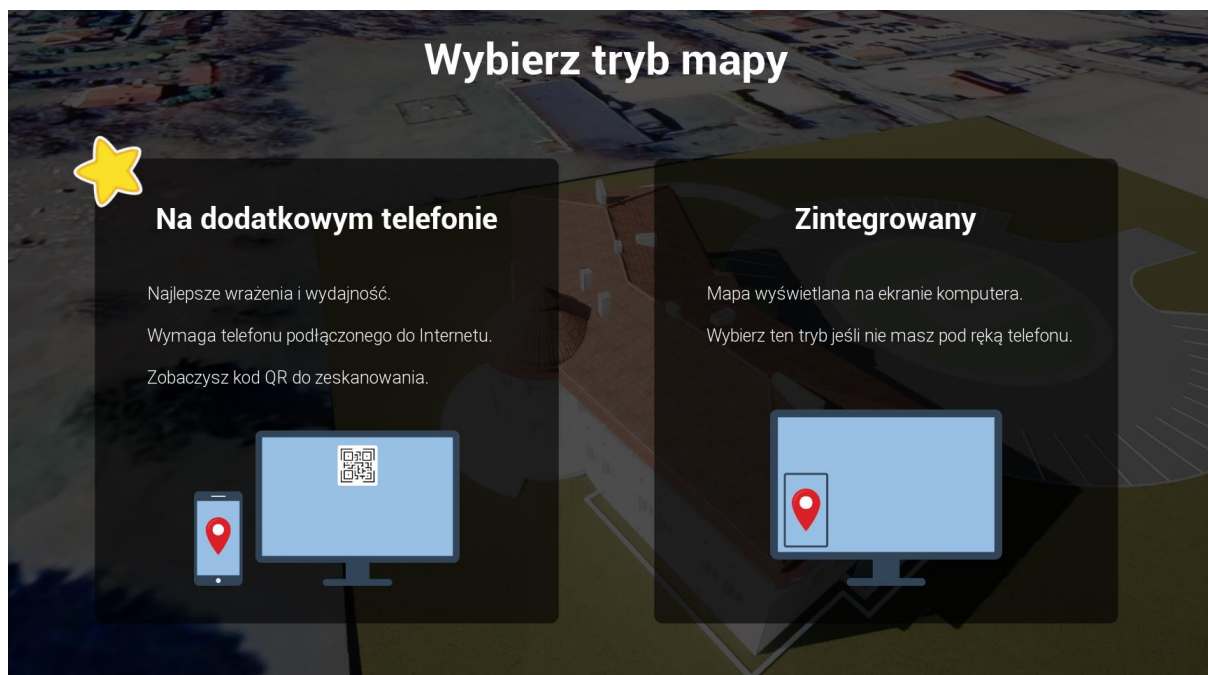
Ostatecznym rozwiązaniem, które zostało zastosowane w przeprowadzonym badaniu, była pełna integracja z rzeczywistością, funkcjonującą i niezależną aplikacją lokalizacyjno-nawigacyjną, która została stworzona w ramach innego projektu realizowanego w Politechnice Warszawskiej (aplikację opisano szczegółowo w załączniku 1). Takie podejście pozwoliło na osiągnięcie wysokiego poziomu symulacji oraz zapewnienie uczestnikom doświadczenia maksymalnie zbliżonego do rzeczywistych warunków korzystania z systemów nawigacji w przestrzeni zamkniętej. **Integracja ta wymagała jednak rozwiązania szeregu wyzwań technicznych związanych z komunikacją pomiędzy silnikiem gry, a zewnętrzną aplikacją nawigacyjną, które opisano szczegółowo w rozdziale 6.9.**

Z uwagi na to, że wykorzystana aplikacja została wykonana w technologii Web (aplikacja jest uruchamiana w przeglądarce internetowej) uczestnikom przedstawiono możliwość wyboru trybu jej wyświetlania.

Dostępne tryby (Rysunek 6.27 i Rysunek 6.28):

- a) Mapa na dodatkowym telefonie – ten tryb wymaga zeskanowania kodu QR telefonem, który następnie należy umieścić obok ekranu używanego do wyświetlania symulacji. Aplikacja mapowa uruchamiana jest w przeglądarce internetowej dostępnej na smartfonie, niezależnie od systemu operacyjnego.
- b) Mapa zintegrowana z symulacją – okno mapy nakładane jest na interfejs użytkownika. Aplikacja mapowa uruchamiana jest w przeglądarce internetowej zintegrowanej z silnikiem Unreal Engine, dołączonej do niego za sprawą pluginu WebView (aSurginRiver 2024)¹⁶. Aplikacja uruchamiana jest w oddzielnym, izolowanym procesie.

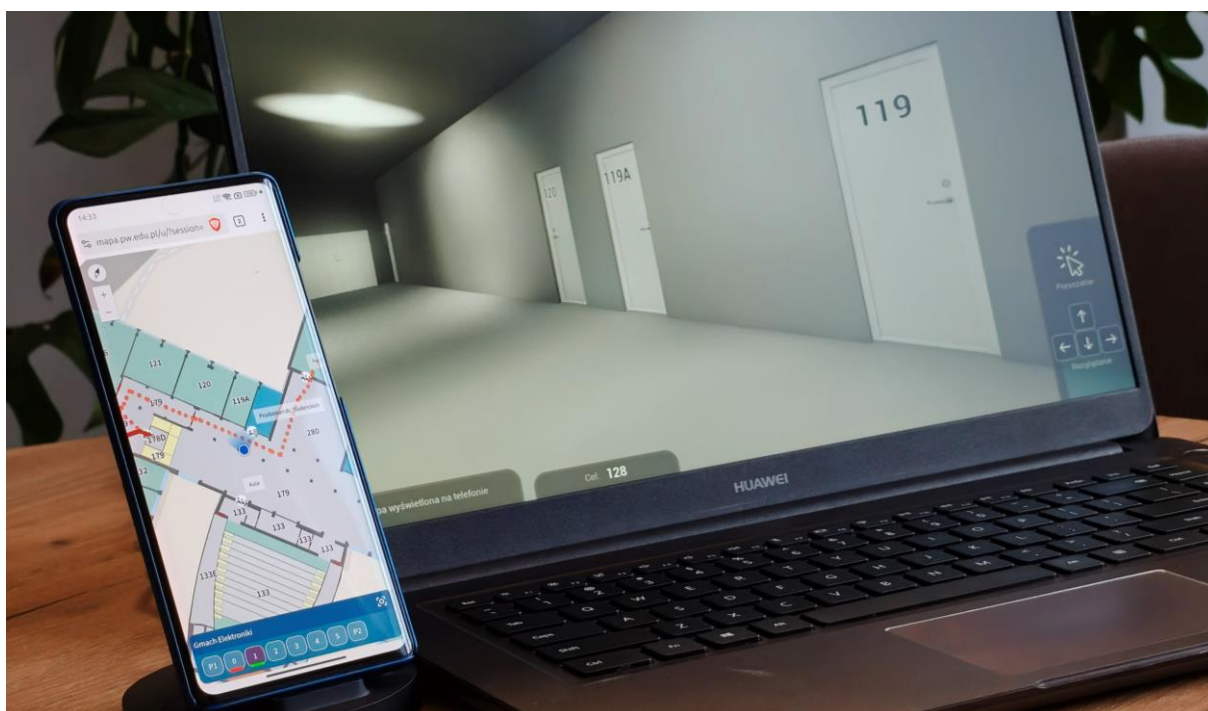
¹⁶ Twórca pluginu udostępnił darmową licencję, na potrzeby przeprowadzenia badań w niniejszej rozprawie.



Rysunek 6.27. Wybór trybu wyświetlania mapy, przedstawiany uczestnikom na początku badania (opracowanie własne)

Okolo 66% uczestników wybrało zintegrowany tryb mapy.

Szczegółowa analiza wyborów uczestników została przeprowadzona w rozdziale 7.4.



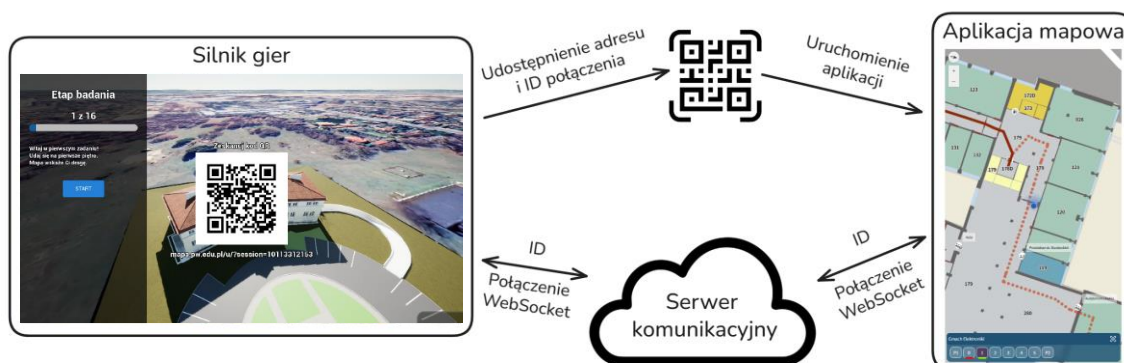
Rysunek 6.28. a) Aplikacja nawigacyjna działa na smartfonie i jest połączona z silnikiem gier
b) Zrzut ekranu uruchomionej symulacji ze zintegrowaną aplikacją nawigacyjną (opracowanie własne)

Przyjęte rozwiązanie polegające na wykorzystaniu istniejącej aplikacji nawigacyjnej nie tylko spełniło założenia badawcze, ale także otworzyło nowe możliwości w zakresie metodyki badań nad systemami nawigacji wewnątrz budynków.

6.9. Komunikacja pomiędzy aplikacją nawigacyjną a symulatorem

6.9.1. Opis wybranej technologii komunikacji

Komunikacja pomiędzy silnikiem gier, a aplikacją nawigacyjną odbywa się z pośrednictwem stworzonego przez autora specjalnego serwera komunikacyjnego (Rysunek 6.29). Zdecydowano się na komunikację z wykorzystaniem protokołu WebSocket, który jest znacząco szybszy od klasycznej komunikacji HTTP, co odpowiada potrzebie przesyłania lokalizacji w czasie rzeczywistym (Pimentel i Nickerson 2012). Po stronie Unreal Engine komunikację z wykorzystaniem tego protokołu uruchomiono w oparciu o plugin o nazwie *WebSocket with Blueprint* (547 Game Studio 2023)¹⁷. Serwer został uruchomiony na Platformie IT CENAGIS¹⁸ w skonteneryzowanym środowisku.



Rysunek 6.29. Schemat uruchamiania aplikacji nawigacyjnej i komunikacji z silnikiem gier podczas trwania badania (opracowanie własne)

Kolejność inicjowania połączenia jest następująca:

- 1) Uruchomienie symulacji
- 2) Wygenerowanie losowego ID, identyfikującego instancję programu i uczestnika

¹⁷ Sposób korzystania z pluginu nie jest udokumentowane przez twórców, dlatego jako dodatkowy artefakt pracy nad rozprawą powstała instrukcja korzystania z tego dodatku pt.: „Unofficial manual - WebSocket with Blueprint – Unreal Engine Plugin”. Instrukcja została udostępniona społeczności i podlinkowana przez twórców pluginu. Jest dostępna pod adresem:

https://docs.google.com/document/d/1qOyEm_PAMDXJ0D5lv9VAOXmVbwCZgYqiJyghrC7Uwzk/

¹⁸ CENAGIS – Centrum Analiz Geoprzestrzennych i Satelitarnych Wydziału Geodezji i Kartografii Politechniki Warszawskiej, dysponujące zasobami obliczeniowymi wykorzystanymi do przeprowadzenia badań, którego kluczową infrastrukturą jest chmurowa Platforma IT CENAGIS (CENAGIS 2023)

- 3) Nawiązanie połączenia z serwerem komunikacyjnym przez silnik symulacji z wykorzystaniem ID (gotowość na połączenie aplikacji nawigacyjnej)
- 4) Uruchomienie aplikacji nawigacyjnej
 - a. Na telefonie poprzez zeskanowanie kodu QR z zaszytym adresem i ID
 - b. W trybie zintegrowanym na podstawie adresu i ID
- 5) Nawiązanie połączenia z serwerem komunikacyjnym przez aplikację nawigacyjną

Dzięki zastosowaniu unikalnego ID identyfikującego komunikację możliwe jest równoczesne uczestnictwo wielu osób. Ich sesje połączenia pozostają niezależne i nie wpływają na siebie nawzajem.

6.9.2. Implementacja serwera komunikacyjnego

Serwer komunikacyjny oprogramowano z wykorzystaniem języka Python zapewniając funkcjonalność tworzenia niezależnych sesji dla każdego z uczestników badania na podstawie przypisanego ID. Każdy nowy klient połączenia dołączając z zadany numerem ID jest przypisywany do grupy rozgłoszeniowej (ang. *broadcast*). Co do zasady do jednej grupy dołączają dwa podmioty, tj. instancja symulatora *IndoorLab* oraz aplikacja mapowa. Nie ma natomiast technicznych przeszkód, aby np. na potrzeby prezentacji lub porównania przesyłaną lokalizację wyświetlać w więcej niż jednej aplikacji. Wiadomości przesyłane przez jednego uczestnika grupy są przesyłane do jej wszystkich pozostałych członków.

Dzięki wykorzystaniu bibliotek programistycznych *asyncio*¹⁹, *ssl*²⁰ oraz *websockets*²¹ implementacja logiki serwera komunikacyjnego jest zwięzła i wydajna (Rysunek 6.30).

¹⁹ *asyncio* – biblioteka umożliwiająca wykonywanie operacji asynchronicznych (Python Software Foundation 2014)

²⁰ *ssl* – biblioteka umożliwiająca wykorzystanie szyfrowania komunikacji przy pomocy certyfikatów (Python Software Foundation 2008)

²¹ *websockets* – biblioteka umożliwiająca tworzenie serwera dla połączeń protokołu websocket (Aymeric 2013)

```

main.py > ...
1  import asyncio
2  import websockets
3  import ssl
4  from collections import defaultdict
5
6  # Websocket server with below functionalities:
7  # Join clients in groups by connection url
8  # Clients who joined group A gets broadcasted messages from group A
9  # Clients who joined group B gets broadcasted messages from group B
10 # Use cenagis.crt and cenagis.key to run the server
11
12 groups = defaultdict(set)
13
14 async def register(websocket, path):
15     group = path.strip('/')
16     group = group.split('/')[1]
17     print(f"Client joined group {group}")
18     groups[group].add(websocket)
19     try:
20         async for message in websocket:
21             await broadcast(group, message, websocket)
22     finally:
23         groups[group].remove(websocket)
24
25 async def broadcast(group, message, sender):
26     websockets_to_remove = set()
27     for websocket in groups[group]:
28         if websocket != sender:
29             try:
30                 await websocket.send(message)
31             except websockets.ConnectionClosed:
32                 websockets_to_remove.add(websocket)
33     for websocket in websockets_to_remove:
34         groups[group].remove(websocket)
35
36 ssl_context = ssl.SSLContext(ssl.PROTOCOL_TLS_SERVER)
37 ssl_context.load_cert_chain(certfile="cenagis.crt", keyfile="cenagis.key")
38
39 async def main():
40     start_server = await websockets.serve(register, "", 3478, ssl=ssl_context)
41     print("Server started at wss://*:3478")
42     await start_server.wait_closed()
43
44 asyncio.run(main(), debug=True)

```

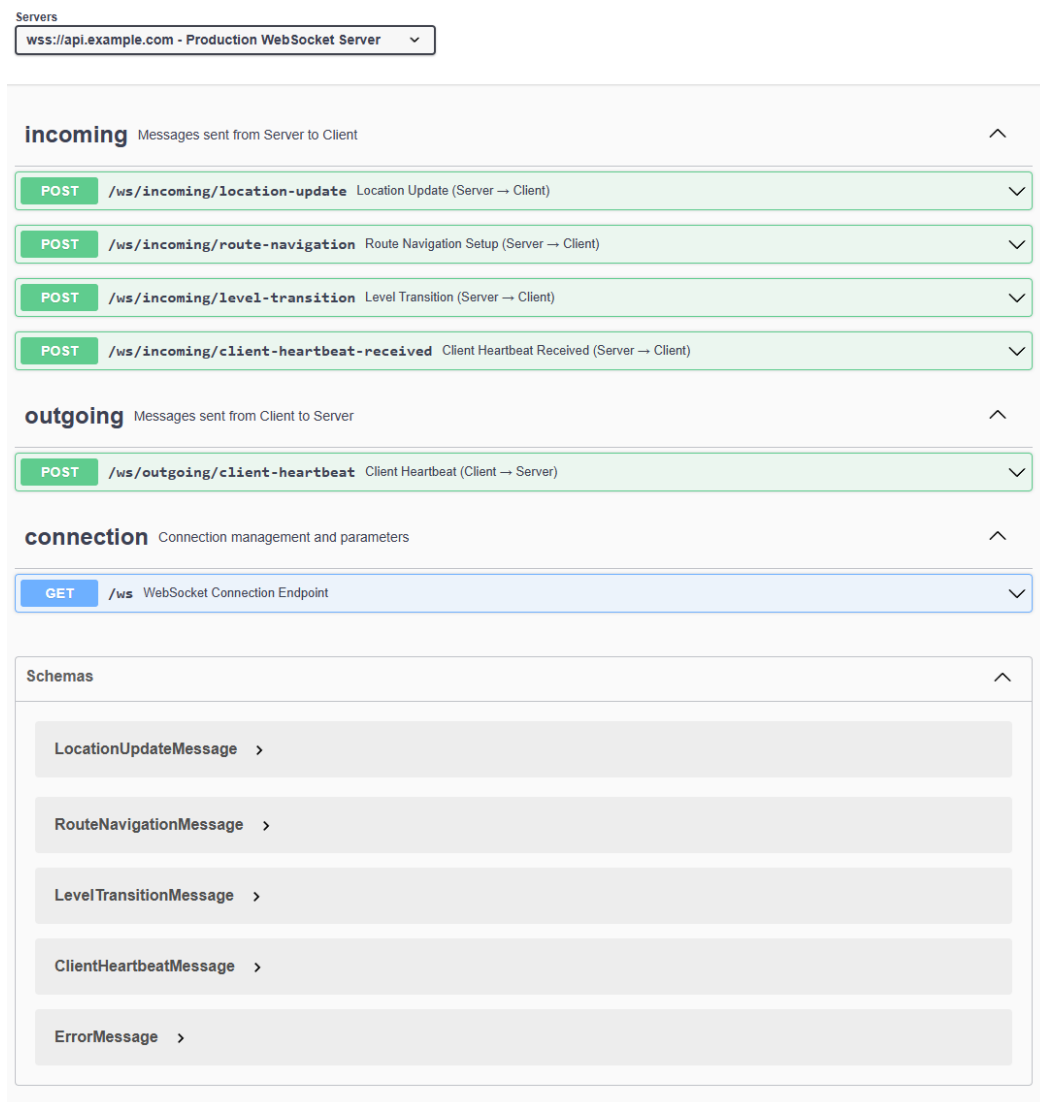
Rysunek 6.30. Główna część kodu serwera komunikacyjnego (opracowanie własne)

6.9.3. Dokumentacja interfejsu API

Komunikacja przesyłana przez serwer pośredniczący została udokumentowana jako uniwersalny interfejs dostępowy API, co pozwala w przyszłości testować również inne aplikacje, które skorzystają z zaproponowanego modelu danych.

Dokumentację można udostępnić jako stronę internetową typu Swagger²² (Rysunek 6.31). Została również dołączona do rozprawy jako załącznik 5.

²² Często wykorzystywany w branży IT interfejs wyświetlania dokumentacji API (Swagger 2011)

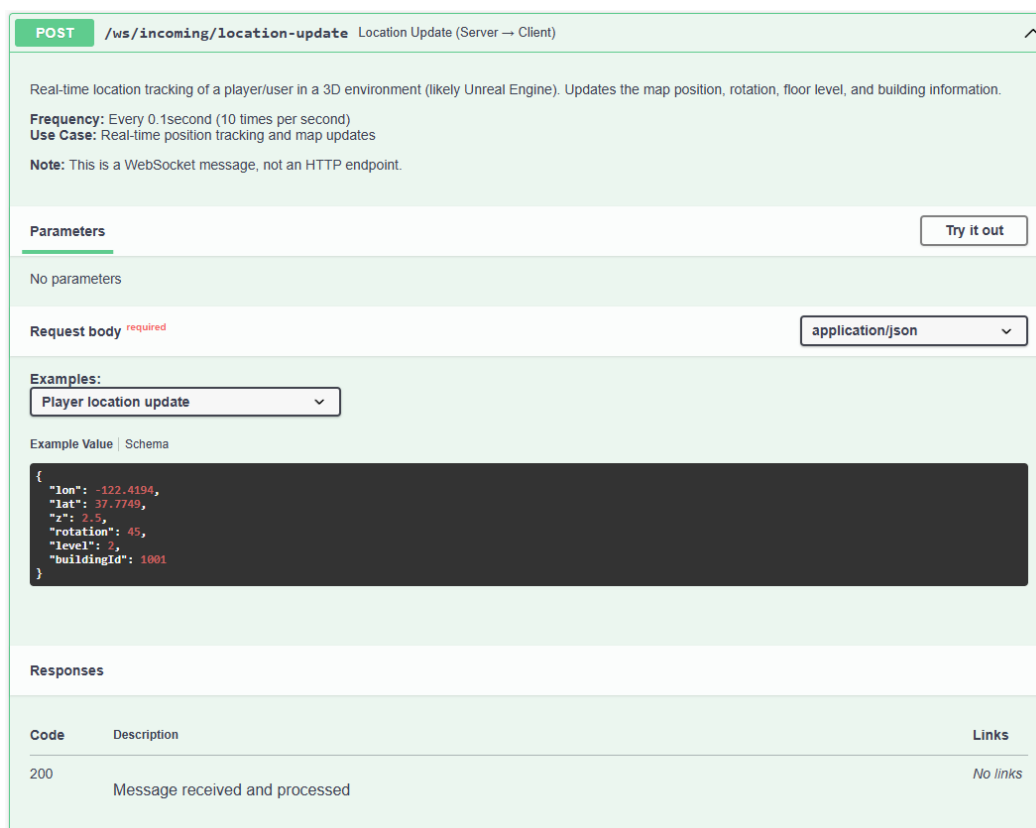


Rysunek 6.31. Podgląd strony dokumentacji API (opracowanie własne)

Oprócz połączenia do serwera WebSocket zdefiniowane są typy obsługiwanych wiadomości, w zakresie:

- 1) Wiadomości przychodzących (wysyłanych przez silnik symulacji, odbieranych przez aplikację mapową):
 - a. Aktualizacja lokalizacji (Rysunek 6.32)
 - b. Wytyczenie nowej trasy
 - c. Tryb przejścia/przerwy pomiędzy scenariuszami badawczymi
 - d. Wiadomość typu heartbeat przesyłana przez potencjalnych innych klientów (powinna być ignorowana przez aplikację mapową)

- 2) Wiadomości wychodzących (wysyłanych przez aplikację mapową, odbieranych przez silnik symulacji):
 - a. Heartbeat klienta – potwierdza utrzymywanie połączenia przez aplikację mapową. Dzięki tej wiadomości silnik symulacji wie, że aplikacja poprawnie wyświetla mapę użytkownikowi, gdy taka wiadomość nie pojawi się przez określony czas silnik wykona sekwencję ponownego nawiązywania połączenia lub poprosi użytkownika o ponowne zeskanowanie kodu QR



Rysunek 6.32. Dokumentacja wiadomości dotyczącej aktualizacji lokalizacji (opracowanie własne)

6.10. Integracja aplikacji „Mapa PW” z symulatorem

Na potrzeby realizowanego badania wykorzystano istniejącą aplikację *Mapa PW* przeznaczoną do nawigacji w budynkach. Autor rozprawy był jednym z jej głównych twórców, dzięki czemu miał możliwość rozbudowania aplikacji na potrzeby niniejszej rozprawy. Opis architektury oraz funkcjonalności wspomnianej aplikacji jest dostępny w załączniku nr 1.

W poniższym rozdziale opisano najistotniejsze elementy, które zostały zaimplementowane specjalnie na potrzeby integracji aplikacji *Mapa PW* z tworzoną platformą badawczą.

Uruchomienie aplikacji *Mapa PW* w trybie przeprowadzanego badania następuje po wypełnieniu ankiety wstępnej, automatycznie w trybie zintegrowanym lub po zeskanowaniu kodu QR (Rysunek 6.33) w trybie wykorzystania dodatkowego telefonu (tryby korzystania z aplikacji opisano w rozdziale 6.8.2). W obu wariantach aplikacja ma dostęp do dołączonego parametru `sessionId`, który służy identyfikacji konkretnej instancji symulatora oraz aplikacji nawigacyjnej (co ma znaczenie, gdy wiele osób wykonuje badanie równocześnie i niezależnie).



Rysunek 6.33. Wycinek ekranu prezentujący QR kod pozwalający na nawiązanie połączenia z symulatorem (opracowanie własne)

Nawiązywanie połączenia

Połączenie z symulatorem odbywa się poprzez serwer pośredniczący (rozdział 6.9). W specjalnym module utworzono zestaw funkcji i tzw. hooków²³ przeznaczonych do obsługi połączenia.

²³ **React hook** to funkcja, która pozwala używać stanu i innych funkcjonalności Reacta (takich jak metody cyklu życia) w komponentach funkcyjnych, bez potrzeby tworzenia komponentów klasowych. Komponenty funkcyjne są istotnie wydajniejsze i korzystanie z nich jest uznawane za dobrą praktykę.

Proces rozpoczyna się inicjalizacją połączenia typu WebSocket z endpointem, czyli adresem, na którym nasłuchuje serwer (Rysunek 6.34). Adres połączenia zawiera dostarczone w parametrze ID sesji, identyczne jak to wykorzystywane przez silnik konkretnej instancji symulatora.

```
export const initializeWebSocket = (sessionId: string) => {
  const socket = new WebSocket(`wss://turn.cenagis.edu.pl:3478/${sessionId}`)

  socket.onopen = () => {
    console.log('Socket Opened')
  }
  socket.onclose = () => {
    console.log('Socket Closed, reloading...')
    location.reload()
  }
  socket.onerror = (error) => {
    console.error('Socket Error: ', error)
    socket.close()
  }

  return socket
}

useEffect(() => {
  if (!sessionId) return

  socketRef.current = initializeWebSocket(sessionId)

  // Cleanup function
  return () => {
    socketRef.current?.close()
  }
}, [sessionId])
```

Rysunek 6.34. Kod w języku JavaScript inicjujący połączenie WebSocket (opracowanie własne)

Kontrola połączenia

Kolejno ustanawiane są funkcje typu `heartbeat`, które w regularnych interwałach kontrolują status połączenia. Aplikacja mapowa odbiera `streamerHeartbeat`, który potwierdza gotowość symulatora, a wysyła `clientHeartbeat`, który informuje symulator o trwającym, aktywnym połączeniu (Rysunek 6.35).

Hooki zostały wprowadzone w React 16.8 i ich nazwy zawsze zaczynają się od słowa "use" (np. useState, useEffect, useContext) (React 2019).

```

// Set up message handler
const setupMessageHandler = useCallback(
  (onMessage: MessageHandler) => {
    if (!socketRef.current) return

    socketRef.current.onmessage = (event) => {
      // Handle heartbeat
      if (event.data === 'streamerHeartbeat') {
        setTimeout(setConnectionAlive, true, 5000)
        return
      }

      // Parse data
      try {
        const data = JSON.parse(event.data)
        onMessage(data)
      } catch (error) {
        console.error('Failed to parse WebSocket message:', error)
      }
    }

    socketRef.current.onerror = (error) => {
      console.error('WebSocket error:', error)
    }
  },
  [setTimeout],
)

// Send message to WebSocket - memoized to prevent unnecessary rerenders
const sendMessage = useCallback((message: any) => {
  if (socketRef.current?.readyState === WebSocket.OPEN) {
    socketRef.current.send(JSON.stringify(message))
  } else {
    console.warn('WebSocket is not open, message not sent')
  }
}, [])

// heartbeat and connection status
useEffect(() => {
  // Handle connection status changes
  if (!connectionAlive) {
    setMapActive(false)
  }

  // Handle client heartbeat
  if (!sessionId) return

  const intervalId = setInterval(() => {
    const stateToSend = { mapVisible: mapActive }
    sendMessage({ type: 'clientHeartbeat', data: stateToSend })
  }, 2000)

  return () => {
    clearInterval(intervalId) // Cleanup interval on unmount
  }
}, [mapActive, sessionId, sendMessage, connectionAlive])

```

Rysunek 6.35. Fragment kodu odpowiedzialnego za działanie funkcji typu heartbeat (opracowanie własne)

Obsługa wiadomości przychodzących

Oprócz kontroli stanu połączenia powyższa funkcja `setupMessageHandler` jest wykorzystywana do obsługi różnorodnych wiadomości przychodzących, które będą wysyłane przez symulator przez cały czas trwania badania. Obsługa tych wiadomości stanowi implementację interfejsu API (rozdział 6.9.3) udostępnianego przez symulator. Za podejmowanie akcji na podstawie odebranych danych odpowiada funkcja `messageHandlerCallback` (Rysunek 6.36). Przysyłane dane to przede wszystkim aktualna i dokładna lokalizacja uczestnika badania w świecie wirtualnym, a także polecenie wyznaczenia nowej trasy na początku każdego ze scenariuszy badawczych. Ta funkcja przetwarza także wszystkie dodatkowe parametry dotyczące modyfikacji zasad wyświetlania mapy, zależnych od realizowanego scenariusza.

```
const messageHandlerCallback = useCallback(
  (data: any) => {
    if (typeof data === 'string') {
      console.error('Unexpected string data in message handler:', data)
      return
    }

    // Check for location data
    if ('lon' in data) {
      const locationData = data
      if (mapAnimationStateRef.current !== MapAnimationStates.routeAnimation) {
        handleLocationChange(
          jimuMapView,
          locationData,
          mapVisibilityMode,
          setCoords,
          setRotation,
          setLevel,
          setBuildingId,
          setShowLocationDot,
          locationUpdateTimeoutRef, // Pass the timeout reference
        )
      }
    }
    // Check for route data
    else if ('startLocation' in data) {
      setMapActive(true)
      changeFloor(data.buildingId, data.startLevel)
      setBuildingId(0)
      setRotation(0)
      setShowLocationDot(false)
      ...
    }
  }
)
```

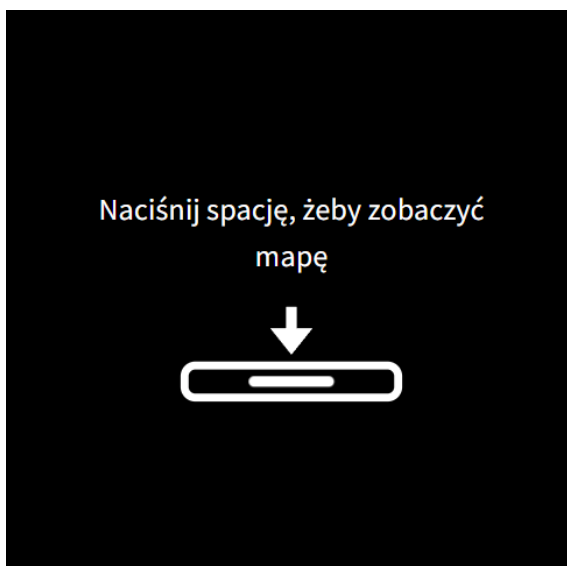
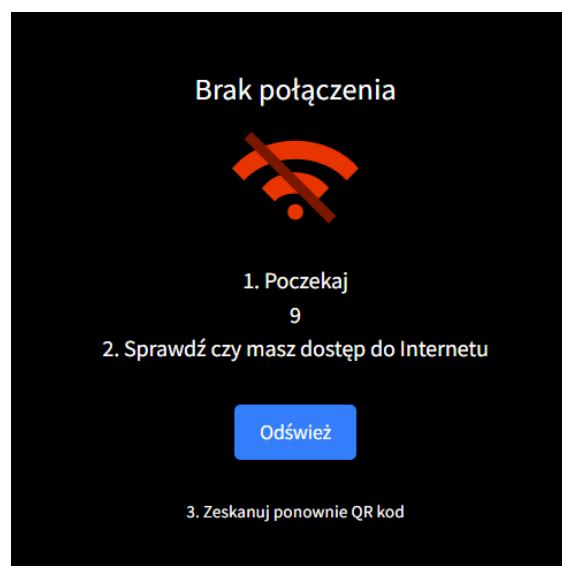
Rysunek 6.36. Fragment funkcji `messageHandlerCallback` (opracowanie własne)

Nakładka zakrywająca mapę

W ramach implementacji narzędzi wspomagających przeprowadzenie badania zaprojektowano mechanizm nakładki (ang. *overlay*), który zarządza widocznością mapy wewnątrz budynków w zależności od etapu badania i stanu aplikacji. Nakładka ta wyświetlana jest w czterech różnych scenariuszach:

- pomiędzy kolejnymi zadaniami badawczymi,
- w sytuacji problemów z połączeniem sieciowym,
- w trybie wyświetlania mapy na żądanie po naciśnięciu klawisza spacji
- gdy w danym scenariuszu mapa nie jest dostępna.

Mechanizm ten poprawia intuicyjność systemu badawczego poprzez komunikowanie aktualnego stanu i umożliwia programistyczne zarządzanie dostępem do interfejsu mapowego. Poniżej przedstawiono ilustrację wyglądu różnych wariantów nakładki (Rysunek 6.37) oraz fragment kodu odpowiedzialny za logikę wyboru odpowiedniego typu komunikatu w zależności od bieżącego kontekstu aplikacji (Rysunek 6.38).



Rysunek 6.37. Cztery rodzaje nakładki zakrywającej mapę (opracowanie własne)

```

// Setup overlay content based on conditions
const getOverlayContent = () => {
  if (!connectionAlive) {
    return (
      <>
        <StyledSpan text={t.noConnection} fontSize="24px" />
        <StatusIcon isActive={false} />
        <StyledSpan text={t.waitAndCheck(countdown)} fontSize="18px" />
        <button
          css={css`
            margin: 20px;
            padding: 10px 20px;
            font-size: 16px;
            color: white;
            background-color: #007bff;
            border: none;
            border-radius: 5px;
            cursor: pointer;
          `}
          onClick={() => {
            window.location.reload()
          }}
        >
          {t.refresh}
        </button>
        <StyledSpan text={t.scanAgain} fontSize="14px" />
      </>
    )
  }
  if (!mapVisible) {
    return (
      <>
        <StyledSpan text={t.connectionEstablished} fontSize="24px" />
        <StatusIcon isActive={true} />
      </>
    )
  }
  if (mapVisibilityMode === MapVisibilityModes.notVisible) {
    return (
      <>
        <StyledSpan text={t.mapNotAvailable} fontSize="24px" />
        <MapCrossoutIcon />
      </>
    )
  }
  if (showLocationDot && mapVisibilityMode === MapVisibilityModes.onRequest) {
    return (
      <>
        <StyledSpan text={t.pressSpaceToSeeMap} fontSize="24px" />
        <SpacebarIcon />
      </>
    )
  }
  return null
}

```

Rysunek 6.38. Fragment kodu wyświetlający odpowiedni wariant nakładki zakrywającej mapę (opracowanie własne)

6.11. Mechanizm rejestrowania i zbierania danych badawczych

Wykorzystany mechanizm zbierania danych został już wcześniej przetestowany i opisany w ramach pracy magisterskiej autora (Łobodecki 2020). Zakłada on przechowywanie bieżących informacji oraz statystyk dotyczących przebiegu badania w stanie aplikacji symulatora, a następnie przesyłanie ich na zewnętrzny serwer po ukończeniu każdego scenariusza badawczego.

Mechanizm składał się z kilku komponentów:

1. Baza danych

Rolę bazy danych na potrzeby badania przyjął rozbudowany arkusz kalkulacyjny Google Sheet. To rozwiązanie wykazało wystarczającą wydajność oraz dużą wygodę w analizie i przetwarzaniu danych. Arkusz zawiera łącznie 55 kolumn zawierających zarówno surowe jak i przetworzone dane (Rysunek 6.39).

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	
1	id	timestamp	sessionId	PixelSt	PixelSt	lan	lan	email	Con	Wiek	Co	Pieć	Com	Grani	Corr	Uży	Corr	Orie	kno	Architek	Chemia	Elektron	Elektro	Gmach	Gmo
413	413	13/06/2025 16:07:08	619795889	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	3	Cod	2	Dob	fals	0	0	0	0	1	Firs
414	414	13/06/2025 16:07:44	619795889	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	3	Cod	2	Dob	fals	0	0	0	0	1	Firs
415	415	13/06/2025 16:08:39	619795889	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	3	Cod	2	Dob	fals	0	0	0	0	1	Firs
416	416	13/06/2025 16:10:12	619795889	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	3	Cod	2	Dob	fals	0	0	0	0	1	Firs
417	417	13/06/2025 16:10:47	619795889	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	3	Cod	2	Dob	fals	0	0	0	0	1	Firs
418	418	13/06/2025 16:11:24	619795889	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	3	Cod	2	Dob	fals	0	0	0	0	1	Firs
419	419	13/06/2025 16:13:25	619795889	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	3	Cod	2	Dob	fals	0	0	0	0	1	Firs
420	420	13/06/2025 16:47:44	1135057832	TRUE	1				1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
421	421	13/06/2025 16:48:18	1135057832	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
422	422	13/06/2025 16:49:36	1135057832	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
423	423	13/06/2025 16:50:09	1135057832	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
424	424	13/06/2025 16:51:09	1135057832	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
425	425	13/06/2025 16:52:01	1135057832	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
426	426	13/06/2025 16:53:30	1135057832	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
427	427	13/06/2025 16:58:28	1151059421	TRUE	1				1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
428	428	13/06/2025 16:58:56	1151059421	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
429	429	13/06/2025 16:59:40	1151059421	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
430	430	13/06/2025 17:00:09	1151059421	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
431	431	13/06/2025 17:01:24	1151059421	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
432	432	13/06/2025 17:02:05	1151059421	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
433	433	13/06/2025 17:03:01	1151059421	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
434	434	13/06/2025 17:04:19	1151059421	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
435	435	13/06/2025 17:05:11	1151059421	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
436	436	13/06/2025 17:06:13	1151059421	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
437	437	13/06/2025 17:07:12	1151059421	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
438	438	13/06/2025 17:08:10	1151059421	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
439	439	13/06/2025 17:08:44	1151059421	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
440	440	13/06/2025 17:09:28	1151059421	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
441	441	13/06/2025 17:10:26	1151059421	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
442	442	13/06/2025 17:11:11	1151059421	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
443	443	13/06/2025 17:11:44	1151059421	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
444	444	13/06/2025 17:14:34	1151059421	TRUE	1	pl	1		1	17-24	1	Meż	2	Tak	2	Czę	1	Śrec	true	1	0	0	0	1	Firs
445	445	13/06/2025 17:42:29	13524727456	TRUE	1				2	25-34	1	Meż	1	Troch	1	Rzac	1	Śrec	true	1	0	1	0	1	Moi
446	446	13/06/2025 17:43:16	13524727456	TRUE	1	pl	1		2	25-34	1	Meż	1	Troch	1	Rzac	1	Śrec	true	1	0	1	0	1	Moi
447	447	13/06/2025 17:44:32	13524727456	TRUE	1	pl	1		2	25-34	1	Meż	1	Troch	1	Rzac	1	Śrec	true	1	0	1	0	1	Moi

Rysunek 6.39. Podgląd arkusza kalkulacyjnego pełniącego rolę bazy danych zbieranych podczas badania (opracowanie własne)

2. Usługa wystawiająca API gotowe do odbierania paczek z informacjami do zapisania

Do realizacji komponentu wykorzystano Google Apps Script²⁴. Rozwiązanie to umożliwia uzyskanie unikalnego linku będącego adresem API, na który można przestać dane w formacie JSON z wykorzystaniem zapytania HTTP typu POST (funkcja *doPost* – Rysunek 6.40). Dane te następnie trafiają do specjalnej funkcji, która je przetwarza i zapisuje do odpowiednich komórek arkusza (funkcja *appendRowToDataSheet* – Rysunek 6.40).

```
1 function appendRowToDataSheet(dataObject) {
2   const spreadsheet = SpreadsheetApp.getActive();
3   const sheet = spreadsheet.getSheetByName("Final");
4
5   const date = new Date();
6   dataObject['timestamp'] = Utilities.formatDate(date, Session.getScriptTimeZone(), "dd/MM/yyyy HH:mm:ss");
7
8   const headers = sheet.getRange(1, 1, 1, sheet.getLastColumn()).getValues()[0];
9   const row = headers.map(header =>
10    | dataObject[header] !== undefined && dataObject[header] !== null ? dataObject[header] : ''
11    | );
12   row[0] = sheet.getLastRow();
13
14   sheet.appendRow(row);
15 }
16
17 function doPost(e) {
18   appendRowToDataSheet(JSON.parse(e.postData.contents))
19
20   return HtmlService.createHtmlOutput('Data saved!');
21 }
```

Rysunek 6.40. Kod w usłudze Google Apps Script pozwalający na uruchomienie API, przetworzenie danych oraz ich zapis w arkuszu kalkulacyjnym (opracowanie własne)

3. Plugin VaREST dla silnika Unreal

Plugin VaREST dla silnika Unreal umożliwia wykonywanie zapytań HTTP/HTTPS z wykorzystaniem różnych metod (GET, POST, PUT, DELETE) i typów zawartości. Pozwala na wywoływanie zewnętrznych API i przetwarzanie odpowiedzi w formacie JSON bezpośrednio w systemie blueprintów, bez konieczności programowania w C++. VaRest oferuje pełną obsługę JSON z możliwością odczytywania, tworzenia i zapisywania obiektów JSON (GitHub 2015).

²⁴ Usługa umożliwiająca programowanie makr i funkcji współpracujących z programami udostępnianymi w ramach pakietu GSuite, takimi jak Google Docs, Google Drive czy Gmail (Google 2009).

4. Własny komponent AC_VaREST

Własny komponent AC_VaREST napisany przez autora pracy pozwala na odczytanie zgromadzonych statystyk oraz przesyłanie zebranych danych do API. Agreguje dane z różnych źródeł w projekcie (takie jak czas sesji, pozycje użytkownika czy dane demograficzne) i serializuje je do formatu JSON przed wysłaniem. Wykorzystuje plugin VaRest do przesłania zapytania typu POST, umożliwiając asynchroniczną komunikację z zewnętrznym serwerem bez blokowania głównego wątku gry. Komponent zarządza również obsługą błędów połączenia oraz ponawianiem nieudanych zapytań, co zwiększa niezawodność transmisji danych.

6.12. Badanie pilotażowe - wstępna ewaluacja narzędzia badawczego

Implementowane rozwiązanie, obejmujące integrację zaawansowanych technologii renderowania grafiki, mechanizmów komunikacji sieciowej, dynamicznego ładowania modeli architektonicznych oraz wielomodalnych interfejsów użytkownika charakteryzuje się wysokim poziomem skomplikowania. Stwarza to konieczność systematycznej weryfikacji działania wszystkich komponentów systemu w warunkach zbliżonych do rzeczywistego zastosowania badawczego poprzez przeprowadzenie iteracyjnych testów wewnętrznych. **W celu wstępnej ewaluacji opracowanego rozwiązania technologicznego zaplanowano także i zrealizowano badanie pilotażowe na małej grupie uczestników.** Badanie pilotażowe stanowi nieodzowny element procesu rozwoju narzędzi badawczych, szczególnie w kontekście złożonych systemów technologicznych wymagających precyzyjnej weryfikacji funkcjonalnej przed docelowym wdrożeniem. **W przypadku opracowanego symulatora nawigacji wewnątrz budynków, przeprowadzenie wstępnej ewaluacji okazało się kluczowe dla identyfikacji potencjalnych ograniczeń technicznych oraz metodologicznych, które mogłyby negatywnie wpłynąć na jakość i wiarygodność danych pozyskanych w ramach głównego badania.**

Zgodnie z metodologią badań pilotażowych, proces ten oprócz przetestowania aspektów technicznych, koncentrował się na ocenie adekwatności zastosowanych metod badawczych, identyfikacji potencjalnych problemów proceduralnych oraz weryfikacji skuteczności narzędzi pomiarowych w kontekście specyfiki planowanych badań.

Szczególne znaczenie w kontekście projektu *IndoorLab* miała weryfikacja użyteczności (ang. *usability*) opracowanego środowiska symulacyjnego, uwzględniając różnorodność kompetencji cyfrowych potencjalnych uczestników. Proces ewaluacji obejmował elementy związane z ergonomią interfejsu użytkownika oraz intuicyjnością mechanizmów sterowania.

Przeprowadzenie badania pilotażowego pozwoliło również na ocenę adekwatności czasu przewidzianego na udział w eksperymencie. Dodatkowo, proces ten umożliwił wstępną analizę danych zbieranych przez system badawczy oraz optymalizację parametrów rejestracji zachowań uczestników, co miało bezpośredni wpływ na jakość analiz statystycznych przeprowadzanych w ramach badania głównego.

W badaniu pilotażowym wzięło udział 10 uczestników. Każdy został poproszony o przetestowanie różnych wariantów dostępu do środowiska oraz trybów badania. Trasy wyświetlane w wirtualnym budynku badania pilotażowego nie pokrywały się w żadnym stopniu z trasami zaplanowanymi dla badania docelowego. W badaniu pilotażowym nie były dostępne następujące funkcjonalności:

- Poziom treningowy
- Windy (wymagano korzystania ze schodów)

Po udziale w badaniu uczestnicy poproszeni zostali o przesłanie swoich szczegółowych uwag, które zostały później przeanalizowane i zaadresowane w planie dalszego usprawniania narzędzia badawczego. Wybór najistotniejszych pozyskanych uwag od uczestników badania oraz akcje podjęte w ich wyniku przedstawiono w tabeli 6.

Tabela 6. Uwagi uczestników badania pilotażowego oraz zastosowane rozwiązania zgłaszanych problemów (opracowanie własne)

Uwaga uczestników badania pilotażowego	Zastosowane rozwiązanie
„Ze względu na zauważalne opóźnienie, standardowy tryb sterowania wydaje mi się mało komfortowy w przypadku strumieniowania. W tym wypadku dużo wygodniej korzysta się z trybu uproszczonego, ponieważ nie opiera się on na natychmiastowym feedbacku.” „W wersji webowej w mode gracz AWSO czułem zbyt duże opóźnienie między moim działaniem a reakcją aplikacji i prezentacją efektu na ekranie (spodziewam się, że zaraz się obróć o 15 stopni, a na ekranie obraz się “obraca” za sekundę) Powodowało to zaburzenie poczucia świadomości sytuacyjnej. Tryb sterowania “kliknij a pójdziesz w kliknięte miejsce” przy istniejącym opóźnieniu reakcji na kliknięcie nie dawało takiego wrażenia. Ten tryb wydaje się lepszym do użycia w trybie online.”	Wybór trybu sterowania został uzupełniony o dodatkową rekomendację poprzez dopisanie do Trybu Uprozczonego informacji „Zalecane przy streamingu” (Rysunek 6.13).
„Ekran startowy nie mieści się na ekranie. Przycisk start jest przycięty”	Naprawiono błędy wyświetlania strony dostępowej
„W wielu scenach było za ciemno”	Poprawiono konfigurację kamery pod względem czułości na światło oraz dodano wiele nowych źródeł oświetlenia we wnętrzach budynków.
„Gwiazdka zbiera się podwójnie i trochę za długi jest delay po zebraniu (wrażenie jakby coś nie działało, bo zebrałem, a nic się nie dzieje przez 3 sekundy)”	Uniemożliwiono kilkukrotne zebranie gwiazdki oraz skrócono czas oczekiwania na kolejny scenariusz.

„Źle dobrane parametry czułości myszy i FOV? - w mojej opinii za bardzo ekran się kręci i za szybko się biega. W grach raczej wszystko dzieje się spokojniej i zawsze dla osób, które nie grają dużo w gry to może być niełatwe.”	Domyślna czułość myszy została zredukowana. Dodano opcję zmiany czułości myszy w menu pauzy.
„Czarny ekran ładowania pomiędzy gmachami może być przyjemniejszy”	Do ekranu doładowania dodano szkice prezentujące aktualnie ładowany gmach. (szkice wygenerowano z wykorzystaniem AI)
„Rozważyłbym obowiązek kliknięcia w gwiazdkę, żeby ją zebrać. Gdy pędziłem po korytarzu (mając świadomość, że “dalej prosto”) to przypadkowo dotarłem do celu, a w celach badania pewnie ważną informacją jest to, że ktoś przeszedł docelowe drzwi.”	Zaimplementowano czas oczekiwania – aby zebrać gwiazdkę trzeba stanąć przy niej przez 1 sekundę.
„Trochę nie wiadomo kiedy po schodach iść do góry, a kiedy do dołu (dopiero po 3 poziomach zobaczyłem podświetlenie zielone czerwone z docelowym punktem. trochę mało widoczne ono jest)” „Nie wiedziałem czy powinienem iść do góry czy do dołu w zadaniach, gdzie konieczne było zmienienie piętra. Potem zauważyłem, że są wskaźniki przy numerach pięter w apce, ale bym jakoś mocno to zaznaczył bo na pierwszy rzut oka ich nie widać”	W poziomie treningowym wprost przekazano informację, o sposobie oznaczenia piętra docelowego w aplikacji nawigacyjnej.
„Czy przez drzwi powinno dać się przechodzić? Może nie? Przypadkowo wpadłem do windy”	W jednym gmachu testowo możliwe było przejście przez drzwi. Po badaniu

	pilotażowym całkowicie zrezygnowano z tej opcji.
„Przez kilka sekund nie jest aktualizowana lokalizacja na mapce”	Uwaga dotyczyła samego początku scenariusza, gdy prezentowana jest całość trasy, przed przejściem do aktualnej pozycji. Poprawki w zakresie animacji pozwoliły zniwelować wrażenie oczekiwania na lokalizację.
„Nie da się wyłączyć aplikacji po badaniu”	Dodano ekran końcowy z możliwością zamknięcia aplikacji badawczej.
„Zamieniona góra z dołem w trybie uproszczonym”	W trybie uproszczonym klawisze strzałek używane są do zmiany kierunku patrzenia. Zmieniono domyślne ustawienia rozglądania w osi Y (góra-dół), tak aby strzałka w górę powodowała spojrzenie w górę oraz dodano w menu pauzy możliwość dostosowania tego aspektu do własnych preferencji.
„Znacznik jakości połączenia zastania piętra”	W finalnej wersji badania całkowicie zrezygnowano z wyświetlania ikony jakości połączenia, która była przydatna podczas testów.
„Brakuje opcji restartu scenariusza”	Opcja rozpoczęcia poziomu od nowa mogłaby zaburzyć wyniki badania, z uwagi na znajomość części trasy, którą udało się przejść za pierwszym razem. Zamiast tego dodano możliwość ominięcia poziomu w menu pauzy.

„Brakuje informacji, że w trybie uproszczonym można się obracać używając klawiszy WASD”	Dodano stosowną informację w poziomie treningowym.
„Postęp badania zwłaszcza jak jest pusty wygląda jakby coś się nie załadowało”	Poprawiono interfejs ekranu wyświetlanego pomiędzy scenariuszami.
„Poruszanie wydaje się być trochę za szybkie względem chodu – to raczej bieg”	Zmniejszono prędkość poruszania.
„Wersja desktop wymagała instalacji dodatkowych bibliotek”	Dodano informację, o ewentualnej konieczności instalacji brakujących bibliotek.

Niektóre z uwag uczestników badania pilotażowego z różnych względów nie mogły być uwzględnione. W poniższej tabeli przedstawiono przykłady takich sytuacji.

Tabela 7. Uwagi uczestników badania pilotażowego, które nie mogły zostać poprawione wraz z komentarzem (opracowanie własne)

Uwaga uczestników badania pilotażowego	Komentarz
„Jak pójść w zupełnie złą stronę na inne piętro to nie mam żadnej informacji jak się odnaleźć. Nawet mi nie pokaże gdzie są schody.”	Z uwagi na przewidywane zaburzenie wyników badania nie zaimplementowano możliwości ponownego wyznaczania trasy. Jest to natomiast istotna uwaga, gdyż taka funkcja jest niezbędna w rzeczywistej aplikacji nawigacyjnej.
„W trybie uproszczonym drzwi z gwiazdką powinny przyciągać punkt dochodzenia”	Istotnie takie usprawnienie ułatwiłoby poruszanie, ale zrezygnowano z niego z uwagi na potencjalny wpływ na wynik badania – mechanika ta byłaby dodatkową wskazówką, ułatwiającą znalezienie odpowiednich drzwi.
„Na piętrach pośrednich nie ma wyznaczonej trasy”	Algorytm nawigacyjny zakładał wyznaczanie trasy z wykorzystaniem windy

	tam gdzie było to możliwe, a w takim wypadku trasa na piętrach pośrednich nie jest potrzebna. Dzięki wprowadzeniu wind w badaniu głównym ta uwaga przestała być aktualna.
„Z sieci firmowej oraz z mobilnej (hotspot z telefonu) utknąłem na WebRTC Negotiations”	Jest to znany problem z połączeniem z niektórych sieci. W interfejsie połączenia dodano informacje o rozwiązywaniu problemów i możliwości wystąpienia takiej ewentualności.
„Mam czasem poczucie, że pomieszczenia wydają się być mniejsze niż są w rzeczywistości”	Na poczucie przestrzeni w środowisku wirtualnym wpływ ma wiele czynników, takich jak pole widzenia, rozmiar ekranu, zastosowane tekstury, materiały i oświetlenie (Tural i Tural 2024). Przygotowując symulator starano jak najlepiej oddać odczucie właściwego rozmiar budynków, ale co do zasady nie jest to w pełni możliwe na płaskim ekranie.
„Ciekawie było by przechodzić podobne scenariusze w dzień i w nocy ze światłem sztucznym”	Ten pomysł był poza zakresem planowanych badań, ale istotnie jest to ciekawa możliwość ich rozszerzenia.

6.13. Profilowanie i optymalizacja środowiska wirtualnego

Podobnie jak w przypadku gier komputerowych problemy z wydajnością i płynnością działania środowiska wirtualnego mogą zaburzyć percepcję uczestników i zaciemnić wyniki samych badań. W tym podrozdziale nakreślono jakie zagadnienia optymalizacyjne były istotne podczas przygotowywania symulatora *IndoorLab* oraz jaką przyjęto strategię ich wdrożenia.

Cele i założenia wydajnościowe

Podstawowym wskaźnikiem wydajności działania aplikacji 3D jest zbieżność z założoną docelową liczbą klatek na sekundę (ang. *target framerate*), który zapewnia płynność animacji i komfort użytkownika. Uzyskiwana liczba klatek jest pochodną wielu czynników, takich jak konfiguracja ustawień graficznych, zastosowane techniki optymalizacji, czy wydajność używanego sprzętu.

Istotne jest, iż w zaprojektowanym i zrealizowanym systemie celem nie była maksymalizacja jakości wizualnej i wydajności poprzez agresywne optymalizacje, lecz utrzymanie zrównoważonej płynności, zwłaszcza z uwagi na różnorodne środowiska sprzętowe, w których symulator będzie uruchamiany. Pierwszy wariant zakłada uruchomienie lokalne na komputerze uczestnika badań, gdzie pełna moc obliczeniowa sprzętu jest dostępna bezpośrednio dla aplikacji. Drugi wariant obejmuje uruchomienie w środowisku chmurowym na serwerowych jednostkach graficznych NVIDIA T4 w trybie strumieniowania obrazu do klienta poprzez technologię Pixel Streaming. W tym przypadku większy nacisk kładzie się na stabilny FPS i minimalizację opóźnienia transmisji niż na maksymalną jakość graficzną, co wynika z konieczności efektywnego wykorzystania dostępnego pasma sieciowego.

Wartości powyżej 120 klatek na sekundę są przeznaczone przede wszystkim do zastosowań profesjonalnych/sportowych, w których ułamki sekund czasu reakcji decydują o wyniku rozgrywki pomiędzy graczami. Jako że opracowane środowisko badawcze *IndoorLab* nie jest przeznaczone do rywalizacji, dużo istotniejsze jest dobranie docelowego poziomu klatek na sekundę tak, aby zapewniał przede wszystkim komfort korzystania z symulatora. Co do zasady za w pełni komfortowe dla komputerów PC uznawane jest 60 klatek na sekundę, co też przyjęto

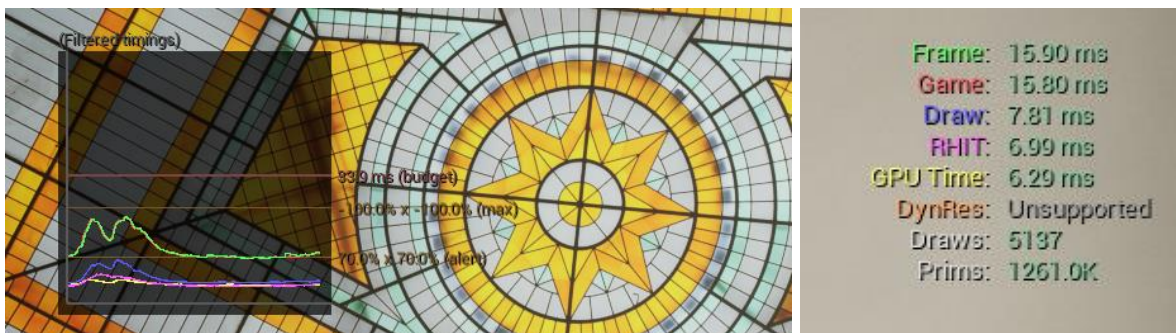
jako punkt odniesienia, w sytuacji uruchamiania symulatora na komputerze uczestnika. Z kolei 30 klatek na sekundę to powszechnie uznawane minimum, i taką też wartość przyjęto dla sytuacji streamingu z chmury, który oprócz czasu renderowania klatek jest również podatny na opóźnienia związane z internetowym przesyłaniem obrazu.

W praktyce pomiar klatek na sekundę (ang. *frames per second* – FPS) to innymi słowy pomiar czasu renderowania pojedynczej klatki. 120 FPS odpowiada 8,33 ms czasu na renderowanie, 60 FPS to 16,67 ms, a 30 FPS, to 33,33 ms. Oznacza to, że na wolniejszym sprzęcie uzyskamy mniejszą liczbę klatek na sekundę.

Profilowanie i narzędzia diagnostyczne

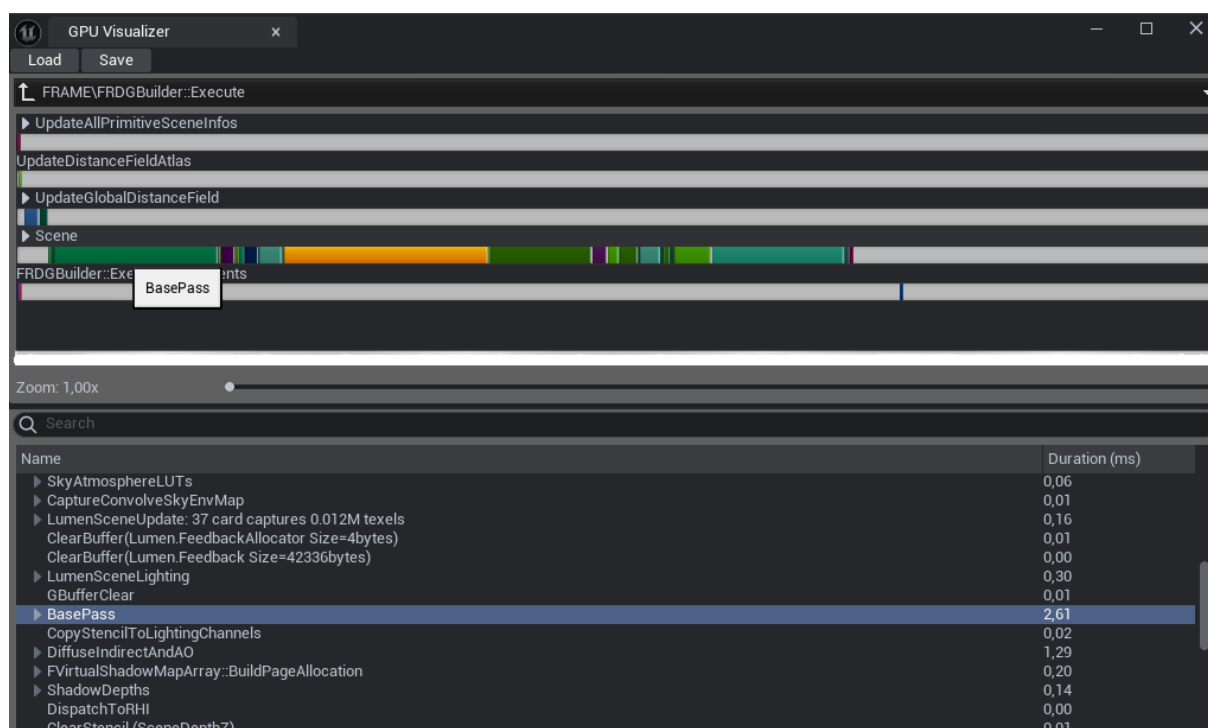
Profilowanie w branży tworzenia gier i symulacji, to forma dynamicznej analizy programu, która polega na badaniu jego zachowania, z wykorzystaniem danych zbieranych w czasie rzeczywistym. Dzięki profilowaniu programiści mogą precyzyjnie określić, które elementy kodu, lub które etapy renderowania, spowalniają program i wymagają poprawy, co przekłada się na płynniejsze działanie finalnego produktu.

Unreal Engine oferuje zaawansowane wbudowane narzędzia profilowania, które umożliwiają szczegółową analizę wydajności symulatora. Kluczowe znaczenie ma polecenie ``Stat Unit``, które wyświetla cztery podstawowe metryki: Frame (całkowity czas generowania klatki), Game (czas procesora CPU poświęcony na logikę gry), Draw (czas CPU na przygotowanie danych dla karty graficznej) oraz GPU (czas renderowania na karcie graficznej). Jeśli któraś z wartości zbliża się do wartości Frame, wskazuje to na „wąskie gardło” w danym komponencie systemu. Polecenie ``Stat UnitGraph`` rozszerza funkcjonalność stat unit o wizualizację w formie wykresu, co jest szczególnie użyteczne podczas przemieszczania się po scenie, ponieważ ułatwia lokalizację obszarów o największym obciążeniu obliczeniowym (Rysunek 6.41).



Rysunek 6.41. Efekt działania poleceń `Stat Unit` oraz `Stat UnitGraph` podczas wyświetlania wykorzystywanego w symulacji modelu Gmachu Głównego PW (opracowanie własne)

Narzędzie GPU Visualizer (Rysunek 6.42) stanowi graficzną reprezentację czasu renderowania, rozkładając go na poszczególne etapy przetwarzania sceny. Umożliwia to identyfikację najbardziej kosztownych operacji, takich jak renderowanie geometrii, obliczanie cieni czy efekty post-processingu (operacje przetwarzania obrazu po jego wyrenderowaniu).



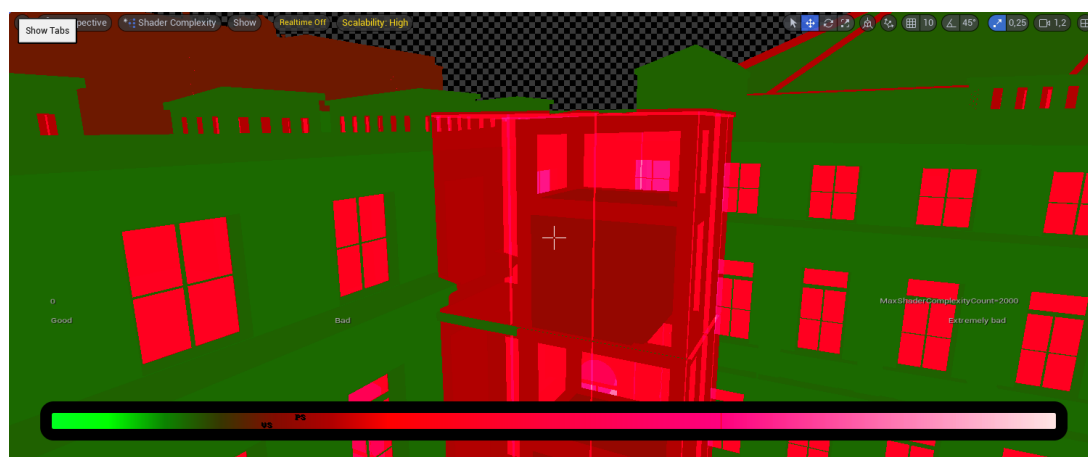
Rysunek 6.42. Okno narzędzia GPU Visualizer służącego do profilowania w Unreal Engine (opracowanie własne)

W kontekście analizy „wąskich gardeł” wynikających z dużej liczby geometrii w scenie, która charakteryzuje wykorzystane w opracowanym środowisku modele budynków, polecenie `Stat InitViews` oraz parametr `r.visualizeoccludedprimitives` pozwalają na identyfikację problemów związanych z przetwarzaniem widoczności i przestaniania obiektów.



Rysunek 6.43. Dynamiczne statystyki wyświetlane po wywołaniu polecenia `Stat InitViews` w silniku Unreal Engine (opracowanie własne)

Kolejnym narzędziem umożliwiającym identyfikację obciążających elementów sceny jest widok Shader Complexity (pol. złożoność cieniowania). W niektórych sytuacjach wykorzystanie zbyt skomplikowanego materiału lub efektów przezroczystości może znacząco pogarszać wydajność, gdy taki obiekt jest w polu widzenia. Na poniższym przykładzie (Rysunek 6.44) widać jak szyb szklanej windy oraz okna budynku są oznaczone na czerwono jako wymagające dla karty graficznej.



Rysunek 6.44. Widok Shader Complexity jako jedno z narzędzi profilowania silnika Unreal (opracowanie własne)

Wykorzystanie powyższych narzędzi pozwoliło na identyfikację przyczyn problemów z wydajnością. Z uwagi na poziom skomplikowania projektu *IndoorLab* istotne jest, aby proces profilowania przeprowadzać regularnie wraz z dodawaniem nowych funkcji środowiska wirtualnego. Pozwala to na dużo szybszą identyfikację potencjalnego źródła ograniczenia

wydajności. Istotne jest również regularne przeprowadzanie testów na platformach docelowych (lokalne uruchomienie na komputerze PC oraz renderowanie chmurowe). **Podczas pracy nad symulatorem wszelkie wykryte ograniczenia wydajności stanowiły podstawę do podejmowania decyzji optymalizacyjnych.**

Stosowane techniki optymalizacyjne

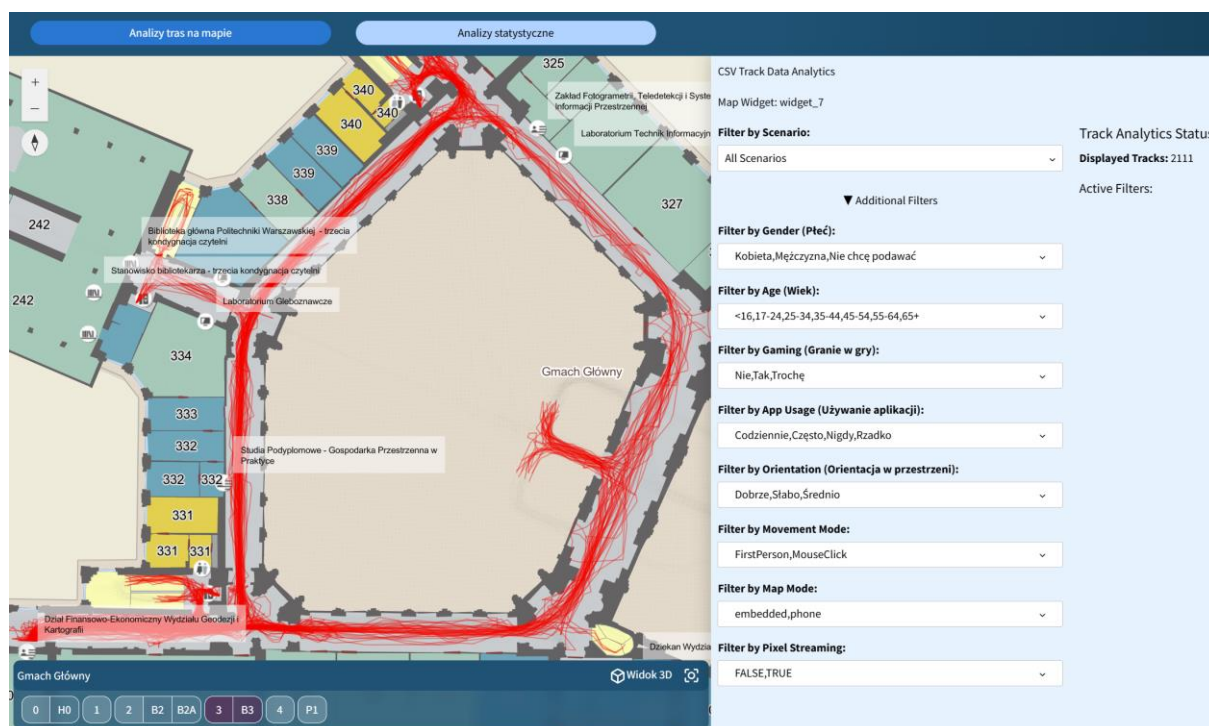
W ramach procesu rozwoju symulatora i środowiska wirtualnego *IndoorLab* oraz jego regularnego testowania i profilowania zastosowano wiele technik optymalizacyjnych. Poniżej wypisano najważniejsze z nich:

- Rozdzielenie geometrii każdego z budynków jako oddzielnego podpoziomu silnika Unreal.
- Stosowanie ISM tam gdzie to możliwe
- Zrezygnowanie z technologii Nanite jako nieadekwatnej do specyfiki projektu, a wpływającej na wydajność
- Zastosowanie technologii Lumen oraz jej konfiguracja w sposób preferujący wydajność ponad jakość oświetlenia
- Redukcja złożoności tekstur
- Dostosowanie ustawień skalowalności silnika
- Optymalizacje kodu w szczególności funkcji odpowiedzialnych za komunikację sieciową
- Optymalizacje konfiguracji procesów post-processingu obrazu
- Ograniczenie rozdzielczości w trybie chmurowym
- Optymalizacja parametrów strumieniowania obrazu w trybie chmurowym

Podsumowując, optymalizacja środowiska wirtualnego wiąże się z szeregiem złożonych wyzwań technicznych wymagających kompromisów między jakością a wydajnością. **Głównym wyzwaniem napotkanym podczas pracy nad rozprawą była przede wszystkim duża liczba obiektów tworzących modele budynków, która znacząco wpływa na płynność renderowania symulacji.** Kluczowe są optymalizacje wyświetlania geometrii przy jednoczesnym zachowaniu wiarygodnego odwzorowania pierwowzoru rzeczywistego, które nie wprowadzi w błąd uczestników badań ani nie zniekształci wyników eksperymentu. Dodatkowym wyzwaniem było pogodzenie wymagań dotyczących płynności działania aplikacji i jakości wyświetlanego obrazu z realnymi ograniczeniami sprzętowymi urządzeń użytkowników, a także w szczególności z ograniczeniami zasobów obliczeniowych dostępnych w infrastrukturze chmurowej.

6.14. Aplikacja wspomagająca analizę wyników badań

Oddzielnym komponentem, który również powstał w oparciu o aplikację *Mapa PW* po przeprowadzeniu badań, jest aplikacja wspierająca analizę uzyskanych wyników (Rysunek 6.45).



Rysunek 6.45. Ekran aplikacji przeznaczonej do analizy wyników (opracowanie własne)

Narzędzie to umożliwia wyświetlanie wszystkich zarejestrowanych tras jakimi przeszli uczestnicy badania w wirtualnych budynkach oraz filtrowanie ich według nazw i pięter, co pozwala na przestrzenną analizę wzorców ruchu użytkowników. Opracowana aplikacja oferuje zaawansowane możliwości filtrowania według zdefiniowanych scenariuszy badawczych i ich parametrów. Interfejs wizualizacyjny umożliwia jednocześnie porównywanie wielu tras, identyfikację problematycznych obszarów w budynkach oraz wykrywanie anomalii w zebranych danych. Rozwiązanie to pozwala na efektywne wykorzystanie danych – bez niego analiza byłaby znacznie utrudniona ze względu na bardzo dużą liczbę punktów pomiarowych tworzących zarejestrowane trasy. Implementacja narzędzia analitycznego stanowi kluczowy element w procesie badawczym oraz pomaga w walidacji użyteczności przygotowanego systemu i została wykorzystana do analizy wyników uzyskanych podczas badania, co opisano w rozdziale 7.

Poniżej przedstawiono listę dostępnych filtrów (filtry odpowiadają danym zbieranym podczas procesu badawczego, które opisano szczegółowo w rozdziale 5.4):

1. Scenariusz (Scenario) - filtr jednokrotnego wyboru do selekcji konkretnego scenariusza badawczego

Filtry warunkowe (zależne od wybranego scenariusza)

2. Tryb widoczności mapy (Map Visibility Mode) - dostępny dla scenariuszy: *et1*, *et2*
3. Tryb trasy (Route Mode) - dostępny dla scenariuszy: *chem1*, *chem2*, *chem3*, *gg2*, *archi1*
4. Tryb lokalizacji (Location Mode) - dostępny dla scenariuszy: *chem1*, *chem2*, *chem3*
5. Tryb rotacji (Rotation Mode) - dostępny dla scenariusza: *gg1*
6. Tryb wskazówek (Hints Mode) - dostępny dla scenariusza: *gg3*
7. Tryb zoomu (Zoom Mode) - dostępny dla scenariuszy: *eti1*, *eti2*

Filtry dodatkowe (w sekcji rozwijanej "Additional Filters")

8. Spojrzenie na mapę (Looked at map at least once) - checkbox filtrujący trasy, gdzie użytkownik przynajmniej raz spojrzał na mapę
9. Znajomość budynku (Does not know this building) - checkbox filtrujący trasy użytkowników nieznających budynku
10. Płeć (Gender) - wybór płci uczestników
11. Wiek (Age) - wybór grup wiekowych
12. Granie w gry (Gaming) - wybór opcji dotyczących doświadczenia z grami
13. Używanie aplikacji (App Usage) - wybór opcji dotyczących używania aplikacji nawigacyjnych
14. Orientacja w przestrzeni (Orientation) - wybór opcji dotyczących zdolności orientacji przestrzennej
15. Tryb ruchu (Movement Mode) - wybór trybów poruszania się
16. Tryb mapy (Map Mode) - wybór trybów wyświetlania mapy
17. Pixel Streaming - wybór opcji strumieniowania pikseli

Łącznie opracowana aplikacja oferuje 17 różnych filtrów, z czego 6 filtrów jest wyświetlanych warunkowo w zależności od wybranego scenariusza badawczego.

7. Badanie na użytkownikach z wykorzystaniem symulatora

7.1. Cele i ogólna charakterystyka badania

Podstawowym celem przeprowadzonego badania z udziałem użytkowników była weryfikacja zaproponowanej metodyki badawczej oraz demonstracja jej praktycznej użyteczności w kontekście testowania rozwiązań nawigacyjnych dla przestrzeni wewnątrz budynków.

Projektując przebieg badania, szczególną uwagę zwracano na możliwie szerokie przetestowanie różnych funkcji symulatora oraz aplikacji nawigacyjnej, a także możliwości, jakie daje ich połączenie. Priorytetem było wykazanie faktycznej użyteczności i potencjału ukrytego w środowisku wirtualnym jako narzędziu badawczym. Przeprowadzone eksperymenty miały zatem przede wszystkim charakter demonstracyjny i eksploracyjny, pokazując spektrum możliwości analitycznych oferowanych przez zaproponowane podejście metodyczne.

Badanie zostało zaprojektowane jako wieloaspektowy eksperyment, który umożliwił równoczesne testowanie różnych wariantów przekazu kartograficznego, typów wskazówek nawigacyjnych oraz funkcjonalności aplikacji w kontrolowanych warunkach symulacyjnych. Zastosowanie środowiska wirtualnego pozwoliło na odwzorowanie niektórych warunków i procesów mentalnych, które zachodzą podczas nawigacji po wnętrzach budynków, zapewnienie pełnej kontroli nad zmiennymi eksperymentalnymi i umożliwiło automatyczne rejestrowanie szczegółowych danych behawioralnych uczestników. Dzięki temu możliwa była zarówno analiza efektywności nawigacji (mierzona czasem realizacji zadań i poprawnością wyborów tras) jak i aspektów poznawczych związanych z zapamiętywaniem przestrzeni budynku. Elementem badania było także zebranie opinii uczestników eksperymentu na temat wykorzystywanej aplikacji badawczej.

Należy podkreślić, że przeprowadzone badanie nie miało na celu dostarczenia, ostatecznych odpowiedzi na postawione pytania związane z optymalnym projektowaniem interfejsów aplikacji nawigacyjnych czy szczegółowymi zasadami kartografii wnętrz budynków, a jedynie

potwierdzenie poprawności działania zaprojektowanej platformy badawczej i powiązanej z nią metodyki badań. Niemniej jednak, zgromadzone dane empiryczne umożliwiają wyciąganie wstępnych wniosków dotyczących pewnych aspektów projektowania tych aplikacji. Wyniki badania mają charakter przede wszystkim inspirujący dla dalszych, bardziej szczegółowych analiz poszczególnych aspektów nawigacji wewnątrz budynków z wykorzystaniem skonstruowanej platformy.

Podczas badania uczestnicy najpierw zapoznawali się z podstawową obsługą symulatora na Poziomie treningowym, a następnie przechodzili przez 16 zdefiniowanych scenariuszy badawczych. Pierwsze trzy scenariusze są z założenia scenariuszami rozgrzewkowymi, które pozwalają na dalsze zapoznanie się ze wszystkimi funkcjami wirtualnego środowiska. Pozostałe 13 scenariuszy składa się na 7 eksperymentów, różniących się pomiędzy sobą założeniami, celem oraz stawioną hipotezą. Eksperymenty te zostały szczegółowo opisane w podrozdziale 7.5.

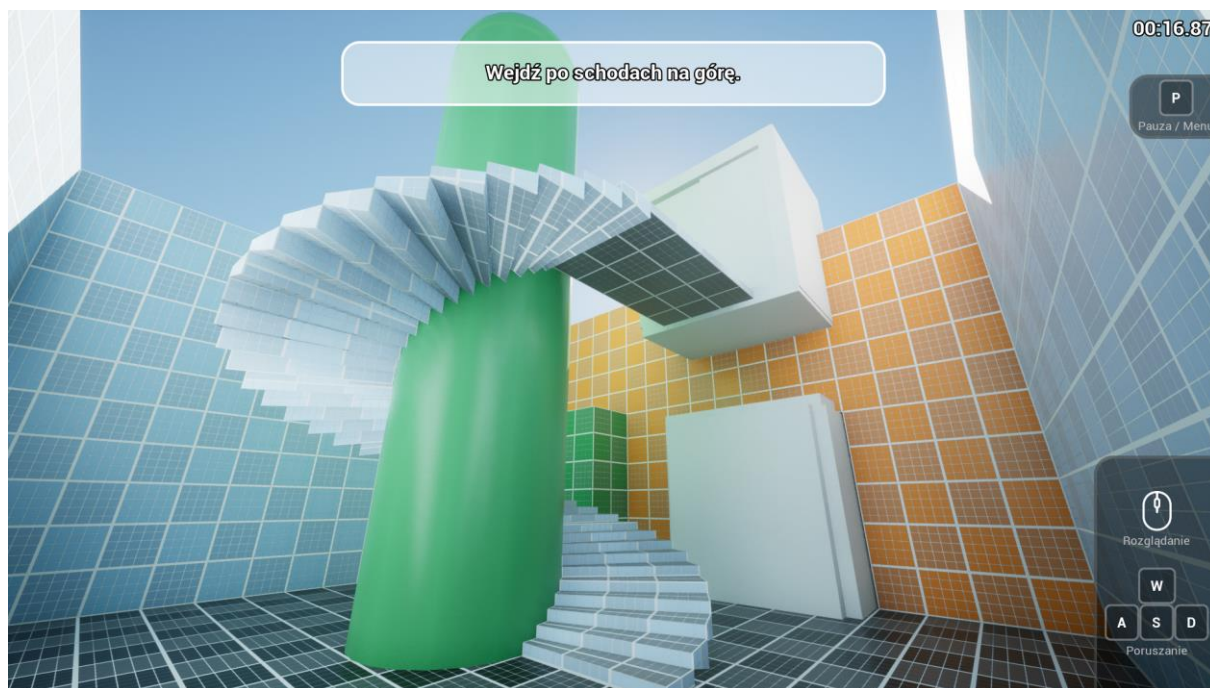
7.2. Poziom treningowy oraz scenariusze testowe

Przed przystąpieniem do właściwego badania uczestnicy przechodzili dwuetapowy proces przygotowawczy, mający na celu zapoznanie ich z mechaniką symulatora oraz wypracowanie umiejętności niezbędnych do sprawnego poruszania się w wirtualnym środowisku. Proces ten składał się z poziomu treningowego oraz trzech scenariuszy testowych, przeprowadzonych w cyfrowym odpowiedniku rzeczywistego obiektu.

Poziom treningowy został zaprojektowany jako interaktywne środowisko przypominające grę komputerową, w którym uczestnicy mogli w kontrolowanych warunkach przećwiczyć podstawowe mechanizmy sterowania i poruszania się w symulacji. Środowisko to zawierało szereg charakterystycznych elementów architektonicznych, takich jak schody, winda oraz inne przeszkody przestrzenne, które wymagały od użytkowników opanowania poruszania się w przestrzeni wirtualnej.

Przejście poziomu treningowego miało kluczowe znaczenie dla zapewnienia rzetelności właściwego badania. Pozwalało uczestnikom na praktyczne poznanie interfejsu sterowania, zrozumienie zasad interakcji z elementami środowiska oraz wypracowanie intuicyjnego

podejścia do nawigacji w trójwymiarowej przestrzeni symulowanego budynku. Dodatkowo, etap ten umożliwiał identyfikację potencjalnych trudności w obsłudze symulatora oraz ich ograniczenie przed rozpoczęciem właściwego badania.



Rysunek 7.1. Zrzut ekranu prezentujący fragment poziomego treningowego (opracowanie własne)

Po ukończeniu poziomego treningowego uczestnicy przystępowali do wypełnienia kwestionariusza danych demograficznych, informacji o swoich kompetencjach oraz znajomości budynków, w których odbywać się będą eksperymenty właściwe (Rysunek 7.2).

Ankieta początkowa

Email - otrzymasz podsumowanie wyników badań

Wiek *

Wybierz opcję...

Płeć *

Wybierz opcję...

Czy potrafisz grać w gry komputerowe? *

Wybierz opcję...

Jak często korzystasz z aplikacji nawigacyjnych w codziennym życiu? *

Wybierz opcję...

Jak oceniasz swoją orientację w przestrzeni w nieznanymi miejscach? *

Wybierz opcję...

Zaznacz budynki kampusu Politechniki Warszawskiej,
w których już kiedyś byłeś:

☐ Gmach Elektroniki

☐ Gmach Elektrotechniki

☐ Gmach Główny

☐ Gmach Architektury

☐ Gmach Chemii

Rozpocznij

Rysunek 7.2. Zrzut ekranu prezentujący ankietę początkową (opracowanie własne)

Następnie konieczna była realizacja trzech scenariuszy testowych, osadzonych w cyfrowym bliźniaku budynku CENAGIS w Józefostawiu. Scenariusze te stanowiły bezpośrednie przygotowanie do właściwego badania, prezentując różne warianty wspomagania nawigacyjnego oraz mechaniki, które miały być wykorzystane w późniejszych etapach eksperymentu. W wybranych miejscach w scenariuszach testowych umieszczono obiekty ze znakiem zapytania. Stanięcie w ich pobliżu powodowało wyświetlenie przez aplikację badawczą instrukcji w formie tekstowej (Rysunek 7.3).



Rysunek 7.3. Zrzut ekranu prezentujący pierwszy scenariusz testowy, z widocznym znakiem zapytania przy którym w górnej części ekranu prezentowane są dodatkowe informacje w formie tekstowej (opracowanie własne)

Scenariusze testowe pełniły również funkcję standaryzującą – dzięki przejściu przez identyczny proces przygotowawczy wszyscy uczestnicy badania rozpoczynali właściwy eksperyment z porównywalnym poziomem znajomości mechanik symulatora oraz zasad obowiązujących podczas nawigacji. To podejście umożliwiało bardziej precyzyjne porównanie wyników między różnymi grupami uczestników.

Zarówno podczas poziomu treningowego jak i scenariuszy testowych nie były jeszcze zbierane dane statystyczne o wynikach uczestników.

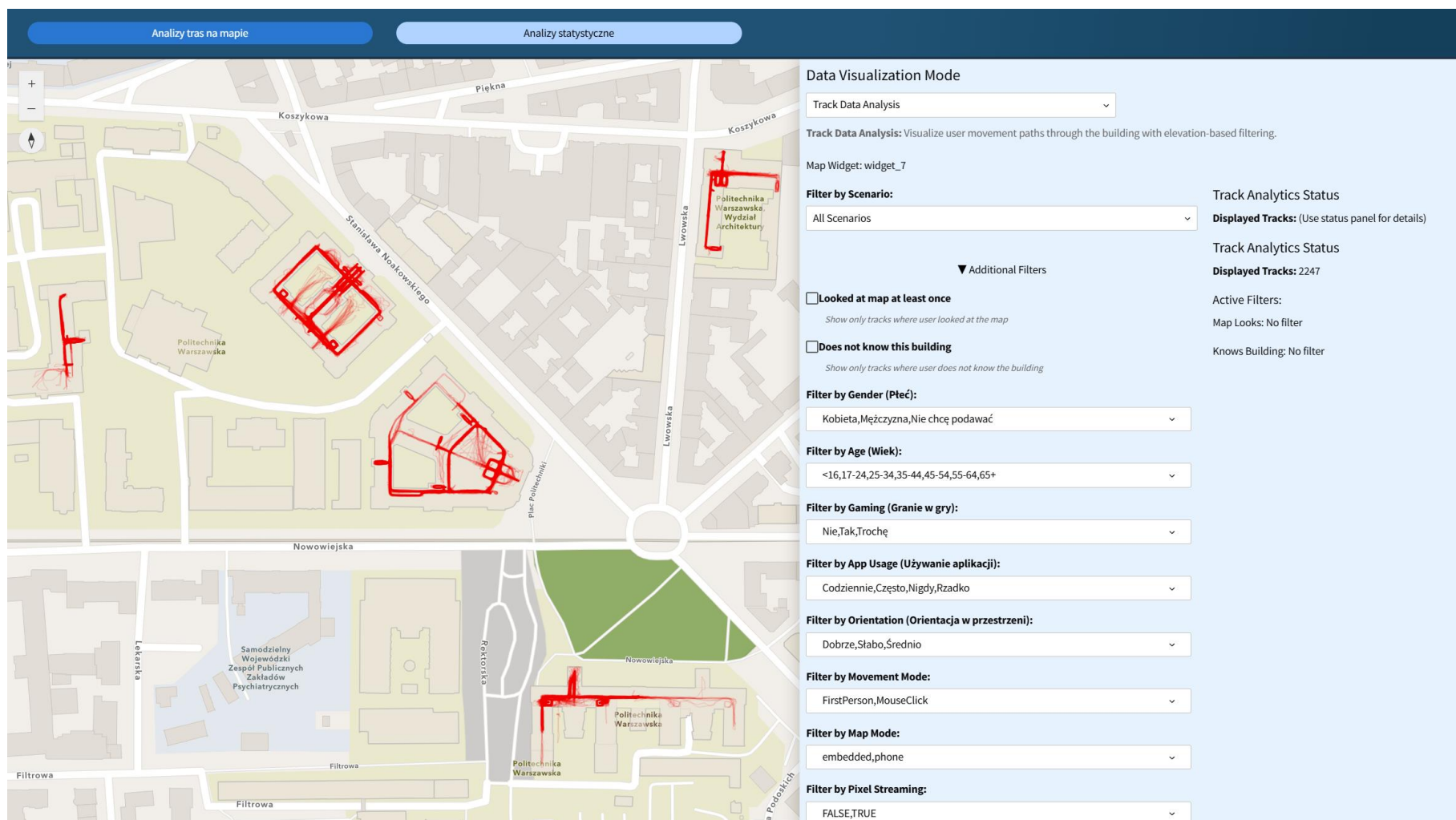
7.3. Narzędziach wykorzystane do analizy danych

Do analizy wszystkich danych zebranych w formie arkusza kalkulacyjnego wykorzystano trzy narzędzia. Narzędzie Looker Studio²⁵, specjalnie opracowaną aplikację mapową, służącą do wyświetlania i analizy zarejestrowanych tras oraz zestaw skryptów języka Python, służących do analizy statystycznej zebranych danych.

²⁵ Adres internetowy narzędzia Looker Studio: <https://lookerstudio.google.com/>

Opracowano zarówno raporty dotyczące ogólnych statystyk wszystkich zebranych danych jak i bardziej szczegółowe raporty dla każdego z eksperymentów. Wnioski oraz uzyskane wykresy zostały przedstawione w kolejnych podrozdziałach. Oryginalne, dynamiczne raporty zostały również osadzone jako zakładka w opisanej poniżej aplikacji analitycznej.

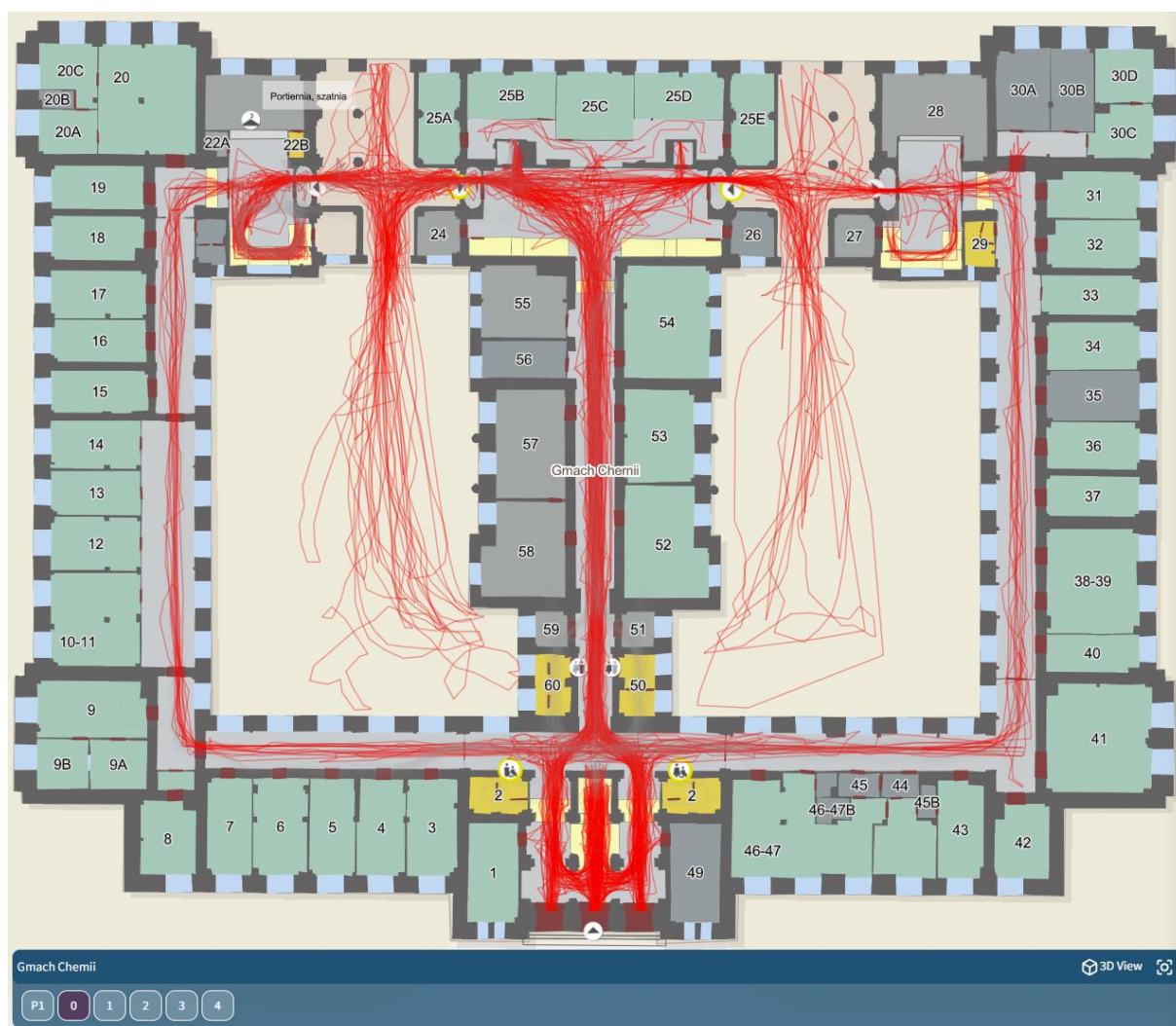
W ramach przeprowadzonego eksperymentu opracowano aplikację analityczną (Rysunek 7.4) umożliwiającą kompleksową analizę zebranych danych. Narzędzie zostało zaprojektowane z myślą o wizualizacji i interpretacji danych przestrzennych oraz statystycznych zgromadzonych podczas badań nad nawigacją wewnątrz budynków w środowisku wirtualnym.



Rysunek 7.4. Wygląd aplikacji analitycznej bezpośrednio po uruchomieniu. Na górze dwa przyciski umożliwiające zmianę trybu, po lewej mapa z naniesionymi trasami przejścia uczestników, po prawo widok filtrów wykorzystywanych w trybie analiz przestrzennych (opracowanie własne)

Aplikacja analityczna oferuje dwa podstawowe tryby pracy: tryb analizy danych przestrzennych oraz tryb analiz statystycznych. W trybie analiz statystycznych wyświetlana jest strona z raportami Looker Studio. W trybie analiz danych przestrzennych dostępne są dwa moduły analityczne, w zależności od rodzaju zebranych danych.

Pierwszy moduł analizy przestrzennej koncentruje się na wizualizacji zarejestrowanych tras przejść uczestników badania (Rysunek 7.5). Analiza zmiany trendów ruchu osób, z wykorzystaniem wielu dostępnych filtrów pozwala na identyfikację wzorców i potencjalnie wysuwanie wniosków dotyczących wpływu poszczególnych czynników na trasę obieraną przez uczestników badania.



Rysunek 7.5. Przykładowy fragment wizualizacji zarejestrowanych tras, jakie uczestników badania przebyli w wirtualnym Gmachu Chemii na poziomie parteru (opracowanie własne)

Drugi moduł wykorzystuje dane o dokładnej lokalizacji miejsc, w których uczestnicy badania posiłkowali się wyświetleniem mapy budynku (Rysunek 7.6). Lokalizacja była rejestrowana w momencie naciśnięcia przez uczestnika klawisza spacji, co podczas badania skutkowało otwarciem podglądu mapy. Na podstawie zgromadzonych danych generowana jest tzw. mapa ciepła (ang. *heat map*). Wizualizacja ta umożliwia określenie, w których częściach budynku i na jakich piętrach mapa była najczęściej wykorzystywana przez uczestników podczas realizacji zadania nawigacyjnego, a tym samym pozwala łatwo zidentyfikować obszary budynku sprawiające największe problemy podczas poszukiwania miejsc docelowych (największa liczba spojrzeń na mapę w poszukiwaniu trasy). Przygotowana mapa ciepła (ang. *heat map*) nie została ostatecznie wykorzystana podczas analizy poszczególnych eksperymentów opisanych w dalszej części tego rozdziału, z uwagi na potrzebę dalszej normalizacji danych, co było poza zakresem prac planowanych w rozprawie. Mimo to zebrane dane oraz wstępnie przygotowane narzędzie wykazuje duży potencjał do dalszych badań.



Rysunek 7.6. Przykładowy fragment wizualizacji miejsc, w których uczestnicy badania korzystali z mapy w wirtualnym Gmachu Chemii na poziomie parteru (opracowanie własne)

Kluczową funkcjonalnością aplikacji analitycznej jest zaawansowany system filtrowania danych. Wszystkie wyświetlane informacje mogą być filtrowane na podstawie parametrów zbieranych dla każdego uczestnika. Szczególnie istotna jest możliwość filtrowania danych według konkretnego scenariusza eksperymentalnego, co pozwala na wyświetlenie wyłącznie tych wyników, które dotyczą jednej z badanych zmiennych niezależnych – parametrów nawigacji modyfikowanych w danym scenariuszu.

Podobnie jak aplikacja *Mapa PW* aplikacja analityczna została zrealizowana w środowisku ArcGIS Experience Builder, co pozwoliło na wykorzystanie wcześniej opracowanych widżetów odpowiedzialnych za wyświetlanie mapy wnętrza budynków. W ramach implementacji utworzono nowy widżet przeznaczony do analizy, wyświetlania i filtrowania zebranych danych. Wybrane wizualizacje wygenerowane za pomocą aplikacji analitycznej zostały wykorzystane do szczegółowej analizy wyników eksperymentu przedstawionej w kolejnych podrozdziałach niniejszego rozdziału. Aplikacja jest dostępna online dla osób zainteresowanych pogłębioną analizą zebranych danych lub weryfikacją przedstawionych wniosków badawczych.

<https://mapa.pw.edu.pl/indoorlab/analytics/>

(dostępność przynajmniej do końca roku 2026)

W celu przeprowadzenia pogłębionej analizy statystycznej zebranych danych eksperymentalnych opracowano także zestaw skryptów napisanych w języku Python. Skrypty te zostały zaprojektowane do analizy każdego z przeprowadzonych eksperymentów pod kątem wpływu zmiennych niezależnych na zmierzony czas oraz przebyty dystans, a także poszukiwania korelacji pomiędzy poszczególnymi zmiennymi. Wszystkie skrypty zostały zaimplementowane w formacie Jupyter Notebook, co umożliwiło interaktywną analizę danych oraz bezpośrednie dokumentowanie procesu badawczego. Każdy fragment kodu został opatrzony opisem wyjaśniającym jego funkcjonalność oraz zastosowanie w kontekście prowadzonego eksperymentu. Do realizacji zadań analitycznych wykorzystano następujące, popularne biblioteki języka Python:

- pandas - do zarządzania i manipulacji danymi w formie tabelarycznej
- numpy - do obliczeń numerycznych i operacji na macierzach
- matplotlib.pyplot - do tworzenia wizualizacji i wykresów

Opracowane skrypty wspomagają proces analizy danych, obejmujący kilka kluczowych etapów. Pierwszym z nich jest załadowanie surowych danych zebranych podczas przeprowadzonych eksperymentów, które zostały wcześniej wyeksportowane z arkusza Google do formatu csv. Następnie dane przechodzą proces przetwarzania i oczyszczania, który obejmuje usuwanie nieprawidłowych wartości, standaryzację formatów, kodowanie wartości oraz weryfikację spójności zebranych informacji. Po przygotowaniu danych do analiz statystycznych, skrypty przeprowadzają analizę wzajemnych korelacji wszystkich zmiennych, co pozwala na identyfikację istotnych zależności między parametrami nawigacyjnymi a zachowaniami uczestników badania.

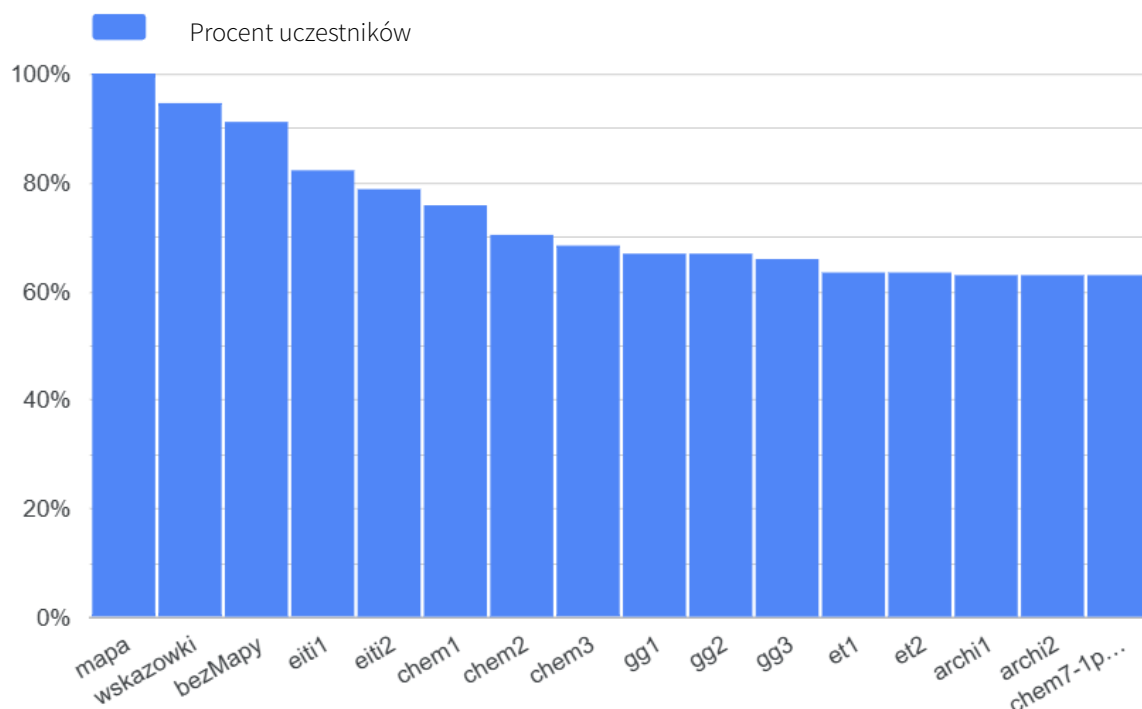
Przygotowano również skrypty wykonujące analizy specyficzne dla każdego ze scenariuszy eksperymentalnych, uwzględniające unikalne cechy i zmienne charakterystyczne dla poszczególnych wariantów badania. Umożliwia to szczegółową interpretację wyników w kontekście konkretnych hipotez badawczych.

Wnioski z analiz statystycznych, a także najistotniejsze wykresy wykorzystano do opisu poszczególnych eksperymentów w kolejnych podrozdziałach. Wszystkie opracowane skrypty analityczne zostały dołączone do niniejszej pracy w załączniku 6, co umożliwia weryfikację zastosowanych metod oraz odtworzenie przeprowadzonych analiz.

7.4. Analiza grupy badawczej

W badaniu wzięło udział **205 uczestników**, z czego **127 ukończyło** je w pełni przechodząc przez wszystkie 16 scenariuszy. Łącznie zarejestrowano 2679 przejść zaplanowanymi trasami (2032 przejścia przez osoby, które ukończyły badanie w pełni oraz 647 przez osoby, które zrezygnowały przed końcem). Poniższy wykres przedstawia jaki procent uczestników brał udział w kolejnych etapach badania (Rysunek 7.7).

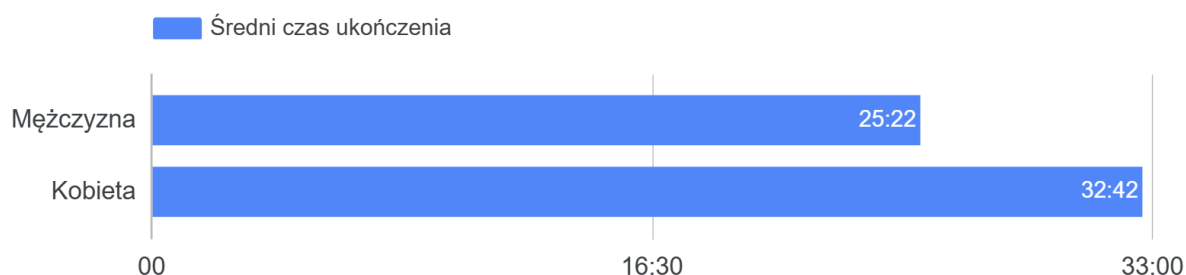
Liczba uczestników w kolejnych scenariuszach



Rysunek 7.7. Procentowy udział uczestników eksperymentu, którzy pomyślnie zakończyli poszczególne zadania nawigacyjne. Wysokość słupków przedstawia odsetek użytkowników, którzy ukończyli dany scenariusz. Na osi X widoczne skrótowe nazwy poszczególnych scenariuszy, w większości są to skróty nazw budynków po których poruszali się użytkownicy (opracowanie własne).

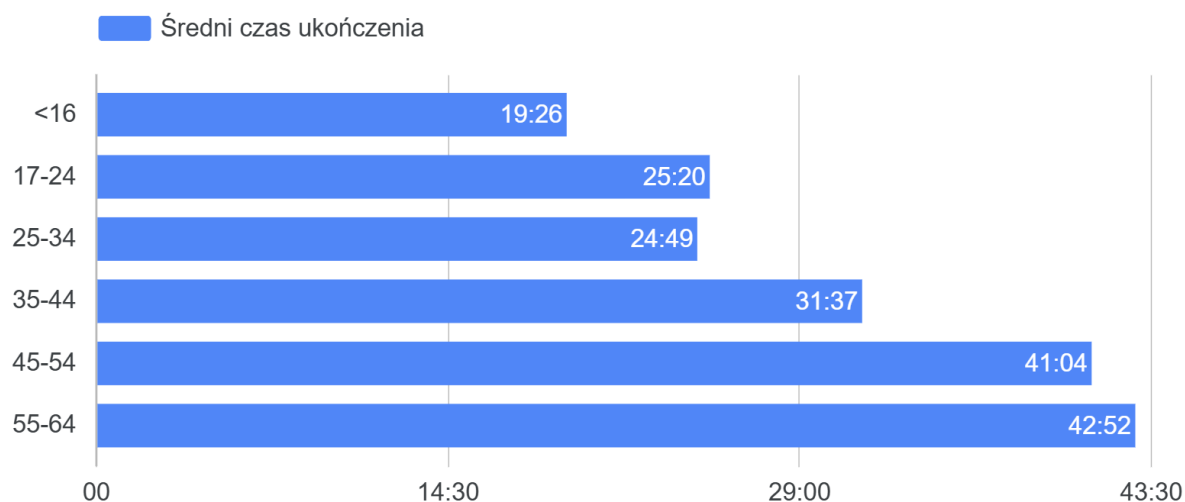
Średni łączny czas ukończenia całego badania, to ponad **27 minut**. Na kolejnych wykresach (Rysunek 7.8, Rysunek 7.9) przedstawiono średni czas w podziale na płeć oraz grupy wiekowe.

Średni czas ukończenia badania według płci



Rysunek 7.8. Średni czas ukończenia całego badania w podziale na płeć (opracowanie własne)

Średni czas ukończenia badania według wieku

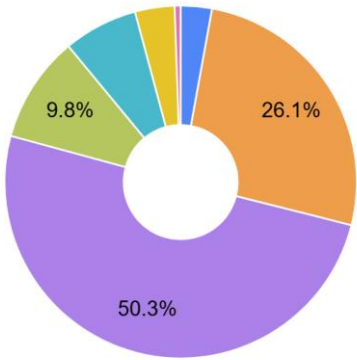
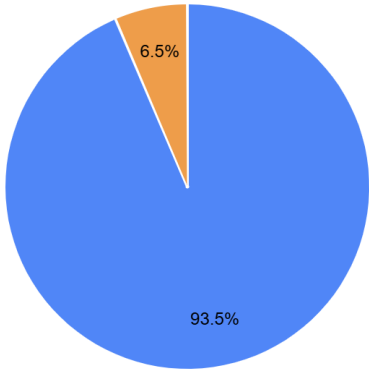
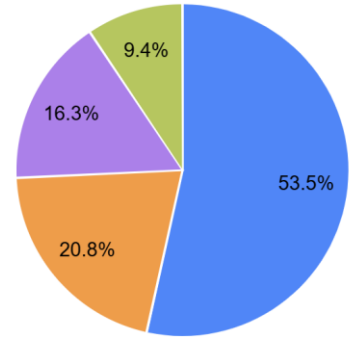
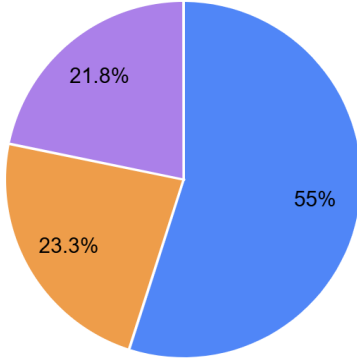


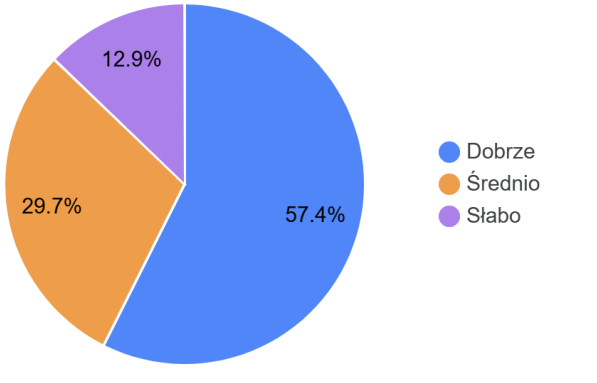
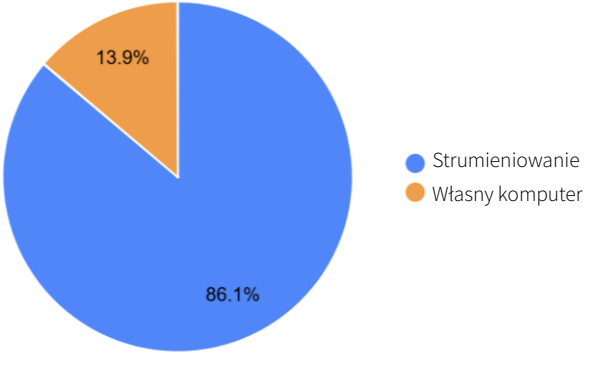
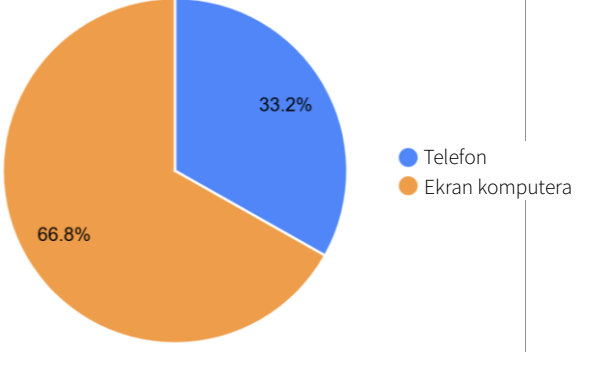
Rysunek 7.9. Średni czas ukończenia całego badania w podziale na grupy wiekowe (opracowanie własne)

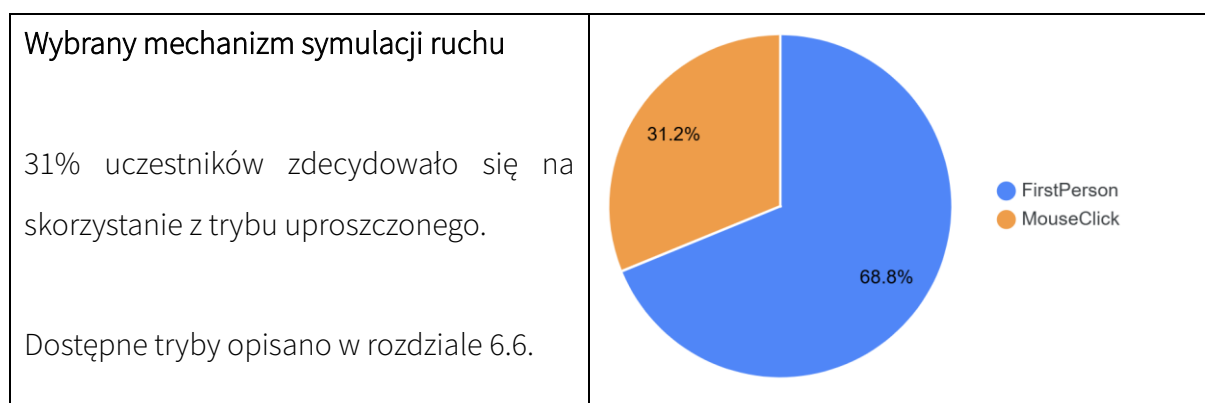
Tabela 8 przedstawia podstawowe informacje o uczestnikach badania, w tym ich dane demograficzne (wiek, płeć), poziom doświadczenia i umiejętności oraz decyzje/preferencje wyrażone na początku eksperymentu, zwizualizowane za pomocą wykresów kołowych dla każdego z parametrów.

Tabela 8. Statystyki dotyczące danych demograficznych, ich doświadczeń i umiejętności, a także ich wyborów na wczesnym etapie badania (opracowanie własne)

Opis	Wykres								
Płeć uczestników Prawie 70% uczestników badania to mężczyźni.	<table border="1"> <thead> <tr> <th>Płeć</th> <th>Procent</th> </tr> </thead> <tbody> <tr> <td>Mężczyzna</td> <td>67.3%</td> </tr> <tr> <td>Kobieta</td> <td>29.7%</td> </tr> <tr> <td>Nie chcę podawać</td> <td>3.0%</td> </tr> </tbody> </table>	Płeć	Procent	Mężczyzna	67.3%	Kobieta	29.7%	Nie chcę podawać	3.0%
Płeć	Procent								
Mężczyzna	67.3%								
Kobieta	29.7%								
Nie chcę podawać	3.0%								

Grupy wiekowe Okolo 75% uczestników mieści się w przedziale od 17 do 34 roku życia.	 <16 17-24 25-34 35-44 45-54 55-64 65+
Wybrany język Zdecydowana większość uczestników wybrała polski język interfejsu.	 pl en
Częstotliwość korzystania z aplikacji nawigacyjnych Prawie 70% uczestników korzysta z aplikacji nawigacyjnych często lub codziennie.	 Często Rzadko Codziennie Nigdy
Umiejętność grania w gry komputerowe 22% uczestników badania zadeklarowało brak umiejętności grania w gry komputerowe.	 Tak Trochę Nie

Orientacja w przestrzeni Jedynie 13% uczestników określiło własną orientację w przestrzeni jako „słabą”.	 Legend: Dobrze (blue), Średnio (orange), Słabo (purple) <table border="1"> <thead> <tr> <th>Orientacja</th> <th>Procent</th> </tr> </thead> <tbody> <tr> <td>Dobrze</td> <td>57.4%</td> </tr> <tr> <td>Średnio</td> <td>29.7%</td> </tr> <tr> <td>Słabo</td> <td>12.9%</td> </tr> </tbody> </table>	Orientacja	Procent	Dobrze	57.4%	Średnio	29.7%	Słabo	12.9%
Orientacja	Procent								
Dobrze	57.4%								
Średnio	29.7%								
Słabo	12.9%								
Wariant dostępu do środowiska Jedynie 14% uczestników zdecydowało się na pobranie symulatora badawczego na własny komputer. 86% preferowało skorzystanie z trybu strumieniowania z chmury, mimo ostrzeżenia o potencjalnie gorszej jakości, co potwierdza iż zastosowane podejście technologiczne istotnie ułatwia udział w badaniu. Warianty dostępu opisano w rozdziale 6.7.	 Legend: Strumieniowanie (blue), Własny komputer (orange) <table border="1"> <thead> <tr> <th>Wariant dostępu</th> <th>Procent</th> </tr> </thead> <tbody> <tr> <td>Strumieniowanie</td> <td>86.1%</td> </tr> <tr> <td>Własny komputer</td> <td>13.9%</td> </tr> </tbody> </table>	Wariant dostępu	Procent	Strumieniowanie	86.1%	Własny komputer	13.9%		
Wariant dostępu	Procent								
Strumieniowanie	86.1%								
Własny komputer	13.9%								
Metody korzystania z aplikacji nawigacyjnej 1/3 uczestników zdecydowała się na skorzystanie ze smartfonu jako urządzenia wyświetlającego mapę. Metody korzystania z mapy opisano w rozdziale 6.8.2.	 Legend: Telefon (blue), Ekran komputera (orange) <table border="1"> <thead> <tr> <th>Metoda korzystania</th> <th>Procent</th> </tr> </thead> <tbody> <tr> <td>Telefon</td> <td>33.2%</td> </tr> <tr> <td>Ekran komputera</td> <td>66.8%</td> </tr> </tbody> </table>	Metoda korzystania	Procent	Telefon	33.2%	Ekran komputera	66.8%		
Metoda korzystania	Procent								
Telefon	33.2%								
Ekran komputera	66.8%								



W celu zbadania wzajemnych zależności między cechami użytkowników oraz wybranymi przez nich zmiennymi interfejsu, przeprowadzono analizę korelacji wewnętrznych metodą Spearmana dla 8 kluczowych zmiennych (* korelacje istotne statystycznie przy $p < 0,05$ Rysunek 7.10). Analiza objęła 28 unikalnych par zmiennych, z czego 21 (75%) wykazało istotność statystyczną przy $p < 0,05$, w tym 2 korelacje silne ($|r| \geq 0,3$), 12 umiarkowanych ($0,1 \leq |r| < 0,3$) i 7 słabych ($|r| < 0,1$).

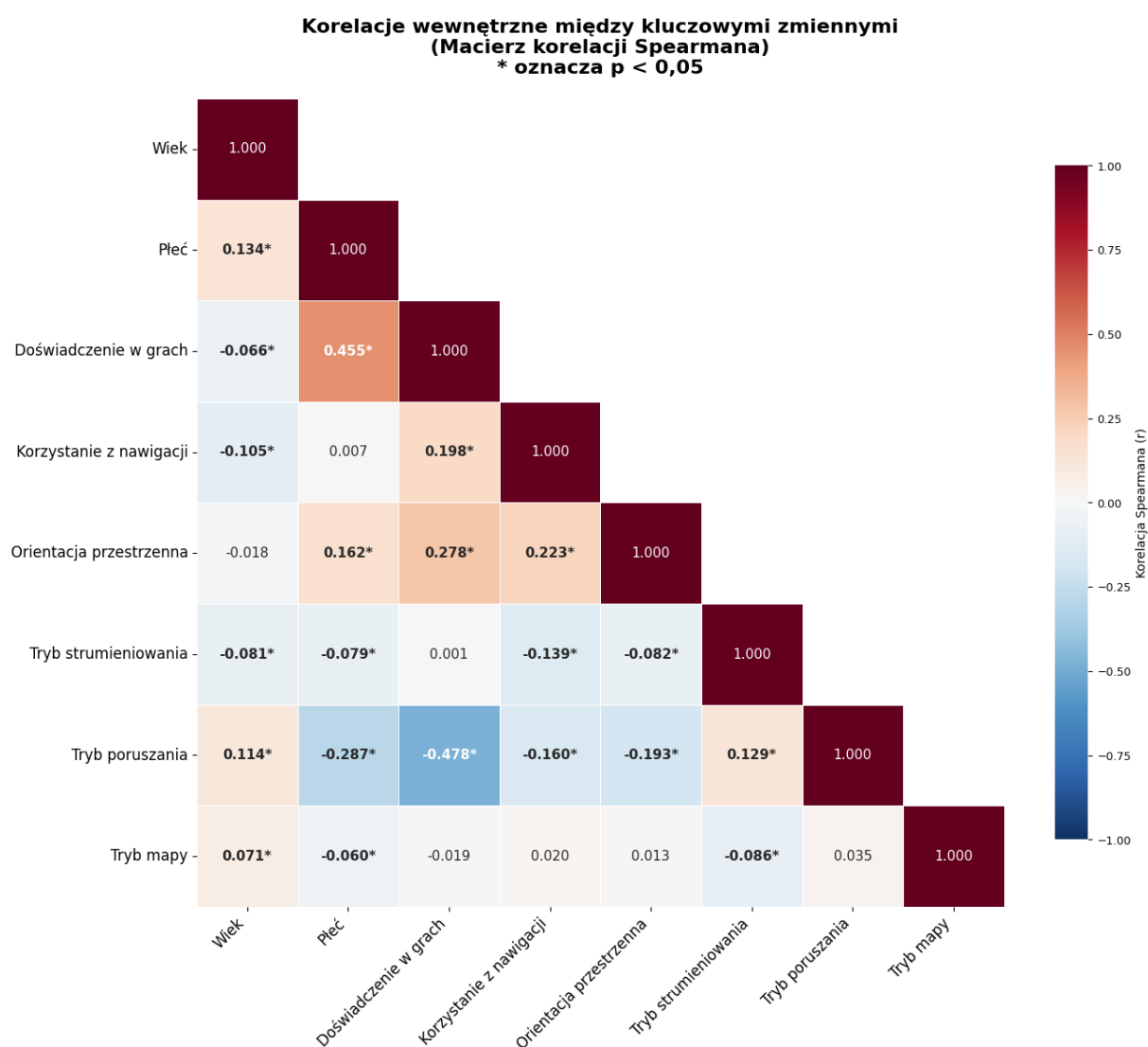
Najsilniejsze korelacje wewnętrzne:

1. **Doświadczenie w grach, a tryb poruszania** ($r = -0,478$, $p < 0,001$) – najsilniejsza ujemna korelacja w całej macierzy, wskazująca, że osoby z większym doświadczeniem w grach wideo preferują standardowy tryb kontroli poruszania.
2. **Płeć, a doświadczenie w grach** ($r = 0,455$, $p < 0,001$) – silna dodatnia korelacja sugerująca znaczące różnice w doświadczeniu z grami komputerowymi pomiędzy płciami w badanej populacji (mężczyźni częściej grają w gry).

Wysokie wskaźniki istotności statystycznej wskazują na istnienie złożonej sieci powiązań między badanymi zmiennymi. Szczególnie istotne są silne korelacje doświadczenia w grach z trybem poruszania i płcią, co sugeruje, że doświadczenie z grami komputerowymi może być kluczowym moderatorem w badaniach opartych o środowiska wirtualne.

Dodatnia korelacja między orientacją przestrzenną a doświadczeniem w grach ($r = 0,278$) oraz korzystaniem z aplikacji nawigacyjnych ($r = 0,223$) potwierdza związek między różnymi aspektami zdolności przestrzennych i ich wykorzystaniem w życiu codziennym. Ujemna korelacja orientacji przestrzennej z trybem poruszania ($r = -0,193$) może wskazywać, że osoby o słabszej orientacji przestrzennej preferują lub efektywniej wykorzystują określony tryb kontroli.

Stosunkowo słabe korelacje wieku z innymi zmiennymi (większość $|r| < 0,15$) sugerują, że wiek sam w sobie nie jest silnym predyktorem doświadczenia w grach czy orientacji przestrzennej w badanej populacji, chociaż można zaobserwować lekką ujemną korelację wieku oraz doświadczenia z grami ($r = -0,066$) (osoby młodsze częściej grają w gry).

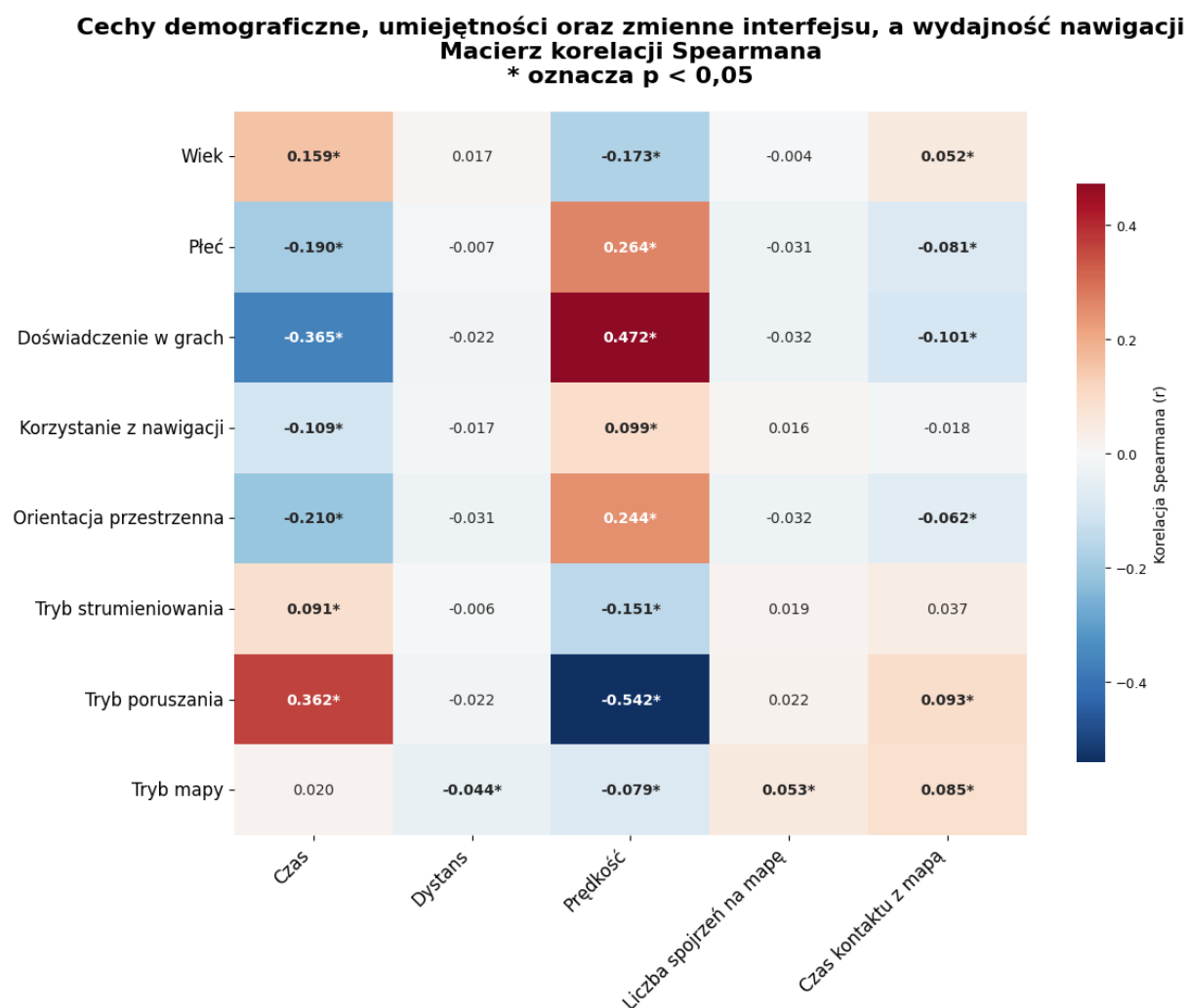


* korelacje istotne statystycznie przy $p < 0,05$

Rysunek 7.10. Macierz korelacji wewnętrznych Spearmana między cechami demograficznymi, umiejętnościami użytkowników i zmiennymi interfejsu. Przedstawiono tylko dolny trójkąt macierzy ze względu na symetrię korelacji (opracowanie własne).

W celu zbadania związków między charakterystykami użytkowników, ich wyborami dotyczącymi interfejsu, a wydajnością nawigacji, przeprowadzono analizę korelacji Spearmana (* korelacje istotne statystycznie przy $p < 0,05$)

Rysunek 7.11). Analiza objęła 2354 obserwacje i wykazała 23 istotne statystycznie korelacje (57,5% wszystkich testowanych par, $p < 0,05$), w tym 4 korelacje silne ($|r| \geq 0,3$), 9 umiarkowanych ($0,1 \leq |r| < 0,3$) i 10 słabych ($|r| < 0,1$).



* korelacje istotne statystycznie przy $p < 0,05$

Rysunek 7.11. Macierz korelacji Spearmana między cechami demograficznymi, umiejętnościami użytkowników, zmiennymi interfejsu, a wydajnością nawigacji w środowisku wirtualnym (opracowanie własne)

Najsilniejsze efekty zaobserwowano dla zmiennej prędkość poruszania będącej pochodną czasu i dystansu:

1. **Tryb poruszania** (patrz rozdział 6.6) wykazał najsilniejszą korelację ujemną z prędkością ($r = -0,542$, $p < 0,001$), wskazując, że wybór trybu kontroli znacząco wpływa na szybkość poruszania się w środowisku wirtualnym.
2. **Doświadczenie w grach** było silnie skorelowane dodatnio z prędkością ($r = 0,472$, $p < 0,001$), co potwierdza, że użytkownicy z większym doświadczeniem w grach wideo poruszają się sprawniej w przestrzeni wirtualnej.

3. **Płeć** wykazała umiarkowaną korelację dodatnią z prędkością ($r = 0,264$, $p < 0,001$), oraz istotne korelacje z czasem wykonania zadania ($r = -0,190$, $p < 0,001$), co oznacza, że statystycznie mężczyźni byli nieco szybsi od kobiet.
4. **Orientacja przestrzenna** była umiarkowanie skorelowana z prędkością ($r = 0,244$, $p < 0,001$) i ujemnie z czasem ($r = -0,210$, $p < 0,001$), co sugeruje, że osoby z lepszą orientacją przestrzenną poruszają się szybciej i potrzebują mniej czasu na ukończenie zadania.

Warto zauważyć, że dystans pokonany przez uczestników wykazał znacznie słabsze korelacje z cechami użytkowników i zmiennymi interfejsu w porównaniu do czasu i prędkości. Może to wskazywać, że długość pokonanej trasy jest bardziej zdeterminowana przez układ budynku i lokalizację celu niż przez indywidualne cechy użytkowników, co czyni dystans mało istotną miarą wydajności nawigacyjnej w kontekście różnic indywidualnych.

Wyniki analizy statystycznej wskazują, że doświadczenie w grach wideo oraz wybór trybu poruszania są kluczowymi czynnikami wpływającymi na wydajność odnajdowania celu z pomocą wskazówek nawigacyjnych w środowisku wirtualnym. Cechy demograficzne (wiek, płeć) oraz zdolności poznawcze (orientacja przestrzenna) również odgrywają istotną rolę, szczególnie w kontekście czasu wykonania zadania i prędkości poruszania.

7.5. Przeprowadzone eksperymenty – założenia i analiza wyników

Informacje dotyczące wszystkich eksperymentów

W prowadzonych eksperymentach analizowano zarówno średnią, jak i medianę wyników, jednak jako istotniejszą dla tego badania traktowano medianę. Wynika to z charakterystyki rozkładu uczestników eksperymentów, który mógł być asymetryczny lub zawierać wartości odstające, co czyni medianę bardziej reprezentatywną miarą tendencji centralnej w tym przypadku (Manikandan 2011). W związku z tym, **zmianę procentową raportowano i obliczano w odniesieniu do mediany, co pozwalało na bardziej wiarygodną interpretację efektów eksperymentów.**

We wszystkich eksperymentach podział uczestników na grupy następował losowo i automatycznie, przy czym ze względów technicznych losowanie nie uwzględniało

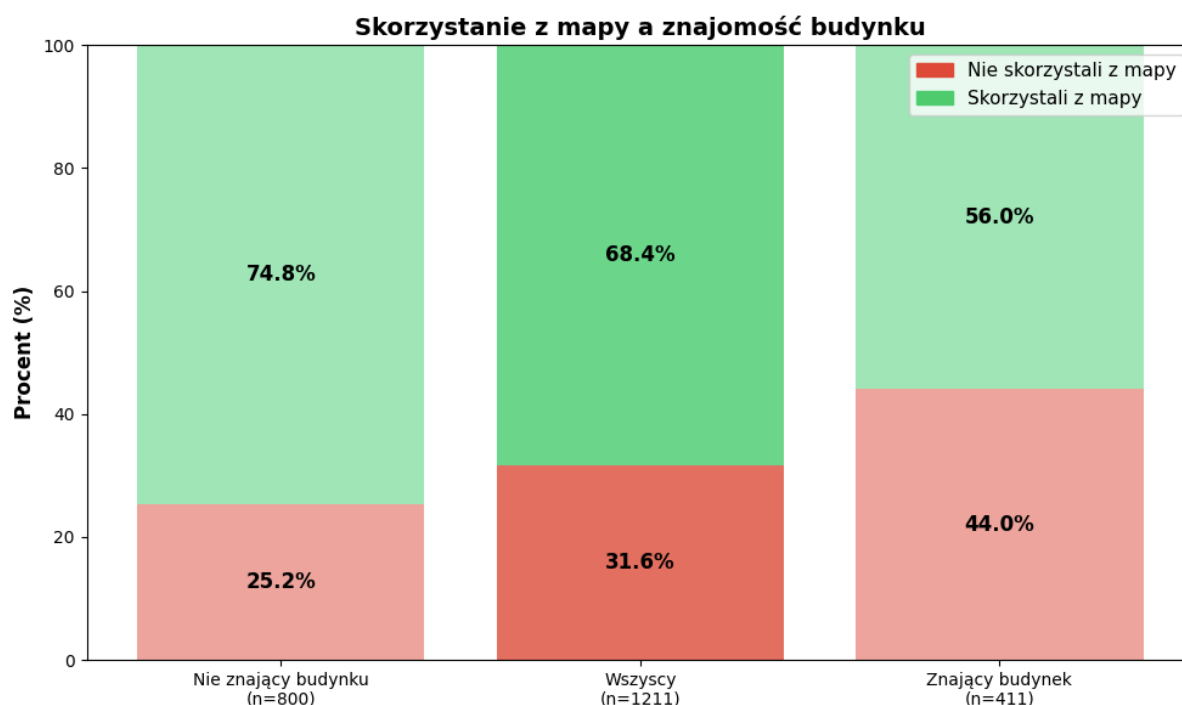
dotychczasowego rozkładu na grupy. W związku z tym **grupy badawcze mają zbliżoną, ale nie idealnie równą licznosc**. Dzięki stosunkowo dużej grupie badawczej nie ma to istotnego statystycznego wpływu na otrzymane wyniki.

Na początku trwania badania występował błąd w mechanizmie przekazywania danych. W jego wyniku dla uczestników, którzy realizowali badanie jako pierwsi (**około 20% wszystkich uczestników**) **nie zarejestrowano obserwacji „Czasu kontaktu z mapą”**. Na wykresach analizujących statystyki tego parametru należy więc oczekiwać mniejszej grupy badawczej N. Nie było możliwe powtórzenie całego eksperymentu ze względu na ograniczenia czasowe oraz trudność w pozyskaniu uczestników badania. Ze względu na postawiony cel badań uznano, że nie wpływa to na końcowe wnioski i nie jest kluczowe dla realizacji postawionych celów.

Ważną informacją wynikającą z badania i niezależną od poszczególnych scenariuszy jest fakt, iż średnio około 32% uczestników w ogóle nie wyświetliło mapy podczas poszukiwania celu. Nie mieli oni możliwości przekonać się jak wygląda mapa prezentująca wskazówki nawigacyjne. Ta obserwacja wnosi interesującą informację, iż wielu uczestników preferowało odmienną strategię nawigacyjną, polegającą na samodzielnej eksploracji budynku. Pojedyncze bezpośrednie obserwacje podczas przeprowadzania badań potwierdziły to zjawisko.

Głębsza analiza danych ujawnia, że procentowy udział osób niekorzystających z mapy był znacząco zależny od znajomości budynku przez uczestnika. Przeanalizowano liczbę zrealizowanych scenariuszy, przez wszystkich uczestników, ograniczoną tylko do scenariuszy, w których konieczne było, wyświetlenie mapy (naciśnięcie spacji) ($n=1211$) (n – liczba scenariuszy zrealizowanych przez wszystkich uczestników, ograniczona tylko do scenariuszy, w których konieczne było wyświetlenie mapy (naciśnięcie spacji) Rysunek 7.12). W grupie osób nieznających budynku (liczba zrealizowanych scenariuszy $n=800$) wskaźnik niekorzystających z mapy wyniósł 25% uczestników, natomiast w grupie osób znających budynek (liczba zrealizowanych scenariuszy $n=411$) odsetek osób niekorzystających z mapy wzrósł do 44%. Co więcej, osoby znające budynek, które zdecydowały się na użycie mapy, korzystały z niej rzadziej (średnio 2,5 razy badana osoba spojrzała na mapę budynku) w porównaniu z osobami nieznającymi budynku (średnio 3,5 razy badana osoba spojrzała na mapę budynku). Te wyniki sugerują, że znajomość przestrzeni znacząco redukuje potrzebę korzystania z pomocy nawigacyjnych, a uczestnicy częściowo polegają na wcześniej nabytej

wiedzy przestrzennej. Dane te potwierdzają dość oczywiste wnioski z badań nad sposobem korzystania z systemów nawigacji samochodowej, mianowicie im wyższy poziom znajomości przestrzeni, tym niższe wykorzystanie aplikacji nawigacyjnych (Topete i in. 2024).



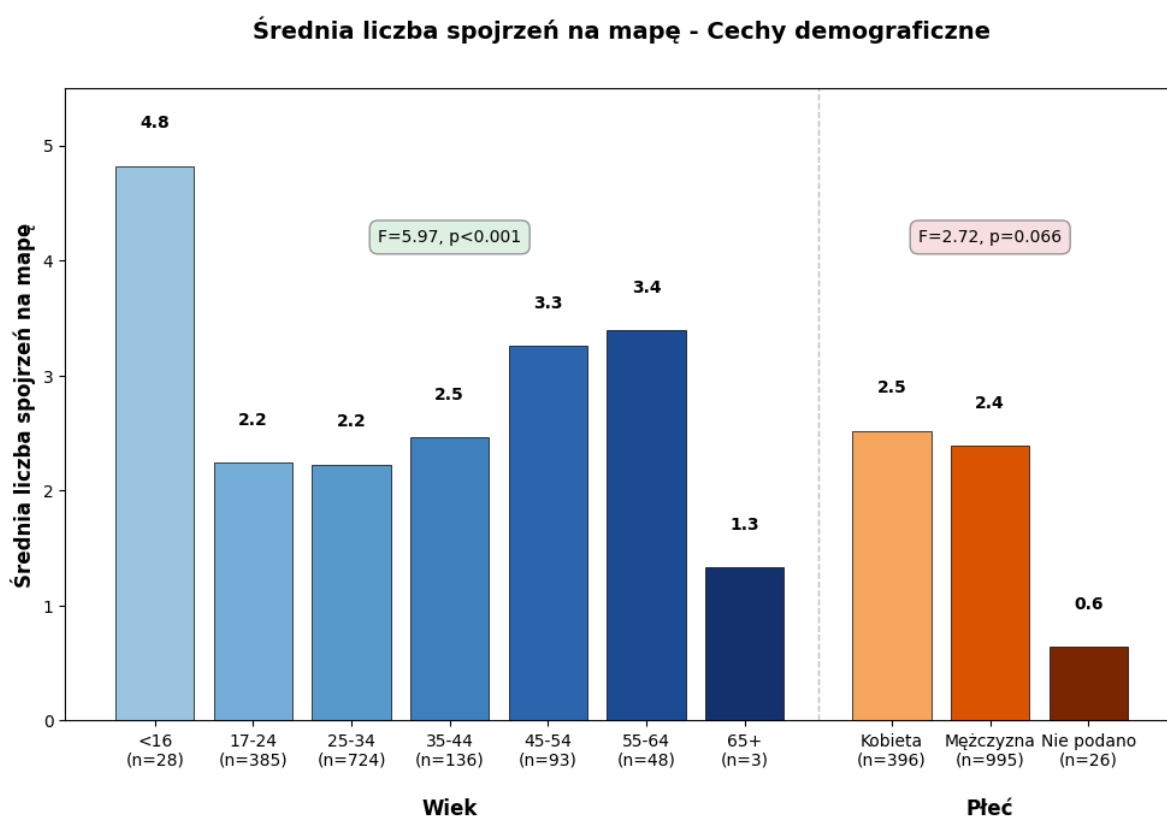
n – liczba scenariuszy zrealizowanych przez wszystkich uczestników, ograniczona tylko do scenariuszy, w których konieczne było wyświetlenie mapy (naciśnięcie spacji)

Rysunek 7.12. Korzystanie z widoku mapy przez uczestników badania – procent scenariuszy, w których uczestnicy skorzystali lub nie skorzystali z mapy chociaż raz, z porównaniem dla osób znających oraz nie znających danego budynku (opracowanie własne)

Zjawisko to można wyjaśnić w kontekście teorii strategii nawigacyjnych identyfikowanych w literaturze. Na przykład w artykule „Understanding Differences in Wayfinding Strategies” (Hegarty i in. 2023) przedstawiono dwie podstawowe strategie nawigacyjne identyfikowane zarówno u ludzi jak i u zwierząt:

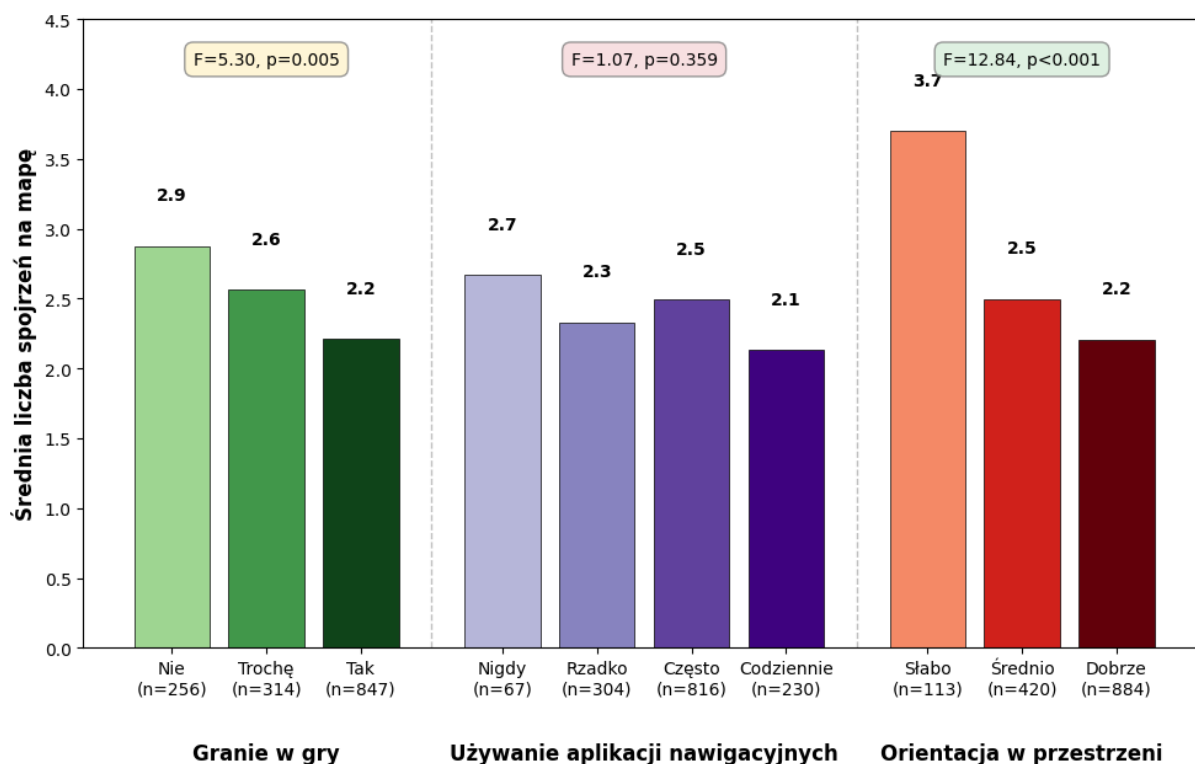
- Strategia Odpowiedzi (ang. *response strategy*) polega na przemieszczaniu się do celu za pomocą nawykowych, dobrze wyuczonych tras. Ta strategia opiera się na aktywacji prążkowiec (striatum) w mózgu i wykorzystuje zapamiętane sekwencje ruchów i punkty orientacyjne.
- Strategia Miejsca (ang. *place strategy*) polega na odkrywaniu nowych tras, takich jak skróty, na podstawie zdobytej wcześniej wiedzy przestrzennej o układzie środowiska. Ta strategia jest związana z aktywacją hipokampu i pozwala na elastyczne poruszanie się w przestrzeni dzięki poznawczej mapie otoczenia.

Na wybraną strategię wpływ mogą mieć m.in. czynniki takie jak płeć, wiek, poziom hormonów czy skłonność do ryzyka (Hegarty i in. 2023). Również w zebranych danych wykazano istotny statystycznie wpływ niektórych czynników personalnych na skłonność do korzystania z mapy (n – liczba scenariuszy zrealizowanych przez wszystkich uczestników, ograniczona tylko do scenariuszy, w których konieczne było wyświetlenie mapy (naciśnięcie spacji) Rysunek 7.13, n – liczba scenariuszy zrealizowanych przez wszystkich uczestników, ograniczona tylko do scenariuszy, w których konieczne było wyświetlenie mapy (naciśnięcie spacji) Rysunek 7.14). Najbardziej istotne statystycznie okazały się wiek oraz zadeklarowana zdolność orientacji w przestrzeni. Jeżeli chodzi o wiek uczestników to należy zwrócić uwagę, iż grupy „<16” oraz „65+” nie są wystarczająco liczne dla wyciągania wniosków z uzyskanych wartości średnich (powodem jest niska reprezentacja uczestników w tych grupach, a także trudności z ukończeniem całości badania).



n – liczba scenariuszy zrealizowanych przez wszystkich uczestników, ograniczona tylko do scenariuszy, w których konieczne było wyświetlenie mapy (naciśnięcie spacji)
Rysunek 7.13. Średnia liczba spojrzeń na mapę, a cechy demograficzne (opracowanie własne)

Średnia liczba spojrzeń na mapę - Doświadczenie i umiejętności



n – liczba scenariuszy zrealizowanych przez wszystkich uczestników, ograniczona tylko do scenariuszy, w których konieczne było wyświetlenie mapy (naciśnięcie spacji)

Rysunek 7.14. Średnia liczba spojrzeń na mapę, a doświadczenie i umiejętności (opracowanie własne)

W literaturze wyróżnia się także trzy przestrzenne style poznawcze (ang. *spatial cognitive styles*) oparte o: punkty orientacyjne (ang. *landmarks*), trasę (ang. *route*), przegląd (ang. *survey*) (Dang i in. 2023; Nori i Giusberti 2006). Style te odnoszą się do zdolności poznawczych (ang. *spatial ability*) ludzi i są kwestią preferencji oraz inteligencji przestrzennej (ang. *intelligence in spatial cognition*).

W kontekście niniejszego badania należy rozważyć również dodatkowe czynniki, które mogły zaburzać zachowanie uczestników. Po pierwsze, środowisko wirtualne mogło zachęcać do eksploracji – mimo sugestii presji czasowej na początku badania brak pośpiechu nie wiązał się z konsekwencjami, podobnie jak błędna nawigacja. Mogło to sprawić, że eksploracja była postrzegana jako ciekawsza lub bardziej angażująca niż użycie mapy i dotarcie prosto do celu. Po drugie, nawyki nawigacyjne wykształcone w rzeczywistych budynkach, gdzie mapy cyfrowe nie są powszechnie dostępne, mogły wpływać na zachowania uczestników. Po trzecie, niektórzy uczestnicy mogli nie w pełni zrozumieć możliwości skorzystania z mapy lub jej potencjalnej

użyteczności w kontekście postawionego przed nimi zadania. Wykluczenie tych czynników to potencjalne pole do kolejnych badań.

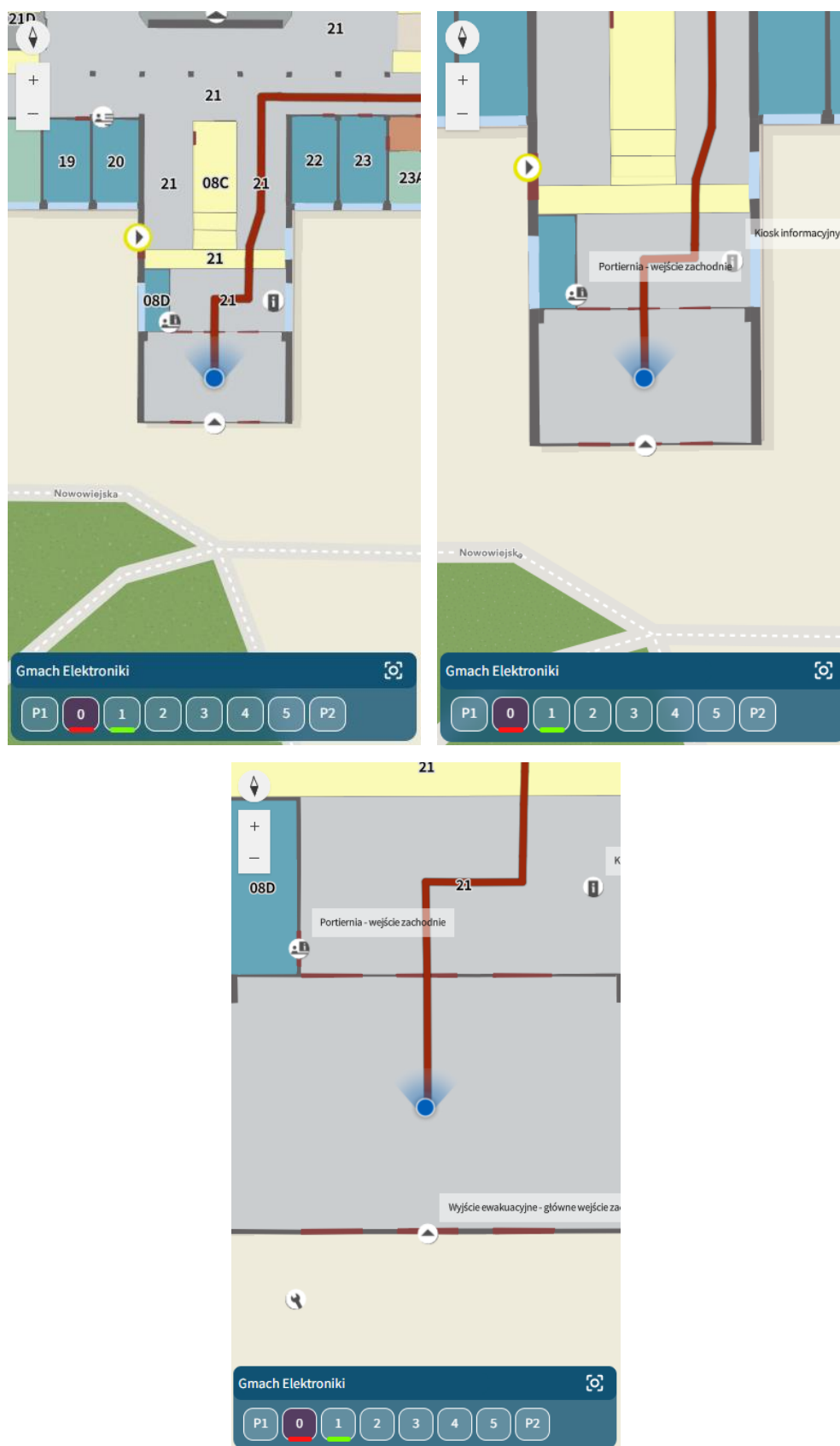
W dalej opisanych eksperymentach, których istotą było wyświetlenie mapy, nie brano pod uwagę uczestników, którzy nie spojrzeli na mapę ani razu (nie otworzyli mapy poprzez naciśnięcie klawisza spacji).

7.5.1. Eksperyment 1 – wpływ skali wyświetlanej mapy na efektywność nawigacji

Cele badawcze i założenia eksperymentu

Głównym celem pierwszego eksperymentu było zbadanie wpływu skali wyświetlanej mapy (w zasadzie poziomu przybliżenia) na efektywność nawigacji w środowisku wewnętrznym budynku. Uczestnicy w czasie całego badania mieli dostęp do mapy z zaznaczoną trasą, jednakże poszczególne grupy badawcze różniły się zakresem obszaru pokazywanego na ekranie urządzenia mobilnego. Istotą eksperymentu było określenie, w jakim stopniu zasięg kontekstu przestrzennego prezentowanego na mapie wpływa na zdolność użytkowników do efektywnego poruszania się po budynku oraz utrzymania orientacji przestrzennej.

Projekt eksperymentu zakładał podział uczestników na trzy grupy, z których każda korzystała z innego poziomu przybliżenia mapy (Rysunek 7.15). Pierwsza grupa widziała mapę z szerokim kontekstem przestrzennym obejmującym większy fragment budynku („Odległy zoom”), druga grupa miała dostęp do mapy o średnim poziomie przybliżenia pokazującej bezpośrednie otoczenie użytkownika wraz z fragmentem trasy („Średni zoom”), natomiast trzecia grupa korzystała z mapy silnie przybliżonej, koncentrującej się wyłącznie na najbliższym otoczeniu pozycji uczestnika („Bliski zoom”). Wszystkie grupy otrzymały identyczną trasę do przebycia, a różnice w poziomie przybliżenia stanowiły jedyną zmienną niezależną w eksperymencie.



Rysunek 7.15. Ilustracja trzech poziomów przybliżenia określanych od lewej jako: Odległy zoom, Średni zoom, Bliski zoom (opracowanie własne)

Sformułowana hipoteza zakładała, że przedstawienie szerszego kontekstu przestrzennego na mapie ułatwi uczestnikom orientację podczas poruszania się po budynku, co przełoży się na krótszy czas realizacji zadania oraz mniejszą liczbę błędów nawigacyjnych. Oczekiwano, że użytkownicy korzystający z mapy o większym zakresie będą w stanie lepiej planować swoją trasę, wcześniej identyfikować punkty orientacyjne oraz pewniej podejmować decyzje w punktach decyzyjnych. Spodziewano się również, że mapa z mniejszym kontekstem przestrzennym może utrudniać zrozumienie relacji między kolejnymi segmentami trasy, prowadząc do większej dezorientacji i częstszych wahań w miejscach wymagających zmiany kierunku. Informacje podstawowe o eksperymencie 1 przedstawiono w tabeli 9.

Tabela 9. Informacje podstawowe o eksperymencie 1.

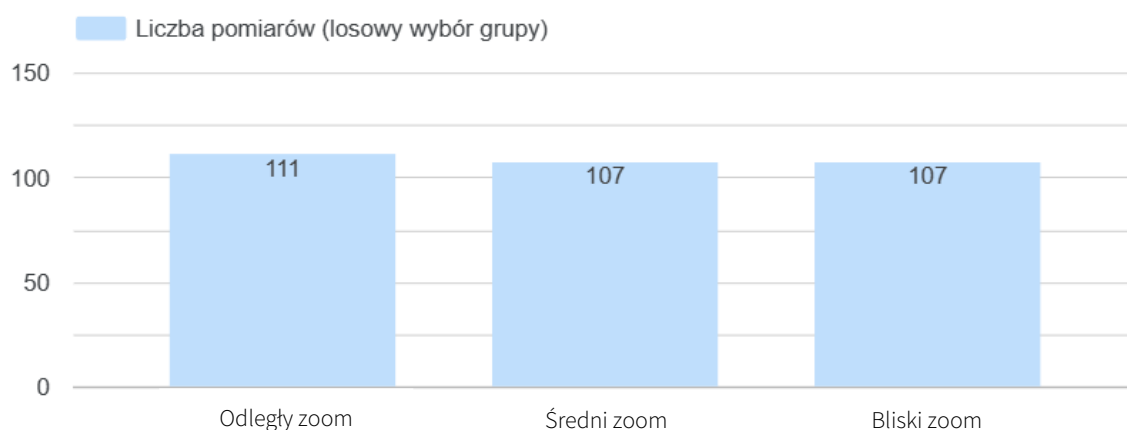
Eksperyment 1					
Kod	Budynek	Cel	Długość trasy wyznaczonej przez aplikację	Modyfikowane zmienne niezależne	Liczba uczestników
eiti1	Gmach Elektroniki	p. 128	120m	Poziom przybliżenia	166
eiti2	Gmach Elektroniki	p. 415	103m	Poziom przybliżenia	159

Eksperyment realizowany jest w wirtualnym modelu Gmachu Elektroniki PW. Punkt początkowy został wyznaczony w lokalizacji innej niż ta, w której uczestnicy zakończyli poprzedni test. Układ korytarzy i pomieszczeń w budynku umożliwia realizację zadania przy użyciu trasy z wykorzystaniem wind, jak i klatek schodowych. Poniżej przedstawiono trasę wyznaczoną przez aplikację nawigacyjną kolejno dla scenariusza *eiti1* i *eiti2* (Rysunek 7.16, Rysunek 7.17).



Przedstawienie i analiza wyników

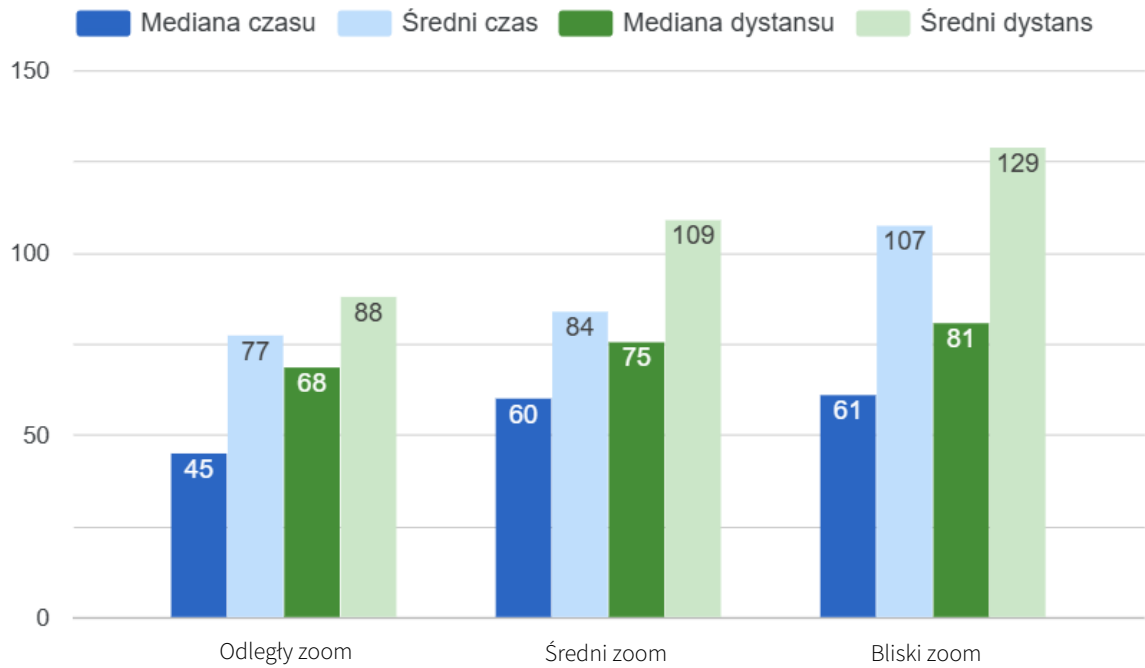
Do analizy wyników eksperymentu wzięto pod uwagę dane z **325 przejść** uczestników zebranych na dwóch scenariuszach *eiti1* oraz *eiti2*. Podział uczestników na trzy grupy eksperymentalne przedstawiono na rysunku poniżej (Rysunek 7.18).



Rysunek 7.18. Rozkład liczebności grup według wylosowanego poziomu przybliżenia mapy (opracowanie własne)

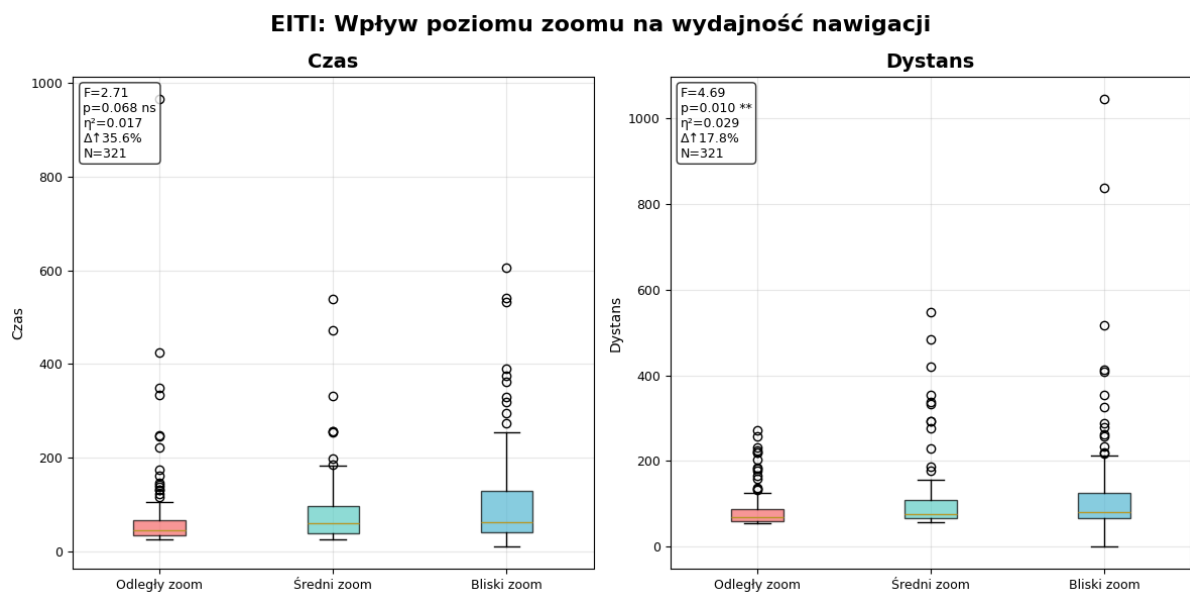
Poniższy wykres (Rysunek 7.19) prezentuje medianę oraz średnią, kolejno dla czasu przejścia (kolor niebieski) oraz przebytego dystansu (kolor zielony) przez uczestników w każdej z grup.

Można zaobserwować, iż zmiana poziomu przybliżenia mapy pomiędzy grupą „Odległy zoom”, a grupą „Bliski zoom” spowodowała podniesienie mediany czasu wykonania zadania o **36%**, a mediany dystansu o **18%**.



Rysunek 7.19. Czas oraz przebyty dystans w poszukiwaniu pomieszczenia w eksperymencie 1 (opracowanie własne)

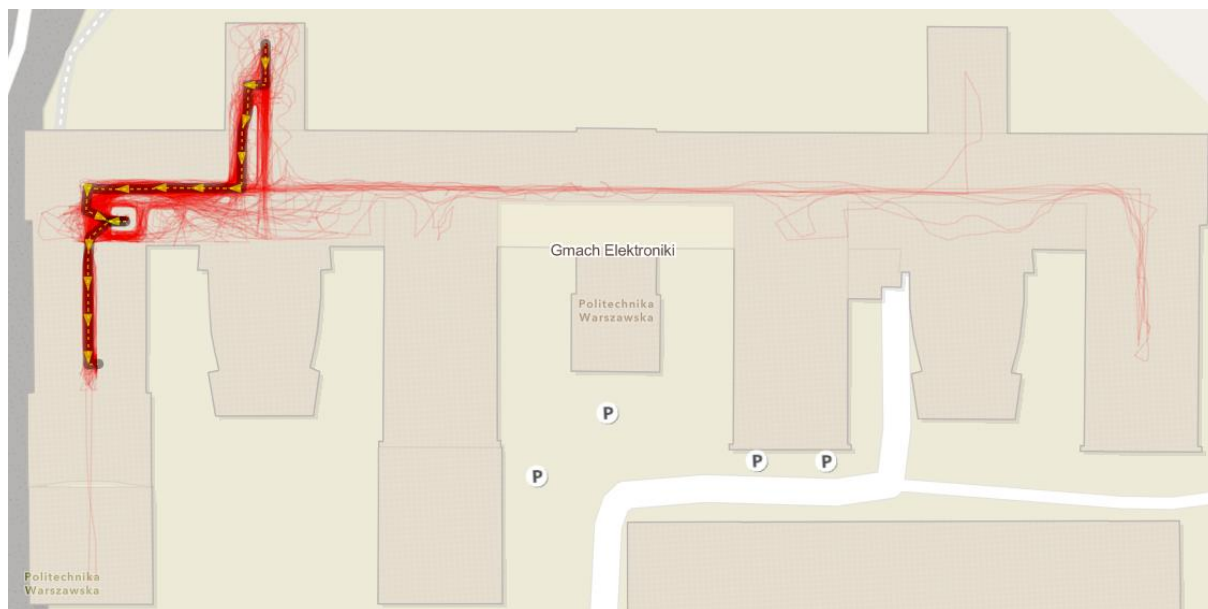
Analiza statystyczna (Rysunek 7.20) potwierdza te obserwacje oraz wskazuje na istotność statystyczną uzyskanego wyniku zwłaszcza w przypadku pomiarów przebytego dystansu ($p=0.01$).



Rysunek 7.20. Rozkład statystyczny wyników realizacji eksperymentu 1. Czas ukończenia po lewej, przebyty dystans po prawej (opracowanie własne)

Na uwagę zasługuje również większa liczba wartości odstających w grupie z bliskim oraz średnim poziomem zoom. Obserwacja ta może wskazywać na sytuacje, w których uczestnik

odsunął się od wyznaczonej trasy na tyle daleko, iż całkowicie zniknęła ona z jego pola widzenia, a następnie miał problem z jej ponownym odnalezieniem. W efekcie zdarzało się, iż zagubiony uczestnika błądził po całym budynku w poszukiwaniu odpowiedniego pomieszczenia, czego dowodzą pojedyncze zapisy długich tras (Rysunek 7.21).



Rysunek 7.21. Pojedyncze zapisy przebytych tras wskazują na błądzenie po całym budynku. Taką sytuację zaobserwowano tylko dla grupy z bliskim oraz średnim poziomem zoom (opracowanie własne)

Wnioski i podsumowanie eksperymentu

Wyniki przeprowadzonego eksperymentu potwierdziły przyjętą hipotezę badawczą. Szerszy kontekst przestrzenny prezentowany na mapie istotnie ułatwił uczestnikom nawigację w środowisku wewnętrznym budynku. Grupa korzystająca z odległego poziomu przybliżenia osiągnęła wyraźnie lepsze wyniki w porównaniu z grupą używającą bliskiego zoomu.

Istotnym odkryciem jest zwiększona liczba wartości odstających w grupach ze średnim i bliskim poziomem przybliżenia. Przypadki te dokumentują sytuacje całkowitej dezorientacji, w których uczestnicy tracili wyznaczoną trasę z pola widzenia i nie potrafili jej ponownie odnaleźć. Analiza zapisanych trajektorii wykazała, że w skrajnych przypadkach zagubieni uczestnicy błądzili po całym budynku, co bardzo wydłużało czas realizacji zadania i znacząco zwiększało przebyty dystans. Tego typu sytuacje nie wystąpiły w grupie korzystającej z odległego zoomu.

Rezultaty eksperymentu dostarczają istotnych wskazówek dla projektowania aplikacji nawigacyjnych w budynkach. Zbyt silne przybliżenie mapy (duża skala), choć może wydawać

się intuicyjne dla uproszczenia interfejsu, w praktyce utrudnia orientację i zwiększa ryzyko dezorientacji. System nawigacyjny powinien domyślnie prezentować mapę z wystarczająco szerokim kontekstem, aby użytkownik mógł widzieć nie tylko bezpośrednie otoczenie, ale również fragment trasy przed sobą oraz kluczowe punkty orientacyjne. Alternatywnie, system powinien umożliwiać dynamiczne dostosowywanie poziomu przybliżenia w zależności od kontekstu – na przykład automatyczne powiększanie/pomniejszenie widoku w miejscach wymagających szczególnej uwagi. Ponadto aplikacja powinna oferować ponowne wyznaczenie trasy w przypadku oddalenia się od tej pierwotnie wyznaczonej.

Ekspertyment przeprowadzono w tylko jednym budynku o określonej strukturze przestrzennej, co może ograniczać możliwość uogólniania wyników na inne typy obiektów. Przyszłe badania powinny uwzględnić różnorodność architektoniczną budynków. Interesującym kierunkiem byłoby także zbadanie interakcji między poziomem przybliżenia a innymi elementami interfejsu nawigacyjnego, takimi jak sposób wizualizacji trasy czy prezentowanie instrukcji tekstowych. Ponadto w dalszych badaniach należy uwzględnić większy zakres poziomów przybliżenia (w tym bardzo małą skalę), aby precyzyjnie wskazać poziom optymalny dla zadanych warunków.

7.5.2. Ekspertyment 2a – wpływ prezentacji wyznaczonej trasy lub aktualnej lokalizacji na efektywność nawigacji

Cele badawcze i założenia eksperymentu

Głównym celem drugiego eksperymentu było zbadanie indywidualnego oraz łącznego wpływu prezentacji wyznaczonej trasy i oznaczenia aktualnej lokalizacji użytkownika na efektywność nawigacji wewnątrz budynku. Ekspertyment miał na celu określenie, który z tych elementów wspomagających nawigację wywiera silniejszy wpływ na wydajność poruszania się po budynku, a także zweryfikowanie, czy ich jednoczesne zastosowanie generuje efekt synergii przekraczający sumę indywidualnych wpływów.

Projekt eksperymentu zakładał realizację trzech kolejnych scenariuszy nawigacyjnych w modelu Gmachu Chemii Politechniki Warszawskiej. Uczestnicy zostali losowo przydzieleni do czterech grup badawczych, a przynależność do grupy pozostawała stała przez wszystkie scenariusze.

- **Brak wsparcia** – grupa dysponowała podstawową wersją mapy bez wizualizacji trasy oraz bez oznaczenia aktualnej lokalizacji, stanowiąc grupę kontrolną.
- **Lokalizacja** – grupa korzystała z mapy z zaznaczoną aktualną lokalizacją użytkownika, co umożliwiło ciągłe monitorowanie własnej pozycji w przestrzeni budynku.
- **Trasa** – grupa otrzymała mapę z wizualizowaną trasą prowadzącą do celu, lecz bez oznaczenia aktualnej lokalizacji.
- **Pełne wsparcie** – grupa miała dostęp do pełnej funkcjonalności mapy obejmującej zarówno wizualizację trasy, jak i ciągłe prezentowanie aktualnej lokalizacji.

Wszystkie grupy otrzymały identyczne zadania nawigacyjne polegające na dotarciu do tych samych pomieszczeń w kolejnych scenariuszach.



Rysunek 7.22. Ilustracja czterech trybów mapy, które były do dyspozycji poszczególnych grup. Kolejno: Brak wsparcia, Lokalizacja, Trasa, Pełne wsparcie (opracowanie własne)

Sformułowana hipoteza zakładała, że zarówno wizualizacja trasy, jak i oznaczenie aktualnej lokalizacji przyniosą istotne korzyści w efektywności nawigacji w porównaniu z grupą kontrolną. Oczekiwano, że wizualizacja trasy będzie miała silniejszy wpływ na skrócenie czasu realizacji zadania poprzez eliminację potrzeby planowania drogi, podczas gdy oznaczenie lokalizacji w większym stopniu przyczyni się do redukcji błędów nawigacyjnych i utrzymania orientacji przestrzennej. Spodziewano się również wystąpienia efektu synergii w grupie czwartej, gdzie jednoczesne zastosowanie obu form wsparcia nawigacyjnego przyniesie wyniki przekraczające sumę indywidualnych korzyści. Informacje podstawowe o eksperymencie 2a przedstawiono w tabeli 10.

Tabela 10. Informacje podstawowe o eksperymencie 2a.

Eksperyment 2a					
Kod	Budynek	Cel	Długość trasy wyznaczonej przez aplikację	Modyfikowane zmiennie niezależne	Liczba uczestników
chem1	Gmach Chemii	p. 159	88m	Wyświetlanie trasy Wyświetlanie pozycji	151
chem2	Gmach Chemii	p. 404	69m	Wyświetlanie trasy Wyświetlanie pozycji	139
chem3	Gmach Chemii	Główne wejście	155m	Wyświetlanie trasy Wyświetlanie pozycji	135

Realizacja wszystkich trzech scenariuszy następuje jeden po drugim, a punkt początkowy każdego kolejnego scenariusza wyznaczony jest w punkcie zakończenia poprzedniego. Dzięki temu przejście przez budynek zachowuje ciągłość i promuje uzyskanie ogólnej orientacji w układzie pięter i korytarzy. Poniżej przedstawiono trasę wyznaczoną przez aplikację nawigacyjną kolejno dla scenariusza *chem1*, *chem2* oraz *chem3* (Rysunek 7.23, Rysunek 7.24, Rysunek 7.25).



Rysunek 7.23. Trasa wyznaczona przez aplikację dla scenariusza chem1, linia ciągła pokazuje trasę na piętrze 1, linia przerywana pokazuje trasę na piętrze 0 (opracowanie własne)



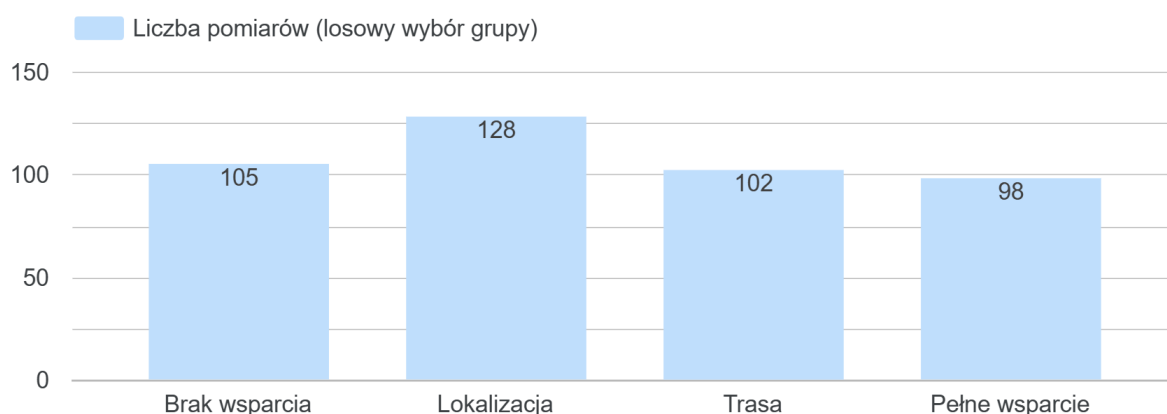
Rysunek 7.24. Trasa wyznaczona przez aplikację dla scenariusza chem2, linia ciągła pokazuje trasę na piętrze 1, linia przerywana pokazuje trasę na piętrze 4 (opracowanie własne)



Rysunek 7.25. Trasa wyznaczona przez aplikację dla scenariusza chem3, linia ciągła pokazuje trasę na piętrze 4, linia przerywana pokazuje trasę na piętrze 0 (opracowanie własne)

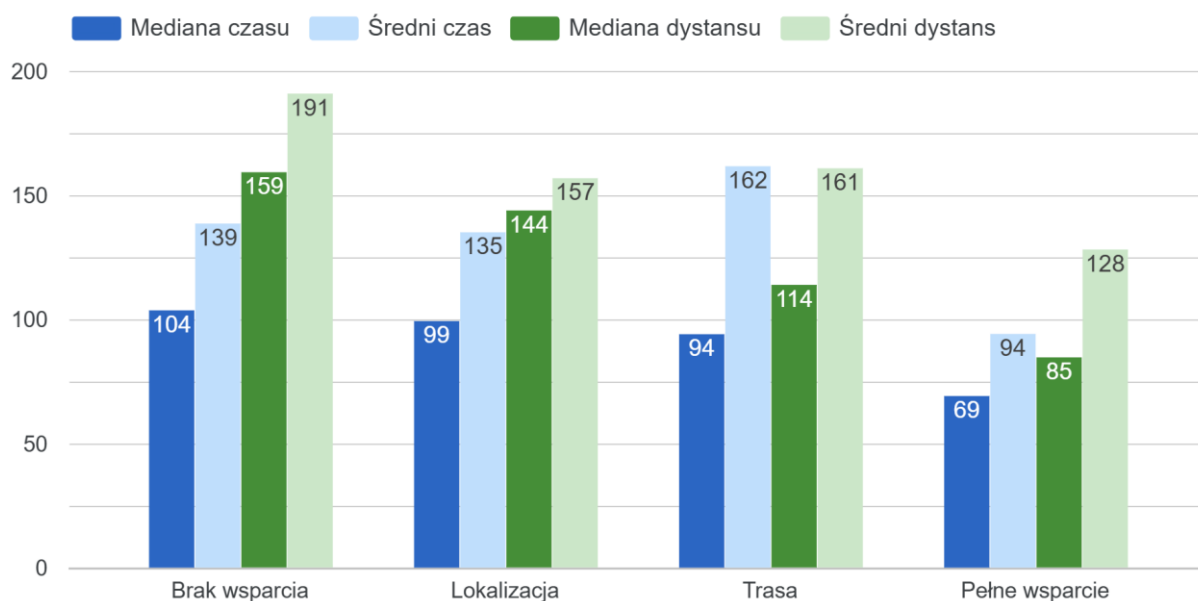
Przedstawienie i analiza wyników

W eksperymencie 2a przeanalizowano statystyki **433** przejść uczestników zebranych w trzech scenariuszach *chem1*, *chem2* oraz *chem3*. Podział uczestników na cztery grupy badawcze przedstawiono na rysunku poniżej (Rysunek 7.26).



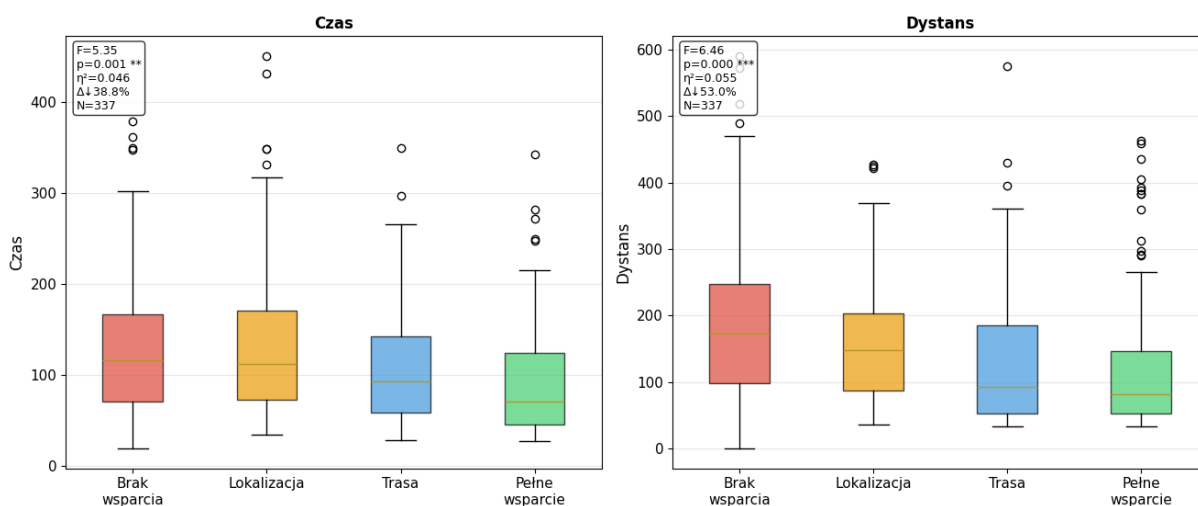
Rysunek 7.26. Rozkład liczebności grup według wylosowanego trybu (opracowanie własne)

Poniższy wykres (Rysunek 7.27) prezentuje medianę oraz średnią, kolejno dla czasu przejścia (kolor niebieski) oraz przebytego dystansu (kolor zielony) przez uczestników w każdej z grup.



Rysunek 7.27. Czas oraz przebyty dystans w poszukiwaniu pomieszczenia w eksperymencie 2a (opracowanie własne)

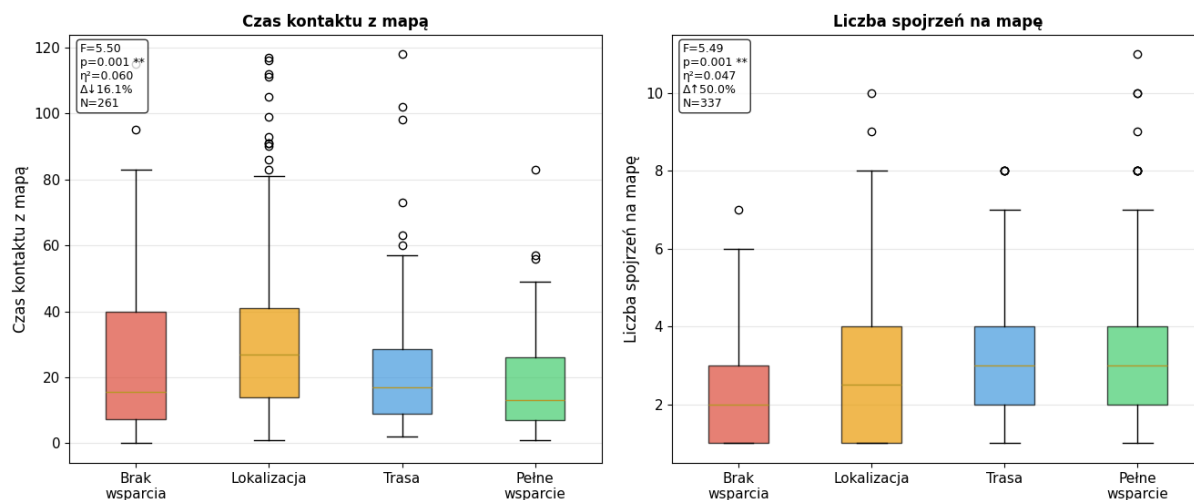
Zarówno mierzony czas oraz przebyty dystans w poszukiwaniu pomieszczenia wykazały statystycznie istotną ($p < 0.001$) korelację ze wsparciem dostępnym na mapie. Wyświetlenie trasy okazało się wyraźnie bardziej pomocne od samego prezentowania lokalizacji. W trybie z pełnym wsparciem mediana obniżyła się względem braku wsparcia kolejno o **39%** dla pomiaru czasu oraz **53%** dla pomiaru dystansu.



Rysunek 7.28. Rozkład statystyczny wyników realizacji eksperymentu 2a. Czas ukończenia po lewej, przebyty dystans po prawej (opracowanie własne)

Pomiar czas kontaktu z mapą oraz liczby spojrzeń na mapę potwierdzają większą istotność prezentacji trasy na mapie ponad lokalizację (Rysunek 7.29). Ponadto wraz z poprawieniem

wyników (skróceniem czasu wykonania zadania oraz przebytego dystansu) obserwowany jest również krótszy czas kontaktu z mapą.



Rysunek 7.29. Rozkład statystyczny statystyk interakcji z mapą podczas eksperymentu 2a. Czas kontaktu z mapą po lewej, liczba spojrzeń na mapę po prawej (opracowanie własne)

Wnioski i podsumowanie eksperymentu

Eksperyment 2a dostarczył istotnych dowodów na temat skuteczności poszczególnych elementów wspomagających nawigację wewnątrz budynków oraz ich wzajemnych relacji. Wyniki badania pozwalają na sformułowanie kilku kluczowych wniosków.

Prezentacja wyznaczonej trasy okazała się zdecydowanie bardziej istotnym czynnikiem wpływającym na efektywność nawigacji niż samo oznaczenie aktualnej lokalizacji użytkownika. Różnica ta manifestowała się zarówno w skróceniu czasu realizacji zadania, jak i w redukcji przebytego dystansu. Wizualizacja trasy eliminuje konieczność samodzielnego planowania drogi przez użytkownika, co znacząco redukuje obciążenie poznawcze i minimalizuje ryzyko błędnych decyzji nawigacyjnych. Ponadto wyświetlana trasa zawęża obszar zainteresowania osoby poruszającej się po budynku do bliskich okolic trasy – dzięki temu orientacja na mapie oraz w przestrzeni jest znacznie szybsza i mniej obciążająca.

Oznaczenie aktualnej lokalizacji, choć przynosiło wymierne korzyści w porównaniu z grupą kontrolną, nie generowało tak widocznej poprawy efektywności. Funkcja ta wymaga od użytkownika ciągłego odnoszenia własnej pozycji na mapie na rzeczywiste otoczenie oraz samodzielnego wyznaczania kolejnych punktów orientacyjnych na trasie do celu.

Jednoczesne zastosowanie obu form wsparcia nawigacyjnego w grupie z pełnym wsparciem przyniosło najlepsze rezultaty, przekraczające korzyści wynikające z zastosowania każdego elementu osobno. Mediana czasu realizacji zadania względem grupy kontrolnej skróciła się o 9% w przypadku wsparcia w postaci wyświetlania lokalizacji, 17% w przypadku wyświetlania trasy na mapie oraz aż 39% przy dostępności obu form wsparcia. Efekt ten potwierdza hipotezę o synergii między wizualizacją trasy a oznaczeniem lokalizacji - elementy te wzajemnie się uzupełniają.

Połączenie wizualizacji trasy z ciągłym monitorowaniem własnej pozycji umożliwia użytkownikom bieżącą weryfikację postępów i szybką identyfikację ewentualnych odchyłeń od planowanej trasy. Zwiększa to pewność podejmowanych decyzji nawigacyjnych.

Analiza czasu kontaktu z mapą oraz liczby spojrzeń dostarczyła dodatkowych dowodów na skuteczność wizualizacji trasy. Paradoksalnie, grupy korzystające z bardziej zaawansowanych form wsparcia wykazywały krótszy czas kontaktu z mapą, co wskazuje na większą efektywność przetwarzania informacji. Użytkownicy potrzebowali mniej czasu na zrozumienie sytuacji przestrzennej i podjęcie decyzji nawigacyjnych.

Wyniki te sugerują, że dobrze zaprojektowane wsparcie nawigacyjne nie zwiększa zależności od urządzenia, lecz optymalizuje proces podejmowania decyzji. Użytkownicy mogli szybciej uzyskać potrzebne informacje i skupić uwagę na rzeczywistym otoczeniu, co przekładało się na płynniejszą nawigację.

Wyniki eksperymentu wskazują, że systemy nawigacji wewnątrz budynków powinny priorytetowo implementować wizualizację wyznaczonej trasy. Funkcja ta przynosi największe korzyści przy stosunkowo niewielkim zwiększeniu złożoności interfejsu. Oznaczenie aktualnej lokalizacji, choć mniej krytyczne, stanowi istotne uzupełnienie, szczególnie w złożonych strukturach architektonicznych z wieloma podobnymi korytarzami i rozgałęzieniami.

7.5.3. Eksperyment 2b – wpływ prezentacji wyznaczonej trasy lub aktualnej lokalizacji na zapamiętywanie przebytej drogi

Cele badawcze i założenia eksperymentu

Eksperyment 2b bezpośrednio nawiązywał do eksperymentu 2a i miał na celu zbadanie wpływu różnych form wsparcia nawigacyjnego na zapamiętywanie przebytej trasy. Głównym zamierzeniem było określenie, czy typ pomocy nawigacyjnej udostępnionej uczestnikom podczas pierwszego przejścia trasy wpływa na efektywność jej ponownego przebycia po przerwie – bez dostępu do mapy.

Projekt eksperymentu zakładał realizację scenariusza w identycznych warunkach przestrzennych jak początkowy scenariusz *chem1* z eksperymentu 2a, z kluczową modyfikacją polegającą na całkowitym pozbawieniu uczestników dostępu do mapy. Scenariusz był przeprowadzany na samym końcu badania, po przerwie i realizacji pozostałych eksperymentów, co zapewniało odstęp czasowy od pierwszego przejścia trasy. Uczestnicy musieli polegać wyłącznie na swoim doświadczeniu i pamięci przestrzennej nabytej podczas początkowego poszukiwania pomieszczenia w Gmachu Chemii. Podział na cztery grupy badawcze został zachowany z eksperymentu 2a, co umożliwiło porównanie efektywności zapamiętywania trasy w zależności od typu wcześniej otrzymanego wsparcia nawigacyjnego.

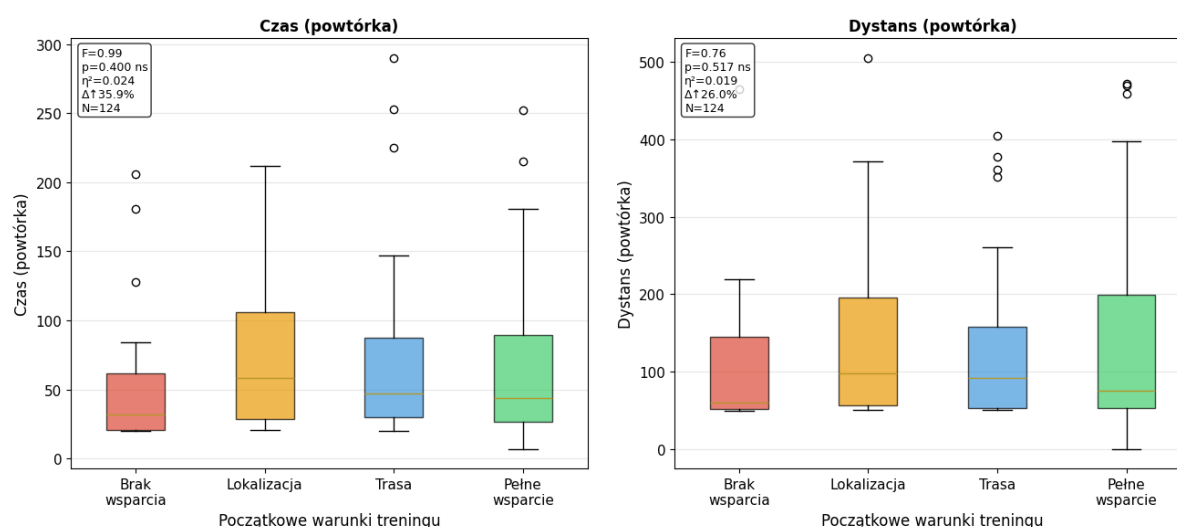
Sformułowana hipoteza zakładała, że forma wsparcia nawigacyjnego zastosowana w eksperymentach 2a będzie miała istotny wpływ na jakość zapamiętania trasy. Oczekiwano, że uczestnicy korzystający wcześniej z mapy zawierającej wyłącznie oznaczenie aktualnej lokalizacji wykażą lepsze wyniki w ponownej nawigacji niż ci, którzy mieli dostęp do wizualizacji trasy, gdyż konieczność aktywnego planowania drogi sprzyja głębszemu przetwarzaniu informacji przestrzennej. Spodziewano się również, że grupa kontrolna, która podczas pierwszego przejścia nie dysponowała żadnym wsparciem nawigacyjnym, może paradoksalnie osiągnąć konkurencyjne wyniki ze względu na większy wysiłek poznawczy włożony w eksplorację i zapamiętywanie środowiska. Informacje podstawowe o eksperymentach 2b przedstawiono w tabeli 11.

Tabela 11. Informacje podstawowe o eksperymencie 2b.

Eksperyment 2b					
Kod	Budynek	Cel	Długość trasy wyznaczonej przez aplikację	Modyfikowane zmiennie niezależne	Liczba uczestników
chem1- powtórka	Gmach Chemii	p. 159	-	-	124

Przedstawienie i analiza wyników

Poniższe wykresy pudełkowe (Rysunek 7.30) ilustrują wyniki eksperymentu 2b, prezentując dwie kluczowe zmienne: czas realizacji zadania oraz dystans przebytej drogi w powtórnym scenariuszu nawigacyjnym.



Rysunek 7.30. Rozkład statystyczny wyników realizacji eksperymentu 2b w relacji do trybu wsparcia nawigacyjnego otrzymanego w eksperymencie 2a. Czas ukończenia po lewej, przebyty dystans po prawej (opracowanie własne)

Analiza czasu oraz dystansu powtórznego przejścia trasy nie wykazała istotnych statystycznych różnic między grupami ($p = 0.400$ oraz $p = 0.517$), co sugeruje brak wpływu rodzaju wcześniejszego wsparcia nawigacyjnego na szybkość ponownej realizacji zadania.

Warto zauważyć, że grupa bez wsparcia charakteryzowała się najmniejszą zmiennością wyników oraz najniższą wartością mediany. We wszystkich grupach zaobserwowano wartości

odstające, co wskazuje na przypadki znacznej dezorientacji przestrzennej niektórych uczestników.

Wnioski i podsumowanie eksperymentu

Wyniki eksperymentu 2b prowadzą do zaskakujących i istotnych poznawczo wniosków dotyczących wpływu wsparcia nawigacyjnego na zapamiętywanie przebytej trasy.

Potwierdził się spodziewany fakt, iż grupa kontrolna nieposiadająca wcześniej żadnego wsparcia nawigacyjnego osiągnęła średnio najkrótsze czasy realizacji zadania oraz najmniejszą zmienność wyników. Zjawisko to można interpretować w kontekście teorii głębokości przetwarzania informacji (Craik i Lockhart 1972; Craik i Tulving 1975) – uczestnicy zmuszeni do aktywnej eksploracji przestrzeni bez pomocy mapy musieli włożyć większy wysiłek poznawczy w zapamiętanie trasy, co prawdopodobnie prowadziło do tworzenia bogatszych i trwalszych reprezentacji mentalnych układu budynku.

Analiza statystyczna nie wykazała istotnych różnic między grupami zarówno pod względem czasu realizacji zadania, jak i przebytego dystansu, co uniemożliwia jednoznaczne potwierdzenie postawionej hipotezy badawczej. Rezultat ten sugeruje, że rodzaj wsparcia nawigacyjnego udostępnionego podczas pierwszego przejścia trasy nie przekłada się w przewidywalny sposób na efektywność jej ponownego przebycia z pamięci.

Wyniki eksperymentu wskazują na istotny dylemat w projektowaniu systemów wsparcia nawigacyjnego dla środowisk wewnętrznych. Chociaż mapy z wizualizacją trasy zwiększają efektywność nawigacji w momencie ich używania (jak wykazał eksperyment 2a), mogą one jednocześnie hamować rozwój umiejętności orientacji przestrzennej i zapamiętywania tras. Zjawisko to ma szczególne znaczenie w kontekście wielokrotnego odwiedzania tych samych lokalizacji – użytkownicy stale polegający na mapach mogą nie wykształcić trwałej wiedzy przestrzennej o budynku. W związku z tym jest to istotny potencjalny kierunek dla dalszych badań, których celem byłoby po pierwsze potwierdzenie istotności wpływu wsparcia nawigacyjnego na zapamiętywanie trasy, a po drugie poszukiwanie metod przekazu informacji przestrzennej, które minimalizowałyby to zjawisko.

Obecność licznych wartości odstających we wszystkich grupach badawczych wskazuje na znaczące indywidualne różnice w zdolnościach przestrzennych uczestników. Niektórzy

uczestnicy, niezależnie od typu otrzymanego wsparcia, mieli znaczną trudność w zapamiętaniu i odtworzeniu trasy, co podkreśla konieczność uwzględnienia zróżnicowanych kompetencji użytkowników przy projektowaniu uniwersalnych systemów nawigacyjnych.

7.5.4. Eksperyment 3 – wpływ sposobu prezentacji kierunku patrzenia użytkownika na efektywność nawigacji

Cele badawcze i założenia eksperymentu

Głównym celem eksperymentu było zbadanie wpływu sposobu prezentacji kierunku, w którym patrzy uczestnik, na efektywność nawigacji wewnątrz budynku. Uczestnicy podczas całego badania mieli dostęp do mapy na żądanie (po naciśnięciu spacji). Na mapie zawsze zaznaczona była trasa oraz aktualna lokalizacja, jednakże poszczególne grupy badawcze różniły się metodą wizualizacji obrotu uczestnika (kierunku patrzenia). Istotą eksperymentu było określenie, w jakim stopniu sposób reprezentacji kierunku patrzenia wpływa na zdolność użytkowników do efektywnego poruszania się po budynku oraz na sposób interakcji z mapą.

Projekt eksperymentu zakładał podział uczestników na trzy grupy, z których każda korzystała z innego trybu prezentowania rotacji (Rysunek 7.31). Pierwsza grupa widziała mapę bez oznaczenia kierunku patrzenia uczestnika (Brak rotacji), druga grupa miała dostęp do mapy z niebieską kropką oznaczającą lokalizację uczestnika, przy czym kropka wskazywała dodatkowo kierunek patrzenia (rotacja niebieskiej kropki), natomiast trzecia grupa korzystała z mapy, która rotowała się dynamicznie, aby niebieski punkt reprezentujący użytkownika zawsze był skierowany ku górze ekranu (rotacja mapy).



Rysunek 7.31. Ilustracja trzech trybów prezentacji rotacji określanych od lewej jako: Brak rotacji, Rotacja niebieskiej kropki, Rotacja mapy (opracowanie własne)

Wszystkie grupy otrzymały identyczną trasę do przebycia, prowadzącą przez aulę główną Politechniki Warszawskiej oraz wymagającą skorzystania z windy lub schodów w celu dotarcia na drugie piętro. Różnice w sposobie prezentacji kierunku patrzenia stanowiły jedyną zmienną niezależną w eksperymencie.

Sformułowana hipoteza zakłada, że dynamiczna rotacja mapy ułatwi uczestnikom utrzymanie orientacji przestrzennej podczas poruszania się po budynku, co przełoży się na krótszy czas realizacji zadania, mniejszą liczbę błędów nawigacyjnych oraz krótszy czas potrzebny na zapoznanie się z mapą. Oczekiwany jest, że uczestnicy korzystający z mapy rotującej będą w stanie szybciej podejmować decyzje w punktach decyzyjnych, ponieważ reprezentacja kierunku na mapie będzie bezpośrednio odpowiadać ich perspektywie. Spodziewano się również, że mapa bez oznaczenia kierunku patrzenia może utrudniać przyswojenie informacji kartograficznej oraz orientację w przestrzeni. Z kolei rozwiązanie pośrednie, prezentujące kierunek patrzenia za pomocą strzałki na statycznej mapie, może stanowić kompromis między obciążeniem poznawczym a intuicyjnością nawigacji. Informacje podstawowe o eksperymencie 3 przedstawiono w tabeli 12.

Tabela 12. Informacje podstawowe o eksperymencie 3.

Eksperyment 3					
Kod	Budynek	Cel	Długość trasy wyznaczonej przez aplikację	Modyfikowane zmienne niezależne	Liczba uczestników
gg1	Gmach Główny	p. 232	154 metry	Tryb prezentacji kierunku ruchu	135

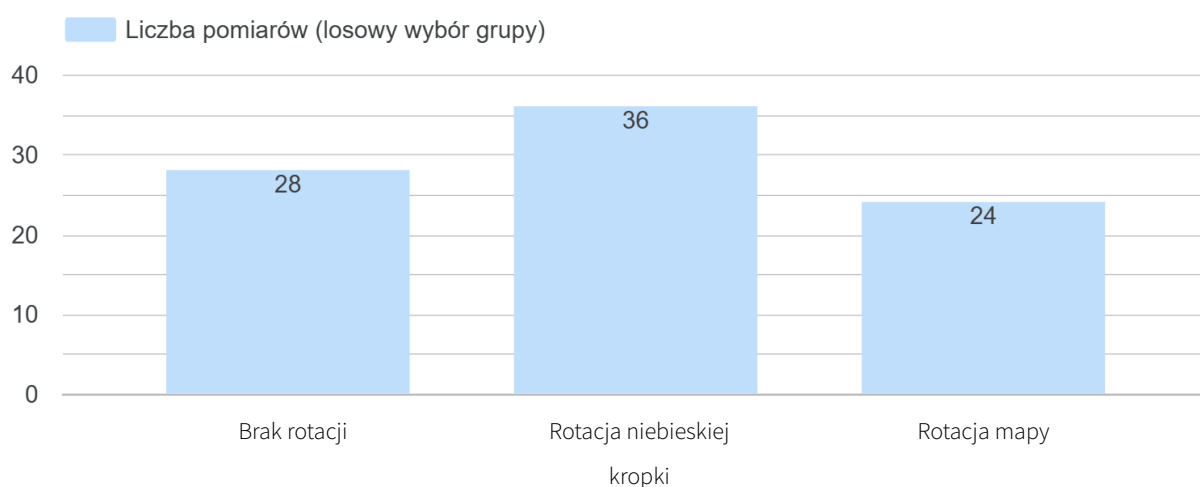
Eksperyment bazuje na modelu Gmachu Głównego PW. Punkt początkowy został wyznaczony przy głównym wejściu do gmachu, a scenariusz prezentuje standardową sytuację, w której znajdują się goście odwiedzający ten budynek po raz pierwszy. Układ korytarzy i pomieszczeń w budynku umożliwia realizację zadania przy użyciu różnych tras, w tym zarówno z wykorzystaniem wind, jak i klatek schodowych. Poniżej przedstawiono trasę wyznaczoną przez aplikację nawigacyjną:



Rysunek 7.32. Trasa wyznaczona przez aplikację dla eksperymentu 3, linia ciągła pokazuje trasę na piętrze 0, linia przerywana pokazuje trasę na piętrze 2 (opracowanie własne)

Przedstawienie i analiza wyników

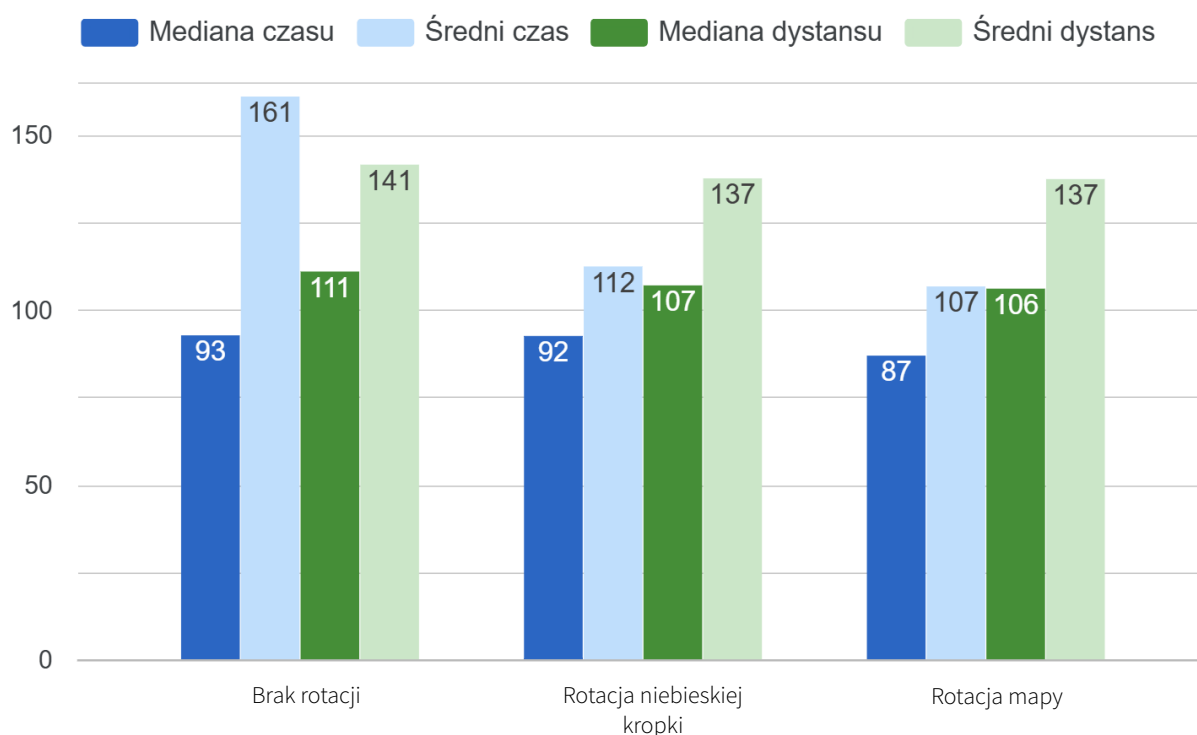
Do analizy wyników eksperymentu wzięto pod uwagę dane z **88 przejść** (47 odrzucono z uwagi na brak spojrzeń na mapę) uczestników zebranych w scenariuszu *gg1*. Rozkład uczestników na trzy grupy eksperymentalne przedstawiono na rysunku poniżej (Rysunek 7.33).



Rysunek 7.33. Rozkład liczebności grup według wylosowanego trybu prezentacji rotacji (opracowanie własne)

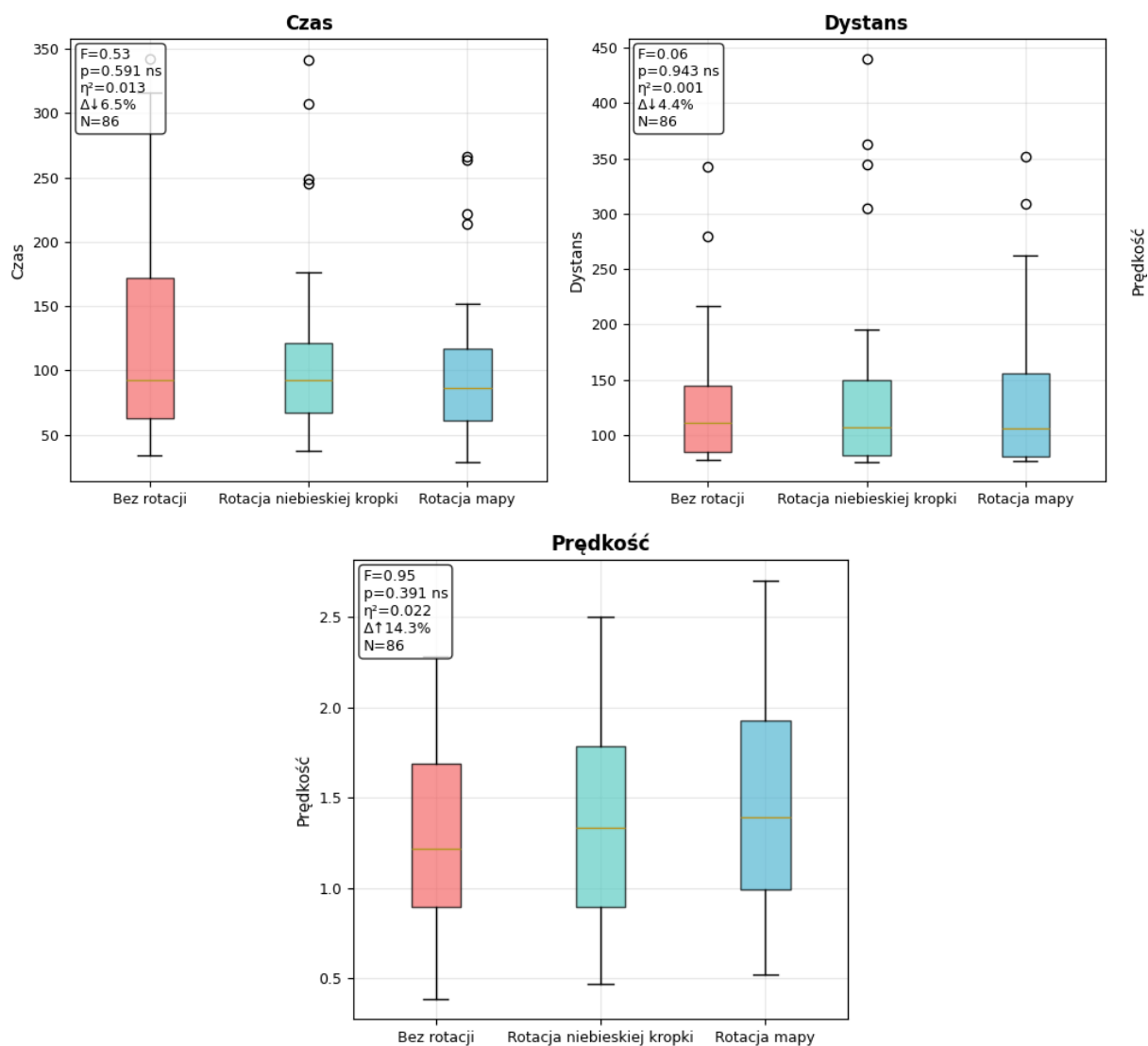
Poniższy wykres (Rysunek 7.34) prezentuje medianę oraz średnią, kolejno dla czasu przejścia (kolor niebieski) oraz przebitego dystansu (kolor zielony) przez uczestników w każdej z grup.

Można zaobserwować, iż zmiana trybu pomiędzy grupą „Brak rotacji”, a grupą „Rotacja mapy” spowodowała spadek mediany czasu wykonania zadania o **6%**, a mediany dystansu o **4%**. Zmiany te są relatywnie niewielkie w porównaniu do pozostałych eksperymentów.



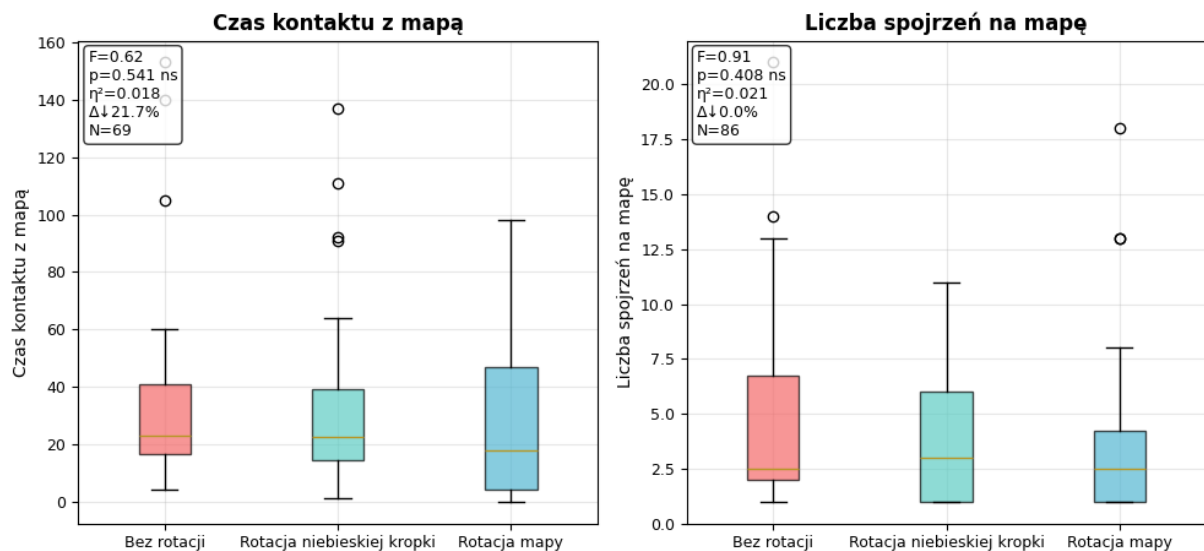
Rysunek 7.34. Czas oraz przebyty dystans w poszukiwaniu pomieszczenia w eksperymencie 3 (opracowanie własne)

Analiza statystyczna (Rysunek 7.35) potwierdza te tendencje oraz wskazuje na wzrost średniej prędkości poruszania o około 14%. Niestety zebrane pomiary nie mogą być uznane za istotne statystycznie (dla czasu $p=0.59$, dla odległości $p=0.94$, dla prędkości $p=0.39$), więc potwierdzenie wpływu prezentowania rotacji na wydajność nawigacji wymaga przeprowadzenia dodatkowych badań w przyszłości.



Rysunek 7.35. Rozkład statystyczny wyników realizacji eksperymentu 3. Czas ukończenia po lewej, przebyty dystans po prawej, średnia prędkość na dole (opracowanie własne).

Podobnie statystyki obserwacji interakcji z mapą (Rysunek 7.36) nie wykazały statystycznie istotnych obserwacji, choć na uwagę zasługuje mediana liczby spojrzeń na mapę, która nie zmieniła się pomiędzy trybami rotacji, natomiast mediana czasu kontaktu z mapą zmalała o ponad 21%.



Rysunek 7.36. Rozkład statystyczny statystyk interakcji z mapą podczas eksperymentu 3. Czas kontaktu z mapą po lewej, liczba spojrzeń na mapę po prawej (opracowanie własne).

Wnioski i podsumowanie eksperymentu

Eksperyment wykazał obiecujące tendencje, ale nie zostały one potwierdzone dużą istotnością statystyczną. Efekt jest więc niewielki i może wynikać z przypadku. Przyczyną problemu może być zbyt niska liczebność grup badawczych, co ograniczyło moc statystyczną przeprowadzonych testów. Dodatkowo, względna prostota wybranej trasy mogła zmniejszyć różnice między grupami, gdyż nawet przy mniej intuicyjnych sposobach prezentacji kierunku uczestnicy byli w stanie skutecznie dotrzeć do celu. Dalsze badania z większą próbą oraz bardziej złożonymi scenariuszami trasowymi są niezbędne do jednoznacznego potwierdzenia zaobserwowanych tendencji.

7.5.5. Eksperyment 4 – wpływ prezentacji wyznaczonej trasy na efektywność nawigacji

Cele badawcze i założenia eksperymentu

Zamierzeniem niniejszego scenariusza jest empiryczna weryfikacja, czy wyświetlenie uczestnikom wyznaczonej przez aplikację trasy na mapie istotnie wpływa na efektywność realizacji zadania nawigacyjnego. Analiza koncentruje się na pomiarze czasu potrzebnego do osiągnięcia celu oraz długości przebytej drogi, traktowanych jako kluczowe wskaźniki efektywności. Niskie wartości w obu wymiarach można interpretować jako przejaw

sprawniejszego wykonania zadania. Ten eksperyment częściowo pokrywa się zakresem z eksperymentem 2a, ale jest przeprowadzony w innym budynku, co potencjalnie może wzmocnić siłę uzyskanych wcześniej wyników.

Projekt eksperymentu opiera się na założeniu, iż wyświetlenie w aplikacji nawigacyjnej wyznaczonej trasy nie tylko przyspiesza osiągnięcie wyznaczonego punktu docelowego, lecz również przyczynia się do istotnej redukcji ewentualnych odchyłeń od optymalnej ścieżki, ograniczając tym samym zjawisko błędzenia w budynku. Wyniki badania mogą stanowić podstawę do formułowania wniosków dotyczących zasadności implementacji rozwiązań tego typu w systemach wspomagających nawigację wewnątrz budynków, a także wyznaczać ramy dla dalszych, bardziej pogłębionych badań w zakresie projektowania narzędzi nawigacyjnych. Informacje podstawowe o eksperymencie 4 przedstawiono w tabeli 13.

Tabela 13. Informacje podstawowe o eksperymencie 4.

Eksperyment 4					
Kod	Budynek	Cel	Długość trasy wyznaczonej przez aplikację	Modyfikowane zmienne niezależne	Liczba uczestników
gg2	Gmach Główny	p. 307	123 metry	Wyświetlanie trasy	133

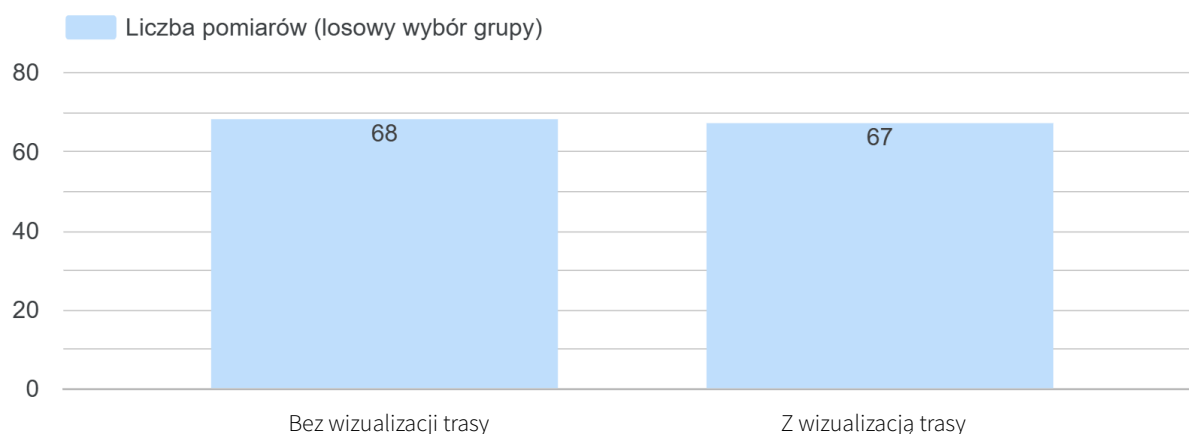
Eksperyment bazuje na modelu trójwymiarowym Gmachu Głównego PW. Punkt początkowy został wyznaczony w lokalizacji innej niż ta, w której uczestnicy zakończyli poprzedni test. Układ korytarzy i pomieszczeń w budynku umożliwia realizację zadania przy użyciu różnych tras, w tym zarówno z wykorzystaniem wind, jak i klatek schodowych. Poniżej przedstawiono trasę wyznaczoną przez aplikację nawigacyjną:



Rysunek 7.37. Trasa scenariusza gg2, linia ciągła pokazuje trasę na piętrze 1, linia przerywana pokazuje trasę na piętrze 4 (opracowanie własne)

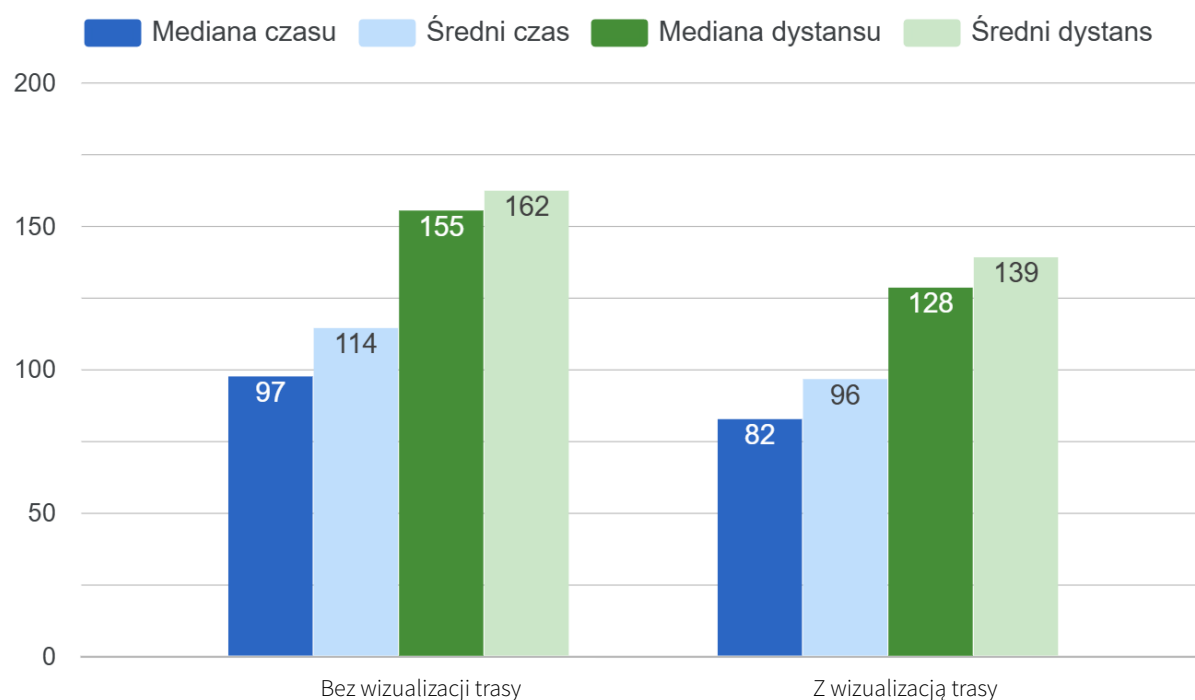
Przedstawienie i analiza wyników

Przed rozpoczęciem eksperymentu każdy ze 135 uczestników został losowo przydzielony do jednej z dwóch grup (Rysunek 7.38). Grupa „Bez wizualizacji trasy” to uczestnicy, którzy do dyspozycji mieli mapę z oznaczoną własną lokalizacją, ale bez pokazanej trasy. Grupa „Z wizualizacją trasy” to uczestnicy, którzy dodatkowo widzieli na mapie trasę prowadzącą do celu, wyznaczoną przez aplikację.



Rysunek 7.38. Rozkład liczebności grup eksperymentu 4, według wylosowanego przydziału do grupy z lub bez wizualizacji trasy (opracowanie własne)

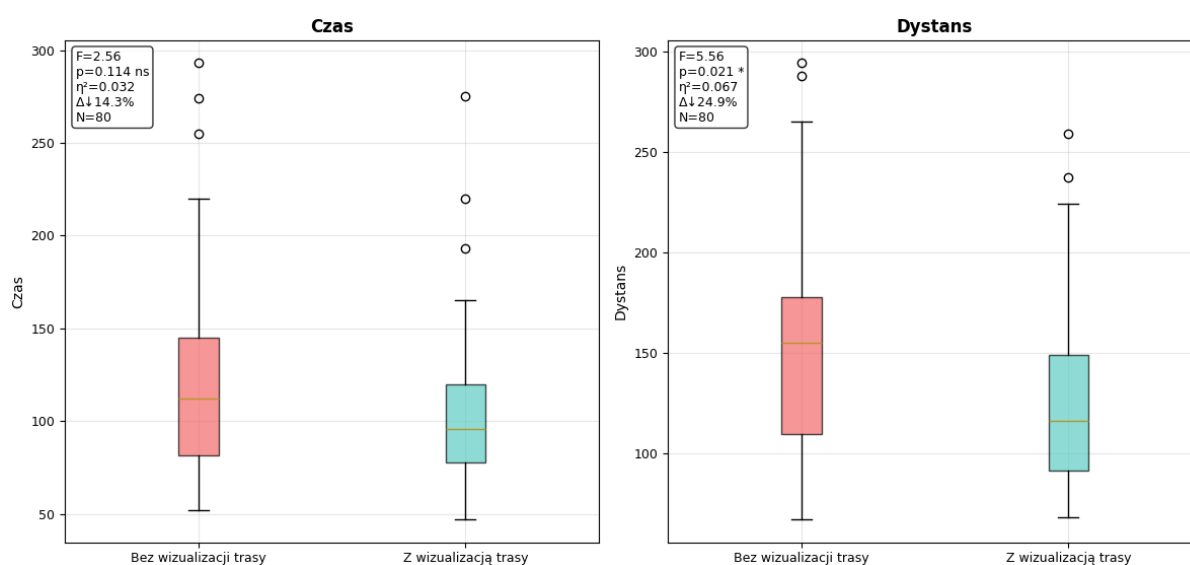
Poniższy wykres (Rysunek 7.39) prezentuje medianę oraz średnią, kolejno dla czasu przejścia (kolor niebieski) oraz przebytego dystansu (kolor zielony) przez uczestników w każdej z grup.



Rysunek 7.39. Czas oraz przebyty dystans w poszukiwaniu pomieszczenia w eksperymencie 4 (opracowanie własne)

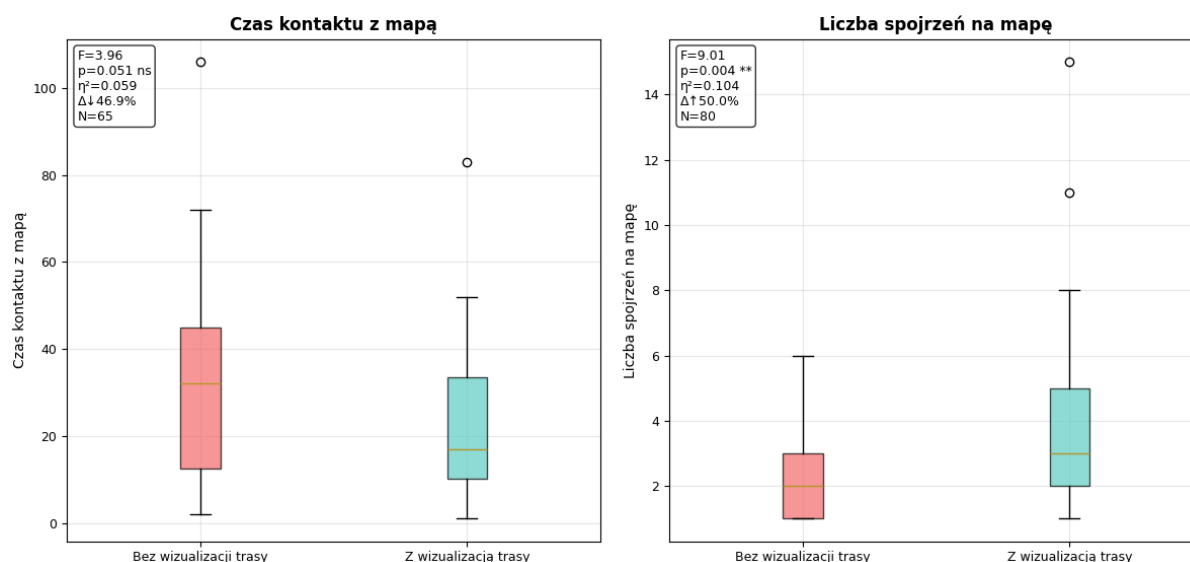
Każda mierzona wartość obniżyła się w grupie, której była prezentowana trasa. Konkretnie mediana czasu obniżyła się o 15%, a mediana dystansu o 17%.

Rezultat ten można również zaobserwować na wykresach rozkładu wyników (Rysunek 7.40).



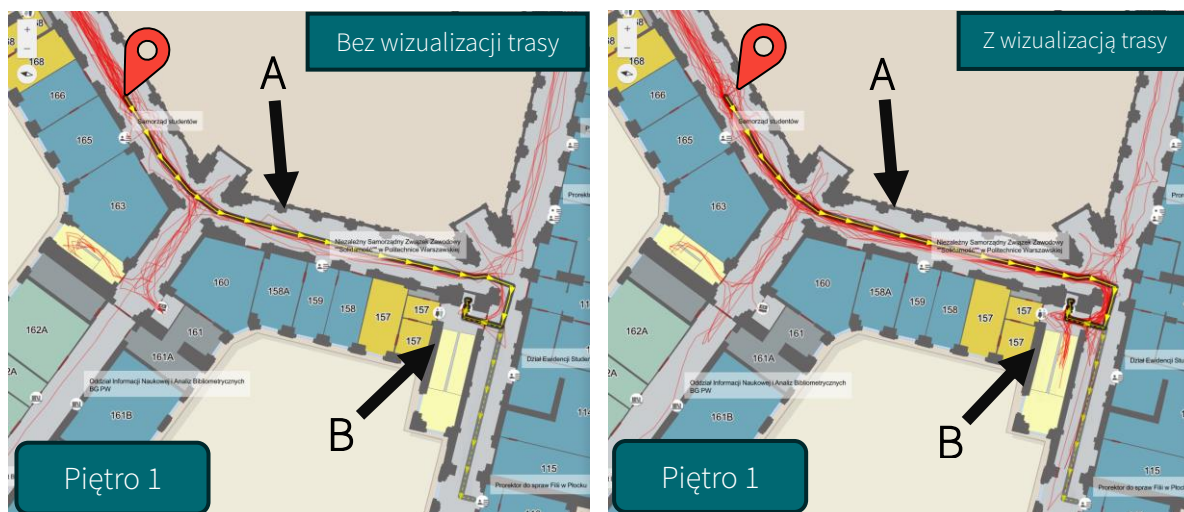
Rysunek 7.40. Rozkład wyników realizacji eksperymentu 4. Czas ukończenia po lewej, przebyty dystans po prawej (opracowanie własne)

Na uwagę zasługuje także analiza czasu kontaktu z mapą oraz liczby spojrzeń na mapę (Rysunek 7.41). Zasadnym jest wniosek, iż uczestnicy z grupy, która na mapie miała wyświetloną trasę z jednej strony spędzili mniej czasu na analizowaniu mapy, a z drugiej strony częściej na nią spoglądali, by podążać za wyznaczoną trasą.



Rysunek 7.41. Rozkład wyników realizacji eksperymentu 4. Czas kontaktu z mapą po lewej, liczba spojrzeń na mapę po prawej (opracowanie własne).

Na rysunku poniżej (Rysunek 7.42) przedstawiono porównanie zarejestrowanych tras grupy eksperymentalnej oraz kontrolnej. Możliwe jest zaobserwowanie, iż znacznie większa część badanych z grupy, której wyświetlono trasę obrała od początku najkrótszą możliwą trasę oraz skorzystała z windy znajdującej się najbliżej celu.



Rysunek 7.42. Porównanie zarejestrowanych tras uczestników eksperymentu 4.

Po lewej uczestnicy, którzy nie widzieli trasy, po prawej uczestnicy, którzy widzieli trasę.

Czerwoną pinezką oznaczono punkt startu scenariusza.

Strzałki „A” wskazują obszar, na którym widać, iż znacznie większa część badanych z grupy z wizualizacją trasy obrała najkrótszą możliwą drogę do celu, zgodną z trasą wyznaczoną przez aplikację nawigacyjną.

Strzałki „B” wskazują obszar, na którym widać, iż znacznie większa część badanych z grupy z wizualizacją trasy skorzystała z optymalnie ulokowanej windy lub klatki schodowej, co skracало czas dotarcia do celu i przebyty dystans (opracowanie własne).

Wnioski i podsumowanie eksperymentu

Eksperyment potwierdził hipotezę o pozytywnym wpływie wyświetlenia obliczonej przez aplikację optymalnej trasy na mapie na efektywność rzeczywistej drogi uczestników do podanego celu. Uczestnicy, którym udostępniono wizualizację trasy, wykonywali zadanie sprawniej, co przełożyło się na przeciętnie około 15% krótszy czas realizacji oraz 17% redukcję długości przebytej drogi. Skala efektu pozostaje umiarkowana, jednakże potwierdza zasadność stosowania tego typu rozwiązań jako czynnika wspierającego orientację w przestrzeni i ograniczającego ryzyko błędzenia. Wyniki sugerują, że choć prezentacja trasy nie eliminuje w pełni trudności w nawigacji wewnątrzbudynkowej, stanowi istotny krok w kierunku poprawy efektywności użytkowników i może stanowić podstawę do dalszych badań oraz praktycznych wdrożeń systemów wspomagających nawigację.

Dodatkowo, w toku realizacji scenariusza zaobserwowano zjawisko, które nie było pierwotnie przewidziane w założeniach badawczych. W grupie, której wyświetlano obliczoną trasę, zauważono interesującą dychotomię zachowań: użytkownicy poświęcali mniej czasu na szczegółową analizę mapy, lecz jednocześnie częściej do niej spoglądali, traktując ją głównie jako narzędzie kontrolne do podążania za trasą. **Obserwacje te podkreślają, że sposób**

prezentacji informacji kartograficznych nie tylko wpływa na efektywność samej nawigacji, lecz także kształtuje charakter interakcji użytkowników z mapą.

7.5.6. Eksperyment 5 – wpływ wyświetlenia na ekranie dodatkowych wskazówek nawigacyjnych na efektywność nawigacji oraz sposób korzystania z mapy

Cele badawcze i założenia eksperymentu

Celem eksperymentu było zbadanie wpływu wyświetlania dodatkowych wskazówek nawigacyjnych na efektywność nawigacji wewnątrz budynku oraz na sposób korzystania z mapy przez użytkowników. Uczestnicy po naciśnięciu klawisza spacji mieli dostęp do mapy z zaznaczoną aktualną pozycją oraz wyświetloną trasą wyznaczoną przez aplikację, jednakże grupy badawcze otrzymywały lub nie otrzymywały komunikatów nawigacyjnych w postaci strzałek kierunkowych. Istotą eksperymentu było określenie, w jakim stopniu dodatkowe wskazówki wizualne wpływają na skuteczność poruszania się po budynku oraz czy mogą one częściowo lub całkowicie zastąpić potrzebę korzystania z mapy.

Strzałki wskazujące najbliższy manewr do wykonania, wyświetlane były w lewym górnym rogu ekranu w okrągłej ramce imitującej widok, który mógłby wyświetlać się na tarczy zegarka typu smartwatch (Rysunek 7.43). Strzałki pojawiały się w odpowiednich momentach (przed zbliżeniem się do skrzyżowania), sugerując kierunek ruchu bez konieczności analizowania mapy. Obie grupy badawcze miały do przebycia identyczną trasę, a różnice w dostępności kierunkowych strzałek nawigacyjnych stanowiły jedyną zmienną niezależną w eksperymencie.

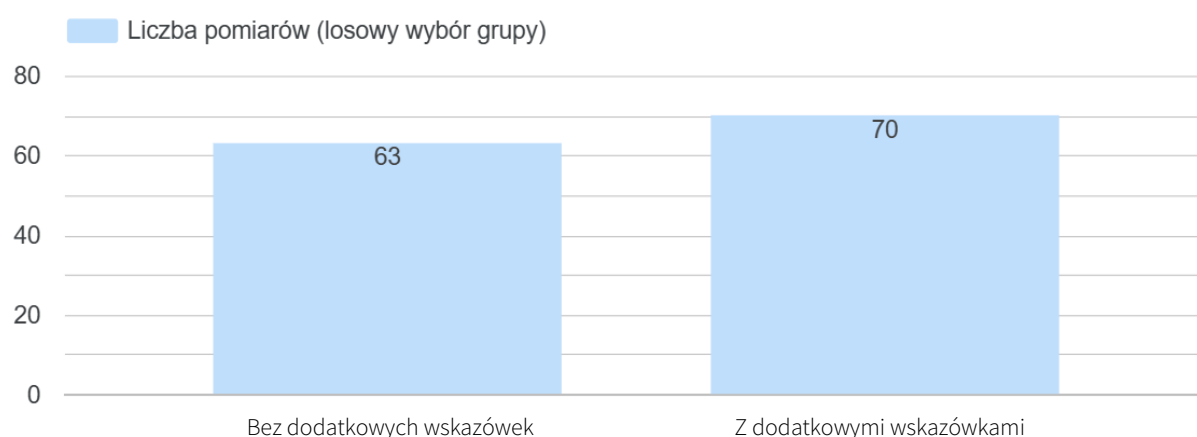


Rysunek 7.43. Zrzut ekranu symulatora, na którym w lewym górnym rogu widać aktywny tryb dodatkowych wskazówek w formie strzałek (opracowanie własne)

Hipotezą eksperymentu było założenie, że wyświetlanie dodatkowych wskazówek ułatwi uczestnikom nawigację, co przełoży się na krótszy czas realizacji zadania oraz zmniejszoną częstotliwość korzystania z mapy. Oczekiwano, że użytkownicy mający dostęp do strzałek będą w stanie podejmować szybsze decyzje w punktach decyzyjnych. Spodziewano się również, że grupa korzystająca ze strzałek będzie rzadziej aktywować mapę, a w przypadku jej użycia będzie poświęcać jej mniej czasu. Dodatkowym celem badania było sprawdzenie, czy uproszczone komunikaty kierunkowe mogą samodzielnie zastąpić mapę w scenariuszach nawigacyjnych, czy raczej stanowią one użyteczne uzupełnienie tradycyjnych narzędzi kartograficznych, redukujące obciążenie poznawcze związane z częstym odwoływaniem się do mapy. Informacje podstawowe o eksperymencie 5 przedstawiono w tabeli 14.

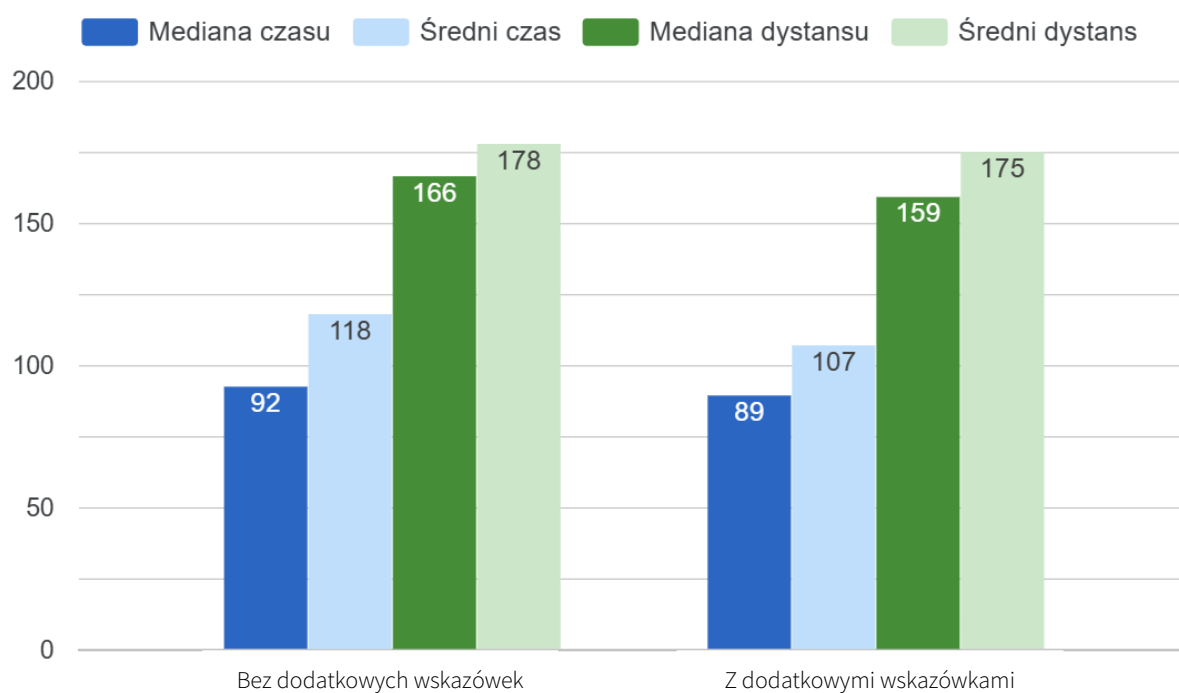
Przedstawienie i analiza wyników

Do analizy wyników eksperymentu wzięto pod uwagę dane ze **133 przejść** uczestników zebranych w scenariuszu *gg3*. Podział uczestników na grupę „Bez dodatkowych wskazówek” oraz „Z dodatkowymi wskazówkami” przedstawiono na rysunku poniżej (Rysunek 7.45).



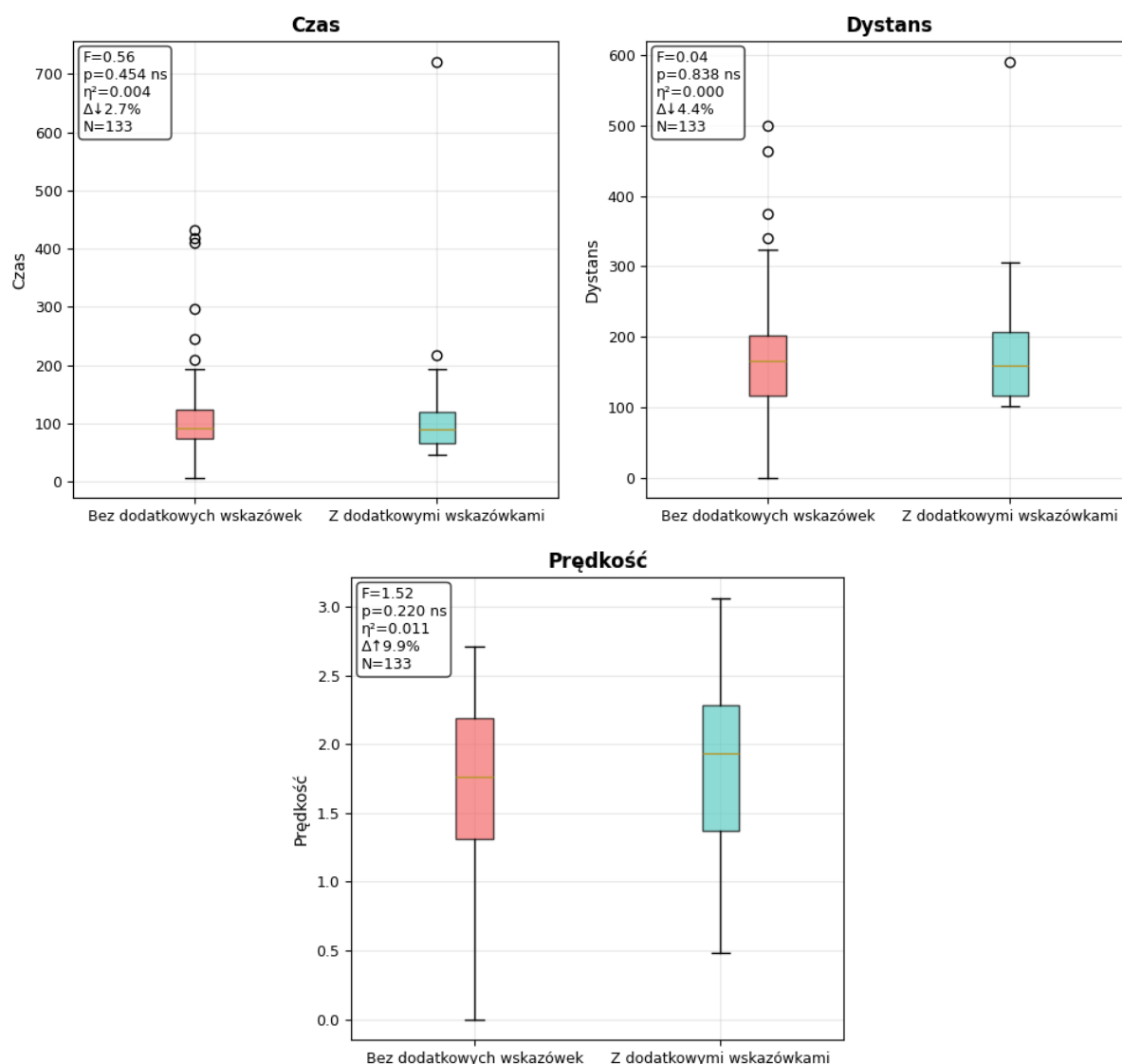
Rysunek 7.45. Rozkład liczebności grup eksperymentu 5, według wylosowanego przydziału do grupy z lub bez wyświetlania dodatkowych wskazówek (opracowanie własne)

Poniższy wykres (Rysunek 7.46) prezentuje medianę oraz średnią, kolejno dla czasu przejścia (kolor niebieski) oraz przebytego dystansu (kolor zielony) przez uczestników w każdej z grup.



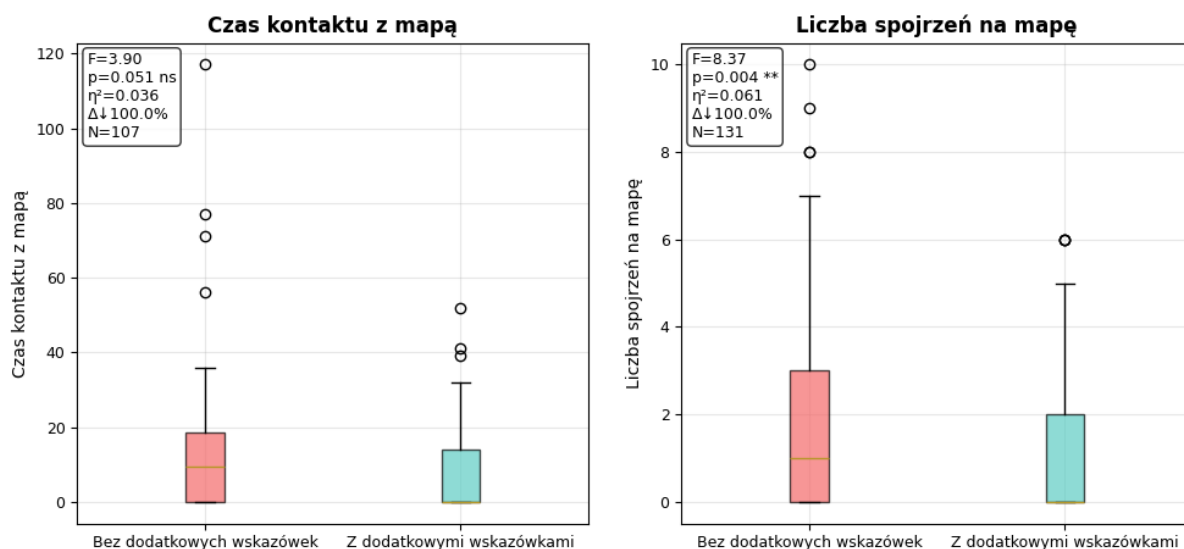
Rysunek 7.46. Czas oraz przebyty dystans w poszukiwaniu pomieszczenia w eksperymencie 5 (opracowanie własne)

Obserwowalne są jedynie minimalne zmiany – mediana czasu obniżyła się o około **3%**, a mediana dystansu o około **4%**. Przełożyło się to na prawie **10%** wzrost średniej prędkości poruszania. Czas, dystans oraz prędkość nie wykazały istotności statystycznej ($p > 0.05$) pomimo dużej grupy badawczej (Rysunek 7.47).



Rysunek 7.47. Rozkład statystyczny wyników realizacji eksperymentu 5. Czas ukończenia po lewej, przebyty dystans po prawej, średnia prędkość na dole (opracowanie własne).

Z kolei inaczej wyglądają statystyki interakcji z mapą (Rysunek 7.48). W wyniku prezentowania dodatkowych wskazówek do zera spadły mediana czasu kontaktu z mapą, oraz mediana liczby spojrzeń na mapę, co oznacza, że ponad połowa uczestników w ogóle nie potrzebowała mapy, aby dotrzeć do celu. Wynik jest istotny statystycznie przy $p = 0.004$ dla liczby spojrzeń na mapę i blisko istotności dla czasu kontaktu z mapą ($p = 0.051$). Różnicę w wartości p częściowo tłumaczy mniejsza próba badawcza dla czasu kontaktu z mapą.



Rysunek 7.48. Rozkład statystyczny statystyk interakcji z mapą podczas eksperymentu 5. Czas kontaktu z mapą po lewej, liczba spojrzeń na mapę po prawej (opracowanie własne).

Wnioski i podsumowanie eksperymentu

Eksperyment wykazał, że dodatkowe wskazówki nawigacyjne w formie strzałek nie wpłynęły istotnie statystycznie na podstawowe parametry efektywności nawigacji, takie jak czas dotarcia do celu czy przebyty dystans. Kluczowym odkryciem eksperymentu był natomiast istotny wpływ wskazówek na sposób korzystania z mapy. W grupie z dodatkowymi wskazówkami ponad połowa uczestników w ogóle nie potrzebowała mapy do dotarcia do celu.

Badanie potwierdza, że uproszczone wskazówki nawigacyjne w formie strzałek mogą w znacznym stopniu zastąpić tradycyjną mapę podczas nawigacji wewnątrz budynków, zmniejszając obciążenie poznawcze użytkowników. Wskazówki nie przyspieszyły znacząco nawigacji ani nie skróciły przebytej drogi, natomiast umożliwiły ponad połowie uczestników całkowitą rezygnację z odwoływania się do mapy. Sugeruje to, że tego typu wskazówki są szczególnie wartościowe jako narzędzie redukujące konieczność interakcji z mapą, co może mieć praktyczne zastosowanie np. w urządzeniach ubieralnych typu smartwatch, na których wyświetlenie mapy może nie być efektywne.

7.5.7. Eksperyment 6 – sprawdzenie potrzeby korzystania z mapy w dobrze oznakowanym budynku

Cele badawcze i założenia eksperymentu

Głównym celem szóstego eksperymentu było określenie, czy w warunkach stosunkowo dobrego oznakowania wnętrza budynku korzyści wynikające z udostępnienia mapy będą równie istotne, jak w pozostałych przeprowadzonych badaniach. Eksperyment miał za zadanie zweryfikować, na ile umieszczone na korytarzach oznaczenia informujące o numerach pomieszczeń w poszczególnych segmentach budynku mogą stanowić wystarczające wsparcie nawigacyjne, eliminując lub ograniczając potrzebę korzystania z mapy/planu budynku.

Uczestnicy eksperymentu poruszali się po wirtualnym modelu Gmachu Elektrotechniki Politechniki Warszawskiej. W modelu dodano dodatkowe (nie istniejące w rzeczywistości) oznaczenia na korytarzach – informacje wskazujące zakresy numerów pomieszczeń dostępnych w danym segmencie korytarza (Rysunek 7.49). Projekt eksperymentu zakładał podział uczestników na dwie grupy badawcze. Pierwsza grupa nie miała dostępu do mapy i musiała polegać wyłącznie na własnej orientacji w przestrzeni oraz widocznych oznaczeniach. Druga grupa otrzymała możliwość wyświetlenia mapy po naciśnięciu przycisku spacji.



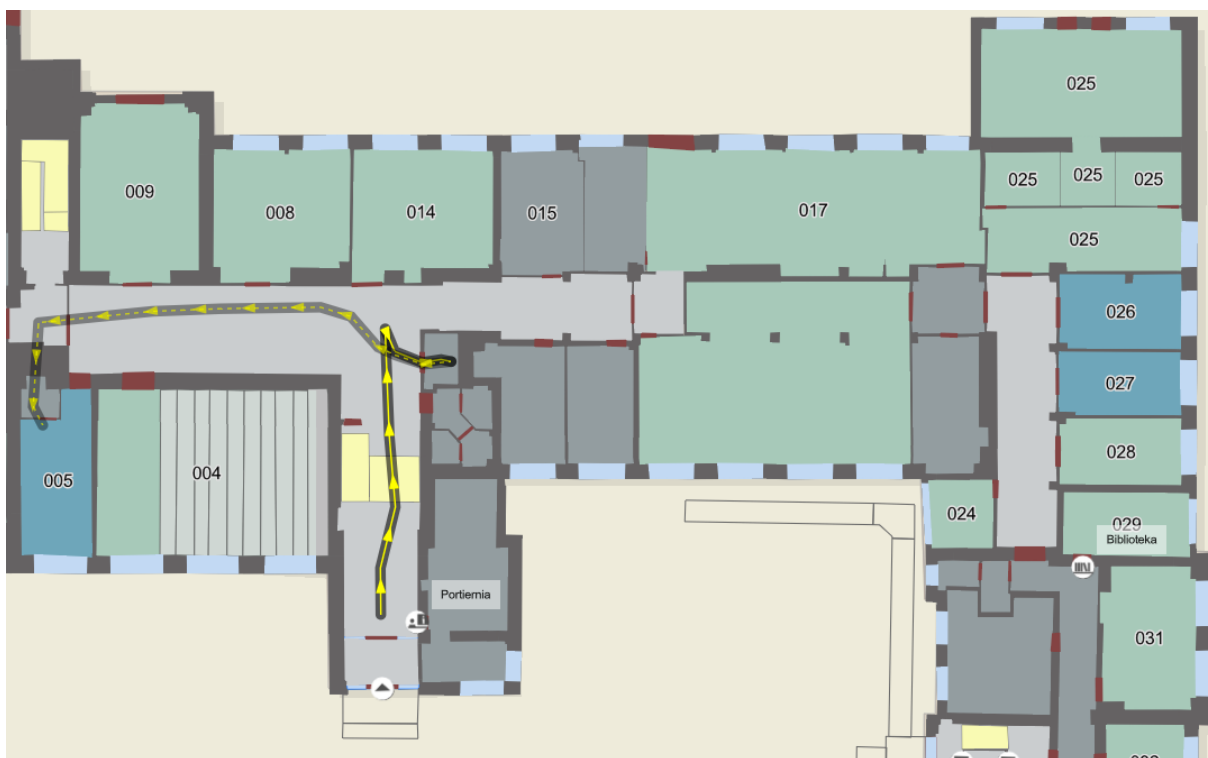
Rysunek 7.49. Zrzut ekranu symulatora, w centralnej części widać przykład oznakowania korytarzy (opracowanie własne)

Sformułowana hipoteza zakładała, że nawet przy obecności dodatkowych oznakowań, dostęp do mapy będzie nadal przynosił wymierne korzyści w postaci skrócenia czasu realizacji zadania i przebytej drogi. Oczekiwano, że oznaczenia korytarzowe, choć pomocne w zawężeniu obszaru poszukiwań, nie zapewnią tak precyzyjnego wsparcia nawigacyjnego jak mapa z zaznaczoną trasą. Spodziewano się również, że uczestnicy nieposiadający dostępu do mapy będą częściej eksplorować niewłaściwe segmenty korytarzy, pomimo obecności oznakowań, co przełoży się na dłuższy czas wykonania zadania oraz większą dezorientację przestrzenną. Informacje podstawowe o eksperymencie 6 przedstawiono w tabeli 15.

Tabela 15. Informacje podstawowe o eksperymencie 6.

Eksperyment 6					
Kod	Budynek	Cel	Długość trasy wyznaczonej przez aplikację	Modyfikowane zmiennie niezależne	Liczba uczestników
et1	Gmach Elektrotechniki	p. 102	76m	Dostępność mapy	126
et2	Gmach Elektrotechniki	p. 423A	111m	Dostępność mapy	126

Na eksperyment szósty składały się dwa scenariusze o takiej samej zasadzie działania. W Gmachu Elektroniki punkt początkowy został wyznaczony przy głównym wejściu do budynku dla obu scenariuszy. Układ korytarzy i pomieszczeń w budynku umożliwia realizację zadania przy użyciu trasy z wykorzystaniem wind, jak i klatek schodowych. Poniżej przedstawiono trasę wyznaczoną przez aplikację nawigacyjną kolejno dla scenariusza *et1* i *et2* (Rysunek 7.50, Rysunek 7.51).



Rysunek 7.50. Trasa wyznaczona przez aplikację dla scenariusza et1, linia ciągła pokazuje trasę na piętrze 0, linia przerywana pokazuje trasę na piętrze 1 (opracowanie własne)

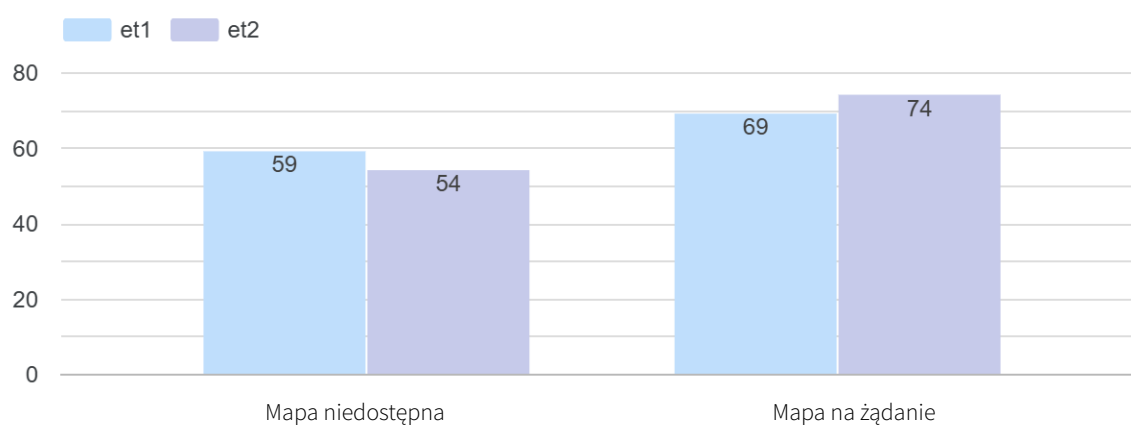


Rysunek 7.51. Trasa scenariusza et2, linia ciągła pokazuje trasę na piętrze 0, linia przerywana pokazuje trasę na piętrze 4 (opracowanie własne)

Przedstawienie i analiza wyników

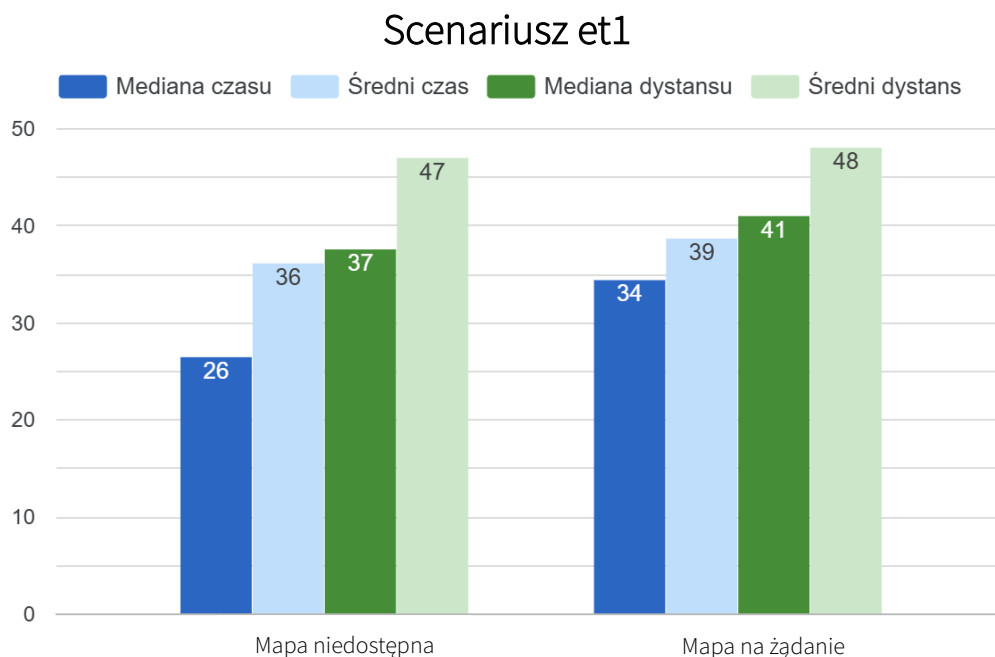
W eksperymencie przeanalizowano statystyki **256 przejść** uczestników zebranych w dwóch scenariuszach *et1* oraz *et2*. Wstępna analiza wykazała rozbieżności w wynikach pomiędzy oboma scenariuszami, dlatego poniżej są one raportowane osobno. Podział uczestników na dwie grupy badawcze przedstawiono na rysunku poniżej (Rysunek 7.52).

Liczba pomiarów (losowy wybór grupy)

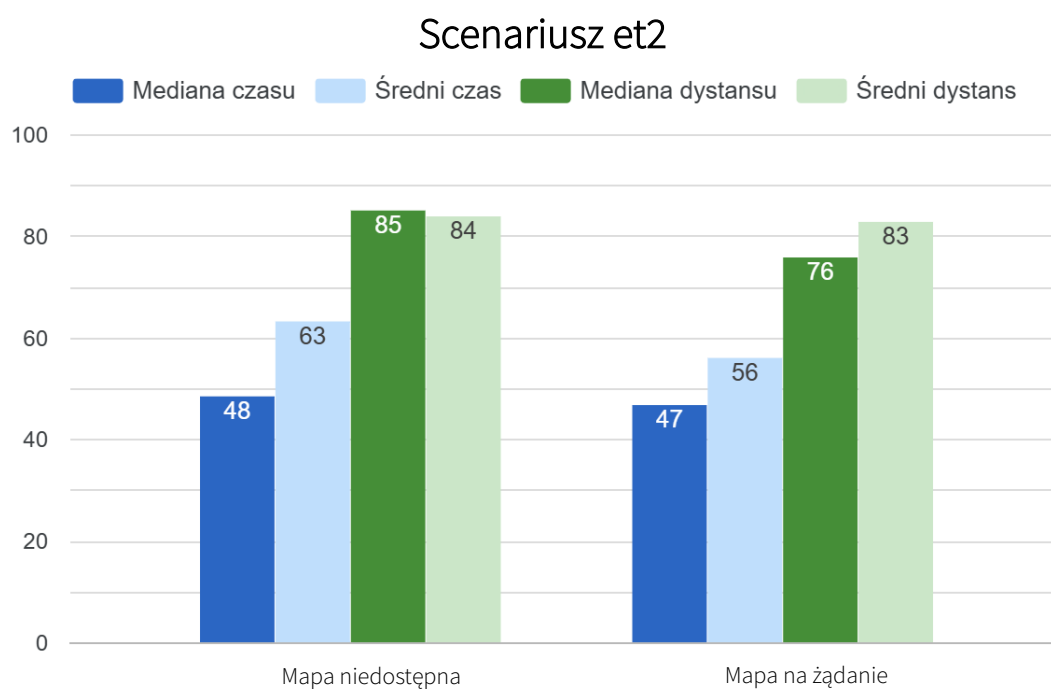


Rysunek 7.52. Rozkład liczebności grup eksperymentu 6, według wylosowanej dostępności mapy (opracowanie własne)

Poniższe wykresy (Rysunek 7.54, Rysunek 7.54) prezentują medianę oraz średnią, kolejno dla czasu przejścia (kolor niebieski) oraz przebytego dystansu (kolor zielony) przez uczestników w każdej z grup w scenariuszach *et1* i *et2*.



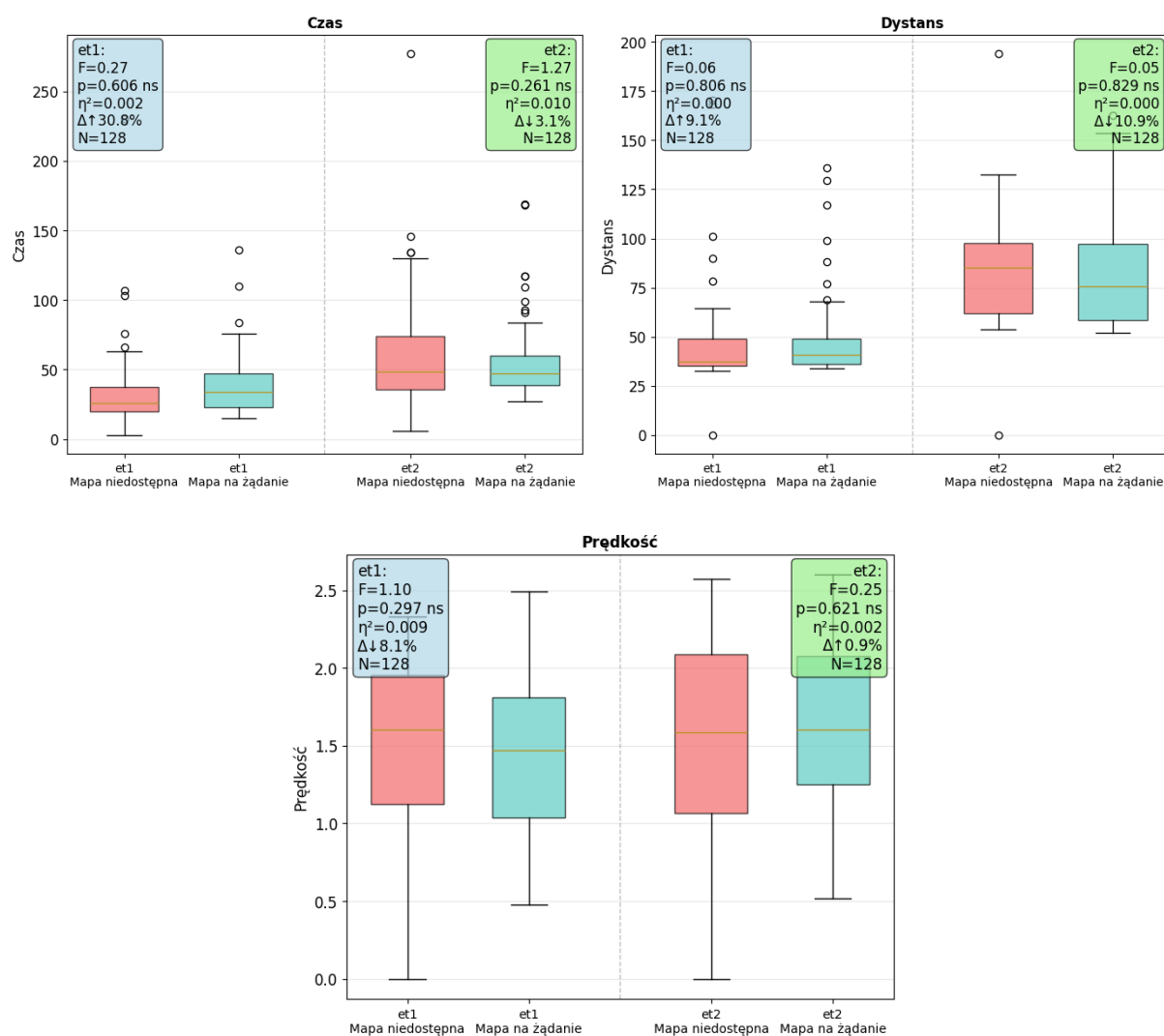
Rysunek 7.53. Czas oraz przebyty dystans w poszukiwaniu pomieszczenia w scenariuszu et1 (opracowanie własne)



Rysunek 7.54. Czas oraz przebyty dystans w poszukiwaniu pomieszczenia w scenariuszu et2 (opracowanie własne)

Analiza statystyczna wykazuje niespójność obserwowanych efektów pomiędzy różnymi trasami. Scenariusz et1 wykazuje lekko negatywny efekt udostępnienia mapy uczestnikom – czas oraz przebyty dystans wydłużyły się, a średnia prędkość spadła o około 8%. Z kolei scenariusz et2 wykazuje efekt odwrotny – uczestnicy mający na żądanie dostępną mapę

minimalnie skrócili czas wykonania zadania oraz przebyty dystans, a ich średnia prędkość wzrosła o niespełna 1%. Wszystkie uzyskane wyniki nie wykazują istotności statystycznej ($p > 0.05$) pomimo wystarczającej grupy badawczej $N = 128$.



Rysunek 7.55. Rozkład statystyczny wyników realizacji eksperymentu 6 z podziałem na dwa scenariusze. Czas ukończenia po lewej, przebyty dystans po prawej, średnia prędkość na dole (opracowanie własne).

Wnioski i podsumowanie eksperymentu

Eksperyment 6 wykazał nieoczekiwaną niespójność wyników między dwoma przeprowadzonymi scenariuszami. W scenariuszu et1 udostępnienie mapy przyniosło paradoksalnie negatywny efekt – uczestnicy z dostępem do mapy potrzebowali więcej czasu i pokonali dłuższą trasę niż grupa bez mapy. Natomiast w scenariuszu et2 zaobserwowano

odwrotną tendencję – dostęp do mapy nieznacznie skrócił czas wykonania zadania i przebyty dystans, choć różnice te były minimalne.

Przypuszczalnie na rozbieżność wyników wpłynęły dwa kluczowe czynniki. Po pierwsze, scenariusz *et1* obejmował znacznie krótszą trasę (76m) niż *et2* (111m), co mogło sprawić, że przy krótkim dystansie korzyści z mapy były trudniejsze do zaobserwowania, a sam proces jej uruchomienia i konsultacji mógł stanowić dodatkowe obciążenie czasowe. Po drugie, scenariusz *et2* był realizowany bezpośrednio po *et1*, co oznacza, że uczestnicy mieli już wstępną mapę mentalną układu budynku – znali lokalizację klatki schodowej oraz ogólną strukturę korytarzy. Ta wcześniejsza wiedza przestrzenna mogła zredukować różnice między grupami, ponieważ obie grupy korzystały z już nabytego zrozumienia topologii budynku.

Sformułowana hipoteza zakładająca, że dostęp do mapy przyniesie wymierne korzyści nawigacyjne nawet w dobrze oznakowanym budynku, nie została potwierdzona w żadnym ze scenariuszy. Brak istotności statystycznej wyników ($p > 0.05$) oraz niewielkie różnice między grupami sugerują, że w warunkach stosunkowo dobrego oznakowania i prostego układu architektonicznego Gmachu Elektrotechniki, korzyści z udostępnienia mapy są znikome lub w ogóle nie występują. Dodatkowe oznaczenia informujące o zakresach numerów pomieszczeń w poszczególnych segmentach korytarzy okazały się wystarczającym wsparciem nawigacyjnym dla większości uczestników.

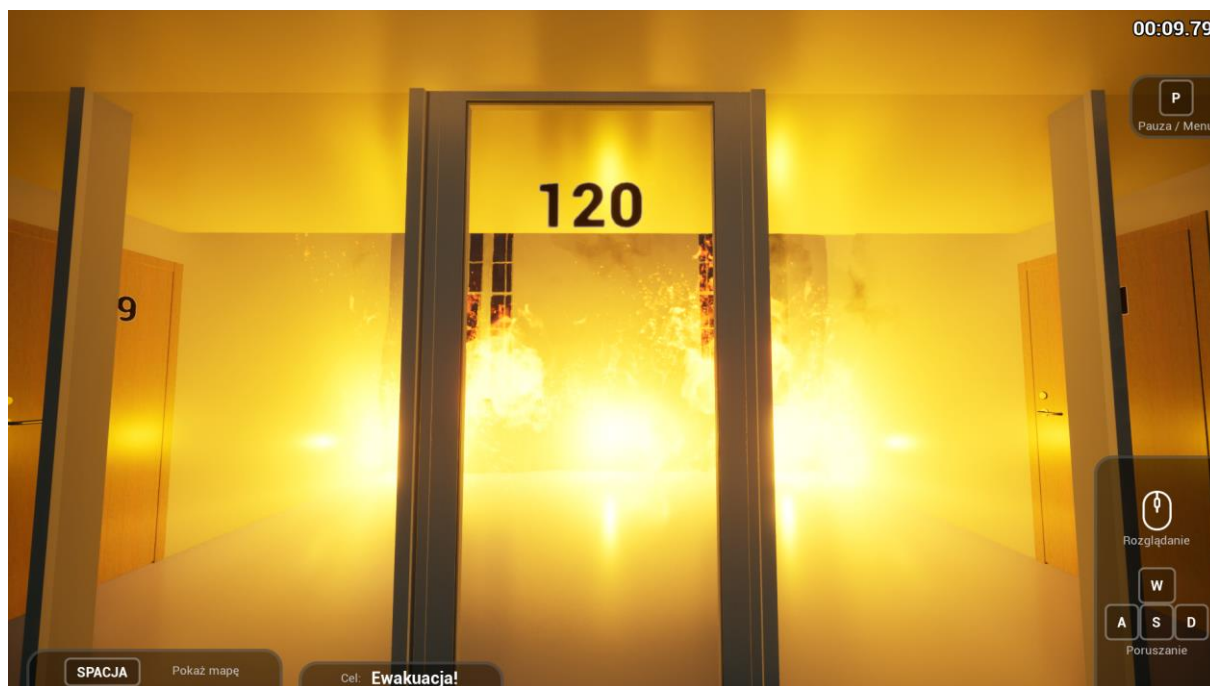
Obserwowana niespójność wyników ma istotne implikacje metodologiczne dla przyszłych badań. Wyniki wskazują, że efektywność rozwiązań nawigacyjnych może być silnie zależna od charakterystyki konkretnej trasy – jej długości, złożoności oraz kontekstu wykonywania zadania. Formułowane hipotezy powinny być weryfikowane na bardziej zróżnicowanych trasach, obejmujących różne poziomy trudności nawigacyjnej, długości i stopnie skomplikowania układu przestrzennego. Eksperyment pokazuje również wagę kontrolowania efektu uczenia się – kolejność wykonywania scenariuszy oraz nabywanie wiedzy przestrzennej w trakcie badania mogą znacząco wpływać na obserwowane efekty i prowadzić do trudności w interpretacji wyników.

7.5.8. Eksperyment 7 – wpływ wyświetlania trasy podczas poszukiwania pomieszczenia w budynku na obroną później trasę ewakuacji

Cele badawcze i założenia eksperymentu

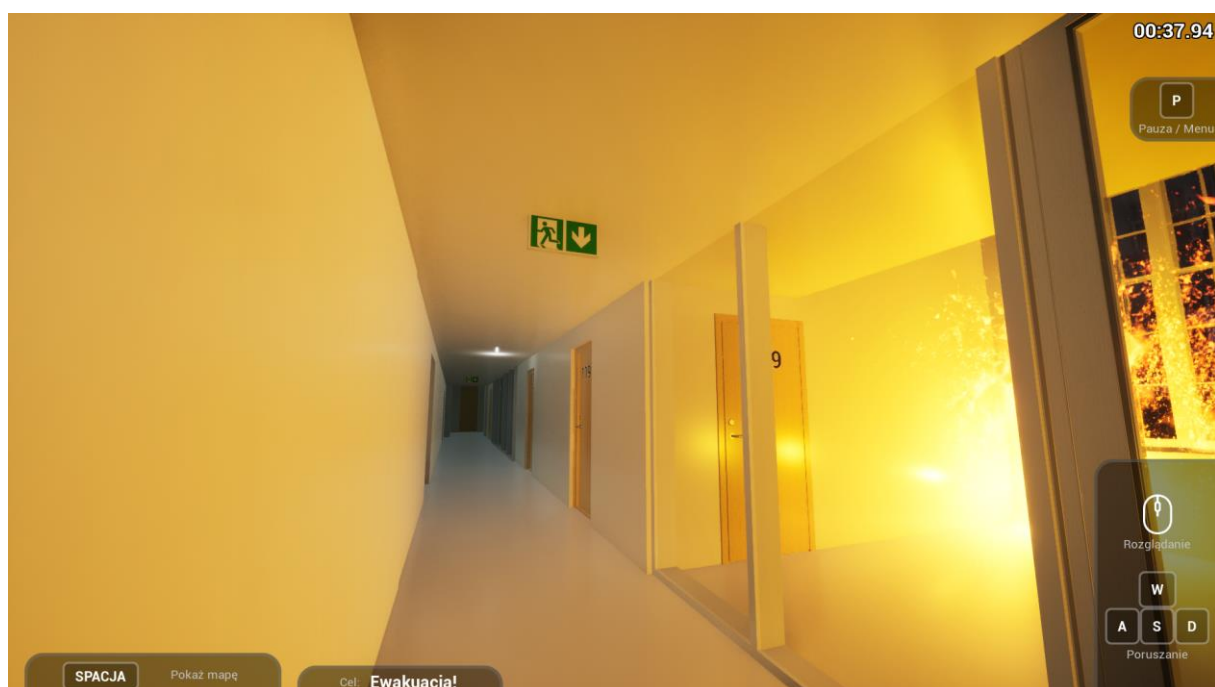
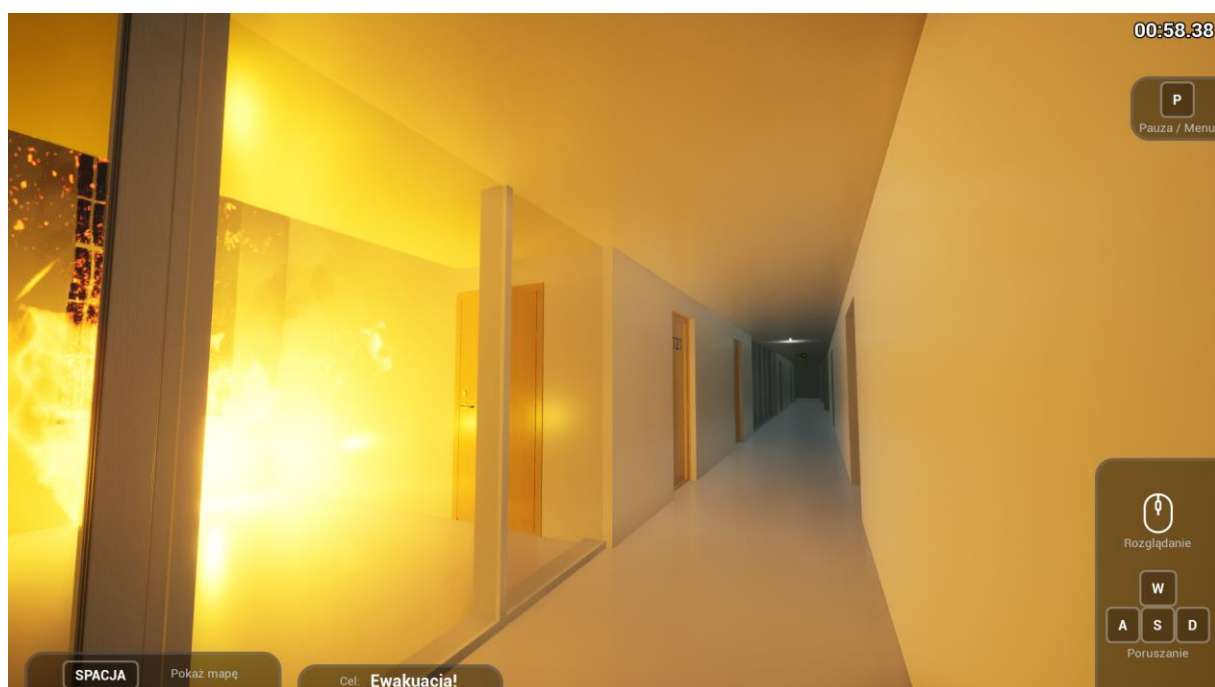
Głównym celem ostatniego eksperymentu było zbadanie, czy wyświetlenie trasy na mapie w pierwszym etapie, podczas poszukiwania pomieszczenia w budynku, wpływa na wybór drogi ewakuacyjnej w drugim etapie, polegającym na konieczności ewakuacji z tego pomieszczenia. Eksperyment miał zweryfikować hipotezę dotyczącą zjawiska preferowania znanej trasy nawigacyjnej w warunkach stresu, nawet gdy dostępna jest krótsza, lecz nieznana alternatywa.

Projekt eksperymentu zakładał realizację dwóch kolejnych scenariuszy. W pierwszym scenariuszu uczestnicy zostali podzieleni na dwie grupy. Pierwsza grupa dysponowała mapą bez wizualizacji trasy i musiała samodzielnie planować drogę do wskazanego pomieszczenia, natomiast druga grupa korzystała z mapy z zaznaczoną trasą prowadzącą bezpośrednio do celu. Obie grupy otrzymały identyczne zadanie nawigacyjne polegające na dotarciu do tego samego pomieszczenia rozpoczynając od głównego wejścia do budynku. W drugim scenariuszu, następującym bezpośrednio po pierwszym, zasymulowano sytuację pożaru (Rysunek 7.56).



Rysunek 7.56. Zrzut ekranu początku scenariusza archi2, przedstawiający pożar w p. 120, który był celem scenariusza archi1 (opracowanie własne)

Zadaniem uczestników była natychmiastowa ewakuacja z budynku w możliwie najkrótszym czasie. Model wyposażono w symbole drogi ewakuacyjnej umieszczone na ścianach (rozmieszczenie symboli nie było odwzorowaniem rzeczywistego oznakowania budynku). Układ przestrzenny budynku oferował uczestnikom dwa kierunki ewakuacji: pierwszy pokrywał się z trasą dotarcia do pomieszczenia docelowego z pierwszego scenariusza, natomiast drugi prowadził do bliższego wyjścia, jednakże wymagał poruszania się nieznaną wcześniej trasą. Bliższe umiejscowienie symbolu drogi ewakuacyjnej sugeruje optymalny kierunek ucieczki (Rysunek 7.57).



Rysunek 7.57. Widok na możliwe drogi ewakuacji. Na pierwszym rysunku droga dalsza, którą dotarto do tego miejsca w poprzednim scenariuszu. Znak ewakuacji jest ledwo widoczny w głębi korytarza. Na drugim rysunku droga bliższa, nieznana jeszcze uczestnikom, wyraźnie oznaczona znakiem ewakuacji (opracowanie własne).

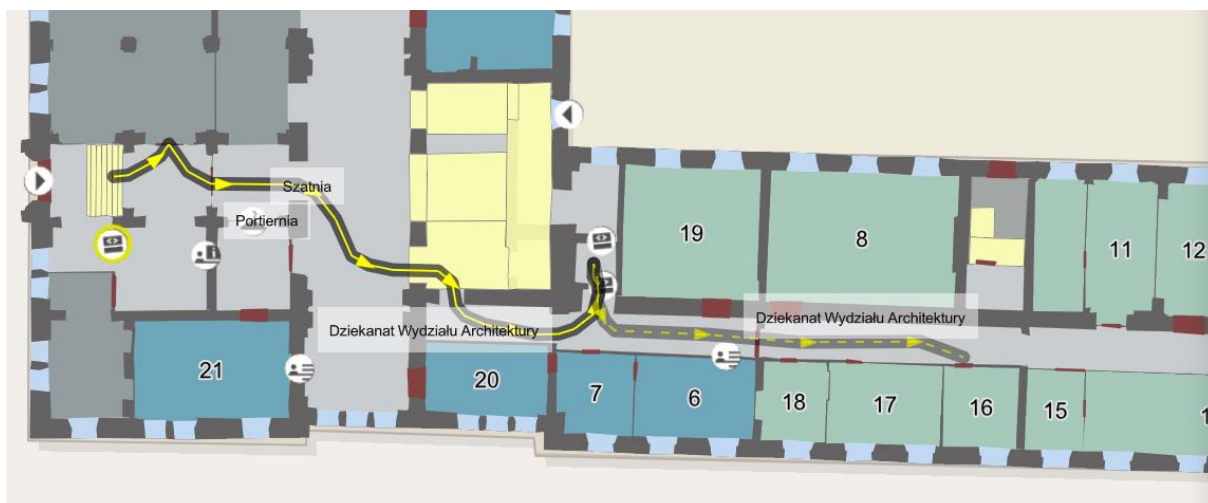
Sformułowana hipoteza zakładała, że uczestnicy, którzy w pierwszym scenariuszu korzystali z mapy z zaznaczoną trasą, będą częściej wybierać w sytuacji ewakuacyjnej znaną drogę prowadzącą z powrotem, pomimo dostępności krótszej alternatywy. Oczekiwano, że wizualizacja trasy na mapie przyczyni się do silniejszego zapamiętania określonej drogi i ukształtowania mentalnej reprezentacji przestrzennej skoncentrowanej wokół tej konkretnej

ścieżki. Spodziewano się również, że uczestnicy nieposiadający wcześniej wizualizacji trasy będą bardziej skłonni do wyboru krótszej, nieznanym drogi ewakuacyjnej, kierując się oznaczeniami oraz intuicyjną oceną odległości do wyjścia, związaną z wcześniejszą dogłębną analizą mapy, która była potrzebna do samodzielnego odnalezienia pomieszczenia w pierwszym scenariuszu tego eksperymentu. Informacje podstawowe o eksperymencie 7 przedstawiono w tabeli 16.

Tabela 16. Informacje podstawowe o eksperymencie 7

Eksperyment 7					
Kod	Budynek	Cel	Długość trasy wyznaczonej przez aplikację	Modyfikowane zmienne niezależne	Liczba uczestników
archi1	Gmach Architektury	p. 120	92 metry	Wyświetlanie trasy	127
archi2	Gmach Architektury	Dowolne wyjście	–	–	127

Scenariusze tego eksperymentu realizowane są na terenie wirtualnego modelu Gmachu Architektury Politechniki Warszawskiej. Zadanie można było zrealizować zarówno korzystając z głównej klatki schodowej, jak i odnajdując windę. Poniżej przedstawiono trasę wyznaczoną przez aplikację nawigacyjną kolejno dla scenariusza *archi1* (Rysunek 7.58).



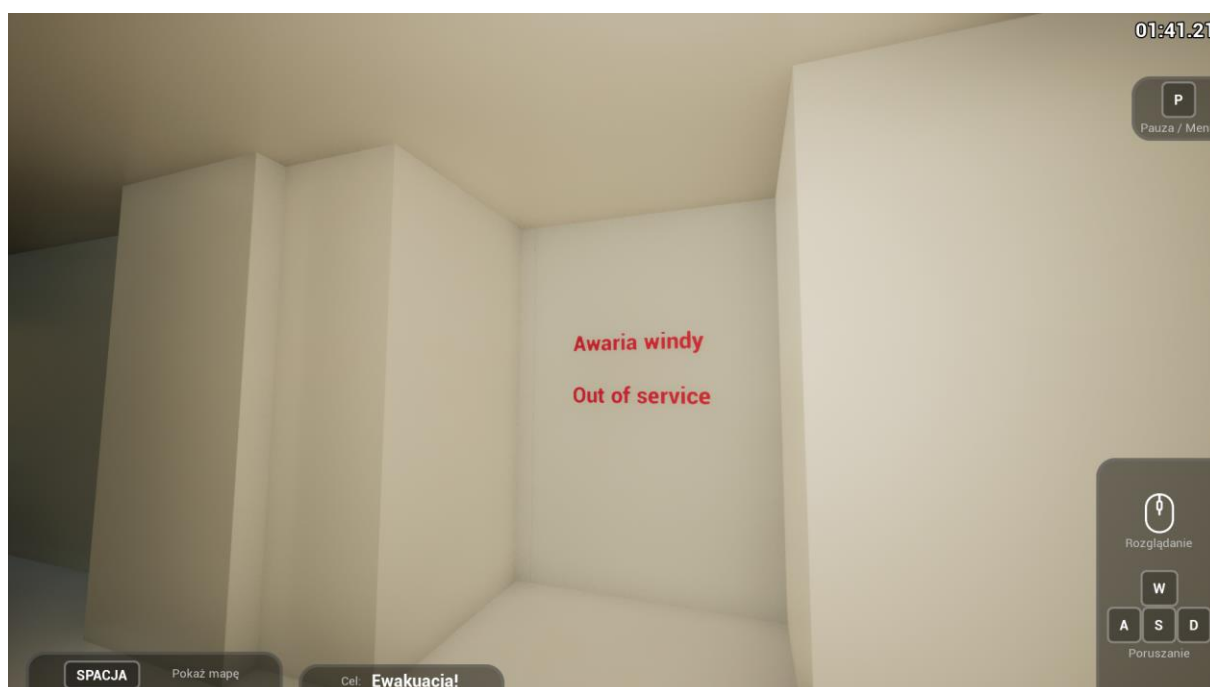
Rysunek 7.58. Trasa wyznaczona przez aplikację dla scenariusza archi1, linia ciągła pokazuje trasę na piętrze 0, linia przerywana pokazuje trasę na piętrze 1 (opracowanie własne)

Jak wyjaśniono powyżej w scenariuszu archi2 możliwa była ewakuacja dłuższą i krótszą drogą. Rysunek 7.59 prezentuje ścieżki obrane przez uczestników – dłuższą po lewej stronie od miejsca startu i krótszą po prawej stronie od miejsca startu.



Rysunek 7.59. Wizualizacja dostępnych dróg ewakuacji w scenariuszu archi2. Czerwoną pinezką oznaczono punkt startu. Niebieskimi flagami oznaczono dostępne wyjścia ewakuacyjne, do których dotarcie użytkownika kończyło scenariusz. Czerwone linie to zarejestrowane ścieżki uczestników (opracowanie własne).

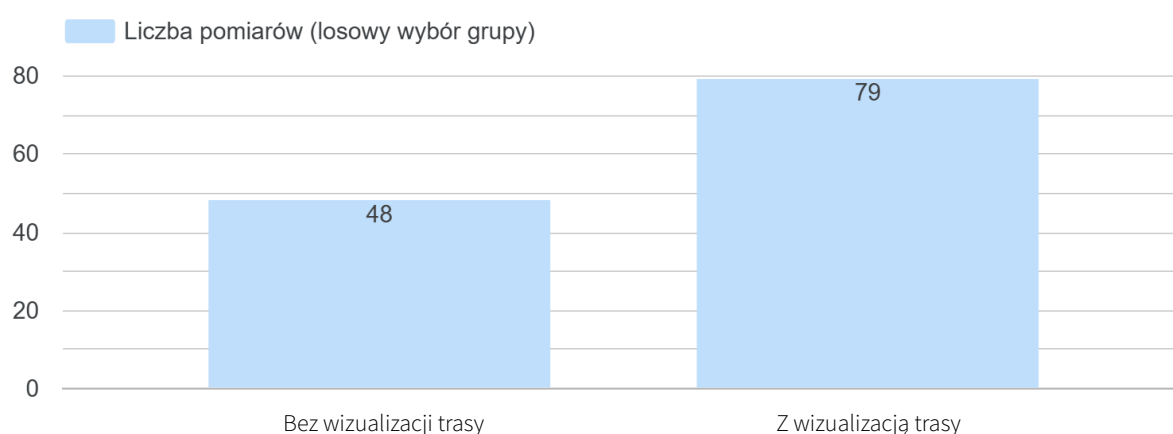
Ponadto w scenariuszu archi2, który symulował sytuację zagrożenia wyłączono możliwość korzystania z windy (Rysunek 7.60), co mogło być utrudnieniem dla osób, które skorzystały z niej, aby dostać się na pierwsze piętro w scenariuszu archi1.



Rysunek 7.60. Zrzut ekranu prezentujący dezaktywowaną windę (opracowanie własne)

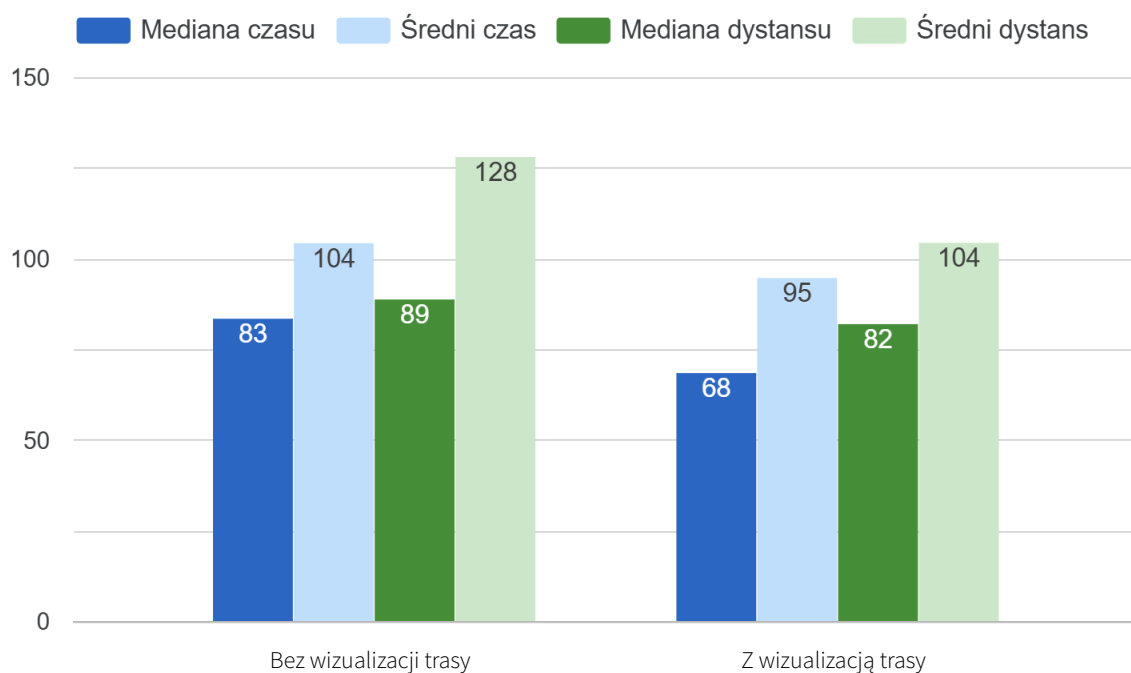
Przedstawienie i analiza wyników

Przed rozpoczęciem scenariusza *archi1* każdy ze 127 uczestników został losowo przydzielony do jednej z dwóch grup (Rysunek 7.61). Grupa „Bez wizualizacji trasy” to uczestnicy, którzy do dyspozycji mieli mapę z oznaczoną własną lokalizacją, ale bez pokazanej trasy. Grupa „Z wizualizacją trasy” to uczestnicy, którzy dodatkowo widzieli na mapie trasę prowadzącą do celu, wyznaczoną przez aplikację.



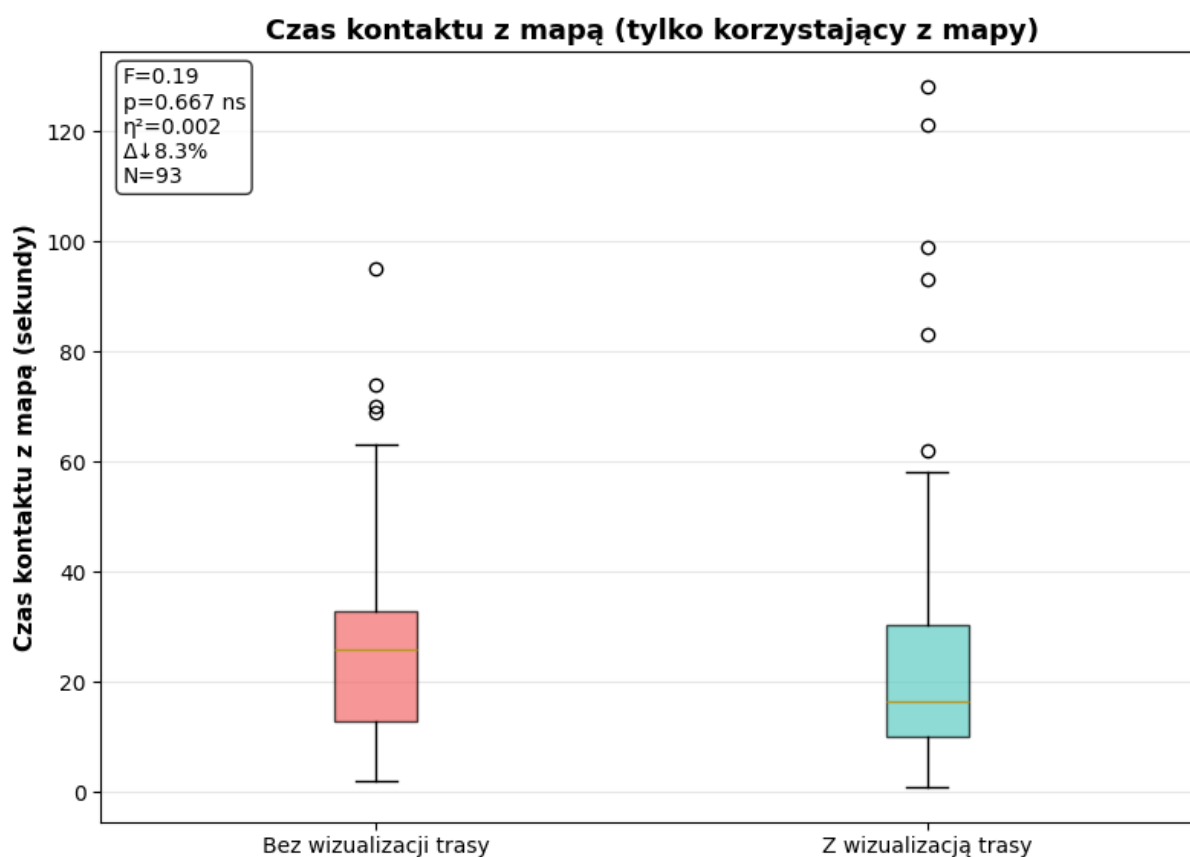
Rysunek 7.61. Rozkład liczebności grup eksperymentu w scenariuszu *arch1*, według wylosowanego przydzielenia do grupy z lub bez wizualizacji trasy (opracowanie własne)

Poniższy wykres (Rysunek 7.62) prezentuje medianę oraz średnią, kolejno dla czasu przejścia (kolor niebieski) oraz przebytego dystansu (kolor zielony) przez uczestników w każdej z grup.



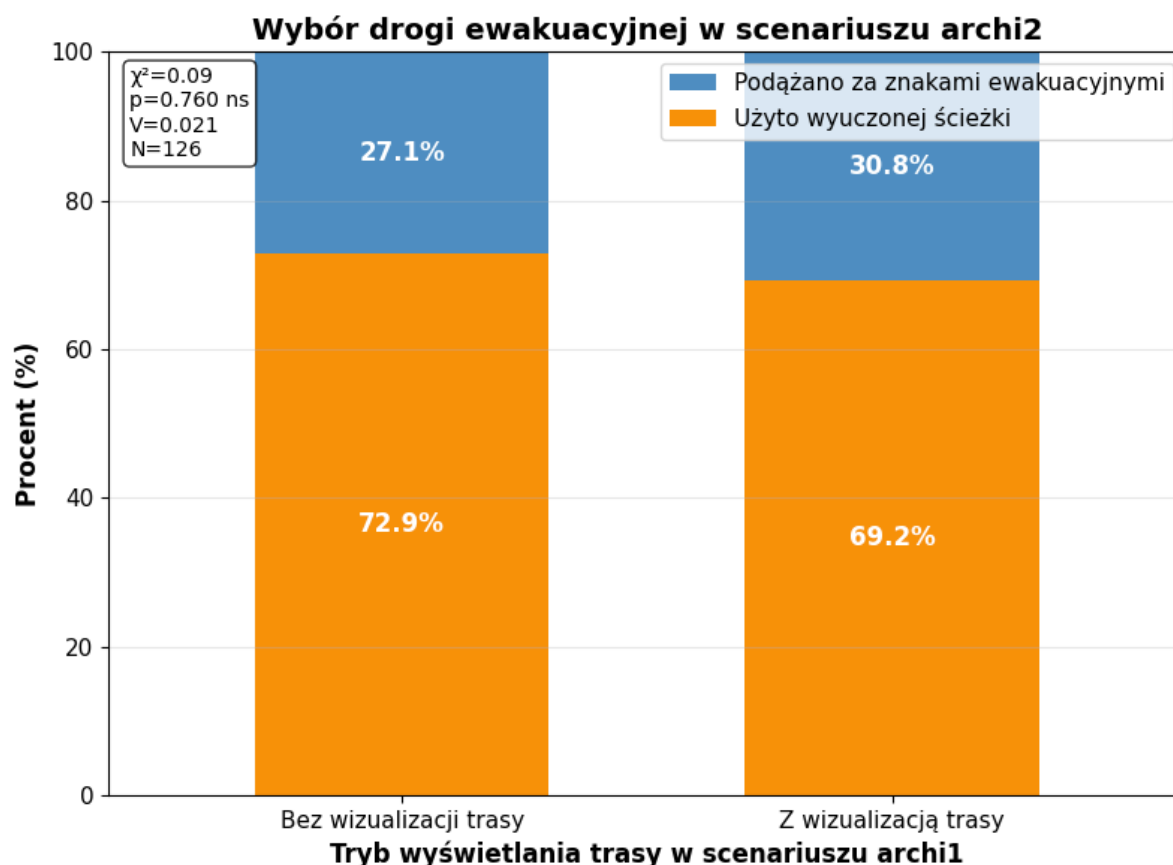
Rysunek 7.62. Czas oraz przebyty dystans w poszukiwaniu pomieszczenia w scenariuszu *archi1* (opracowanie własne)

Z perspektywy prowadzonego eksperymentu istotne było uzyskanie spodziewanego efektu krótszego kontaktu z mapą w grupie, która miała dostępną wizualizację trasy na mapie (Rysunek 7.63). Efekt został zaobserwowany (8% spadek mediany czasu kontaktu z mapą), ale nie jest on istotny statystycznie ($p = 0.67$). Jest również wyraźnie mniejszy niż w analogicznym eksperymencie nr 4. Zmniejszenie siły efektu mogło wynikać z różnicy w specyfice trasy, układzie budynku lub z efektu nabywania doświadczenia w korzystaniu z mapy i symulatora (scenariusze eksperymentu 7 były pod koniec całego badania). Warto również odnotować, że jedynie **7%** (8 spośród 101) uczestników w scenariuszu *archi1* w ogóle nie otworzyło mapy, co jest wartością znacznie niższą od średniej dla wszystkich scenariuszy wynoszącej około 30%.



Rysunek 7.63. Czas kontaktu z mapą w zależności od wizualizacji trasy w scenariuszu *archi1* (opracowanie własne)

Aby sprawdzić postawioną w eksperymencie hipotezę przeanalizowano jaki procent uczestników podążył za znakami ewakuacyjnymi, a jaki użył wyuczonej wcześniej ścieżki w scenariuszu *archi2*. Wyniki zgrupowano według pokazywania lub nie pokazywania wizualizacji trasy w scenariuszu *archi2* (Rysunek 7.64).



Rysunek 7.64. Wybór drogi ewakuacyjnej w scenariuszu archi2 w rozdzielaniu na prezentację trasy w scenariuszu archi1 (opracowanie własne)

W obu grupach za znakami ewakuacyjnymi podążyło jedynie około 30% badanych. Efekt wpływu prezentacji trasy jest minimalny, a różnica niecałych 4% nie wykazuje istotności statystycznej ($p = 0.76$).

Przeanalizowano również potencjalne korelacje z innymi mierzonymi zmiennymi dotyczącymi interfejsu symulatora lub cech demograficznych czy umiejętności uczestników. Żadna zmienna nie wykazała istotnego wpływu na wybór drogi ewakuacji.

Wnioski i podsumowanie eksperymentu

Przeprowadzony eksperyment nie potwierdził postawionej hipotezy, że wizualizacja trasy podczas pierwszego scenariusza nawigacyjnego wpłynie na wybór drogi ewakuacyjnej w sytuacji zagrożenia. Zarówno uczestnicy, którzy w scenariuszu *archi1* korzystali z mapy z zaznaczoną trasą, jak i ci, którzy planowali drogę samodzielnie, wykazywali podobne wzorce zachowań ewakuacyjnych. Ponadto żadna ze zbieranych w badaniach danych nie wykazała istotnej korelacji z wyborem drogi ewakuacji.

Eksperyment ujawnił jednak istotny i niepokojący fakt dotyczący efektywności oznakowania ewakuacyjnego. Jedynie 30% uczestników w obu grupach podążyło za znakami ewakuacyjnymi wskazującymi krótszą drogę ucieczki. Oznacza to, że aż 70% badanych zignorowało dostępne oznakowanie i wybrało dłuższą, ale znaną trasę, którą wcześniej dotarło do pomieszczenia docelowego. Ten wynik wskazuje na silną tendencję do preferowania wyuczonej ścieżki nawigacyjnej w sytuacji stresu, niezależnie od dostępności krótszej alternatywy. **Zaobserwowane zjawisko może mieć istotne implikacje praktyczne dla bezpieczeństwa w obiektach użyteczności publicznej.** Jeśli w warunkach symulowanych, bez rzeczywistego zagrożenia życia, większość uczestników ignoruje oznakowanie ewakuacyjne, można przypuszczać, że w sytuacji rzeczywistego pożaru lub innego zagrożenia tendencja ta może być jeszcze silniejsza ze względu na zwiększony poziom stresu i ograniczone zdolności poznawcze. Szczegółowa analiza tego zjawiska i potwierdzenie jego występowania w różnych sytuacjach i różnych budynkach to potencjalny kierunek dalszych badań. Warto również porównać inne sposoby prezentacji informacji o najkrótszej trasie ewakuacji, np. poprzez intensywniejsze oświetlenie znaków, większe rozmiary, dodatkowe oznakowanie podłogowe lub systemy dynamiczne reagujące na lokalizację zagrożenia. W miejscach szczególnie narażonych, gdzie wybór niewłaściwej drogi ewakuacji może mieć krytyczne konsekwencje, standardowe oznakowanie może okazać się niewystarczające.

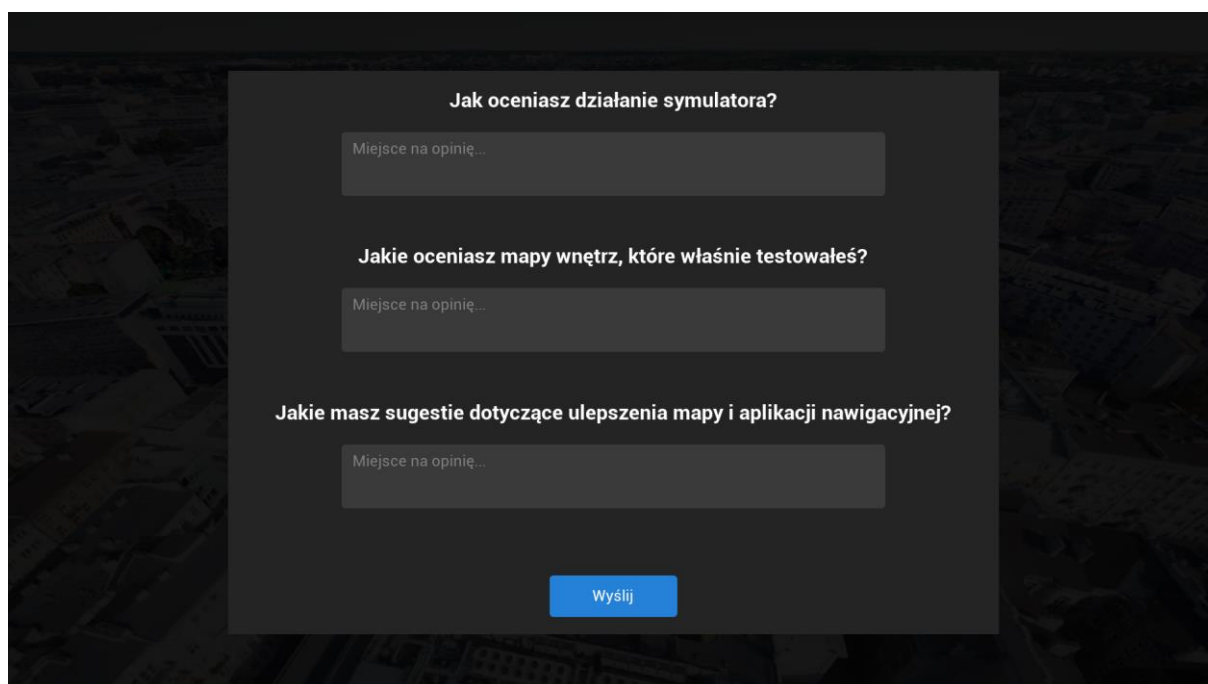
Warto również odnotować, że zmniejszenie siły efektu wizualizacji trasy w porównaniu z analogicznym eksperymentem nr 4, może wynikać z nabywania przez uczestników doświadczenia w korzystaniu z mapy i symulatora. Scenariusze eksperymentu 7 były realizowane pod koniec całego badania, co mogło wpłynąć na większą pewność uczestników w poruszaniu się po wirtualnym środowisku i mniejszą zależność od wizualizacji trasy.

7.6. Analiza dodatkowych swobodnych odpowiedzi użytkowników w badaniu ankietowym

Po ukończeniu wszystkich eksperymentów uczestnicy badania dostali możliwość jakościowej oceny różnych aspektów platformy badawczej, a także przedstawienia własnych sugestii w zakresie jej ulepszenia. Wygląd formularza zbierania opinii przedstawiono na rysunku poniżej (Rysunek 7.65).

Pytania w ankiecie zostały podzielone na trzy kategorie:

1. Ocena działania symulatora
2. Ocena map wewnątrz budynków
3. Sugestie ulepszeń



Jak oceniasz działanie symulatora?

Miejsce na opinię...

Jakie oceniasz mapy wewnątrz, które właśnie testowałeś?

Miejsce na opinię...

Jakie masz sugestie dotyczące ulepszenia mapy i aplikacji nawigacyjnej?

Miejsce na opinię...

Wyslij

Rysunek 7.65. Ankieta końcowa – formularz zbierania danych jakościowych wyświetlany na końcu procesu badawczego (opracowanie własne)

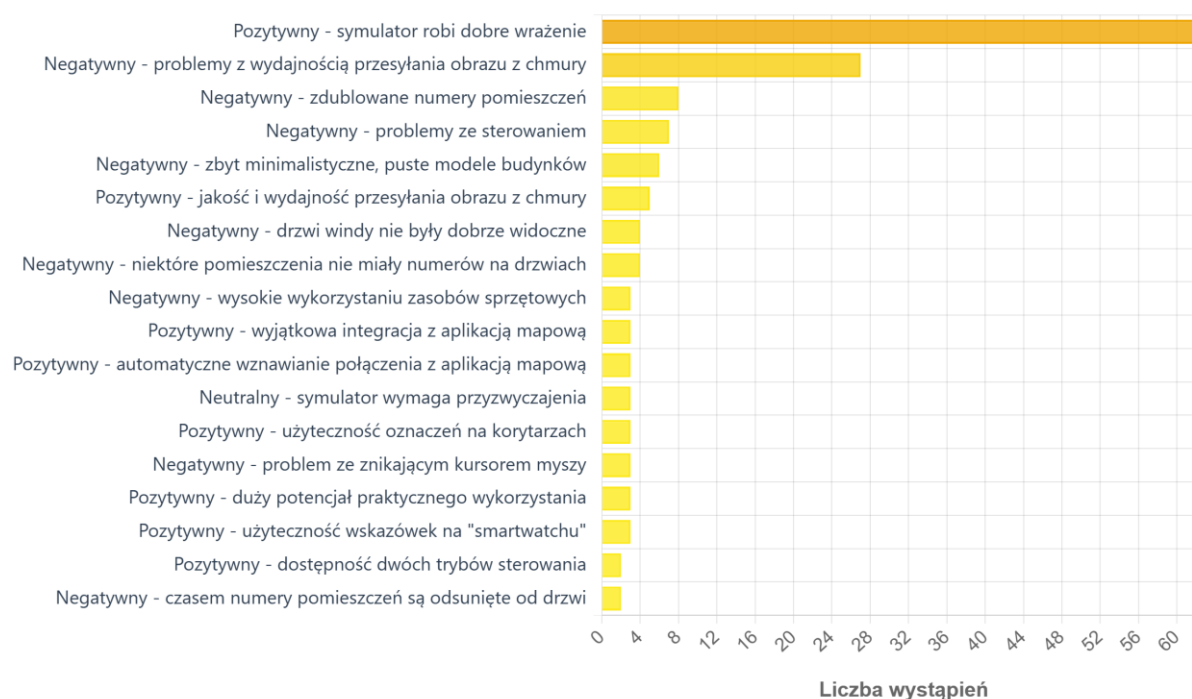
Analizę zebranych danych rozpoczęto od ręcznego nadania etykiet porządkujących każdej z przesłanych opinii. Etykiety reprezentowały zbliżone do siebie sugestie i pozwoliły na zliczenie tych, które powtarzały się pomiędzy uczestnikami.

Spośród 127 osób, które ukończyły badanie 109 zdecydowało się na pozostawienie choćby minimalnej opinii opisowej.

Utworzono 38 etykiet, dla tych rodzajów uwag i propozycji, które powtórzyły się przynajmniej dwukrotnie.

Na rysunkach poniżej (Rysunek 7.66, Rysunek 7.67, Rysunek 7.68) przedstawiono wykresy prezentujące częstotliwość występowania poszczególnych uwag w podziale na kategorie. Każda z sugestii i opinii została także krótko skomentowana.

7.6.1. Ocena ogólna działania symulatora



Rysunek 7.66. Częstotliwość występowania poszczególnych etykiet w kategorii „Ocena działania symulatora” (opracowanie własne)

Pozytywny - symulator robi dobre wrażenie (63 wystąpienia)

Znaczna większość uczestników wyraziła pozytywną ogólną opinię o symulatorze.

Negatywny - problemy z wydajnością przesyłania obrazu z chmury (27 wystąpień)

Okolo 25% uczestników zgłosiła częściowe problemy z działaniem symulatora w trybie przesyłania obrazu z chmury. Część z tych problemów mogła wynikać z niskiej jakości połączenia po stronie uczestnika, a część np. z chwilowego obciążenia klastra obliczeniowego. Jednym ze zgłoszonych sposobów rozwiązania tego problemu jest przeprowadzenie testu jakości i stabilności połączenia przed rozpoczęciem badania.

Negatywny - zdublowane numery pomieszczeń (8 wystąpień)

Problem wynikał z samego układu poszczególnych budynków, w rzeczywistości również do niektórych pomieszczeń prowadzi dwoje drzwi. Na potrzeby badania celem scenariusza nie powinny być takie pokoje, aby nie wprowadzać dodatkowej dezorientacji, mogącej zaburzać wyniki.

Negatywny - problemy ze sterowaniem (7 wystąpień)

Problem dotyczył zwłaszcza uproszczonego trybu sterowania, w którym w rzadkich przypadkach wskazanie miejsca docelowego prowadziło do niespodziewanych rezultatów. Zalecane jest dopracowanie tego sposobu sterowania. Ponadto uczestnicy zgłaszali potrzebę łatwej możliwości przerywania trwającego ruchu, w przypadku nieumyślnego kliknięcia.

Negatywny - zbyt minimalistyczne, puste modele budynków (6 wystąpień)

Zgłoszono uwagi dotyczące nadmiernej prostoty wizualizacji wnętrza budynków, która mogła wpływać na poczucie realizmu i immersji. Brak detali architektonicznych oraz wyposażenia pomieszczeń mógł utrudniać orientację przestrzenną oraz zapamiętywanie charakterystycznych punktów odniesienia. W przyszłych iteracjach symulatora zaleca się wzbogacenie modeli o elementy wyposażenia oraz detale architektoniczne, zachowując jednak równowagę między realizmem a wydajnością renderowania. Należy zwrócić uwagę, że liczba tych opinii stanowiła niewielki procent w stosunku do wszystkich opinii.

Pozytywny - jakość i wydajność przesyłania obrazu z chmury (5 wystąpień)

Uczestnicy korzystający z trybu przesyłania obrazu z chmury ocenili pozytywnie jakość oraz płynność otrzymywanego strumienia wideo.

Negatywny - drzwi windy nie były dobrze widoczne (4 wystąpienia)

Zgłoszono trudności z lokalizacją drzwi windy w symulowanym środowisku. Problem ten mógł wynikać z niewystarczającego kontrastu między materiałem drzwi windy, a materiałem otaczających ścian, a także z podobieństwa drzwi windy do wszystkich pozostałych drzwi do pomieszczeń. Zaleca się wprowadzenie wyraźniejszego oznakowania wind oraz zwiększenie kontrastu tekstur, aby ułatwić ich identyfikację w przestrzeni wirtualnej.

Negatywny - niektóre pomieszczenia nie miały numerów na drzwiach (4 wystąpienia)

Zgłoszono braki w oznaczeniu drzwi numerami. Był to znany problem wynikający z niepełnej spójności danych modeli 3D oraz danych GIS, zawierających w sobie numerację pomieszczeń. Rozwiązaniem problemu jest wprowadzeniem poprawek w danych źródłowych.

Negatywny - wysokie wykorzystaniu zasobów sprzętowych (3 wystąpienia)

Zgłoszono znaczne obciążenie sprzętu komputerowego podczas użytkowania symulatora w trybie lokalnym. Problem ten mógł wpływać na płynność działania aplikacji oraz komfort pracy, szczególnie na urządzeniach o niższej specyfikacji. Optymalizacja symulatora została dopasowana tak, aby maksymalnie wykorzystać możliwości sprzętu. Zalecanym rozwiązaniem jest umożliwienie wyboru niższej jakości renderowania, co zmniejszy obciążenie komputera.

Pozytywny - wyjątkowa integracja z aplikacją mapową (3 wystąpienia)

Doceniono niestandardowe rozwiązanie umożliwiające synchronizację symulatora z aplikacją mapową w czasie rzeczywistym. Rozwiązanie to wyróżnia się innowacyjnością i może stanowić istotny wkład do badań nad jakością aplikacji nawigacyjnych.

Pozytywny - automatyczne wznowianie połączenia z aplikacją mapową (3 wystąpienia)

Doceniono mechanizm automatycznego przywracania połączenia między symulatorem a aplikacją mapową w przypadku jego przerwania. Funkcja ta zwiększa niezawodność systemu oraz eliminuje konieczność ręcznej interwencji użytkownika, co przekłada się na płynność prowadzenia badań oraz komfort użytkowania.

Neutralny - symulator wymaga przyzwyczajenia (3 wystąpienia)

Zgłoszono potrzebę przyzwyczajenia do środowiska symulatora oraz jego mechanizmów sterowania. Obserwacja ta jest typowa dla narzędzi wykorzystujących środowiska wirtualne i nie wskazuje na istotne problemy z użytecznością, lecz podkreśla znaczenie fazy zapoznawczej oraz szkolenia użytkowników przed właściwym badaniem.

Pozytywny - użyteczność oznaczeń na korytarzach (3 wystąpienia)

Doceniono czytelność i wsparcie w nawigacji w postaci oznaczeń umieszczonych na ścianach korytarzy w budynku Gmachu Elektrotechniki.

Negatywny - problem ze znikającym kursorem myszy (3 wystąpienia)

Zgłoszono incydentalne przypadki znikania kursora myszy podczas użytkowania symulatora, co utrudniało interakcję z interfejsem. Problem ten mógł wynikać z konfliktów w zarządzaniu stanem kursora, które było skomplikowane z uwagi na konieczność obsługi różnych trybów sterowania oraz wymiany informacji o stanie kursora z przeglądarką internetową w trybie chmurowym. Pomimo wielokrotnego ulepszania obsługi zarządzania kursorem nie udało się uniknąć wszystkich problemów.

Pozytywny - duży potencjał praktycznego wykorzystania (3 wystąpienia)

Podkreślono możliwości zastosowania symulatora w praktyce np. by przygotować się na wizytę w budynku.

Pozytywny - użyteczność wskazówek na "smartwatchu" (3 wystąpienia)

Doceniono funkcjonalność wyświetlania wskazówek nawigacyjnych na wirtualnym urządzeniu typu smartwatch w lewym górnym rogu ekranu symulatora. Opinie wskazywały na dużą wygodę tego rozwiązania zastosowanego w scenariuszu *gg3*. Rozwiązanie to pozwala na szybki dostęp do istotnych danych bez konieczności otwierania widoku mapy.

Pozytywny - dostępność dwóch trybów sterowania (2 wystąpienia)

Pozytywnie oceniono możliwość wyboru między dwoma trybami sterowania, co umożliwiło dostosowanie interakcji do indywidualnych preferencji oraz poziomu zaawansowania użytkowników.

Negatywny - czasem numery pomieszczeń są odsunięte od drzwi (2 wystąpienia)

Zgłoszono przypadki, w których tabliczki z numeracją pomieszczeń znajdowały się w pewnej odległości od drzwi, co wynikało z niedoskonałości algorytmu wyliczającego odpowiednią pozycję umiejscowienia numeru. Konieczne jest wprowadzenie poprawek w algorytmie i ponowne wygenerowanie numeracji drzwi w modelach dotkniętych tym problemem.

7.6.2. Ocena map wewnątrz budynków



Rysunek 7.67. Częstotliwość występowania poszczególnych etykiet w kategorii „Ocena map wewnątrz budynków” (opracowanie własne)

Pozytywny - mapa jest użyteczna (42 wystąpień)

Doceniono funkcjonalność mapy jako narzędzia wspierającego orientację w budynkach.

Pozytywny - wygląd mapy (24 wystąpień)

Pozytywnie oceniono aspekty wizualne interfejsu mapowego, w tym czytelność symboli, dobór kolorystyki oraz ogólną estetykę prezentacji danych przestrzennych.

Pozytywny - mapa dobrze odwzorowuje rzeczywistość (16 wystąpień)

Często podkreślono zgodność danych o budynkach wyświetlanych na mapie z faktycznym układem przestrzennym symulowanych budynków.

Negatywny - konieczność chowania mapy podczas poruszania (7 wystąpień)

Zgłoszono niewygodę związaną z koniecznością ukrywania interfejsu mapy w trakcie nawigacji po środowisku wirtualnym. Wymuszenie zatrzymywania było jednak przemyślaną decyzją metodyczną, mającą podnieść poziom odwzorowania rzeczywistych warunków korzystania z mapy (nie trzymamy podczas ruchu pieszego w budynku aplikacji cały czas przed oczami).

Neutralny - mapa w budynkach nie jest potrzebna (5 wystąpień)

Część opinii wskazywała na ograniczoną przydatność mapy podczas nawigacji wewnątrz budynków. Obserwacja ta sugeruje, że potrzeba korzystania z mapy może być zależna od złożoności struktury przestrzennej oraz indywidualnych preferencji użytkowników.

Neutralny - do map trzeba się przyzwyczaić (2 wystąpienia)

Zgłoszono potrzebę okresu adaptacji do sposobu prezentacji informacji przestrzennych. Choć potrzeba przyzwyczajenia jest naturalne, to wskazuje, że z jednej strony prezentowane mapy mogą nie być autoczytelne, a z drugiej, że wraz z nabywaniem doświadczenia mogą zmieniać się uzyskiwane wyniki dotyczące skuteczności mapy we wspomaganiu nawigacji.

Negatywny - zbyt duży poziom przybliżenia (2 wystąpienia)

Uwaga zgłaszana przez uczestników, którzy nie byli świadomi, iż poziom przybliżenia mapy był częścią eksperymentu (patrz rozdział 7.5.1).

7.6.3. Sugestie ulepszeń



Rysunek 7.68. Częstotliwość występowania poszczególnych etykiet w kategorii „Sugestie ulepszeń” (opracowanie własne)

Poprawka - lepsze oznaczenie wind i klatek schodowych (18 wystąpień)

Zgłoszono potrzebę wyraźniejszego wizualizowania elementów komunikacji pionowej na mapach budynków. Ulepszenie oznakowania wind oraz klatek schodowych, poprzez zwiększenie kontrastu, wprowadzenie charakterystycznych symboli lub dodanie podświetleń, mogłoby znacząco ułatwić ich lokalizację oraz poprawić efektywność nawigacji między piętrami budynku.

Poprawka - lepsze oznaczenie wyznaczonej trasy na różnych piętrach (9 wystąpień)

Wskazano na trudności z interpretacją wyznaczonej trasy w przypadku, gdy obejmowała ona przemieszczanie się między poziomami budynku. Sugerowane usprawnienie obejmuje implementację bardziej czytelnego oznaczenia trasy oraz wyraźne wskazanie miejsc zmiany piętra, co zwiększyłoby zrozumiałość wytyczonych ścieżek nawigacyjnych.

Dodatek - wyznaczanie trasy w czasie rzeczywistym (5 wystąpień)

Zgłoszono zapotrzebowanie na funkcjonalność dynamicznego ponownego wyznaczania trasy w przypadku zboczenia z jej pierwotnej wersji.

Dodatek - możliwość zmiany czułości myszy (4 wystąpienia)

Sugestia zgłaszana przez osoby, które nie zauważyły, że udostępniono możliwość zmiany czułości myszy w menu pauzy, o czym poinformowano na początku samouczka. Jednym ze sposobów rozwiązania tego problemu byłoby wymuszenie przetestowania wszystkich funkcji prezentowanych w samouczku przed rozpoczęciem badania.

Poprawka - lepsze oznaczenie początku i końca trasy (4 wystąpienia)

Zgłoszono potrzebę wyraźniejszego wizualizowania punktów startowych i docelowych wyznaczonych tras nawigacyjnych, umożliwiającego ich jednoznaczne rozróżnienie. Zastosowane oznaczenie jedynie kolorem zielonym oraz czerwonym nie było jasne dla wszystkich uczestników.

Dodatek - oznaczenie już przebytej części trasy (4 wystąpienia)

Zaproponowano wprowadzenie wizualnego rozróżnienia między odcinkami trasy już pokonanymi a tymi, które pozostają do przebycia. Funkcjonalność ta mogłaby przyjąć formę zmiany koloru lub przezroczystości odcinków ścieżki, zapewniając użytkownikowi ciągłą informację zwrotną o postępie w realizacji zadania nawigacyjnego.

Dodatek - automatyczna rotacja mapy (3 wystąpienia)

Sugestia zgłaszana przez uczestników, którzy nie byli świadomi, że zmiana sposobu rotacji mapy była częścią eksperymentu. Co ciekawe jeden uczestnik zgłosił zapotrzebowanie przeciwne, aby mapa zawsze była skierowana w kierunku północy.

Dodatek - zapobieganie automatycznej blokadzie telefonu (3 wystąpienia)

Wskazano na problem automatycznego wygaszania ekranu urządzenia mobilnego podczas korzystania z aplikacji nawigacyjnej towarzyszącej symulatorowi. Wygaszenie ekranu powodowało zerwanie połączenia z symulatorem i konieczność chwilowego przerwania badania, aby wybudzić telefon. Niedogodność ta nie została wykryta na etapie testów i powinna być rozwiązana np. poprzez zapobieganie wygaszeniom ekranu podczas korzystania z aplikacji nawigacyjnej.

Dodatek - możliwość biegania (2 wystąpienia)

Zgłoszono zapotrzebowanie na wprowadzenie dodatkowego trybu przemieszczania się o zwiększonej prędkości. Tryb taki istniał w rozwojowej wersji symulatora, ale został wyłączony, aby zapewnić spójność pomiaru czasu wykonania zadań przez wszystkich uczestników.

Dodatek - dostępność trybu obniżonej jakości dla lepszej wydajności (2 wystąpienia)

Zaproponowano wprowadzenie opcji redukcji szczegółowości wizualizacji w celu poprawy płynności działania symulatora na urządzeniach o ograniczonych zasobach sprzętowych. To dobre rozwiązanie, na sytuacje, w których automatyczne dopasowywanie jakości nie będzie wystarczające.

Poprawka - dopracowanie samouczka (2 wystąpienia)

Zgłoszono potrzebę udoskonalenia materiałów wprowadzających, tak aby miały jeszcze bardziej formę interaktywnego szkolenia z obsługi symulatora. Uczestnicy nie zawsze czytali komunikaty wyświetlane w samouczku.

Poprawka - pokazywania numerów pomieszczeń przy oddalonej mapie (2 wystąpienia)

Wskazano na problem z brakiem etykiet numeracji pomieszczeń podczas wyświetlania mapy w małej skali. Zalecane usprawnienie obejmuje implementację dynamicznej generalizacji kartograficznej, która zachowałaby widoczność kluczowych oznaczeń niezależnie od poziomu przybliżenia, lub wprowadzenie adaptacyjnej wielkości czcionek etykiet.

Dodatek - informacja / wibracja, w przypadku zboczenia z trasy (2 wystąpienia)

Zaproponowano wprowadzenie mechanizmów informacji zwrotnej sygnalizujących odchylenie od wyznaczonej trasy nawigacyjnej. Implementacja powiadomień wizualnych,

dźwiękowych lub haptycznych (w przypadku aplikacji mobilnej) mogłaby zwiększyć świadomość błędów nawigacyjnych oraz ułatwić powrót na właściwą ścieżkę.

7.7. Podsumowanie wniosków oraz rekomendacje dla twórców aplikacji nawigacyjnych wynikające z przeprowadzonych testów

W niniejszym rozdziale podsumowano uzyskane dane ze wszystkich eksperymentów, które zostały przeprowadzone podczas badań z wykorzystaniem zaprojektowanej platformy badawczej.

Poniżej przedstawiono najważniejsze wnioski, jakie wynikły z analizy danych:

1. Korzystanie z mapy oraz znajomość budynku

Okolo 32% uczestników w ogóle nie wyświetliło mapy podczas poszukiwania celu, preferując samodzielną eksplorację budynku. W grupie osób znających budynek wskaźnik niekorzystających z mapy wzrósł do 44%, podczas gdy w grupie nieznających budynku wynosił 25%. Osoby znające budynek, które zdecydowały się na użycie mapy, korzystały z niej rzadziej (średnio 2,5 otwarcia mapy na trasę) w porównaniu z osobami nieznającymi budynku (średnio 3,5 otwarcia mapy na trasę).

2. Wpływ czynników demograficznych

Badania wykazały istotne statystycznie różnice w efektywności nawigacji w zależności od charakterystyk uczestników. Mężczyźni byli statystycznie szybsi od kobiet, podobnie jak osoby młodsze w porównaniu do starszych. Wiek oraz zadeklarowana zdolność orientacji w przestrzeni okazały się najbardziej istotnymi statystycznie czynnikami wpływającymi na skłonność do korzystania z mapy.

3. Wpływ doświadczenia w grach

Doświadczenie w grach wideo oraz wybór trybu poruszania okazały się kluczowymi czynnikami wpływającymi na wydajność nawigacji w środowisku wirtualnym. Osoby z większym doświadczeniem w grach poruszały się sprawniej w przestrzeni wirtualnej. Zaobserwowano

silną korelację między płcią a doświadczeniem w grach, sugerującą znaczące różnice w doświadczeniu z grami komputerowymi między płciami - mężczyźni częściej grają w gry.

4. Orientacja przestrzenna

Osoby z lepszą orientacją przestrzenną poruszały się szybciej i potrzebowały mniej czasu na ukończenie zadania. Dodatnia korelacja między orientacją przestrzenną a doświadczeniem w grach oraz korzystaniem z aplikacji nawigacyjnych potwierdza związek między różnymi aspektami zdolności przestrzennych.

5. Wpływ wsparcia nawigacyjnego na charakter interakcji użytkownika z mapą

Grupy korzystające z bardziej zaawansowanych form wsparcia nawigacyjnego (mapa, trasa, pozycja, kierunek, strzałki nawigacyjne) wykazywały krótszy czas kontaktu z mapą, co wskazuje na większą efektywność przetwarzania informacji. Użytkownicy potrzebowali mniej czasu na zrozumienie sytuacji przestrzennej i podjęcie decyzji nawigacyjnych. Dobrze zaprojektowane wsparcie nawigacyjne nie zwiększa zależności od urządzenia, lecz optymalizuje proces podejmowania decyzji.

6. Podążanie za znakami ewakuacyjnymi

Jedynie 30% uczestników podążało za znakami ewakuacyjnymi wskazującymi krótszą drogę ucieczki, podczas gdy 70% wybrało dłuższą, ale znaną trasę. To niepokojące odkrycie pokazuje, że ludzie w sytuacjach zagrożenia preferują znane sobie ścieżki, i nie są skłonni do wykorzystania oznaczeń ewakuacyjnych.

7. Wpływ na pamięć przestrzenną

Mapy zawierające wizualizację trasy przejścia zwiększają efektywność nawigacji w momencie ich używania, ale mogą jednocześnie hamować rozwój umiejętności orientacji przestrzennej i zapamiętywania tras. Użytkownicy stale polegający na mapach mogą nie wykształcić trwałej wiedzy przestrzennej o budynku.

Na podstawie wszystkich zebranych danych, wniosków oraz sugestii uczestników badania przygotowano również listę rekomendacji dla twórców aplikacji przeznaczonych do nawigacji wewnątrz budynków:

1. Wizualizacja trasy

Systemy nawigacji wewnątrz budynków powinny priorytetowo implementować wizualizację wyznaczonej trasy. Funkcja ta przynosi największe korzyści przy stosunkowo niewielkim zwiększeniu złożoności interfejsu. Trasa eliminuje konieczność samodzielnego planowania drogi przez użytkownika, co znacząco redukuje obciążenie poznawcze i minimalizuje ryzyko błędnych decyzji nawigacyjnych. Wyświetlana trasa zawęża obszar zainteresowania osoby poruszającej się po budynku do bliskich okolic trasy, dzięki czemu orientacja na mapie oraz w przestrzeni jest znacznie szybsza i mniej obciążająca.

2. Szeroki kontekst przestrzenny

System nawigacyjny powinien domyślnie prezentować mapę budynku z wystarczająco szerokim kontekstem, aby użytkownik mógł widzieć nie tylko bezpośrednie otoczenie miejsca w którym się znajduje, ale również fragment trasy przed sobą oraz kluczowe punkty orientacyjne. Zbyt silne przybliżenie mapy, w praktyce utrudnia orientację i zwiększa ryzyko dezorientacji. Alternatywnie, system powinien umożliwiać dynamiczne dostosowywanie poziomu przybliżenia w zależności od kontekstu - na przykład automatyczne powiększanie/pomniejszenie widoku w miejscach wymagających szczególnej uwagi.

3. Oznaczenie lokalizacji użytkownika

Oznaczenie aktualnej lokalizacji, choć mniej krytyczne niż wizualizacja trasy, stanowi istotne uzupełnienie, szczególnie w złożonych strukturach architektonicznych z wieloma podobnymi korytarzami i rozgałęzieniami. Połączenie wizualizacji trasy z ciągłym monitorowaniem własnej pozycji umożliwia użytkownikom bieżącą weryfikację postępów i szybką identyfikację ewentualnych odchyłeń od planowanej trasy, zwiększając pewność podejmowanych decyzji nawigacyjnych.

Jednocześnie badania wykazały sens udostępniania aplikacji nawigacyjnej również w budynkach, które nie mają infrastruktury umożliwiającej lokalizację. Sama prezentacja trasy jest już istotnym ułatwieniem nawigacji po budynku.

4. Dynamiczne ponowne wyznaczanie trasy

Aplikacja powinna oferować automatyczne ponowne wyznaczenie trasy w przypadku oddalenia się od tej pierwotnie wyznaczonej. Warto rozważyć wprowadzenie powiadomień wizualnych, dźwiękowych lub haptycznych sygnalizujących odchylenie od wyznaczonej trasy. Funkcja ta jest kluczowa dla zapobiegania sytuacjom całkowitej dezorientacji.

5. Uprozczone wskazówki kierunkowe

Dodatkowe wskazówki w formie strzałek kierunkowych mogą w znacznym stopniu zastąpić tradycyjną mapę, zmniejszając obciążenie poznawcze. Rozwiązanie to ma szczególne zastosowanie w urządzeniach typu smartwatch, gdzie wyświetlenie mapy może nie być efektywne. W eksperymencie z wskazówkami ponad połowa uczestników w ogóle nie potrzebowała mapy do dotarcia do celu.

6. Wizualizacja elementów komunikacji pionowej

Wymagane jest wyraźne wizualizowanie wind i klatek schodowych na mapach budynków. Ulepszenie oznakowania poprzez zwiększenie kontrastu, wprowadzenie charakterystycznych symboli lub dodanie podświetleń mogłoby znacząco ułatwić ich lokalizację oraz poprawić efektywność nawigacji między piętrami.

7. Oznaczenie tras wielopoziomowych

Implementacja bardziej czytelnego oznaczenia trasy na różnych piętrach z wyraźnym wskazaniem miejsc zmiany piętra zwiększyłoby zrozumiałość wyznaczonych ścieżek nawigacyjnych. Uczestnicy wskazywali na trudności z interpretacją wyznaczonej trasy w przypadku przemieszczania się między różnymi piętrami budynku.

8. Punkty startowe i docelowe

Wymagane jest wyraźne oznaczanie punktów startowych i docelowych tras nawigacyjnych. Zastosowane oznaczenie jedynie kolorem zielonym oraz czerwonym nie było jasne dla wszystkich uczestników. Należy wprowadzić jednoznaczne rozróżnienie tych punktów za pomocą charakterystycznych symboli lub ikon.

9. Wizualizacja postępu na trasie

Funkcjonalność wizualnego rozróżnienia między odcinkami trasy już pokonanymi a tymi do przebycia zapewniłaby użytkownikowi ciągłą informację zwrotną o postępie. Może to przybrać formę zmiany koloru lub przezroczystości odcinków ścieżki, zwiększając pewność użytkownika co do prawidłowości obranej drogi.

10. Adaptacyjna generalizacja kartograficzna

Uczestnicy wskazywali na problematyczność braku etykiet numeracji pomieszczeń podczas wyświetlania mapy w małej skali. Zalecana jest implementacja dynamicznej generalizacji kartograficznej zachowującej widoczność kluczowych oznaczeń niezależnie od poziomu przybliżenia.

11. Wsparcie dla oznakowania fizycznego

W budynkach warto rozważyć integrację aplikacji z istniejącym oznakowaniem fizycznym. Dodatkowe oznaczenia na ścianach korytarzy (np. informacje o zakresach numerów pomieszczeń w poszczególnych segmentach) zostały docenione przez uczestników jako użyteczne wsparcie nawigacyjne. W budynkach z dobrym oznakowaniem fizycznym korzyści z udostępnienia mapy cyfrowej mogą być znikome.

12. Systemy ewakuacyjne

W kontekście ewakuacji standardowe oznakowanie może okazać się niewystarczające. Warto porównać inne sposoby prezentacji informacji o najkrótszej trasie ewakuacji: intensywniejsze oświetlenie znaków, większe rozmiary, dodatkowe oznakowanie podłogowe lub systemy dynamiczne reagujące na lokalizację zagrożenia. Aplikacje nawigacyjne mogłyby odgrywać rolę w sytuacjach ewakuacyjnych, kierując użytkowników najbezpieczniejszymi trasami, ale jednocześnie taka funkcjonalność musiałaby działać błyskawicznie i niezawodnie oraz mieć minimalny, bardzo prosty interfejs, aby nie tracić cennego czasu na ewakuację.

13. Uwzględnienie pamięci przestrzennej

Należy uwzględnić istotny dylemat projektowy dotyczący wpływu na pamięć przestrzenną człowieka. Warto rozważyć wprowadzenie trybów adaptacyjnych, które stopniowo redukują ilość wsparcia nawigacyjnego dla regularnych użytkowników danego budynku, zachęcając do rozwoju umiejętności orientacji przestrzennej. Jest to potencjalny kierunek dla dalszych badań

dotyczących metod przekazu informacji przestrzennej minimalizujących negatywny wpływ na rozwój wiedzy przestrzennej.

Większość powyższych rekomendacji i obserwacji ma charakter uniwersalny, a przedstawione zalecenia są już standardem projektowym w aplikacjach do nawigacji na zewnątrz budynków i zostały potwierdzone poprzez przeprowadzone badania. Na szczególną uwagę zasługują wnioski nr 6, 7, 11 oraz 12, które dotyczą się w szczególności wnętrza budynków. Są to nowe obserwacje, które mogą mieć znaczenie praktyczne.

8. Ewaluacja opracowanego narzędzia

Celem niniejszego rozdziału jest całościowa ocena przygotowanego środowiska symulacyjnego jako narzędzia badawczego do testowania rozwiązań kartograficznych i aplikacji przeznaczonych do nawigacji wewnątrz budynków. Ewaluacja obejmuje zarówno weryfikację założeń metodologicznych, analizę aspektów technicznych, jak i ocenę narzędzia z perspektywy użytkowników biorących udział w eksperymentach. Rozdział syntetyzuje wyniki badania na użytkownikach oraz analizę danych ilościowych i jakościowych zebranych podczas realizacji wszystkich scenariuszy testowych.

Ewaluację narzędzia oparto na zestawie kryteriów wynikających z celów szczegółowych rozprawy oraz przeglądu literatury dotyczącej metod badania map i aplikacji nawigacyjnych. Za kluczowe uznano następujące obszary oceny: skuteczność badawczą, użyteczność i ergonomię, wiarygodność i realizm środowiska, powtarzalność i kontrolowalność eksperymentów, skalowalność i elastyczność oraz aspekty techniczne związane z wydajnością i stabilnością. Dodatkowo uwzględniono ograniczenia narzędzia oraz potencjalne źródła błędów, które mogą wpływać na interpretację wyników eksperymentów.

Skuteczność badawcza i zgodność z metodyką

Jednym z podstawowych kryteriów była odpowiedź na pytanie, czy przygotowane środowisko pozwala na realizację założonej metodyki badawczej opisanej w rozdziale piątym. Symulator umożliwił przeprowadzenie złożonego procesu badawczego, obejmującego poziom treningowy, serię siedmiu eksperymentów z kontrolowaną randomizacją zmiennych niezależnych oraz ankiety wstępne i końcowe. Wszystkie zaplanowane scenariusze testowe zostały poprawnie zaimplementowane, a ich realizacja umożliwiła zebranie danych ilościowych i jakościowych odpowiadających na postawione pytania badawcze. Zaimplementowany mechanizm rejestrowania danych pozwolił na zbiór szczegółowych informacji o przebytych trasach, zachowaniach nawigacyjnych i interakcjach z mapą, co potwierdza, że narzędzie spełniło założoną funkcję.

Ewaluacja techniczna i wydajnościowa

W rozdziale szóstym przedstawiono szczegółowe wyniki profilowania oraz optymalizacji środowiska, które zapewniły płynność symulacji na typowych komputerach użytkowników. Zarówno wariant przesyłania obrazu z chmury, jak i wariant lokalnego uruchamiania aplikacji na komputerze uczestnika okazały się możliwe do wykorzystania praktycznego, przy czym każdy miał swoje zalety i wady. W przypadku pracy na komputerze lokalnym uczestnicy zgłaszali wysokie wykorzystanie zasobów obliczeniowych komputera, a w wariantcie przesyłania obrazu z chmury wielu uczestników doświadczyło niedogodności w postaci opóźnienia, niskiej jakości obrazu lub zacinania się całej symulacji. Mimo to technologia jest obiecująca i wraz z jej dopracowaniem może udać się wyeliminować tego typu problemy.

Komunikacja pomiędzy symulatorem a aplikacją mapową, oparta na serwerze udostępniającym interfejs API, działała stabilnie i umożliwiała dwukierunkową wymianę danych w czasie zbliżonym do rzeczywistego.

Powtarzalność, kontrolowalność i skalowalność eksperymentów

Opracowane środowisko spełniło kluczowe założenie dotyczące powtarzalności i kontrolowalności eksperymentów. Zastosowanie silnika gier oraz cyfrowych modeli budynków umożliwiło wielokrotne odtwarzanie identycznych warunków testowych. Kontrolowana randomizacja zmiennych niezależnych, takich jak poziom przybliżenia mapy, tryb wsparcia nawigacyjnego czy sposób prezentacji kierunku, została z sukcesem w pełni zautomatyzowana.

Z punktu widzenia skalowalności, architektura projektu pozwala na stosunkowo łatwą integrację kolejnych modeli budynków oraz rozszerzanie zestawu scenariuszy testowych, bez konieczności zasadniczej przebudowy środowiska.

Ocena z perspektywy uczestników badania

Analiza odpowiedzi w ankiecie końcowej wykazała, że większość uczestników oceniła działanie symulatora pozytywnie, zwracając uwagę na wysoki poziom immersji i możliwość wygodnego trenowania nawigacji w złożonych budynkach.

W strukturze odpowiedzi jakościowych pojawiały się zarówno etykiety wskazujące na intuicyjność sterowania i czytelność prezentacji przestrzeni, jak i uwagi dotyczące

przypadkowych problemów technicznych czy sugestie dotyczące konkretnych ulepszeń zarówno samego symulatora jak i mapy oraz aplikacji nawigacyjnej. Wyniki ankiety potwierdzają, że narzędzie zostało dobrze przyjęte przez użytkowników i może być traktowane jako wygodna platforma do dalszych badań z zakresu nawigacji wewnątrz budynków.

Identyfikacja ograniczeń i źródeł błędów

Ewaluacja ujawniła szereg ograniczeń, które należy uwzględnić przy interpretacji uzyskanych wyników. Badanie przeprowadzono głównie na uczestnikach o zbliżonym profilu demograficznym, co może ograniczać możliwość uogólniania wyników na inne populacje użytkowników. Mimo przeprowadzonego profilowania, działanie środowiska mogło różnić się w zależności od konfiguracji sprzętowej i sieciowej sprzętu komputerowego uczestników badania. Dodatkowym źródłem potencjalnych błędów są różnice indywidualne w zakresie doświadczenia z grami komputerowymi, które mogły wpływać zarówno na komfort poruszania się w środowisku, jak i na interpretację wskazówek nawigacyjnych.

Podsumowując, opracowane środowisko symulacyjne spełniło założone funkcje badawcze i umożliwiło realizację założonego, przykładowego programu eksperymentów dotyczących nawigacji wewnątrz budynków. Narzędzie okazało się skuteczne w badaniu wpływu różnych metod prezentacji informacji kartograficznej na efektywność nawigacji, zapamiętywanie przestrzeni oraz wybór strategii poruszania się. Z perspektywy technicznej i użytkowej środowisko można uznać za wykonane na poziomie gotowości technologicznej TLR 9, choć wciąż pozostaje przestrzeń do optymalizacji wydajności oraz dalszego rozwoju.

Zidentyfikowane ograniczenia pozwoliły na wskazanie kierunków i zakresu przyszłych badań oraz możliwe usprawnienia narzędzia w kolejnych iteracjach rozwoju.

9. Podsumowanie i wnioski końcowe

Głównym celem rozprawy było opracowanie metodyki badawczej wykorzystującej środowisko symulacyjne i cyfrowe modele budynków do wieloaspektowego testowania metod przekazu kartograficznego oraz funkcjonalności aplikacji nawigacyjnych wewnątrz budynków.

Zrealizowany program prac, obejmujący projekt, implementację i ewaluację środowiska badawczego oraz badanie na użytkownikach, pozwala stwierdzić, że cel ten został osiągnięty. Zaprojektowano i wdrożono uniwersalne środowisko symulacyjne (informatyczną platformę badawczą), zintegrowano cyfrowe modele wybranych budynków, przygotowano interfejs komunikacyjny z aplikacją nawigacyjną, opracowano system automatycznego zbierania danych oraz przygotowano zestaw zróżnicowanych scenariuszy testowych. Przeprowadzono szerokie badanie na 205 uczestnikach, z których 127 ukończyło wszystkie etapy, a następnie wykonano kompleksowe analizy ilościowe i jakościowe, identyfikując jednocześnie ograniczenia metodyki i formułując wytyczne dla twórców aplikacji nawigacyjnych.

W rozprawie postawiono tezę, iż **wykorzystanie środowiska wirtualnego umożliwi systematyczne, powtarzalne i kontrolowalne testowanie aplikacji przeznaczonych do nawigacji i lokalizacji wewnątrz budynków, a oparta o symulację metodyka pozwoli na wieloaspektową analizę percepcji użytkowników oraz doskonalenie różnorodnych metod przekazu kartograficznego.**

Zrealizowane badania empiryczne potwierdziły, że zaproponowane środowisko pozwala na prowadzenie eksperymentów odpowiadających różnym scenariuszom nawigacyjnym, w tym złożonym trasom w budynkach wielopiętrowych oraz sytuacjom niemożliwym lub trudnym do odtworzenia w rzeczywistości. Uzyskane wyniki pokazały, że symulacja może być traktowana jako pełnoprawne narzędzie badawcze, pozwalające na testowanie wariantów interfejsu aplikacji, metod prezentacji trasy i lokalizacji oraz wskazówek nawigacyjnych w warunkach zbliżonych do rzeczywistych.

Na tej podstawie tezę rozprawy należy uznać za zweryfikowaną pozytywnie, przy świadomości wskazanych w poprzednim rozdziale ograniczeń.

Analiza wyników siedmiu eksperymentów pozwoliła sformułować odpowiedzi na pytania badawcze zdefiniowane we wstępie rozprawy.

- 1) Wykazano, że różne metody prezentacji informacji kartograficznej – w szczególności poziom przybliżenia mapy, wizualizacja trasy, prezentacja aktualnej lokalizacji i sposób rotacji mapy – w istotny sposób wpływają na efektywność nawigacji, długość przebytych tras oraz sposób korzystania z mapy.
- 2) Środowisko wirtualne okazało się użyteczne w identyfikacji typowych błędów nawigacyjnych użytkowników poruszających się w wirtualnych budynkach oraz trudnych miejsc wynikających z architektury tych budynków, co uwidoczniło się m.in. w skupiskach błędnie wybranych tras i sytuacjach błędzenia po całym budynku. Z kolei analiza odpowiedzi i opinii użytkowników pozwoliła na wykrycie niedostatków testowanej jako przykład aplikacji *Mapa PW*.
- 3) Wyniki eksperymentów dotyczących zapamiętywania przebytej drogi i wyboru trasy ewakuacyjnej potwierdziły, że sposób wsparcia nawigacyjnego wpływa na budowę map mentalnych uczestników badania i późniejsze ich decyzje użytkowników zakresie poruszania się po budynku.
- 4) Zrealizowany scenariusz ewakuacyjny pokazał, że symulator pozwala bezpiecznie testować sytuacje o podwyższonym poziomie ryzyka, które w warunkach rzeczywistych wymagałyby kosztownych i trudnych logistycznie ćwiczeń.
- 5) W toku badań zidentyfikowano ograniczenia badań w środowisku wirtualnym, w szczególności związane z technicznymi parametrami środowiska oraz charakterystyką badanej grupy uczestników.

Opracowana w rozprawie metodyka badawcza oraz zrealizowane środowisko symulacyjne zdaniem autora rozprawy mogą być przydatne zarówno dla rozwoju kartografii wnętrz budynków, jak i badań nad aplikacjami nawigacyjnymi.

Po pierwsze, zaproponowano skalowalny i uniwersalny system eksperymentalny, który może być wykorzystany do testowania różnych koncepcji interfejsu i przekazu kartograficznego w warunkach zbliżonych do rzeczywistych, lecz pozbawionych ich ograniczeń organizacyjnych.

Po drugie, praca stanowi przykład metodycznej implementacji środowiska opartego na technologiach gier komputerowych i cyfrowych modeli budynków w badaniach naukowych nad zachowaniami użytkowników.

Po trzecie, na podstawie wyników badań sformułowano rekomendacje dla twórców aplikacji nawigacyjnych, które mogą zostać wykorzystane w praktyce przy projektowaniu systemów dla uczelni, lotnisk, szpitali czy innych złożonych obiektów.

Kierunki dalszych badań

Wyniki rozprawy wskazują kilka perspektywicznych kierunków rozwoju badań.

Pierwszym jest porównanie wyników uzyskanych w środowisku wirtualnym z wynikami analogicznych testów terenowych, co umożliwiłoby ocenę stopnia transferowalności wniosków.

Drugim interesującym kierunkiem rozszerzenia badań jest przeniesienie przygotowanego środowiska do przestrzeni VR (Rysunek 9.1) w oparciu o gogle wirtualnej rzeczywistości lub jaskinię VR. Większy poziom immersji takiej prezentacji środowiska wirtualnego może podnosić jakość wyciąganych wniosków. Dodatkowo interesujące może być zastosowanie technologii eyetrackingu do głębszego zrozumienia sposobu interakcji z mapą.



Rysunek 9.1. Koncept - mapa wnętrza budynku widoczna w goglach VR (opracowanie własne)

Po trzecie warto rozszerzyć poszczególne eksperymenty przedstawione w tej pracy, tak aby poszczególne hipotezy potwierdzić w większej liczbie zróżnicowanych pod względem architektury budynkach oraz na większej liczbie różnorodnych tras.

Kolejnym możliwym kierunkiem rozwoju badań jest głębsza analiza miejsc dezorientacji osób poruszających się po nieznanym wcześniej budynku. Na podstawie zebranych danych potencjalnie możliwa jest predykcja miejsc szczególnie kłopotliwych w budynku z wykorzystaniem uczenia maszynowego.

Wreszcie, interesującym kierunkiem wydaje się wykorzystanie zbudowanego środowiska do długoterminowych badań nad wpływem różnych strategii i pomocy nawigacyjnych na rozwój lub osłabianie umiejętności orientacji przestrzennej użytkowników.

Opracowane środowisko i metodyka badań stanowią punkt wyjścia dla kolejnych prac rozwijających ideę wykorzystania wirtualnych bliźniaków budynków w badaniach nad percepcją przestrzeni, bezpieczeństwem ewakuacji, czy rozwijaniem orientacji przestrzennej.

Bibliografia

- 547 Game Studio. 2023. „WebSocket with Blueprint Plugin”. *Fab.com*. <https://www.fab.com/listings/b7c82850-be49-4d98-af63-6a6a8d5fe652> [dostęp: 12 stycznia 2026].
- Abdelrahman Mahmoud, Edgardo Macatulad, Binyu Lei, Matias Quintana, Clayton Miller, i Filip Biljecki. 2025. „What is a Digital Twin anyway? Deriving the definition for the built environment from over 15,000 scientific publications”. *Building and Environment* 274: 112748. doi:10.1016/j.buildenv.2025.112748.
- Abdollahi Ali, Hossein Arefi, Shirin Malihi, i Mehdi Maboudi. 2023. „Progressive Model-Driven Approach for 3D Modeling of Indoor Spaces”. *Sensors (Basel, Switzerland)* 23(13): 5934. doi:10.3390/s23135934.
- Ahmed Wasiu Akande, Lukman Alage Isiaka, Baba Mala, i Seyi Festus Olatoyinbo. 2025. „Evolution of GNSS/GPS Technology and Its Applications from Ancient Times to the Present”. *Next Research* 2(3): 100387. doi:10.1016/j.nexres.2025.100387.
- AlDhaheri Ali. 2023. „Exploring the Impact of 3D Models on Real Estate Sales Performance: A Comparative Study Using Machine Learning”.
- Altair. 2023. „Global Digital Twin Survey Report Vertical Breakdown: Automotive”. https://altair.com/docs/default-source/pdfs/altair_survey-report-automotive-web.pdf [dostęp: 12 września 2025].
- Amedi Amir, Shahar Shelly, Nira Saporta, i Merav Catalogna. 2024. „Perceptual learning and neural correlates of virtual navigation in subjective cognitive decline: A pilot study”. *iScience* 27(12): 111411. doi:10.1016/j.isci.2024.111411.
- Andrade Chittaranjan. 2021. „A Student’s Guide to the Classification and Operationalization of Variables in the Conceptualization and Design of a Clinical Study: Part 1”. *Indian Journal of Psychological Medicine* 43(2): 177–79. doi:10.1177/0253717621994334.
- Antonin Miroslav. 2024. „Simulation Technology in Firefighter Training: Entering Enclosed Spaces”. *Chemical Engineering Transactions* 111: 271–76. doi:10.3303/CET24111046.
- aSurgingRiver. 2024. „aSurgingRiver/WebView”. <https://github.com/aSurgingRiver/WebView> [dostęp: 12 stycznia 2026].
- Aymeric Augustin. 2013. „Websockets”. *websockets*. <https://websockets.readthedocs.io/en/stable/index.html> [dostęp: 12 stycznia 2026].
- Bai Jinqiang, Shiguo Lian, Zhaoxiang Liu, Kai Wang, i Dijun Liu. 2018. „Virtual-Blind-Road Following-Based Wearable Navigation Device for Blind People”. *IEEE Transactions on Consumer Electronics* 64: 136–43. doi:10.1109/TCE.2018.2812498.

- Balciunas Andrius. 2013. „User-Driven Usability Assessment of Internet Maps”. *International Cartographic Conference Extended Abstracts*.
- Baranowski Marek, Dariusz Gotlib, i Robert Olszewski. 2016. „Properties of Cartographic Modelling under Contemporary Definitions of a Map”. *Polish Cartographical Review* 48(3): 91–100. doi:10.1515/pcr-2016-0011.
- Basiri A., E. S. Lohan, T. Moore, A. Winstanley, P. Peltola, C. Hill, P. Amirian, i P. Figueiredo e Silva. 2017. „Indoor Location Based Services Challenges, Requirements and Usability of Current Solutions”. *Computer Science Review* 24: 1–12.
- Bergmann Martins Vinicius, Fabrício Rosa Amorim, Marcio Augusto Reolon Schmidt, i Luciene Stamato Delazari. 2023. „Study about the appropriate number of participants in map user studies”. *International Journal of Cartography* 9(2): 196–209. doi:10.1080/23729333.2022.2163142.
- Brumana R., F. Banfi, L. Cantini, M. Previtali, i S. Della Torre. 2019. „HBIM LEVEL OF DETAIL-GEOMETRY-ACCURACY AND SURVEY ANALYSIS FOR ARCHITECTURAL PRESERVATION”. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XLII-2-W11*: 293–99. doi:10.5194/isprs-archives-XLII-2-W11-293-2019.
- Brunyé Tad T., Aaron Gardony, Caroline R. Mahoney, i Holly A. Taylor. 2012. „Going to Town: Visualized Perspectives and Navigation through Virtual Environments”. *Computers in Human Behavior* 28(1): 257–66. doi:10.1016/j.chb.2011.09.008.
- Burian Jaroslav, Stanislav Popelka, i Marketa Beitlova. 2018. „Evaluation of the Cartographical Quality of Urban Plans by Eye-Tracking”. *ISPRS International Journal of Geo-Information* 7(5): 192. doi:10.3390/ijgi7050192.
- Campos Thiago, Maria Castello, Eduardo Damasceno, i Natasha Valentim. 2025. „An Updated Systematic Mapping Study on Usability and User Experience Evaluation of Touchable Holographic Solutions”. *Journal on Interactive Systems* 16(1): 172–98. doi:10.5753/jis.2025.4694.
- CENAGIS. 2023. „Centrum Analiz Geoprzestrzennych i Satelitarnych”. <https://cenagis.edu.pl/platforma/> [dostęp: 12 stycznia 2026].
- Cesium. 2021. „Cesium for Unreal”. <https://cesium.com/platform/cesium-for-unreal/> [dostęp: 12 stycznia 2026].
- Charalampous Vlasios, Jerjes Besharat, i Chrysostomos Stylios. 2021. „Designing and Developing a VR Environment for Indoor Fire Simulation”. W *Proceedings of the 24th Pan-Hellenic Conference on Informatics, PCI '20*, New York, NY, USA: Association for Computing Machinery, 237–40. doi:10.1145/3437120.3437315.
- Cheema Muhammad Aamir. 2018. „Indoor Location-Based Services: Challenges and Opportunities”. *SIGSPATIAL Special* 10(2): 10–17. doi:10.1145/3292390.3292394.

- Chen Jorge, i Keith C. Clarke. 2020. „Indoor Cartography”. *Cartography and Geographic Information Science* 47(2): 95–109. doi:10.1080/15230406.2019.1619482.
- Cohen-Or D., Y.L. Chrysanthou, C.T. Silva, i F. Durand. 2003. „A Survey of Visibility for Walkthrough Applications”. *IEEE Transactions on Visualization and Computer Graphics* 9(3): 412–31. doi:10.1109/TVCG.2003.1207447.
- Craik Fergus I. M., i Robert S. Lockhart. 1972. „Levels of processing: A framework for memory research”. *Journal of Verbal Learning and Verbal Behavior* 11(6): 671–84. doi:10.1016/S0022-5371(72)80001-X.
- Craik Fergus I. M., i Endel Tulving. 1975. „Depth of processing and the retention of words in episodic memory”. *Journal of Experimental Psychology: General* 104(3): 268–94. doi:10.1037/0096-3445.104.3.268.
- Cybulski Paweł. 2025. „Predicting Cartographic Symbol Location with Eye-Tracking Data and Machine Learning Approach”. *Journal of Eye Movement Research* 18(4): 35. doi:10.3390/jemr18040035.
- Dang Pei, Jun Zhu, Xiaoqi Qiao, Jianlin Wu, Weilian Li, Jigang You, i Lin Fu. 2023. „How Does Spatial Cognitive Style Affect Indoor Fire Evacuation Wayfinding in Mobile Virtual Reality?” *Cartography and Geographic Information Science* 50(3): 272–88. doi:10.1080/15230406.2023.2176928.
- Dirin Amir, Teemu.H Laine, i Ari Alamäki. 2018. „Managing Emotional Requirements in a Context-Aware Mobile Application for Tourists”. *International Journal of Interactive Mobile Technologies (iJIM)* 12(2): 177. doi:10.3991/ijim.v12i2.7933.
- Dong Weihua, Yulin Wu, Tong Qin, Xinran Bian, Yan Zhao, Yanrou He, Yawei Xu, i Cheng Yu. 2021. „What Is the Difference between Augmented Reality and 2D Navigation Electronic Maps in Pedestrian Wayfinding?” *Cartography and Geographic Information Science* 48(3): 225–40. doi:10.1080/15230406.2021.1871646.
- Epic Games. 2024a. „Datasmith Plugins Overview | Unreal Engine 5.4 Documentation”. <https://dev.epicgames.com/documentation/en-us/unreal-engine/datasmith-plugins-overview> [dostęp: 11 stycznia 2026].
- Epic Games. 2024b. „Hardware and Software Specifications for Unreal Engine | Unreal Engine 5.4 Documentation”. <https://dev.epicgames.com/documentation/en-us/unreal-engine/hardware-and-software-specifications> [dostęp: 12 stycznia 2026].
- Epic Games. 2024c. „Instanced Static Mesh Component in Unreal Engine | Unreal Engine 5.4 Documentation”. <https://dev.epicgames.com/documentation/en-us/unreal-engine/instanced-static-mesh-component-in-unreal-engine> [dostęp: 12 stycznia 2026].
- Epic Games. 2024d. „Nanite Virtualized Geometry in Unreal Engine | Unreal Engine 5.4 Documentation”. <https://dev.epicgames.com/documentation/en-us/unreal-engine/nanite-virtualized-geometry-in-unreal-engine> [dostęp: 8 stycznia 2026].

- Epic Games. 2024e. „XR Best Practices in Unreal Engine | Unreal Engine 5.4 Documentation”. <https://dev.epicgames.com/documentation/en-us/unreal-engine/xr-development> [dostęp: 12 stycznia 2026].
- Esri. 2010. „Feature layers - ArcGIS Documentation”. <https://doc.arcgis.com/en/arcgis-online/reference/feature-layers.htm> [dostęp: 12 stycznia 2026].
- Esri. 2020. „ArcGIS Experience Builder | Esri Developer”. <https://developers.arcgis.com/experience-builder/> [dostęp: 12 stycznia 2026].
- Fan Hongchao, Gefei Kong, i Chaoquan Zhang. 2021. „An Interactive Platform for Low-Cost 3D Building Modeling from VGI Data Using Convolutional Neural Network”. *Big Earth Data* 5(1): 49–65. doi:10.1080/20964471.2021.1886391.
- Feng Huixuan, Huan Lu, Feng Zhang, i Zheng Li. 2024. „Application and Research of High Quality Pixel Streaming Architecture Based on Unreal Engine”. W *Proceedings of the 3rd International Conference on Computer, Artificial Intelligence and Control Engineering, CAICE '24*, New York, NY, USA: Association for Computing Machinery, 17–21. doi:10.1145/3672758.3672762.
- Gems Bernhard, Bruno Mazzorana, Thomas Hofer, Michael Sturm, Roman Gabl, i Markus Aufleger. 2016. „3-D Hydrodynamic Modelling of Flood Impacts on a Building and Indoor Flooding Processes”. *Natural Hazards and Earth System Sciences* 16(6): 1351–68. doi:10.5194/nhess-16-1351-2016.
- Gilboa Shaked, Iris Vilnai-Yavetz, i Vince Mitchell. 2024. „Shopping Mall Detachment: Why Do Some Consumers Avoid Malls?” *Journal of Retailing and Consumer Services* 78: 103740. doi:10.1016/j.jretconser.2024.103740.
- GitHub. 2015. „vengaunt/VaRest”. <https://github.com/vengaunt/VaRest> [dostęp: 8 stycznia 2026].
- Google. 2009. „Apps Script | Google for Developers”. <https://developers.google.com/apps-script> [dostęp: 12 stycznia 2026].
- Gotlib Dariusz. 2011. „Metodyka prezentacji kartograficznych w mobilnych systemach lokalizacyjnych i nawigacyjnych”. *Prace Naukowe Politechniki Warszawskiej. Geodezja* z. 48: 5–158.
- Gotlib Dariusz, Kamil Choromański, i Bogusław Kaczalek. 2022. „CENAGIS geo-cyberinfrastructure as a R&D environment for a Surveyor 4.0”.
- Gotlib Dariusz, Krzysztof Lipka, i Hubert Świech. 2025. „Interactive Indoor Audio-Map as a Digital Equivalent of the Tactile Map”. *Applied Sciences* 15(16): 8975. doi:10.3390/app15168975.
- Gotlib Dariusz, i Hubert Świech. 2023. „Wybrane zagadnienia procesu projektowania i implementacji aplikacji mobilnych wykorzystujących tekstowy i głosowy opis przestrzeni na przykładzie «Mapy Dostępności Budynków»”. *Roczniki Geomatyki* T. 21, z.

- 1(100). <http://yadda.icm.edu.pl/baztech/element/bwmeta1.element.baztech-25721494-3123-4039-9553-fb863407017d> [dostęp: 7 października 2025].
- Gotlib Dariusz, Michał Wyszomirski, i Miłosz Gnat. 2020. „A Simplified Method of Cartographic Visualisation of Buildings’ Interiors (2D+) for Navigation Applications”. *ISPRS International Journal of Geo-Information* 9(6): 407. doi:10.3390/ijgi9060407.
- Grand Review Research. 2024. „Indoor Positioning And Navigation Market Report 2030”. <https://www.grandviewresearch.com/industry-analysis/indoor-positioning-navigation-market-report> [dostęp: 18 lipca 2025].
- Hadj Sassi Mohamed Saifeddine, Federica Battisti, i Marco Carli. 2022. „Simulation-Based Virtual Reality Training for Firefighters”. *Electronic Imaging* 34(10): 194-1-194-95. doi:10.2352/EI.2022.34.10.IPAS-194.
- Hailu Tesfay Gidey, Xiansheng Guo, Haonan Si, Lin Li, i Yukun Zhang. 2024. „Theories and Methods for Indoor Positioning Systems: A Comparative Analysis, Challenges, and Prospective Measures”. *Sensors (Basel, Switzerland)* 24(21): 6876. doi:10.3390/s24216876.
- van der Ham Ineke J. M., Annemarie M. E. Faber, Matthijs Venselaar, Marc J. van Kreveld, i Maarten Löffler. 2015. „Ecological validity of virtual environments to assess human navigation ability”. *Frontiers in Psychology* 6: 637. doi:10.3389/fpsyg.2015.00637.
- Hegarty Mary, Chuanxiuyue He, Alexander P. Boone, Shuying Yu, Emily G. Jacobs, i Elizabeth R. Chrastil. 2023. „Understanding Differences in Wayfinding Strategies”. *Topics in Cognitive Science* 15(1): 102-19. doi:10.1111/tops.12592.
- Hsu Kai-Wei, Ching I Chen, i Meng-Cong Zheng. 2023. „Exploring the Correlation between Campus Map Representation and Wayfinding Behavior through Virtual Environment”. doi:10.54941/ahfe1003194.
- Huang Bo-Chen, Jiun Hsu, Edward T.-H. Chu, i Hui-Mei Wu. 2020. „ARBIN: Augmented Reality Based Indoor Navigation System”. *Sensors* 20(20): 5890. doi:10.3390/s20205890.
- Huang Haosheng, Georg Gartner, Jukka M. Krisp, Martin Raubal, i Nico Van de Weghe. 2018. „Location based services: ongoing evolution and research agenda”. *Journal of Location Based Services* 12(2): 63-93. doi:10.1080/17489725.2018.1508763.
- Huang Sheng, i Jing Kang. 2024. „Optimization Design of Indoor Space Layout Driven by Big Data of User Behaviour”. *Computer-Aided Design and Applications*: 195-208. doi:10.14733/cadaps.2025.S9.195-208.
- J N. Syazwani C., Nur Haliza Abdul Wahab, Noorhazirah Sunar, Sharifah H. S. Ariffin, Keng Yinn Wong, i Yichiet Aun. 2022. „Indoor Positioning System: A Review”. *International Journal of Advanced Computer Science and Applications* 13(6). doi:10.14569/ijacsa.2022.0130659.

- Kattenbeck Markus, Ioannis Giannopoulos, Negar Alinaghi, Antonia Golab, i Daniel R. Montello. 2025. „Predicting Spatial Familiarity by Exploiting Head and Eye Movements during Pedestrian Navigation in the Real World”. *Scientific Reports* 15(1): 7970. doi:10.1038/s41598-025-92274-4.
- Khan Sulaiman, Shah Nazir, i Habib Ullah Khan. 2021. „Analysis of Navigation Assistants for Blind and Visually Impaired People: A Systematic Review”. *IEEE Access* 9: 26712–34. doi:10.1109/ACCESS.2021.3052415.
- Khanzada Feeza Khan, Elahe Delavari, Woojin Jeong, Young Seek Cho, i Jaerock Kwon. 2024. „Comparative Study on Simulated Outdoor Navigation for Agricultural Robots”. *Sensors (Basel, Switzerland)* 24(8): 2487. doi:10.3390/s24082487.
- Kinateder Max, Enrico Ronchi, Daniel Gromer, Mathias Müller, Michael Jost, Markus Nehfischer, Andreas Mühlberger, i Paul Pauli. 2014. „Social Influence on Route Choice in a Virtual Reality Tunnel Fire”. *Transportation Research Part F: Traffic Psychology and Behaviour* 26: 116–25. doi:10.1016/j.trf.2014.06.003.
- Kinateder Max, Enrico Ronchi, Daniel Nilsson, Margrethe Kobes, Mathias Müller, Paul Pauli, i Andreas Mühlberger. 2014. „Virtual Reality for Fire Evacuation Research”. *W*, 313–21. doi:10.15439/2014F94.
- Kong Min-Chul, i Myung-Il Roh. 2024. „A method for implementing a ship navigation simulator for the generation and utilization of virtual data”. *International Journal of Naval Architecture and Ocean Engineering* 16: 100604. doi:10.1016/j.ijnaoe.2024.100604.
- Kraak Menno-Jan, i Sara Irina Fabrikant. 2017. „Of maps, cartography and the geography of the International Cartographic Association”. *International Journal of Cartography* 3(sup1): 9–31. doi:10.1080/23729333.2017.1288535.
- Krassanakis Vassilios, i Paweł Cybulski. 2019. „A Review on Eye Movement Analysis in Map Reading Process: The Status of the Last Decade”. *Geodesy and Cartography*: 191–209. doi:10.24425/gac.2019.126088.
- Kubota Masaya, Masaki Kuribayashi, Seita Kayukawa, Hironobu Takagi, Chieko Asakawa, i Shigeo Morishima. 2024. „Snap&Nav: Smartphone-based Indoor Navigation System For Blind People via Floor Map Analysis and Intersection Detection”. *Proc. ACM Hum.-Comput. Interact.* 8(MHCI): 275:1-275:22. doi:10.1145/3676522.
- van der Kuil M. N. A., A. W. M. Evers, J. M. A. Visser-Meily, i I. J. M. van der Ham. 2021. „Spatial knowledge acquired from first-person and dynamic map perspectives”. *Psychological Research* 85(6): 2137–50. doi:10.1007/s00426-020-01389-y.
- Kulawiak Marcin, Dominik Krajnik, Marta Czaplicka, i Agnieszka Dawidowicz. 2023. „A Web-GIS Tool for Diagnosing Spatial Orientation of Young Adults: Design and Evaluation of Geo-Survey”. *Scientific Reports* 13(1): 18621. doi:10.1038/s41598-023-45268-z.

- Kuribayashi Masaki, Tatsuya Ishihara, Daisuke Sato, Jayakorn Vongkulbhisal, Karnik Ram, Seita Kayukawa, Hironobu Takagi, Shigeo Morishima, i Chieko Asakawa. 2023. „PathFinder: Designing a Map-less Navigation System for Blind People in Unfamiliar Buildings”. W *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, CHI '23, New York, NY, USA: Association for Computing Machinery, 1–16. doi:10.1145/3544548.3580687.
- Laor Tal, i Yair Galily. 2022. „In WAZE We Trust? GPS-Based Navigation Application Users' Behavior and Patterns of Dependency”. *PLOS ONE* 17(11): e0276449. doi:10.1371/journal.pone.0276449.
- Li Bing, Juan Pablo Muñoz, Xuejian Rong, Q. Chen, Jizhong Xiao, Yingli Tian, A. Arditi, i M. Yousuf. 2019. „Vision-Based Mobile Indoor Assistive Navigation Aid for Blind People”. *IEEE Transactions on Mobile Computing* 18: 702–14. doi:10.1109/TMC.2018.2842751.
- Li Yi-Shan, i Fang-Shii Ning. 2018. „Low-Cost Indoor Positioning Application Based on Map Assistance and Mobile Phone Sensors”. *Sensors (Basel, Switzerland)* 18(12): 4285. doi:10.3390/s18124285.
- Liu Jingbin, Ruizhi Chen, Ling Pei, Robert Guinness, i Heidi Kuusniemi. 2012. „A Hybrid Smartphone Indoor Positioning Solution for Mobile LBS”. *Sensors (Basel, Switzerland)* 12(12): 17208–33. doi:10.3390/s121217208.
- Lu Fangfang, Hao Zhou, Lingling Guo, Jingjing Chen, i Licheng Pei. 2021. „An ARCore-Based Augmented Reality Campus Navigation System”. *Applied Sciences* 11(16): 7515. doi:10.3390/app11167515.
- Lütjens Mona, Thomas Kersten, Boris Dorschel, i Felix Tschirschwitz. 2019. „Virtual Reality in Cartography: Immersive 3D Visualization of the Arctic Clyde Inlet (Canada) Using Digital Elevation Models and Bathymetric Data”. *Multimodal Technologies and Interaction* 3(1): 9. doi:10.3390/mti3010009.
- Łobodecki Jakub. 2020. „Opracowanie środowiska testowego do badań percepcji użytkowników aplikacji do nawigacji w budynkach z wykorzystaniem technologii gier”. Praca magisterska. Wydział Geodezji i Kartografii.
- Łobodecki Jakub, i Dariusz Gotlib. 2022. „Developing a Simulator of a Mobile Indoor Navigation Application as a Tool for Cartographic Research”. *Polish Cartographical Review* 54(1): 108–22. doi:10.2478/pcr-2022-0008.
- Manap Zahariah, Azmi Awang Md Isa, Abdul Majid Darsono, i Suraya Zainuddin. 2025. „A Review of Technologies and Techniques for Indoor Positioning Systems”. *International Journal of Academic Research in Business and Social Sciences* 15(2). doi:10.6007/ijarbss/v15-i2/24735.
- Manikandan S. 2011. „Measures of central tendency: Median and mode”. *Journal of Pharmacology & Pharmacotherapeutics* 2(3): 214–15. doi:10.4103/0976-500X.83300.

- Marciniak Jacek. 2018. „Wykorzystanie kartograficznych modeli wewnątrz budynków do podniesienia jakości procesu nawigacji”. Rozprawa doktorska. Wydział Geodezji i Kartografii.
- Martins Vinicius Bergmann, Marcio Augusto Reolon Schmidt, Luciene Stamato Delazari, i Marciano da Costa Lima. 2024. „Configuration of Evaluation Methods for Mobile Digital Maps”. *Boletim de Ciências Geodésicas* 30: e2024016. doi:<https://doi.org/10.1590/s1982-21702024000100016>.
- Mehta Ranjana K., John Kang, Yangming Shi, i Jing Du. 2025. „Effectiveness of Training under Stress in Immersive VR: An Investigation of Firefighter Performance, Gaze Entropy, and Pupillometry”. *Frontiers in Virtual Reality* 6. doi:10.3389/frvir.2025.1542507.
- Murphy Christian E., Nargiz Kurumbayeva, i Brandon Serrao. 2023. „Evaluating Gaze-Based Interfaces for Map Interaction”. *Abstracts of the ICA* 6: 1–2. doi:10.5194/ica-abs-6-178-2023.
- Nori Raffaella, i Fiorella Giusberti. 2006. „Predicting Cognitive Styles from Spatial Abilities”. *The American Journal of Psychology* 119(1): 67–86. doi:10.2307/20445319.
- Nossum Alexander Salveson, Hengshan Li, i Nicholas A. Giudice. 2013. „Vertical Colour Maps – A Data-Independent Alternative to Floor-Plan Maps”. *Cartographica* 48(3): 225–36. doi:10.3138/carto.48.3.1641.
- NVIDIA. 2023. „RTXGI update status”. *NVIDIA Developer Forums*. <https://forums.developer.nvidia.com/t/differences-to-5-2/243643> [dostęp: 12 stycznia 2026].
- Open Geospatial Consortium. 2002. „Web Feature Service (WFS) Standard | OGC Publications”. <https://www.ogc.org/publications/standard/wfs/> [dostęp: 12 stycznia 2026].
- Pimentel Victoria, i Bradford G. Nickerson. 2012. „Communicating and Displaying Real-Time Data with WebSocket”. *IEEE Internet Computing* 16(4): 45–53. doi:10.1109/MIC.2012.64.
- Prema K., Abish Chhetri, Sowmya Pratap Singh, i D Yaswanth Rao. 2024. „Virtual Reality for Real Estate: Revolutionizing Property Visualization and Enhancing Client Engagement”. W *2024 IEEE 4th International Conference on ICT in Business Industry & Government (ICTBIG)*, , 1–7. doi:10.1109/ICTBIG64922.2024.10911827.
- Python Software Foundation. 2008. „Ssl — TLS/SSL Wrapper for Socket Objects”. *Python documentation*. <https://docs.python.org/3/library/ssl.html> [dostęp: 12 stycznia 2026].
- Python Software Foundation. 2014. „Asyncio — Asynchronous I/O”. *Python documentation*. <https://docs.python.org/3/library/asyncio.html> [dostęp: 12 stycznia 2026].
- Quin Federico, Danny Weyns, Matthias Galster, i Camila Costa Silva. 2024. „A/B Testing: A Systematic Literature Review”. *Journal of Systems and Software* 211: 112011. doi:10.1016/j.jss.2024.112011.

- React. 2019. „Built-in React Hooks”. <https://react.dev/reference/react/hooks> [dostęp: 12 stycznia 2026].
- Rehn Adam. 2021. „Unreal Engine For Research”. <https://ue4research.org/> [dostęp: 11 stycznia 2026].
- Rehn Adam, i Luke Bermingham. 2021. „An Open Architecture for Scalable Pixel Streaming”. <https://tensorworks.com.au//blog/an-open-architecture-for-scalable-pixel-streaming/> [dostęp: 12 stycznia 2026].
- Rehrl Karl, Elisabeth Häusler, Sven Leitinger, i Daniel Bell. 2014. „Pedestrian navigation with augmented reality, voice and digital map: final results from an in situ field study assessing performance and user experience”. *Journal of Location Based Services* 8(2): 75–96. doi:10.1080/17489725.2014.946975.
- Richter Julia, Robin Thomas, David Lange, Thomas Graichen, i Ulrich Heinkel. 2022. „Development of a Platform-Independent Renderer for the Rendering of OpenStreetMap Indoor Maps in Flutter”. W *Proceedings of the 17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, Online Streaming, --- Select a Country ---: SCITEPRESS - Science and Technology Publications, 147–54. doi:10.5220/0010781500003124.
- Saback Vanessa, Cosmin Popescu, Thomas Blanksvärd, i Björn Täljsten. 2024. „Analysis of Digital Twins in the Construction Industry: Practical Applications, Purpose, and Parallel with Other Industries”. *Buildings* 14(5): 1361. doi:10.3390/buildings14051361.
- Sakpere Wilson, Michael Adeyeye Oshin, i Nhlanhla Bw Mlitwa. 2017. „A State-of-the-Art Survey of Indoor Positioning and Navigation Systems and Technologies”. *South African Computer Journal* 29(3). doi:10.18489/sacj.v29i3.452.
- Sarot Rhaíssa Viana, Luciene Stamato Delazari, i Silvana Philippi Camboim. 2021. „Proposal of a Spatial Database for Indoor Navigation”. *Acta Scientiarum. Technology* 43: e51718. doi:10.4025/actascitechnol.v43i1.51718.
- Sartayeva Yerkezhan, Henry C.B. Chan, Yik Him Ho, i Peter H.J. Chong. 2023. „A Survey of Indoor Positioning Systems Based on a Six-Layer Model”. *Computer Networks* 237: 110042. doi:10.1016/j.comnet.2023.110042.
- Schim Raph. 2019. „From perspective picture to orthographic picture”. *Stack Overflow*. <https://stackoverflow.com/q/36573283> [dostęp: 11 stycznia 2026].
- Sedlák Michal, Čeněk Šašinka, Zdeněk Stachoň, Jiří Chmelík, i Milan Doležal. 2022. „Collaborative and Individual Learning of Geography in Immersive Virtual Reality: An Effectiveness Study” red. Stefano Triberti. *PLOS ONE* 17(10): e0276267. doi:10.1371/journal.pone.0276267.

- Shao Zhi Kang. 2022. „Journeyman’s Minimap”. *Fab.com*. <https://www.fab.com/listings/2fce6e44-aed4-4a59-bb69-6d82f4ac35da> [dostęp: 12 stycznia 2026].
- Shih Guan-Rong, i Pei-Hsuan Tsai. 2023. „Safest-Path Planning Approach for Indoor Fire Evacuation”. *International Journal of Disaster Risk Reduction* 93: 103760. doi:10.1016/j.ijdr.2023.103760.
- Simões Walter C. S. S., Guido S. Machado, André M. A. Sales, Mateus M. de Lucena, Nasser Jazdi, i Vicente F. de Lucena. 2020a. „A Review of Technologies and Techniques for Indoor Navigation Systems for the Visually Impaired”. *Sensors (Basel, Switzerland)* 20(14): 3935. doi:10.3390/s20143935.
- Simões Walter C. S. S., Guido S. Machado, André M. A. Sales, Mateus M. de Lucena, Nasser Jazdi, i Vicente F. de Lucena. 2020b. „A Review of Technologies and Techniques for Indoor Navigation Systems for the Visually Impaired”. *Sensors (Basel, Switzerland)* 20(14): 3935. doi:10.3390/s20143935.
- Stachoň Z., K. Jochecová, O. Kvarda, D. Snopková, P. Ugwitz, A. Šašinková, S. Ježek, i in. 2025. „The Possibilities of Using Immersive Virtual Environments in Research on Way Finding”. *International Journal of Human-Computer Studies* 196: 103442. doi:10.1016/j.ijhcs.2024.103442.
- Swagger. 2011. „REST API Documentation Tool | Swagger UI”. <https://swagger.io/tools/swagger-ui/> [dostęp: 12 stycznia 2026].
- Świech Hubert. 2024. „Koncepcja i implementacja algorytmu nawigacji wewnątrz budynków opartego o geo-opisy oraz hybrydowy system pozycjonowania”.
- Tahir Rabail, i John Krogstie. 2023. „Impact of Navigation Aid and Spatial Ability Skills on Wayfinding Performance and Workload in Indoor-Outdoor Campus Navigation: Challenges and Design”. *Applied Sciences* 13(17): 9508. doi:10.3390/app13179508.
- Tan Jonathan Koon Ngee, Shuai Zhang, Adrian Wing-Keung Law, i Sai Hung Cheung. 2025. „Digital-twin enabled evacuation to improve individual and community resilience of building occupants against indoor Fires: Framework and route-finding algorithm”. *Developments in the Built Environment* 22: 100672. doi:10.1016/j.dibe.2025.100672.
- Tang Fangqin, i Aizhu Ren. 2012. „GIS-Based 3D Evacuation Simulation for Indoor Fire”. *Building and Environment* 49: 193–202. doi:10.1016/j.buildenv.2011.09.021.
- Tariq Sadia, Anis Ur Rahman, Tahir Azim, i Rehman Gull Khan. 2021. „Instanced model simplification using combined geometric and appearance-related metric”. doi:10.48550/arXiv.2101.02570.
- Topete Alexis, Chuanxiuyue He, John Protzko, Jonathan Schooler, i Mary Hegarty. 2024. „How is GPS used? Understanding navigation system use and its relation to spatial ability”. *Cognitive Research: Principles and Implications* 9: 16. doi:10.1186/s41235-024-00545-x.

- Tural Alp, i Elif Tural. 2024. „Exploring Sense of Spaciousness in Interior Settings: Screen-Based Assessments with Eye Tracking, and Virtual Reality Evaluations”. *Frontiers in Psychology* 15. doi:10.3389/fpsyg.2024.1473520.
- Wenig Dirk, Alexander Steenbergen, Johannes Schöning, Brent Hecht, i Rainer Malaka. 2016. „ScrollingHome: Bringing Image-Based Indoor Navigation to Smartwatches”. W *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services*, Florence Italy: ACM, 400–406. doi:10.1145/2935334.2935373.
- Wikipedia. 2025a. „Containerization (Computing)”. [https://en.wikipedia.org/w/index.php?title=Containerization_\(computing\)](https://en.wikipedia.org/w/index.php?title=Containerization_(computing)) [dostęp: 3 lutego 2026].
- Wikipedia. 2025b. „NASA-TLX”. <https://en.wikipedia.org/wiki/NASA-TLX> [dostęp: 11 stycznia 2026].
- Wikipedia. 2025c. „System Usability Scale”. https://en.wikipedia.org/wiki/System_usability_scale [dostęp: 11 stycznia 2026].
- Wikipedia. 2025d. „Unreal Engine 1”. https://en.wikipedia.org/wiki/Unreal_Engine_1 [dostęp: 11 stycznia 2026].
- Xiao Yi, Tinghua Ai, Min Yang, i Xiang Zhang. 2020. „A Multi-Scale Representation of Point-of-Interest (POI) Features in Indoor Map Visualization”. *ISPRS International Journal of Geo-Information* 9(4): 239. doi:10.3390/ijgi9040239.
- Yan Wanling, Jialing Li, Can Mi, Wei Wang, Zhengjia Xu, Wenjing Xiong, Longxing Tang, i in. 2022. „Does Global Positioning System-Based Navigation Dependency Make Your Sense of Direction Poor? A Psychological Assessment and Eye-Tracking Study”. *Frontiers in Psychology* 13: 983019. doi:10.3389/fpsyg.2022.983019.
- Yan Zhenbin, Zixuan Meng, i Yong Tan. 2025. „Does Virtual Reality Help Property Sales? Empirical Evidence from a Real Estate Platform”. *Information Systems Research* 36(2): 1011–30. doi:10.1287/isre.2021.9138.
- Yi Xiaolie, Shizhu Lu, Yumeng Zhong, Jifa Zhang, i Yuqing Guo. 2024. „Numerical Simulation and Safety Assessment of Fires in Historic Timber Structures Based on Fire Load Investigation”. *Heritage Science* 12(1): 222. doi:10.1186/s40494-024-01344-9.
- Ying Qi, Weihua Dong, i Sara I. Fabrikant. 2024. „How Do In-Car Navigation Aids Impair Expert Navigators’ Spatial Learning Ability?” *Annals of the American Association of Geographers* 114(7): 1483–1504. doi:10.1080/24694452.2024.2356858.
- Zagata Krzysztof, i Beata Medyńska-Gulij. 2023. „Mini-Map Design Features as a Navigation Aid in the Virtual Geographical Space Based on Video Games”. *ISPRS International Journal of Geo-Information* 12(2): 58. doi:10.3390/ijgi12020058.

- Zghyer Rami, Runar Ostnes, Karl Henning Halse, i Odd Sveinung Hareide. 2022. „Applications of Maritime Simulators in Industry and Research”.
- Zhang Zhang, Qu Wang, Wenfeng Wang, Meijuan Feng, i Liangliang Guo. 2025. „A Testing and Evaluation Framework for Indoor Navigation and Positioning Systems”. *Sensors* 25(7): 2330. doi:10.3390/s25072330.
- Zlatanova Sisi, Jinjin Yan, Yijing Wang, Abdoulaye Diakité, Umit Isikdag, George Sithole, i Jack Barton. 2020. „Spaces in Spatial Science and Urban Applications—State of the Art Review”. *ISPRS International Journal of Geo-Information* 9(1): 58. doi:10.3390/ijgi9010058.

Spis rysunków

Rysunek 2.1. a) Uproszczony widok 3D (Xiao i in. 2020) b) Lokalizacja osób na piętrach oznaczona kolorem ramki (Nossum, Li, i Giudice 2013) c) Koncepcja mapy 2D+ dodająca przekrój pionowy do prezentacji kartograficznej (Gotlib, Wyszomirski, i Gnat 2020)	22
Rysunek 2.2. Klasyfikacja systemów pozycjonowania w przestrzeniach zamkniętych (Simões i in. 2020b)	24
Rysunek 2.3. Prognozowany rozmiar rynku pozycjonowania i nawigacji wewnątrz budynków w latach 2020-2030 (Grand Review Research 2024)	26
Rysunek 2.4. Zrzuty ekranów przykładowych aplikacji z różnych kategorii. Kolejno: Dostępny UŚ, Aena, Galeria Wileńska w Warszawie, MazeMap, GdzieToMetro, Google Maps nawigacja wewnątrz budynku Westfield Arkadia (na podstawie Świech 2024)	31
Rysunek 3.1. Podział metod ewaluacji map (Martins i in. 2024)	33
Rysunek 3.2. Mapy pokazujące zagregowane ruchy oczu uczestników podczas rozwiązywania wskazanego podczas badań zadania związanego z mapą topograficzną (Burian, Popelka, i Beitlova 2018)	38
Rysunek 4.1. Powszechność wykorzystywanie cyfrowych bliźniaków w różnych branżach (Altair 2023)	45
Rysunek 4.2. Porównanie dwóch sposobów prezentacji wirtualnego miasta: widok "z lotu ptaka" po lewej oraz widok z pierwszej osoby po prawej (Brunyé i in. 2012)	47
Rysunek 4.3. Przykłady aplikacji badawczych wykorzystujących środowiska wirtualne do pracy z danymi przestrzennymi (Lütjens i in. 2019; Sedlák i in. 2022)	48
Rysunek 4.4. Symulator poruszania po wnętrzu budynku, przeznaczony do testowania wskazówek nawigacyjnych (Łobodecki i Gotlib 2022)	49
Rysunek 4.5. Symulacja ewakuacji z budynku podczas pożaru (Tang i Ren 2012)	51
Rysunek 4.6. Kolejne fazy badania potwierdzającego wpływ treningu w VR na sytuacje w świecie rzeczywistym (Antonin Miroslav 2024)	52
Rysunek 4.7. Ilustracja ścieżki przepływu wody do oraz z budynku (Gems i in. 2016)	52
Rysunek 4.8. Porównanie różnic pomiędzy środowiskiem rzeczywistym a wirtualnym (Stachoń i in. 2025)	53
Rysunek 5.1. Diagram metodyki badawczej (opracowanie własne)	56

Rysunek 5.2. Diagram procesu badawczego z użyciem projektowanego symulatora (opracowanie własne)	60
Rysunek 6.1. Okładka gry Unreal z 1998 roku (archive.org). Hasło reklamowe: „Najlepiej wyglądająca gra wszechczasów.”	70
Rysunek 6.2. Podgląd podstawowego diagramu zależności systemów w projekcie Unreal Engine (dev.epicgames.com)	74
Rysunek 6.3. Lista komponentów rozszerzających funkcjonalność klasy PlayerController (opracowanie własne)	76
Rysunek 6.4. Lista widżetów składających się na interfejs użytkownika symulatora IndoorLab (opracowanie własne)	77
Rysunek 6.5. Poziomy wykorzystywane w projekcie IndoorLab (opracowanie własne)	78
Rysunek 6.6. Konfiguracja komponentu przechowującego metadane budynku (opracowanie własne)	78
Rysunek 6.7. Proces importu danych z wykorzystaniem pluginu DataSmith (Epic Games 2024a)	82
Rysunek 6.8. Numery pomieszczeń wyświetlane na drzwiach w modelu budynku Gmachu Elektroniki (opracowanie własne)	83
Rysunek 6.9. Przykład Gmachu Fizyki wymodelowanego jako jeden pojedynczy obiekt, który po zaimportowaniu automatycznie podzielono na mniejsze (opracowanie własne)	84
Rysunek 6.10. Ilustracja zasady działania technik Occlusion Culling oraz Frustum Culling (Cohen-Or i in. 2003)	87
Rysunek 6.11. Przykłady różnych trybów wizualizacji w technologii Nanite w silniku Unreal Engine (Epic Games 2024d)	88
Rysunek 6.12. Wizualizacja technologii RTXGI pozwalającej na generowanie globalnego oświetlenia w oparciu o regularnie rozmieszczone próbki (opracowanie własne)	89
Rysunek 6.13. Ekran wyboru trybu sterowania (opracowanie własne)	91
Rysunek 6.14. Zrzut ekranu prezentujący podświetlenie spodziewanego punktu docelowego w uproszczonym trybie sterowania (opracowanie własne)	93
Rysunek 6.15. Pierwsza część strony dostępowej systemu badawczego, zawierająca charakterystykę badania (opracowanie własne)	95

Rysunek 6.16. Druga część strony dostępowej systemu badawczego, zawierająca dane dostępne oraz przycisk przenoszący na stronę logowania (opracowanie własne)	96
Rysunek 6.17. Ekran wyboru wariantu A i B dostępu do aplikacji badawczej (opracowanie własne)	97
Rysunek 6.18. Ekran ładowania trybu "Przesyłanie z chmury" (opracowanie własne)	99
Rysunek 6.19. Proces nawiązywania połączenia z serwerem streamingu (opracowanie własne)	99
Rysunek 6.20. Diagram architektury konfiguracji sieciowej wdrożonej na potrzeby technologii Pixel Streaming (opracowanie własne)	101
Rysunek 6.21. Ekran prezentujący przyciski prowadzące do pobierania potrzebnych plików (opracowanie własne)	103
Rysunek 6.22. Ilustracja wyglądu mapy w prototypie systemu (Łobodecki 2020)	104
Rysunek 6.23. Widok przekroju budynku (opracowanie własne)	105
Rysunek 6.24. Różnica pomiędzy projekcją perspektywną, a ortograficzną (Schim 2019)	105
Rysunek 6.25. Porównanie projekcji perspektywicznej (obraz po lewej) i ortograficznej (obraz po prawej) (opracowanie własne)	106
Rysunek 6.26. Po lewej obraz przekroju poziomego przez budynek o wysokim poziomie ekspozycji. Po prawej efekt działania inwersji kolorów na przechwyconym obrazie (opracowanie własne)	106
Rysunek 6.27. Wybór trybu wyświetlania mapy, przedstawiany uczestnikom na początku badania (opracowanie własne)	108
Rysunek 6.28. a) Aplikacja nawigacyjna działa na smartfonie i jest połączona z silnikiem gier b) Zrzut ekranu uruchomionej symulacji ze zintegrowaną aplikacją nawigacyjną (opracowanie własne)	109
Rysunek 6.29. Schemat uruchamiania aplikacji nawigacyjnej i komunikacji z silnikiem gier podczas trwania badania (opracowanie własne)	110
Rysunek 6.30. Główna część kodu serwera komunikacyjnego (opracowanie własne)	112
Rysunek 6.31. Podgląd strony dokumentacji API (opracowanie własne)	113
Rysunek 6.32. Dokumentacja wiadomości dotyczącej aktualizacji lokalizacji (opracowanie własne)	114

Rysunek 6.33. Wycinek ekranu prezentujący QR kod pozwalający na nawiązanie połączenia z symulatorem (opracowanie własne)	115
Rysunek 6.34. Kod w języku JavaScript inicjujący połączenie WebSocket (opracowanie własne)	116
Rysunek 6.35. Fragment kodu odpowiedzialnego za działanie funkcji typu heartbeat (opracowanie własne)	117
Rysunek 6.36. Fragment funkcji messageHandlerCallback (opracowanie własne)	118
Rysunek 6.37. Cztery rodzaje nakładki zakrywającej mapę (opracowanie własne)	120
Rysunek 6.38. Fragment kodu wyświetlający odpowiedni wariant nakładki zakrywającej mapę (opracowanie własne)	121
Rysunek 6.39. Podgląd arkusza kalkulacyjnego pełniącego rolę bazy danych zbieranych podczas badania (opracowanie własne)	122
Rysunek 6.40. Kod w usłudze Google Apps Script pozwalający na uruchomienie API, przetworzenie danych oraz ich zapis w arkuszu kalkulacyjnym (opracowanie własne)	123
Rysunek 6.41. Efekt działania poleceń `Stat Unit` oraz `Stat UnitGraph` podczas wyświetlania wykorzystywanego w symulacji modelu Gmachu Głównego PW (opracowanie własne)	133
Rysunek 6.42. Okno narzędzia GPU Visualizer służącego do profilowania w Unreal Engine (opracowanie własne)	133
Rysunek 6.43. Dynamiczne statystyki wyświetlane po wywołaniu polecenia `Stat InitViews` w silniku Unreal Engine (opracowanie własne)	134
Rysunek 6.44. Widok Shader Complexity jako jedno z narzędzi profilowania silnika Unreal (opracowanie własne)	134
Rysunek 6.45. Ekran aplikacji przeznaczonej do analizy wyników (opracowanie własne)	136
Rysunek 7.1. Zrzut ekranu prezentujący fragment poziomego treningowego (opracowanie własne)	140
Rysunek 7.2. Zrzut ekranu prezentujący ankietę początkową (opracowanie własne)	141
Rysunek 7.3. Zrzut ekranu prezentujący pierwszy scenariusz testowy, z widocznym znakiem zapytania przy którym w górnej części ekranu prezentowane są dodatkowe informacje w formie tekstowej (opracowanie własne)	142
Rysunek 7.4. Wygląd aplikacji analitycznej bezpośrednio po uruchomieniu. Na górze dwa przyciski umożliwiające zmianę trybu, po lewej mapa z naniesionymi trasami przejścia	

uczestników, po prawo widok filtrów wykorzystywanych w trybie analiz przestrzennych (opracowanie własne)	144
Rysunek 7.5. Przykładowy fragment wizualizacji zarejestrowanych tras, jakie uczestników badania przebyli w wirtualnym Gmachu Chemii na poziomie parteru (opracowanie własne)	145
Rysunek 7.6. Przykładowy fragment wizualizacji miejsc, w których uczestnicy badania korzystali z mapy w wirtualnym Gmachu Chemii na poziomie parteru (opracowanie własne)	146
Rysunek 7.7. Procentowy udział uczestników eksperymentu, którzy pomyślnie zakończyli poszczególne zadania nawigacyjne. Wysokość słupków przedstawia odsetek użytkowników, którzy ukończyli dany scenariusz. Na osi X widoczne skrótowe nazwy poszczególnych scenariuszy, w większości są to skróty nazw budynków po których poruszali się użytkownicy (opracowanie własne).	149
Rysunek 7.8. Średni czas ukończenia całego badania w podziale na płeć (opracowanie własne)	149
Rysunek 7.9. Średni czas ukończenia całego badania w podziale na grupy wiekowe (opracowanie własne)	150
Rysunek 7.10. Macierz korelacji wewnętrznych Spearmana między cechami demograficznymi, umiejętnościami użytkowników i zmiennymi interfejsu. Przedstawiono tylko dolny trójkąt macierzy ze względu na symetrię korelacji (opracowanie własne).	154
Rysunek 7.11. Macierz korelacji Spearmana między cechami demograficznymi, umiejętnościami użytkowników, zmiennymi interfejsu, a wydajnością nawigacji w środowisku wirtualnym (opracowanie własne)	155
Rysunek 7.12. Korzystanie z widoku mapy przez uczestników badania – procent scenariuszy, w których uczestnicy skorzystali lub nie skorzystali z mapy chociaż raz, z porównaniem dla osób znających oraz nie znających danego budynku (opracowanie własne)	158
Rysunek 7.13. Średnia liczba spojrzeń na mapę, a cechy demograficzne (opracowanie własne)	159
Rysunek 7.14. Średnia liczba spojrzeń na mapę, a doświadczenie i umiejętności (opracowanie własne)	160
Rysunek 7.15. Ilustracja trzech poziomów przybliżenia określanych od lewej jako: Odległy zoom, Średni zoom, Bliski zoom (opracowanie własne)	162

Rysunek 7.16. Trasa wyznaczona przez aplikację dla scenariusza eiti1, linia ciągła pokazuje trasę na piętrze 0, linia przerywana pokazuje trasę na piętrze 1 (opracowanie własne)	164
Rysunek 7.17. Trasa scenariusza eiti2, linia ciągła pokazuje trasę na piętrze 1, linia przerywana pokazuje trasę na piętrze 4 (opracowanie własne)	164
Rysunek 7.18. Rozkład liczebności grup według wylosowanego poziomu przybliżenia mapy (opracowanie własne)	165
Rysunek 7.19. Czas oraz przebyty dystans w poszukiwaniu pomieszczenia w eksperymencie 1 (opracowanie własne)	166
Rysunek 7.20. Rozkład statystyczny wyników realizacji eksperymentu 1. Czas ukończenia po lewej, przebyty dystans po prawej (opracowanie własne)	166
Rysunek 7.21. Pojedyncze zapisy przebytych tras wskazują na błędzenie po całym budynku. Taką sytuację zaobserwowano tylko dla grupy z bliskim oraz średnim poziomem zoom (opracowanie własne)	167
Rysunek 7.22. Ilustracja czterech trybów mapy, które były do dyspozycji poszczególnych grup. Kolejno: Brak wsparcia, Lokalizacja, Trasa, Pełne wsparcie (opracowanie własne)	170
Rysunek 7.23. Trasa wyznaczona przez aplikację dla scenariusza chem1, linia ciągła pokazuje trasę na piętrze 1, linia przerywana pokazuje trasę na piętrze 0 (opracowanie własne)	172
Rysunek 7.24. Trasa wyznaczona przez aplikację dla scenariusza chem2, linia ciągła pokazuje trasę na piętrze 1, linia przerywana pokazuje trasę na piętrze 4 (opracowanie własne)	172
Rysunek 7.25. Trasa wyznaczona przez aplikację dla scenariusza chem3, linia ciągła pokazuje trasę na piętrze 4, linia przerywana pokazuje trasę na piętrze 0 (opracowanie własne)	173
Rysunek 7.26. Rozkład liczebności grup według wylosowanego trybu (opracowanie własne)	173
Rysunek 7.27. Czas oraz przebyty dystans w poszukiwaniu pomieszczenia w eksperymencie 2a (opracowanie własne)	174
Rysunek 7.28. Rozkład statystyczny wyników realizacji eksperymentu 2a. Czas ukończenia po lewej, przebyty dystans po prawej (opracowanie własne)	174
Rysunek 7.29. Rozkład statystyczny statystyk interakcji z mapą podczas eksperymentu 2a. Czas kontaktu z mapą po lewej, liczba spojrzeń na mapę po prawej (opracowanie własne)	175

Rysunek 7.30. Rozkład statystyczny wyników realizacji eksperymentu 2b w relacji do trybu wsparcia nawigacyjnego otrzymanego w eksperymencie 2a. Czas ukończenia po lewej, przebyty dystans po prawej (opracowanie własne)	178
Rysunek 7.31. Ilustracja trzech trybów prezentacji rotacji określanych od lewej jako: Brak rotacji, Rotacja niebieskiej kropki, Rotacja mapy (opracowanie własne)	181
Rysunek 7.32. Trasa wyznaczona przez aplikację dla eksperymentu 3, linia ciągła pokazuje trasę na piętrze 0, linia przerywana pokazuje trasę na piętrze 2 (opracowanie własne)	183
Rysunek 7.33. Rozkład liczebności grup według wylosowanego trybu prezentacji rotacji (opracowanie własne)	183
Rysunek 7.34. Czas oraz przebyty dystans w poszukiwaniu pomieszczenia w eksperymencie 3 (opracowanie własne)	184
Rysunek 7.35. Rozkład statystyczny wyników realizacji eksperymentu 3. Czas ukończenia po lewej, przebyty dystans po prawej, średnia prędkość na dole (opracowanie własne).	185
Rysunek 7.36. Rozkład statystyczny statystyk interakcji z mapą podczas eksperymentu 3. Czas kontaktu z mapą po lewej, liczba spojrzeń na mapę po prawej (opracowanie własne).	186
Rysunek 7.37. Trasa scenariusza gg2, linia ciągła pokazuje trasę na piętrze 1, linia przerywana pokazuje trasę na piętrze 4 (opracowanie własne)	188
Rysunek 7.38. Rozkład liczebności grup eksperymentu 4, według wylosowanego przydziału do grupy z lub bez wizualizacji trasy (opracowanie własne)	188
Rysunek 7.39. Czas oraz przebyty dystans w poszukiwaniu pomieszczenia w eksperymencie 4 (opracowanie własne)	189
Rysunek 7.40. Rozkład wyników realizacji eksperymentu 4. Czas ukończenia po lewej, przebyty dystans po prawej (opracowanie własne)	189
Rysunek 7.41. Rozkład wyników realizacji eksperymentu 4. Czas kontaktu z mapą po lewej, liczba spojrzeń na mapę po prawej (opracowanie własne).	190
Rysunek 7.42. Porównanie zarejestrowanych tras uczestników eksperymentu 4. Po lewej uczestnicy, którzy nie widzieli trasy, po prawej uczestnicy, którzy widzieli trasę. Czerwoną pinezką oznaczono punkt startu scenariusza. Strzałki „A” wskazują obszar, na którym widać, iż znacznie większa część badanych z grupy z wizualizacją trasy obrała najkrótszą możliwą drogę do celu, zgodną z trasą wyznaczoną przez aplikację nawigacyjną. Strzałki „B” wskazują obszar, na którym widać, iż znacznie większa część badanych z grupy z wizualizacją trasy skorzystała z	

optymalnie ulokowanej windy lub klatki schodowej, co skracało czas dotarcia do celu i przebyty dystans (opracowanie własne).	191
Rysunek 7.43. Zrzut ekranu symulatora, na którym w lewym górnym rogu widać aktywny tryb dodatkowych wskazówek w formie strzałek (opracowanie własne)	193
Rysunek 7.44. Trasa wyznaczona przez aplikację dla scenariusza gg3, linia ciągła pokazuje trasę na piętrze 1, linia przerywana pokazuje trasę na piętrze 4 (opracowanie własne)	194
Rysunek 7.45. Rozkład liczebności grup eksperymentu 5, według wylosowanego przydziału do grupy z lub bez wyświetlania dodatkowych wskazówek (opracowanie własne)	195
Rysunek 7.46. Czas oraz przebyty dystans w poszukiwaniu pomieszczenia w eksperymencie 5 (opracowanie własne)	195
Rysunek 7.47. Rozkład statystyczny wyników realizacji eksperymentu 5. Czas ukończenia po lewej, przebyty dystans po prawej, średnia prędkość na dole (opracowanie własne).	196
Rysunek 7.48. Rozkład statystyczny statystyk interakcji z mapą podczas eksperymentu 5. Czas kontaktu z mapą po lewej, liczba spojrzeń na mapę po prawej (opracowanie własne).	197
Rysunek 7.49. Zrzut ekranu symulatora, w centralnej części widać przykład oznakowania korytarzy (opracowanie własne)	199
Rysunek 7.50. Trasa wyznaczona przez aplikację dla scenariusza et1, linia ciągła pokazuje trasę na piętrze 0, linia przerywana pokazuje trasę na piętrze 1 (opracowanie własne)	201
Rysunek 7.51. Trasa scenariusza et2, linia ciągła pokazuje trasę na piętrze 0, linia przerywana pokazuje trasę na piętrze 4 (opracowanie własne)	201
Rysunek 7.52. Rozkład liczebności grup eksperymentu 6, według wylosowanej dostępności mapy (opracowanie własne)	202
Rysunek 7.53. Czas oraz przebyty dystans w poszukiwaniu pomieszczenia w scenariuszu et1 (opracowanie własne)	203
Rysunek 7.54. Czas oraz przebyty dystans w poszukiwaniu pomieszczenia w scenariuszu et2 (opracowanie własne)	203
Rysunek 7.55. Rozkład statystyczny wyników realizacji eksperymentu 6 z podziałem na dwa scenariusze. Czas ukończenia po lewej, przebyty dystans po prawej, średnia prędkość na dole (opracowanie własne).	204
Rysunek 7.56. Zrzut ekranu początku scenariusza archi2, przedstawiający pożar w p. 120, który był celem scenariusza archi1 (opracowanie własne)	207

Rysunek 7.57. Widok na możliwe drogi ewakuacji. Na pierwszym rysunku droga dalsza, którą dotarto do tego miejsca w poprzednim scenariuszu. Znak ewakuacji jest ledwo widoczny w głębi korytarza. Na drugim rysunku droga bliższa, nieznana jeszcze uczestnikom, wyraźnie oznaczona znakiem ewakuacji (opracowanie własne).	208
Rysunek 7.58. Trasa wyznaczona przez aplikację dla scenariusza archi1, linia ciągła pokazuje trasę na piętrze 0, linia przerywana pokazuje trasę na piętrze 1 (opracowanie własne)	210
Rysunek 7.59. Wizualizacja dostępnych dróg ewakuacji w scenariuszu archi2. Czerwoną pinezką oznaczono punkt startu. Niebieskimi flagami oznaczono dostępne wyjścia ewakuacyjne, do których dotarcie użytkownika kończyło scenariusz. Czerwone linie to zarejestrowane ścieżki uczestników (opracowanie własne).	210
Rysunek 7.60. Zrzut ekranu prezentujący dezaktywowaną windę (opracowanie własne)	211
Rysunek 7.61. Rozkład liczebności grup eksperymentu w scenariuszu arch1, według wylosowanego przydziału do grupy z lub bez wizualizacji trasy (opracowanie własne)	211
Rysunek 7.62. Czas oraz przebyty dystans w poszukiwaniu pomieszczenia w scenariuszu archi1 (opracowanie własne)	212
Rysunek 7.63. Czas kontaktu z mapą w zależności od wizualizacji trasy w scenariuszu archi1 (opracowanie własne)	213
Rysunek 7.64. Wybór drogi ewakuacyjnej w scenariuszu archi2 w rozdzieleniu na prezentację trasy w scenariuszu archi1 (opracowanie własne)	214
Rysunek 7.65. Ankieta końcowa – formularz zbierania danych jakościowych wyświetlany na końcu procesu badawczego (opracowanie własne)	216
Rysunek 7.66. Częstotliwość występowania poszczególnych etykiet w kategorii „Ocena działania symulatora” (opracowanie własne)	217
Rysunek 7.67. Częstotliwość występowania poszczególnych etykiet w kategorii „Ocena map wewnątrz budynków” (opracowanie własne)	221
Rysunek 7.68. Częstotliwość występowania poszczególnych etykiet w kategorii „Sugestie ulepszeń” (opracowanie własne)	222
Rysunek 9.1. Koncept - mapa wnętrza budynku widoczna w goglach VR (opracowanie własne)	236
Rysunek 0.1. Kiosk informacyjny umiejscowiony w Gmachu Głównym Politechniki Warszawskiej (opracowanie własne)	263

Rysunek 0.2. Aplikacja Mapa PW z wyznaczoną trasą na mapie (opracowanie własne)	264
Rysunek 0.3. Widok widżetu wybierania piętra oraz wyznaczonej trasy wraz z punktami startu i końca trasy (opracowanie własne)	265
Rysunek 0.4. Wizualizacja modelu Gmachu Fizyki w aplikacji Mapa PW (opracowanie własne)	266
Rysunek 0.5. Widok modułu "Ułatwień dostępności" w aplikacji Mapa PW (opracowanie własne)	267
Rysunek 0.6. Aplikacja Mapa PW uruchomiona na urządzeniu mobilnym z aktualną pozycją oznaczoną na podstawie skanowania kodu QR (opracowanie własne)	268
Rysunek 0.7. Diagram architektury aplikacji Mapa PW (opracowanie własne, na podstawie (Gotlib, Lipka, i Świech 2025))	270

Spis tabel

Tabela 1. Liczba uczestników w badaniach map (na podstawie (Bergmann Martins i in. 2023))	35
Tabela 2. Wykaz zmiennych niezależnych, których randomizacja została zaimplementowana w symulatorze i wykorzystana w badaniu na użytkownikach (opracowanie własne)	64
Tabela 3. Listę danych rejestrowanych i przesyłanych przez system badawczy (opracowanie własne)	67
Tabela 4. Wady i zalety przesyłania obrazu z chmury (opracowanie własne)	98
Tabela 5. Wady i zalety uruchomienia aplikacji badawczej na komputerze uczestnika (opracowanie własne)	102
Tabela 6. Uwagi uczestników badania pilotażowego oraz zastosowane rozwiązania zgłaszanych problemów (opracowanie własne)	126
Tabela 7. Uwagi uczestników badania pilotażowego, które nie mogły zostać poprawione wraz z komentarzem (opracowanie własne)	129
Tabela 8. Statystyki dotyczące danych demograficznych, ich doświadczeń i umiejętności, a także ich wyborów na wczesnym etapie badania (opracowanie własne)	150
Tabela 9. Informacje podstawowe o eksperymencie 1.	163
Tabela 10. Informacje podstawowe o eksperymencie 2a.	171
Tabela 11. Informacje podstawowe o eksperymencie 2b.	178
Tabela 12. Informacje podstawowe o eksperymencie 3.	182
Tabela 13. Informacje podstawowe o eksperymencie 4.	187
Tabela 14. Informacje podstawowe o eksperymencie 5.	194
Tabela 15. Informacje podstawowe o eksperymencie 6.	200
Tabela 16. Informacje podstawowe o eksperymencie 7	209

Spis załączników

Załącznik 1. Opis architektury i funkcjonalności aplikacji „Mapa PW”

Załącznik 2. Statystyki oraz wizualizacja wszystkich modeli budynków wykorzystanych w symulatorze badawczym

Załącznik 3. Pliki instalacyjne umożliwiające uruchomienie symulatora na własnym komputerze (załącznik elektroniczny)

Załącznik 4. Nagranie ekranu podczas przykładowej realizacji całości badania (załącznik elektroniczny)

Załącznik 5. Dokumentacja interfejsu API umożliwiającego komunikację między symulatorem, a aplikacją nawigacyjną (załącznik elektroniczny)

Załącznik 6. Skrypty języka Python wykorzystane do przetwarzania i analizy zebranych danych (załącznik elektroniczny)

Załącznik 1. Opis architektury i funkcjonalności aplikacji „Mapa PW”

Aplikacja „Mapa PW” oraz zasilająca ją baza danych powstały w latach 2022-2024 w ramach realizowanego na Wydziale Geodezji i Kartografii zadania pt. “Mapy Dostępności Budynków PW” będącego elementem ogólnouczelnianego projektu pt. "PW ambasadorem innowacji na rzecz dostępności". Projekt był współfinansowany ze środków Unii Europejskiej w ramach Europejskiego Funduszu Społecznego, Programu Operacyjnego Wiedza Edukacja Rozwój, Oś priorytetowa III – Szkolnictwo Wyższe dla gospodarki i rozwoju, Działanie 3.5. Kompleksowe programy szkół wyższych (POWR.03.05.00-00-A022/19). Autor rozprawy był jednym z głównych wykonawców wspomnianego zadania i głównym autorem aplikacji webowej dostępnej na urządzeniach mobilnych i na kioskach internetowych. Dzięki temu istniała możliwość rozbudowania aplikacji na potrzeby niniejszej rozprawy.

Opis funkcjonalności aplikacji oraz trybów jej wykorzystania

Mapa PW to aplikacja webowa, czyli taka, która uruchamiana jest z wykorzystaniem przeglądarki internetowej. Jest dostosowana do uruchamiania w trybie dla komputerów PC oraz urządzeń przenośnych, a także tzw. kiosków informacyjnych (Rysunek 0.1). Obecnie w często uczęszczanych miejscach kampusu uczelni rozstawiono sześć takich kiosków. Konstrukcja kiosków jest również przystosowana do osób z niepełnosprawnościami – możliwa jest zmiana kąta pochylenia ekranu oraz dostępna jest klawiatura z nadrukiem alfabetu Braille’a.



Rysunek 0.1. Kiosk informacyjny umiejscowiony w Gmachu Głównym Politechniki Warszawskiej (opracowanie własne)

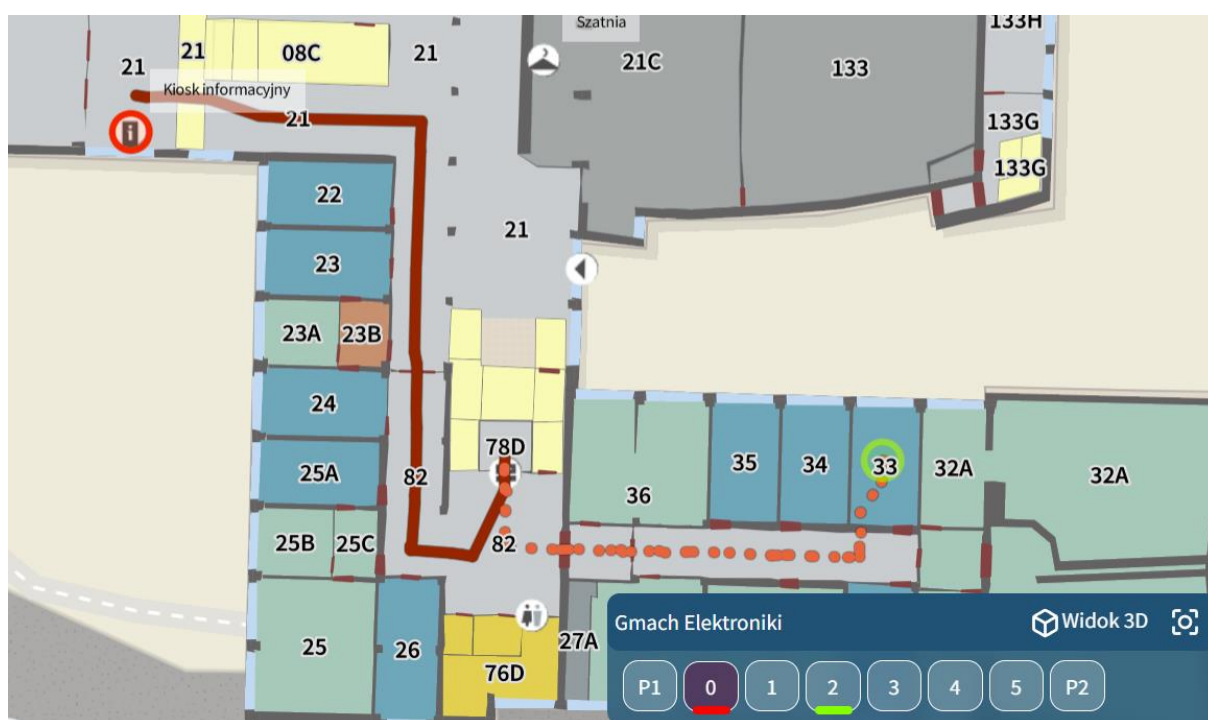
Podstawowa funkcjonalność aplikacji to wyświetlanie mapy 2D Politechniki Warszawskiej, a także możliwość wyszukiwania pomieszczeń oraz punktów POI (ang. *Points of Interest*). W wersji dla kiosku informacyjnego istnieje możliwość wyświetlania widoku 3D budynku oraz wyznaczania trasy od miejsca posadowienia urządzenia do wybranego miejsca docelowego (Rysunek 0.2).



Rysunek 0.2. Aplikacja Mapa PW z wyznaczoną trasą na mapie (opracowanie własne)

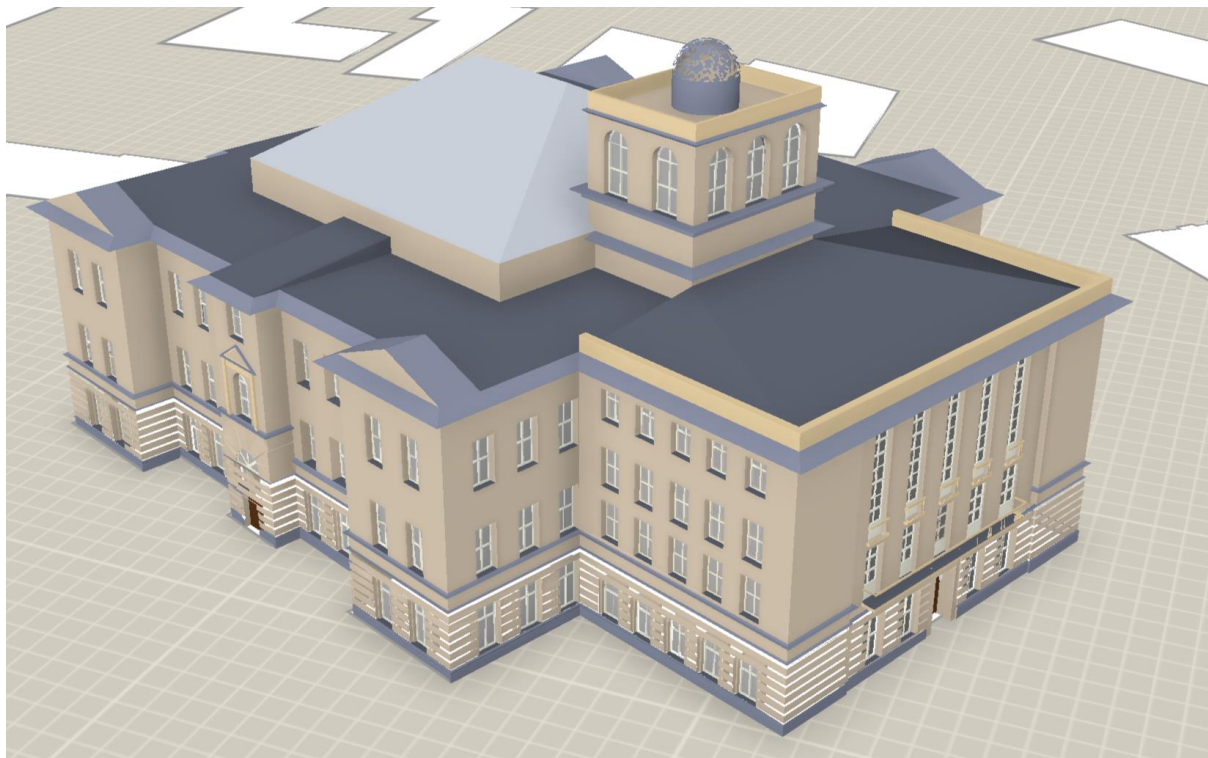
Aktualnie wyświetlane piętro w budynku można zmienić z wykorzystaniem widżetu wybierania piętra umieszczonego w prawym dolnym rogu obszaru mapy.

Wyznaczona trasa oznaczona jest kolorem czerwonym, przy czym odcinek trasy znajdujący się na tym samym piętrze co piętro aktualnie wybrane symbolizuje linia ciągła o ciemnej barwie, a piętro aktualnie wybrane linia kropkowana o jasnej barwie. Punkt startowy oraz punkt końcowy trasy oznaczono na mapie okręgami w kolorach kolejno czerwonym oraz zielonym. Piętra, na których znajdują się te punkty oznaczono na widżecie wybierania piętra odpowiadającymi kolorami (Rysunek 0.3).



Rysunek 0.3. Widok widżetu wybierania piętra oraz wyznaczonej trasy wraz z punktami startu i końca trasy (opracowanie własne)

Z poziomu widżetu wybierania piętra można również uruchomić widok 3D, który uruchamia okno z interaktywną, trójwymiarową wizualizacją fasady budynku (Rysunek 0.4), która pomaga w zwiększeniu świadomości sytuacyjnej użytkownika szukającego trasy dojścia do określonego miejsca w budynku.



Rysunek 0.4. Wizualizacja modelu Gmachu Fizyki w aplikacji Mapa PW (opracowanie własne)

Dodatkowo aplikacja wyposażona jest w moduł „Ułatwień dostępności” przeznaczony dla osób z niepełnosprawnościami. Umożliwia on wyświetlenie specjalnych geo-opisów obszarowych oraz punktowych (lokalizacyjnych, ostrzegających oraz poznawczych) (Gotlib i Świech 2023). Ponadto geo-opisy mogą zostać odczytane z wykorzystaniem syntezy mowy, a wybrane z nich również wyświetlone w Polskim Języku Migowym (Rysunek 0.5).



Rysunek 0.5. Widok modułu "Ułatwień dostępności" w aplikacji Mapa PW (opracowanie własne)

Wersja mobilna aplikacji (czyli wyświetlana w przypadku uruchomienia w przeglądarce internetowej urządzenia mobilnego typu smartphone lub tablet) zapewnia szybki dostęp do kluczowych funkcjonalności tj. mapy budynków oraz wyszukiwarki pomieszczeń. Dodatkowo może być uruchomiona poprzez skanowanie kodu QR lub tagu NFC, które rozmieszczone są na

tablicach informacyjnych znajdujących się na terenie kampusu oraz na drzwiach w wybranych budynkach. W tej sytuacji użytkownik zobaczy oznaczenie swojej lokalizacji na mapie (Rysunek 0.6).



Rysunek 0.6. Aplikacja Mapa PW uruchomiona na urządzeniu mobilnym z aktualną pozycją oznaczoną na podstawie skanowania kodu QR (opracowanie własne)

Architektura aplikacji

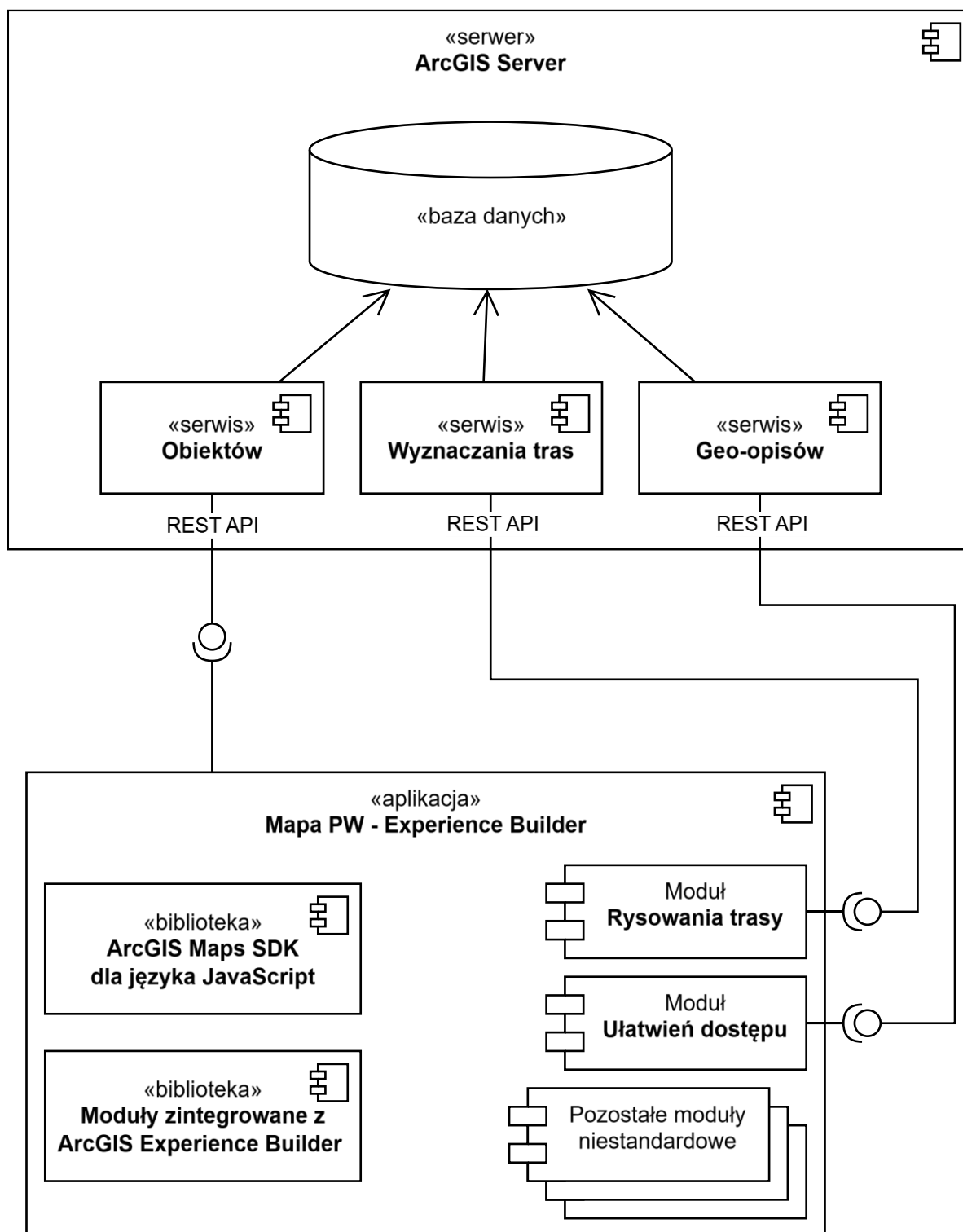
Aplikacja *Mapa PW* została zbudowana w oparciu o środowisko ArcGIS Experience Builder Developer Edition, które stanowi elastyczną platformę służącą do tworzenia interaktywnych aplikacji webowych wykorzystujących dane przestrzenne. Experience Builder zapewnia

podejście modularne, w którym poszczególne widżety stanowią samodzielne komponenty realizujące określone funkcje interfejsu użytkownika oraz komunikacji z warstwą danych. W prezentowanym rozwiązaniu platforma ta została rozszerzona o własne moduły opracowane w technologii React, co umożliwiło dostosowanie logiki i działania aplikacji do specyficznych wymagań (Esri 2020).

Aplikacja *Mapa PW* korzysta z ArcGIS Server pełniącego rolę zaplecza serwerowego (backendu), odpowiadającego za udostępnianie usług mapowych, serwisu wyznaczania trasy oraz danych wektorowych i rastrowych. Po stronie klienta (frontend) dane te są pozyskiwane za pośrednictwem interfejsów REST API oraz biblioteki ArcGIS Maps SDK for JavaScript, umożliwiającej dynamiczne renderowanie map i warstw danych przestrzennych bezpośrednio w przeglądarce.

Każdy z opracowanych autorskich modułów stanowi niezależny komponent, implementujący określony interfejs narzucony przez środowisko Experience Builder. Dzięki temu możliwe było zachowanie pełnej kompatybilności z istniejącymi widżetami oraz mechanizmami konfiguracji aplikacji, takimi jak zarządzanie stanem, komunikacja między widżetami oraz integracja z systemem motywów (themes). Najważniejsze z zaimplementowanych funkcji opisano w rozdziale 6.10.

Schemat logiczny aplikacji przedstawiono na diagramie UML (Rysunek 0.7), prezentuje główne komponenty oraz powiązania pomiędzy warstwą aplikacji i serwerem danych.



Rysunek 0.7. Diagram architektury aplikacji Mapa PW (opracowanie własne, na podstawie (Gotlib, Lipka, i Świech 2025))

Na diagramie przedstawiono kluczowe elementy architektury:

- **ArcGIS Server** – źródło danych przestrzennych, publikujące usługi mapowe, usługi danych oraz usługi wyznaczania trasy.
- **Warstwa aplikacji (Experience Builder)** – rdzeń aplikacji, odpowiedzialny za wizualizację map, interfejs użytkownika, zarządzanie modułami i logiką interakcji.
- **Moduły niestandardowe** – rozszerzenia Experience Builder implementujące ponadstandardowe funkcjonalności.
- **ArcGIS Maps SDK for JavaScript** – biblioteka rozwijana przez firmę Esri, zapewniająca wysokopoziomowe API do pracy z mapami, warstwami, danymi oraz usługami przestrzennymi w czasie rzeczywistym.
- **Warstwa komunikacji (REST API)** – kanał wymiany danych pomiędzy klientem a serwerem ArcGIS, realizujący zapytania HTTP typu GET/POST dotyczące danych przestrzennych, metadanych i analiz.

Architektura aplikacji jest w pełni modularna, co umożliwia zarówno skalowalność, jak i łatwą modyfikację poszczególnych komponentów. Dzięki podejściu opartemu na Experience Builder i React, dodanie nowego widżetu wymaga jedynie implementacji interfejsu obsługującego stan aplikacji oraz rejestracji komponentu w środowisku konfiguratora. W ten sposób możliwa jest rozbudowa systemu o nowe funkcje bez ingerencji w istniejący kod.

Integracja warstw i przepływ danych

Proces przepływu danych przebiega zgodnie z następującą sekwencją:

1. Użytkownik wykonuje akcję w interfejsie (np. wybór budynku).
2. Jeden z modułów aplikacji wysyła zapytanie do ArcGIS Server przez REST API.
3. ArcGIS Server zwraca wynik w znanym formacie.
4. Moduł, który otrzymał dane z serwera przekazuje informacje (np. listę dostępnych pomieszczeń i punktów POI) do innych komponentów poprzez kontekst aplikacji lub mechanizmy komunikacji między widżetami.
5. Biblioteka ArcGIS Maps SDK renderuje zaktualizowane dane na mapie w czasie rzeczywistym.

Tak zaprojektowana architektura umożliwia płynne działanie aplikacji w środowisku przeglądarkowym, a jednocześnie zapewnia bezpośredni dostęp do złożonych operacji przestrzennych oferowanych przez ArcGIS Server. Wykorzystanie standardów REST i komponentowej struktury React zapewnia niezależność od warstwy danych oraz możliwość łatwej integracji z zewnętrznymi źródłami przestrzennymi lub narzędziami analitycznymi.

Struktura danych mapowych oraz sposób ich serwowania w technologii ArcGIS Server

Mapy budynków generowane są na podstawie bazy danych stanowiącej podstawowy komponent Systemu Informacji o Nieruchomościach Politechniki Warszawskiej (SION PW). Została ona zaprojektowana jako zaawansowane narzędzie o wielorakim przeznaczeniu, służące nie tylko do wizualizacji struktury budynków oraz wyznaczania tras, ale również do zarządzania nieruchomościami, planowania remontów, archiwizacji danych o budynkach oraz prowadzenia szczegółowej inwentaryzacji wszystkich elementów infrastruktury budynków. Ze względu na szeroki zakres funkcjonalności, struktura całej bazy danych charakteryzuje się wysokim stopniem złożoności. Projekt bazy danych zawiera 229 klas obiektów, które w sumie mają nadanych 3024 atrybutów.

Baza ta była wykorzystywana w tworzeniu symulatora badawczego. Na potrzeby niniejszej pracy przedstawiono jedynie najistotniejsze tabele wykorzystywane w procesie wyświetlania map budynków.

Istotnym aspektem architektury bazy danych jest hierarchiczny układ przechowywanych informacji. Poszczególne poziomy struktury organizacyjnej są wzajemnie powiązane:

- Budynki przypisane są do klastrów budynków
- Piętra przypisane są do budynków
- Pomieszczenia, punkty POI, ściany, drzwi, okna oraz wszystkie inne elementy i instalacje wewnętrzne przypisane są do poszczególnych pięter

Taka struktura zapewnia logiczną organizację danych oraz umożliwia ich wydajne filtrowanie, co jest kluczowe dla procesu renderowania wizualizacji.

ArcGIS Server umożliwia aplikacjom dostęp do informacji zapisanych w bazie danych poprzez udostępnienie tzw. warstwy (ang. *layer*). W tym przypadku wykorzystana została warstwa obiektowa (ang. *feature layer*), a komponentem odpowiedzialnym za jej udostępnienie jest serwis obiektowy (ang. *feature service*), który w nomenklaturze ArcGIS Server stanowi technologię serwowania danych wektorowych poprzez interfejs REST API (Esri 2010). Udostępniana jest geometria obiektów, a także atrybuty oraz informacje o symbolizacji obiektów geograficznych. W przeciwieństwie do Map Image Layer (pl. *warstwa obrazowa mapy*), który generuje i przesyła gotowe obrazy rastrowe (kafelki lub dynamicznie renderowane mapy), Feature Service dostarcza surowe dane, które są następnie renderowane po stronie aplikacji klienckiej.

Wybór Feature Service zamiast Map Service w kontekście map wnętrz budynków jest podyktowany kilkoma istotnymi zaletami technologicznymi:

- Feature Service umożliwia bezpośredni dostęp do geometrii i atrybutów poszczególnych pomieszczeń, korytarzy czy innych elementów budynku. Klient otrzymuje nie obraz, ale strukturalne dane zawierające wszystkie informacje o obiekcie (numer pomieszczenia, piętro, powierzchnia, typ)
- Technologia umożliwia wykonywanie zaawansowanych zapytań SQL bezpośrednio na poziomie serwisu. W przypadku filtrowania danych według piętra, zapytanie do bazy danych jest wykonywane po stronie serwera, co znacząco redukuje ilość przesyłanych danych przez sieć. Dzięki temu aplikacja otrzymuje tylko dane dotyczące aktualnie wyświetlanego poziomu budynku.
- W przeciwieństwie do Map Service, gdzie symbolizacja jest ustalana na serwerze, Feature Service pozwala na dynamiczne zmiany wyglądu obiektów po stronie aplikacji klienckiej.
- Feature Service serwuje dane bezpośrednio z bazy danych, dzięki czemu są one aktualizowane w czasie rzeczywistym, bez konieczności ponownego generowania kafelków.

Feature Service wykorzystuje architekturę REST API, gdzie komunikacja odbywa się poprzez standardowe żądania HTTP GET/POST. Odpowiedzi serwisu są zwracane w formacie zakodowanym base64, z którego potrafią wydajnie skorzystać biblioteki ArcGIS dla różnych platform. Rozwiązanie to jest komercyjnym odpowiednikiem otwartego standardu OGC WFS (Open Geospatial Consortium 2002).

Należy zauważyć, że Feature Service ma istotne ograniczenie względem rozwiązań rastrowych. Map Service może być buforowany w postaci kafelków, co znacząco przyspiesza dostarczanie danych. Feature Service natomiast przesyła surowe dane geometryczne, co generuje większy ruch sieciowy i potencjalnie dłuższy czas oczekiwania na pierwsze załadowanie mapy. Raz pobrane dane mogą natomiast być lokalnie buforowane po stronie klienta, ograniczając konieczność ponownego pobierania przy każdym wyświetleniu tego samego piętra lub obszaru.

W środowisku budynków wielokondygnacyjnych ta sama lokalizacja w układzie X,Y może odnosić się do wielu różnych obiektów przestrzennych znajdujących się na różnych poziomach. W tradycyjnym środowisku GIS obie te geometrie zostaną wyrenderowane w tej samej lokalizacji na mapie, powodując wizualny konflikt polegający na nakładaniu się wielokątów reprezentujących pomieszczenia. Użytkownik przeglądający taką mapę zobaczy jedynie górną warstwę danych, podczas gdy pozostałe piętra będą całkowicie lub częściowo zasłonięte. Chociaż współrzędna Z mogłaby teoretycznie rozwiązać ten problem (wprowadzenie trzeciego wymiaru), większość systemów GIS nie wykorzystuje tej informacji podczas renderowania map. Nawet jeśli geometrie przechowują informację o wysokości, standardowe narzędzia do wizualizacji map ignorują ten atrybut podczas wyświetlania danych na płaszczyźnie mapy.

Efektywnym rozwiązaniem problemu nakładania się danych wielopiętrowych jest implementacja mechanizmu filtrowania po stronie serwera na poziomie zapytania do Feature Service. Podejście to polega na selektywnym pobieraniu wyłącznie tych obiektów przestrzennych, które należą do aktualnie wyświetlanego piętra, eliminując w ten sposób konflikt wizualny i redukując ilość przesyłanych danych.

W strukturze bazy danych każdy obiekt przestrzenny (np. pomieszczenie, korytarz, klatka schodowa) posiada atrybut liczbowy identyfikujący przynależność do konkretnego piętra. Do każdego zapytania dodawany jest parametr `WHERE` zawierający filtr zgodny ze standardem SQL, np.:

```
budynek_id = 27 AND poziom IN (1)
```

Pełne zapytanie REST API do Feature Service zawiera szereg dodatkowych parametrów optymalizujących przesyłanie i przetwarzanie danych.

Architektura backendu opartego na ArcGIS Server i technologii Feature Service stanowi fundament dla efektywnego zarządzania danymi przestrzennymi wielopiętrowych budynków. Kluczowym elementem rozwiązania jest przechowywanie danych w bazie z atrybutem identyfikującym przynależność do konkretnego piętra oraz serwowanie ich poprzez Feature Service z wykorzystaniem filtrowania na poziomie zapytań SQL. By skorzystać z tych danych aplikacja kliencka musi implementować mechanizm filtrowania.

Załącznik 2. Statystyki oraz wizualizacja wszystkich modeli budynków wykorzystanych w symulatorze badawczym

Budynek: Centrum Analiz Geoprzestrzennych i Satelitarnych w Józefosławiu (CENAGIS)		
Wykorzystane narzędzie	Liczba obiektów	Liczba trójkątów
SketchUp	836	298 262



Budynek: Gmach Główny PW		
Wykorzystane narzędzie	Liczba obiektów	Liczba trójkątów
Revit	11 384	6 140 792
		

Gmach Elektroniki		
Wykorzystane narzędzie	Liczba obiektów	Liczba trójkątów
Revit	8 638	21 811 855
		

Gmach Elektrotechniki		
Wykorzystane narzędzie	Liczba obiektów	Liczba trójkątów
SketchUp	3 926	7 804 603



Gmach Chemii		
Wykorzystane narzędzie	Liczba obiektów	Liczba trójkątów
Revit	4 976	10 926 888



Gmach Architektury		
Wykorzystane narzędzie	Liczba obiektów	Liczba trójkątów
Revit	6 959	10 939 531

