



Politechnika Wrocławska

**Filia w Jeleniej Górze**

kierunek studiów: Informatyka przemysłowa

specjalność: Inteligentne systemy przemysłowe

## Praca dyplomowa - inżynierska

### Aplikacja mobilna z elementami rozszerzonej rzeczywistości

Andrzej Klimek

słowa kluczowe:  
Rozszerzona rzeczywistość  
Aplikacja mobilna  
Geolokacja

krótkie streszczenie:  
Aplikacja wykorzystująca  
technologie geolokacji oraz  
rozszerzonej rzeczywistości

opiekun pracy	.....		
dyplomowej	Tytuł/stopień naukowy/imię i nazwisko		
Ostateczna ocena za pracę dyplomową			
Przewodniczący Komisji egzaminu dyplomowego	.....	.....	.....
	..	..	..
	Tytuł/stopień naukowy/imię i nazwisko	ocena	podpis

Do celów archiwalnych pracę dyplomową zakwalifikowano do:\*

a kategorii A (akta wieczyste)

b kategorii BE 50 (po 50 latach podlegające ekspertyzie)

\* niepotrzebne skreślić

pieczęćka wydziałowa

Wrocław, 2021



## Spis treści

Praca dyplomowa - inżynierska.....	1
Aplikacja mobilna z elementami rozszerzonej rzeczywistości.....	1
Streszczenie pracy inżynierskiej.....	4
1. Wprowadzenie.....	5
1.1. Cel i zakres pracy.....	6
2. Rzeczywistość rozszerzona(ang. Augmented reality).....	7
2.1. Czym jest Rzeczywistość rozszerzona?.....	7
2.2. Urządzenia wykorzystywane do projekcji augmented reality.....	8
2.3. Różnica pomiędzy Augmented reality a Virtual reality.....	10
2.4. Jak dokładnie działa technologia rozszerzonej rzeczywistości ?.....	11
2.5. Przykłady zastosowań augmented reality.....	13
2.6. Czym jest Geolokacja i jak działa?.....	14
3. Narzędzia wykorzystane do wykonania projektu.....	15
3.1. Mapbox API & SDK.....	15
3.2. Unity game engine.....	15
3.3. GIMP(Gnu Image Manipulation Program).....	16
3.4. Blender.....	16
4. Opis aplikacji i jego wykonanie.....	17
4.1. Wykonanie aplikacji.....	17
4.2. Instrukcja dotycząca gry.....	24
5. Podsumowanie i wnioski.....	28
6. Bibliografia.....	30
7. Spis ilustracji.....	31
8. Załączniki.....	32

## **Streszczenie pracy inżynierskiej.**

W niniejszej pracy inżynierskiej celem było stworzenie aplikacji wykorzystującej moduł geolokacji oraz wykorzystanie technologii rozszerzonej rzeczywistości. Taka aplikacja została stworzona i dołączona do pracy. Wyjaśniono jej działanie oraz zastosowanie.

Zaprezentowano fragmenty kodu oraz wytłumaczono do czego została użyta geolokacja. Dodatkowo zostały omówione narzędzie wykorzystane do zrealizowania programu.

W dokumencie tak że zostały poruszony temat rozszerzonej rzeczywistości w zakresie jej funkcjonalności, szczegółów dotyczących działania oraz zastosowania na różnych urządzeniach.

## 1. Wprowadzenie

Wdrażanie nowych i coraz bardziej zaawansowanych technologii ułatwiających pracę w wielu dziedzinach następuje ostatnimi latami coraz szybciej. Takie technologie nie tylko są używane do bardzo precyzyjnych lotów kosmicznych czy też zaawansowanych projektów inżynierskich gdzie minimalny błąd może mieć katastrofalne skutki w przełożeniu na zastosowanie go w realnym życiu. Ale także są stosowane do prostych czynności i codziennego użytku gdzie wielu użytkowników nawet nie jest tego świadoma.

Mimo tego że niektóre technologie jak dotykowe telefon komórkowe dziś się wydają niczym specjalnym, tak paręnaście lat temu było to coś z czym ludzie spotykali się tylko w filmach. Sama myśl że coś co dziś jest łatwo dostępne w przeszłości mogło być przeznaczone tylko dla osób wyjątkowo bogatych. Innym przykładem mogą być takie tablety elektroniczne, które kiedyś będąc znakiem biznesmenów dziś są dostępne bez problemu w sklepach po bardzo rozsądnych cenach dla przeciętnego szarego człowieka.

Kolejnym przykładem niewyobrażalnego postępu technologicznego jest coś czego większość ludzi dziś jest niekoniecznie świadoma, targetowane reklamy. To wcale nie jest przypadek kiedy ktoś szuka opon zimowych do auta, a na następny dzień czy też po paru godzinach większość reklam prezentuje mu właśnie ten produkt. Wszystko to jest oparte o zaawansowane algorytmy, które były doskonalone latami aż po dzień dzisiejszy. Podobnie działają sugestie proponowane nam przez wyszukiwarkę internetową którą używamy na co dzień. Ułatwia nam to codzienną pracę czy też pozwala odnajdywać rzeczy, które pokrywają się z naszymi zainteresowaniami.

Można powiedzieć że algorytm odpowiedzialny za to na bieżąco się nas uczy. Inteligentne aplikacje dostosowujące się pod użytkownika są dziś na porządku dziennym. Samo istnienie programów będących w stanie przeprowadzać rozmowę w realnym czasie z człowiekiem dowodzi tego. Tak że technologia jutra, która zaczyna gościć w naszych czasach jak wszelkiego rodzaju pojazdy potrafią się przemieszczać samodzielnie bez kierowcy na zatłoczonych ulicach.

Grafika trójwymiarowa podbiła branżę rozrywkową z wielką łatwością jako że odbiorcy szukają coraz to nowszych doświadczeń. Co prawda grafika 3D jest z nami już od długiego czasu była ważnym krokiem w stronę przeniesienia nas do świata wirtualnego

w postaci wirtualnej rzeczywistości. Lecz zamiast uciekać do świata wirtualnego dlaczego nie wyciągnąć go do świata rzeczywistego ? Taka technologia już istnieje i mianuje się nazwą rozszerzonej rzeczywistości.

### **1.1. Cel i zakres pracy:**

Celem pracy jest stworzenie aplikacji typu rozszerzona rzeczywistość (augmented reality), która w wybranych lokalizacjach geograficznych będzie na ekranie smartfona wyświetlała "ukryte elementy krajobrazu" zawierające wskazówki do gry w warcaby, "poszukiwania skarbu" czy odnajdywania geoskrytek. Zakres pracy obejmuje:

- Stworzenie modułu rzeczywistości rozszerzonej
- Stworzenie modułu geolokalizacji
- Stworzenie modułu prezentacji graficznej na zdjęciach / ekranie urządzenia mobilnego
- Stworzenie modułu administracyjnego i konfiguracyjnego

Praca składa się z sześciu rozdziałów wliczając to wprowadzenie.

W drugim rozdziale zostaje poruszony temat technologii rozszerzonej rzeczywistość. Jest tam mowa o jej szczegółowym działaniu oraz zostają przytoczone przykłady zastosowania.

Trzecim rozdział został poświęcony w celu opisanie geolokacji i krótkiemu opisowi jak ona działa.

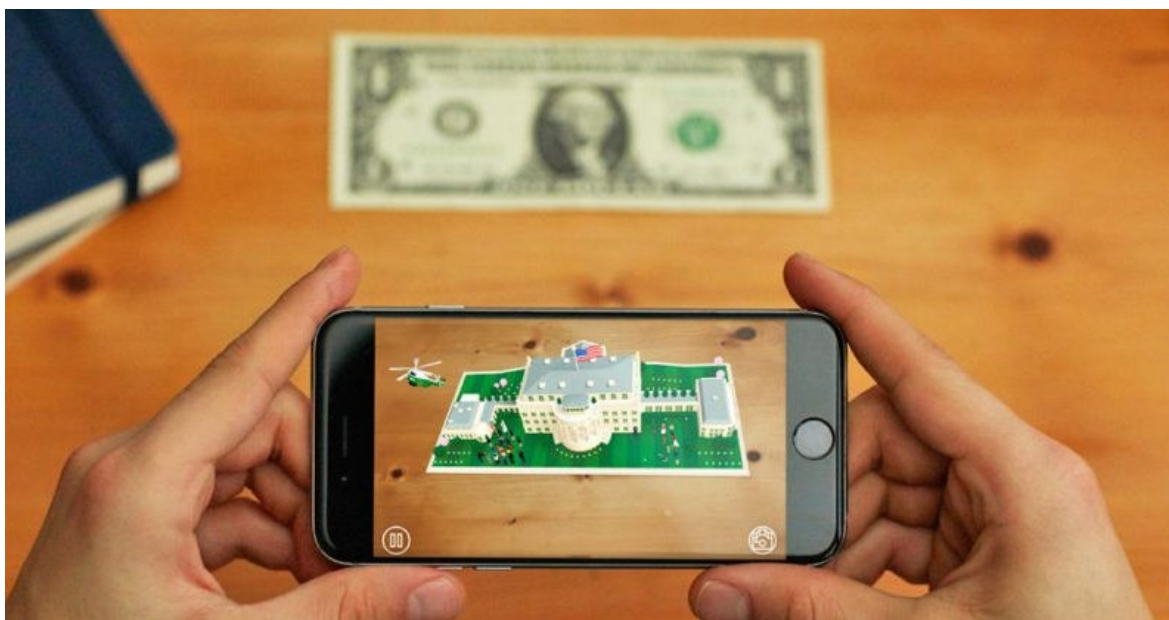
Natomiast w czwartym wymienione zostają narzędzia zastosowane do wykonania aplikacji załączonej do pracy inżynierskiej. Również zostają one szczegółowo opisane.

Przedostatni rozdział porusza tematykę samego programu stworzonego dla tematu oraz opisuje jej działanie. Tak że jest rozpisana instrukcja dla użytkownika.

## 2. Rzeczywistość rozszerzona(ang. Augmented reality).

### 2.1. Czym jest Rzeczywistość rozszerzona?

Jest ona interaktywnym połączeniem obiektów znajdujących się zarówno w realnym świecie jak i wygenerowanych przez komputer. Kiedy używamy Augmented Reality (z angielskiego rzeczywistość rozszerzona) dodaje ona wirtualnie wygenerowane obiekty w czasie rzeczywistym na obraz świata. Najłatwiej porównać ją do filtra który nakładamy na świat poprzez użycie aparatu/kamery lub innego sprzętu elektronicznego. Obiekty w ten sposób nałożone mogą być w pełni interaktywne z użytkownikiem lub pełnić po prostu funkcję informacyjną. Dla przykładu przy użyciu takiej aplikacji i naszego telefonu jesteśmy w stanie zobaczyć dymki nad punktami zainteresowania. Takimi jak zabytki, parki czy też ważne budynki administracyjne aż po restauracje. Takie dymki mogą zawierać wszelkie szczegółowe informacje obiektu nad którym się znajdują jak popularność odwiedzanego miejsca czy też godziny otwarcia oraz cennik. Przykład użycia technologii możemy zobaczyć na rysunku 2.1 gdzie jest wyświetlana ona przy pomocy smartfona. Na banknocie amerykańskiego dolara pojawia się cyfrowy model białego domu.



*Rysunek 2.1: Zdjęcie przedstawiające wykorzystanie aplikacji typu AR w celu wyświetlenia białego domu na banknocie amerykańskiego dolara.*

(Źródło: <https://www.linx.com/blog/augmented-reality-how-can-it-be-used-to-enhance-your-brand/> )

Augmented reality aktualnie przeżywa bardzo wysokie zainteresowanie co za tym idzie, jest kładziony duży nacisk na jego rozwijanie. Największym skokiem w popularności stosowania jej jest aplikacja z 2016 roku o nazwie Pokemon Go, która w przeciągu paru dni zdobyła wielką bazę fanów. Trzeba zaznaczyć że technologia jest używana w bardzo wielu zróżnicowanych dziedzinach życia.

## **2.2 Urządzenia wykorzystywane do projekcji augmented reality.**

Jako że do używania technologii rozszerzonej rzeczywistości urządzenie nie musi się składać z bardzo dużej ilości elementów można ją wykorzystać na wielu sprzętach. Wszystko czego potrzebuje takie urządzenie to Input device w postaci kamery w celu odczytania obrazu lub inny podzespół wejścia, wyświetlacz oraz procesor do przetwarzania danych.

Z tego też powodu istniały próby wdrożenia i spopularyzowania wykorzystania augmented reality między innymi przy zastosowaniu okularów tak zwanych smart glasses. Takie okulary miały wyświetlać konfigurowalne informacje. Niestety nie weszły one do powszechnego użytku na dzień dzisiejszy, choć ich budowa i funkcje nadal się rozwijają. Przykład takich okularów możemy zobaczyć na rysunku 2.2



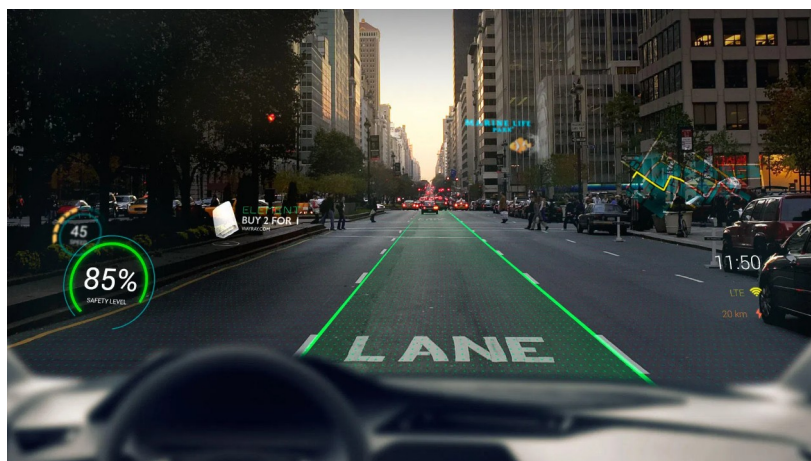
*Rysunek 2.2: Zdjęcie przedstawiające smart glasses.*

(Źródło: <https://www.eenewseurope.com/news/google-buys-ar-canadian-smart-glasses-startup>)

Jednym z aktualnie coraz bardziej popularnych trendów jest zastosowanie modułu w pojazdach osobowych i nie tylko. Perfekcyjnym przykładem są auta Tesli. Gdzie technologia jest szeroko stosowana w podzespołach nawigacyjnych dla auta i nie tylko.



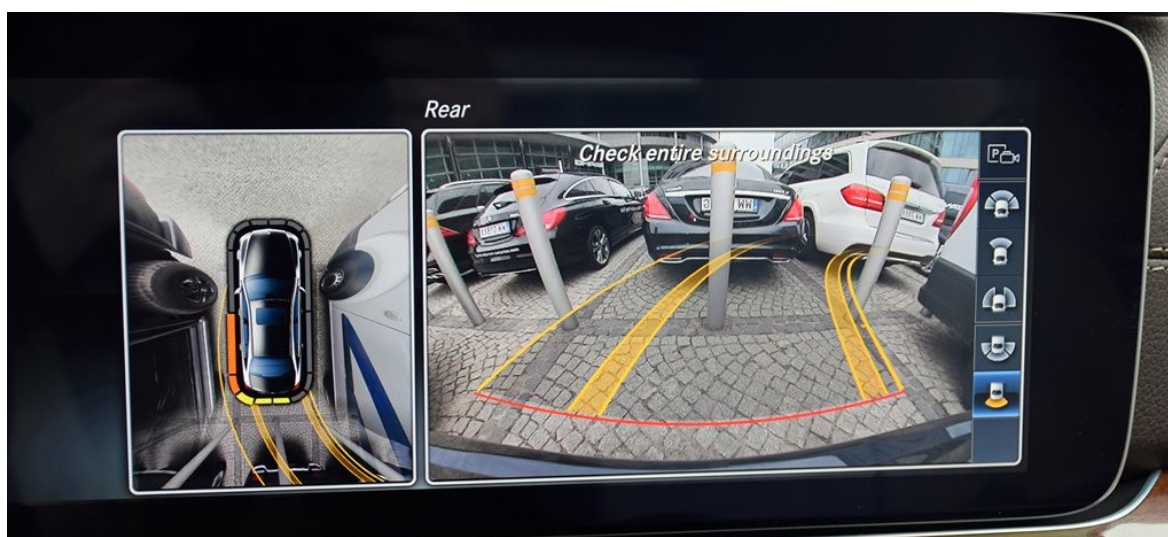
Użycie AR w smart windshield (inteligenta przednia szyba auta) umożliwia na pokazywanie prędkościomierza czy też mapy jak na rysunku 2.3. Takie zastosowanie powoduje zmniejszenie rozpraszania kierowcy, który nie musi już patrzeć na nawigację GPS lub też telefon na, którym ową nawigację może mieć włączoną. Przekłada się to na znaczne zwiększenie bezpieczeństwa jazdy.



*Rysunek 2.3: Zdjęcie przedstawiające przykładowe informacje które mogą się znajdować na smart windshield.*

(Źródło: <https://venturebeat.com/2017/08/23/ar-will-drive-the-evolution-of-automated-cars/>)

Od paru lat wykorzystuje się rozszerzoną rzeczywistość w kamerach służących do pomocy w parkowaniu. Idealnie prezentują to kamery pokazujące obraz w obrębie 360 stopni. Taka aplikacja odczytuje obraz z 4 lub więcej kamer w celu zaprezentowania auta z widoku trzeciej osoby. Taki sprzęt może również zasymulować tor jazdy przy skręceniu kół auta jak widać to na rysunku 2.4.



Rysunek 2.4: Zdjęcie urządzenia, które wyświetla obraz z kamer umieszczonych po każdej stronie auta, i próbuje zwizualizować widok z trzeciej osoby.

(Źródło: <https://www.whichcar.com.au/car-advice/360-degree-parking-monitors-explained>)

### 2.3. Różnica pomiędzy Augmented reality a Virtual reality.

Osoby nie znające tematu często myślą że pojęcia rzeczywistości rozszerzonej a rzeczywistości wirtualnej oznaczają to samo, co nie jest do końca prawdą. Co prawda podane wcześniej terminy są sobie dość bliskie a nawet istnieje ich skrzyżowanie. Zasadniczą różnicą między tymi dwoma systemami rzeczywistości jest fakt, że o ile rzeczywistość rozszerzona łączy ze sobą elementy świata wirtualnego i rzeczywistego, to rzeczywistość wirtualna próbuje w sposób immersyjny przenieść użytkownika w świat sztucznie wygenerowany przez program. Najczęściej odbywa się to za pomocą specjalnych gogli VR. Idealnym przykładem i prawie pokrywającym się z sposobem dzisiejszego “wchodzenia” do takiej symulacji jest film z 1999 roku pod tytułem “The Matrix”.

Przedstawienie takiego wygenerowanego świata wirtualnego może się pokrywać wyglądem i formą przedstawienia z rzeczywistym światem, ale może być to tak że całkiem fantastycznie wykreowany świat.

## 2.4. Jak dokładnie działa technologia rozszerzonej rzeczywistości ?

Technologia rozszerzonej rzeczywistości opiera swoją funkcjonalność na znacznikach (markerach). Aktualnie wyróżnia się trzy technologie:

- znacznikową,
- bez znacznikową,
- pseudo znacznikową.

### 2.4.1 Technologia znacznikowa.

Technologia która, opiera swoje działanie na markerach odczytywanych z różnego rodzaju predefiniowanych znaków, ciągów symboli, grafik lub innych rodzajów danych wejściowych (Na przykład kodów QR). Na podstawie wielu skomplikowanych obliczeń oraz algorytmów które opisuje artykuł Kurpytė D. Navakauskas D. „An Efficiency Analysis of Augmented Reality Marker Recognition Algorithm”. Pozwala ona o wzbogacenie na przykład plakatu, wycinku z gazety i nie tylko. Po zczytaniu kształtu i ułożenia informacji z takiego markera nasze urządzenie jest w stanie stwierdzić gdzie ma się znajdować i jak ma wyglądać obiekt zapisany w nim, co za tym idzie jest w stanie nam go wyświetlić. Przykładem jest aplikacja pokazująca trójwymiarowy model budynku kiedy się zeskanuje jego plan przedstawiona na na rysunku 2.5



*Rysunek 2.5: Wykorzystanie aplikacji typu AR do przedstawienia trójwymiarowego budynku na podstawie jego planu.*

(Źródło: <https://www.whichcar.com.au/car-advice/360-degree-parking-monitors-explained>)

#### 2.4.2. Technologia bez znacznikowa.

Technologia opracowana w roku 2017 w której nie trzeba się posługiwać wcześniej predefiniowanymi punktami odniesienia typu kody QR i tym podobne.

Pozwala ona na umieszczeniu obiektu w dowolnym miejscu gdzie jest tylko możliwy punkt zaczepienia.

Należy pamiętać że bez znacznikowa rozszerzona rzeczywistość ironicznie nadal wykorzystuje znaczniki ponieważ obiekty które chcemy przedstawić muszą mieć jakiś punkt zaczepny w innym wypadku ich pojawienie się było by zupełnie nieprzewidywalne i losowe. Obiekty są tworzone na punktach które powstają na zasadzie wykrywania krawędzi na płaszczyznach i kolorystyki na przestrzeni widzianej przez naszą kamerę.

To właśnie ten typ augmented reality jest wykorzystywany w aplikacjach meblowych czy też do przedstawienia i dostosowaniu modelu auta, jak w fanowskim programie do prezentowania auta Tesli Modelu Trzeciego co widać na rysunku 2.6 gdzie możemy zmienić kolor czy też zmienić niektóre podzespoły takie jak felgi czy też wygląd przedniej maski.

#### 2.4.3. Technologia pseudo znacznikowa.

Polega ona zdefiniowaniu markera który pojawi się w losowej pozycji widzianej przez aparat naszego urządzenia. Przez co sprawia iluzję że obiekt generowany znajduje się



Rysunek 2.6: Przykład zastosowania rozszerzonej rzeczywistości do pokazania auta Tesli Model 3.

(Źródło: <https://www.whichcar.com.au/car-advice/360-degree-parking-monitors-explained>)

gdzieś w świecie rzeczywistym. Taka technologia została przykładowo użyta w popularnej grze z 2016 roku "Pokemon Go". Jako że aplikacja powstała przed rokiem 2017 kiedy to

wymyślono jak bez znaczników używać augmented reality, trzeba było wymyślić inny sposób, ponieważ rozstawienie znaczników na których obiekty jakimi były tytułowe pokemony mogłyby się pojawić oznaczałoby przedsięwzięcie na absurdalną skalę. Jako że ilość zasobów nie tylko materialnych ale także czasu na dokonanie tego było niebotyczna twórcy wpadli na takie rozwiązanie ówczesnego problemu. Dużym plusem takiej technologii jest fakt że urządzenia potrzebne do użycia jej nie musiały być bardzo nowe, więc przekładało się to do trafienia w większą grupę odbiorców.

## **2.5. Przykłady zastosowań augmented reality**

Jak już było to wspomniane zastosowanie technologii jest bardzo szerokie i nie ograniczy się tylko do gier czy nawigacji. Używa się jej na meczach piłki nożnej czy też skoków narciarskich do wyznaczania linii dla widzów jak zaprezentowania najdalszego skoku lub też pokazania w jakim polu znajdują się zawodnicy do analizowania lub zilustrowania taktyki.

Sieć sklepów IKEA posiada własną aplikację w której jesteśmy w stanie wyświetlić projekcję wybranego przez nas mebla. Dzięki temu jesteśmy w stanie sprawdzić czy dany mebel będzie pasować wielkościowo czy też stylistycznie w pokoju którym chcemy go postawić. Często jesteśmy też w stanie zmodyfikować taki obraz w celu zmienienia koloru czy też modelu.

Neurochirurdzy często używają ARu żeby wyświetlać trójwymiarowy mózg do pomocy w przeprowadzaniu operacji na nim.

W artykule Abramovich G. (2018) 5 „Innovative examples of augmented reality in action.” możemy wyczytać o innowacyjnym zastosowaniu technologii w wojsku i nie tylko. Piloci statków powietrznych używają specjalnych hełmów które przedstawiają im ważne informacje takie jak prędkość, wysokość czy inne ważne dla nich informacje. W ten sposób nie są oni rozproszeni ciągłym rozglądaniem się po kokpicie w poszukiwaniu informacji ważnych dla nich w danym momencie.

Na lotniskach ekipa zajmująca się ładowaniem bagażów może wykorzystać augmented reality w celu widzenia co się znajduje w danym kontenerze i jaki jest jego cel podróży co znacznie skraca czas potrzebny na załadunek towaru.



## 2.6. Czym jest Geolokacja i jak działa?

Geolokacja jest geograficzną (długością i szerokością) lokacją na której dane urządzenie jest połączone z internetem. Miejsce które aktualnie zajmujemy podczas korzystania z takiej technologii często jest kalkulowane poprzez dodatkowe informacje takie jak ulica, numer domu czy też miasto w naszym otoczeniu. Jako że korzysta ona z danych z twojego urządzenia jak i z sieci, oznacza to że tym więcej ludzi dookoła ciebie będzie z niej korzystać tym większa będzie dokładność pozycji w której się znajdujesz.

Wyznaczanie lokacji odbywa się za pomocą ustalenia adresu IP urządzenia czy też połączenia z GPS-em.

Technologia geolokacji ma wiele popularnych zastosowań takich jak:

- Programów przewidujących pogodę dla naszego miejsca pobytu.
- Do aplikacji typu GPS jak Google Maps czy innej nawigacji.
- W programach sportowych odmierzających jaki dystans użytkownik przebiegł czy też przejechał.
- Do zbierania informacji o najczęściej odwiedzanych przez nas lokacjach i generacji pozycji które mogą nas zainteresować na ich podstawie.

Ostatni podpunkt z listy wymienionej powyżej jest praktycznie głównym zastosowaniem geolokacji choć wydawać by się mogło inaczej.

Według artykułu stworzonego przez Janelle Nanos pod tytułem „Every step you take” na podstawie właśnie tej technologii korporacje z dużą łatwością mogą dopasowywać swoje oferty i zakres cenowy do naszych preferencji oraz zarobków.

Na podstawie pozycji urządzenia z którego na co dzień korzystamy czyli smartfona są one w stanie takie informacje jak ile spędzamy czasu w domu czy też stwierdzić gdzie żyjemy, co za tym idzie użytkownik żyjący z daleka od miasta lecz dojeżdżający do niego autobusem może zostać przywitany reklamą samochodu.

Takie wykorzystywanie geolokacji co prawda daje nam wiele udogodnień i może nam zaoferować produkty, które naprawdę mogą nam pomóc w codziennym życiu, ale także wiąże się z wieloma niebezpieczeństwami. Przykładowo takie często newralgiczne dane w rękach nieodpowiedniej firmy co za tym idzie ludzi, mogą zostać wykorzystane do narobienia nam wielu kłopotów. Jako że nie jesteśmy już więcej anonimowi w internecie.

### **3. Narzędzia wykorzystane do wykonania projektu.**

#### **3.1. Mapbox API & SDK.**

Mapbox jest zbiorem wszelkich interfejsów oraz innych narzędzi ułatwiających tworzenie aplikacji na iOS, Androida oraz innych platformach wykorzystujących GPS. Jest on dostępny do pobrania na jego oficjalnej stronie za darmo na bazie licencji. Zapełnia on niszę rynkową jeśli chodzi o kategorie tego typu dostarczanych usług, wraz z Google Maps.

Jako że jest to środowisko mocno otwarte na użytkowników nie brak jest wszelkich poradników lub też innych utworów literackich które pomogą nam w zrealizowaniu projektu. W książce Kastanakis B. (2016) „Mapbox Cookbook” możemy wyczytać jak możemy szybko pobierać mapy naszej planety, a także z łatwością modyfikować ich wygląd lub szczegółowość informacji które mają one zawierać. Dzięki zastosowaniu Mapboxa jesteśmy także w stanie stworzyć tak że aplikacje do monitorowania miejsca pobytu pojazdów firmowych, lub też programy do wizualizacji topologii terenu lub układ budynków miasta.

#### **3.2. Unity game engine.**

Unity engine jest multiplatformowym silnikiem tworzony przez Unity Technologies. Jego zastosowanie jest bardzo szerokie jako że używa się go nie tylko do tworzenia gier, ale także do kreacji artystycznych jak filmy, krótkie lub długie animacje aż po pełnometrażowe filmy. Twórcy silnika pokazują także przydatność ich engine w tworzeniu aplikacji pomagających w architekturze, budownictwie lub innego typu projektach wykorzystywanych w realnym życiu.

Unity na początku swojego żywota chciało ułatwić dostęp do developmentu gier i oryginalnie było wydane tylko na komputery z systemem operacyjnym od firmy Apple. Po dużym sukcesie na platformie amerykańskiej firmy po pierwszym roku dodano support także dla systemów Microsoftu oraz przeglądarek internetowych. Zainteresowanie silnikiem było na tyle wysokie że spowodowało to jego ekstremalnie szybki rozwój, oraz dodanie możliwości używania silnika w kolejnych środowiskach. Dzisiejszego dnia Unity wspiera ponad 25 platform.

Silnik przez pierwsze 10 lat swojego istnienia operował na jednorazowej sprzedaży, dopiero od roku 2016 zmienił model biznesowy na licencje oraz możliwość bezpłatnego korzystania z jego na podstawie jej darmowej wersji. Licencja ta jest skierowana dla pojedynczych developerów oraz mniejszych firm, które nie przekraczają rocznego dochodu 100tys. dolarów rocznie przyniesionych przez Unity. Trzeba zaznaczyć że model ten pozwala na pełen dostęp do silnika bez żadnych ograniczeń aż do osiągnięcia pułapu zarobkowego.

Programuje się w nim za pomocą języka C# (C sharp), które został stworzony przez Microsoft ich środowiska .NET. Język ten jest wysokopoziomowym i obiektowym językiem projektowania.

### **3.3. GIMP(Gnu Image Manipulation Program).**

GIMP jest darmowym programem do obróbki grafiki dwuwymiarowej działającym na licencji open-source. Aplikacja jest wysoce rozbudowana i posiada wiele narzędzi które można wykorzystać przy tworzeniu wszelkich statycznych grafik, jak i także obrazów typu GIF.

### **3.4. Blender.**

Blender jest programem, które dostarcza wszelkich narzędzi potrzebnych do tworzenia modeli 3D ale także używa się go do tworzenia filmów lub projektowania wydruków dla drukarek 3D.

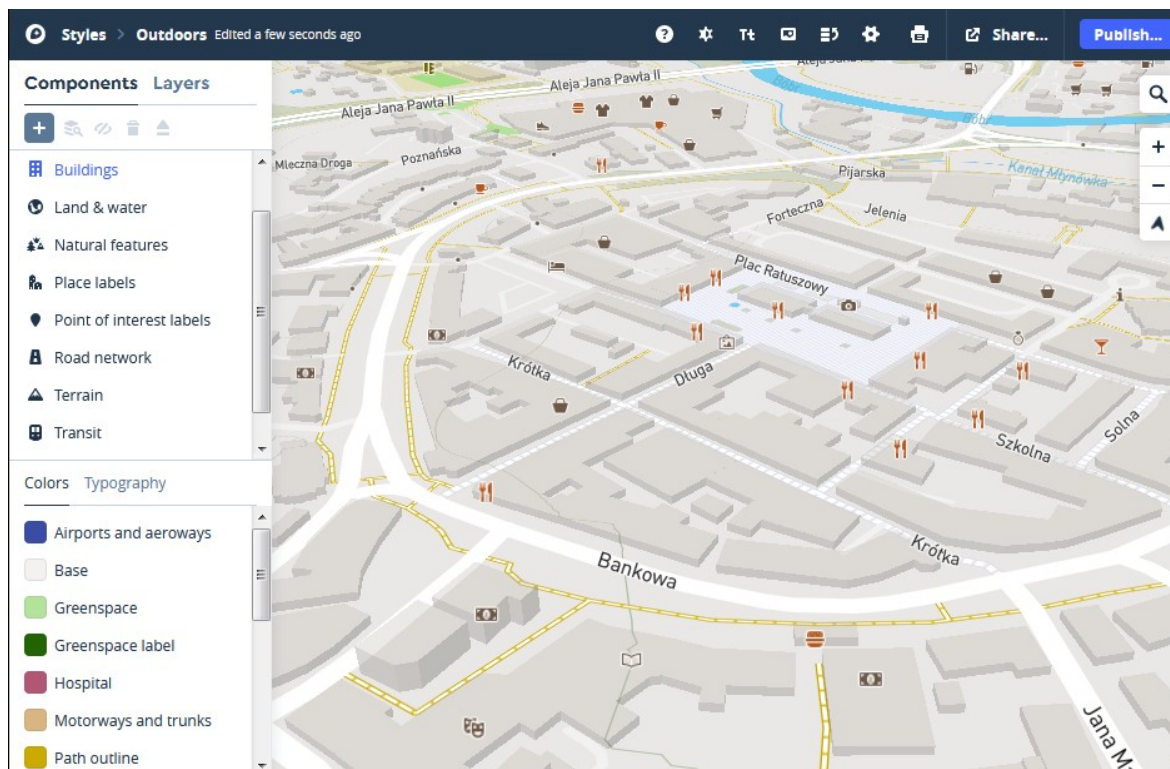


## 4. Opis aplikacji i jego wykonanie.

### 4.1. Wykonanie aplikacji.

Głównym celem gry jest zachęcenia użytkownika do wyjścia z domu na świeże powietrze i zapoznanie się z okolicznymi atrakcjami naturalnymi jak lasy czy parki. Za przebywanie i zwiedzanie terenów zielonych gracz jest w odpowiedni sposób nagradzany. A sama aplikacja zachęca do dłuższych spacerów na świeżym powietrzu.

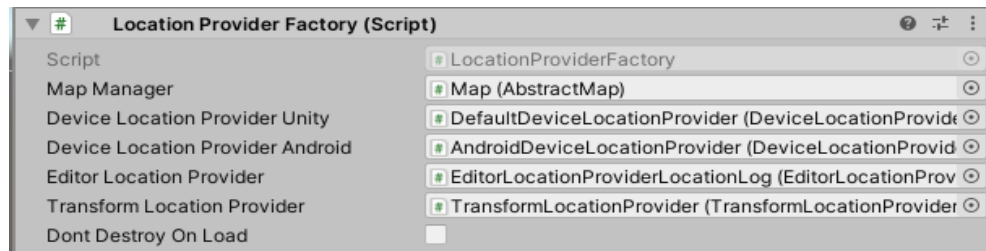
Według początkowych założeń program miał być oparty o Google maps SDK & API, lecz Mapbox SDK & API oferowało o wiele łatwiejszą implementację oraz możliwość dostosowania takich elementów jak wygląd mapy. Mapbox pozwala na proste zmiany w stylu mapy i łatwe dzielenia się szablonem z innymi twórcami jak i już gotowe próbki. Dostosowywanie odbywa się poprzez kreator na oficjalnej stronie (rysunek 4.1). Posiada on wiele elementów które możemy dowolnie dodawać, usuwać lub też modyfikować takie jak zaznaczanie budynków specjalnych typu szkoła, szpitale, posterunki policji. Dzięki temu narzędziu możemy też zmienić charakter mapy aby przypominała takie mapy jak geograficzna, liniowa czy satelitarna.



Rysunek 4.1: Zrzut ekranu przedstawiający kreator stylu mapy z strony Mapbox-a.

(Źródło: Opracowanie własne)

Bardzo dobrze przemyślaną opcją w SDK Mapbox-a jest możliwość symulacji pozycji na mapie geograficznej z poziomu silnika Unity. Dzięki skryptom dołączonym do całego pakietu możemy łatwo je dodać i przetestować nasz program. Taka symulacja GPS pozwala nam na testowania aplikacji nawet jeśli w danym momencie nie posiadamy sprzętu zdolnego ją wykorzystać.

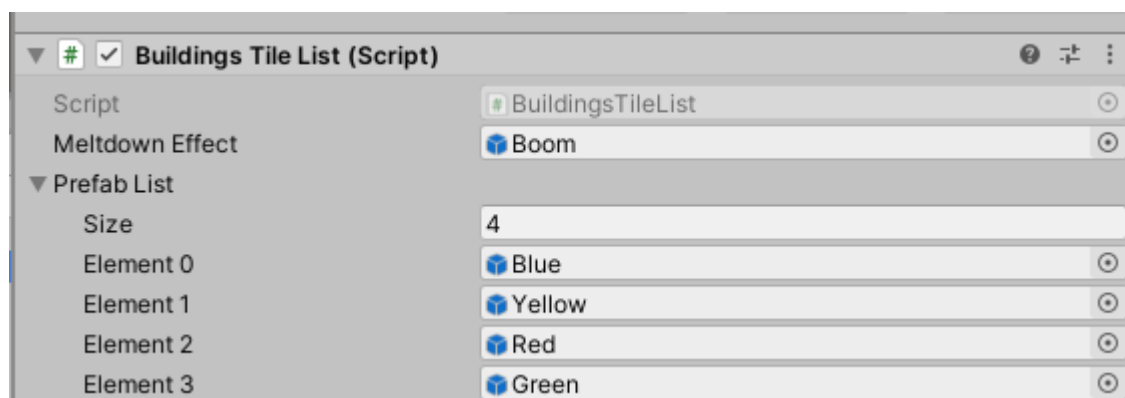


*Rysunek 4.2: Zrzut ekranu Location Providera i skryptów, których używa.  
(Źródło: Opracowanie własne)*

Aplikacja składa się z trzech osobnych scen:

- World która jest odpowiedzialna za mapę świata i pozycje gracza na niej
- Board gdzie znajduje się plansza kryształów zebranych.
- Capture gdzie zostajemy przeniesieni kiedy klikniemy złożę kryształów znajdujące się na mapie świata.

Plansza oraz jej komponenty zostały tak zaimplementowane by w przyszłości dodanie nowych elementów nie stanowiło problemu. Kiedy dodamy skrypt odpowiedzialny za definiowanie następnych elementów do listy budynków tak jak widać to na zdjęciu 4.3, co daje im możliwość korzystania z klas zdefiniowanych specjalnie dla tego typu obiektu jak stawianie budynku, generacja energii itp.

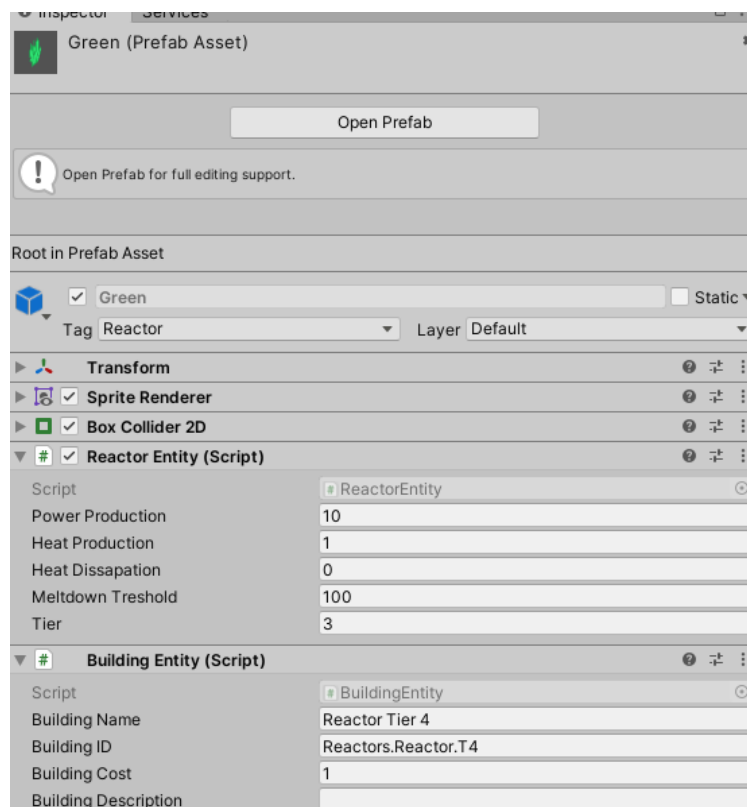


*Rysunek 4.3: Zrzut ekranu z Unity przedstawiający skrypt Buildings Tile List i komponenty podłączone do niego.  
(Źródło: Opracowanie własne)*

Po stworzeniu tak zwanego prefab-a czyli obiektu składającego się ze wszystkich wcześniej dodanych do niego skryptów i innych elementów, którego używa się do szybkiego powielania zamiast tworzenia na nowo tego samego na przykład drzewa, musimy zdefiniować jego wartości. Mamy możliwość edycji takich parametrów jak ilość produkowanej energii, Heat Production i Meltdown Threshold gdzie pierwsze definiuje jak szybko dany kryształ będzie się nagrzewał a drugie pułap nagrzania po którym zniknie. Także jest możliwość zmiany ceny danego budynku.

Taki sposób znacznie przyspiesza pracę jeśli chcemy szybko zmodyfikować właściwości danego obiektu, który później będzie się znajdować na planszy.

Obiektom też jest przypisywane unikatowe ID oraz nazwa w celu rozpoznania i odwołania się do nich w programie.



*Rysunek 4.4: Zrzut ekranu przedstawiający przykładowy prefab wykorzystany w projekcie.*

*(Źródło: Opracowanie własne)*

Aby punkty były wyświetlane i zajmowały sensowną ilość miejsca został zaimplementowany prosty kod konwertujący zapis dziesiętny do notacji naukowej. W ten sposób gdy gracz osiągnie bardzo dużą ilość punktów nie wyjdą one poza obszar pola przeznaczonego do wyświetlania ich, co za tym idzie będą zawsze w pełni widoczne i czytelne. Przykład kodu wykorzystanego właśnie do tego widać na zdjęciu 4.5.

```
if (IdleReactov.Instance.power > 1000)
{
    var exponent = (System.Math.Floor(System.Math.Log10(System.Math.Abs(IdleReactov.Instance.power))));
    var mantissa = (IdleReactov.Instance.power / System.Math.Pow(10, exponent));
    PowerText.text = "Punkty: " + mantissa.ToString("F2") + "e" + exponent;
}
else
    PowerText.text = "Punkty: " + IdleReactov.Instance.power.ToString("F0");
```

*Rysunek 4.5: Zrzut ekranu przedstawiający część zastosowaną do konwersji typu liczbowego kodu.*

*(Źródło: Opracowanie własne)*

Plansza została wykonana w formie tablicy dwuwymiarowej a jej wizualna generacja jest tworzona dzięki części kodu zawartemu na rysunku 4.6

```
void CreateGrid()
{
    float tileSize = tile.GetComponent().sprite.bounds.size.x;
    Vector3 offset = new Vector3(-9, -41, 100);

    /*GameObject board = Instantiate(this.board, offset, Quaternion.identity);
    board.transform.localScale = new Vector3(scale, scale, 0);*/

    for (int y=0; y<10; y++)
    {
        for (int x=0; x< 10; x++)
        {
            GameObject newTile = Instantiate(tile);
            newTile.GetComponent<TileEntity>().Setup(new Vector2(x, y), new Vector3(offset.x - ((tileSize * scale + interval) * (x+1)), offset.y + interval + ((tileSize * scale + interval) * y), 0), scale);
            newTile.transform.parent = GridTiles.transform;
            grid.Add(new Vector3(x, y, 100), newTile);
        }
    }
}
```

*Rysunek 4.6: Kod używany do generacji planszy na której stawia się fragmenty kryształu.*

*(Źródło: Opracowanie własne)*

Dzięki temu że każdy kryształ umieszczony na planszy tak naprawdę zajmuje jedną z komórek tablicy znacznie to upraszcza zapisywanie i odczytywanie jej elementów. Cała tablica jest odczytywana i zapisywana w formie pliku tekstowego. Kod wykorzystany do zapisu możemy zobaczyć na Rysunku 4.7

```

public void Save()
{
    string saveLocation = Application.dataPath + "/save.txt";

    Vector2[] gridPositions = new Vector2[225];
    string[] buildingIDs = new string[225];
    double[] storedHeats = new double[225];
    int buildingsOnBoard = 0;

    for (int x = 0; x < 15; x++)
    {
        for (int y = 0; y < 15; y++)
        {
            Vector2 pos = new Vector2(x, y);
            if (BuildingsTileList.Instance.GetBuildingAtPosition(pos) != null)
            {
                gridPositions[buildingsOnBoard] = BuildingsTileList.Instance.GetBuildingAtPosition(pos).GetComponent<BuildingEntity>().GridPosition();
                buildingIDs[buildingsOnBoard] = BuildingsTileList.Instance.GetBuildingAtPosition(pos).GetComponent<BuildingEntity>().BuildingID();
                storedHeats[buildingsOnBoard] = BuildingsTileList.Instance.GetBuildingAtPosition(pos).GetComponent<BuildingEntity>().HeatStorage();
                buildingsOnBoard++;
                //Debug.Log(objects[objects.Count-1].Item1 + " : " + objects[objects.Count-1].Item2 + " : " + objects[objects.Count-1].Item3);
            }
        }
    }
    SaveObject save = new SaveObject
    {
        gridPosition = gridPositions,
        buildingID = buildingIDs,
        storedHeat = storedHeats,
        buildingsPlaced = buildingsOnBoard
    };
    string json = JsonUtility.ToJson(save);
    File.WriteAllText(saveLocation, json);
}

```

*Rysunek 4.7: Kod przedstawiający zapis elementów tablicy do pliku tekstowego.*

*(Źródło: Opracowanie własne)*

Do samego pliku są zapisywane takie informacje jak:

- Pozycja,
- ID budynku znajdującego się w polu,
- Ilość ciepła zmagazynowanego która jest kluczowa do zapisania jako że odmierza ona czas życia, które pozostało obiektowi na planszy.
- Informacja o sumie budynków postawionych na planszy w celu sprawdzenia poprawności.

Przykładowy stan zapisu takiego pliku możemy zobaczyć na rysunku 4.8.

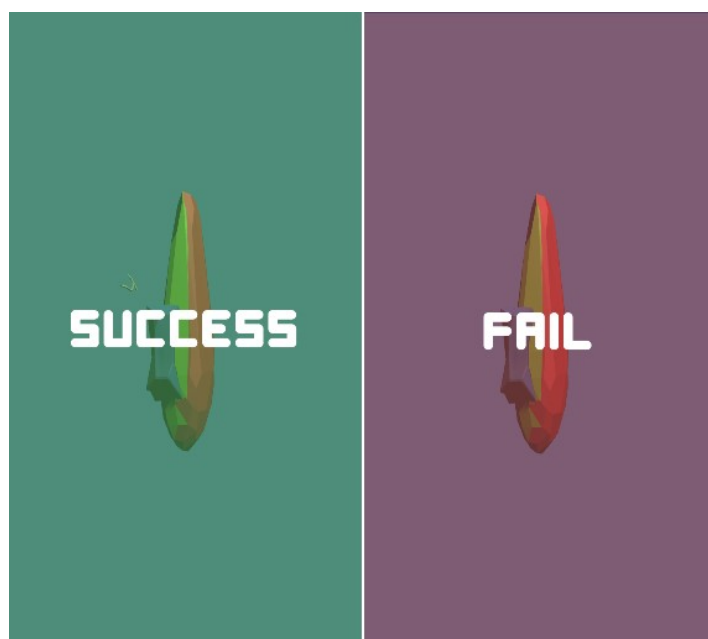




Na tło w kolorze fioletowym podczas uruchomienia aplikacji z kamerą jest naniesiony obraz kamery. Sam kilof jest połączony do głównej kamery co pozwala mu za podążanie za nią. Dzięki podłączeniu kamery do żyroskopu jesteśmy w stanie się obracać i znaleźć złożę kryształu. Efekt można zobaczyć na pokazanym w następnej sekcji rysunku 4.12.

Niestety nie wszystkie urządzenia posiadają żyroskop więc aplikacja może być z nimi nie kompatybilna.

Jeśli gracz trafi lub nie trafi trzy razy z rzędu zostanie mu wyświetlony komunikat informujący o pozytywnym lub negatywnym wykopaniu surowca. Prezentuje to rysunek numer 4.10



*Rysunek 4.10: Zrzut ekranu przedstawiający wykopania oraz porażkę w wykopaniu surowca.*

*(Źródło: Opracowanie własne)*

Gracz zostaje przeniesiony do sceny wydobywania kryształu po kliknięciu lub po kolizji pozycji gracza z obiektem umieszczonym na mapie. Następuje to dzięki fragmentowi kodu zapisanego poniżej.

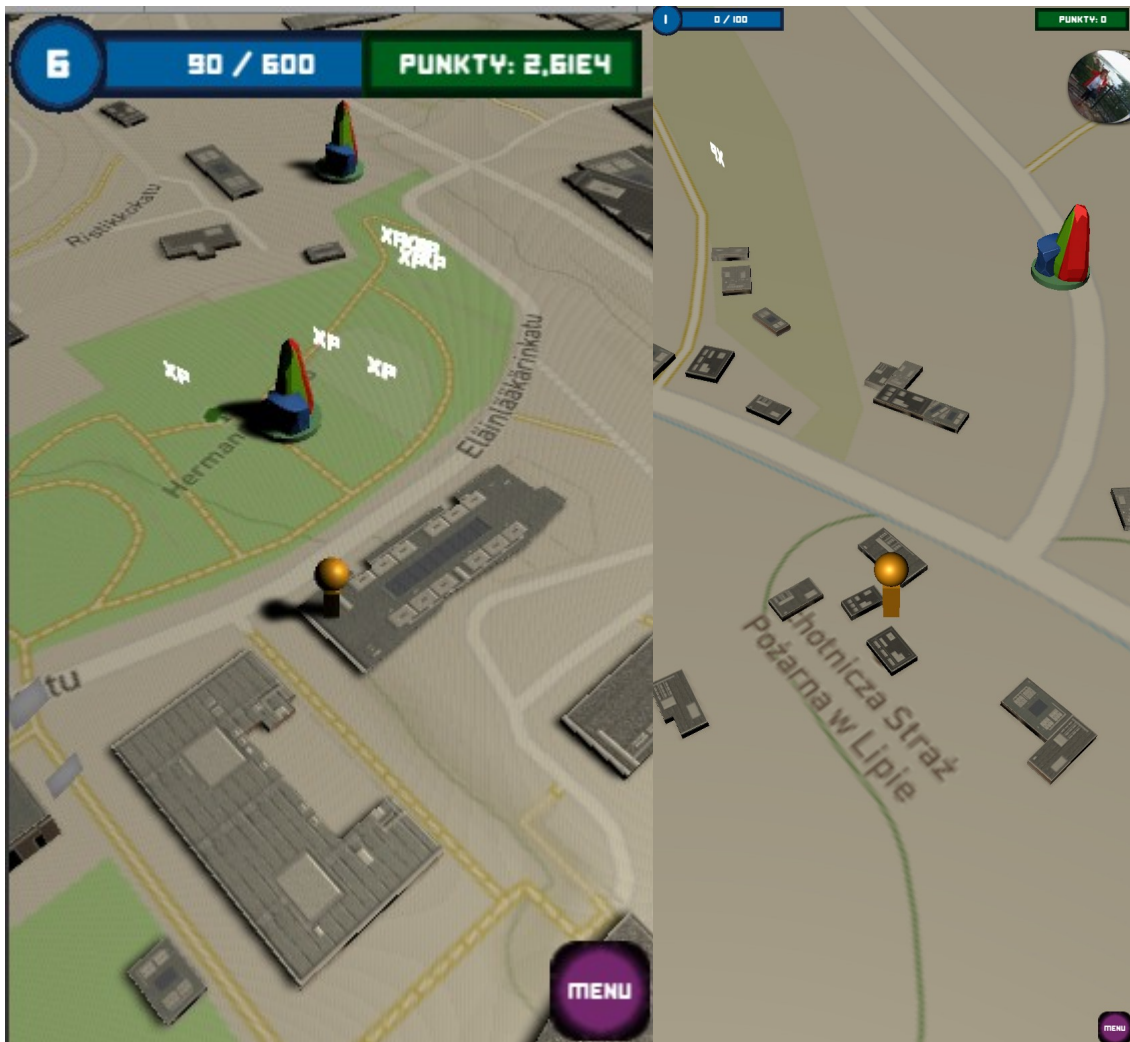
```
public override void clusterCollision(GameObject cluster, Collision other)
{
    List<GameObject> list = new List<GameObject>();
    list.Add(cluster);
    SceneManager.Instance.GoToScene(GemClusterConstants.SCENE_CAPTURE, list);
}
```

*Rysunek 4.11: Fragment kodu przedstawiający przeniesienie gracza do sceny wydobywania po dotknięciu kryształu na mapie świata.*

## 4.2. Instrukcja dotycząca gry.

Po uruchomieniu aplikacji gracz pojawia się na mapie świata jak na rysunku 4.12 w odniesieniu do lokacji geograficznej jego urządzenia. Na planszy znajdują się złoża krysztalów, które pojawiają się losowo. Celem gracza jest odnajdowanie ich i wykopywanie w celu pozyskania fragmentów, które później użyje do generacji punktów.

Na mapie również znajdują się specjalne pola (biały napis XP) przyznające użytkownikowi doświadczenie. Jest ono potrzebne do zwiększania poziomu. Większe poziomy dają benefit w postaci większej ilości zdobywanych skałek oraz definiują liczbę możliwych rzutów kilofem podczas kopania minerału.



*Rysunek 4.12: Zrzuty ekranu z aplikacji przedstawiający gracza na mapie świata.*

*(Źródło: Opracowanie własne)*



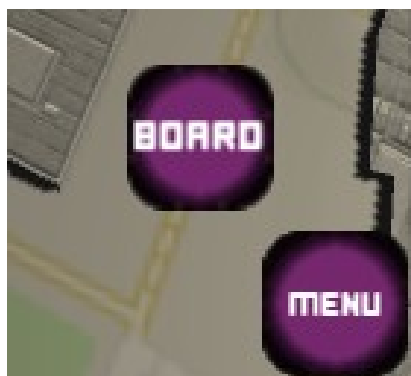
Po wciśnięciu lub podejściu bardzo blisko do minerału znajdującego się na mapie gracz zostaje przeniesiony do specjalnej minigry, gdzie musi trafić w kryształ znajdujący się w jego otoczeniu kilofem rysunek 4.13. W przypadku niepowodzenia będzie zmuszony znaleźć nowy depozyt minerałów. Lecz jeśli mu się uda trafić zostanie wynagrodzony fragmentami. Ilość fragmentów jest losowa i zależna od poziomu.



*Rysunek 4.13: Zrzut ekranu przedstawiający minigrę w której należy trafić kilofem w rudę.*

*(Źródło: Opracowanie własne)*

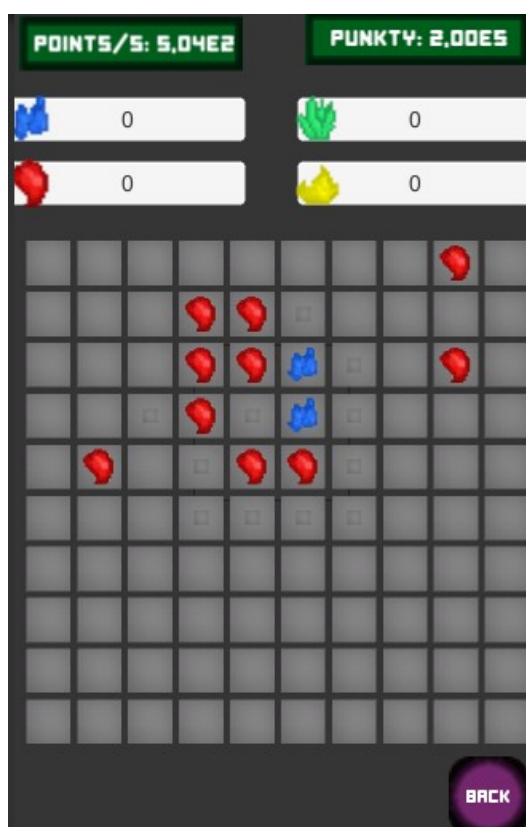
Kiedy użytkownik zdobędzie parę fragmentów może się on przenieść z poziomu mapy świata do swojej planszy za pomocą guzika menu znajdującego się w prawym dolnym rogu. Po naciśnięciu guzika wyświetli się dodatkowy guzik który pozwala na wczytanie planszy. Jest to przedstawione na rysunku 4.14.



*Rysunek 4.14: Zrzut ekranu przedstawiający guzik menu i guzik planszy.*

*(Źródło: Opracowanie własne)*

Kiedy gracz naciśnie guzik zostaje przeniesiony do widoku planszy zaprezentowanej na rysunku 4.15.



*Rysunek 4.15: Zrzut ekranu przedstawiający plansze z postawionymi minerałami.*

*(Źródło: Opracowanie własne)*

Z poziomu planszy gracz może umieścić na niej kryształy które zdobył wykopując złoża z mapy świata. Postawienie fragmentu następuje poprzez wciśnięcie guzika z jego ikoną a potem naciśnięciem na kwadrat planszy.

Żeby nie dało się wypełnić całej planszy kawałkami, co za tym idzie “skończyć” gry mają one swoją wytrzymałość, co zmusza gracza do ciągłego poszukiwania nowych zasobów w celu uzupełniania planszy i wydłużeniu rozgrywki.

Dodatkowo kryształy reagują na siebie jeśli się ze sobą stykają zyskując większy bonus w generowaniu punktów.

Aktualnie jedynym celem jest zwiększenie ilości punktów w nieskończoność i nie wiążą się z nimi żadne inne benefity .

## 5. Podsumowanie i wnioski.

Celem pracy było wykonanie aplikacji z wykorzystaniem technologii rozszerzonej rzeczywistości oraz geolokacji.

Wykonanie projektu zostało dokonane na silniku Unity oraz z pomocą API i SDK Mapbox.

Praca obejmowała następujące etapy:

- 1 Obmyślenie podstaw mechanik aplikacji i jej zastosowania w praktyce.
- 2 Przystąpienie do pisania skryptów oraz zastosowania odpowiedniego zastosowania ich w Unity.
- 3 Przygotowania niezbędnych grafik oraz modeli.
- 4 Przetestowanie aplikacji oraz naniesienie poprawek w oprogramowaniu.

Przy pracy napotkano następujące trudności.

Na etapie tworzenie aplikacji początkowo miało zostać wykorzystane Api Google Maps lecz niekompatybilność niektórych elementów oraz nie do końca jasna dokumentacja sprawiły poszukiwanie nowej alternatywy, którą był Mapbox.

System zapisu postępu, który został oryginalnie napisany wymagał usunięcia i stworzenia od nowa, jako że najmniejsza zmiana w projekcie oznaczała konieczność usuwania już istniejącego zapisu stanu aplikacji.

Urządzenie używane pierwotnie do testowania nie było w stanie używać technologii rozszerzonej rzeczywistości jako że wymaga ona minimalnego poziomu API 24 (Android 7.0 Nugat), a urządzenie użyte do testów posiadało poziom 21 (Android 5.0 Lolipop).

Należy zaznaczyć że aplikacja nie będzie w stanie poprawnie działać na urządzeniach bez żyroskopu.

W podsumowania trzeba zaznaczyć możliwości rozwojowe programu.

Aplikacje można rozwinąć o takie właściwości jak:

- Możliwość ulepszania kryształów i planszy gdzie kod został zaprojektowany z myślą o tym więc sama implementacja nie była by trudna.

- Bardziej skomplikowany system sąsiedztwa kryształów jak szybsze zmniejszenie żywotności fragmentu na rzecz zwiększonej produktywności lub też negatywnego oddziaływania na siebie wzajemnie.
- Nadanie rozgrywce głębszego celu niż zbieranie punktów w nieskończoność.
- Zaimplementowanie dziennych misji wynagradzające gracza różnymi benefitami, które polegałyby na przejściu wyznaczonego dystansu lub też odwiedzenia paru różnych terenów zielonych.
- Dodanie większej różnorodności kryształów.
- Wzbogacenie mapy o dodatkowe punkty zainteresowania. Takie jak wyświetlanie zabytków.
- Progresywne zwiększanie nagrody tym dalej użytkownik zajdzie od domu.

## 6. Bibliografia.

1. Kurpytė D. Navakauskas D. „An Efficiency Analysis of Augmented Reality Marker Recognition Algorithm”  
pozyskano z [https://www.researchgate.net/publication/272261081\\_An\\_Efficiency\\_Analysis\\_of\\_Augmented\\_Reality\\_Marker\\_Recognition\\_Algorith](https://www.researchgate.net/publication/272261081_An_Efficiency_Analysis_of_Augmented_Reality_Marker_Recognition_Algorith)
2. Abramovich G. (2018) 5 „Innovative examples of augmented reality in action.”  
pozyskano z <https://www.adobe.com/insights/5-realworld-examples-of-augmented-reality-innovation.html>
3. Nanos J. (2018) „Every step you take. - How companies use geolocation data to target you and everyone around in ways you're not even aware of”.  
pozykano z <https://apps.bostonglobe.com/business/graphics/2018/07/foot-traffic/>
4. Glover J., Linowes J. (2019) „Complete Virtual Reality and Augmented Reality Development with Unity”.
5. Kastanakis B. (2016) „Mapbox Cookbook”.

## 7. Spis ilustracji.

Rysunek 2.1 Zdjęcie przedstawiające wykorzystanie aplikacji typu AR w celu wyświetlenia białego domu na banknocie amerykańskiego dolara.

Rysunek 2.2 Zdjęcie przedstawiające smart glasses.

Rysunek 2.3 Zdjęcie przedstawiające przykładowe informacje które mogą się znajdować na smart windshield.

Rysunek 2.4 Zdjęcie urządzenia, które wyświetla obraz z kamer umieszczonych po każdej stronie auta, i próbuje zwizualizować widok z trzeciej osoby.

Rysunek 2.5 Wykorzystanie aplikacji typu AR do przedstawienie trójwymiarowego budynku na podstawie jego planu.

Rysunek 2.6 Przykład zastosowania rozszerzonej rzeczywistości do pokazania auta Tesli Model 3.

Rysunek 4.1 Zrzut ekranu przedstawiający kreator stylu mapy z strony Mapbox-a.

Rysunek 4.2 Zrzut ekranu Location Providera i skryptów, których używa.

Rysunek 4.3 Zrzut ekranu z Unity przedstawiający skrypt Buildings Tile List i komponenty podłączone do niego.

Rysunek 4.4 Zrzut ekranu przedstawiający przykładowy prefab wykorzystany w projekcie.

Rysunek 4.5 Zrzut ekranu przedstawiający część zastosowaną do konwersji typu liczbowego kodu.

Rysunek 4.6 Kod używany do generacji planszy na której stawia się fragmenty kryształu.

Rysunek 4.7 Kod przedstawiający zapis elementów tablicy do pliku tekstowego.

Rysunek 4.8 Zrzut ekranu pliku zapisy otworzonego za pomocą Notepad++.

Rysunek 4.9 Zrzut ekranu sceny wydobywania minerałów z silnika i podglądu aplikacji na komputerze.

Rysunek 4.10 Zrzut ekranu przedstawiający wykopania oraz porażkę w wykopaniu surowca.

Rysunek 4.11 Fragment kodu przedstawiający przeniesienie gracza do sceny wydobywania po dotknięciu kryształu na mapie świata.

Rysunek 4.12 Zrzuty ekranu z aplikacji przedstawiający gracza na mapie świata.

Rysunek 4.13 Zrzut ekranu przedstawiający minigrę w której należy trafić kilofem w rudę.

Rysunek 4.14 Zrzut ekranu przedstawiający guzik menu i guzik planszy.

Rysunek 4.15 Zrzut ekranu przedstawiający plansze z postawionymi minerałami.

## **8. Załączniki.**

- 1    Praca inżynierska w pliku PDF
- 2    Program dla urządzeń operujących na systemie android