 POLITECHNIKA WARSZAWSKA

WYDZIAŁ MATEMATYKI

I NAUK INFORMACYJNYCH

PRACA DYPLOMOWA

INŻYNIERSKA

INFORMATYKA

**Gogle do rzeczywistości wirtualnej i rozszerzonej oparte o ekran i kamerę smartfona**

Goggles for virtual and augmented reality based on smartphone’s screen and camera

Autor:

Anna Zawadzka

Współautorzy:

Sylwia Nowak

Jakub Cieślik

Promotor: dr inż. Paweł Kotowski

Warszawa, luty 2016

............................................. .............................................

podpis promotora podpis autora

**STRESZCZENIE**

Celem niniejszej pracy dyplomowej było stworzenie aplikacji przeznaczonej na urządzenia mobilne realizującej koncepcję gogli do rozszerzonej i wirtualnej rzeczywistości.

Praca składa się z X części….

SŁOWA KLUCZOWE

Rzeczywistość rozszerzona | Android | Google Cardboard

**ABSTRACT**

The purpose of present thesis vas to create application for mobile devices…

KEY WORDS

Ambient reality | Android | Google Cardboard

Spis treści

[1 Wstęp 6](#_Toc441172524)

[1.1 Rzeczywistość wirtualna i rozszerzona 6](#_Toc441172525)

[1.2 Google Cardboard 6](#_Toc441172526)

[2 Aplikacja FindMyMeal 8](#_Toc441172527)

[2.1 Opis aplikacji 8](#_Toc441172528)

[2.2 Diagram przypadków użycia 9](#_Toc441172529)

[3 Wymagane środowisko sprzętowe i systemowe 10](#_Toc441172530)

[3.1 Wymagania systemowe 10](#_Toc441172531)

[3.2 Wymagania sprzętowe 11](#_Toc441172532)

[3.3 Wymagane moduły 11](#_Toc441172533)

[4 Dokumentacja techniczna 12](#_Toc441172534)

[4.1 Model dziedziny 12](#_Toc441172535)

[4.2 Opis klas 12](#_Toc441172536)

[4.3 Biblioteki 12](#_Toc441172537)

[4.4 Algorytmy 12](#_Toc441172538)

[5 Proces wytwarzania oprogramowania 12](#_Toc441172539)

[5.1 Scenariusze testów 12](#_Toc441172540)

[5.2 Wersjonowanie 12](#_Toc441172541)

[6 Testy użytkowe S 12](#_Toc441172542)

[7 Opis działania aplikacji A 12](#_Toc441172543)

[7.1 Instrukcja użytkownika 12](#_Toc441172544)

[7.2 Porównanie działania rzeczywistego z zakładanym 12](#_Toc441172545)

[8 Podsumowanie 12](#_Toc441172546)

[8.1 Problemy nierozwiązane 12](#_Toc441172547)

[8.2 Dalsze plany rozwojowe 13](#_Toc441172548)

[8.3 Wnioski 13](#_Toc441172549)

[9 Bibliografia 13](#_Toc441172550)

# Wstęp

## Rzeczywistość wirtualna i rozszerzona

Rzeczywistość rozszerzona (*poszerzona rzeczywistość*, ang. *augmented reality*) to metodologia pracy z systemami informatycznymi, polegająca na łączeniu świata rzeczywistego z obiektami pochodzącymi z świata wirtualnego. Nakładanie informacji odbywa się za pośrednictwem *wirtualnej powłoki* i następuje w czasie rzeczywistym. Rzeczywistość rozszerzona zdobywa informacje o otoczeniu dzięki przetwarzaniu danych z kamery, czujników lokalizacji (takich jak GPS lub poprzez wykonanie tzw. trangulacji) oraz dzięki odczytom z sensorów położenia mówiących np. o jego fizycznym położeniu względem powierzchni ziemi. Dzięki temu generowane w ramach *wirtualnej powłoki* rozszerzenia mogą przybierać wszelaką postać. Poczynając od prostych informacji nałożonych na świat rzeczywisty (np. nazwy ulic lub nawigacja) aż do skomplikowanych obiektów fotorealistycznych, które wtapiają się w świat realny i tworzą z nim jedną całość (np. rekonstrukcje zniszczonych historycznych budynków czy symulacje militarne).

Rzeczywistość wirtualną (ang. *virtual reality*) od rzeczywistości rozszerzonej odróżnia fakt niewchodzenia w interakcje z obiektami rzeczywistymi, poprzestając jedynie na wyświetlaniu generowanych komputerowo obiektów.

W 1994 roku Paul Milgram i Fumio Kishino sformułowali definicję rzeczywistości mieszanej (ang. *mixed reality*) za pomocą koncepcji schematu ciągłości rzeczywistość – wirtualność (ang. *virtuality continuum*) przedstawiającego relacje między rzeczywistym a wirtualnym światem.



Rysunek 1. Schemat ciągłości rzeczywistość – wirtualność

Za jeden z końców powyższego schematu przyjęto środowisko rzeczywiste (ang. *real environment*), obok którego umiejscowiono rzeczywistość rozszerzoną (ang. *augmented reality*). Im bliżej systemowi do wirtualnej rzeczywistości, tym bardziej zredukowana jest liczba elementów rzeczywistych. Rzeczywistość może być „rozszerzana” o wirtualne obiekty, analogicznie do tego wirtualny świat może być „rozszerzany” przez rzeczywiste obiekty. Takie środowisko nazywane jest rozszerzoną wirtualnością (ang. *augmented virtuality*). Na schemacie umiejscowiona jest tuż przy środowisku wirtualnej rzeczywistości (ang. real environment). Opanowanie idei całego schematu pomaga w klasyfikacji wszystkich systemów, w których mieszany jest świat rzeczywisty i wirtualny.

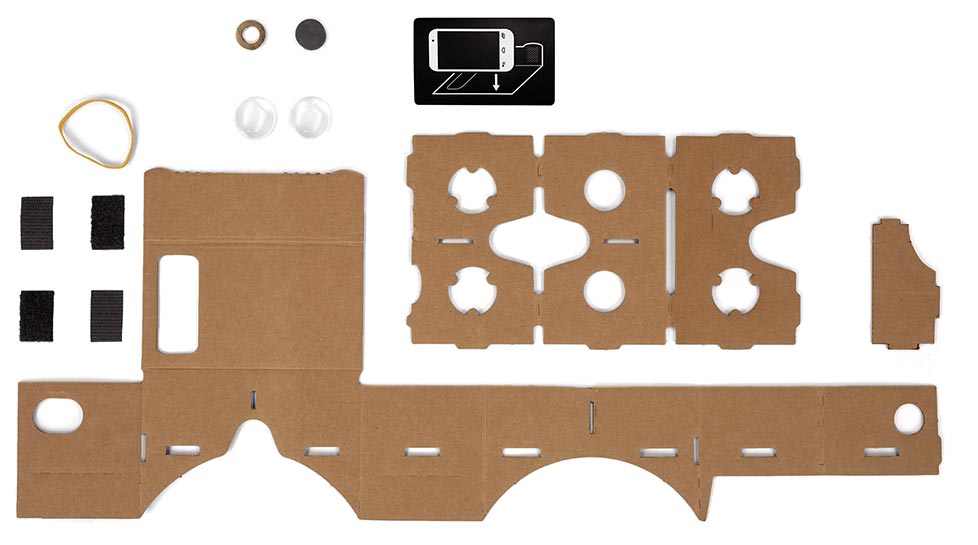
Aplikacje wykorzystujące rzeczywistość wirtualną i rozszerzoną najczęściej spotykane są na takich urządzeniach jak telefony, tablety i PDA. Proces miniaturyzacji sprzętu oraz chęć uczynienia go coraz bardziej funkcjonalnym powodują, iż rynek domaga się rozwiązań zintegrowanych, łączących maksimum funkcjonalności w jednym urządzeniu. Jedyną wadą wymienionych wcześniej urządzeń jest stosunkowo niewielki obszar roboczy wyświetlacza, ograniczający w pewien sposób możliwości pełnej interakcji użytkownika z aplikacją. Dlatego też pionierzy w dziedzinie rozwiązań mobilnych opartych o rozszerzoną rzeczywistość dążą do opracowania okularów przeziernikowych, pozwalających oglądać świat własnymi oczami. Firma Google jako jedna z pierwszych postanowiła poważnie zainwestować w projekt oparty o takie rozwiązanie. Jednak ich pomysł na Google Glass z szerzej nie znanych powodów nie został oddany do szerokiej sprzedaży. Amerykański producent postanowił wystartować z dużo bardziej dostępnym produktem Google Cardboard, który odróżnia się tym, iż obraz trafia do użytkownika za pośrednictwem ekranu i kamery telefonu przytwierdzonego do specjalnego, zewnętrznego modułu.

http://www.e-edukacja.net/piata/referaty/sesja\_IIIb/26\_e-edukacja.pdf

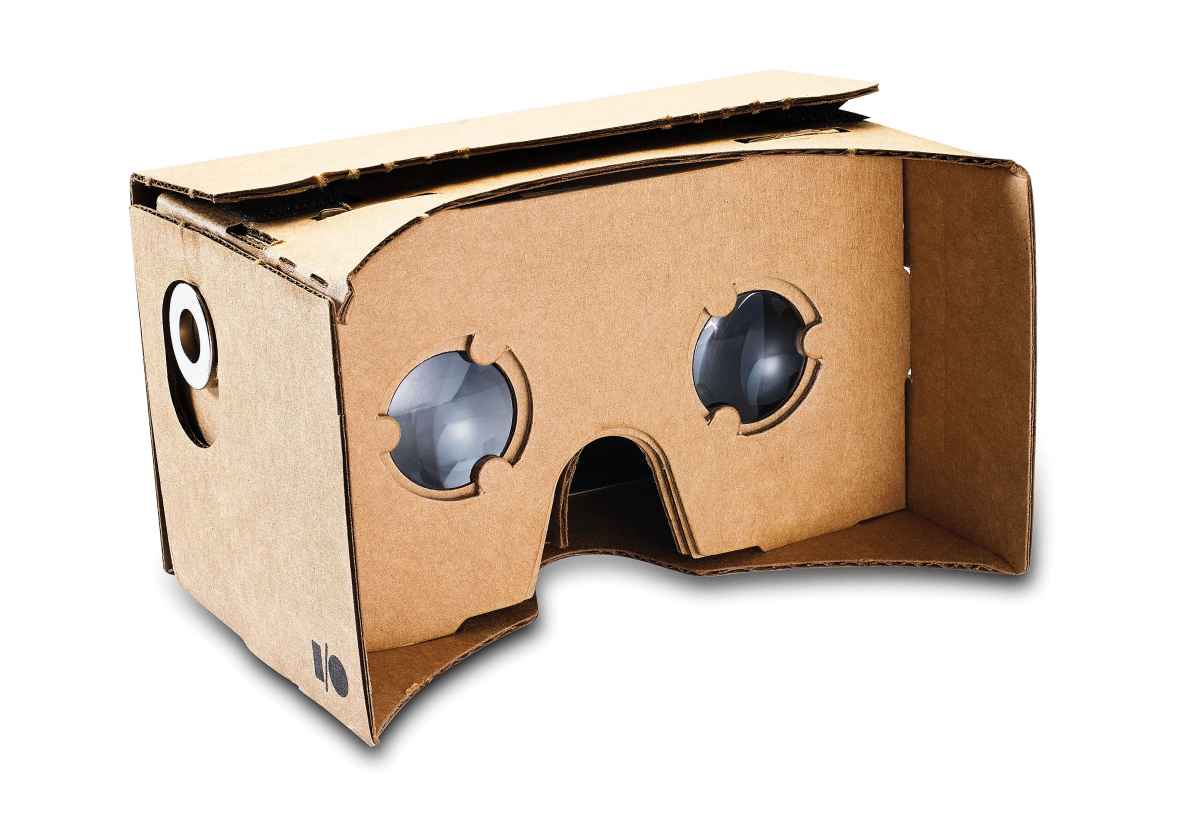
## Google Cardboard

*Google Cardboard* to platforma wirtualnej rzeczywistości stworzona przez firmę *Google* do użycia z kartonowymi goglami oraz smartfonem.

Jednym z elementów *Google Carboard* są gogle, składające się z odpowiednio wyciętych kartonowych części, dwóch soczewek o średnicy 25 mm i ogniskowej około 40 mm, dwóch magnesów (jeden neodymowy oraz jeden ferrytowy lub ceramiczny), rzepów oraz gumki podtrzymującej smartfon.



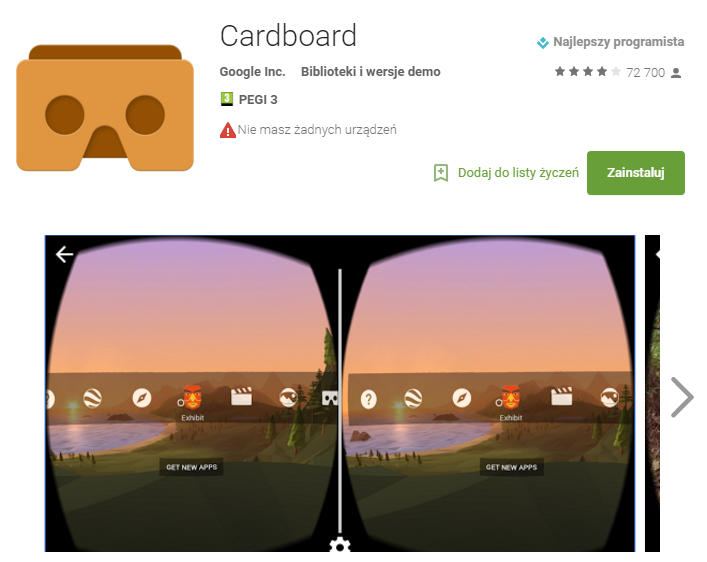
Rysunek 2. Elementy gogli Google Cardboard przed złożeniem



Rysunek 3. Gogle Google Cardboard po złożeniu

Aplikacje kompatybilne z Google Cardboard dostępne są do zainstalowania w Sklepie Google Play lub Apple App Store. Najważniejszym ich elementem jest podzielenie ekranu na dwie części i wyświetlenie jednocześnie dwóch identycznych obrazów - po jednym dla każdego oka. Dzięki niewielkiemu przesunięciu wyświetlanych części uzyskuje się efekt widzenia trójwymiarowego. Kolejnym niezbędnym aspektem jest śledzenie ruchów głowy (ang. *head tracking*). Dzięki akcelerometrom, które śledzą zarówno położenie jak i kąt nachylenia urządzenia, możliwe jest swobodne rozglądanie się po wirtualnym otoczeniu. W efekcie ruchy na ekranie odzwierciedlają ruchy głowy użytkownika w rzeczywistości.

Jedną z popularniejszych aplikacji dostępnych w Google Play jest *Cardboard*. Pozwala ona poznać możliwości jakie daje rozszerzona i wirtualna rzeczywistość, odkrywać nowe aplikacje oraz ustawiać gogle.



Rysunek . Aplikacja Cardboard w sklepie Google Play

Urządzenia, kompatybilne z Google Cardboard:

* Google/LG Nexus 4, 5, 6
* HTC Eco 3D, One (Mini, S, X, X+), Sensation, Sensation XE, Velocity 4G
* Huawei Ascend G 615 + P1
* iOcean X7
* LG G2, G3, Optimus 3D Max (P720), Optimus 4X HD (P880), Optimus G (E975), Optimus G Pro + P940 Prada 3
* Samsung Aktiv S, Galaxy (Beam, S2, S3, S3 Mini), S4 (Active, Mini)
* Sony Xperia S, SP, T + Z1
* Wiko Highway

Źródło: <http://shop.zaak.io/pages/compatible-smartphones>, dostęp 9.01.2016

# Aplikacja FindMyMeal

## Opis aplikacji

Realizacją tematu pracy dyplomowej jest aplikacja mobilna wykorzystująca rzeczywistość rozszerzoną, która może być elementem pomocnym w codziennym życiu użytkownika.

Głównym założeniem aplikacji jest wskazywanie użytkownikowi najbliższych restauracji, stąd też nazwa projektu – FindMyMeal. Dodatkowo umożliwia ona użytkownikowi zapamiętywanie ulubionych miejsc oraz pozwala na nawigację do wybranego przez niego lokalu.

Aby umożliwić stały dostęp do usług świadczonych przez aplikację na bieżąco przetwarzane są dane o obecnym położeniu użytkownika pobierane z modułu GPS. Dane o lokalizacji restauracji pobierane są przez API Google Maps.

Interfejs użytkownika zbudowany jest przy użyciu biblioteki Google Cardboard. Jego działanie opiera się na informacjach pochodzących z dwóch akcelerometrów: żyroskopu oraz akcelerometru orientacji. Dodatkowo, w ramach API Google Cardboard odczytywane są akcje związane z wciśnięciem przez użytkownika klawisza akceptującego dane działanie bądź akcję. Dane z interfejsu użytkownika używane są dalej do generowania odpowiednich zachowań aplikacji, przez intuicyjną obsługę menu głównego do wyświetlania przybliżonych informacji o kierunku włącznie.

Jak już wspomniano, dodatkowa funkcjonalność aplikacji pozwala użytkownikowi na zapisywanie ulubionych lokalizacji (restauracji, lokali, pubów). Zebrane w ten sposób dane są dostępne z menu głównego aplikacji po wybraniu opcji Ulubione miejsca (ang. *Favourites*). Dane zaprezentowane są w postaci wybieralnej listy. Wybranie elementu powoduje przejście do okna z nawigacją do wskazanej przez użytkownika lokalizacji.

W celu zapewnienia jak najwyższej dostępności aplikacji dostępna jest opcja pomocy w postaci filmu instruktażowego.

## Diagram przypadków użycia

Diagram przypadków użycia przedstawia pokrótce funkcjonalność systemu wraz z jego otoczeniem.



Rysunek 5. Diagram przypadków użycia

## Instrukcja użytkownika

Przed uruchomieniem aplikacji należy upewnić się, że moduł *Google Cardboard* został prawidłowo założony na telefonie.

### Uruchomienie aplikacji

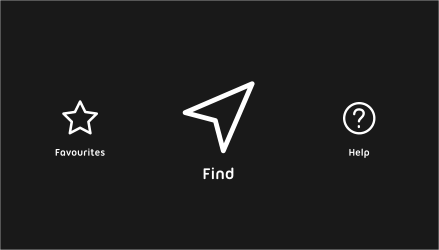
Po uruchomieniu aplikacji na ekranie wyświetlane jest jej logo. W tym momencie w tle aktywowane są wszystkie usługi umożliwiające poprawne działanie poszczególnych modułów aplikacji.



Rysunek 2. Okno ładowania aplikacji

### Menu główne

Po załadowaniu wszystkich komponentów pojawia się ekran zawierający menu główne aplikacji, które zawiera trzy opcje: *Favourites* (*Ulubione*), *Find* (*Znajdź*) oraz *Help* (*Pomoc*).



Rysunek 3. Menu główne

Przechodzenie między różnymi elementami menu głównego następuje poprzez poruszanie modułem *Cardboard* przez użytkownika w prawą i lewą stronę.



Rysunek 4. Kierunki poruszania modułem Cardboard

Wybieranie zaznaczonej opcji wykonywane jest przy pomocy przycisku magnetycznego modułu Google Cardboard.



Rysunek 5. Wciśnięcie przycisku magnetycznego

### Opcja *Find*

Po wybraniu opcji *Find* następuje przejście do okna nawigacji.



Rysunek 66. Okno nawigacji

Na ekranie widoczny jest obraz z kamery wzbogacony o informacje o aktualnym położeniu (bieżący adres) oraz o najbliżej znajdujących się restauracjach.

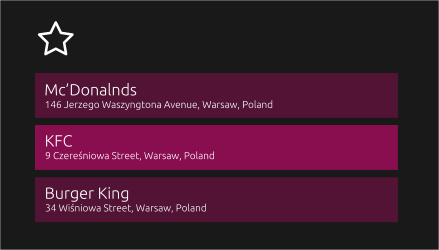
Na pasku stanu (dolna belka) wyświetlany jest pełny adres oraz kompas informujący, w jakim kierunku zwrócony jest użytkownik.

Komunikaty o najbliższych lokalach (co najwyżej trzy komunikaty, zawierające nazwę restauracji, odległość do pokonania oraz strzałkę, wskazującą kierunek poruszania) wyświetlane są w lewym górnym rogu okna. Informacje te będą zmieniać się wraz ze zmianą położenia użytkownika.

Z poziomu tej opcji możliwe jest dodatkowo dodanie nowego miejsca do listy ulubionych. Następuje to poprzez użycie przycisku magnetycznego – do listy dostępnej pod opcją *Favourites* dodawany jest rekord zawierający nazwę lokalu i adres pierwszego elementu z listy najbliższych restauracji.

### Opcja Favourites

Kolejną opcją, którą można wybrać z menu głównego jest *Favourites.* W tym miejscu na ekranie pojawia się lista zapisanych przez użytkownika ulubionych miejsc (nazwa lokalu oraz adres) posortowana malejąco po dacie dodania rekordu. W przypadku braku zapisanych miejsc wyświetlany jest odpowiedni komunikat.



Rysunek 7. Lista ulubionych miejsc

Przechodzenie po liście odbywa się poprzez poruszanie przez użytkownika modułem *Cardboard* w dół. W momencie osiągnięcia końca listy kolejnym zaznaczonym elementem jest pierwsza restauracja na liście (dodana jako ostatnia).



Rysunek 8. Schemat poruszania modułem Cardboard w dół

Wybranie elementu listy odbywa się poprzez użycie przycisku magnetycznego. Po wybraniu elementu następuje przejście do okna widoku z kamery uzupełnionego informacją o kierunku poruszania i pozostałym dystansie do wybranego miejsca.



Rysunek 97. Okno nawigacji wybranego ulubionego miejsca

### Opcja *Help*

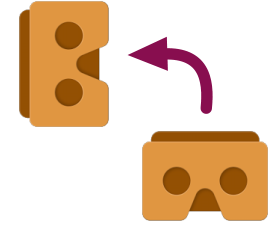
Po wybraniu tej opcji następuje uruchomienie filmu instruktażowego.



Rysunek 10. Okno pomocy

### Powrót

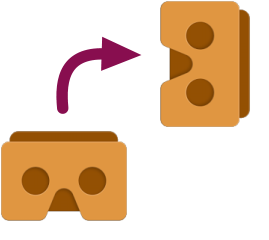
Aby powrócić do poprzedniego okna aplikacji należy przechylić moduł Google Carboard w lewo pod kątem 90 stopni, następnie powrócić do normalnego położenia.



Rysunek 11. Ruch modułem Google Cardboard – powrót

### Wyjście z aplikacji

Zgodnie z oficjalnymi wytycznymi Google przechylenie modułu Carboard w prawo pod kątem 90 stopni skutkuje wyjściem z aplikacji FindMyMeal.



Rysunek 12. Ruch modułem Google Cardboard - wyjście

# Wymagane środowisko sprzętowe i systemowe

## Wymagania systemowe

Projekt realizowany jest w technologii *Android*.

Minimalne wymagania systemu to *API 19 – Android KitKat*.

## Wymagania sprzętowe

Projekt przeznaczony jest na urządzenia mobilne o minimalnej przekątnej ekranu 4.4’’. Przekątna ekranu telefonu nie powinna przekraczać 7’’ .

Urządzenie mobilne musi posiadać kamerę oraz czujnik ruchu (akcelerometr).

Urządzenie musi mieć stały dostęp do Internetu oraz wbudowany moduł GPS.

## Wymagane moduły

Aby w pełni korzystać z funkcji udostępnianych przez aplikację FindMyMeal, użytkownik powinien posiadać dodatkowo zakupiony moduł *Google Cardboard*, który umożliwia obsługę interfejsu aplikacji. Moduł ten powinien być dopasowany do wielkości ekranu urządzenia.

# Dokumentacja techniczna

## Model dziedziny

## Opis klas

## Biblioteki

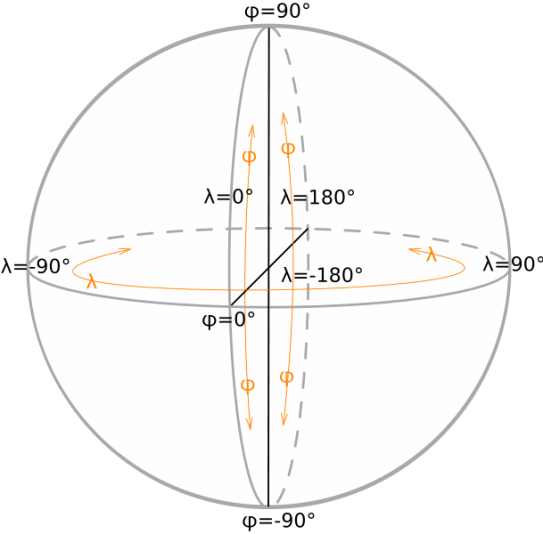
## Algorytmy

### Przekształcenia współrzędnych geograficznych

W aplikacji zastosowano dwa algorytmy służące do przekształcania współrzędnych geograficznych - szerokości geograficznej (ang. *latitude*) oznaczanej przez φi długości geograficznej (ang. *longitude*) oznaczanej przez λ. Obie te wartości mierzone są w stopniach, minutach i sekundach kątowych. Początkiem układu współrzędnych geograficznych jest przecięcie się południka zerowego z równikiem.

Kąt λ przyjmuje wartości od -180º do 180º i jest on zawarty między półpłaszczyzną południka przechodzącego przez dany punkt i półpłaszczyzną południka zerowego. Wartości ujemne liczone są w kierunku zachodnim natomiast dodatnie w kierunku wschodnim.

Kąt φ przyjmuje wartości od -90º do 90º i zawarty jest między kierunkiem normalnej do powierzchni Ziemi (od jej jądra) a płaszczyzną równika ziemskiego. Wartości dodatnie otrzymywane są w kierunku północnym, ujemne natomiast w kierunku południowym.



Rysunek 8. Sfera przedstawiająca wartości współrzędnych geograficznych  
https://pl.wikipedia.org/wiki/Plik:Geographic\_coordinates\_sphere.svg

https://pl.wikipedia.org/wiki/Wsp%C3%B3%C5%82rz%C4%99dne\_geograficzne  
http://encyklopedia.pwn.pl/haslo/wspolrzedne-geograficzne;3998457.html

#### Obliczanie kierunku nawigowania

##### Cel algorytmu

Celem algorytmu jest obliczenie kierunku, w którym ma podążać użytkownik aby osiągnąć miejsce docelowe.

##### Opis ogólny algorytmu

W celu ułatwienia zrozumienia działania algorytmu został on przedstawiony w sposób opisowy. Podczas działania naszej aplikacji algorytm ten wykonywany jest wielokrotnie dla wielu restauracji w pętli. Zastosowane w programie z tego tytułu optymalizacje zostały pominięte.

Argumenty wejściowe:

* *bieżąca lokalizacja użytkownika* - wielkości kątów współrzędnych geograficznych,
* *lokalizacja docelowej restauracji* - wielkości kątów współrzędnych geograficznych,
* *miara kąta odchylenia urządzenia od kierunku północnego* - w radianach, pochodząca z sensora magnetycznego.

Zwracana wartość:

* *miara kąta między punktem docelowym a kierunkiem północnym* - w radianach.

Kolejne kroki algorytmu:

1. Dla wejściowych obiektów lokalizacji wykonywane jest przekształcenie z współrzędnych geograficznych do współrzędnych płaszczyzny i zapisanie otrzymanych wartości *x* i *y* w postaci punktu. Wykorzystywane do tego celu jest rozwiązanie problemu *odwzorowania walcowego równokątnego*. Wielkości *x* i *y* obliczane są według wzorów:

Wielkość *R* jest stałą skalowania mapy. Do naszych obliczeń przyjmujemy *R = 256* z *8-ktornym przybliżeniem* (tj. nasz obszar odwzorowywanej mapy zostaje podzielony na *28 kawałków*). Obliczenia te w swej postaci uwzględniają krzywiznę Ziemi.

https://en.wikipedia.org/wiki/Mercator\_projection#Derivation\_of\_the\_Mercator\_projection  
https://pl.wikipedia.org/wiki/Odwzorowanie\_walcowe\_r%C3%B3wnok%C4%85tne

1. W drugim kroku dokonywane jest obliczenie kąta miedzy dwoma obliczonymi wcześniej wartościami punktów (tj. dla bieżącej lokalizacji i lokalizacji docelowej). Punkty traktowane są jako dwa wektory. W celu ułatwienia i znormalizowania obliczeń używana jest funkcja *atan2*, której jako parametr zostaje podany wektor różnicy dwóch wcześniej wymienionych wektorów. Otrzymany wynik w radianach mówi nam jak wielki jest kąt między wektorem różnicy a *osią* *X*.
2. Aby obliczony kąt był zrozumiały dla użytkownika musi on odnosić się do kierunku północnego - *osi* *Y*. Dlatego też do obliczonego kąta należy dodać miarę *π/2* co odpowiada obrotowi o *90º* przeciwnie do ruchu wskazówek zegara.
3. Ostatnim krokiem algorytmu jest dodanie obliczonej wartości w radianach do informacji o kierunku położenia telefonu otrzymanego z sensora magnetycznego.

Do implementacji wykorzystywane są następujące klasy pomocnicze:

* klasa Point,
* klasa Location.

#### Obliczanie odległości między dwiema współrzędnymi geograficznymi

##### Cel algorytmu

Celem algorytmu jest obliczenie odległości między dwiema współrzędnymi geograficznymi uwzględniając krzywiznę powierzchni Ziemi.

##### Opis algorytmu

Dane wejściowe:

* *bieżąca lokalizacja* - wielkości kątów współrzędnych geograficznych,
* *lokalizacja docelowa* - wielkości kątów współrzędnych geograficznych,
* *długość promienia Ziemi* - zależny od bieżącego położenia, wyrażony w jednostce metrycznej *układu SI* (np. metry lub kilometry).

Zwracana wartość:

* *odległość między dwiema lokalizacjami* - wyrażona w określonej wcześniej jednostce metrycznej *układu SI* dla promienia Ziemi*.*

Pseudokod algorytmu:

obliczOdleglosc(biezacaLokalizacja, docelowaLokalizacja, promienZiemi) {

roznicaLatitude = docelowaLokalizacja.Latitude - biezacaLokalizacja.Latitude

roznicaLongitude = biezacaLokalizacja.Longitude - biezacaLokalizacja.Longitude

radianyLatitude = stopnieDoRadianow(roznicaLatitude);

radianyLongitude = stopniedoRadianow(roznicaLongitude);

a = cos(stopnieDoRadianow(docelowaLokalizacja.Latitude))

a \*= cos(stopnieDoRadianow(biezacaLokalizacja.Latitude))

a \*= sin(radianyLongitude /2)^2

a += sin(radianyLatitude/2)^2

c = 2 \* atan2(,);

zwróć promienZiemi \* c;

}

stopnieDoRadianow(stopnie) {

zwróć stopnie \* (π/180)

}

# Proces wytwarzania oprogramowania

## Scenariusze testów

## Wersjonowanie

# Testy użytkowe S

# Podsumowanie

## Problemy nierozwiązane

## Dalsze plany rozwojowe

## Wnioski

# Bibliografia

[1] Ronald Azuma, Yohan Baillot, Reinhold Behringer, Steven Feiner, Simon Julier, Blair MacIntyre. Recent Advances in Augmented Reality. IEEE Computer Graphics and Applications 21, 6 (Nov/Dec 2001), 34-47

Warszawa, dnia ...............

Oświadczenie

Oświadczam, że pracę inżynierską pod tytułem: „Gogle do rzeczywistości wirtualnej i rozszerzonej oparte o ekran i kamerę smartfona”, której promotorem jest dr inż. Paweł Kotowski, wykonałem/am samodzielnie, co poświadczam własnoręcznym podpisem.

...........................................