

POLITECHNIKA WROCŁAWSKA  
WYDZIAŁ ELEKTRONIKI

---

KIERUNEK: INFORMATYKA  
SPECJALNOŚĆ: INŻYNIERIA SYSTEMÓW INFORMATYCZNYCH

PRACA DYPLOMOWA  
INŻYNIERSKA

System wspomagający organizację gry miejskiej  
wykorzystującej urządzenia z systemem Android

Management system for location-based game for  
Android devices

AUTOR:  
Adam Zimny

PROWADZĄCY PRACĘ:  
dr inż. Paweł Rogaliński, Jednostka

OCENA PRACY:

# Spis treści

<b>1. Wstęp . . . . .</b>	<b>6</b>
1.1. Cel projektu . . . . .	6
1.2. Zakres projektu . . . . .	6
1.3. Układ pracy . . . . .	6
<b>2. Zasady gry . . . . .</b>	<b>7</b>
2.1. Zlecenia . . . . .	7
2.2. Lokalizacja gracza . . . . .	8
2.3. Punktacja . . . . .	8
2.4. Rejestracja do gry . . . . .	8
2.5. Przegląd istniejących rozwiązań . . . . .	9
2.6. Koncepcja rozwiązania . . . . .	9
2.7. Słownik pojęć . . . . .	9
<b>3. Analiza wymagań systemu . . . . .</b>	<b>10</b>
3.1. Architektura systemu . . . . .	10
3.2. Wymagania funkcjonalne . . . . .	10
3.2.1. Aplikacja kliencka . . . . .	10
3.2.2. Aplikacja serwerowa . . . . .	12
3.2.3. Panel administracyjny . . . . .	14
3.3. Wymagania niefunkcjonalne . . . . .	15
3.4. Projekt interfejsu . . . . .	15
3.4.1. Aplikacja kliencka . . . . .	15
3.4.2. Panel administracyjny . . . . .	15
3.5. Wykorzystywane technologie i narzędzia . . . . .	15
<b>4. Implementacja systemu . . . . .</b>	<b>16</b>
<b>Literatura . . . . .</b>	<b>17</b>
<b>A. Tytuł dodatku . . . . .</b>	<b>18</b>
<b>B. Opis załączonej płyty CD/DVD . . . . .</b>	<b>19</b>

# Spis rysunków

# Spis tabel

# Skróty

**OGC** (ang. *Open Geospatial Consortium*)

# Rozdział 1

## Wstęp

Niniejszy dokument opisuje przebieg realizacji projektu dyplomowego w temacie *System wspomagający organizację gry miejskiej wykorzystującej urządzenia z systemem Android* na kierunku Informatyka, na wydziale Elektroniki Politechniki Wrocławskiej.

Grą miejską nazywana jest gra, która wykorzystuje przestrzeń świata rzeczywistego jako element swojej mechaniki. Uczestnicy gry wykonują zlecane przez organizatorów zadania na podstawie wskazówek lub instrukcji. Gry miejskie często posiadają element fabularny, nadający kontekstu wykonywanym zadaniom i pozwalający uczestnikom wcielać się w postaci opowiadanej historii. Tematyka gier może również zawierać motywy historyczne, promujące miasto i zachęcające do poznawania jego kultury, szczególnie jeśli prowadzona jest na terenie jednego miasta.

### 1.1. Cel projektu

Celem projektu jest stworzenie systemu pozwalającego na zorganizowanie i zarządzanie grą miejską oraz branie w niej udziału na terenie wybranego obszaru. System powinien prowadzić grę bez konieczności nadzoru ze strony osób zarządzających.

### 1.2. Zakres projektu

Projekt systemu obejmuje następujące zagadnienia:

- opracowanie zasad gry
- analizę wymagań systemu
- projekt interfejsu użytkownika
- wybór technologii wykonania systemu
- projekt i implementację systemu mobilnej aplikacji klienckiej
- projekt i implementację serwera prowadzącego grę
- projekt i implementację panelu administracyjnego
- przeprowadzenie testów systemu
- stworzenie dokumentacji projektu
- wdrożenie systemu z wykorzystaniem sklepu Google Play

### 1.3. Układ pracy

# Rozdział 2

## Zasady gry

W projektowanej w tej pracy grze zadaniem uczestnika jest fizyczne odnalezienie w wyznaczonym czasie innych uczestników. Po odnalezieniu określonej osoby należy uzyskać od niej hasło, pozwalające na potwierdzenie wykonania zadania. Za wykonywanie zadań uczestnicy są nagradzani punktami, na podstawie których tworzony jest ranking.

Obszar gry nie jest z góry ograniczony przez system. Uczestnicy gry korzystając z aplikacji udostępniają swoją lokalizację, na podstawie której tworzone są dla nich zadania.

### 2.1. Zlecenia

**Zleceniami** nazywane są zadania, które wykonują uczestnicy gry. Celem zadania jest odnalezienie wyznaczonego gracza w określonym czasie na podstawie informacji dostępnych w aplikacji. Wyznaczony gracz nazywany jest **szukanym**, zaś gracz, któremu przydzielone zostaje zlecenie **szukającym**.

Na informacje udostępniane o szukanym składają się: imię, nazwisko, wiek i zdjęcie, a także mapa zawierająca znane lokalizacje szukanego zarejestrowane podczas udziału w grze. Szczegóły dotyczące rejestrowania lokalizacji opisane zostały w punkcie 2.2.

Zlecenia generowane są automatycznie w procesie zwanym **losowaniem**. Losowania odbywają się w stałych odstępach czasu, ustalanych przez administratorów gry. Każdy gracz może posiadać maksymalnie 5 zleceń prywatnych w danym momencie. Jeśli gracz posiada mniej niż 5 zleceń, mogą one zostać uzupełnione do 5 podczas losowania, jeśli pozwala na to liczba graczy. Gracze, którzy najdłużej oczekiwali na nowe zlecenie mają pierwszeństwo w Losowaniu. Szczegóły dotyczące przebiegu losowania i sposobu parowania graczy zostały opisane w punkcie ??.

Szukani są przydzielani szukającym na podstawie ich lokalizacji zarejestrowanej w trakcie uczestnictwa w grze. Gracz może ustalić maksymalną odległość, w jakiej mają mu być przydzielane cele. Jeżeli to możliwe, gracz otrzyma zlecenia natychmiast po rejestracji. W przypadku, gdy po rejestracji gracza nie ma możliwości utworzenia nowych zleceń, gracz zostaje dodany do na początek kolejki graczy oczekujących na losowanie. W przypadku gdy Gracz powraca do gry po nieaktywności, zostaje on dodany na koniec kolejki. W sytuacji, gdy szukany staje się nieaktywny, szukający zostaje dodany na początek Kolejki.

Wykonanie zlecenia lub Listu gończego potwierdza się poprzez wygenerowanie Kodu w aplikacji na urządzeniu szukanego i wprowadzenie go w aplikacji na urządzeniu szukającego. Kod domyślnie jest aktywny przez 2 minuty. W momencie generowania i wpisania kodu oba urządzenia muszą znajdować się w pobliżu siebie. Wylogowanie z aplikacji powoduje wygaśnięcie wszystkich kodów przypisanych do danego użytkownika.

## 2.2. Lokalizacja gracza

Aplikacja mobilna korzysta z danych lokalizacyjnych telefonu. Z tego względu wymagane jest włączenie ich w telefonie podczas użytkowania aplikacji i nadanie aplikacji odpowiednich uprawnień. Jeśli w momencie logowania nie jest możliwe uzyskanie tych danych korzystanie z aplikacji zostanie wstrzymane.

Aplikacja rejestruje pozycje w momencie uruchomienia (logowania) oraz podczas generowania i wprowadzania kodu.

Zarejestrowane przez aplikacje lokalizacje będą udostępniane innym graczom wśród danych na temat celu lub listu gończego oraz będą wykorzystywane przy losowaniach celów do ograniczenia obszaru gry. Gracz może ustalić maksymalną odległość, w jakiej mają mu być przydzielane Zlecenia.

## 2.3. Punktacja

Punkty w grze przydzielane są za:

- wykonanie zlecenia,
- zrealizowanie listu gończego,
- uniknięcie odnalezienia w trakcie trwania zlecenia,
- uniknięcie odnalezienia w trakcie trwania listu gończego.
- codzienne logowanie - gracz otrzymuje punkty za logowanie do aplikacji. Kolejne logowania pod rząd przez kilka dni zwiększają punkty za logowanie.

Punkty uzyskane w grze tracą ważność po upływie 1 roku od ich uzyskania.

Gracze są klasyfikowani w rankingu na podstawie sumarycznej wartości punktów uzyskanych w ciągu ostatniego roku. Graczowi przyznaje się jeden z 5 poziomów rankingowych zależnie od jego miejsca w rankingu. Liczba graczy na każdym z poziomów określona jest procentowo w następujący sposób, zaczynając od graczy z największą liczbą punktów:

- 1% użytkowników
- kolejne 10% użytkowników
- kolejne 20% użytkowników
- kolejne 30% użytkowników
- pozostali użytkownicy

Podczas Losowania gracz ma zwiększoną szansę przydzielenia graczy z tego samego poziomu rankingowego.

## 2.4. Rejestracja do gry

W celu wzięcia udziału w grze wymagane jest stworzenie konta. Rejestracja do gry odbywa się za pomocą aplikacji mobilnej. Podczas rejestracji wymagane jest podanie nazwy użytkownika, hasła, imienia, nazwiska, daty urodzenia, adresu email, krótkiego opisu oraz przesłania swojego zdjęcia. Informacje podane przy rejestracji mogą zostać zmienione w późniejszym czasie z poziomu edycji profilu. Gracze, którzy nie będą aktywni przez 28 dni zostaną tymczasowo wyłączeni z losowań celów i zostaną o tym powiadomieni mailowo. Gracz może powrócić do normalnego trybu rozgrywki z poziomu ustawień aplikacji. Gracze łamiący zasady gry mogą zostać zgłoszeni przez innych graczy z poziomu aplikacji. Po otrzymaniu zgłoszeń od więcej niż X użytkowników, gracz otrzyma powiadomienie mailowe. Po otrzymaniu kolejnych Y zgłoszeń, konto gracza zostanie zablokowane w celu kontroli przez administratora.



## 2.5. Przegląd istniejących rozwiązań

## 2.6. Koncepcja rozwiązania

## 2.7. Słownik pojęć

W tym punkcie przedstawiona została lista pojęć używanych w pracy do określania istotnych elementów lub aspektów rozgrywki.

**Gracz** - zarejestrowany użytkownik systemu nie posiadający uprawnień administracyjnych, biorący udział w rozgrywce

**Administrator** - zarejestrowany użytkownik systemu posiadający uprawnienia do zmiany parametrów gry

**Zlecenie** - zadanie znalezienia innego gracza przydzielane zarejestrowanym użytkownikom, może być prywatne lub publiczne. Zlecenia prywatne widoczne są tylko dla użytkownika, któremu są przydzielone, zaś publiczne dla wszystkich użytkowników

**Szukany** - gracz, którego odnalezienie zostało nakazane w zleceniu lub w Liście gończym

**Szukający** - gracz, któremu przydzielone zostało zlecenie

**Kod** - hasło o krótkim okresie ważności używane do potwierdzania znalezienia celu, generowane w aplikacji

**Aplikacja** lub **Klient** - aplikacja kliencka gry na telefony z systemem Android, przeznaczona do rejestracji i odczytywania zleceń

**Serwer** - aplikacja serwerowa zapewniająca komunikację między klientem a bazą danych, a także zajmująca się prowadzeniem gry

**Losowanie zleceń** - procedura dobierania w pary szukanych i szukających, której wynikiem jest utworzenie nowego zlecenia

**Wykonanie zlecenia** - znalezienie szukanego gracza i wprowadzenie wygenerowanego przez niego kodu do aplikacji

**Kolejka** - lista graczy oczekujących na losowanie, posiadających mniej niż maksimum zleceń prywatnych.

**Ranking** - określana na podstawie punktacji klasyfikacja graczy

**Poziomy rankingowe** - wyróżnione w rankingu stopie osiągnięte przez zdobywanie punktów

# Rozdział 3

## Analiza wymagań systemu

### 3.1. Architektura systemu

System zostanie podzielony na cztery części:

- aplikację mobilną na telefony z systemem Android,
- aplikację internetową realizującą funkcje konsoli administracyjnej
- serwer prowadzący grę
- bazę danych

Aplikacja mobilna będzie pełnić rolę klienta gry i przeznaczona będzie dla graczy. Z jej pomocą użytkownicy będą mogli odczytywać i wykonywać przydzielone im zlecenia, edytować swój profil, przeglądać ranking i zarządzać swoim kontem. Urządzenie mobilne zostanie wykorzystane do określenia lokalizacji użytkowników na potrzeby gry.

Konsola administracyjna służyć będzie administratorom gry to obserwowania jej przebiegu i modyfikowania parametrów, a także zarządzania kontami użytkowników.

Zadaniem serwera jest automatyczne prowadzenie gry bez konieczności udziału administratorów systemu. W ramach pracy serwera powinny być generowane lub usuwane zlecenia prywatne i publiczne, wysyłane powiadomienia do graczy, a także monitorowana aktywność graczy. Serwer wykorzystywany będzie także do stworzenia interfejsu komunikacyjnego pomiędzy aplikacjami klienckimi lub konsolą administracyjną a bazą danych.

### 3.2. Wymagania funkcjonalne

Na podstawie analizy opisu zasad gry i planowanej architektury systemu, opracowano wymagania funkcjonalne systemu. Poniższe podrodziały opisują opracowane wymagania z wyróżnieniem części systemu, do której przynależą.

#### 3.2.1. Aplikacja kliencka

Wymagania funkcjonalne aplikacji klienckiej skupiają się wokół zapewnienia użytkownikom wglądu w ich konto.

1. Logowanie - aplikacja powinna pozwalać na zalogowanie istniejącego użytkownika przy pomocy nazwy użytkownika i hasła. W aplikacji należy umożliwić zapamiętanie danych logowania. Po zapamiętaniu danych logowanie do konta użytkownika powinno nastąpić automatycznie po uruchomieniu aplikacji. Po wprowadzeniu nieprawidłowych danych logowania aplikacja powinna wyświetlić stosowny komunikat. Po poprawnym logowaniu aplikacja przechodzi do zapisu lokalizacji.

2. Rejestracja - aplikacja powinna umożliwiać rejestrację nowego konta. W procesie rejestracji należy podać dane osobowe oraz dane logowania, a także wybrać zdjęcie. Po wyborze zdjęcia aplikacja powinna umożliwić przycięcie go. Po kliknięciu w pole Data urodzenia powinien wyświetlić się kalendarz. Typ pola do wprowadzenia danych powinien odpowiadać wprowadzanym danym (email, hasło, imię itd), tak aby na urządzeniu wyświetliła się właściwa klawiatura. Przed zarejestrowaniem użytkownika należy zweryfikować poprawność wprowadzonych danych oraz zdjęcia. W przypadku niewprowadzenia któregoś z elementów lub braku weryfikacji należy wyświetlić komunikat i zaznaczyć brakujący element. Po poprawnym zakończeniu rejestracji aplikacja powinna powrócić do ekranu startowego.
3. Wykrycie twarzy na zdjęciu - aplikacja przed przesłaniem zdjęcia na serwer powinna dokonać próby wykrycia twarzy na zdjęciu. W przypadku niewykrycia twarzy, wykrycia więcej niż jednej twarzy, lub za małej powierzchni zdjęcia zajętej przez twarz należy wyświetlić komunikat z prośbą o wybór innego zdjęcia lub inne jego przycięcie. Po poprawnym wykryciu twarzy na zdjęciu należy wyświetlić informację o poprawności zdjęcia na ekranie.
4. Przesłanie zdjęcia - aplikacja powinna umożliwić przesłanie zdjęcia na serwer. W przypadku błędu podczas przesyłania należy wyświetlić komunikat o błędzie. Po poprawnym przesłaniu zdjęcia należy przypisać jego identyfikator użytkownikowi.
5. Zapis lokalizacji przy logowaniu - po zalogowaniu użytkownika należy zapisać lokalizację do bazy danych. Jeśli aplikacja nie posiada uprawnień do modułu GPS urządzenia, należy o nie poprosić. Jeśli użytkownik odmówił uprawnień lub moduł GPS jest wyłączony należy wyświetlić ekran z informacją o konieczności jego włączenia. Po uzyskaniu dostępu do modułu aplikacja powinna poczekać na ustalenie lokalizacji z dobrą dokładnością (kilka metrów) a następnie wykonać zapis do bazy. Po poprawnym wykonaniu zapisu aplikacja przechodzi do widoku Zlecenia. W przypadku błędu zapisu do bazy ???
6. Wyświetlenie Zleceń - aplikacja powinna umożliwiać wyświetlenie Zleceń użytkownika z podziałem na publiczne i prywatne. Zlecenia powinny być pobrane automatycznie po wyświetleniu ekranu. Należy umożliwić odświeżenie listy poprzez gest pociągnięcia w dół, gdy lista jest przewinięta do początku. Elementy listy powinny wyświetlać dane osobowe celu, zdjęcie, odległość od użytkownika oraz pozostały czas. Pozostały czas zlecenia powinien być aktualizowany w czasie rzeczywistym. Zlecenia zakończone powinny zamiast czasu wyświetlać napis z ich stanem. W przypadku błędu podczas pobierania listy celów należy wyświetlić komunikat o błędzie. Kliknięcie na element listy powinno otworzyć ekran szczegółów zlecenia.
7. Szczegóły zlecenia - aplikacja powinna wyświetlić szczegóły zlecenia po wyborze z listy Zleceń. Zdjęcie celu powinno zajmować możliwie dużą powierzchnię ekranu. Na ekranie należy wyświetlić informacje o celu, odległość od użytkownika i pozostały czas. Pozostały czas powinien być aktualizowany w czasie rzeczywistym. Ekran Szczegóły powinien zawierać pole tekstowe oraz przycisk służące do wprowadzenia kodu potwierdzającego. Gest pociągnięcia w dół na ekranie powinien pokazać mapę zawierającą lokalizację Szukanego. Po kliknięciu na marker lokalizacji na mapie powinno zostać wyświetlone okno z datą dodania znacznika. Ekran Szczegółów powinien umożliwiać raportowanie użytkowników łamiących zasady.
8. Potwierdzenie wykonania Zlecenia - użytkownik powinien mieć możliwość potwierdzenia wykonania Zlecenia poprzez wprowadzenie kodu generowanego w aplikacji szukanego użytkownika. Po wprowadzeniu kodu aplikacja powinna przesłać kod na serwer w celu sprawdzenia jego poprawności w bazie danych. W przypadku kodu niepoprawnego należy wyświetlić komunikat oraz zablokować możliwość ponownego wprowadzenia kodu na 10 sekund. Po wprowadzeniu poprawnego kodu Zlecenia zmienia status na wykonane.

9. Generowanie kodu - aplikacja umożliwia wygenerowanie losowego kodu służącego do potwierdzania zleceń. W momencie generowania aplikacja powinna zapamiętać czas generowania i zapisać go w bazie danych. Na ekranie powinien zostać wyświetlony czas ważności kodu. Czas ważności powinien być aktualizowany w czasie rzeczywistym.
10. Zgłoszenie gracza - aplikacja powinna umożliwiać zgłoszenie gracza łamiącego zasady gry. Po wyborze opcji Zgłoś gracza powinno pokazać się okno dialogowe z możliwością wyboru jednego powodów zgłoszenia, polem tekstowym do opisanie sytuacji i przyciskami potwierdzającym i zamykającym okno dialogowe. Po potwierdzeniu zgłoszenie zostanie przesłane na serwer, a okno dialogowe powinno zostać zamknięte.
11. Nawigacja w aplikacji - aplikacja powinna umożliwiać nawigowanie pomiędzy dostępnymi ekranami przy pomocy menu bocznego wysuwanego z lewej strony. Kliknięcie elementu listy powinno zmienić ekran i zamknąć menu. Kliknięcie poza wysuniętym menu lub gest przeciągnięcia w lewo powinny zamknąć menu.
12. Ocena - aplikacja powinna posiadać skrót do strony aplikacji w sklepie Google Play.
13. Strona "O aplikacji" - aplikacja powinna umożliwiać wyświetlenie strony informacyjnej o aplikacji zawierającej dane na temat obecnej wersji, autora, wykorzystanych bibliotek i ich licencji.
14. Kontakt z twórcą - aplikacja powinna umożliwiać przesłanie informacji o błędach lub opinii do twórcy aplikacji. Przesyłanie może być obsługiwane za pomocą zewnętrznej aplikacji pocztowej lub wbudowane w aplikację.
15. Profil użytkownika - aplikacja powinna udostępnić ekran wyświetlający profil zalogowanego użytkownika w ten sam sposób, w jaki wyświetlane są szczegóły zlecenia, z pominięciem elementów interfejsu dotyczących czasu zlecenia i kodu potwierdzającego. Widok profilu użytkownika powinien pozwolić na edycję danych użytkownika oraz zmianę zdjęcia. Przy edycji profilu powinna być możliwość zmiany maksymalnej odległości zleceń.
16. Strona startowa - aplikacja powinna zawierać ekran startowy zawierający grafikę promocyjną i przyciski Zarejestruj oraz Zaloguj. Strona startowa powinna być wyświetlana przy pierwszym uruchomieniu aplikacji, oraz przy kolejnych jeśli użytkownik nie skorzystał z opcji automatycznego logowania.
16. Wylogowanie - aplikacja powinna umożliwiać wylogowanie z konta użytkownika. Po wylogowaniu dotychczas zapamiętana nazwa użytkownika i hasło powinny zostać zapomniane. Przy kolejnym uruchomieniu aplikacja powinna wyświetlić ekran startowy.
17. Szkolenie - przy pierwszym uruchomieniu aplikacji powinno zostać wyświetlone krótkie szkolenie w postaci slajdów opisujące krótko najważniejsze funkcje aplikacji oraz zasady gry.
18. Ustawienia aplikacji - w aplikacji powinna być możliwość edycji preferencji dotyczących otrzymywanych powiadomień.

### 3.2.2. Aplikacja serwerowa

1. Połączenie z bazą danych - system w ramach swojego działania powinien mieć możliwość nawiązania połączenia i wykonywania operacji na bazie danych.
2. Rejestracja - system pozwala na zarejestrowanie nowego konta. Podczas rejestracji należy nadać kontu odpowiednie uprawnienia.
3. Uwierzytelnienie użytkownika - system powinien być zabezpieczony przed dostępem osób nieuwierzytelnionych. W celu uzyskania dostępu użytkownik musi podać login i hasło wybrane przy rejestracji. System powinien działać w trybie bezstanowym tj. nie przechowywać informacji o zalogowanym użytkowniku lub sesji.
4. Sprawdzenie dostępności nazwy użytkownika - system powinien umożliwiać sprawdzenie zadanej nazwy użytkownika pod kątem występowania konta o tej nazwie w bazie danych.

5. Sprawdzenie adresu email - system powinien umożliwić sprawdzenie zadanego adresu email pod kątem występowania konta z tym adresem w bazie danych.
6. Pobranie konta użytkownika - system powinien umożliwić pobranie danych o koncie użytkownika na podstawie jego numeru id lub nazwy użytkownika.
7. Pobranie zleceń użytkownika - system powinien umożliwić pobranie zleceń dostępnych dla użytkownika na podstawie jego numeru id.
8. Pobranie lokalizacji użytkownika - system powinien umożliwić pobranie znanych lokalizacji użytkownika nowszych niż zadana data.
9. Zapis lokalizacji - system powinien umożliwić zapis nowej lokalizacji do bazy danych. We wpisie należy zawrzeć datę jej utworzenia, użytkownika, któremu jest przypisana, a także informację o typie wpisu (logowanie, wykonanie zlecenia, wygenerowanie kodu).
10. Zapis kodu - system powinien umożliwić zapis nowo wygenerowanego kodu potwierdzającego wykonanie zlecenia do bazy danych. We wpisie należy zawrzeć informację o dacie wykonania wpisu oraz o użytkowniku, który wygenerował kod.
11. Potwierdzenie wykonania zlecenia - system powinien umożliwić potwierdzenia wykonania zlecenia za pomocą kodu wygenerowanego w aplikacji Szukanego.
12. Pobranie punktacji użytkownika - system powinien umożliwić pobranie informacji o sumarycznej liczbie punktów uzyskanych przez użytkownika oraz miejscu w rankingu na podstawie tej punktacji.
13. Pobranie przedziałów punktowych - system powinien umożliwić wyznaczenie przedziałów punktowych na podstawie progów wymienionych w ..... . Użytkownik powinien mieć możliwość sprawdzenia liczby punktów potrzebnych do zmiany obecnie zajmowanego przez niego przedziału na wyższy lub niższy.
14. Pobranie rankingu najlepszych graczy - system powinien umożliwić pobranie informacji o najlepszych graczach w rankingu.
15. Pobranie uśrednionej lokalizacji - system powinien umożliwić pobranie uśrednionej lokalizacji użytkownika na podstawie jego numeru id. Lokalizację uśrednioną należy wyznaczyć poprzez znalezienie takiej lokalizacji użytkownika, która w swoim otoczeniu ma maksymalną liczbę innych lokalizacji. W przypadku, gdy liczba ta jest równa dla dwóch lub więcej wpisów, należy wybrać najbardziej aktualną.
16. Pobranie najnowszej lokalizacji - system powinien umożliwić pobranie najbardziej aktualnej lokalizacji użytkownika na podstawie jego numeru id.
17. Edycja profilu - system powinien umożliwiać edycję danych na koncie użytkownika
18. Zgłoszenie łamania zasad - system powinien umożliwiać dodanie nowego zgłoszenia o złamaniu zasad przez użytkownika. Danymi wymaganymi do dodania zgłoszenia jest id użytkownika zgłaszającego oraz zgłoszonego i powód zgłoszenia. Opcjonalne jest dodanie opisu słownego. System powinien zarejestrować datę przyjęcia zgłoszenia.
19. Zamrożenie i odmrożenie konta - w przypadku wydłużonej nieaktywności Gracz powinien mieć możliwość wstrzymania działalności na swoim koncie poprzez jego zamrożenie. Po ponownym rozpoczęciu aktywności system powinien umożliwić odmrożenie konta. Konta zamrożone nie biorą udziału w rozgrywce.
20. Usunięcie konta - system powinien umożliwić użytkownikowi usunięcie swojego konta. Po usunięciu konta powinny zostać także usunięte wszystkie inne wpisy w bazie danych powiązane z usuwanym kontem.
21. Zmiana hasła - system powinien umożliwić zmianę hasła użytkownikowi. Po prośbie o zmianę hasła wysłanej z poziomu aplikacji mobilnej system powinien wysłać email z linkiem potwierdzającym zmianę. Po wejściu w link należy wygenerować losowo nowe hasło i wysłać je mailowo użytkownikowi.
22. Utworzenie nowych zleceń - system powinien automatycznie co określony czas przeprowadzić losowanie nowych zleceń dla użytkowników. Zasady i przebieg losowania opisane

- zostały w punkcie ..... . Czas, co jaki tworzone są zlecenia jest parametrem systemu i należy go pobrać z bazy danych po każdym losowaniu.
23. Wygaśnięcie zleceń - system powinien automatycznie zmieniać status aktywnych zleceń na niewykonane, w przypadku gdy zlecenie pozostaje niewykonane po przekroczeniu czasu ważności. Czas ważności zlecenia publicznego to parametr systemu, który należy pobrać z bazy danych.
  24. Zmiana widoczności zleceń - system powinien automatycznie zmieniać widoczność zleceń z prywatnych na publiczne, w przypadku gdy zlecenie prywatne pozostaje niewykonane po przekroczeniu czasu ważności. Czas ważności zlecenia prywatnego to parametr systemu, który należy pobrać z bazy danych.
  25. Wygasanie kodów - system powinien automatycznie zmienić status kodów w bazie danych na nieaktualne po upływie ich czasu ważności. Czas ważności kodu to parametr systemu, który należy pobrać z bazy danych.
  26. Wykrywanie nieaktywnych użytkowników - system powinien automatycznie wykrywać długie okresy nieaktywności użytkowników. Użytkownik po przekroczeniu określonego czasu nieaktywności powinien otrzymać mailową informację o nieaktywności. W przypadku braku aktywności przez kolejny okres czasu, konto użytkownika należy zamrozić. Użytkownik powinien zostać poinformowany o zamrożeniu konta drogą mailową. Czas, po którym wysyłane jest przypomnienie i czas zamrożenia konta to parametry systemu, które należy pobrać z bazy danych.
  27. Blokada użytkowników - system powinien automatycznie blokować konta użytkowników, którzy przekroczą określoną liczbę zgłoszeń o złamaniu regulaminu. Wielokrotne zgłoszenia od tego samego użytkownika należy liczyć pojedynczo. Użytkownik, którego konto zostało zablokowane powinien otrzymać powiadomienie mailowe.
  28. Wysyłanie powiadomień push - system powinien mieć możliwość wysyłania powiadomień push na telefony z systemem Android. Zdarzenia powodujące wysłanie powiadomień zostały opisane w punkcie ..... Powiadomienia.
  29. Zmiana parametrów systemu - system powinien udostępniać API pozwalające na zmianę zadanego parametru systemu przez podanie jego nazwy oraz nowej wartości. Nazwy parametrów zostały opisane w punkcie ..... Parametry systemu.
  30. Odczyt parametrów systemu - system powinien pozwalać na odczyt ustawionej wartości parametru na podstawie jego nazwy.
  - 31.

### 3.2.3. Panel administracyjny

1. Kontrola dostępu - panel administracyjny powinien pozwalać na dostęp jedynie użytkownikom posiadającym odpowiednie uprawnienia.
2. Logowanie - panel powinien pozwalać na logowanie użytkownika
3. Wyświetlenie listy użytkowników - z poziomu panelu administrator powinien mieć możliwość pobrania i wyświetlenia listy użytkowników. Należy zapewnić możliwość filtrowania listy na podstawie kryteriów: statusu konta, daty rejestracji, daty ostatniej aktywności.
4. Wyświetlenie listy zgłoszeń - administrator powinien mieć możliwość wyświetlenia listy zgłoszeń o złamaniu zasad. Należy zapewnić możliwość filtrowania listy na podstawie kryteriów: zgłoszony użytkownik, zgłaszający użytkownik, data zgłoszenia, rodzaj zgłoszenia.
5. Zmiana statusu konta - administrator powinien mieć możliwość zablokowania lub odblokowania konta wybranego użytkownika

6. Zmiana uprawnień administracyjnych - administrator powinien mieć możliwość nadania lub cofnięcia uprawnień administracyjnych wybranego użytkownika poza wyznaczonym kontem nadrzędnym
7. Zmiana parametrów gry - panel administracyjny powinien pozwalać na odczyt i zmianę parametrów systemu. Lista parametrów systemu przedstawiona została w punkcie ??.

### **3.3. Wymagania niefunkcjonalne**

### **3.4. Projekt interfejsu**

#### **3.4.1. Aplikacja kliencka**

#### **3.4.2. Panel administracyjny**

### **3.5. Wykorzystywane technologie i narzędzia**

## **Rozdział 4**

# **Implementacja systemu**



# Literatura

# **Dodatek A**

## **Tytuł dodatku**

Zasady przyznawania stopnia naukowego doktora i doktora habilitowanego w Polsce określa ustawa z dnia 14 marca 2003 r. o stopniach naukowych i tytule naukowym oraz o stopniach i tytule w zakresie sztuki (Dz.U. nr 65 z 2003 r., poz. 595 (Dz. U. z 2003 r. Nr 65, poz. 595)). Poprzednie polskie uregulowania nie wymagały bezwzględnie posiadania przez kandydata tytułu zawodowego magistra lub równorzędnego (choć zasada ta zazwyczaj była przestrzegana) i zdarzały się nadzwyczajne przypadki nadawania stopnia naukowego doktora osobom bez studiów wyższych, np. słynnemu matematykowi lwowskiemu – późniejszemu profesorowi Stefanowi Banachowi.

W innych krajach również zazwyczaj do przyznania stopnia naukowego doktora potrzebny jest dyplom ukończenia uczelni wyższej, ale nie wszędzie.

## **Dodatek B**

# **Opis załączonej płyty CD/DVD**

Tutaj jest miejsce na zamieszczenie opisu zawartości załączonej płyty. Należy wymienić, co zawiera.