

14.01.2023 r.



# **Wyższa Szkoła Bankowa we Wrocławiu**

## **Wydział Finansów i Zarządzania**

Kierunek: Informatyka inżynierska

Anna Myszkier 73105

### **Aplikacja na urządzenia mobilne z systemem Android do zarządzania wynajmem nieruchomości**

Inżynierska praca projektowa

**Opiekun naukowy:**

mgr inż. Stanisław Lota

Wrocław 2023

## Spis treści

1.	Wstęp.....	3
1.1.	Analiza rozwiązań dostępnych na rynku .....	4
1.2.	Analiza SWOT .....	5
2.	Założenia projektowe .....	6
2.1.	Wymagania funkcjonalne i нефункционалне według zasady MoSCoW .....	6
2.2.	Narzędzia i technologie .....	6
2.3.	Diagramy UML i diagram ERD bazy danych .....	7
2.4.	UX .....	7
3.	Implementacja i wdrożenie.....	11
4.	Testy, bezpieczeństwo, archiwizacja.....	11
5.	Podsumowanie.....	11

# 1. Wstęp

Rozwój urządzeń mobilnych na przestrzeni ostatnich lat stworzył zapotrzebowanie na aplikacje, do których dostęp można mieć w dowolnym miejscu i czasie – zarówno te, które pomagają w karierze i edukacji, jak i te ułatwiające organizację życia codziennego. Obecnie możemy wyróżnić dwa najpopularniejsze systemy operacyjne dla urządzeń mobilnych – Android, zaprojektowany i rozwijany przez firmę Google oraz iOS, zaprojektowany i rozwijany przez firmę Apple.

Rosnące ceny nieruchomości i coraz wyższe koszty kredytów hipotecznych, a także mniejsza tendencja wśród społeczeństwa do długoterminowego mieszkania w jednym miejscu, spowodowała ogromny wzrost dostępnych mieszkań i pokoi na wynajem. Celem niniejszej pracy dyplomowej jest wykonanie aplikacji mobilnej na platformę Android, która ułatwi zarządzanie wynajmowaniem nieruchomości, zarówno z perspektywy wynajmujących, jak i najemców.

Aplikacja ta pozwoli na wyliczanie miesięcznej kwoty do zapłaty w oparciu o opłaty stałe oraz opłaty zależne od zużycia dla każdego z najemców nieruchomości. Kwoty opłat zależnych od zużycia będą obliczane na podstawie wartości liczników, wprowadzanych co miesiąc przez jednego z najemców. Najemcy będą mieli również możliwość zgłaszania usterek w mieszkaniu. Dodatkowo, wdrożona zostanie również funkcjonalność kalendarza, która może być wykorzystywana przez wynajmującego do poinformowania o wizycie w mieszkaniu lub planowanych naprawach, a przez najemców do poinformowania o dłuższej nieobecności, która będzie uwzględniona przy wyliczaniu opłat zależnych od zużycia. Korzystanie z aplikacji będzie bezpłatne.

Zakres pracy obejmuje zapoznanie się z dokumentacją techniczną dotyczącą: programowania w języku Kotlin na platformę Android, tworzenia interfejsu użytkownika za pomocą Jetpack Compose, oraz architektury systemu. Ustalone zostaną również:

- potrzeby użytkowników, które powinna zaspokajać aplikacja
- funkcjonalności, których aplikacja będzie potrzebować
- w jakim kontekście i przez kogo aplikacja będzie używana
- co stanowi o jej atrakcyjności

Z założenia aplikacja ma wykorzystywać zewnętrzną usługę Google Firebase do rejestracji, logowania i weryfikacji tożsamości użytkowników (moduł Google Firebase Authentication) oraz przechowywania informacji o nieruchomościach, kosztach stałych i zmiennych utrzymania nieruchomości, opłatach najemców oraz wartościach liczników obsługujących nieruchomość (moduł Google Firebase Database).

W pierwszych rozdziałach niniejszej pracy przedstawiono informację na temat analizy i strategii projektowania aplikacji. Przedstawiono również wymagania funkcjonalne i niefunkcjonalne aplikacji. W rozdziale trzecim przedstawiono architekturę aplikacji na platformę Android oraz narzędzia wykorzystane do budowy aplikacji. W rozdziale czwartym omówiono interfejs aplikacji i funkcjonalności wraz z opisem budowy poszczególnych ekranów. W rozdziale piątym przedstawiono informacje o przeprowadzonych testach manualnych aplikacji. Ostatni rozdział skupia się na podsumowaniu i prognozie dalszych prac nad aplikacją.

### 1.1. Analiza rozwiązań dostępnych na rynku

W celu analizy dostępnych na polskim rynku rozwiązań dotyczących zarządzania najmem, w sklepie Google Play wpisano frazy „zarządzanie najmem”, „zarządzanie wynajmem” i „obsługa najmu” , ale nie znaleziono żadnych aplikacji oferujących usługi zarządzania nieruchomościami pod kątem wynajmu. Dokonując głębszej analizy, udało się znaleźć aplikację „System Obsługi Najmu”, która okazała się aplikacją dostępną tylko dla użytkowników portalu System Obsługi Najmu, posiadających płatną subskrypcję. Nie udało się zatem dokonać analizy samej aplikacji.

Przeprowadzono również analizę rynku zagranicznego pod kątem istniejących rozwiązań, wyszukując w sklepie Google Play frazy „property management”. Znaleziono aplikację „Landlordly”. Po analizie aplikacji okazało się, że aplikacja pozwala na zarządzanie wynajmem tylko z perspektywy właściciela nieruchomości, a wersja bezpłatna aplikacji pozwala na dodanie maksymalnie jednego lokatora dla danej nieruchomości.

## 1.2. Analiza SWOT

Silne i słabe strony aplikacji, a także zewnętrzne szanse i zagrożenia zostały określone zgodnie z zasadami analizy SWOT. Wyniki analizy zostały przedstawione w tabeli 1.

<b>Silne strony:</b>  1. Pomysł na aplikację zrodził się z realnego zapotrzebowania osoby wynajmującej nieruchomość. 2. Uczciwe rozliczanie opłat lokatorów według zużycia, a nie na podstawie zaliczek.	<b>Słabe strony:</b>  1. Aplikacja rozwijana jest przez studenta, który nie ma doświadczenia komercyjnego w budowaniu aplikacji mobilnych. 2. Aplikacja jest rozwijana w czasie wolnym, co może negatywnie wpłynąć na tempo jej rozwoju.
<b>Szanse:</b>  1. Wzrost popularności wynajmu generuje zapotrzebowanie na aplikacje mobilne pozwalające na zarządzanie najmem. 2. Na polskim rynku nie ma bezpłatnych aplikacji mobilnych do zarządzania wynajmem nieruchomości.	<b>Zagrożenia:</b>  1. Firmy zajmujące się projektowaniem aplikacji mobilnych mogą przejąć pomysł i być w stanie stworzyć lepszą aplikację ze względu na posiadane zasoby. 2. Lokatorzy wynajmowanych nieruchomości mogą nie chcieć używać aplikacji. 3. Dostawcy prądu, gazu i wody nie udostępniają API z taryfami – wartości te właściciele będą musieli wpisywać ręcznie do aplikacji.

*Tabela 1 - Wyniki analizy SWOT dla aplikacji.*

## 2. Założenia projektowe

### 2.1. Wymagania funkcjonalne i niefunkcjonalne według zasady MoSCoW

Założenia funkcjonalne i niefunkcjonalne aplikacji zostały określone według priorytetu zgodnie z techniką priorytetyzacji MoSCoW. Wyniki priorytetyzacji zostały przedstawione na rysunku 1.



Rysunek 1 – Wymagania funkcjonalne i niefunkcjonalne aplikacji według metody MOSCOW.

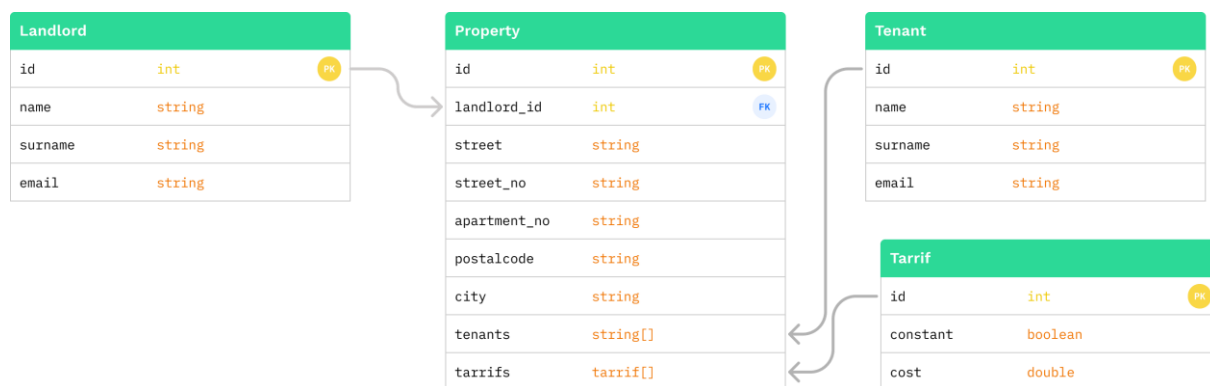
### 2.2. Narzędzia i technologie

Aplikacja będzie rozwijana na urządzenia z systemem Android, a jej kod zostanie napisany w języku Kotlin. Środowiskiem, w którym rozwijana będzie aplikacja będzie Android Studio. Interfejs użytkownika zostanie zaimplementowany z użyciem frameworku Jetpack Compose, zgodnego z zalecanym przez Google systemem projektowania Material Design 3. Do logowania i rejestracji użytkowników zostanie zaimplementowana usługa Google Firebase

Authentication, a informacje o nieruchomości, lokatorach, wartościach liczników, opłatach i zgłaszanych usterkach będą przechowywane w bazie danych przy implementacji Google Firebase Database.

## 2.3. Diagramy UML i diagram ERD bazy danych

Diagram klas przedstawiający relacje między klasami w bazie danych aplikacji został przedstawiony na rysunku 2.



Rysunek 2 – Diagram klas aplikacji

## 2.4. UX

Interfejs użytkownika został zaprojektowany w edytorze grafiki wektorowej Figma, zgodnie z systemem projektowania Material Design 3 stworzonym przez firmę Google. W samej aplikacji, interfejs ten został wdrożony za pomocą frameworku Jetpack Compose.

Przykładowe ekrany aplikacji zostały przedstawione na rysunkach poniżej.

20:01

APPartmenty

Welcome!

E-mail address

First name

Last name

Password

Confirm password

Register

*Rysunek 3 – Ekran rejestracji*

20:00

APPartmenty

Nice to see you again!

E-mail address

Password

Login

*Rysunek 4 – Ekran logowania*





*Rysunek 5 – Ekran powitalny (z perspektywy właściciela nieruchomości)*



*Rysunek 6 – Ekran powitalny (z perspektywy lokatora)*

20:31 37%

## Address

Street

House no.

Apartment no.

Postal code

City

## Utilities

☒ Electricity Price [PLN/kWh]

☒ Gas Price [PLN/m3]

☒ Water Price [PLN/m3]

☒ Internet Price [monthly]

Confirm and save

Rysunek 7 – Ekran dodawania nowej nieruchomości (z perspektywy właściciela nieruchomości)

← Wprowadź odczyty

Data DD/MM/YYYY

Prąd Odczyt [kWh]

Woda Odczyt [m3] Wprowadź wartość

+ Dodaj zdjęcia liczników

Zatwierdź i zapisz

Rysunek 8 – Ekran wprowadzania odczytów liczników (z perspektywy lokatora) - prototyp

### 3. Implementacja i wdrożenie

Dokładna implementacja i wdrożenie zostaną opisane na dalszym etapie prac nad aplikacją.

### 4. Testy, bezpieczeństwo, archiwizacja

Bezpieczeństwo danych aplikacji oraz archiwizacja danych została zapewniona przez wdrożenie do aplikacji usługi Google Firebase. Przypadki testowe zostaną opisane na dalszym etapie prac nad aplikacją.

### 5. Podsumowanie

Na obecnym etapie prac nad aplikacją zostały stworzone ekrany (w Android Studio nazywane „Activities”) pozwalające w przyszłości realizowanie zdefiniowanych wymagań aplikacji. W kolejnych etapach prac nad aplikacją zostanie wdrożone połączenie z usługą Google Firebase pozwalające na rejestrację i logowanie użytkowników oraz zapisywanie danych w bazie danych.

### Spis ilustracji

Rysunek 1 – Wymagania funkcjonalne i нефункционалне aplikacji według metody MOSCOW.....	6
Rysunek 2 – Diagram klas aplikacji.....	7
Rysunek 3 – Ekran rejestracji .....	8
Rysunek 4 – Ekran logowania.....	8
Rysunek 5 – Ekran powitalny (z perspektywy właściciela nieruchomości) .....	9
Rysunek 6 – Ekran powitalny (z perspektywy lokatora) .....	9
Rysunek 7 – Ekran dodawania nowej nieruchomości (z perspektywy właściciela nieruchomości).....	10
Rysunek 8 – Ekran wprowadzania odczytów liczników (z perspektywy lokatora) - prototyp .....	10