|  |  |
| --- | --- |
| **Politechnika Świętokrzyska**  **Wydział Elektrotechniki, Automatyki i Informatyki** | |
| **Laboratorium:** Wprowadzenie do komunikacji człowiek-komputer  **Interfejs:** Opis interfejsu, ekranów, diagram UML i scenariusze przypadków użycia. | |
| **Wykonawca:**  Przemysław Kałuziński  Michał Kaczor | **Grupa:**  3ID13A |
| **Data wykonania:** 03.04.2023 | |

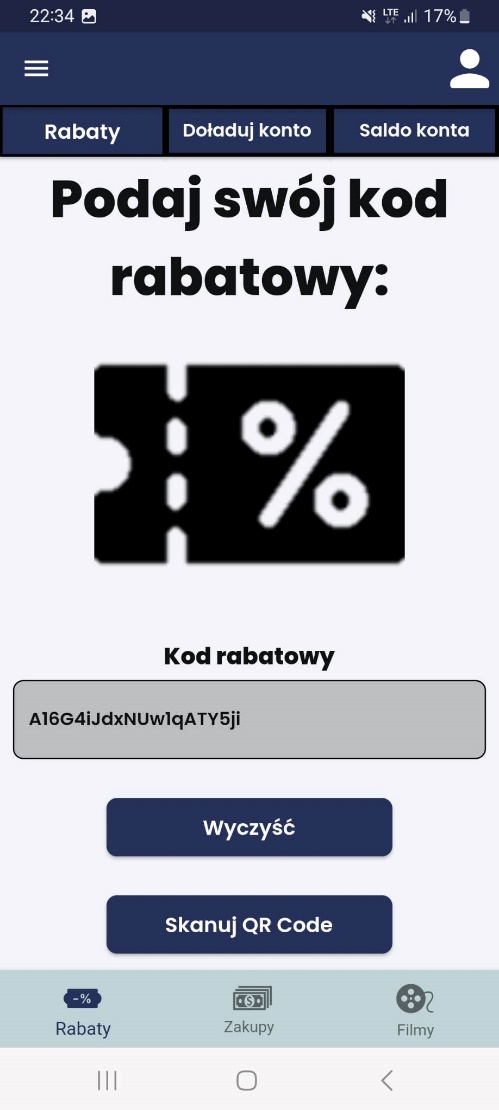
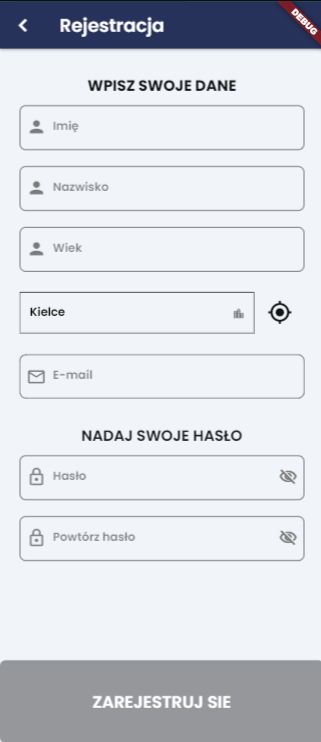
# **Opis wybranego tematu**

Wybranym przez nas tematem jest aplikacja użytkowa, umożliwiająca klientowi korzystanie z usług kinowych oraz streamingowych.Oferuje ona funkcjonalności zawarte w założeniach. Filmy i trailery wyświetlane w naszej aplikacji będą hostowane na serwisie rozrywkowo multimedialnym Youtube. Filmiki będą wyświetlane przy pomocy specjalnego odtwarzacza bezpośrednio w naszej aplikacji – bez przechodzenia do aplikacji Youtube. Część serwerową aplikacji postanowiliśmy rozwiązać przy pomocy zestawu usług hostingowych Firebase, który pozwoli nam na przechowywanie kluczowych informacji o kontach użytkowników oraz odnośników do filmów i trailerów.

# **Urządzenia**

Aplikacja będzie korzystać z poniższych urządzeń telefonu:

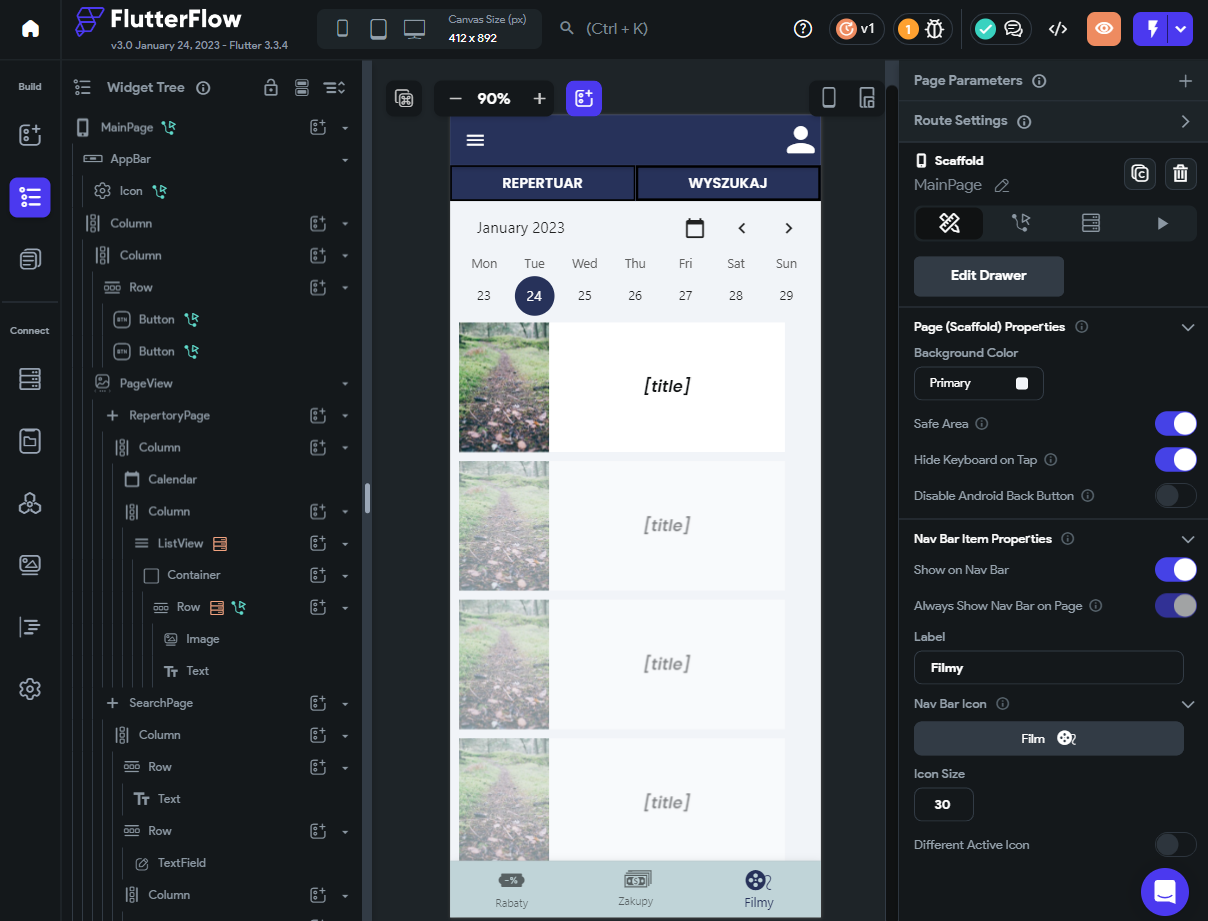
* Czytnik kodów QR, który będzie wykorzystywany do wczytywania kodów rabatowych
* GPS, który będzie wykorzystywany do zlokalizowania użytkownika i przypisania mu najbliższej placówki kinowej

# **Opis narzędzi i środowisk**

## **Flutterflow**

Do prac nad projektem postanowiliśmy wykorzystać zaawansowane narzędzie programistyczne – Flutterflow. Jest to platforma do tworzenia aplikacji mobilnych i webowych za pomocą graficznego interfejsu użytkownika. Narzędzie to wykorzystuje popularny framework Flutter, który pozwala na szybkie tworzenie responsywnych aplikacji z użyciem natywnych elementów interfejsu użytkownika. W naszym projekcie wykorzystujemy Flutterflow jako główne narzędzie do projektowania i budowania aplikacji mobilnej. Dzięki temu możemy w łatwy sposób tworzyć interaktywne elementy interfejsu użytkownika, przeciągać i upuszczać elementy, łączyć je ze sobą i konfigurować zachowanie aplikacji za pomocą wbudowanych w narzędzie funkcji. Korzystając z jego prostego interfejsu typu „przeciągnij i upuść”, można zbudować w pełni funkcjonalną aplikację o wiele prościej niż w normalny sposób. Pozwala ono także testować aplikację w emulatorze telefonu w przeglądarce internetowej. Dzięki temu, nie trzeba za każdym razem debugować projektu przy użyciu kabla lub innego emulatora. Środowisko to pozwala także na stosowanie własnego kodu w postaci funkcji, które realizują rzeczy niewspierane przez narzędzie.

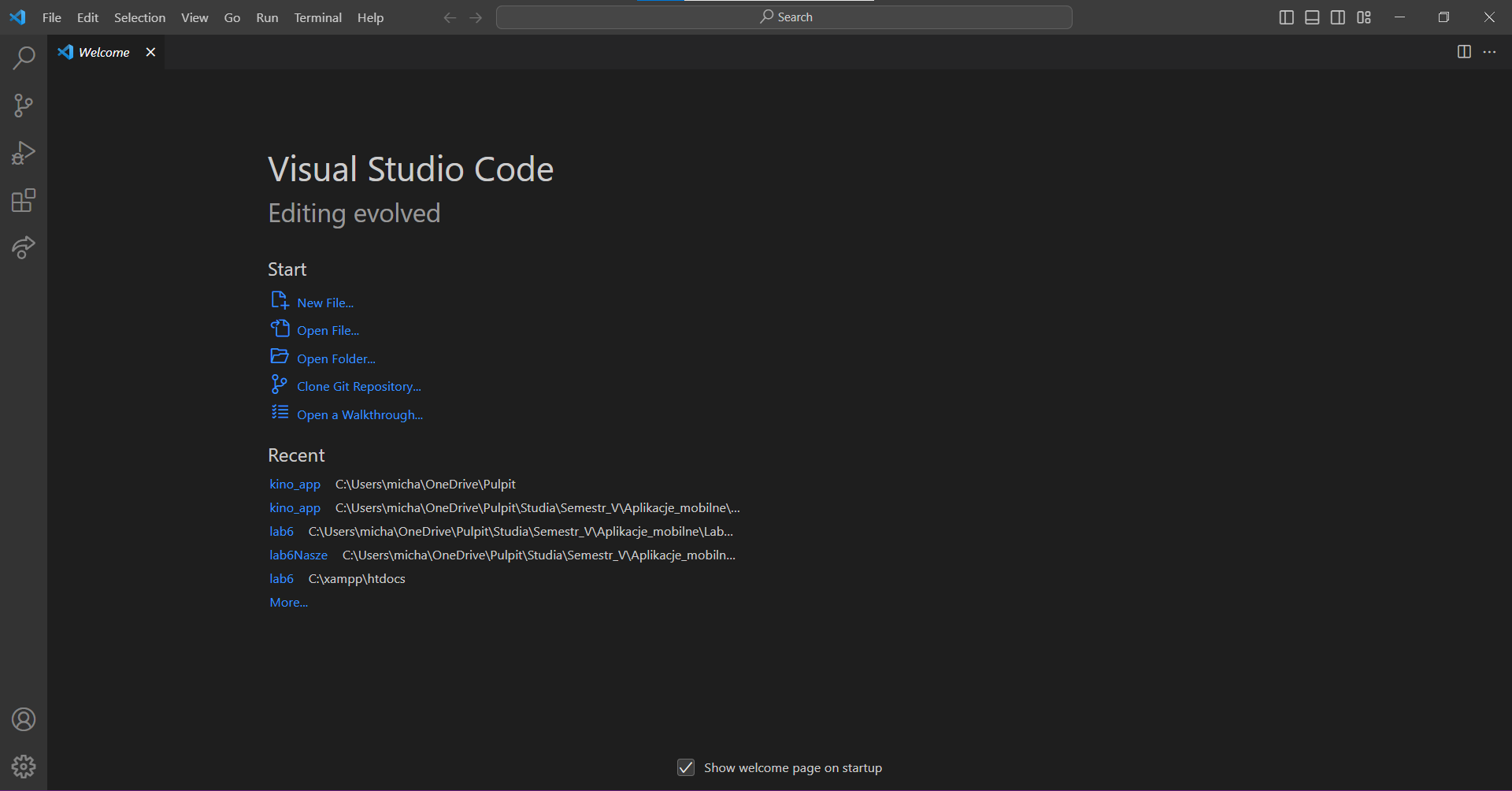


W celu płynniejszego testowania działania aplikacji, można wygenerować plik z rozszerzeniem .apk, a następnie zainstalować go na swoim urządzeniu mobilnym. Na szczęście nie trzeba robić tego manualnie, gdyż narzędzie Flutterflow od razu udostępnia nam opcję pobrania takiego pliku. W ten sposób łatwiej będzie wychwycić błędy. Co więcej, jest to jedyny sposób na sprawdzenie działania wykorzystanych narzędzi (skaner kodów QR oraz GPS), gdyż internetowy emulator nie oferuje takiej możliwości.

## **Visual studio**

Flutterflow umożliwia również integrację z innymi narzędziami programistycznymi, co pozwala na większą elastyczność w pracy nad projektem. Podczas prac planujemy także wykorzystać ogólnodostępne środowisko programistyczne Visual Studio Code, które pozwoli nam na wprowadzanie własnych modyfikacji w kodzie projektu, które byłyby ciężkie do zrealizowania przy użyciu powyższego narzędzia.

Visual Studio Code to popularne, otwarte i darmowe środowisko programistyczne, które umożliwia tworzenie, edycję i debugowanie kodu w wielu językach programowania. W przypadku naszego projektu, zdecydowaliśmy się wykorzystać Visual Studio Code jako dodatkowe narzędzie programistyczne do modyfikacji kodu aplikacji Dzięki wykorzystaniu Visual Studio Code, można wprowadzać bardziej złożone modyfikacje w kodzie projektu, które byłyby trudne do zrealizowania bezpośrednio przy użyciu Flutterflow.



## 

## **Firebase**

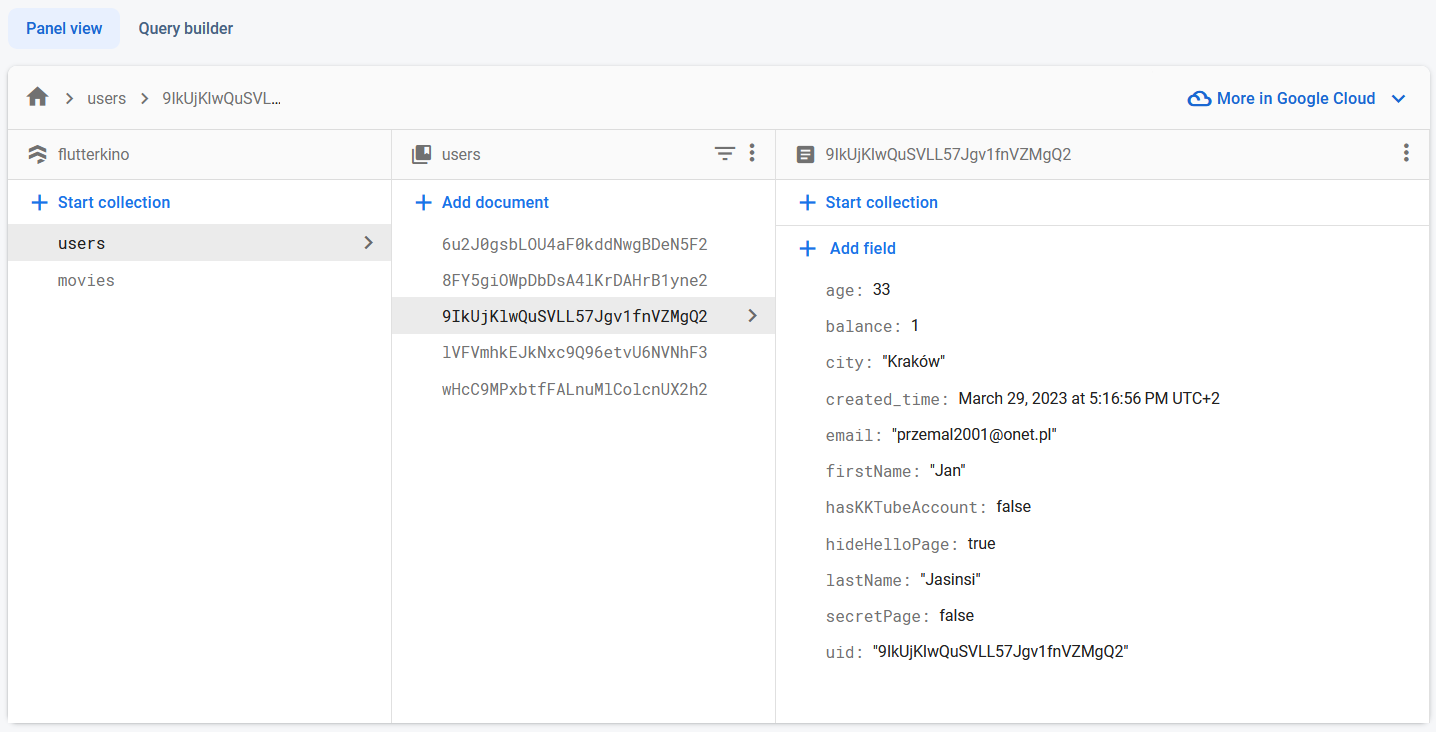
Jako magazyn danych dla naszej aplikacji będziemy wykorzystywać bazę danych Firebase. Posłuży nam ona do przechowywania kluczowych danych, obrazów oraz filmików. Firebase to platforma stworzona przez Google, która umożliwia łatwe przechowywanie, synchronizowanie oraz przetwarzanie danych w czasie rzeczywistym. Oferuje ona szeroki zakres narzędzi do tworzenia aplikacji mobilnych oraz webowych. W jej skład wchodzą m.in. bazy danych, funkcje backendowe, narzędzia do analizy i uwierzytelnianie użytkowników. Korzystanie z Firebase pozwala na łatwe i szybkie przechowywanie oraz synchronizację danych między aplikacją, a bazą danych.

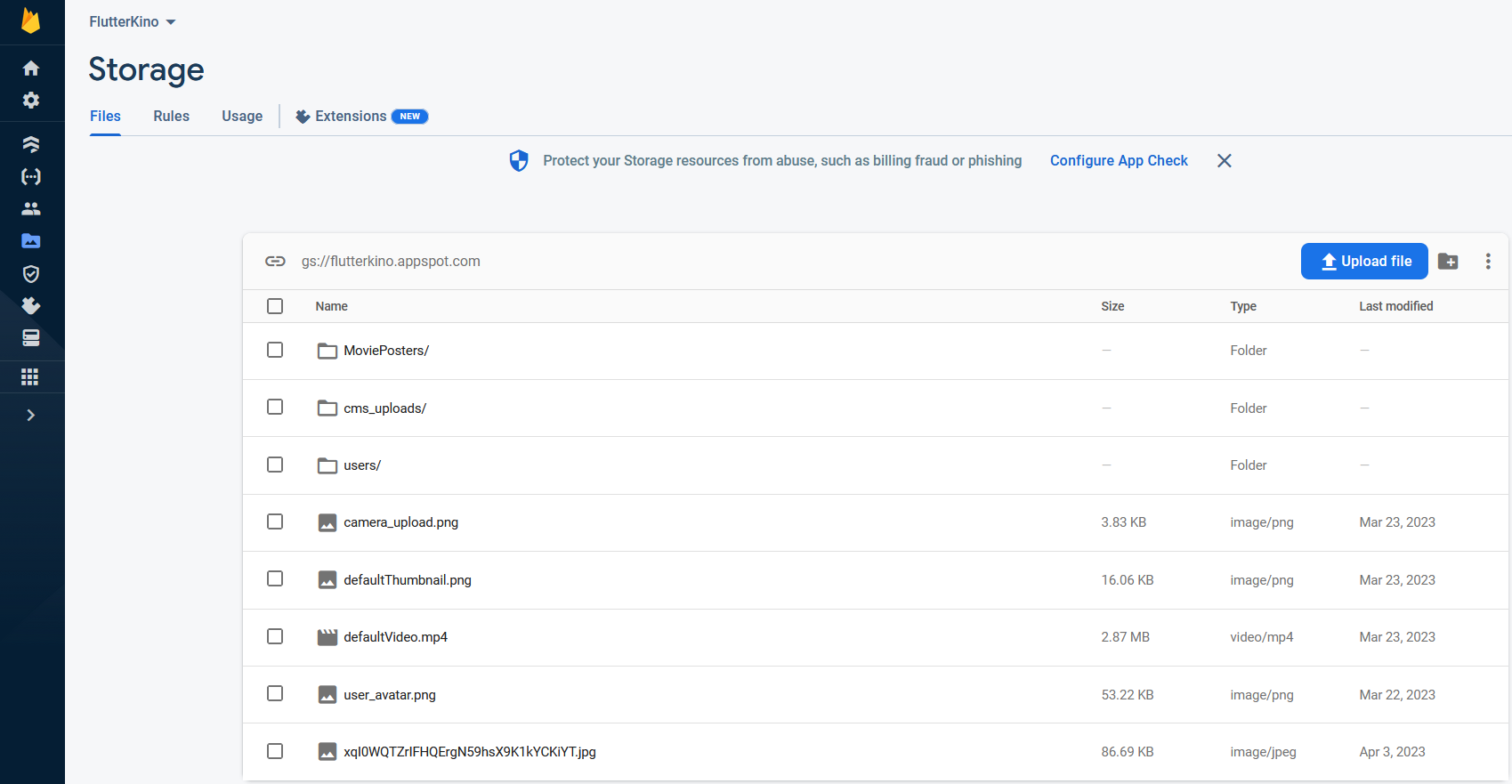
W naszej aplikacji mobilnej wykorzystamy funkcjonalności Firebase, takie jak autoryzacja, której zadaniem będzie umożliwienie użytkownikom utworzenia nowego konta oraz późniejszego zalogowania się do aplikacji w celu dostępu do konkretnych funkcjonalności. Firebase Storage posłuży nam do przechowywania obrazów i filmików, które stanowią nieodłączny element aplikacji kinowo-streamingowej z dodatkiem funkcji Shorts. Firebase Database natomiast pozwoli na przechowywanie i zarządzanie informacjami dotyczącymi użytkowników oraz informacjami o aktywnościach w aplikacji.

Korzystając z Firebase, będziemy mogli w łatwy sposób zarządzać bazą danych, a także zyskać dostęp do wielu przydatnych narzędzi i funkcjonalności, które umożliwią nam tworzenie bardziej skomplikowanych funkcjonalności w naszej aplikacji mobilnej.

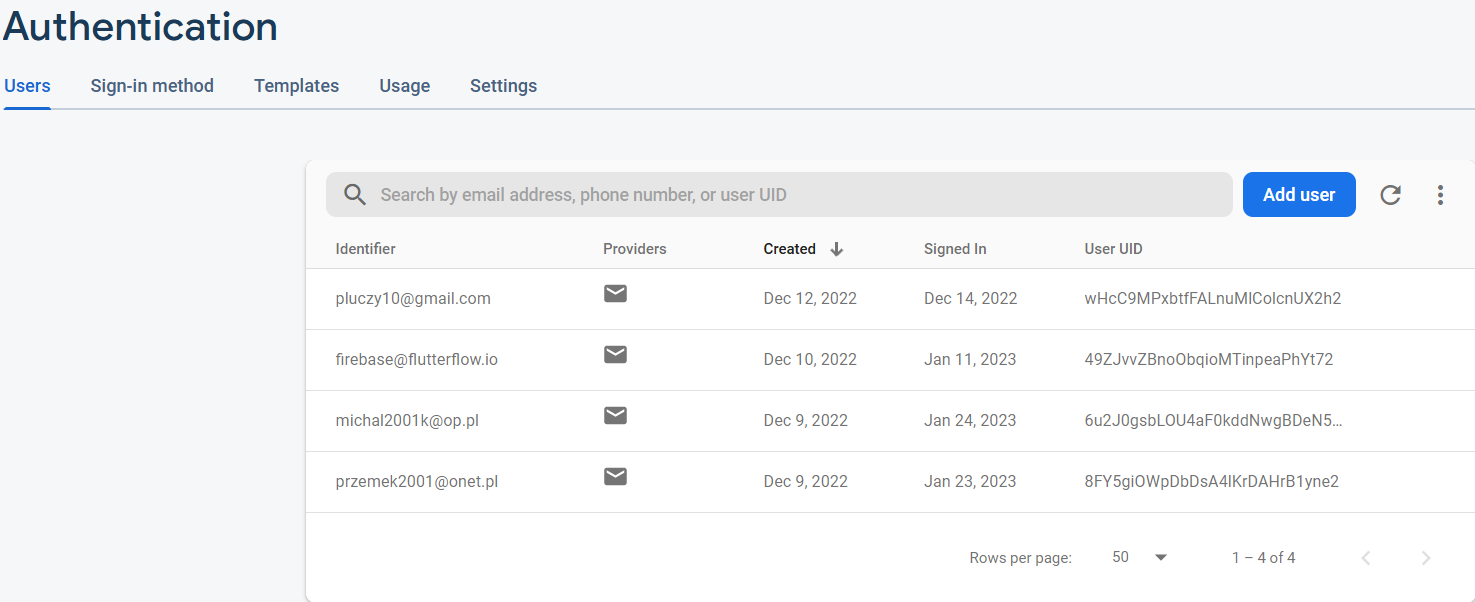


Dane będą przechowywane w kilku zależnych od siebie tabelach. Pojedynczy wiersz danych jednej z takich tabel nazywany jest dokumentem i można go określić jako samodzielny obiekt reprezentujący jakiś zestaw danych, np. informacje o filmie.





Hasła oraz inne dane autoryzacyjne są przechowywane w osobnym miejscu. Firebase oferuje osobną zakładkę o nazwie „Authentication”, w której przechowuje te dane. Wszelakie odwołania do bazy weryfikujące poprawność danych uwierzytelniających (np. podczas logowania) odbywają się w tym miejscu. To właśnie obiekt tej zakładki reprezentuje zarejestrowanego użytkownika, a nie obiekt tabeli users.



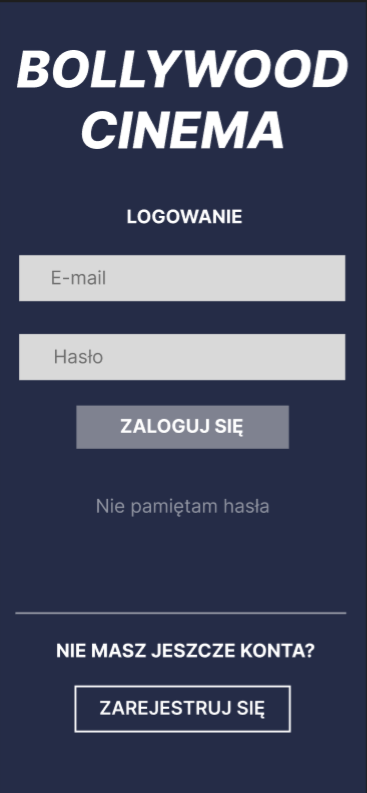
# **Założenia**

Użytkownik będzie posiadać możliwość:

* utworzenia nowego konta oraz późniejszego zalogowania się na nie za pomocą danych podanych podczas rejestracji
* wyświetlenia danych swojego konta oraz ich ewentualnej edycji
* zresetowania hasła przy pomocy skrzynki mailowej w przypadku jego zagubienia
* usunięcia konta jeśli uzna, że nie chce już dalej korzystać z usług naszej aplikacji
* skorzystania z narzędzia GPS urządzenia w celu automatycznego ustalenia miasta zamieszkania użytkownika
* sprawdzenia aktualnie odtwarzanych w kinie filmów oraz zastosowanie kalendarza do filtrowania wyników
* wyszukania interesującej go produkcji w bazie wszystkich oferowanych przez aplikację filmów opierając swoje wyszukiwanie na tytule lub gatunku
* sprawdzenia kluczowych informacji o wybranym filmie oraz obejrzenie jego traileru
* kupienia biletu na wybrany z repertuaru film (przedtem wybór miejsca na sali)
* wykupienia dostępu do filmu w formie streamingu
* sprawdzenia podsumowania przeprowadzanej transakcji
* wyświetlenia historii operacji na swoim profilu aplikacji
* sprawdzenia wykupionych przez siebie biletów oraz streamingów
* obejrzenia każdego z wcześniej wykupionych streamingów
* doładowania środków do portfela swojego profilu przy pomocy kodu BLIK lub kodu rabatowego
* sprawdzenia salda swojego konta
* wykorzystanie czytnika kodów QR do wczytania rabatu
* wstawienia krótkiego filmiku, który będą mogli oglądać inny użytkownicy
* nagrania krótkiego filmiku
* oglądania krótkich materiałów przesłanych przez innych użytkowników
* przeczytania instrukcji obsługi aplikacji aby móc korzystać z pełnej funkcjonalności

# **Ekrany**

* Ekran służący do logowania się przez wcześniej zarejestrowanego użytkownika. Pozwala przejść do ekranu rejestracji oraz przypominania hasła. W przypadku podania prawidłowych danych, użytkownik zostaje przeniesiony do głównego ekranu swojego profilu.



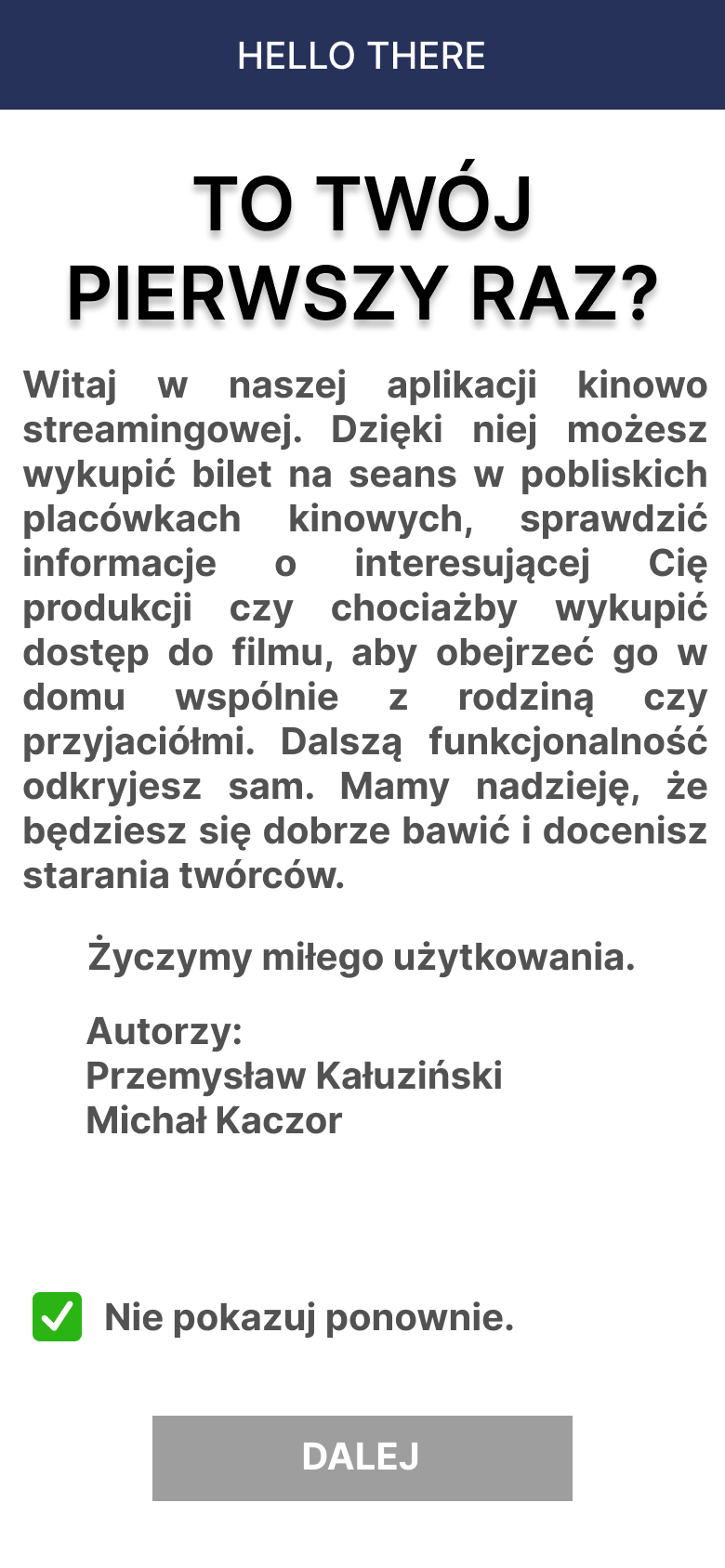
* Ekran służący do utworzenia nowego konta przez użytkownika. Za pomocą podanych tutaj danych dostępowych przeprowadzana będzie później operacja logowania. Po udanej rejestracji, użytkownik będzie od razu logowany do aplikacji i przenoszony na ekran powitalny nowo utworzonego konta.



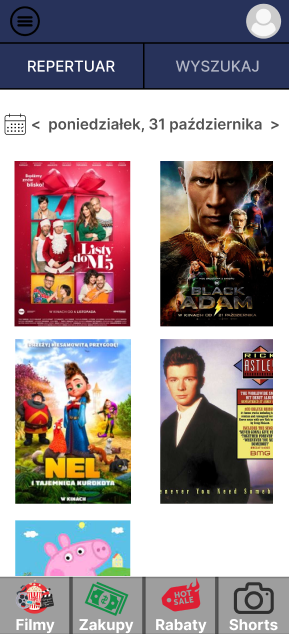
* Ekran służący do resetowania zapomnianego hasła użytkownika. Po wprowadzeniu istniejącego w bazie adresu mailowego.



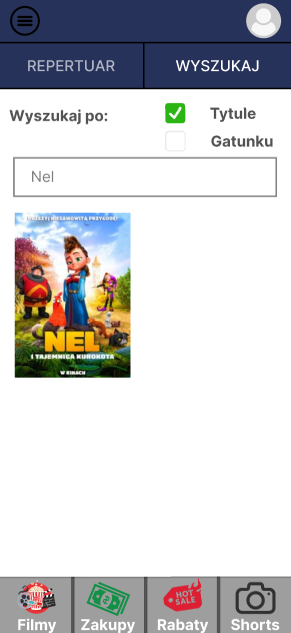
* Ekran powitalny witający każdego nowo zarejestrowanego użytkownika.



* Ekran główny aplikacji zawierający podstronę repertuaru. Wyświetlać będzie aktualnie grane filmy w kinie znajdującym się w mieście wybranym podczas rejestracji.



* Ekran główny aplikacji zawierający podstronę streamingu. Wyświetla wszystkie filmy dostępne w opcji wykupienia dostępu, w celu obejrzenia ich w domu.



* Ekran służący do wyświetlania wszystkich ważnych informacji o filmie wybranym z listy repertuaru. Zawiera player wyświetlający trailer danej produkcji. Po kliknięciu przycisku „Kup bilet” przenosimy się do ekranu wyboru godzin dla seansu.



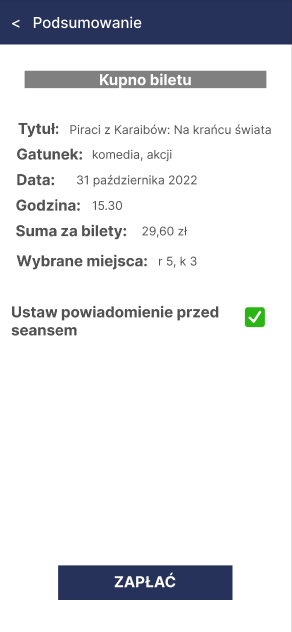
* Ekran służący do wyboru odpowiadającej nam godziny wybranego seansu.



* Ekran służący do wyboru interesujących nas miejsc siedzących na sali kinowej wybranego seansu. Miejsca zajęte, wolne oraz wybrane będą oznaczone różnymi kolorami.



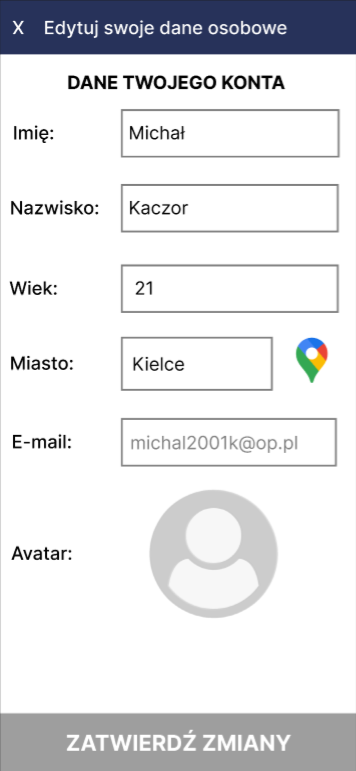
* Ekran służący do wyświetlania informacji na temat zawieranej transakcji zakupu biletu na seans kinowy.



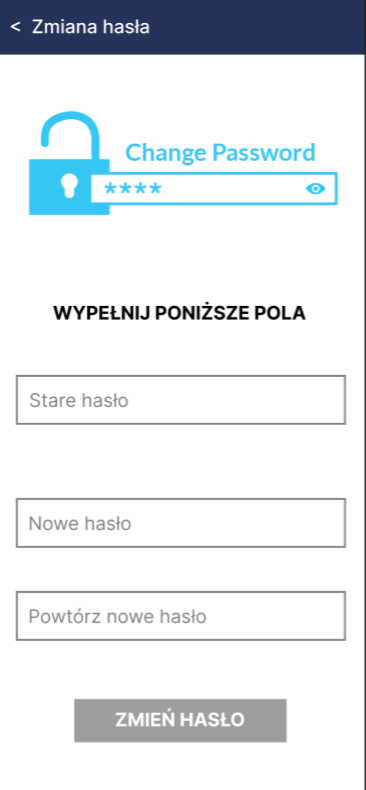
* Ekran służący do usuwania konta użytkownika w przypadku, gdy decydujemy się, że nie będziemy już chcieli korzystać z aplikacji.



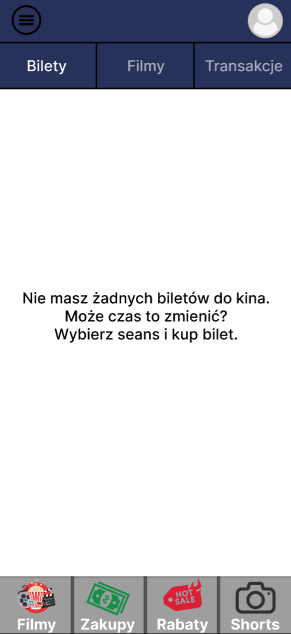
* Ekran służący do wyświetlania oraz modyfikacji danych osobowych klienta.



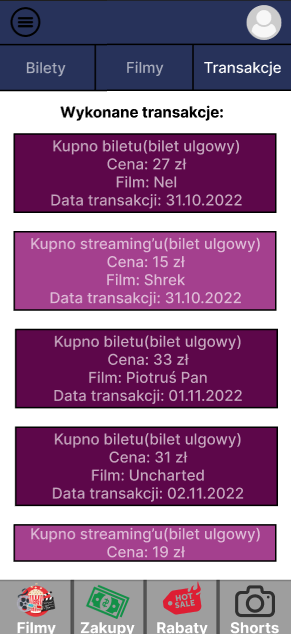
* Ekran służący do zmiany hasła aktualnie zalogowanego użytkownika.



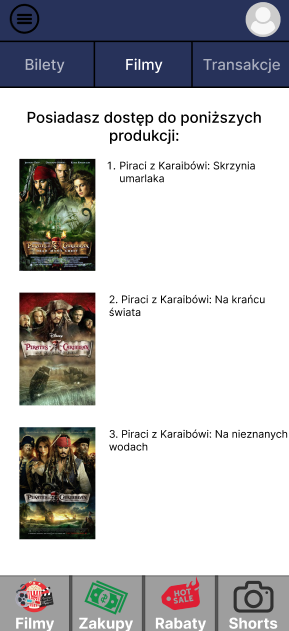
* Ekran służący do wyświetlania zakupionych biletów. Lista wyświetla tylko najważniejsze informacje na temat biletu, a jeśli chcemy sprawdzić szczegóły to wystarczy, że klikniemy na interesujący nas wiersz.



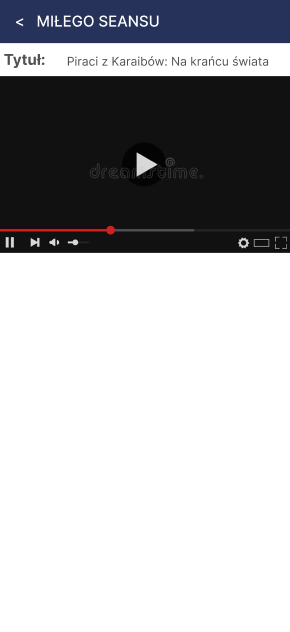
* Ekran służący do wyświetlania wszystkich transakcji zakupowych przeprowadzonych na koncie użytkownika. Każdy wiersz zawiera kluczowe informacje o danej transakcji.



* Ekran służący do wyświetlania wszystkich wykupionych filmów w postaci streamingów. Jeśli chcemy obejrzeć daną produkcję, to wystarczy, że klikniemy w interesujący nas wiersz.



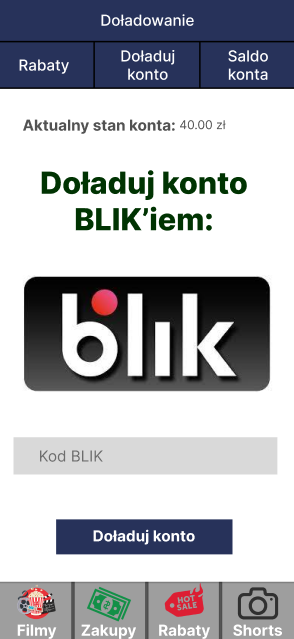
* Ekran służący do oglądania wykupionego streamingu. Zawiera player multimedialny odtwarzający wybraną z listy dostępnych pozycji produkcję.



* Ekran służący do doładowania salda aplikacji poprzez wykorzystanie kodu rabatowego. Na tym ekranie będzie istnieć opcja skanowania kodów QR. Dzięki niej użytkownik może zeskanować wybrany kod, aby ułatwić sobie ten proces.



* Ekran służący do doładowania salda aplikacji poprzez wykorzystanie kodu BLIK. Użytkownik podaje wysokość doładowania, a następnie wpisuje 6 cyfrowy kod bankowy. Jeśli wszystko się powiedzie, to środki zostaną dodane do konta użytkownika.



* Ekran służący do wyświetlania informacji na temat aktualnej wysokości salda konta użytkownika. To właśnie za pomocą tych środków klient ma możliwość zakupu biletów na seanse kinowe lub wykupienia dostępu do filmu w postaci streamingu.



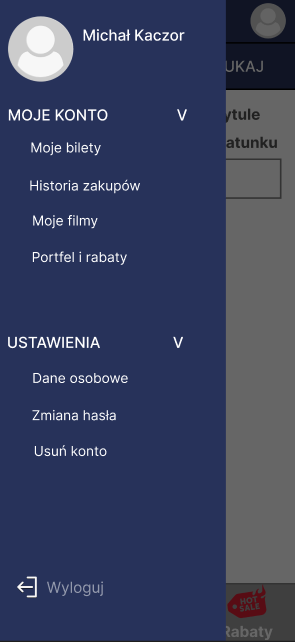
* Ekran dający klientowi możliwość zakupu plakatu wraz z biletem na seans. Plakat byłby dodatkowo płatny i miał wpływ na ostateczną cenę.



* Ekran wyświetlający wszystkie szczegółowe informacje na temat danego biletu oraz seansu. Jest rozszerzeniem ekranu wyświetlającego bilety użytkownika, w celu zmniejszenia ilości wyświetlanych informacji w wierszu listy.



* Trzy rodzaje nawigacji. Stackowa, BottomTab i DrawerMenu. DrawerMenu, zawiera odnośniki do większości funkcjonalności aplikacji, a BottomMenu, służy do przemieszczania się po głównych ekranach.



* Ekran działający jak filmiki typu Shorts na platformie Youtube pozwalający przesuwać losowe filmiki innych użytkowników oraz dający użytkownikom możliwość dawania łapek w górę lub w dół. Ewentualnie dodawanie komentarzy.



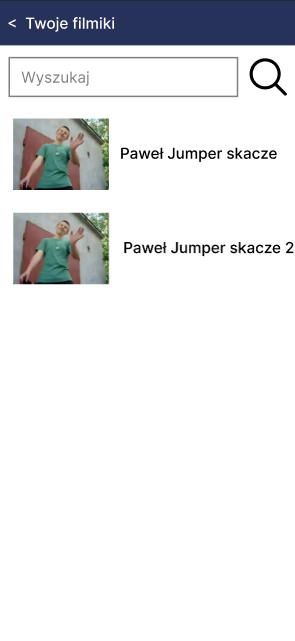
* Ekran zawierający wszystkie kluczowe informacje oraz statystyki konta KK-Tube zalogowanego użytkownika. Pozwala na edycję nicku oraz wyświetla różne statystyki w formie tekstowej oraz graficznej.



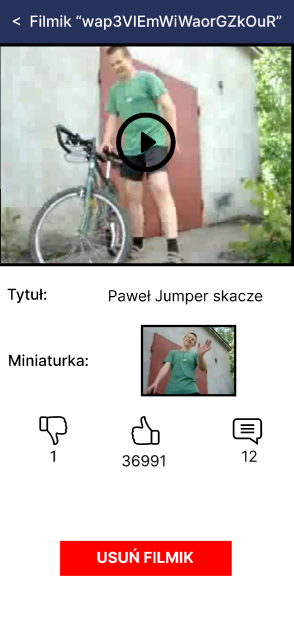
* Ekran służący do uploadu krótkich filmików z pamięci urządzenia lub nagrania ich w czasie rzeczywistym. Możliwość wyboru nazwy dodawanego filmiku.



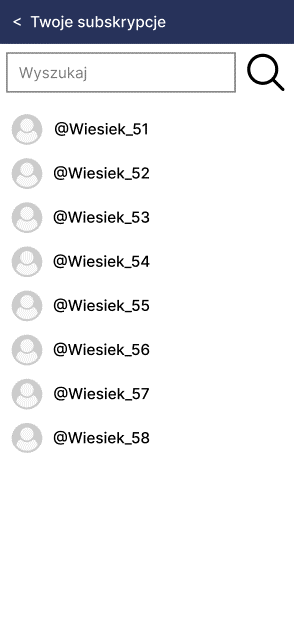
* Ekran wyświetlający listę wrzuconych przez nas filmików. Zawiera wyszukiwarkę w celu filtrowania listy. Po kliknięciu w wybranego filmiku zostaniemy przeniesieni na stronę zawierającą szczegółowe informacje na jego temat.



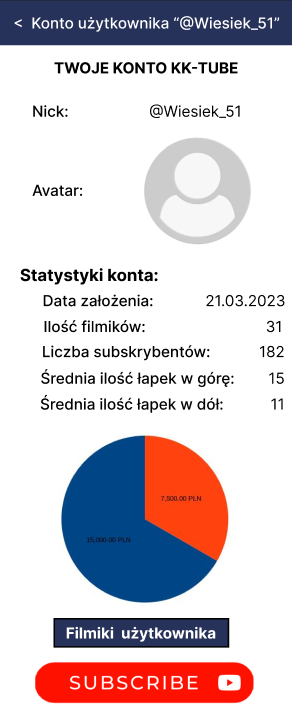
* Ekran wyświetlający wszystkie dodane przez danego użytkownika filmy i udostępniający możliwość usunięcia ich z platformy. Możemy sprawdzić ilość lików dla danego filmu.



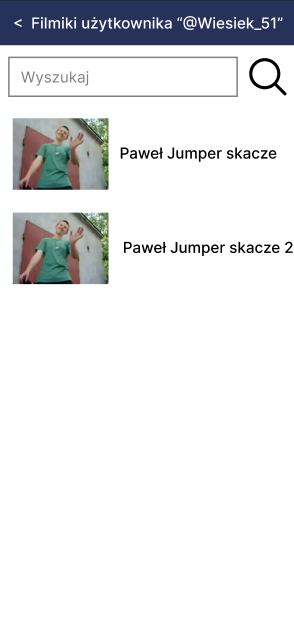
* Ekran wyświetlający listę subskrybowanych przez zalogowanego użytkownika kont innych użytkowników KK-Tube. Zawiera wyszukiwarkę w celu filtrowania listy. Po kliknięciu w wybranego użytkownika zostaniemy przeniesieni na stronę jego profilu.



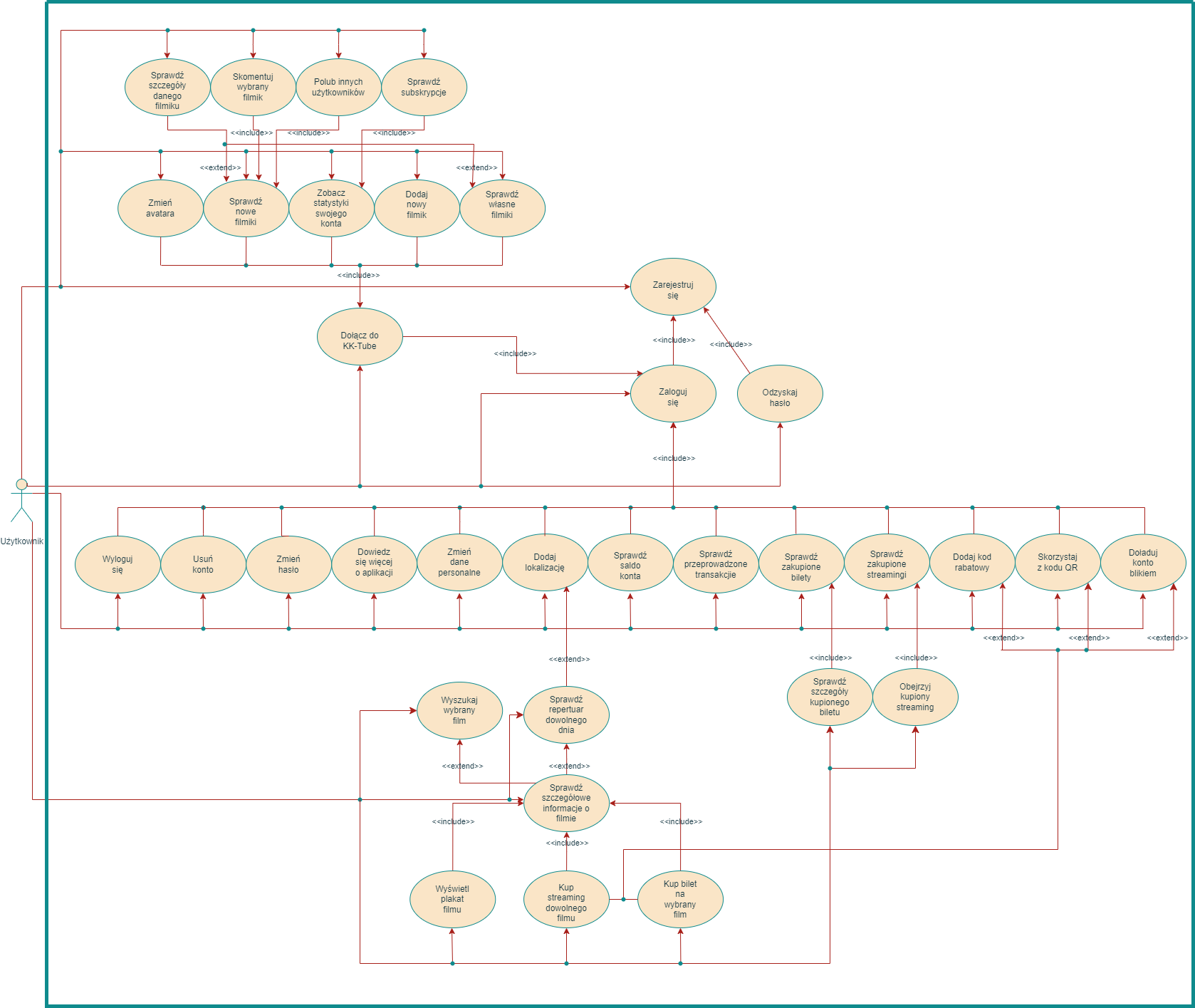
* Ekran konta innego użytkownika KK-Tube. Zawiera dokładnie te same informacje, co na naszym profilu, lecz nie mamy tutaj takich możliwości jak zmiana danych czy wyświetlenie niektórych danych. W tym przypadku na dole znajduje się przycisk subskrypcji, pozwalający nam zasubskrybować konto danego użytkownika.



* Ekran zawierający listę wrzuconych przez wybranego użytkownika filmików. Zawiera wyszukiwarkę w celu filtrowania listy.



# **Diagram UML**



# **Scenariusze przypadków użycia**

**Nazwa:** Rejestracja

**Warunki początkowe:**

Klient musi mieć dostęp do aplikacji.

**Warunki końcowe:**

Klient nie może utworzyć drugiego konta o tym samym mailu.

**Aktorzy:** Klient

**Przepływ zdarzeń:**

1. System przenosi klienta do ekranu logowania.
2. Klient nie może się zalogować bez posiadania konta.
3. Klient wybiera opcje rejestracji.
4. System przenosi użytkownika do panelu rejestracji.
5. Klient podaje dane do rejestracji konta.
6. System przetwarza dane użytkownika.
7. System po udanej rejestracji przenosi użytkownika do ekranu powitalnego.
8. Klient wybiera opcje przejścia dalej.
9. System przenosi użytkownika do głównego ekranu aplikacji.

**Alternatywny przepływ:**

* 6. Klient nie wypełnił wymaganych pól. POWRÓT DO 5

**Nazwa:** Kupno biletu

**Warunki początkowe:**

Klient musi być zalogowany w aplikacji.

Musi istnieć przynajmniej jeden film, na który można kupić bilet.

**Warunki końcowe:**

Klient posiada w aplikacji zakupiony bilet.

Miejsca siedzące na dany film zostały zarezerwowane.

**Aktorzy:** Klient

**Przepływ zdarzeń:**

1. Klient wybiera ekran „Filmy”, a następnie zakładkę „Repertuar”.
2. System przenosi użytkownika do ekranu repertuaru.
3. Klient wybiera film.
4. System sprawdza czy film już nie był grany.
5. Klient wybiera zakup biletu.
6. System przenosi użytkownika do szczegółów seansu.
7. Klient wybiera dogodną godzinę seansu i wybiera opcje przejścia dalej.
8. System przenosi użytkownika do ekranu wyboru plakatu.
9. Klient wybiera czy chce otrzymać plakat.
10. System przenosi użytkownika do ekranu wyboru siedzeń.
11. Klient wybiera miejsca siedzące.
12. System przenosi użytkownika do ekranu podsumowania.
13. Klient wybiera opcje płatności za wybrany film oraz miejsca siedzące.
14. System sprawdza dostępność wybranych miejsc.
15. System pobiera daną kwotę z konta użytkownika i rezerwuje wybrane miejsca.
16. System przypisuje klientowi zakupiony bilet.
17. System prznosi użytkownika do ekranu z biletami.

**Alternatywny przepływ:**

* 3. Dany film był już grany. POWRÓT DO 2
* 13. Wybrane miejsca nie są już dostępne. POWRÓT DO 11
* 14. Klient nie posiada wystarczających środków na koncie. POWRÓT DO 11

**Nazwa:** Kupno streamingu

**Warunki początkowe:**

Klient musi być zalogowany w aplikacji.

Musi istnieć przynajmniej jeden film do kupna streamingu.

**Warunki końcowe:**

Klient posiada w aplikacji zakupiony streaming.

Klient może obejrzeć film w aplikacji.

**Aktorzy:** Klient

**Przepływ zdarzeń:**

1. Klient wybiera ekran „Filmy”, a następnie zakładkę „Wyszukaj”.
2. System przenosi użytkownika do ekranu z listą filmów.
3. Klient wyszukuje film.
4. Klient wybiera film.
5. System przenosi użytkownika do ekranu z informacjami o filmie.
6. Klient wybiera opcje zakupu filmu.
7. System przenosi użytkownika do ekranu podsumowania.
8. Klient wybiera opcje płatności.
9. System pobiera środki z konta użytkownika.
10. System przenosi użytkownika do ekranu z filmami.

**Alternatywny przepływ:**

* 9. Klient nie posiada wystarczającej liczby środków na koncie. POWRÓT DO 7

**Nazwa:** Dodanie filmu na KK-Tube

**Warunki początkowe:**

Klient musi mieć dostęp do aplikacji.

Klient musi posiadać konto na KK-Tube.

**Warunki końcowe:**

Klient posiada dodane filmy w części aplikacji KK-Tube.

**Aktorzy:** Klient

**Przepływ zdarzeń:**

1. Klient wybiera ekran Shorts.
2. System przenosi użytkownika do ekranu początkowego KK-Tube.
3. Klient akceptuje utworzenie konta i podanie nicku.
4. System przenosi użytkownika do głównego ekranu KK-Tube.
5. Klient wybiera ikone użytkownika.
6. System przenosi użytkownika do zakładki „Twpje konto”.
7. Klient wybiera opcje dodania filmiku.
8. System przenosi użytkownika do zakładki dodawania filmików.
9. Klient wybiera lub nagrywa filmik, podaje tytuł, dodaje miniaturkę i wybiera opcje dodania filmiku.
10. System dodaje filmik do bazy danych.
11. System przenosi użytkownika do głównej strony.

**Alternatywny przepływ:**

* 3. Użytkownik nie podał nicku. POWRÓT DO 2
* 10. Użytkownik nie dodał, jednej z wymaganych pozycji. POWRÓT DO 8