Министерство образования и молодежной политики Свердловской области



ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Отчёт по УП

Задание 3.2. Приложение для телевизора «Кинотека»

Выполнил: Никитин Семен Антонович

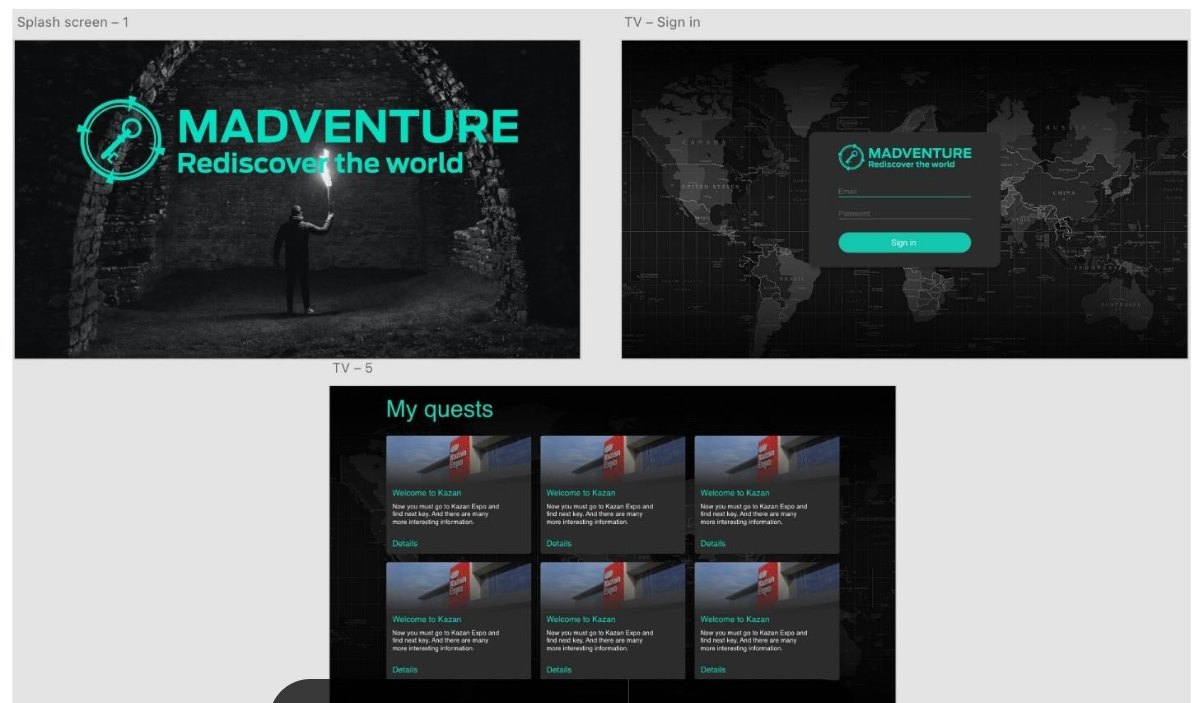
Группа: ПР-31

Преподаватель: Мирошниченко Г.В

2024

1. **Описание задачи:**

****

****

**Входные данные:** login - поле для ввода логина пользователя, password - поле для ввода пароля пользователя

**Выходные данные:** gridLayout – контейнер для вывода списка фильмов

1. **Структура проекта:**

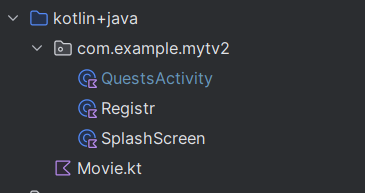
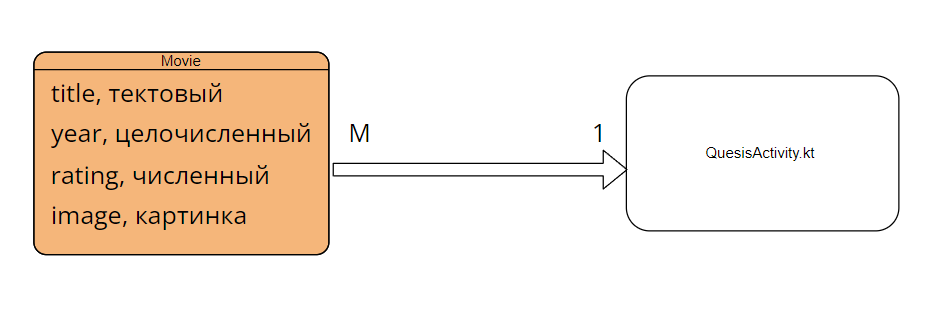


Схема классов



ER - диаграмма

1. **Листинг программы:**

**activity\_splash\_screen.xml**

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="@drawable/firstfon"  
 tools:context=".SplashScreen">  
  
  
 <ImageView  
 android:id="@+id/imageView"  
 android:layout\_width="200dp"  
 android:layout\_height="174dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.498"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.106"  
 tools:src="@drawable/logo" />  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="70dp"  
 android:text="@string/biglogo"  
 android:textColor="@color/green"  
 android:textSize="45dp"  
 android:textStyle="bold"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.47" />  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="70dp"  
 android:text="@string/minilogo"  
 android:textColor="@color/green"  
 android:textSize="20dp"  
 android:textStyle="bold"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.595" />  
</androidx.constraintlayout.widget.ConstraintLayout>

**activity\_registr.xml**

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="@drawable/firstfon"  
 tools:context=".Registr">  
  
 <EditText  
 android:id="@+id/login"  
 android:layout\_width="340dp"  
 android:layout\_height="50dp"  
 android:backgroundTint="@color/gray"  
 android:hint="Login"  
 android:textColor="@color/white"  
 android:textColorHint="@color/gray"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.42">  
  
 </EditText>  
  
 <EditText  
 android:id="@+id/password"  
 android:layout\_width="340dp"  
 android:layout\_height="50dp"  
 android:backgroundTint="@color/gray"  
 android:hint="Password"  
 android:inputType="textPassword"  
 android:textColor="@color/white"  
 android:textColorHint="@color/gray"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.542">  
  
 </EditText>  
  
 <Button  
 android:layout\_width="200dp"  
 android:layout\_height="50dp"  
 android:background="@drawable/button"  
 android:onClick="onQui"  
 android:text="Sign In"  
 android:textColor="@color/white"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.502"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.761">  
  
 </Button>  
  
 <TextView  
 android:layout\_width="269dp"  
 android:layout\_height="49dp"  
 android:text="@string/biglogo"  
 android:textColor="@color/green"  
 android:textSize="33dp"  
 android:textStyle="bold"  
 android:textAlignment="center"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.215" />  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="70dp"  
 android:text="@string/minilogo"  
 android:textColor="@color/green"  
 android:textSize="20dp"  
 android:textStyle="bold"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.329" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

**activity\_quests.xml**

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="@drawable/secondfon"  
 tools:context=".QuestsActivity">  
  
  
 <TextView  
 android:layout\_width="269dp"  
 android:layout\_height="49dp"  
 android:text="My films"  
 android:textColor="@color/green"  
 android:textSize="40dp"  
 android:textStyle="bold"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.057"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.032" />  
  
 <ScrollView  
 android:layout\_width="match\_parent"  
 android:layout\_height="450dp"  
 android:fillViewport="true"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.0"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="1.0">  
  
 <GridLayout  
 android:id="@+id/gridLayout"  
 android:layout\_width="930dp"  
 android:layout\_height="wrap\_content"  
 android:columnCount="10"  
 android:rowCount="10"  
 tools:ignore="MissingConstraints" />  
  
 </ScrollView>  
  
  
</androidx.constraintlayout.widget.ConstraintLayout>

**Movie.kt**

import com.google.gson.annotations.SerializedName  
  
data class Movie(  
 @SerializedName("name") val title: String,  
 @SerializedName("year") val year: Int,  
 @SerializedName("internalRating") val rating: Double,  
 @SerializedName("poster") val poster: Poster  
)  
  
data class Poster(  
 @SerializedName("url") val image: String?  
)

**SplashScreen.kt**

class SplashScreen : AppCompatActivity() {  
  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_splash\_screen*)  
  
 val timer = object : CountDownTimer(3008, 1000) {  
 override fun onTick(millisUntilFinished: Long) {  
 }  
  
 override fun onFinish() {  
 val intent = Intent(this@SplashScreen, Registr::class.*java*)  
 startActivity(intent)  
 finish()  
 }  
 }  
 timer.start()  
 }  
}

**Registr.kt**

class Registr : AppCompatActivity() {  
  
 lateinit var login: EditText  
 lateinit var password: EditText  
  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_registr*)  
  
 login = findViewById(R.id.*login*)  
 password = findViewById(R.id.*password*)  
 }  
 fun onQui(view: View) {  
 if (login.*text*.toString().*isNotEmpty*() && password.*text*.toString().*isNotEmpty*()) {  
 if(login.*text*.toString() == "ects" && password.*text*.toString() == "ects2024") {  
 val intent = Intent(this, QuestsActivity::class.*java*)  
 startActivity(intent)  
 }  
 else {  
 val alert = AlertDialog.Builder(this)  
 .setTitle("Ошибка")  
 .setMessage("Неверный логин или пароль!")  
 .setPositiveButton("ОК") **{** dialog, \_ **->** dialog.dismiss() **}** .create()  
 alert.show()  
 }  
 } else {  
 val alert = AlertDialog.Builder(this)  
 .setTitle("Ошибка")  
 .setMessage("Заполните оба поля!")  
 .setPositiveButton("ОК") **{** dialog, \_ **->** dialog.dismiss() **}** .create()  
 alert.show()  
 }  
 }  
}

**QuestsActivity.kt**

}  
 }  
 **}** }  
 }  
 })  
 **}** }  
  
 private fun parseMovieJson(jsonString: String?): Movie? {  
 if (jsonString.*isNullOrEmpty*()) {  
 return null  
 }  
 val gson = Gson()  
 val movieResponse = gson.fromJson(jsonString, MovieResponse::class.*java*)  
 return movieResponse.docs.*firstOrNull*()  
 }  
  
 data class MovieResponse(  
 @SerializedName("docs") val docs: List<Movie>  
 )  
}

1. **Описание разработанных методов:**

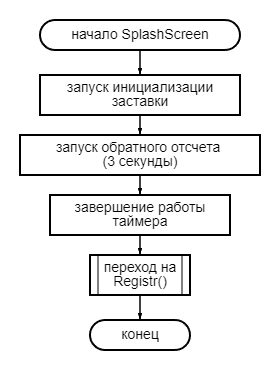
**Методы Registr.kt:**

* onQui() – метод проверки данных, которые ввёл пользователь, и перехода на следующий экран

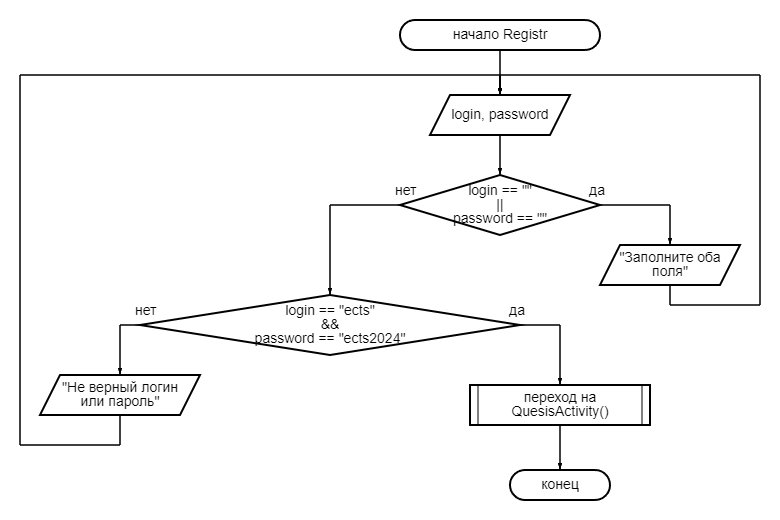
**Методы QuesisActivity.kt:**

* parseMovieJson(jsonString: String?): Movie? – функция принимает на вход строку JSON и пытается преобразовать её в объект Movie. Служит для создания экземпляра JSON и преобразования строки JSON в объект MovieResponse.

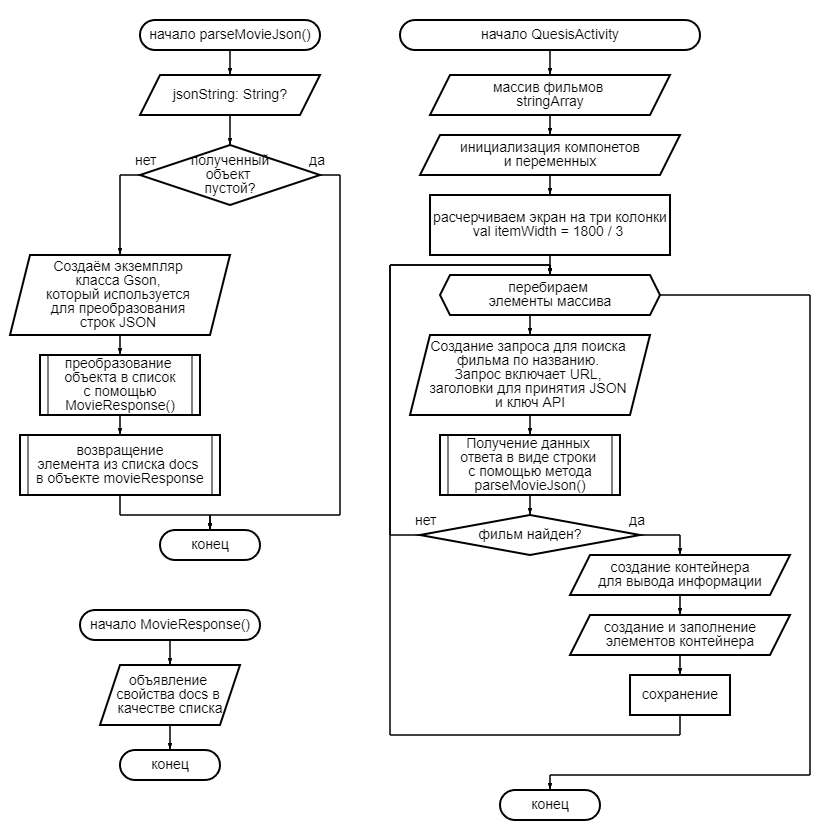
1. **Алгоритм решения:**



Алгоритм SplashScreen.kt



Алгоритм Registr..kt



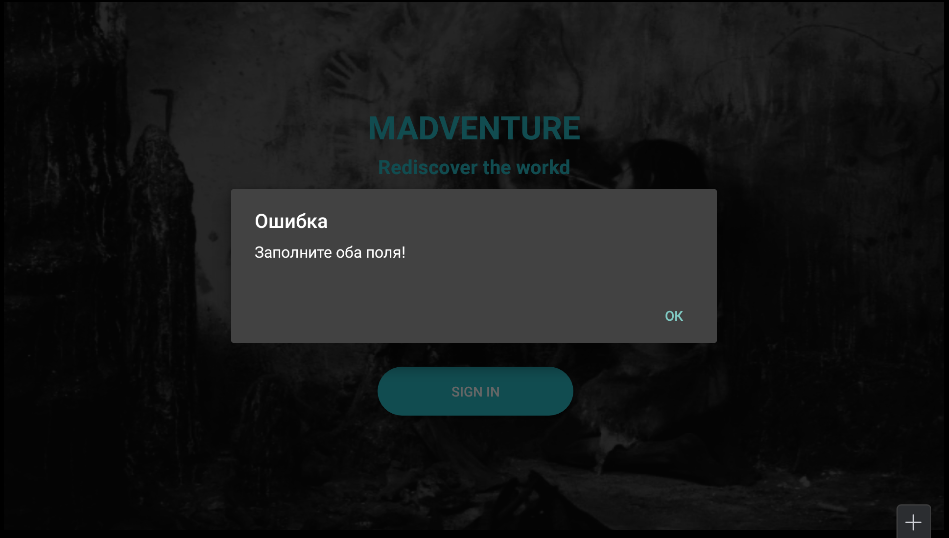
Алгоритм QuesisActivity.kt

1. **Используемые библиотеки:**

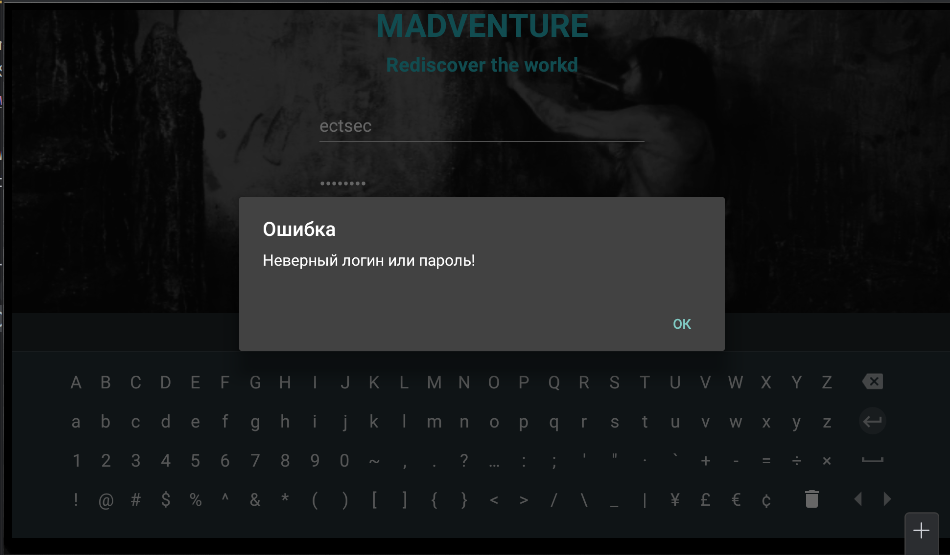
|  |  |
| --- | --- |
| android.annotation.SuppressLint | Эта аннотация используется для подавления предупреждений компилятора о потенциальных проблемах в коде |
| android.graphics.Color | Этот класс предоставляет константы и методы для работы с цветами |
| androidx.appcompat.app.AppCompatActivity | AppCompatActivity — это базовый класс для активностей, который обеспечивает поддержку старых версий Android и позволяет использовать компоненты Material Design |
| android.os.Bundle | Класс Bundle используется для передачи данных между активностями и сохранения состояния активности |
| android.widget.GridLayout | GridLayout — это контейнер, который позволяет размещать дочерние элементы в виде сетки |
| android.widget.ImageView | Этот класс представляет элемент пользовательского интерфейса, который отображает изображения. |
| android.widget.LinearLayout | LinearLayout — это контейнер, который размещает дочерние элементы в одном направлении (горизонтально или вертикально) |
| android.widget.TextView | Этот класс представляет собой элемент пользовательского интерфейса для отображения текста |
| com.google.gson.Gson | Gson — это библиотека для работы с JSON в Java и Kotlin. Она позволяет легко сериализовать и десериализовать объекты, преобразуя их в JSON и обратно |
| com.squareup.picasso.Picasso | Picasso — это библиотека для загрузки и кэширования изображений для Android. Она упрощает процесс загрузки изображений из интернета в ImageView |
| okhttp3 | OkHttp — это мощный HTTP-клиент для приложений Android и Java. Он используется для отправки и получения сетевых запросов на основе HTTP |

1. **Тестовые ситуации:**

Не заполнены оба поля или только одно (любое):



Не верный логин или пароль или и логин и пароль:



1. **Используемые инструменты:**

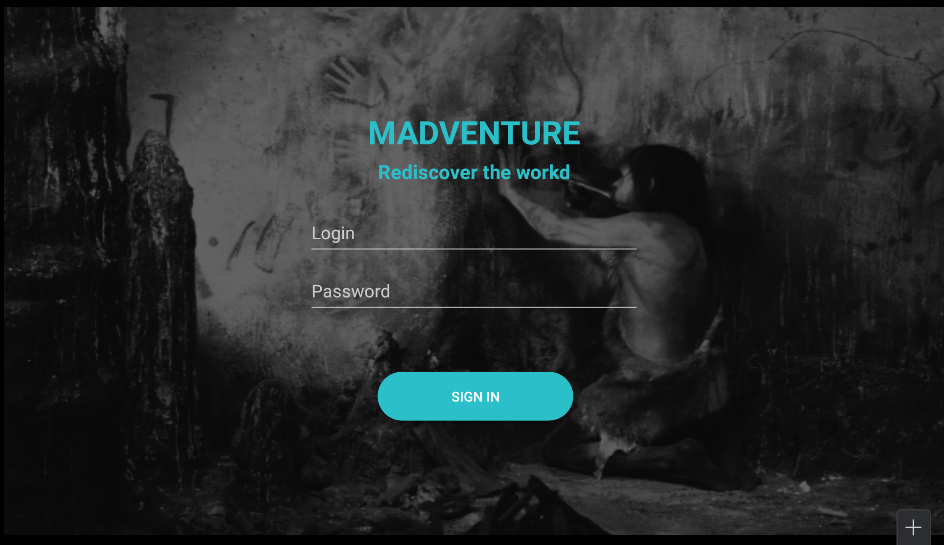
Android Stusio, Kotlin

1. **Описание пользовательского интерфейса:**

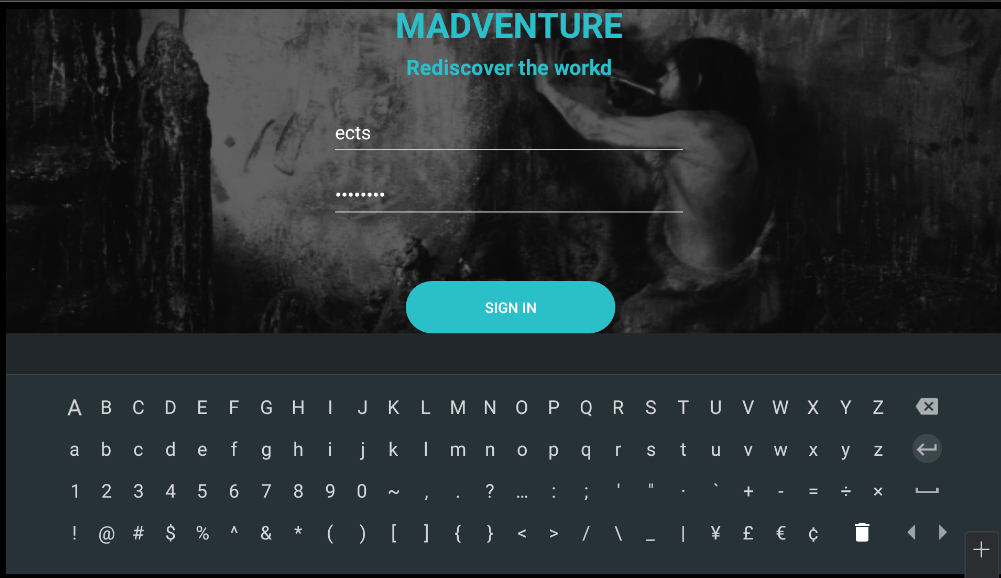
После входа в приложение, пользователя встречает экран загрузки приложения, который будет показан на 3 секунды:



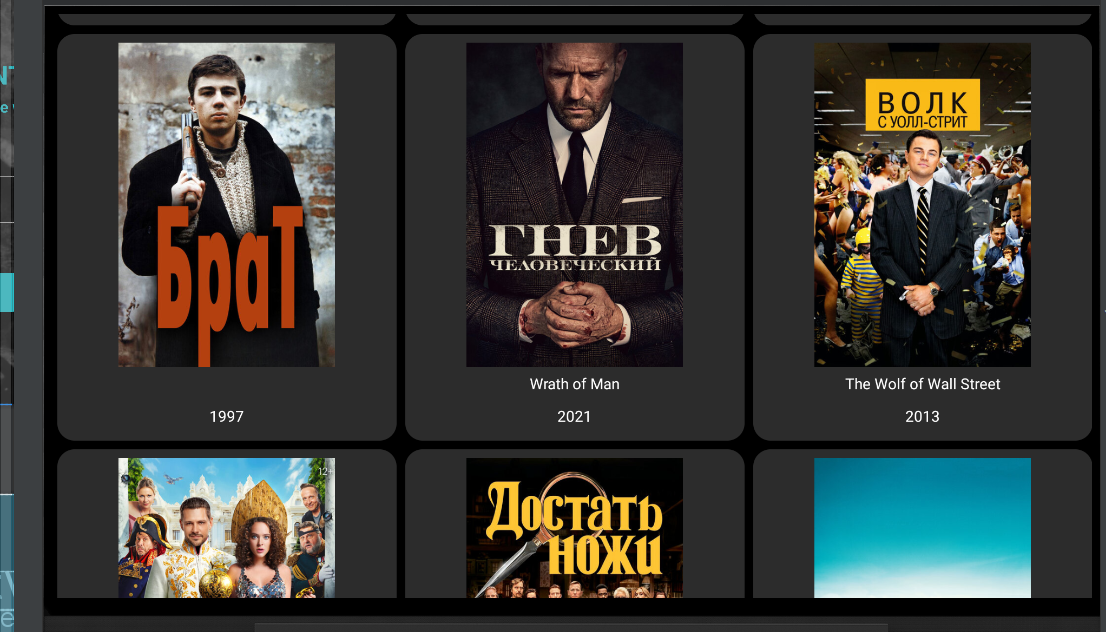
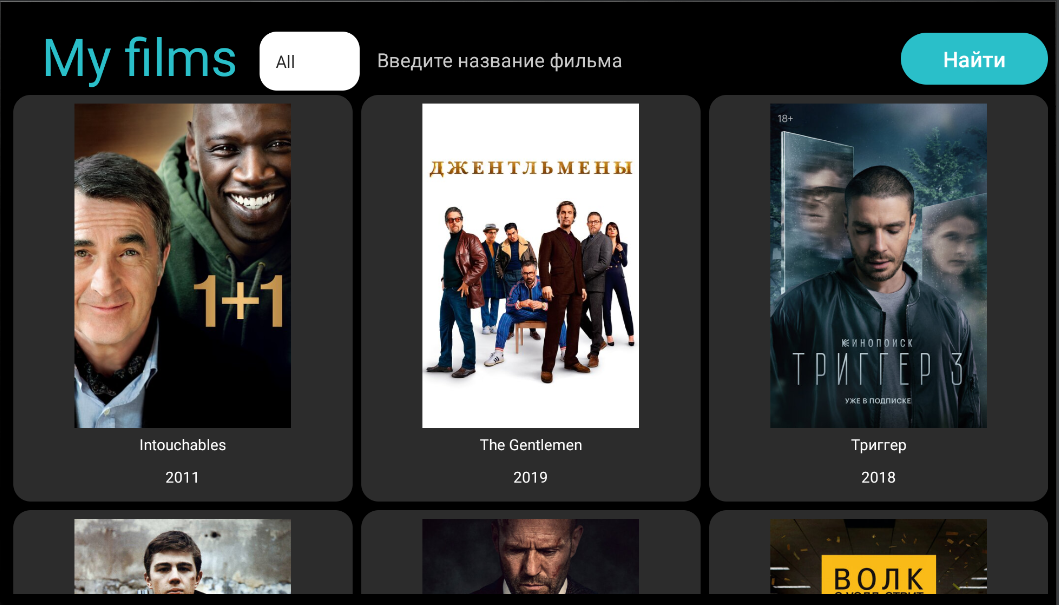
После истечения, происходит автоматический переход на следующий экран:



На данном экране пользователь должен ввести свой логин и пароль. Так как программа учебная, то корректным логином будет – ects, а паролем – ects2024. После чего пользователь может нажать на кнопку и перейти на следующий экран:



При переходе на следующий экран пользователь увидит список фильмов. Для удобства использования был применён ScrollView



Также для удобства предусмотрена фильтрация по жанрам и поиск по названию