

Министерство образования и молодежной политики Свердловской области

ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Отчёт по учебной практике

УП 01.01

Выполнил: Сэпселев Тимофей Дмитриевич

Группа: ПР-22

Преподаватель: Мирошниченко Г.В.

2023

Содержание

[1. **Задание №8 (вариант 3)** 3](#_Toc137062892)

[1.1 Описание задачи 3](#_Toc137062893)

[1.2 Структура проекта 4](#_Toc137062894)

[1.3 Описание разработанных функций 8](#_Toc137062895)

[1.4 Алгоритм решения 9](#_Toc137062896)

[1.5 Используемые библиотеки 10](#_Toc137062897)

[1.8 Описание пользовательского интерфейса 11](#_Toc137062898)

[1.9 Приложение (pr screen экранов) 12](#_Toc137062899)

# 1. **Задание №8 (вариант 8)**

## Описание задачи

Используя фрагменты, Room, RecycleView, SharedPreference создать

мобильное приложение, состоящее из 4-х экранов.

1 экран – SigninScreen (с проверкой на пустоту, с сохранением имени

пользователя при повторном входе и с проверкой сохраненного пароля,

используя механизм SharedPreference)

2 экран – Main Screen (реализовать меню)

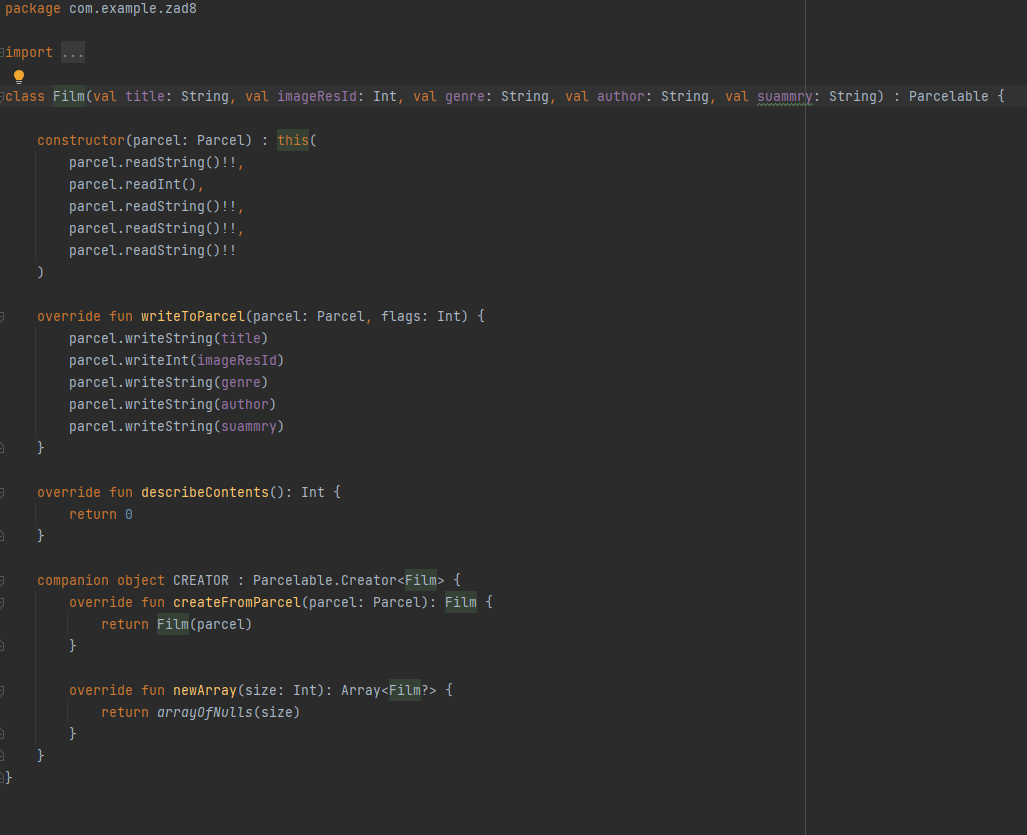
3 экран – Подборка (фильмы с иконкой, названием) – Room

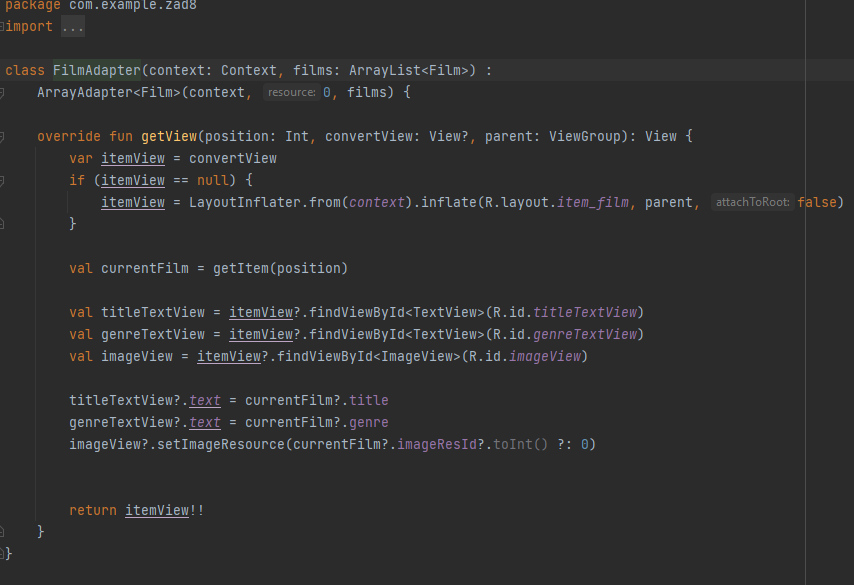
Реализовать возможность выбора, добавления, удаления, поиска фильмов

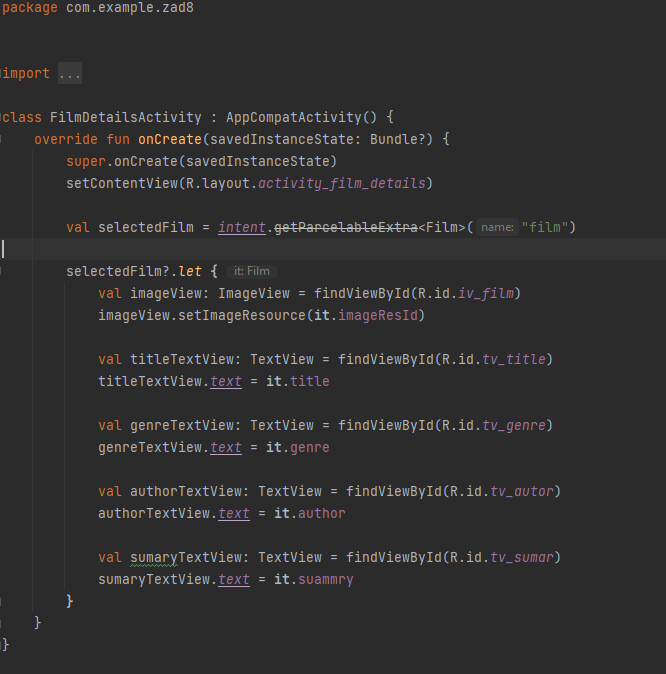
4 экран – при выборе фильма вывод информации о фильме (фото, автор,

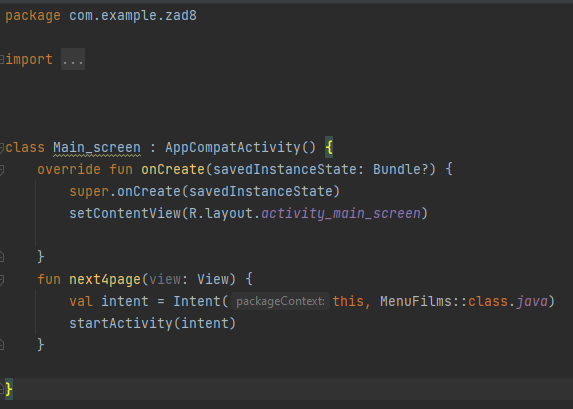
название, жанр)

## Структура проекта









class MenuFilms : AppCompatActivity() {  
 private lateinit var filmList: ArrayList<Film>  
 private lateinit var listView: ListView  
 private val PICK\_IMAGE\_REQUEST = 1  
 private var selectedImageUri: Uri? = null  
 private lateinit var sortedFilmList: ArrayList<Film>  
 private var selectedFilmPosition: Int = -1  
 private lateinit var filteredFilmList: ArrayList<Film>  
 private lateinit var originalFilmList: ArrayList<Film>  
 private lateinit var filteredFilmAdapter: FilmAdapter  
 private lateinit var originalFilmAdapter: FilmAdapter  
  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_menu\_films*)  
  
 originalFilmList = ArrayList()  
 filteredFilmList = ArrayList()  
 filmList = ArrayList()  
 filmList.add(Film("Аватар", R.drawable.*mov1*, "Боевик/Фантастика", "Джеймс Кэмерон", "Бывший морпех Джейк Салли прикован к инвалидному креслу. Несмотря на немощное тело, Джейк в душе по-прежнему остается воином. Он получает задание совершить путешествие в несколько световых лет к базе землян на планете Пандора, где корпорации добывают редкий минерал, имеющий огромное значение для выхода Земли из энергетического кризиса. "))  
 filmList.add(Film("Хищник", R.drawable.*mov2*, "Боевик/Фантастика", "Шейн Блэк", "Бывший военный обнаруживает злобную инопланетную расу на Земле, но никто ему не верит, что эти твари существуют. Его сын-аутист становится ключевой фигурой в схватке с Хищниками, поскольку мальчик умеет невероятно быстро учить языки и, судя по всему, сможет понять пришельцев."))  
 filmList.add(Film("Начало", R.drawable.*mov3*, "Боевик/Фантастика", "Кристофер Нолан", "Кобб – талантливый вор, лучший из лучших в опасном искусстве извлечения: он крадет ценные секреты из глубин подсознания во время сна, когда человеческий разум наиболее уязвим. Редкие способности Кобба сделали его ценным игроком в привычном к предательству мире промышленного шпионажа, но они же превратили его в извечного беглеца и лишили всего, что он когда-либо любил. "))  
 filmList.add(Film("Обливион", R.drawable.*mov4*, "Боевик/Фантастика", "Джозеф Косински", "Земля, пережившая войну с инопланетными захватчиками, опустела; остатки человечества готовятся покинуть непригодную для жизни планету. Главный герой — техник по обслуживанию дронов — находит разбившийся корабль NASA, команда которого погибает у него на глазах. Ему удаётся спасти лишь одну женщину — и вскоре он понимает, что это перевернёт его жизнь. "))  
 filmList.add(Film("Титаник", R.drawable.*mov5*, "Драмма", "Джеймс Кэмерон", "В первом и последнем плавании шикарного «Титаника» встречаются двое. Пассажир нижней палубы Джек выиграл билет в карты, а богатая наследница Роза отправляется в Америку, чтобы выйти замуж по расчёту. Чувства молодых людей только успевают расцвести, и даже не классовые различия создадут испытания влюблённым, а айсберг, вставший на пути считавшегося непотопляемым лайнера."))  
 filmList.add(Film("Пассажиры", R.drawable.*mov6*, "Боевик/Фантастика", "Мортен Тильдум", "Космический корабль с 5 000 пассажирами на борту, погруженными на время полета в искусственный сон, держит курс на планету, которая станет для них новым домом. "))  
 filmList.add(Film("Паркер", R.drawable.*mov7*, "Боевик/Фантастика", "Тейлор Хэкфорд", "Главный герой — профессиональный вор, руководствующийся собственным моральным кодексом, но способный запросто отправить на тот свет любого, кто перейдет ему дорогу."))  
 filmList.add(Film("Чёрная дыра", R.drawable.*mov8*, "Боевик/Фантастика", "Дэвид Туи", "В недалеком будущем космический корабль терпит бедствие, в результате чего совершает вынужденную посадку на отдаленной планете. Но в результате такой посадки погибает командир корабля, часть команды и большинство пассажиров."))  
 filmList.add(Film("Троя", R.drawable.*mov9*, "Прюключения", " Вольфганг Петерсен", "1193 год до нашей эры. Парис украл прекрасную Елену, жену царя Спарты Менелая. За честь Менелая вступается его брат – царь Агамемнон. Его армия под предводительством Ахиллеса подошла к Трое и взяла город в кровавую осаду, длившуюся долгих десять лет… Два мира будут воевать за честь и власть. Тысячи умрут за славу. И за любовь нация сгорит дотла."))  
 filmList.add(Film("Матрица", R.drawable.*mov10*, "Боевик/Фантастика", "Лана Вачовски", "Жизнь Томаса Андерсона разделена на две части: днём он — самый обычный офисный работник, получающий нагоняи от начальства, а ночью превращается в хакера по имени Нео, и нет места в сети, куда он бы не смог проникнуть. Но однажды всё меняется."))  
  
  
  
  
 listView = findViewById(R.id.*listView*)  
 originalFilmAdapter = FilmAdapter(this, filmList)  
 listView.*adapter* = originalFilmAdapter  
 sortedFilmList = ArrayList(filmList)  
 listView.*onItemClickListener* = AdapterView.OnItemClickListener **{** parent, view, position, id **->** val selectedFilm = filmList[position]  
 val intent = Intent(this, FilmDetailsActivity::class.*java*)  
 intent.putExtra("film", selectedFilm)  
 startActivity(intent)  
 selectedFilmPosition=position  
 **}** }  
  
  
  
 override fun onOptionsItemSelected(item: MenuItem): Boolean {  
 when (item.*itemId*) {  
 R.id.*menu\_add* -> {  
 showAddFilmDialog()  
 return true  
 }  
 R.id.*menu\_delete* -> {  
 if (selectedFilmPosition != -1) {  
 filmList.removeAt(selectedFilmPosition)  
 (listView.*adapter* as FilmAdapter).notifyDataSetChanged()  
 Toast.makeText(this, "Фильм удален", Toast.*LENGTH\_SHORT*).show()  
 selectedFilmPosition=-1  
 } else {  
 Toast.makeText(this, "Выберите фильм для удаления", Toast.*LENGTH\_SHORT*).show()  
 }  
 return true  
 }  
 R.id.*menu\_search* -> {  
 showSearchDialog()  
 return true  
 }  
 R.id.*menu\_sort* -> {  
 filmList.*sortBy* **{ it**.genre **}** sortedFilmList.*sortBy* **{ it**.genre **}** listView.*adapter* = FilmAdapter(this, sortedFilmList)  
 (listView.*adapter* as FilmAdapter).notifyDataSetChanged()  
  
 listView.*adapter* = FilmAdapter(this, filmList)  
 (listView.*adapter* as FilmAdapter).notifyDataSetChanged()  
 Toast.makeText(this, "Сортировка по жанру", Toast.*LENGTH\_SHORT*).show()  
 return true  
 }  
 R.id.*menu\_show\_all* -> {  
 showAllFilms()  
 return true  
 }  
 else -> return super.onOptionsItemSelected(item)  
  
 }  
 }  
 private fun showAllFilms() {  
 recreate()  
 }  
  
  
  
  
  
 fun showMenu(view: View) {  
 val popupMenu = PopupMenu(this, view)  
 popupMenu.inflate(R.menu.*menu\_films*)  
  
 popupMenu.setOnMenuItemClickListener **{** item **->** onOptionsItemSelected(item)  
 **}** popupMenu.show()  
 }  
  
  
  
 private fun showAddFilmDialog() {  
 val dialog = Dialog(this)  
 dialog.setContentView(R.layout.*dialog\_add\_film*)  
 dialog.*window*?.setLayout(WindowManager.LayoutParams.*MATCH\_PARENT*, WindowManager.LayoutParams.*WRAP\_CONTENT*)  
  
 val etTitle = dialog.findViewById<EditText>(R.id.*et\_title*)  
 val etGenre = dialog.findViewById<EditText>(R.id.*et\_genre*)  
 val etAutoh = dialog.findViewById<EditText>(R.id.*et\_autor*)  
 val etOpsaine = dialog.findViewById<EditText>(R.id.*et\_sumary*)  
 val btnAdd = dialog.findViewById<Button>(R.id.*btn\_add*)  
 val btnAddPhoto = dialog.findViewById<Button>(R.id.*btn\_choose\_photo*)  
  
 btnAddPhoto.setOnClickListener **{** val intent = Intent(Intent.*ACTION\_PICK*, MediaStore.Images.Media.*EXTERNAL\_CONTENT\_URI*)  
 startActivityForResult(intent, PICK\_IMAGE\_REQUEST)  
 **}** btnAdd.setOnClickListener **{** val title = etTitle.*text*.toString().*trim*()  
 val genre = etGenre.*text*.toString().*trim*()  
 val autor = etAutoh.*text*.toString().*trim*()  
 val opsanie = etOpsaine.*text*.toString().*trim*()  
  
 if (title.*isNotEmpty*() && genre.*isNotEmpty*()) {  
 val newFilm = Film(title, R.drawable.*nofoto*, genre, autor, opsanie)  
 filmList.add(newFilm)  
 (listView.*adapter* as FilmAdapter).notifyDataSetChanged()  
 Toast.makeText(this, "Фильм добавлен", Toast.*LENGTH\_SHORT*).show()  
 dialog.dismiss()  
 } else {  
 Toast.makeText(this, "Введите название и жанр фильма, и выберите фотографию", Toast.*LENGTH\_SHORT*).show()  
 }  
 **}** dialog.show()  
 }  
  
 private fun showSearchDialog() {  
  
 val searchDialog = Dialog(this)  
 searchDialog.setContentView(R.layout.*dialog\_search*)  
 searchDialog.*window*?.setLayout(WindowManager.LayoutParams.*MATCH\_PARENT*, WindowManager.LayoutParams.*WRAP\_CONTENT*)  
  
 val etSearch = searchDialog.findViewById<EditText>(R.id.*et\_search*)  
 val btnSearch = searchDialog.findViewById<Button>(R.id.*btn\_search*)  
  
 btnSearch.setOnClickListener **{** val searchQuery = etSearch.*text*.toString().*trim*()  
 filterFilmList(searchQuery)  
  
 searchDialog.dismiss()  
 **}** searchDialog.show()  
 }  
  
 private fun filterFilmList(searchQuery: String) {  
 filteredFilmList.clear()  
  
 for (film in filmList) {  
 if (film.title.*contains*(searchQuery, ignoreCase = true)) {  
 filteredFilmList.add(film)  
 }  
 }  
  
 if (filteredFilmList.*isNotEmpty*()) {  
 filteredFilmAdapter = FilmAdapter(this, filteredFilmList)  
 listView.*adapter* = filteredFilmAdapter  
 filteredFilmAdapter.notifyDataSetChanged()  
  
 filmList= ArrayList(filteredFilmList)  
 } else {  
 listView.*adapter* = FilmAdapter(this, filmList)  
 (listView.*adapter* as FilmAdapter).notifyDataSetChanged()  
 }  
 }  
  
 override fun onActivityResult(requestCode: Int, resultCode: Int, data: Intent?) {  
 super.onActivityResult(requestCode, resultCode, data)  
 if (requestCode == PICK\_IMAGE\_REQUEST && resultCode == Activity.*RESULT\_OK* && data != null) {  
 selectedImageUri = data.*data* }  
 }  
  
}

class SigninUpScreen : AppCompatActivity() {  
 private lateinit var sharedPreferences: SharedPreferences  
  
 private lateinit var editTextEmail: EditText  
 private lateinit var editTextPassword: EditText  
  
  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_signin\_up\_screen*)  
 sharedPreferences = getSharedPreferences("UserData", *MODE\_PRIVATE*)  
  
 editTextEmail = findViewById(R.id.*editText3*)  
 editTextPassword = findViewById(R.id.*editText4*)  
  
  
 }  
 fun next2page(view: View) {  
  
 val email = editTextEmail.*text*.toString()  
 val password = editTextPassword.*text*.toString()  
  
 if ( email.*isEmpty*() || password.*isEmpty*()) {  
 Toast.makeText(this, "Заполните все поля", Toast.*LENGTH\_SHORT*).show()  
 return  
 }  
  
 saveUserData( email, password)  
 val intent = Intent(this, Main\_screen::class.*java*)  
 startActivity(intent)  
 }  
  
 private fun saveUserData(email: String, password: String) {  
 val editor = sharedPreferences.edit()  
  
 editor.putString("email", email)  
 editor.putString("password", password)  
  
 editor.apply()  
 }  
 fun loadSavedUserData(view: View) {  
  
 val email = sharedPreferences.getString("email", "")  
 val password = sharedPreferences.getString("password", "")  
  
 if ( email.*isNullOrEmpty*() || password.*isNullOrEmpty*()) {  
 Toast.makeText(this, "Нет сохраненных данных", Toast.*LENGTH\_SHORT*).show()  
 } else {  
 val userData = "E-mail: $email\nПароль: $password"  
 Toast.makeText(this, userData, Toast.*LENGTH\_LONG*).show()  
  
 editTextEmail.setText(email)  
 editTextPassword.setText(password)  
  
 }  
 }  
}

## Описание разработанных функций

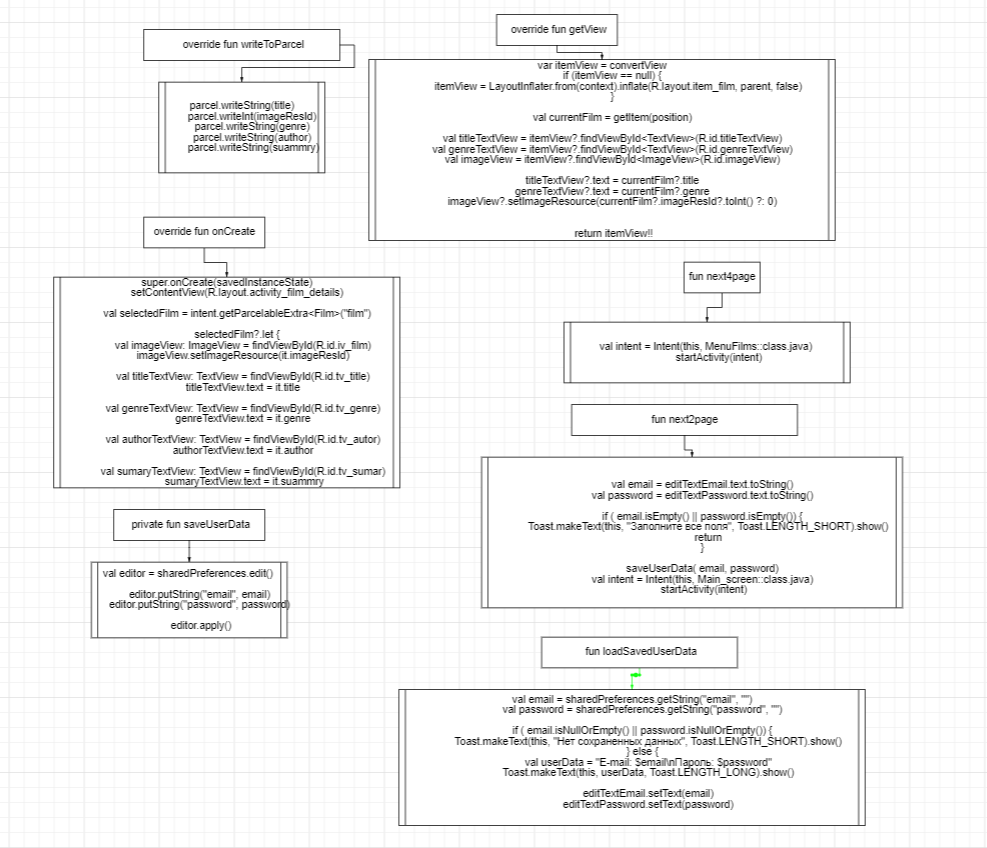
SetPreference – сохраняет информацию о регистрации в SharedPreference.

GetPreference – проверяет введенный логин, почту, пароль и если введенные строки совпадают, то происходит переход на следующий экран.

Addcase – добавление дел в список на втором экране.

Open – если приложения было перезапущено, при нажатии на кнопку «загрузить», у нас появляется список, добавленный из прошлой сессии.

## Алгоритм решения



## 1.5 Используемые библиотеки

import android.content.Intent  
import android.content.SharedPreferences  
import androidx.appcompat.app.AppCompatActivity  
import android.os.Bundle  
import android.view.View  
import android.widget.EditText  
import android.widget.Toast

1.6 Тестовые случаи

Если в строках регистрации будут найдены пустые строки, у нас выведется сообщение:

\

1.7 Используемые инструменты

Язык Kotlin, Android Studio, Emulator Pixel 2 API 28 (Android 9.0).

## 1.8 Описание пользовательского интерфейса

Edit text, edit text2– ввод данных при входе.

## 1.9 Приложение (pr screen экранов)

