

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ

ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»

Кафедра теории функций и стохастического анализа

РАЗРАБОТКА МОБИЛЬНОГО ПРИЛОЖЕНИЯ МАГАЗИНА

КОМПЬЮТЕРНЫХ ИГР

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 451 группы

направления 38.03.05 — Бизнес-информатика

механико-математического факультета

Арсентьева Дмитрия Александровича

Научный руководитель

доцент, к. э. н.

А. Р. Файзлиев

Заведующий кафедрой

д. ф.-м. н., доцент

С. П. Сидоров

Саратов 2022

Введение. Целью бакалаврской работы является разработка мобильного приложения магазина компьютерных игр на базе платформы андроид студии.

Android приложения на сегодняшний день очень востребованы. Android самая популярная мобильная платформа в мире. Согласно последним опросам, в мире свыше миллиарда активных Android устройств, и их количество продолжает стремительно расти. Android полнофункциональная платформа с открытым кодом на базе Linux, разрабатываемая компанией Google. Это мощная платформа разработки, включающая все необходимое для построения современных приложений из кода Java и XML. Более того, построенные приложения могут устанавливаться на множестве разных устройств: телефонах, планшетах и прочих.

Для выполнения данной цели были поставлены следующие задачи:

1. Создание страницы регистрации и авторизации;
2. Подключение андроид приложения к БД Firebase;
3. Реализация функционала магазина;
4. Создание панели администратора с возможностью добавления новых товаров;
5. Создание встроенной игры викторины для пользователей;

Реализация возможности регистрации и авторизации. При создании android приложения магазина видеоигр были учтены такие параметры, как язык разработки и минимальный SDK.

На рисунке 1 показано определение параметров при создании приложения. Были выставлены параметры, позволяющие 98% пользователей использовать приложение.

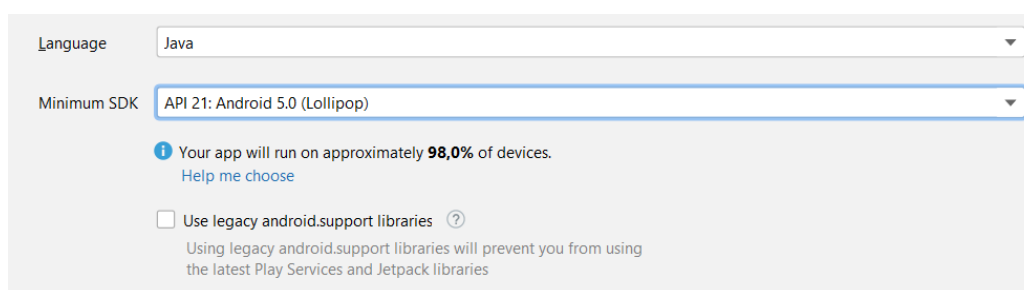


Рисунок 1 – Определение параметров при создании приложения

Приложение Android состоит из одной или нескольких деятельности,

именуемых как `activities`. Оно должно содержать как минимум одну деятельность. Также помимо этих деятельностей в проекте важную роль играет файл `build.gradle`, который помимо всего прочего содержит в себе список подгружаемых для проекта библиотек:

```
1 dependencies {
2     implementation fileTree(dir: "libs", include: ["*.jar"])
3     implementation 'androidx.appcompat:appcompat:1.3.0'
4     implementation 'androidx.cardview:cardview:1.0.0'
5     implementation 'androidx.recyclerview:recyclerview:1.2.0'
6     implementation 'androidx.constraintlayout:constraintlayout:2.0.4'
7     implementation 'com.google.firebase:firebase-analytics:17.2.2'
8     implementation 'com.github.rey5137:material:1.2.5'
9     implementation 'com.google.firebase:firebase-database:19.6.0'
10    implementation 'com.google.firebase:firebase-storage:19.2.1'
11    implementation 'com.firebaseui:firebase-ui-database:6.1.0'
12    implementation 'com.google.android.material:material:1.3.0'
13    implementation 'androidx.navigation:navigation-fragment:2.2.2'
14    implementation 'androidx.navigation:navigation-ui:2.3.5'
15    implementation 'androidx.lifecycle:lifecycle-livedata-ktx:2.2.0'
16    implementation 'androidx.lifecycle:lifecycle-viewmodel-ktx:2.2.0'
17    testImplementation 'junit:junit:4.12'
18    implementation 'io.paperdb:paperdb:2.7.1'
19    implementation 'de.hdodenhof:circleimageview:3.1.0'
20    implementation 'com.theartofdev.edmodo:android-image-cropper
21        :2.8.0'
22    implementation 'com.squareup.picasso:picasso:2.71828'
23    androidTestImplementation 'androidx.test.espresso:espresso-core
24        :3.2.0'
25 }
```

Для разработки и тестирования приложения необходимо виртуальное устройство, чтобы по мере продвижения разработки можно было протестировать приложение и различные его функции.

Эмулятор Android, включенный в состав Android SDK, позволяет создать

имитационную среду для запуска приложений Android под управлением Windows, Mac OS X, либо Linux. Эмулятор отображает весьма реалистичное окно интерфейса пользователя Android.

Для тестировки конкретно этого приложения было выбрано виртуальное устройство имитирующее смартфон Pixel. После добавление данного виртуального устройства оно стало отображаться во вкладке Your Virtual Devices, откуда в любой момент можно запустить данное виртуальное устройство.

Данные пользователей будут храниться на платформе Firebase. Через андроид студию можно в пару кликов подключиться к Firebase. Firebase - полезная для разработки мобильных приложений платформа. Она предоставляет множество функций, однако нас больше интересуют такие функции как аутентификация и безопасность, а хранилище данных в реальном времени и хранилище файлов. После подключения к Firebase автоматически подгружаются необходимые для работы библиотеки.

При создании стартовой страницы особое внимание было обращено на разработку интуитивно понятного для пользователя интерфейса. Были добавлены 2 кнопки регистрации и входа в нижней части экрана. вверху расположилось название приложения - Games magazin под которым находится слоган. Пример программного кода, реализующего отображение названия приложения можно увидеть ниже:

```
1 <EditText
2     android:id="@+id/app_logo"
3     android:layout_width="wrap_content"
4     android:layout_height="118dp"
5     android:layout_alignParentEnd="true"
6     android:layout_marginEnd="11dp"
7     android:fontFamily="sans-serif-smallcaps"
8     android:text="@string/app_in_name"
9     android:textAlignment="center"
10    android:textColor="@android:color/white"
11    android:textSize="55sp" />
```

Помимо прочих параметров, отвечающих за расположение, стиль текста и его размер, можно отметить то, что сам текст данного объекта находится в

файле strings.xml, а на 8 строчки кода за место того, чтобы просто написать нужный текст ссылаемся на название переменной в вышеоговоренном файле, что в целом не обязательно, но является рекомендацией в документациях от андроид студии.

Подобным образом были созданы активности со страницей регистрации и входа. Конечный результат стартовой страницы и страницы регистрации можно увидеть на рисунке 2 и 3 соответственно.

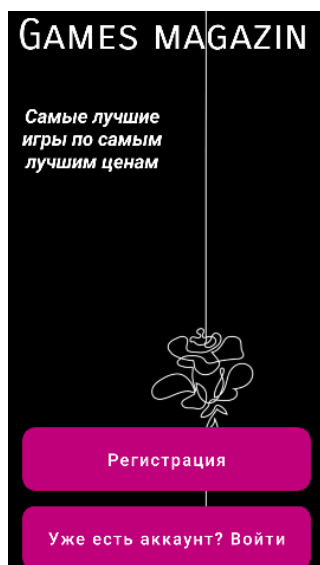


Рисунок 2 – Стартовая страница

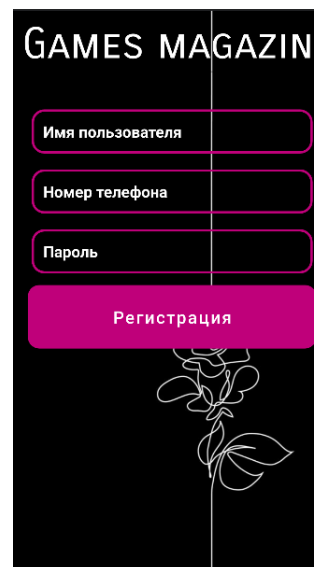


Рисунок 3 – Страница регистрации пользователя

У каждой активности помимо xml файла имеется jail файл, в котором используя ссылки на объекты из xml файла и прописываются действия с этими объектами. Ниже можно увидеть созданные переменные в jail файле страницы регистрации, которые используют ссылки на объекты из xml файла страницы регистрации:

```
1 protected void onCreate(Bundle savedInstanceState) {  
2     super.onCreate(savedInstanceState);  
3     setContentView(R.layout.activity_register);  
4  
5     registerBtn = (Button) findViewById(R.id.register_btn);  
6     usernameInput = (EditText) findViewById(R.id.register_username_input);  
7     phoneInput = (EditText) findViewById(R.id.register_phone_input);  
8     passwordInput = (EditText) findViewById(R.id.register_password_input);
```

```

9     loadingBar = new ProgressDialog(this);
10
11     registerBtn.setOnClickListener(new View.OnClickListener() {
12         @Override
13         public void onClick(View v) {
14             CreateAccount();
15         }
16     });
17 }

```

Подобным образом устроено и в других активностях, там у каждой активности есть xml и java файл. В случае если пользователь не введёт какие-либо необходимые данные можно вывести ему соответствующие сообщения. Как, например, ниже в java файле страницы регистрации:

```

1 private void CreateAccount() {
2     String username = usernameInput.getText().toString();
3     String phone = phoneInput.getText().toString();
4     String password = passwordInput.getText().toString();
5
6     if(TextUtils.isEmpty(username))
7     {
8         Toast.makeText(this, "Введите имя", Toast.LENGTH_SHORT).show()
9         ;
10    }
11    else if(TextUtils.isEmpty(phone))
12    {
13        Toast.makeText(this, "Введите номер", Toast.LENGTH_SHORT).show
14        ();
15    }
16    else if(TextUtils.isEmpty(password))
17    {
18        Toast.makeText(this, "Введите пароль", Toast.LENGTH_SHORT).
19        show();

```

```

17     }
18     else
19     {
20         loadingBar.setTitle("Создание аккаунта");
21         loadingBar.setMessage("Пожалуйста, подождите...");
22         loadingBar.setCanceledOnTouchOutside(false);
23         loadingBar.show();
24
25         ValidatePhone(username, phone, password);
26     }
27 }

```

Реализация функционала магазина. После входа пользователя в приложение он попадёт в меню, которое можно увидеть на рисунке 4.

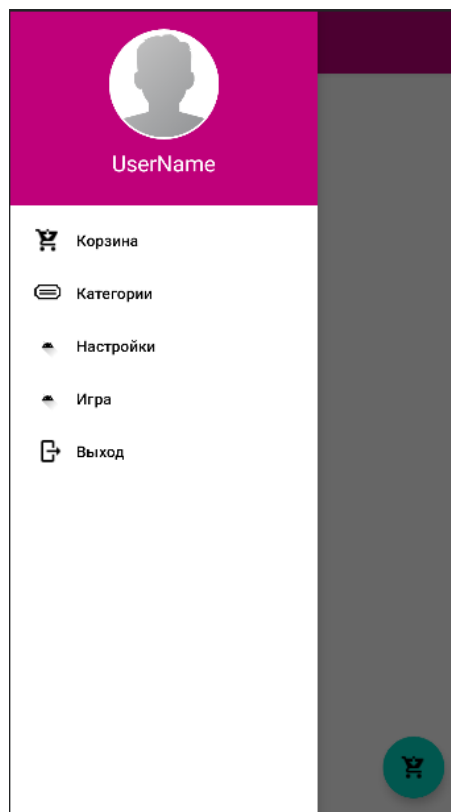


Рисунок 4 – Итоговый вид меню пользователя

Для пользователя доступны следующие разделы:

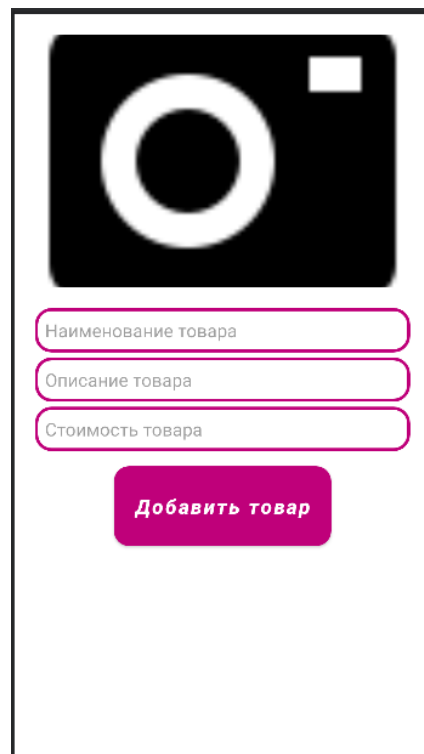
1. Корзина, куда будут попадать выбранные товары;
2. Категории, где пользователь сможет выбрать категории в которой будет

- выбирать игру;
3. Настройки, где пользователь сможет добавить себе аватарку;
 4. Игра, где пользователь сможет пройти небольшую викторину по вопросам видеоигр;
 5. Выход, нажав на который пользователь выйдет из своего аккаунта и вернётся на страницу авторизации;

Товары в категории можно добавить из самого приложения, зайдя как администратор. Для этого нужно войти как админ на странице входа, нажав на соответствующую кнопку. Страница добавления товара содержит следующие поля для ввода:

1. Изображение товара, куда следует загрузить изображение видеоигры;
2. Наименование товара, где должно быть прописано полное название видеоигры;
3. Описание, где должно быть прописано краткое описание видеоигры;
4. Стоимость товара, где должна быть прописана точная стоимость данной видеоигры в приложении в рублях;

Страницу добавления нового товара можно увидеть на рисунке 5.



The image shows a mobile application interface for adding a new item. It features a large black square with a white camera icon at the top. Below this are three input fields with rounded corners and pink borders, labeled 'Наименование товара', 'Описание товара', and 'Стоимость товара'. At the bottom of the form is a red button with white text that says 'Добавить товар'.

Рисунок 5 – Страница добавления нового товара

В разделе настроек находится поле ввода имени пользователя, а также поле ввода номера телефона. Помимо 2 этих полей также имеется изображение, нажав на которое пользователь сможет добавить изображение профиля, которое будет отображаться в странице выбора раздела в верхней части экрана.

Реализация игры викторины по видеоиграм. Викторина содержит 5 фактов, правдивость которых предстоит определить пользователю. Пример самой страницы викторины с одним из вопросов можно увидеть на рисунке 6.

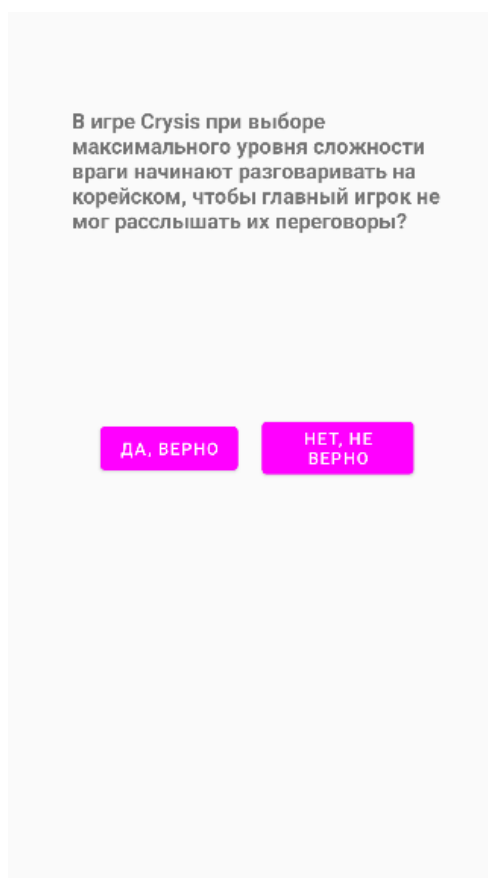


Рисунок 6 – Страница викторины

Пользователь видит вопрос в верхней части экрана, ниже находятся 2 кнопки, по средству которых пользователь сможет выразить своё мнение на счёт правдивости факта. После нажатия на кнопку пользователю выведется сообщение, которое оповестит его о правильном, либо ошибочном ответе на данный вопрос. Далее пользователя переведёт на следующую страницу со вторым вопросом и так далее. В самом конце после всех пяти фактов поль-

зователю выведется число правильных ответов на общее количество фактов. Правильность того или иного факта определяется в отдельном массиве questions заранее созданного класса TrueFalse. Этот массив содержит 2 параметра: сам вопрос, текст которого содержится в файле strings.xml и true или false в зависимости от того правдив ли факт. Данный массив можно увидеть ниже:

```
1  private TrueFalse[] questions = new TrueFalse[] {
2      new TrueFalse(R.string.quest_1, true),
3      new TrueFalse(R.string.quest_2, false),
4      new TrueFalse(R.string.quest_3, true),
5      new TrueFalse(R.string.quest_4, true),
6      new TrueFalse(R.string.quest_5, false),
7  };
```

Заключение. В процессе выполнения бакалаврской работы все поставленные задачи были выполнены, а именно:

1. Создание страницы регистрации и авторизации;
2. Подключение андроид приложения к БД от google - Firebase;
3. Реализация функционала магазина;
4. Создание панели администратора с возможностью добавления новых товаров;
5. Создание встроенной игры викторины для пользователей;

Следовательно и цель бакалаврской работы - создание мобильного приложения магазина компьютерных игр можно считать выполненной.