|  |  |  |
| --- | --- | --- |
|  |  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |

Институт Информационных технологий

Кафедра Математического обеспечения и стандартизации информационных технологий

**КУРСОВАЯ РАБОТА**

по дисциплине «Разработка мобильных приложений»

**Тема курсовой работы**

Создание программного комплекса

«Тестирование обучаемых. Мобильное приложение»

**Студент группы** ИКБО-12-19 Петренко Антон Александрович

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись студента)

**Руководитель курсовой работы** доцент Синицын И.В.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись руководителя)

Работа представлена к защите «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

Допущен к защите «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

Москва 2021

|  |  |  |
| --- | --- | --- |
|  |  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |

Институт Информационных технологий

Кафедра Математического обеспечения и стандартизации информационных технологий

Утверждаю

Заведующий кафедрой МОСИТ

\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Головин С.А.

«\_\_\_» \_\_\_\_\_\_\_\_\_ 2021 г.

**ЗАДАНИЕ**

**на выполнение курсовой работы по дисциплине**

«Разработка мобильных приложений»

Студент Петренко Антон Александрович Группа ИКБО-12-19

**Тема работы:** Создание программного комплекса «Тестирование обучаемых. Мобильное приложение».

**Исходные данные:**

**З**адание на курсовую работу –тема №7. Разработать программный комплекс «Тестирование обучаемых. Мобильное приложение».

**Функционал разрабатываемой системы должен** предоставлять возможности полностью интерактивной системы с понятным дружественным графическим интерфейсом и обеспечивать необходимую функциональность.

**Перечень вопросов, подлежащих разработке, и обязательного графического материала:**

Установка и настройка сред и систем программирования:Android Studio,Eclipse IDE , IntelliJ IDEA, Kivi и подобные.Установка и настройка эмуляторов Android.

Изучение жизненного цикла программ и создание мобильного программного комплекса с применением языков программирования JAVA, Python, Kotlin,Swift, и др. (применяются по отдельному указанию руководителя), согласно темы курсового проектирования. Реализация в создаваемом программном комплексе системы клиент-сервер (клиент-мобильное приложение, сервер-WEB сервер-Apache).

Возможна реализация клиентской части на смартфоне или планшетнике через WAP или подобную систему. Обеспечение безопасности информации при работе планового программного комплекса (обеспечение работы ролевой модели безопасности).

Тестирование и диагностика созданного программного продукта.

Портирование написанной программной системы на внешних хостах в сети Интернет.

**Срок представления к защите курсовой работы:** до «\_09\_» июня 2021 г.

**Задание на курсовую работу выдал** \_\_\_\_\_\_\_\_\_\_\_\_ Синицын И.В.

«15» февраля 2021 г.

**Задание на курсовую работу получил**  \_\_\_\_\_\_\_\_\_\_\_\_ Петренко А.А.

# СОДЕРЖАНИЕ

[**1.** **СОДЕРЖАНИЕ** 4](#_Toc88330887)

[2. ВВЕДЕНИЕ 5](#_Toc88330888)

[3. АНАЛИЗ И ИССЛЕДОВАНИЕ СУЩЕСТВУЮЩИХ СИСТЕМ 7](#_Toc88330889)

[1.1. Исследование существующих систем 7](#_Toc88330890)

[1.1.1. Программный комплекс “Учёт посещаемости” 7](#_Toc88330891)

[1.1.2. Программный комплекс «Alora» 8](#_Toc88330892)

[1.2. Определение постановки задачи 8](#_Toc88330893)  
[1.3. Создание макета 9](#_Toc88330893)

[1.3.1. Создание страницы списка студентов. 10](#_Toc88330894)  
[1.3.2. Диалоговое окно авторизации и регистации . 11](#_Toc88330894)  
[1.3.3. Детализация по студенту. 12](#_Toc88330894)

[4. РЕАЛИЗАЦИЯ ПОСТАВЛЕННОЙ ЗАДАЧИ 13](#_Toc88330895)

[2.1. Определение необходимого программного обеспечения 13](#_Toc88330896)

[2.2. Установка необходимого программного обеспечения 13](#_Toc88330897)

[2.3. Разработка серверной части 15](#_Toc88330898)

[2.4. Разработка клиентской части 17](#_Toc88330899)

[2.4.1 Методы работы с бд. 17](#_Toc88330900)

[2.4.2 Графические формы приложения 21](#_Toc88330901)

[2.4.3 Основная активность 21](#_Toc88330902)

[2.4.4 Авторизация и регистрация 23](#_Toc88330903)

[2.4.1. Активность детализации 25](#_Toc88330904)

[5. ВЫВОДЫ ПО РАБОТЕ 26](#_Toc88330905)

[6. СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 27](#_Toc88330906)

# ВВЕДЕНИЕ

Цель данной курсовой работы – разработать мобильное приложение для тестирования обучаемых.

Задачи:

* реализовать клиент-серверное приложение «Тестирование обучаемых»;
* реализовать клиентскую часть в виде мобильного приложения для смартфона или планшета;
* организовать ролевую модель разграничения доступа к данным.

По поводу актуальности данной курсовой работы можно сказать следующее: В современном мире от обучения зависит очень многое, если же не абсолютно всё. Поэтому не удивительно что я выбрал тему тестирования обучающихся. Благодаря технологиям всё больше образовательных учреждений переходят на тестирование учеников или студентов в цифровом формате. На фоне этого виден рост востребованности в данной сфере.

Несомненно, фундаментом любого успешного проекта, либо же сайта является идея и когда передо мной оказался список тем первым вопросом, которым я задался был, а что собственно я смогу придумать и на сколько это будет полезно мне в плане развития моих навыков, согласитесь каждый может взять простую тему и без всякого экскурса сделать великолепную в рамках проекта работу, но разве это интересно?

Я же выбрал другой путь, несомый духом авантюризма я искал то, что заставит меня пошевелить извилинами и найти новые, неординарные решения, и по итогу мой выбор пал на эту тему.

# АНАЛИЗ И ИССЛЕДОВАНИЕ СУЩЕСТВУЮЩИХ СИСТЕМ

* 1. Определение постановки задачи

В рамках данного курсового проекта было принято решение создать упрощенное приложение для тестирования обучающихся.

Таким образом, приложение будет содержать только базовый функционал, а именно:

* Возможность просмотра и редактирования списка вопросов;
* Возможность зарегистрироваться /Авторизироваться.
* распределение прав пользователей, возможности доступа к различной информации и функциям системы.;
* ведение единой базы данных;

Создание такого программного комплекса можно разбить на следующие подзадачи:

1. Установка и настройка среду для разработки приложений под управлением ОС Android, установить эмулятор Android.
2. Разработка программный комплекс «Тестирование обучаемых. Мобильное приложение».
   1. Установка необходимого программного обеспечения.
   2. Создание серверной части и обеспечение разграничения доступа к данным на основе ролевой модели.
   3. Создание клиентского мобильного приложения для взаимодействия пользователя с серверной частью.
   4. Тестирование и исправление недоработок.

# РЕАЛИЗАЦИЯ ПОСТАВЛЕННОЙ ЗАДАЧИ

## Определение необходимого программного обеспечения

Для создания программного комплекса согласно заданию, необходима реализация двух частей: серверной и клиентской. Для реализации серверной части потребуется:

* СУБД Mysqllite;

Для написания и тестирования клиентской части – мобильного приложения – потребуются следующие компоненты:

* интегрированная среда разработки, которая позволят создавать приложения под управлением ОС Android;
* эмулятор Android для первичного тестирования приложения.

В качестве среды разработки была выбрана Android Studio от компании JetBrains, которая была разработана специально для создания мобильных приложений и, вдобавок, включает в себя встроенный эмулятор.

## Установка необходимого программного обеспечения

Интегрированная среда разработки Android Studio также является бесплатной и доступа для скачивания по ссылке: <https://developer.android.com/studio>.

Как уже упоминалось, Android Studio поставляется вместе со встроенным эмулятором. Для создания виртуального устройства, необходимо перейти в Android Virtual Device Manager (см. рис. 2.1).

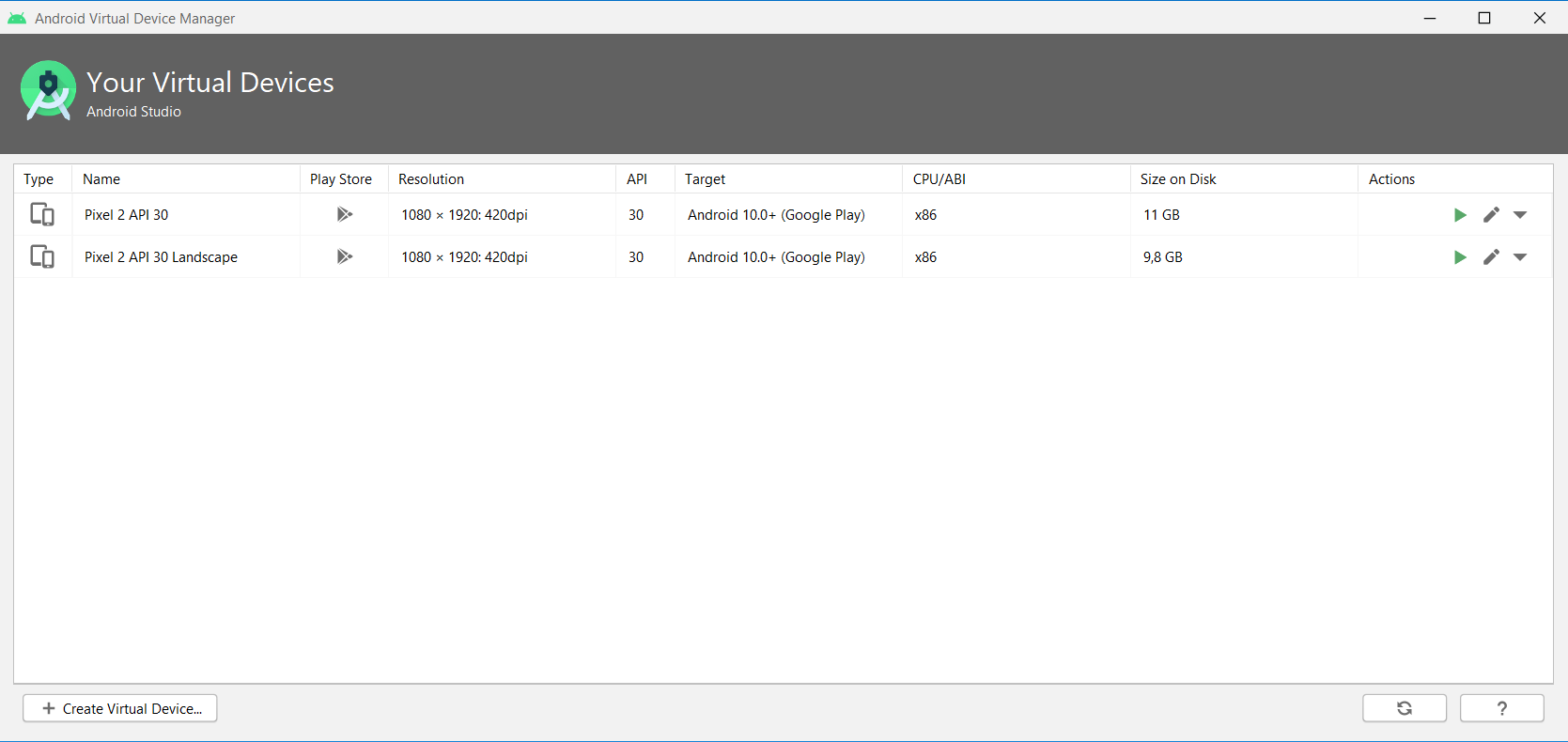


Рисунок 1.1 – Менеджер виртуальных устройств Android

Далее нужно нажать «Create Virtual Device…» и в открывшемся окне (см. рис. 2.3) выбрать подходящие параметры виртуального устройства: тип, размер экрана и т.д.

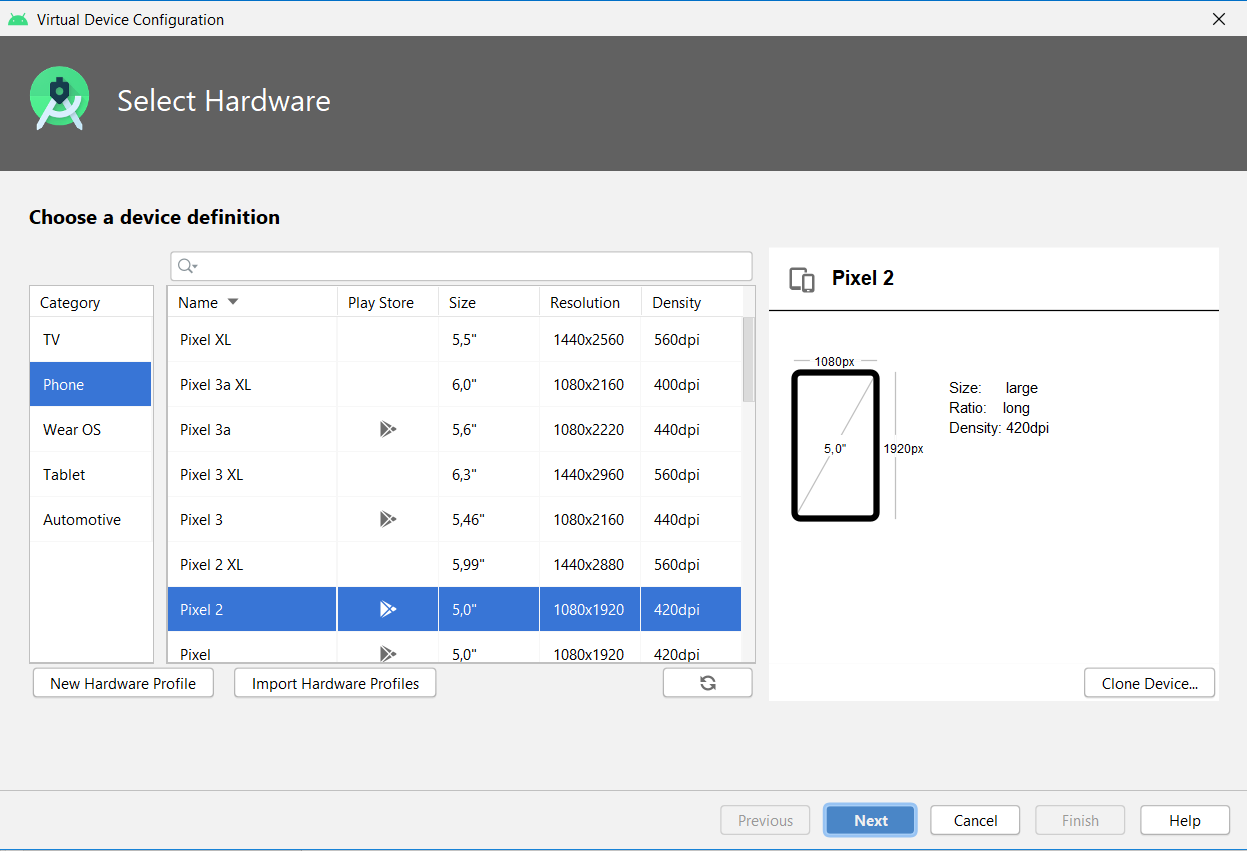


Рисунок 1.2 – Выбор параметров виртуального устройства

## Разработка серверной части

Так как в качестве бд выбрана mysqllite структура бд представляет из себя файлы классов.

В рамках данного курсового проекта были созданы следующие классы:

* Класс AuthDatabase отвечающий за авторизацию регистрацию пользователей;
* Класс DatabaseHelper содержащий в себе таблицу успеваемости студентов;
* Класс MyDatabaseHelper содержащий таблицу списка студентов;

И соответственно были созданы следующие таблицы:

* таблица «Users» (Пользователи) Рисунок 2.3,

Содержит 4 поля – (ID, Login, Password, Allow).

*ID*-идентификационный номер пользователя – является первичным ключом с параметром AUTO\_INCREMENT, это номинальное поле т.е. при работе с базой пользователь его не видит.  
*Login* – хранит логин пользователя, тип данных text.  
Password-содержит пароль пользователя, тип данных text.  
allow-содержит ключ доступа администратора.

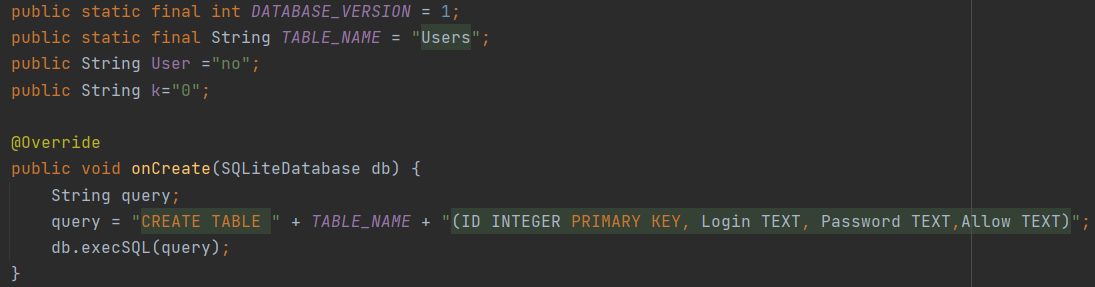


Рисунок 2.3 –Список полей и создание таблицы Users

* таблица «Students» (студенты) Рисунок 2.4,

Содержит 2 поля – (ID, Student\_name).  
*ID –* Также является идентификационным номером студента.  
*Student\_name-*Хранит в себе имя студента.

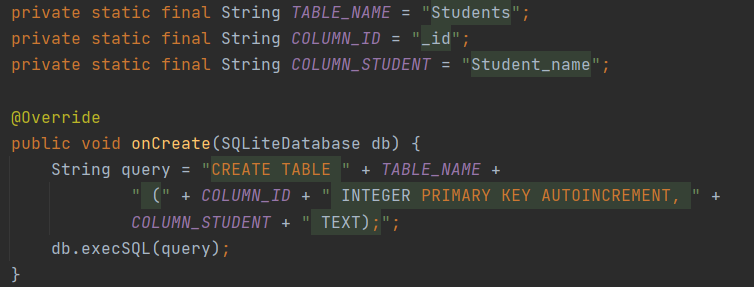


Рисунок 2.4 –Список полей и создание таблицы Students

* таблица «attendance» (посещаемость),

Содержит 4 поля – (ID, Student, Subject, Date) Рисунок 2.5.  
*ID –*Является идентификационным номером записи.  
*Student -*Хранит в себе имя студента.  
Subject – Название предмета.  
*Date* – Дата посещения студентом предмета.

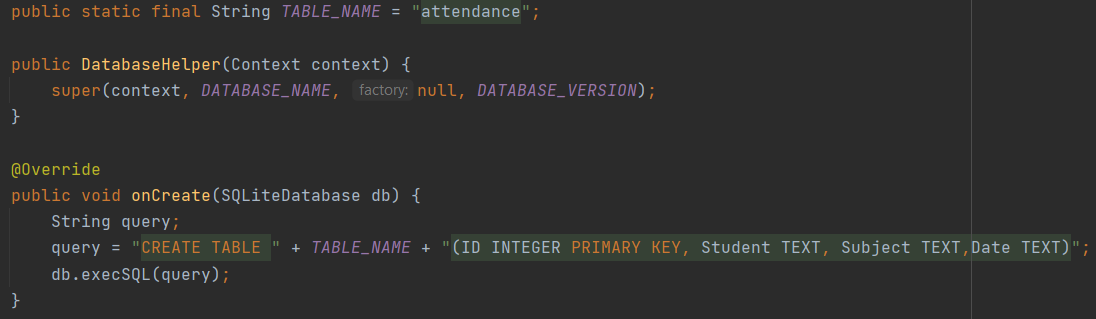


Рисунок 2.5 –Список полей и создание таблицы Attendance

В данном курсовом проекте пользователь может принадлежать к одной из следующих ролей: посетитель или администратор.

Посетитель может Просматривать лист студентов, а также их посещаемость, но не может взаимодействовать с данными.Также ему доступна регистрация (т.е., добавление записи в таблице Users через вызов функции Записи остальных таблиц доступны посетителю только для чтения.

Администратор, помимо вышеописанных операций, могут добавлять, удалять и редактировать список студентов, их посещаемость (все эти действия также осуществляются через функции).

## Разработка клиентской части

### Методы работы с бд.

Для разработки клиентской части курсового проекта (мобильного приложения) была использована среда разработки Android Studio, установка которой была рассмотрена ранее. Для работы с бд нам нужно разработать определённые методы. Так для получения данных с базы данных был создан метод ReadAllData() Рисунок 2.6.

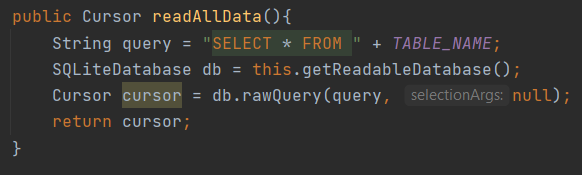


Рисунок 2.6 –Метод для получения всех данных дб таблицы TABLE\_NAME

Ниже, на рисунке 2.7, показан метод auth, который вызывается при попытке авторизации пользователя. На вход метод получает данные полей логина и пароля, затем происходит поиск пользователя и в случае если пользователь найден происходит сверка пароля на соответствие, в случае успешной регистрации в поле класса записывается имя пользователя и его уровень доступа в качестве идентификатора сессии, иначе вызывается всплывающее сообщее сообщающие об ошибке и возвращается 1.

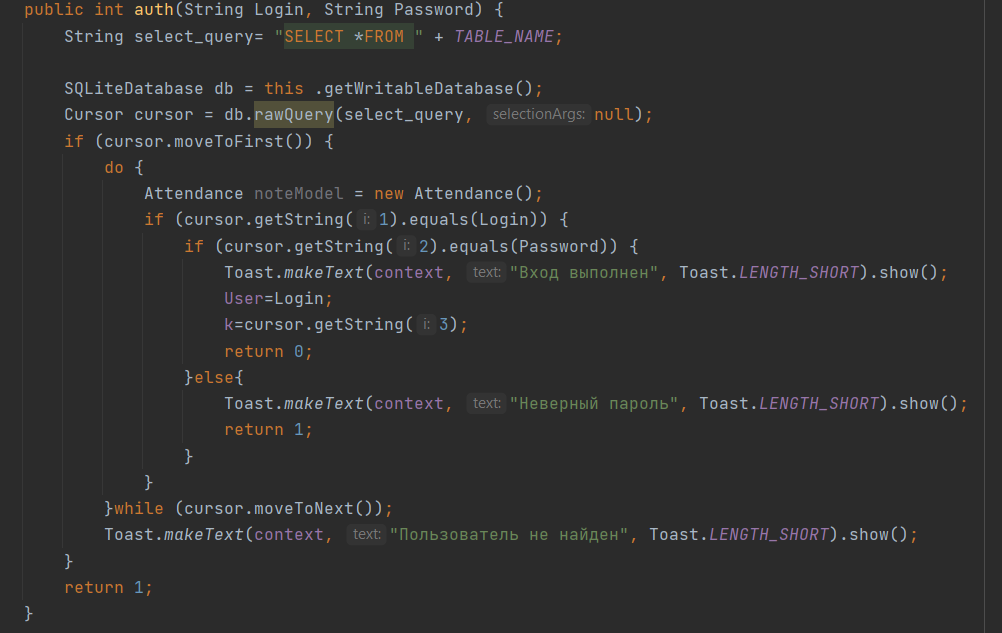


Рисунок 2.7 – Метод auth()

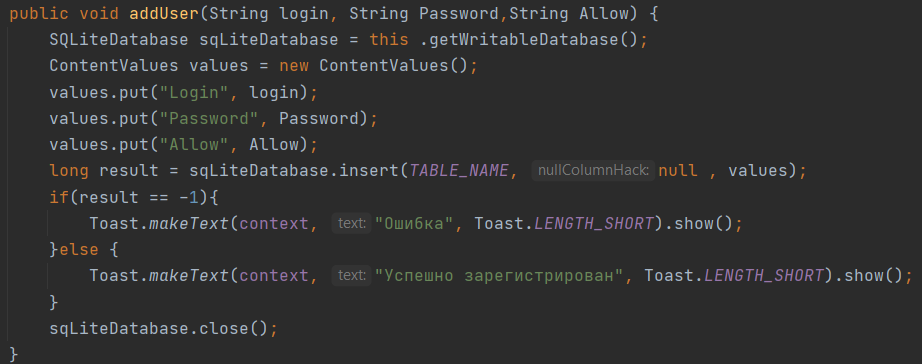
Также был создан метод addUser() Рисунок 2.8 с его помощью осуществляется запись пользователя в бд. Он принимает на вход логин пользователя, пароль, а также его ключ доступа ,а затем записывает данные в соответствующие поля таблицы. В случае если запись прошла успешно выводится соответствующее сообщение, иначе сообщение ошибки. 

Рисунок 2.8 – Метод addUser()

Но как вы уже могли заметит этот метод лишь осуществляет запись в бд, но не проверяет дынне полученных полей, и проферка всех этих данных проходит в методе onclick , что вызывается при срабатывание слушателя кнопки регистрации Рисунок 2.9.

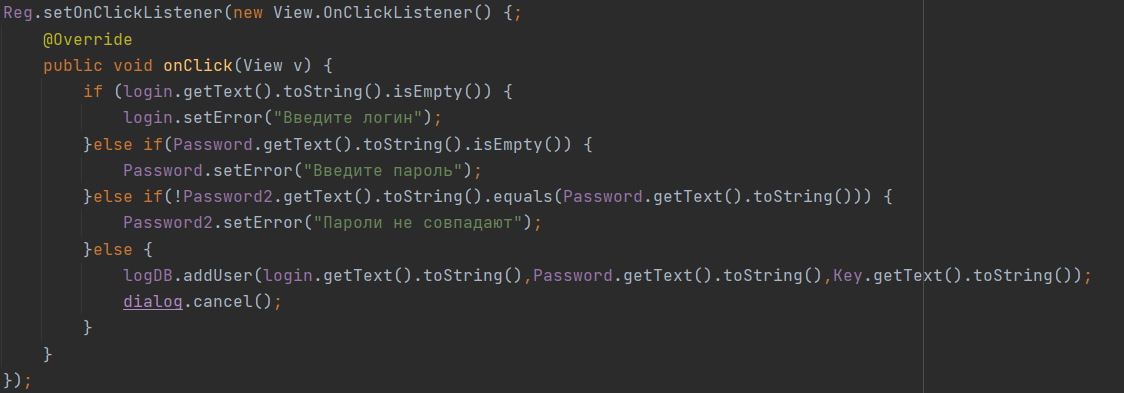


Рисунок 2.9 – Слушатель кнопки регистрации.

Аналогично, у нас существует метод для добавления записи студента и успеваемости Рисунок 2.10.

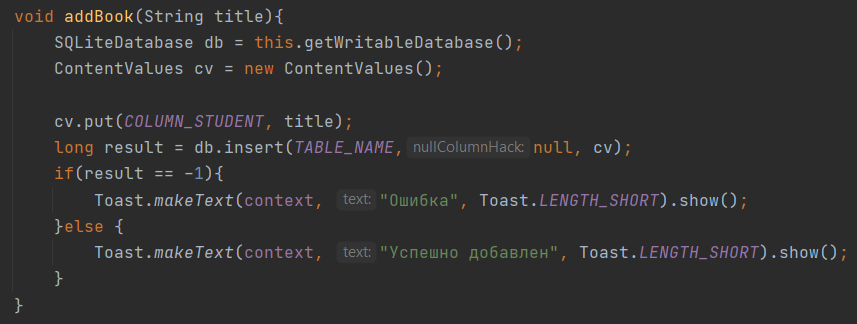


Рисунок 2.10 – Метод добавление записи студента.

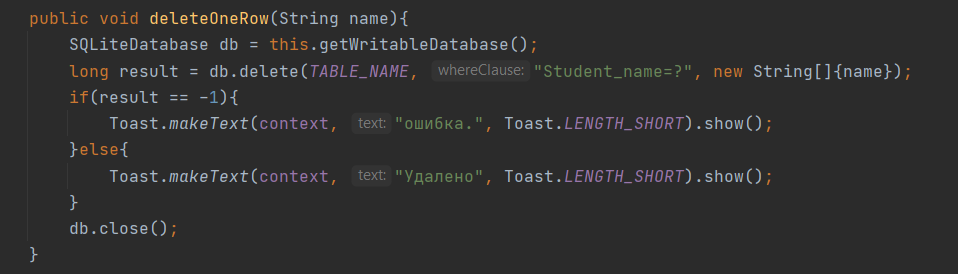
Но нам Нужно не только записывать данные, но и удалять, для этого был написан метод deleteOneRow Рисунок 2.11, что принимает на вход имя студента, которого мы хотим удалить. Первым делом запускается соеденение с базой данной, затем происходит удаление записи бд, что содежит имя полученного студента. После этого получаем сообщение сообщающее о результате удаления и закрываем соединение с базой данных.

Рисунок 2.11 – Метод добавление удаление записи.

### Графические формы приложения

Для взаимодействия с пользователем были созданы следующие графические формы или активности (activity):

* основная активность для отображения списка студентов, добавления студентов в этот список, а также для отображения унопки открытия диалогового окнарегистрации авторизации.
* активность для осуществления регистрации авторизации, включает в себя диалоговое окно.

Для оформления графических форм использовалась XML-размета, для реализации программной логики – язык Java.

### Основная активность

Ниже, на рисункe 2.12, показана начальная панель с кнопками регистраии и авторизации. А также эллементы активности, в том числе поле ввода имени студента, кнопка добавления студента и кнопка открытия диалогового окна.

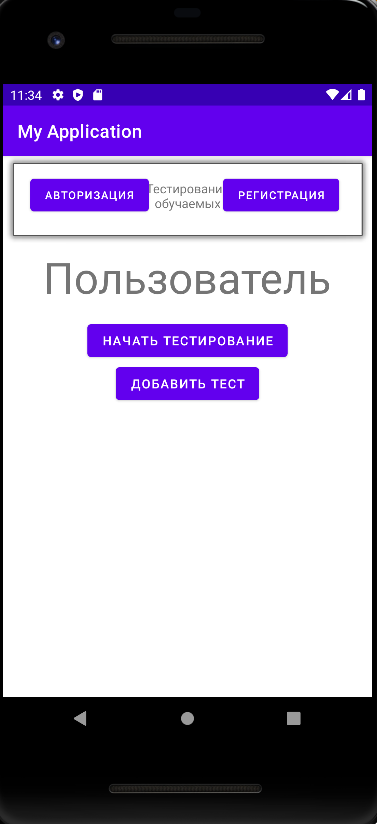


Рисунок 2.12 – Основная активность

При нажатиии на кнопку добавить происходит считывание данных поля студентов, а также данных поля сессии и проверка уровня доступа, если пользователь имеет права администратора происходит вызов метода Add Рисунок 2.13 для добавления студента.

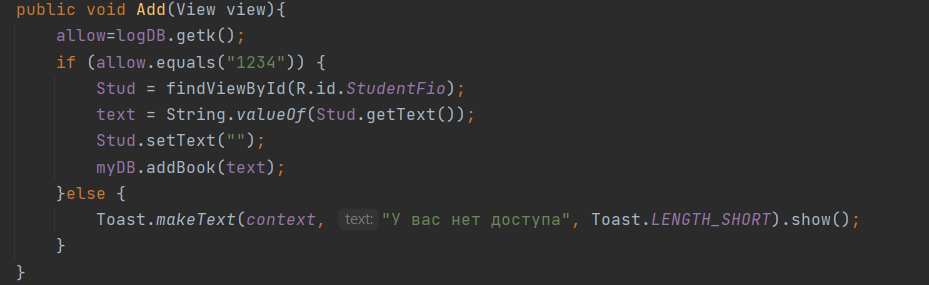


Рисунок 2.13 – Добавление студента

### Авторизация и регистрация

При нажатии на замочек происходит вызов диалогового окна и вызов диалогового окна активности авторизации Рисунок 2.15. Это осуществляется путём вызова метода showDialow Рисунок 2.14.



Рисунок 2.14 – showdialog.

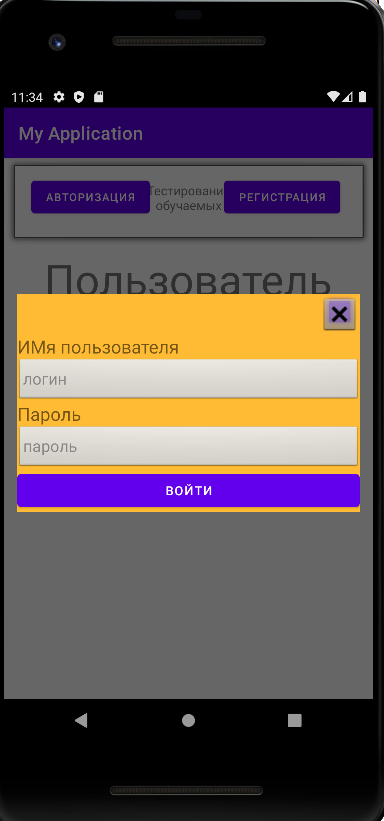
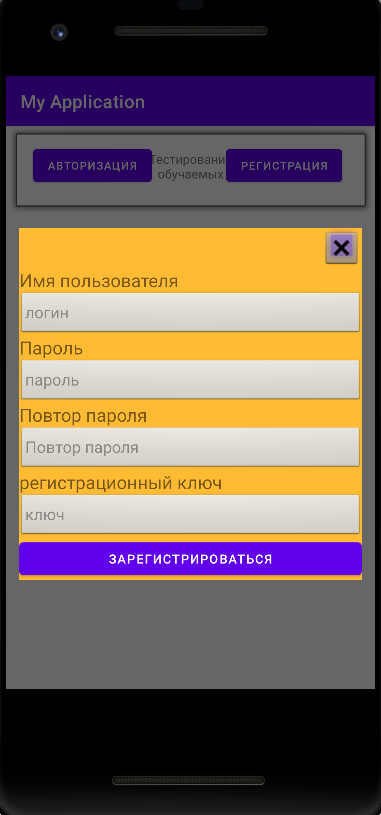


Рисунок 2.15 – Активность авторизации



2.15 Окно регистрации

# ВЫВОДЫ ПО РАБОТЕ

В ходе выполнения данного курсового проекта был разработан программный комплекс «Тестирование обучающихся. Мобильное приложение», который состоит из:

* клиентского мобильного приложения, написанного на языках Kotlin и Java с применением XML-разметки для оформления графических форм;
* базы данных mysqllite;

Пользователи системы поделены на две роли: обычные пользователи (или «посетители») и администраторы. Посетители могут только просматривать записи основных таблиц в то время, как администраторы могут вносить коррективы в данные, заносить новые и удалять ошибочную или неактуальную информацию.

Разграничение прав доступа осуществляется на стороне клиента, за отвечает класс базы данных который хранит логин текущего пользователя и уровень его прав доступа (обычный пользователь или администратор) в сессионной переменной, которая устанавливается при авторизации и сбрасывается при выходе из системы.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Синицын И.В. Конспект лекций по дисциплине «Разработка мобильных приложений» // РТУ МИРЭА
2. Android Studio // Android for Developers URL: <https://developer.android.com/studio>