Министерство науки и высшего образования Российской Федерации

федеральное государственное бюджетное образовательное учреждение высшего образования

«Российский экономический университет имени Г. В. Плеханова»

**МОСКОВСКИЙ ПРИБОРОСТРОИТЕЛЬНЫЙ ТЕХНИКУМ**

Специальность 09 02 07 Информационные системы и программирование

Квалификация: Программист

ПРАКТИЧЕСКАЯ РАБОТА

ПО «РАЗРАБОТКА МОБИЛЬНЫХ ПРИЛОЖЕНИЙ»

Страниц: 6

|  |  |
| --- | --- |
| Выполнил студент  Группы П50-6-21  Стукалов Илья Олегович | Проверил преподаватель  \_\_\_\_\_\_\_\_\_\_\_ К. А. Дзюба  «\_\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_ 2023 года |

**Тема:** разработка мобильного приложения.

**Цель работы:** создать приложение, состоящие из трёх экранов: экран авторизации, профиля и настроек.

**Ход работы:**

1. Создание проекта в Android Studio

Для начала создадим проект с пустым activity.

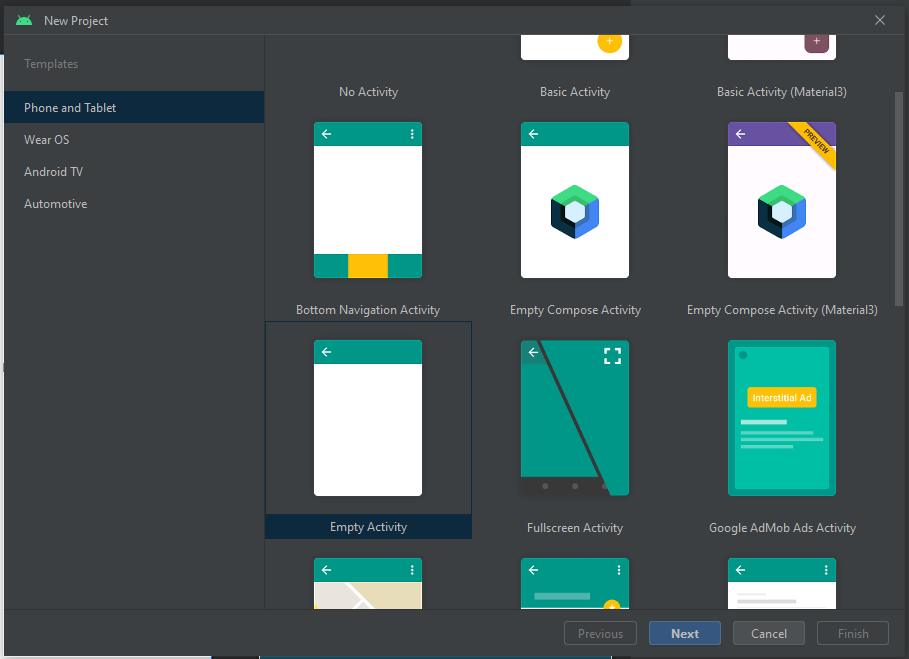


Рис. 1. Создание проекта

Настроим будущий проект. Придумаем название проекта, название пакета с исходниками, язык, место, где будет находиться проект, а также минимальный SDK.

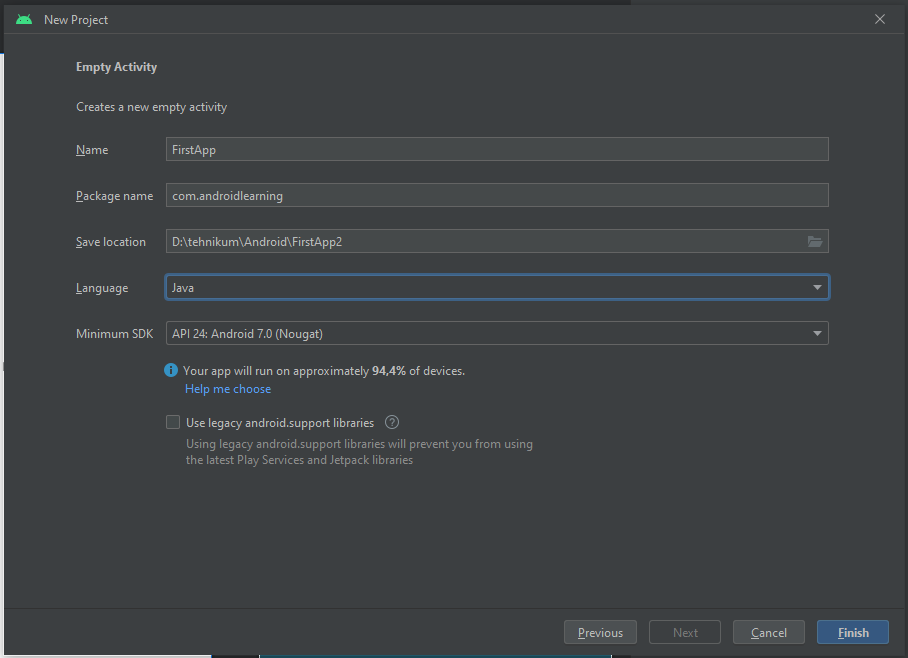


Рис. 2. Создание проекта

Далее создадим виртуальное устройство для тестирования приложения. В вертикальной вкладке справа откроем Device Manager.

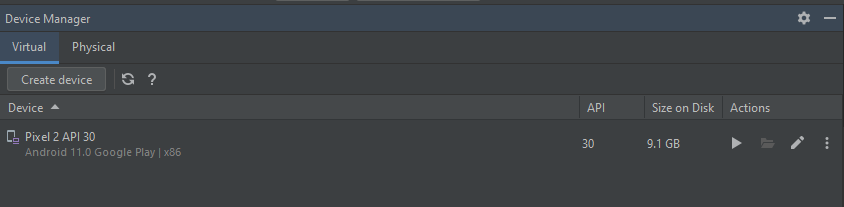


Рис. 3. Device manager

Далее нажмём кнопку Create device. Появится окно с выбором нового устройства. Выберем новое устройство и нажмём на кнопку Next

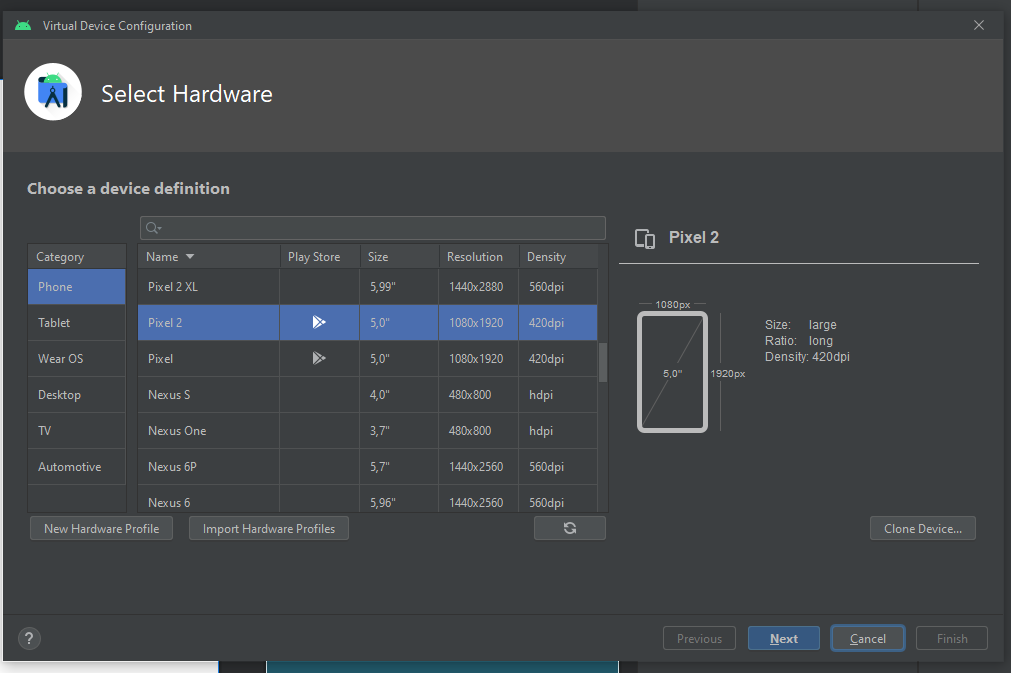


Рис. 4. Выбор устройства

Далее скачаем нужный SDK.

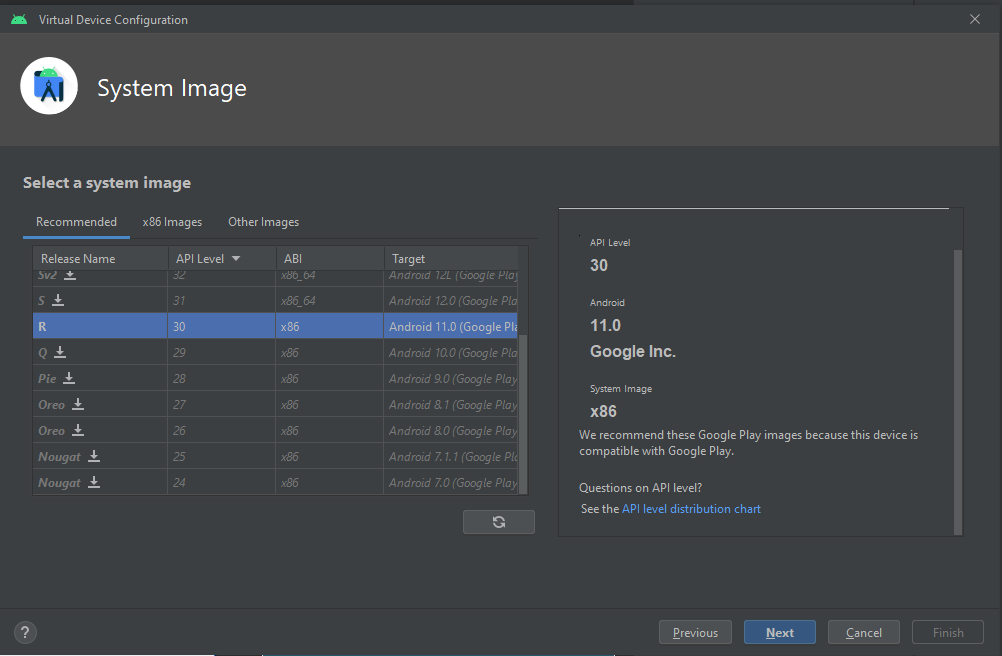


Рис. 5. Окно с SDK

Завершим настройку виртуального устройства. Дадим ему имя и выберем ориентацию.

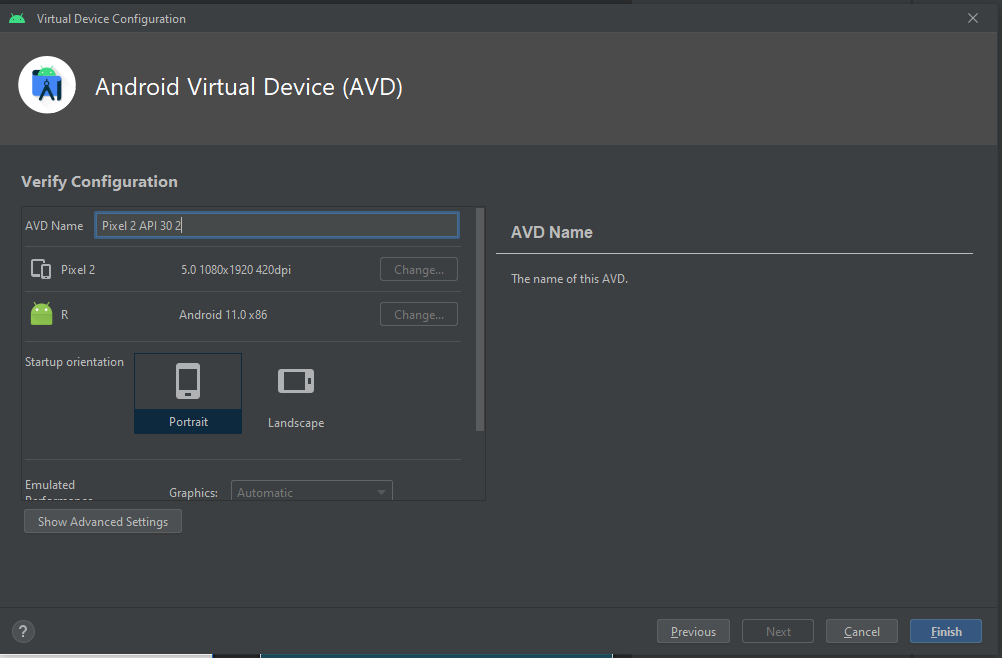


Рис. 6. Финальная настройка устройства

Переходим к созданию приложения.

1. Написание приложения

Для начала реализуем XML файл нашего main activity. Он будет появляться на старте приложения. Для ввода текста используем TextInputLayout, который позволяет установить ошибку, если введённый текст не попадает под какие-либо наши условия. Для вывода текста (нередактируемого) используем TextView. Для кнопок – Button

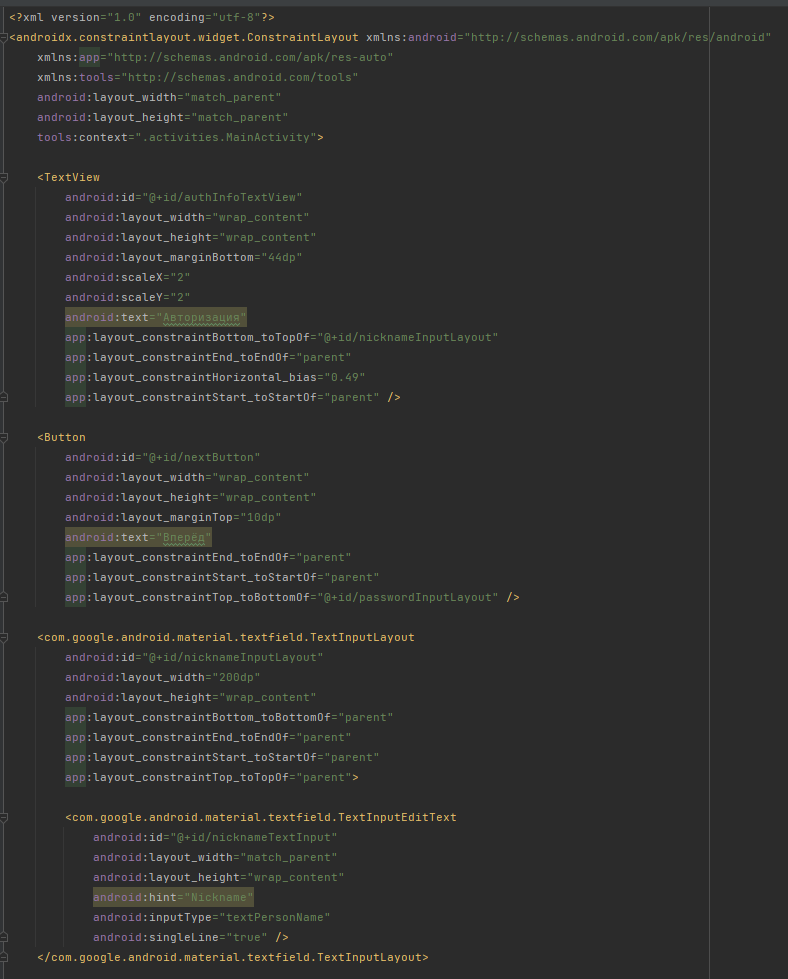


Рис. 1. XML файл main activity

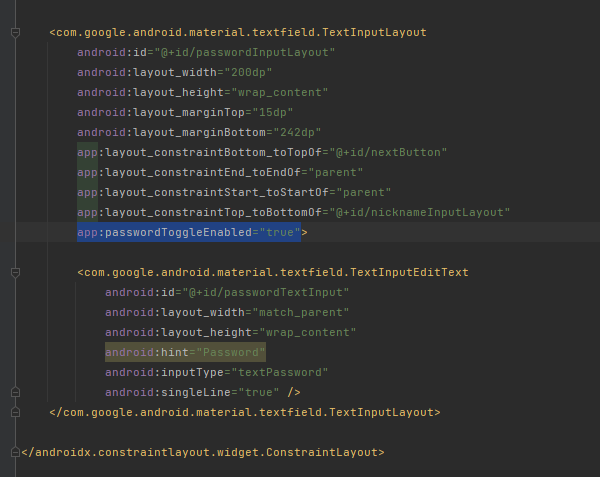


Рис. 2. XML файл main activity

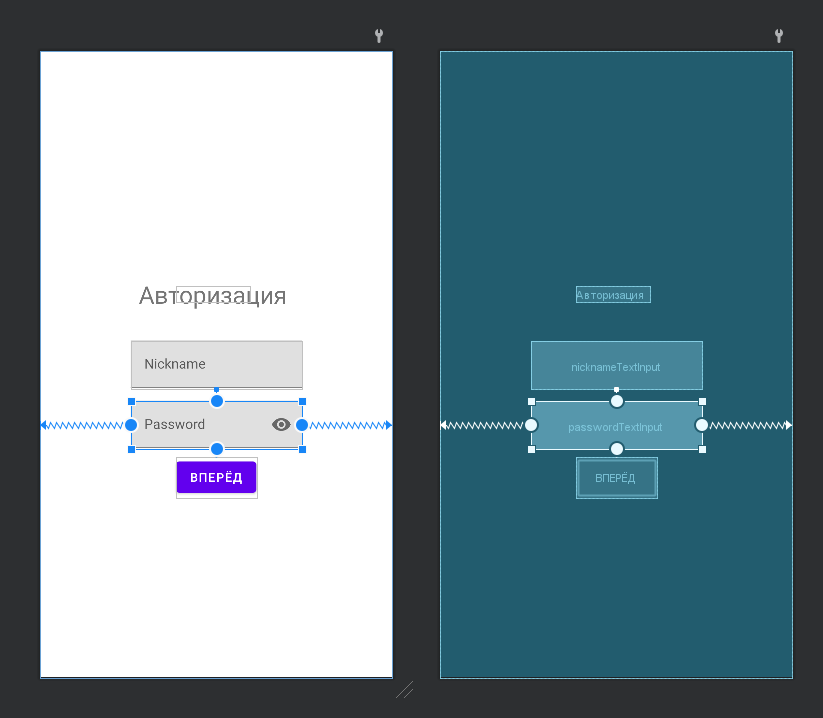


Рис. 3. Получившийся вид main activity

Теперь реализуем класс с утилитами. Там будет находиться паттерн строки для никнейма пользователя, а также небольшая утилита, которая позволит сократить код для проверки TextInputLayout на правильность текста и выполнить какое-либо действие при изменении текста.

****

Рис. 4. Утилиты

Затем реализуем простой класс пользователя, который будет содержать в себе никнейм, пароль, пользователя, а также геттеры и сеттеры для них.

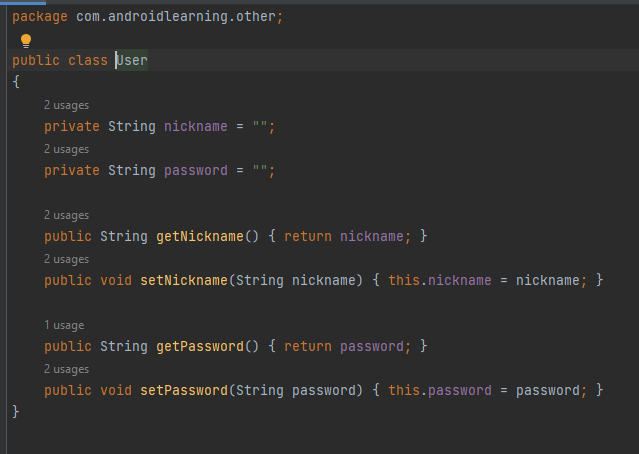


Рис. 5. Класс пользователя

Далее реализуем класс для логики main activity. Установим listener на кнопку для того, чтобы при нажатии на неё (если активна) нас перекидывало на следующий activity (профиль). Кнопка будет активна, если пароль и логин введены с верным форматом. Формат для пароля – не пустая строка, формат для логина – не пустая строка и соответствие с паттерном логина (см. рис. 4).

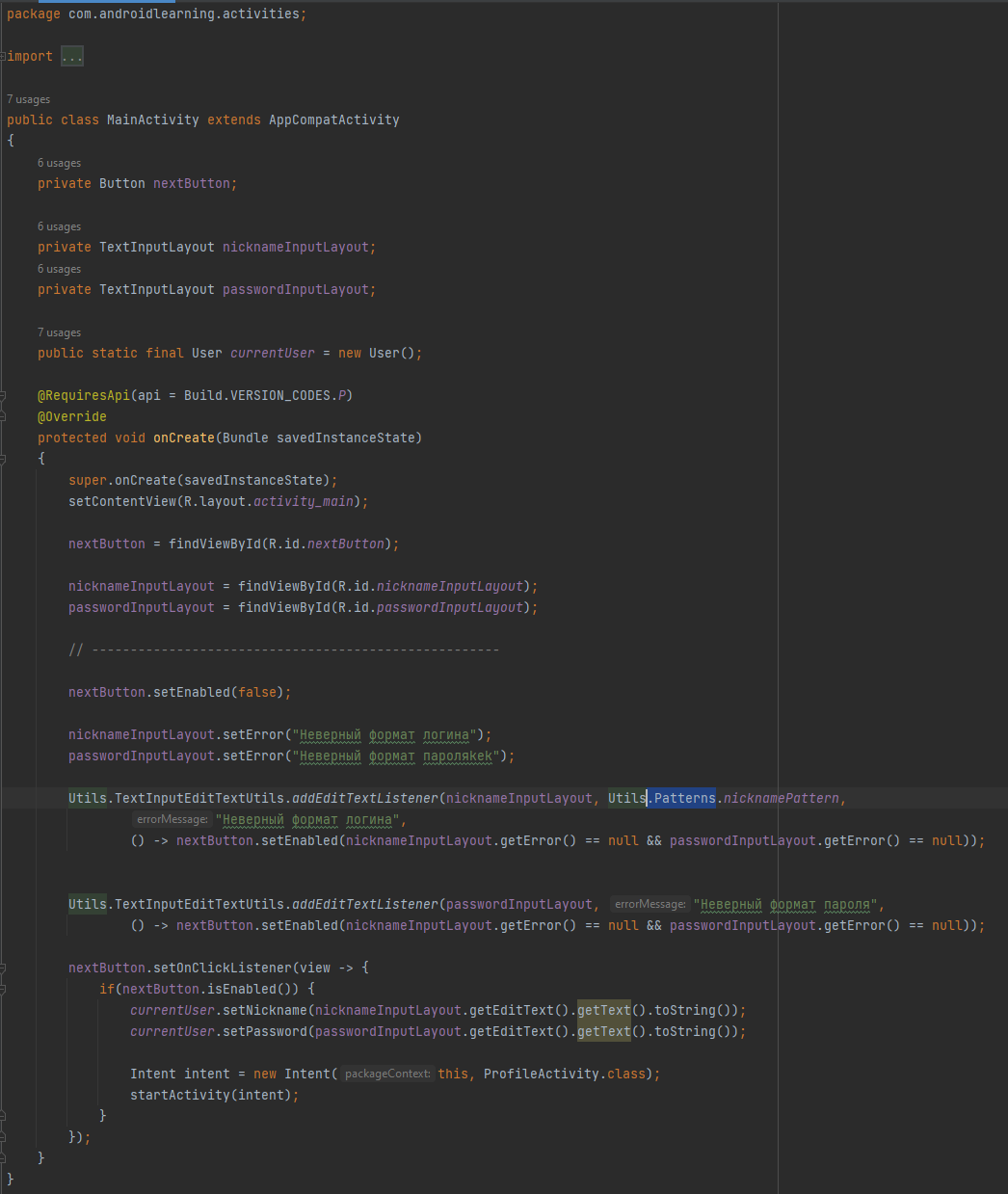


Рис. 6. Класс логики main activity

Далее реализуем XML файл для profile activity. Для отображения картинки используем ImageView. Остальные элементы мы уже использовали.

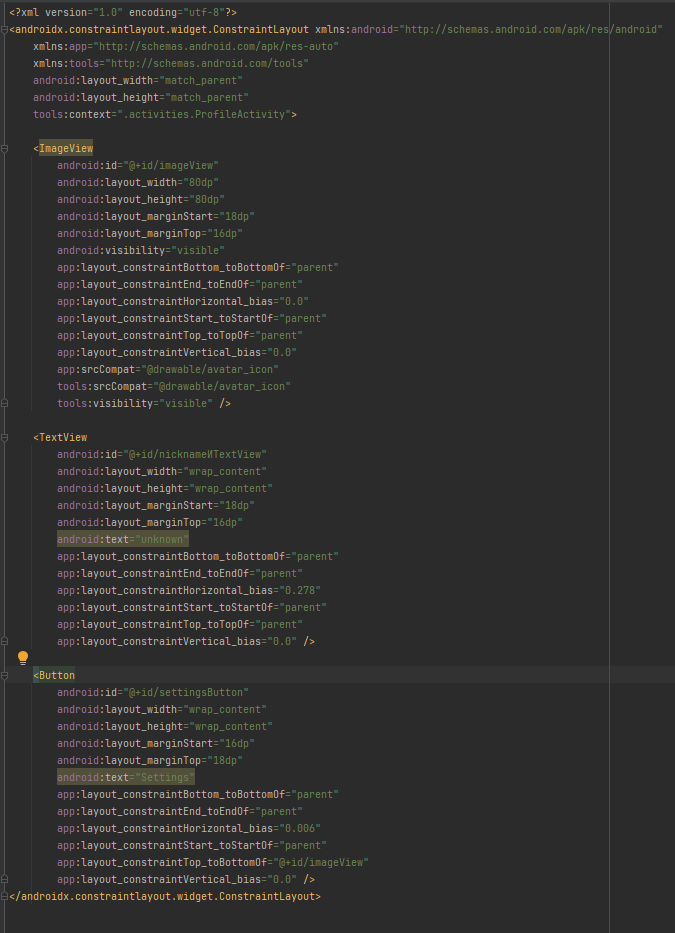


Рис. 7. XML файл для profile activity

Затем реализуем класс с логикой для profile activity. В нем просто будет обрабатываться нажатие на кнопку “Settings”. При нажатии нас перебрасывает в settings activity.

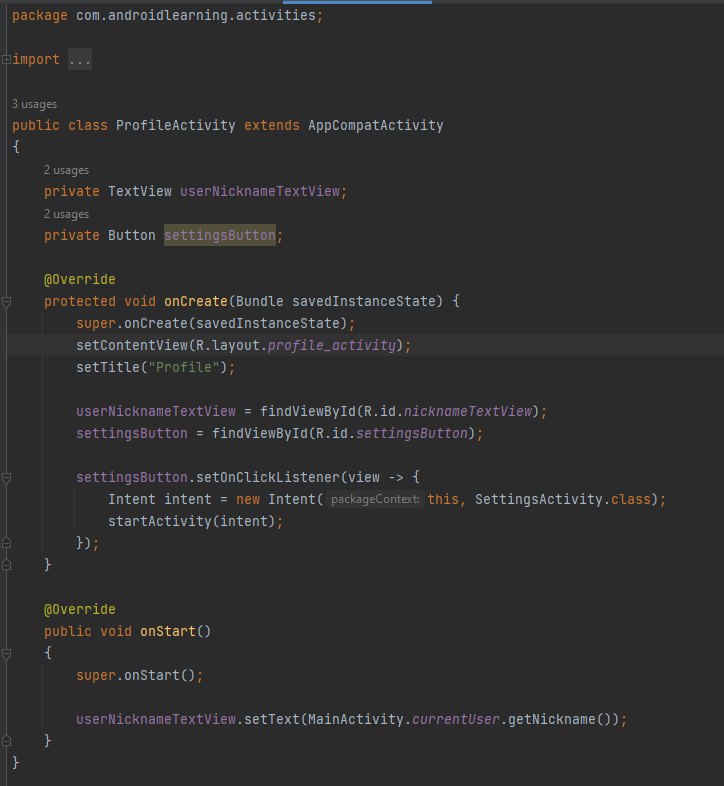


Рис. 8. Класс логики profile activity

Далее реализуем XML файл settings activity. В данном activity пользователь сможет изменить свой никнейм и пароль.

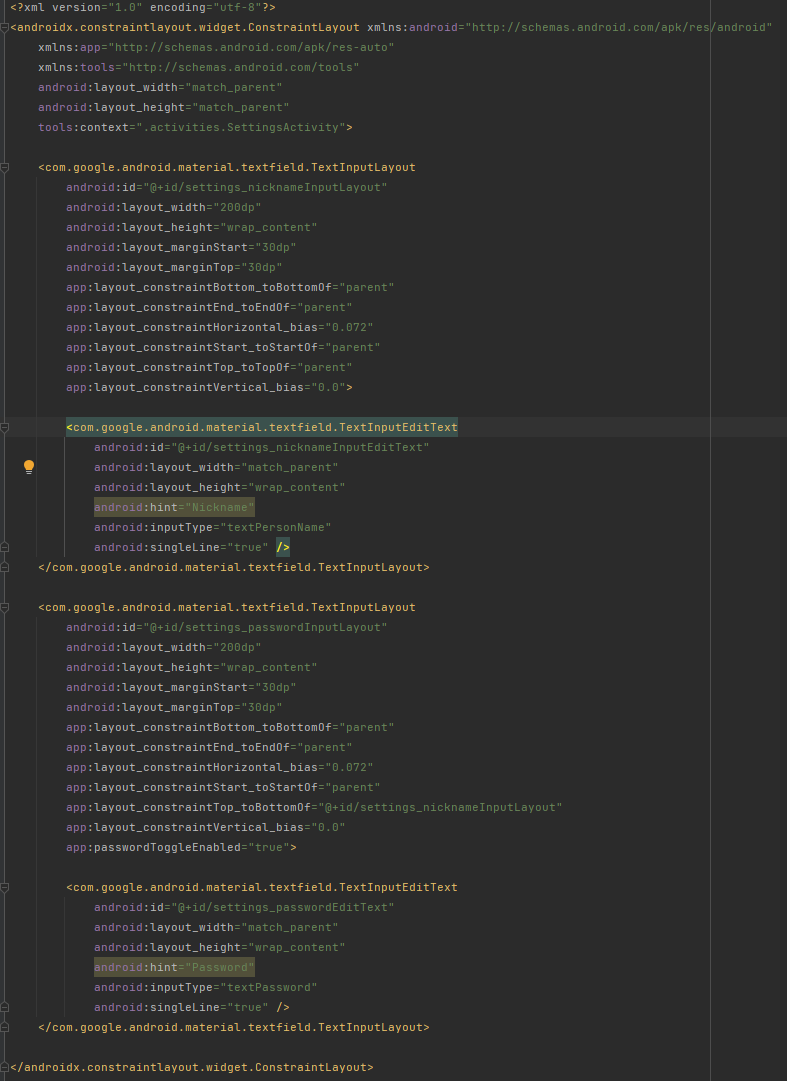


Рис. 9. XML файл для settings activity

Далее реализуем класс с логикой для settings activity.

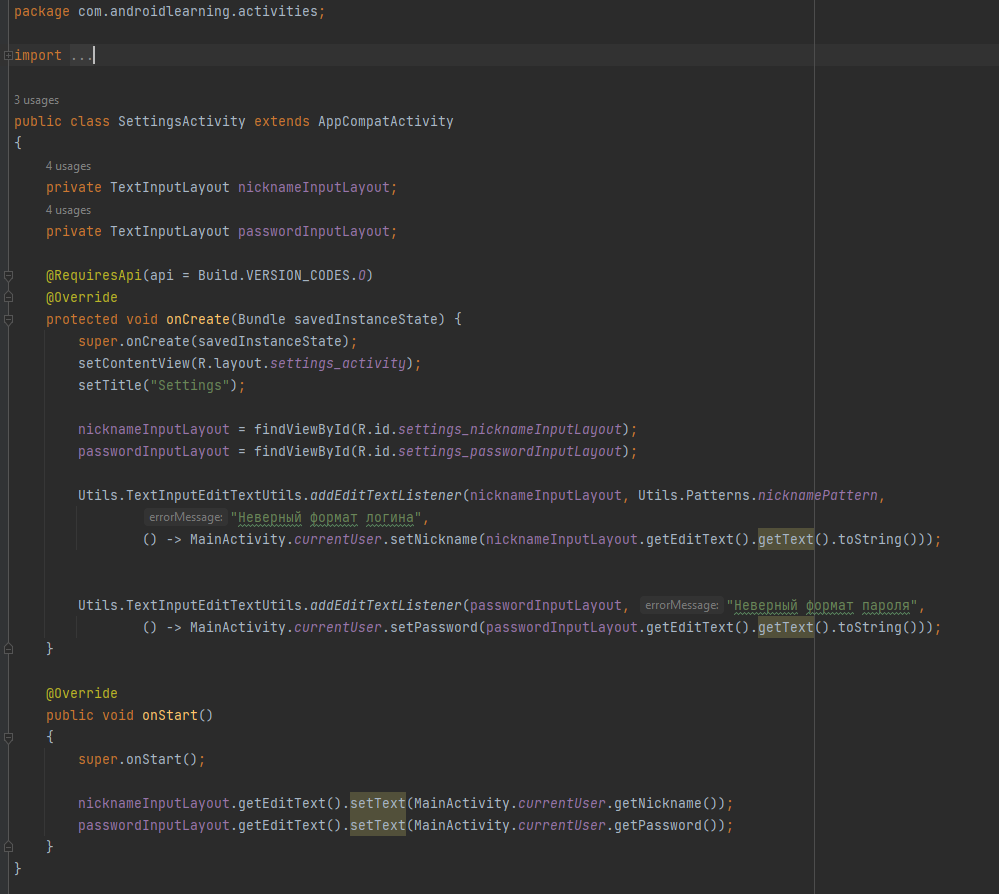


Рис. 10. Класс логики settings activity

Результаты:

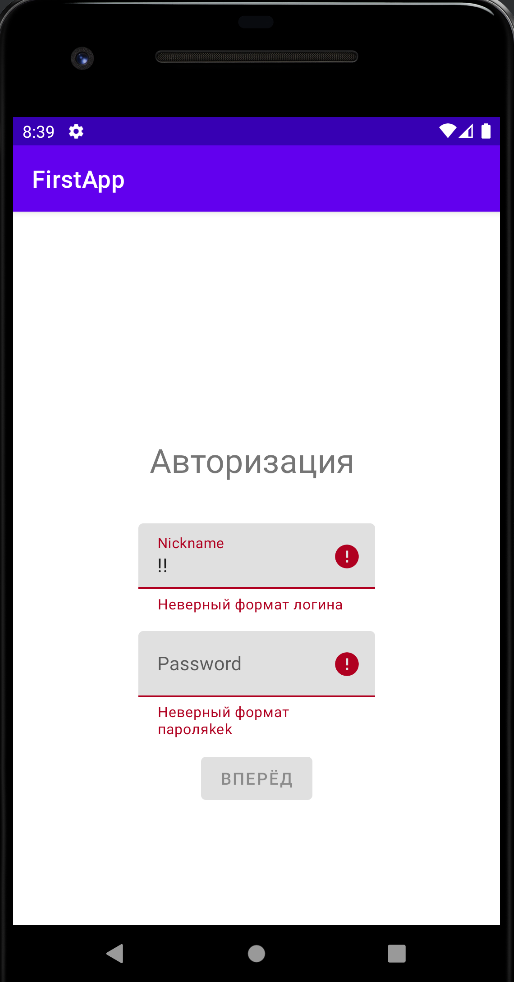


Рис. 11. Результаты

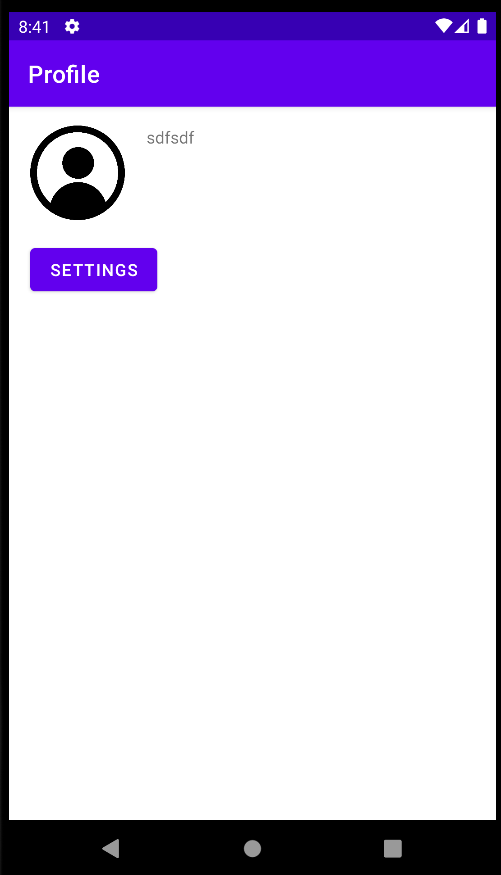


Рис. 12. Результаты

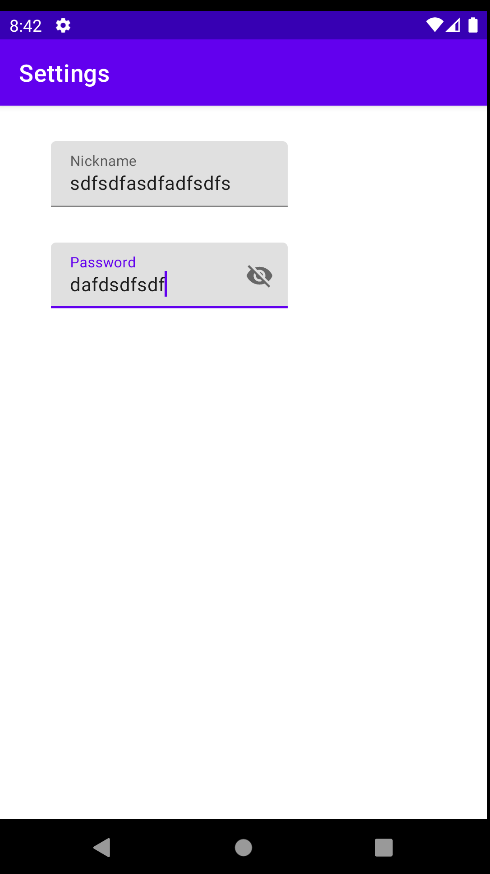


Рис. 13. Результаты

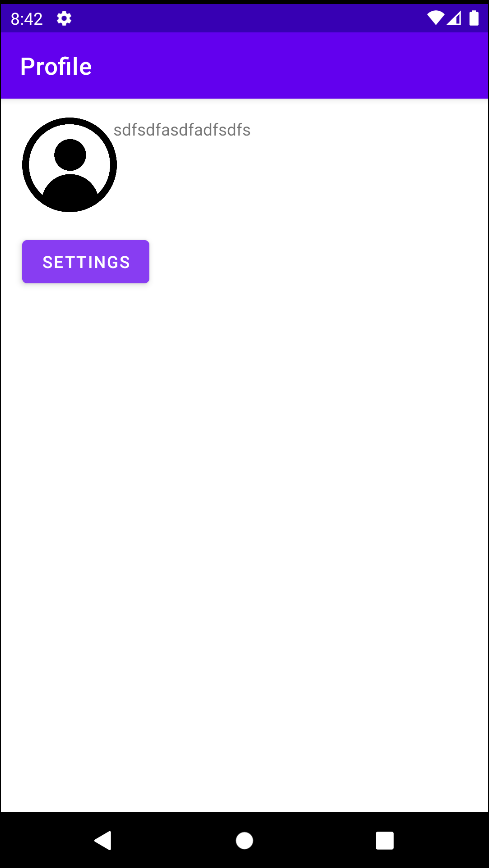


Рис. 14. Результаты

**Вывод**: создали приложение, состоящие из трёх экранов: экран авторизации, профиля и настроек.