

Курсова работа

По дисциплина “Програмиране на приложения за мобилни
устройства”

На тема: “Calculator with DRF”

Изготвил: Том Йорданов

Специалност: Софтуерно инженерство

Курс: III

Факултетен номер: 2101321049

Проверил:

Съдържание

1. Увод
2. Основни функционалности
3. Използвани технологии и библиотеки
4. Потребителски инструктаж
5. Архитектура на приложението
6. Заключение

1. Увод

Целта на курсовия проект е да се изгради мобилно приложение, с което могат да се извършват научни пресмятания, като се записва история на всички пресмятания.

2. Основни функционалности

- Пресмятане
- Записване на информацията в база данни
- Създаване на User account с CRUD операции
- Извеждане на най-често използваната операция ($*$, $/$, $+$, $-$)

3. Използвани технологии и библиотеки

○ *Frontend*

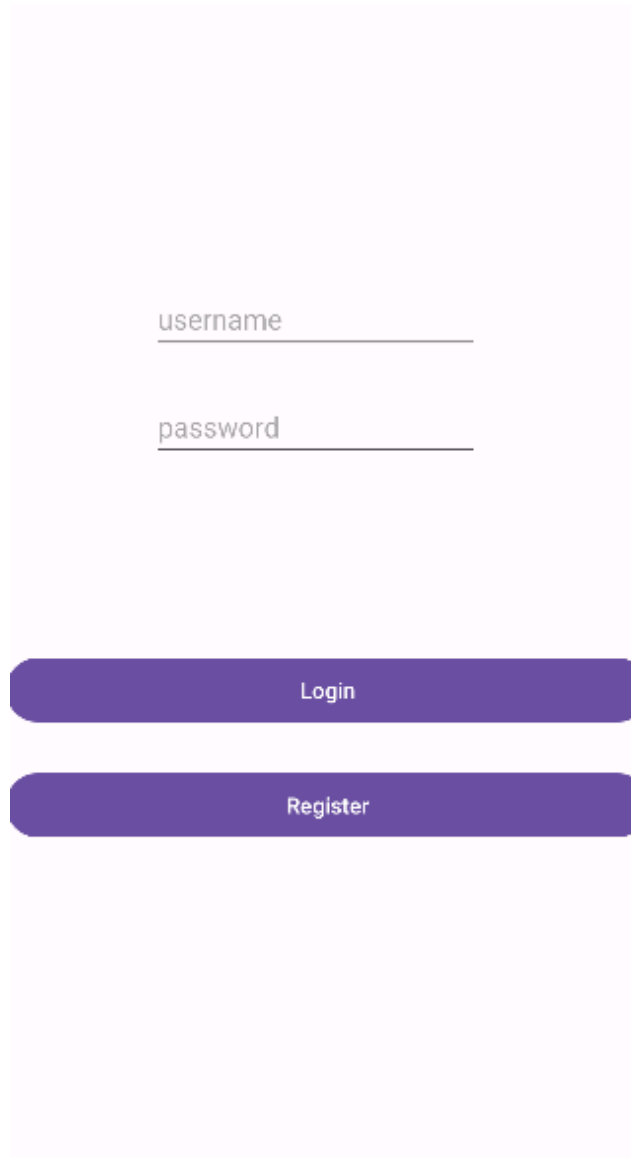
- Android Studio
- Java
- GSON - Десериализация на обекти
- SQLite - използван за съхранение на данните

○ *Backend*

- Visual Studio Code
- Python DRF - (Django Rest Framework)

4. Потребителски инструктаж

Login Screen



A login screen mockup with a light pink background. It features two input fields for 'username' and 'password', each with a horizontal line below the text. Below the input fields are two purple buttons with rounded corners. The first button is labeled 'Login' and the second button is labeled 'Register'.

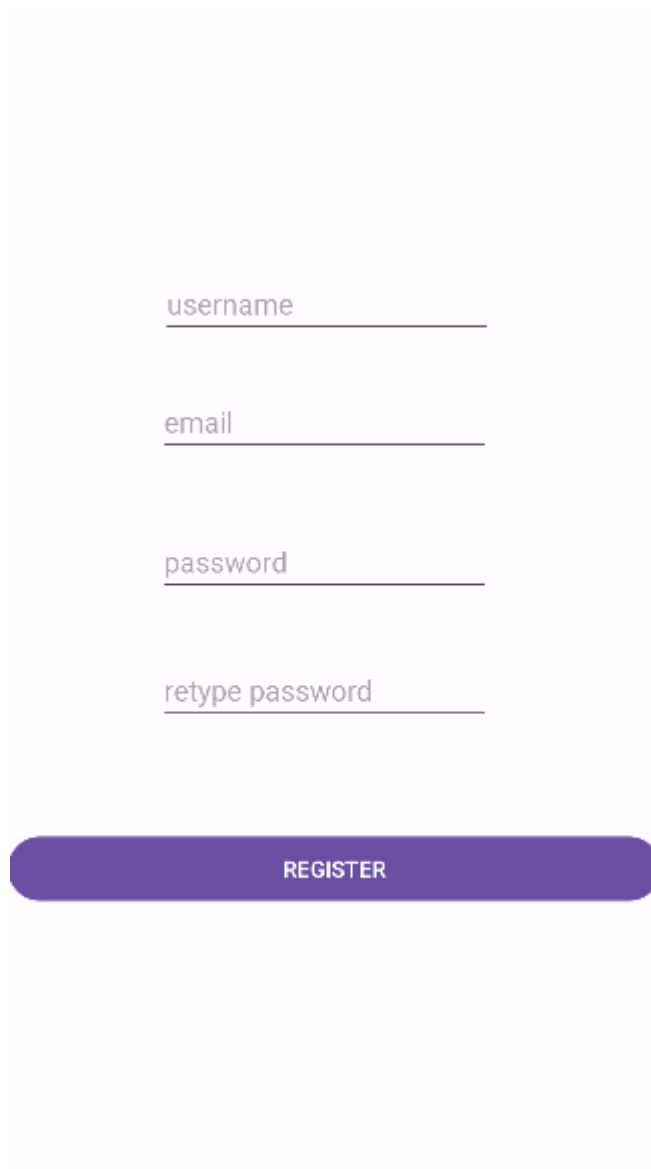
username

password

Login

Register

Register Screen



A vertical registration form on a light pink background. It features four input fields with placeholder text: 'username', 'email', 'password', and 'retype password'. Each field is a thin horizontal line. Below the fields is a purple rounded rectangular button with the text 'REGISTER' in white capital letters.

username

email

password

retype password

REGISTER

При стартиране на приложението ще бъде нужно да се впишете / регистрирате, след успешно влизане в профил, ще Ви се визуализира Main Screen, от който ще бъде нужно да изберете желана от Вас опция.

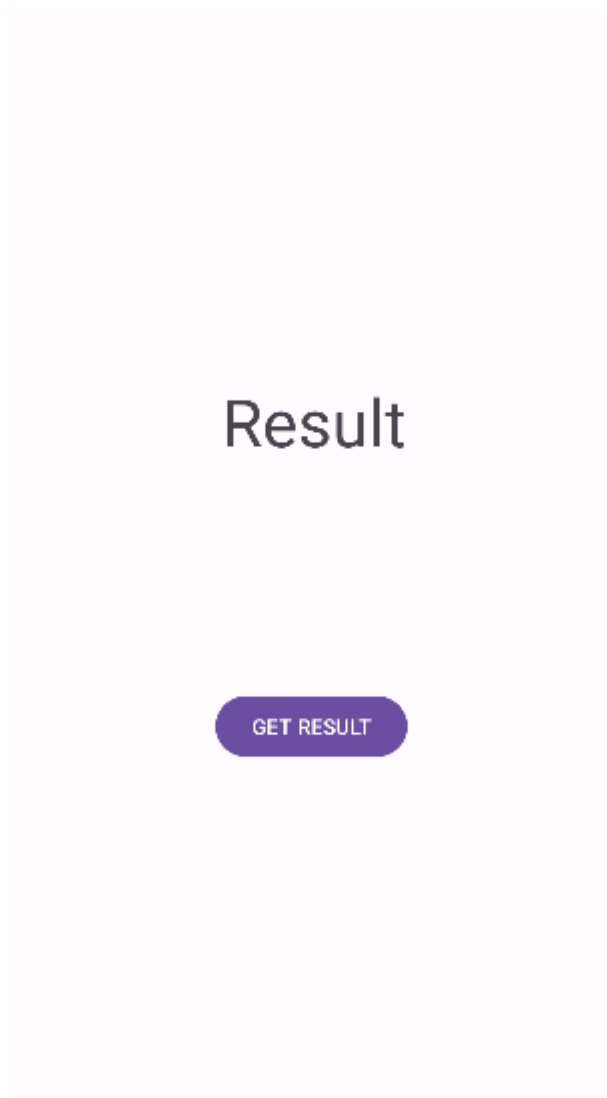
Main Screen

ARCHIMEDES

frequency

Има два варианта за избиране на функционалност. Бутонът 'ARCHIMEDES' ни води към пресмятания, а бутонът 'frequency' извежда най-често използваната операция - $*$, $/$, $+$, $-$.

Frequency Screen



При натискане на бутона 'Get Result' ще ви се визуализира най-често използваната операция, тук се използва GET заявка.

Calculate Screen

420

/

69

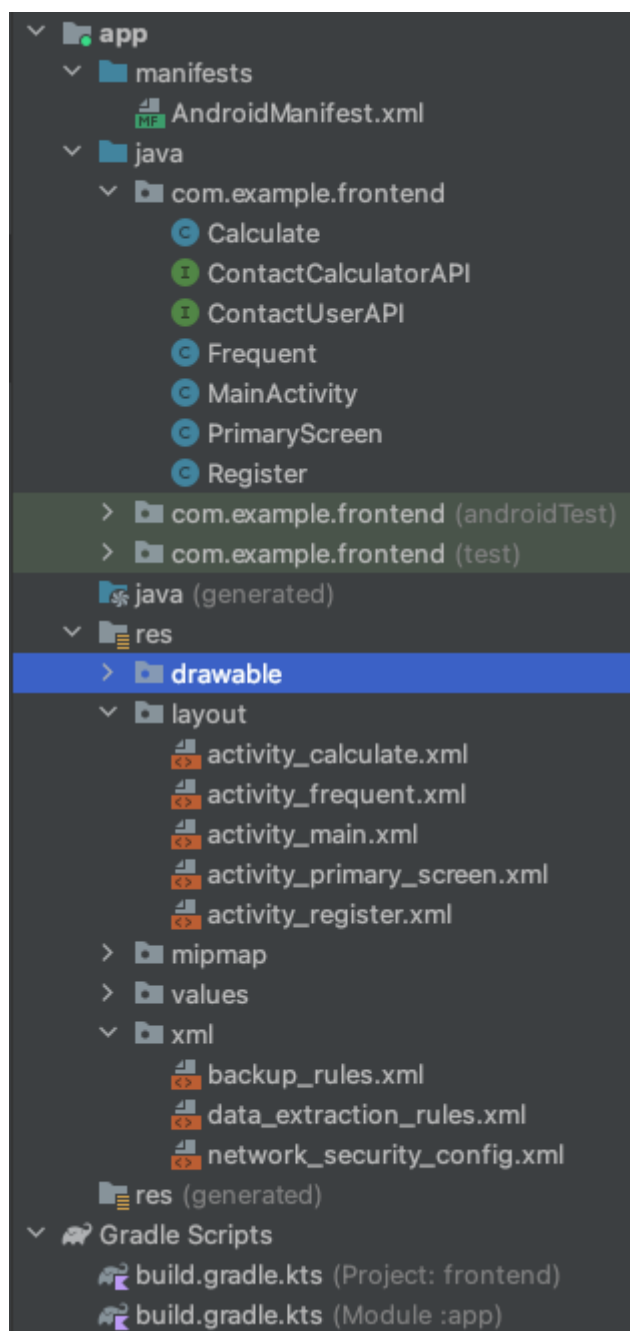
result

CALCULATE

Тук е нужно да се въведат две числа и оператор, възможните оператори са “*”, “/”, “+”, “-”.

При натискане на бутона 'CALCULATE' се изпълняват две заявки, едната е POST, а другата GET, идеята е следната, първо се взима въведената информация и се създава нов запис в базата данни, API-то прави пресмятане и попълва полето 'result', ако тази операция е успешна се изпълнява GET заявка, която визуализира резултатът в полето 'result'.

5. Архитектура на приложението



- **manifests [AndroidManifest.xml]** - съдържа metadata - та за приложението
- **com.example.frontend** [Calculate, Frequent, MainActivity, PrimaryScreen, Register] - Java класове, съдържат бизнес логиката на приложението

- **com.example.frontend** [ContactCalculatorAPI, ContactUserAPI] - Java interfaces, които са необходими за комуникация с DRF - Django Rest Framework
- **layout** [all] - съдържа дизайнът на приложението
- **xml** [network_security_config.xml] - разрешени ip адреси, към които мобилното приложение може да изпрати заявки
- **Gradle Scripts** - съдържа нужните ни библиотеки или казано dependencies

6. Заключение

Курсовият проект даде добра основа за разработване на сложно мобилно приложение, към което има users и работа с БД