2) Нефункционални изисквания

III. Реализация

1)

А) Изисквания - функционални/нефункционални

Б) Диаграма на класовете

В) Блок-схема на действията

4) Реализация

5) Потребителско ръководство

6) Заключение

7) Източници на информация

8) Приложение

В) Front-End (JavaScript - превод на Java за Android и Objective-C за iOS):

- Разработка на Cross-Platform приложението чрез React Native

- Навигация между екраните на приложението чрез библиотеката React Navigation

- Операции с определени жестове чрез React Native Gesture Handler

- Конвертиране на String в аудио чрез React Native TTS (Text-to-Speech)

- Използване на npm за сваляне и обновяване на JS библиотеки

- Използване на асинхронни заявки (AJAX) за комуникация със сървъра

- Използване на promises за постигане на асинхронно програмиране

- Имплементация на component-based architecture

- Използване на готови React Native компоненти от RN Elements

- Използване на ES6 модули за по-добро разпределение на кода и прилагане на Single Responsibility Principle

- Използване на Async Storage за запазване на информация в паметта на мобилното устройство

- Използване на gradlew clean за поддръжка на Android кода

- Използване на SDK Manager за изтриване, обновяване и инсталиране на пакети за Android SDK

- Използване на state от React за запазване на състоянието на приложението и неговото управление

- Използване на новия синтаксис на ES6 на JS (REST, Spread Operators, Lambda Expressions)

- Версия на Andorid 5.0 +

Потребителски истории:

|  |  |  |  |
| --- | --- | --- | --- |
| № | като | искам | за да |
| 1 | система (сървър) | да получавам заявките с JSON Web Token в Header частта | мога да автентикирам потребителя |
| 2 | потребител | да получавам списък с български автори с техните имена и снимки | мога да отворя допълнителна информация за всеки един от тях |
| 3 | потребител | да мога да филтрирам авторите по техните имена | мога по-лесно да намирам конкретен автор |
| 4 | потребител | да правя тестове върху биография и творчество | мога да си затвърдя знанията |
| 5 | потребител | да ми се визуализира статистика на моя прогрес на базата на правените от мен тестове | следя своя напредък |
| 6 | потребител | да ми се визуализира ранглиста | мога да се сравня с другите играчи |
| 7 | потребител | със задържане на екрана приложението да изговаря текстовете | мога да усвоя материала по-пълноценно |
| 8 | система | да държа паролите на потребителите криптирани | за да гарантирам сигурност на личните им данни |
| 9 | потребител | чрез гласово търсене да извличам информация | за да си осигуря по-приятно потребителско изживяване |
| 10 | потребител | приложението да ме помни | за да не се налага да влизам всеки път |

- Workflow диаграма

III. Реализация

1) Кратки фрагменти код (практически реализации)

2) Структура на БД - PG Admin ER диаграма

3) бизнес процеси - код

4) UI - контроли, валидация...

IV. Потребителско ръководство (резултати):

- взаимодействие на потребителя с продукти - преходи и резултати

V. Заключение - Невенчо

VII. Източници на информация

VI. Приложение - по-важни части от кода автентикация и навигация - GitHub Link