A close up of a logo

Description automatically generated

**Факултет по компютърни системи и технологии**

**Проект по Програмиране за мобилни устройства**

**Тема: Българска литература**

Изготвили:

1. Валентин Георгиев Александров, гр. 46, ф.н. 121216074

2. Елена Венцеславова Антонова, гр. 50, ф.н. 121216165

Научен ръководител: доц. д-р. инж. Антония Ташева

1. Използвани технологии

1) Spring Framework е технологична рамка (framework) с отворен код (open source) за Java платформата. Предоставя много функции, улесняващи разработката на Java-базирани enterprise приложения. В настоящия проект е наблегнато на следните му модули:

- Spring MVC (Model-View-Controller) е архитектурен модел за изграждане на потребителски интерфейси. Разделя дадено софтуерно приложение на три взаимносвързани части, така че да се отделят вътрешни представяния на информация от начините, по които тя се представя на потребителя - контролери, изгледи, модели:

- Model - улавя поведението на приложението по отношение на своя домейн, независимо от потребителския интерфейс. Директно управлява данните, логиката и правилата на приложението.

- View - може да бъде всеки изход за представяне на информация, например чрез графика или диаграма.

- Controller - приема вход от потребителя и го конвертира в команди за модела или изгледа.

2) Hibernate

Stuffs:

1) Защо избрахме да правим това приложение?

В днешно време, достъпът до каквато и да е информация е по-голям от всякога, което се отразява на всеки аспект от живота ни, в случая начина, по който учениците в основното училище и гимназията усвояват нови уроци. Една основна част от образованието ни е познанието върху българския език и литература. Ние сме се съсредоточили върху разнообразното творчество на различни български писатели, с цел да повишим интереса както на ученици, така и на хора с увреждания към богатото наследство на родната литература.

2) Какво има на пазара в момента?

След дълго проучване, установихме, че има приложения с аналогична цел - опознаване на българското наследство и култура под формата на конкурентна игра. Но те по-скоро са предназначени за обогатяване на българската обща култура като цяло - подбрани са разнообразни въпроси, най-вече върху историята и география на България. Най-известното приложение у нас с такава цел е Triviador.

3) С какво се отличаваме?

Стигнахме до заключението, че макар да са налични подобни приложения, няма такова, което конкретно се обвързва с точните знания, необходими в училище, особено в областта на литературата. Към днешна дата, младите хора учат и утвърждават нова информация бързо и за кратко време, поради динамичното темпо на развитие, с което светът се движи. Следователно, представянето на тази информация по кратък, но ясен начин, изискващ непрестанното ѝ утвърждаване, би се увенчало с голям успех.

Също така, забелязахме, че много малко продукти са насочени съм хора с увреждания, главно незрящи. Стремим се да направим своето приложение възможно най-достъпно, затова сме добавили допълнителни функционалности, спомагащи за тази цел.

4) Предимства и недостатъци на други приложения от потребителска гледна точка

Обобщили сме следните предимства и недостатъци на други приложения със сходно предназначение:

А) Предимства:

- разнообразен и увлекателен подход на представяне на различни факти и въпроси към тях

- точки за всеки верен отговор, които могат да се инвестират в изграждане на собствен виртуален свят - село, град или цяла империя, например

- в някои приложения времето за отговаряне на въпросите е ограничено, стимулирайки по-бързо мислене

Б) Недостатъци:

- несъобразителност с версията на мобилното устройство, т.е. приложението може да е „тежко“, което води до значително забавяне на заявките

- липса на източници на информация, върху която се задават въпросите

- много силна зависимост от интернет - най-честото оплакване е честопрекъсващата връзка към сървъра

Проектиране:

1) Функционални изисквания:

|  |  |
| --- | --- |
| Бутон (пренасочване) | Дестинация |
| 1. „Вход“ | Страница с главни опции - „Автори“, „Тестове“, „Потребител“ |
| 2. „Автори“ | Списък с всички автори |
| 3. Име на автора | Страница с допълнителни препратки за конкретен автор |
| 4. „Биография“ | Кратка биография на автора |
| 5. „Творби“ | Списък с творбите на автора |
| 6. Заглавие на творба | Кратък анализ на творбата |
| 7. „Мотиви“ | Списък с главните мотиви на автора, всеки с кратко описание |
| 8. „Тестове“ | Опции за тип тест - върху биографии, творби или конкретен автор |
| 9. „Биографии“ | Списък с тестове върху различни биографии |
| 10. „Творби“ | Списък с тестове върху различни творби |
| 11. „Автори“ | Списък с тестове върху определен автор |
| 12. „Потребител“ | Страница с бутони за допълнителна информация за потребителя: „Статистики“, „Ранглисти“ и опция за изход |
| 13. „Статистики“ | Страница с опции за три статистики - всяка върху резултатите от различните типове тестове |
| 14. „Ранглисти“ | Страница с опции за три ранглисти - всеки върху резултатите от различните типове тестове |
| 15. „Изход“ | Връщане към началната страница |
| 16. Задържане на екрана | Активиране на „Text-to-Speech“ функционалност |

2) Нефункционални изисквания:

- Java

А) Организация чрез Spring MVC архитектура

Б) Комбинация на различни технологии и сигурност чрез Spring Data

В) JPA - работа с repository за лесна реализация на CRUD операции (Create-Read-Update-Delete)

Г) Контрол над автентикацията и авторизацията на клиента чрез Spring Security

Д) Трансформация на ООП класове (entities) в таблици на релационна БД чрез Hibernate

- БД:

Е) Структуриране на базата данни чрез PostgreSQL (система за управление на бази данни)

Ж) PG Admin - инструмент за управление на БД

- JS - превод в Objective-C за iOS и в Java за Android чрез React Native

З) React Native - framework за разработка на cross-platform мобилни приложения

И) React Navigation - библиотека за навигация между екраните на мобилното приложение

Й) React Native Gesture Handler - навигацията зависи от нея + операции с жестове

К) React Text-to-Speech - конвертиране на String в аудио

Входни данни - за потребителя и резултати от тестове

3) Workflow Diagram:

- Кой ще използва продукта?

- Какви данни ще се използват?

- Как ще бъдат достъпени функционалностите от потребителя?

4) ER Diagram - шарено

Реализация:

1) PG Admin ER Diagram - PG Admin

2) Screenshots (потребителско ръководство - как се използва)

3) Some Classes - controllers, repositories..

4) GitHub Project Repository Link

Заключение? - Невенчо

Изочници на информация за:

1) БД

2) Технологии (React (Native, Navigation), Spring (Data) Documentations, PostgreSQL, JPA, Hibernate)

Важни моменти от кода. - автентикация и навигация