Министерство образования Республики Беларусь

Учреждение образования

Белорусский государственный университет информатики и радиоэлектроники

Факультет компьютерных систем и сетей

Кафедра информатики

Техническое задание к проекту на тему

**Приложение сервис по аренде бытовых вещей**

|  |  |  |
| --- | --- | --- |
| Студент |  | Скочинский М.В. |
| Руководитель |  | Гриценко Н. Ю. |

Минск 2022

# Техническое задание

## **1. Общие сведения о предметной области**

Приложение должно позволять пользователям приобретать или арендовать вещи по объявлениям либо размещать объявления для продажи своих вещей.

1. **Назначение системы**

Приложение предоставляет поиска, аренды, размещения объявлений о продаже бытовых вещей посредством мобильного и веб приложений и собственной системы оплаты. История покупок и информация о размещенных объявлениях хранится на странице профиля пользователя.

Основная задача приложения - предоставить пользователям удобный интерфейс и возможность легко приобрести или арендовать бытовые вещи различного типа.

1. **Требования к реализации(web)**

Для реализации клиентской части приложения (фронтенд) будет использован фреймворк React JS совместно с библиотекой Redux, которая позволяет управлять состоянием данных и пользовательским интерфейсом в приложении. Для создания компонент на страницах будет использована библиотека MaterialUI, которая также даёт возможность добавлять собственную систему дизайна к готовым компонентам. Также дополнительно для удобного написания CSS будет задействован фреймворк TailwindCSS.

TailwindCSS – это CSS-библиотека, которая упрощает стилизацию HTML, тем же путем, как это делает Bootstrap, – добавляя огромное количество разнообразных классов. Но, в отличие от Bootstrap, который добавляет уже готовые к употреблению компоненты, такие как кнопки, алерты и навбары, классы TailwindCSS нацелены на конкретное свойство.

Для перевода приложения на несолько языков будет использована библиотека **i18-next** - самое популярное и распространенное решения для локализации в React-приложениях. Главными преимуществами данной библиотеки являются:

* Большое комьюнити
* Множество плагинов, которые позволяют, использовать асинхронную подгрузку других локалей
* Автоматическое обновление компонентов при смене языка
* Поддержка concurrent features (React 18), для упрощения работы с асинхронными манипуляциями

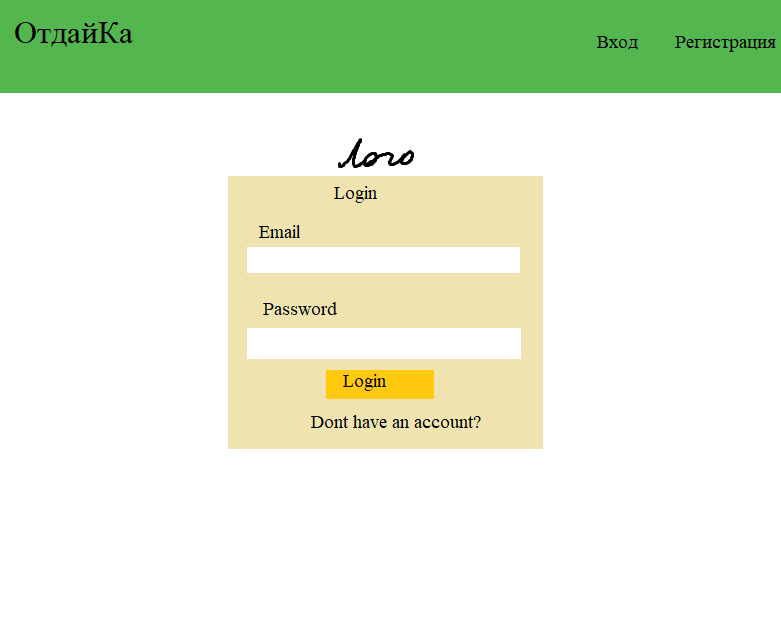


Рис. 1 – Интерфейс авторизации

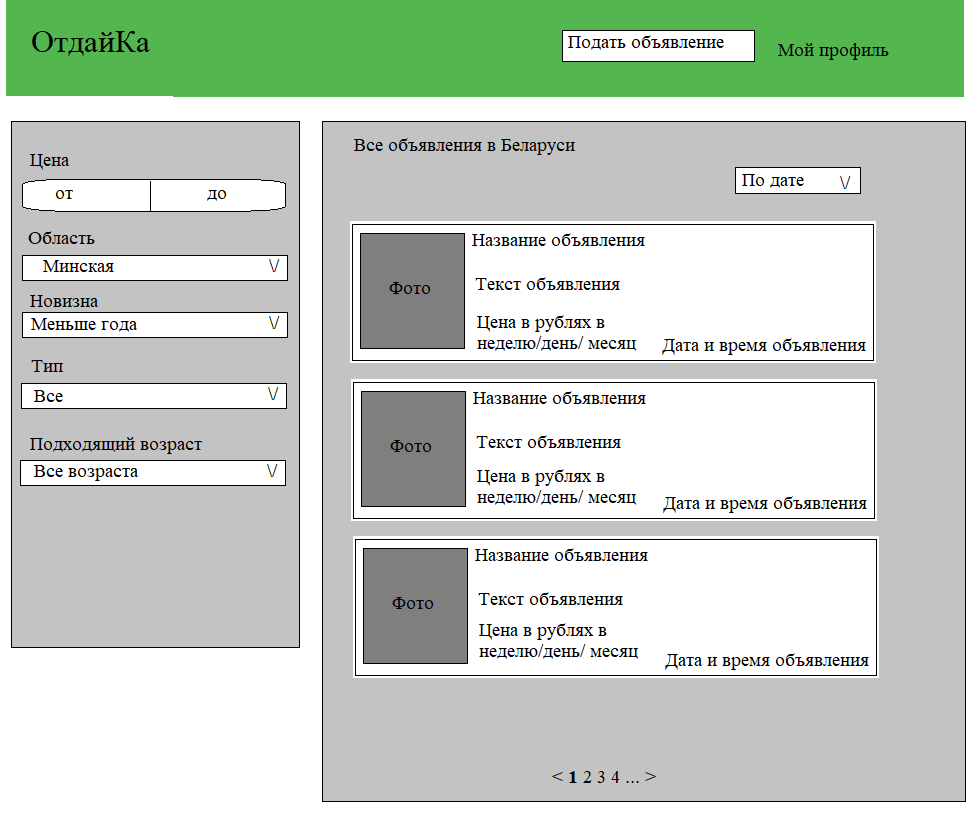


Рис. 2 – Главная страница клиента

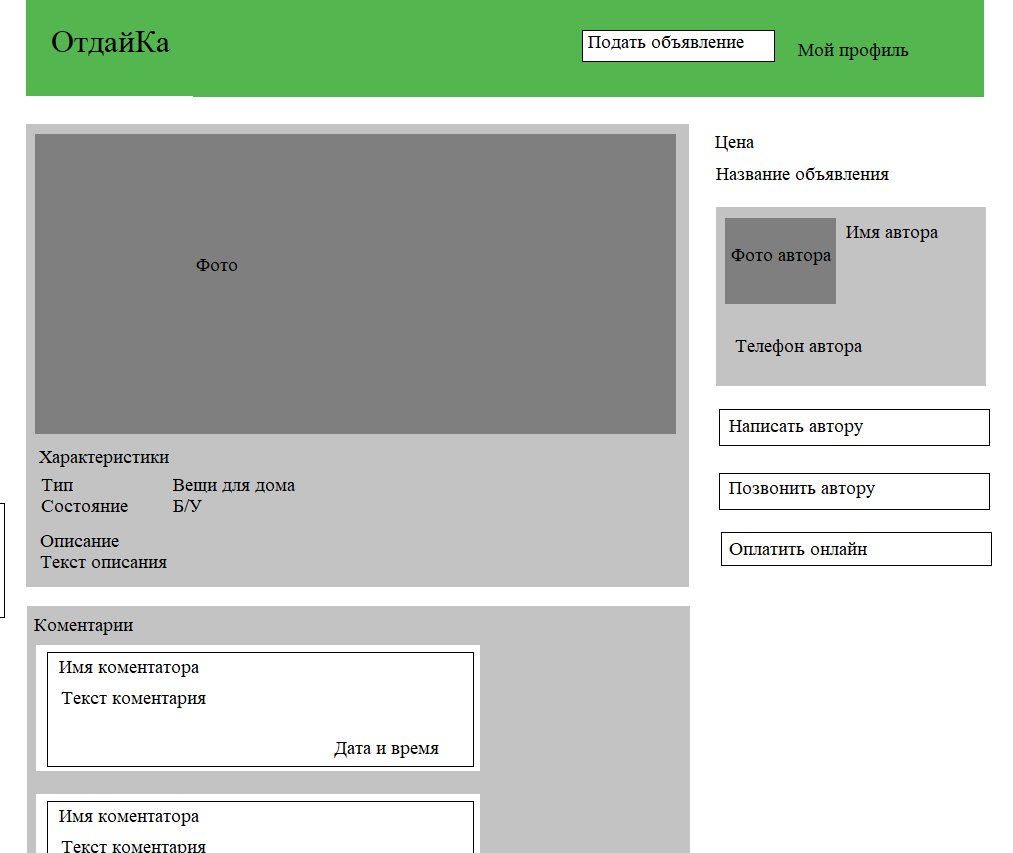


Рис. 3 – страница объявления

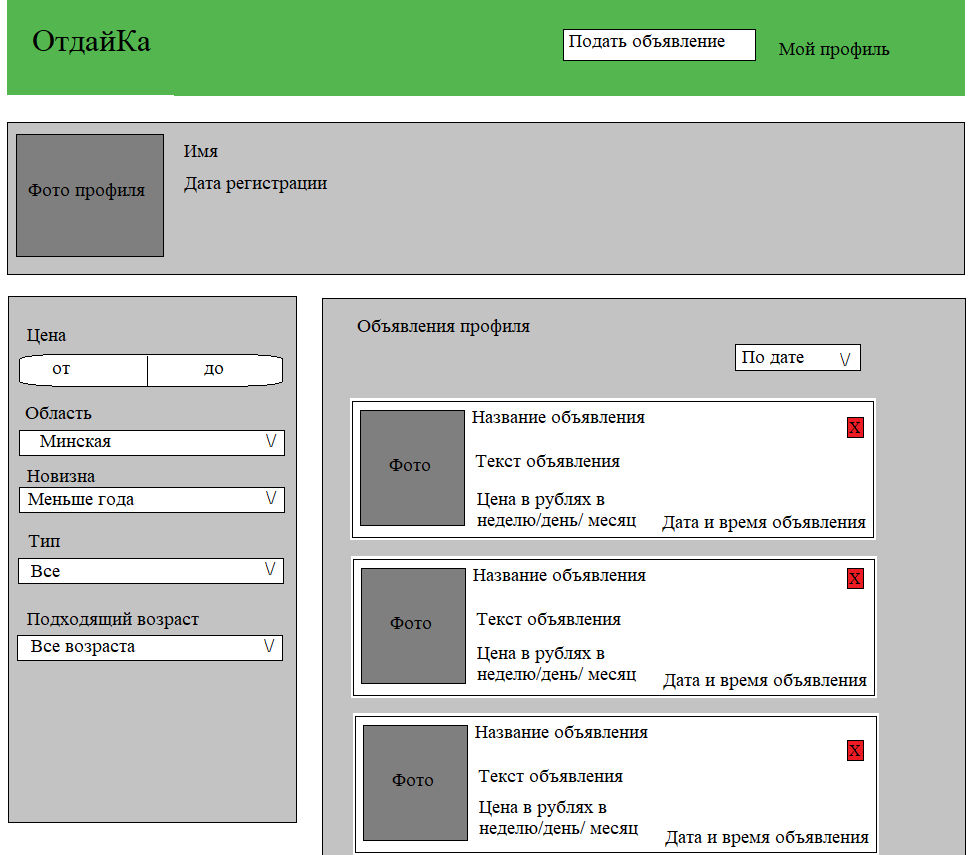


Рис. 4 – страница профиля пользователя

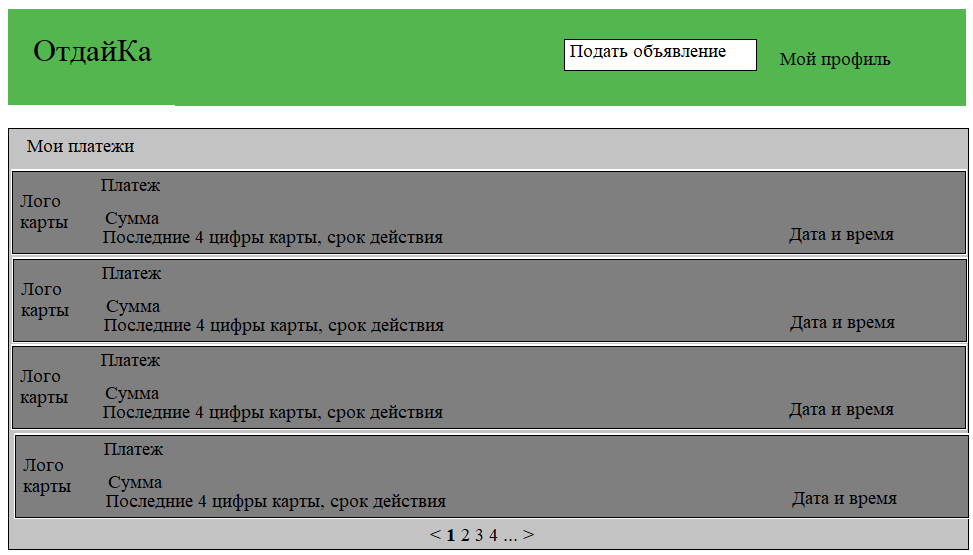


Рис. 5 – страница платежей пользователей

1. **Интеграция с Google Calendar API**

Консоль разработчиков Google предлагает платформу для создания первоклассных приложений с простой интеграцией сервисов и инструментов, предлагаемых Google. **Календарь Google** является одной из этих служб, и использование календаря может быть выгодным для предоставления таких функций, как планирование/создание событий в таких проектах, как приложение со списком дел или планировщик встреч.

В приложении пользователь может использовать Google-календарь для записи встреч с клиентами/продавцами. Каждый пользователь может привязать свою учетную запись Google и в будущем иметь возможность записи встреч в свой собственный календарь, также возможно получение уведомлений напрямую из гугл-календаря, которые будут информировать пользователя о скорой встрече.

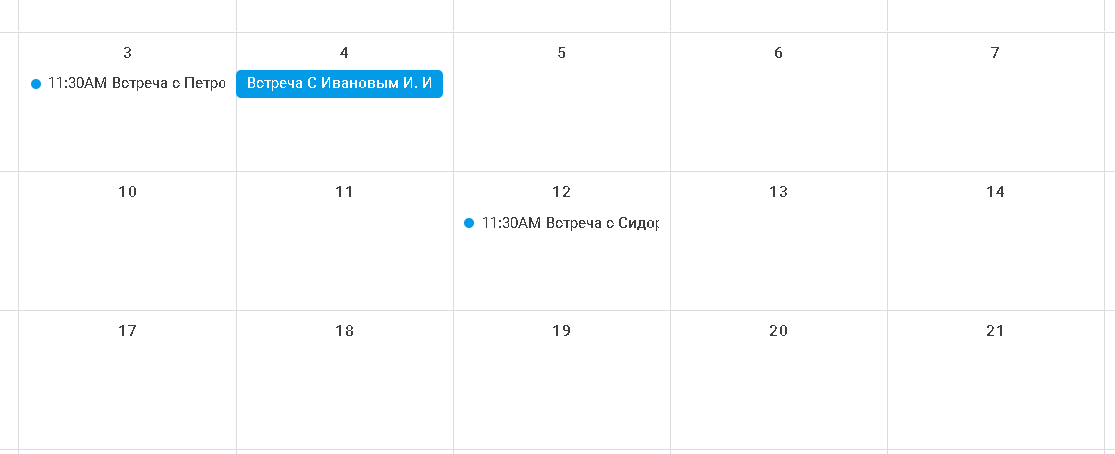


Рис. 6 – страница календаря пользователя

1. **Путь пользователя:**

1. Регистрация/авторизация.

При регистрации пользователь отправляет на сервер cвои данные, они сохраняются в базу данных и используются при последующей авторизации. При авторизации пользователь отправляет в базу данных логин и пароль. В базе данных пароль хранится в зашифрованном виде. В текущей сессии сохраняется username, на стороне сервера генерируется JSON WEB TOKEN. JWT будет использоваться для проверки аутентификации пользователя и последующего получения доступа к определенным видам данных.

2. Главная страница

После авторизации пользователь переходит на главный экран приложения, на которой он может получить информацию по объявлениям.

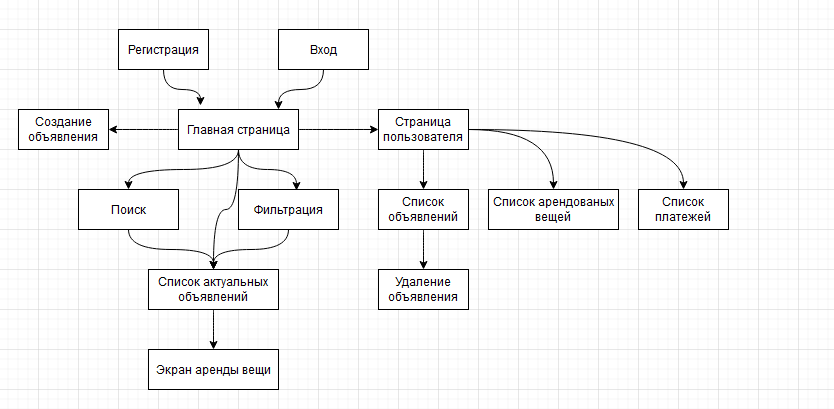
3. Фильтрация и поиск объявлений

Авторизованный пользователь может использовать фильтры (тип товара, область, цена) для поиска объявлений, соответствующих данным ограничениям.

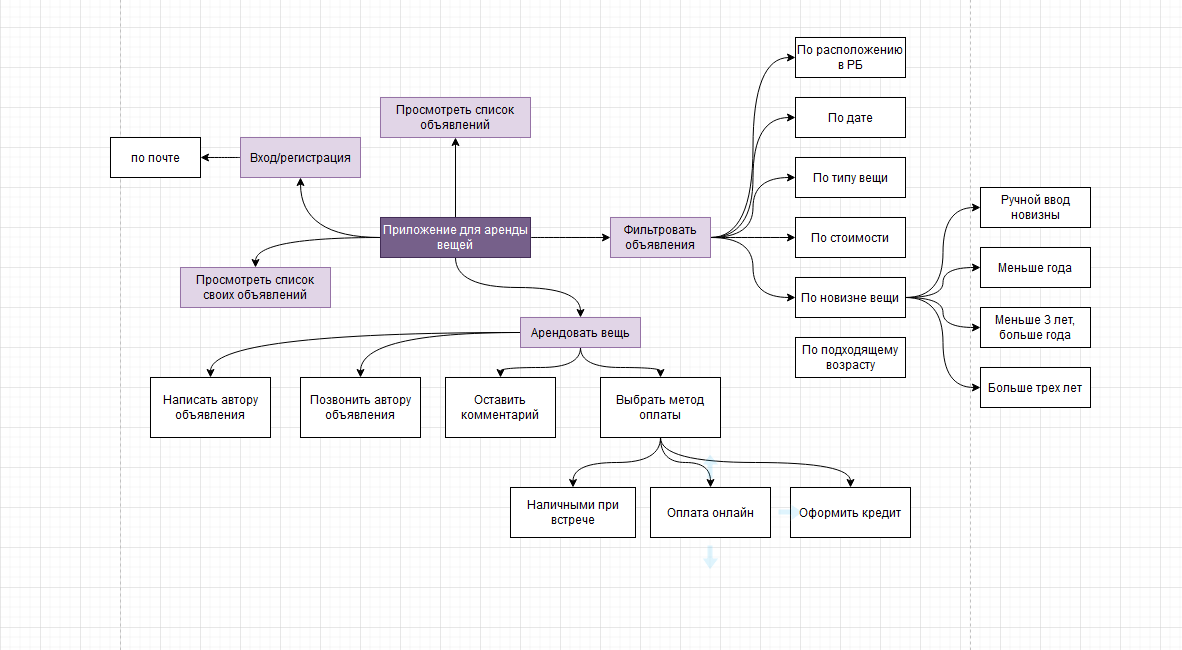
4. Страница объявления

После выбора конкретного объявления пользователем совершается переход на экран аренды товара. API получает GET-запрос по конкретному объявлению “/product/id” и выводит всю информацию по товару на экран.

**6. Диаграмма 1 – Путь пользователя**



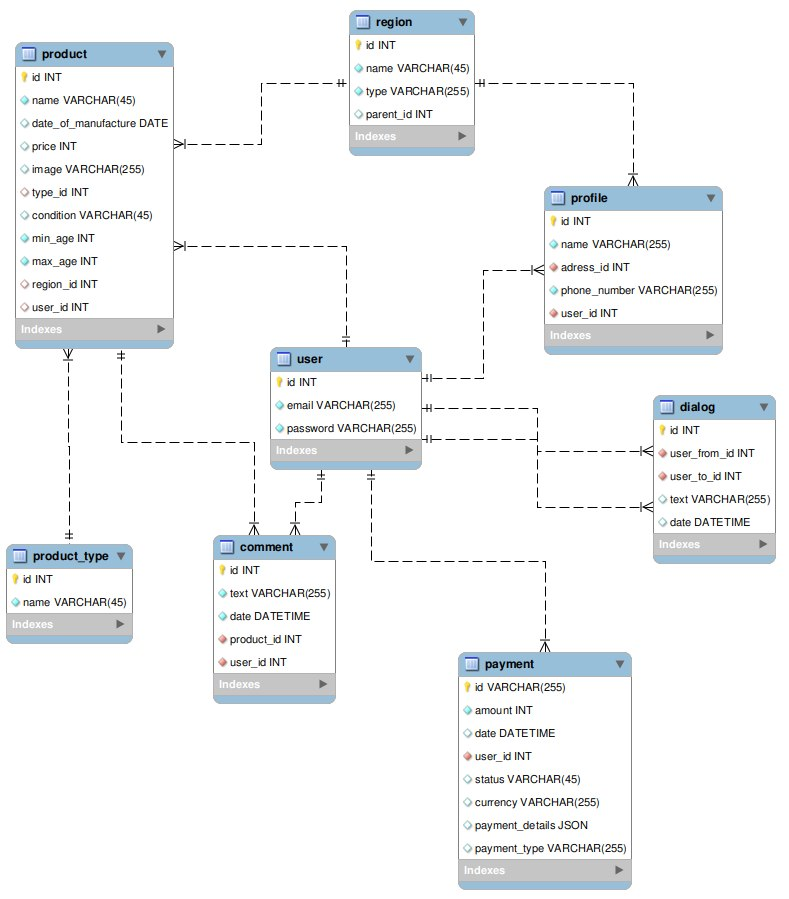
**7. Диаграмма 2 – Функциональная карта**



.

### 8. Ключевые обязанности

* Проектирование базы данных
* Разработка frontend-части приложения
* Интеграция с Google Calendar API
* Тестирование и рефакторинг

**9. Диаграмма 3 – Сущности базы данных**

### 10. Стек применяемых технологий

React+Redux, Tailwind, MaterialUI, Bootstrap, MySQL, Google Calendar API, Node JS.