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.

**2. 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.

**3. Требования к реализации (API)**

Используя Express и Node.js, мы можем реализовать полноценный API в стиле REST для взаимодействия с пользователем. Архитектура REST предполагает применение следующих методов или типов запросов HTTP для взаимодействия с сервером:

* GET
* POST
* PUT
* DELETE

Зачастую REST-стиль особенно удобен при создании всякого рода Single Page Application, которые нередко используют специальные javascript-фреймворки типа Angular, React или Knockout.

**REstfull-приложение** обладает следующими преимуществами:

* Надёжность (за счёт отсутствия необходимости сохранять информацию о состоянии клиента, которая может быть утеряна);
* Производительность (за счёт использования кэша);
* Масштабируемость;
* Прозрачность системы взаимодействия (особенно необходимая для приложений обслуживания сети);
* Простота интерфейсов;
* Портативность компонентов;
* Лёгкость внесения изменений;

К API приложения предъявляются следующие функциональные требования:

* Система пользователей

Приложение должно хранить информацию о всех пользователях в БД. API может предоставлять информацию о конкретных пользователях, сохраненных в базе данных. Обращение за информацией по конкретному пользователю: HTTP GET-запрос по endpoint “/user/id”

* Система объявлений, фильтрация и поиск объявлений

По аналогии с пользователями, инфорамция об объявлениях хранится в БД – модель “Product”. Обращение к конкретному объявлению идет GET-запросом к endpoint “/product”. Фильтрация объявлений по типу товаров будет обеспечиваться за счет передачи параметров в URL-запроса - “/product?type=furniture&min\_age=0&max\_age=5”

**4. Путь пользователя**

1. Регистрация/авторизация.

При регистрации пользователь отправляет на сервер свои 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 главную страницу 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. В данном случае API обрабатывает запрос по endpointу: “/product?type=…&max\_price=…” и возвращает данные по объявлениям в JSON формате.

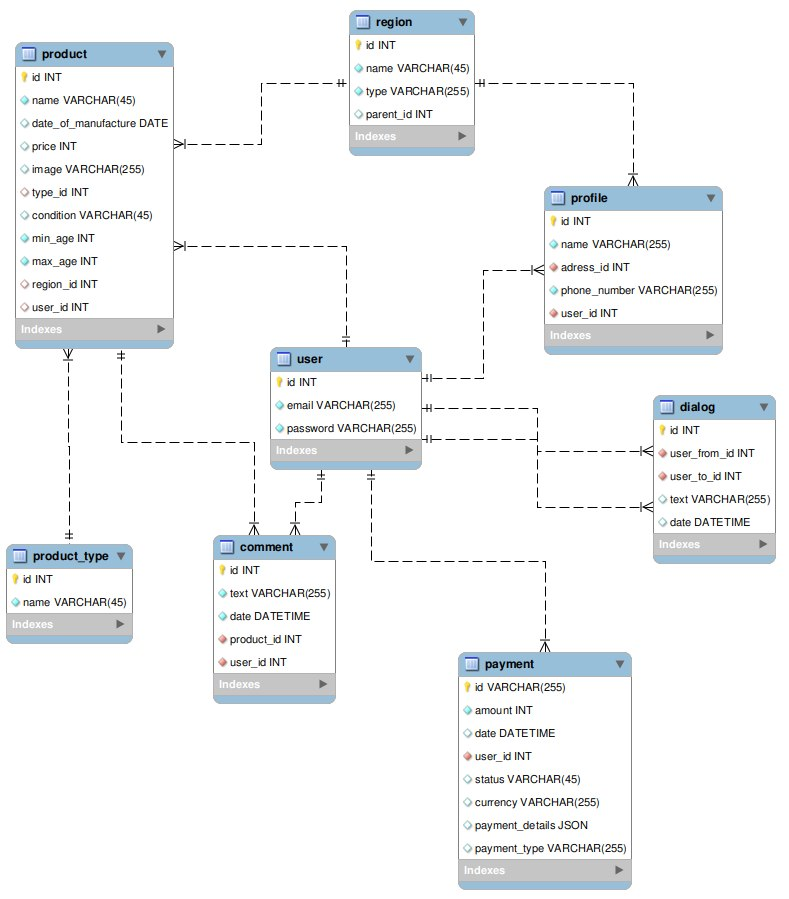
4. Страница объявления

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

**5. Ключевые обязанности**

* Проектирование базы данных
* Написание API (Система пользователей, объявлений, фильтрация, поиск по объявлениям, основной функционал)

**6. Сущности базы данных**



**7. Стек применяемых технологий**

Node JS, Express JS, PostgreSQL.