



МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«МИРЭА – Российский технологический университет»
РТУ МИРЭА


Институт информационных технологий (ИИТ)
Кафедра корпоративных информационных систем (КИС)

ОТЧЁТ ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ
Преддипломная практика

приказ Университета о направлении на практику от «20» апреля 2022 г. № 2410-С

Отчет представлен к
рассмотрению:
Студент группы ИКБО-09-18


«18» мая 2022


Лебедев О.А.
(подпись и расшифровка подписи)

Отчет утвержден.
Допущен к защите:

Руководитель практики
от кафедры

«18» мая 2022


Тарасов И.Е.
(подпись и расшифровка подписи)

Отлично
18.05.2022

Москва 2022 г.



МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«МИРЭА – Российский технологический университет»
РТУ МИРЭА

Институт информационных технологий (ИИТ)
Кафедра корпоративных информационных систем (КИС)

ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ НА ПРОИЗВОДСТВЕННУЮ ПРАКТИКУ
Преддипломная практика

Студенту 4 курса учебной группы ИКБО-09-18

Лебедеву Олегу Алексеевичу

Место и время практики: РТУ МИРЭА кафедра КИС, с 20 апреля 2022 г. по 17 мая 2022 г.

Должность на практике: студент

1. СОДЕРЖАНИЕ ПРАКТИКИ:

1.1. Изучить: техническую литературу, электронные информационно-образовательные ресурсы, применяемые для профессиональной деятельности по теме дипломной работы

1.2. Практически выполнить: в соответствии с темой дипломной работы:

- описать функционал разрабатываемого программного продукта;
- обосновать выбор средств разработки;
- описать реализацию программного продукта, реализацию БД;
- описать процесс тестирования;
- описать работу пользователя с программным продуктом.

1.3. Ознакомиться: с актуальными нормативно-правовыми документами, международными и отечественными стандартами, с СУБД и инструментальными средствами разработки

2. ДОПОЛНИТЕЛЬНОЕ ЗАДАНИЕ: нет

3. ОРГАНИЗАЦИОННО-МЕТОДИЧЕСКИЕ УКАЗАНИЯ: в процессе практики рекомендуется использовать периодические издания и отраслевую литературу годом издания не старше 5 лет от даты начала прохождения практики

Руководитель практики от
кафедры

Подпись

(Тарасов И.Е.)

«20» апреля 2022 г.

Задание получил

Подпись

(Лебедев О.А.)

«20» апреля 2022 г.

СОГЛАСОВАНО:

Заведующий кафедрой:

Подпись

(Андрианова Е.Г.)

«20» апреля 2022 г.



МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«МИРЭА – Российский технологический университет»
РТУ МИРЭА

**РАБОЧИЙ ГРАФИК ПРОВЕДЕНИЯ
ПРОИЗВОДСТВЕННОЙ ПРАКТИКИ**

студента Лебедев О.А. 4 курса группы ИКБО-09-18 очной формы обучения, обучающегося по направлению подготовки 09.03.04 Программная инженерия, профиль «Корпоративные информационные системы».

Неделя	Сроки выполнения	Этап	Отметка о выполнении
1	20.04.2022	Подготовительный этап, включающий в себя организационное собрание (Вводная лекция о порядке организации и прохождения производственной практики, инструктаж по технике безопасности, получение задания на практику)	
1	21.04.2022-26.04.2022	1) Описание функционала разрабатываемого программного продукта в соответствии с темой дипломной работы; 2) Обоснование выбора средств разработки.	
2	27.04.2022-03.05.2022	3) Описание этапов реализации программного продукта, реализации базы данных в соответствии с темой дипломной работы.	
3	04.05.2022-10.05.2022	4) Описание процесса тестирования программного продукта в соответствии с темой дипломной работы; 5) Описание работы пользователя с программным продуктом.	
4	11.05.2022-14.05.2022	Представление руководителю предварительной версии отчета с обеспечением согласованности материала по всем его частям, полученных на предыдущих этапах	
4	17.05.2022	Подготовка окончательной версии отчета по практике (Оформление материалов отчета в полном соответствии с требованиями на оформление письменных учебных работ студентов)	

Руководитель практики от кафедры

/Тарасов И.Е., д.т.н., профессор/

Обучающийся

/Лебедев О.А./

Согласовано:

Заведующий кафедрой


/Андрианова Е.Г., к.т.н., доцент/

Проведенные инструктажи:

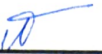
Охрана труда:

Инструктирующий

Инструктируемый



Подпись



Подпись


«20» апреля 2022 г.

Карусевич Т.Е., старший
преподаватель кафедры КИС
Лебедев О.А.


Техника безопасности:

Инструктирующий

Инструктируемый



Подпись



Подпись


«20» апреля 2022 г.

Карусевич Т.Е., старший
преподаватель кафедры КИС
Лебедев О.А.

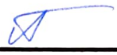
Пожарная безопасность:

Инструктирующий

Инструктируемый



Подпись




Подпись

«20» апреля 2022 г.

Карусевич Т.Е., старший
преподаватель кафедры КИС
Лебедев О.А.

С правилами внутреннего распорядка ознакомлен:



Подпись

«20» апреля 2022 г.

Лебедев О.А.

Оглавление

Оглавление.....	1
Введение.....	2
1 Функционал программного продукта	3
1.1 Функционал работы со складом.....	3
1.2 Функционал работы с заказами на аренду	4
1.3 Функционал работы с геолокацией.....	6
2 Выбор средств разработки.....	8
3 Реализация программного продукта, базы данных.....	9
3.1 Выбор системы управления базами данных	9
3.2 Разработка модели базы данных.....	10
3.3 Разработка веб-приложения.....	11
3.3.1 Клиентская часть приложения	11
3.3.2 Серверная часть приложения	12
4 Тестирование	14
5 Инструкция пользователя	16
5.1 Основные операции с оборудованием	16
5.1.1 Бронирование оборудования.....	16
5.1.2 Просмотр данных о заказе.....	17
5.1.3 Добавление нового оборудования.....	17
Заключение.....	19
Список литературы.....	19

Введение

Данная преддипломная практика выполняется по теме «веб-сервис определения местонахождения звукового оборудования для сетевой арендной организации» которая заключается в разработке системы, которая способна вывести пользователю отметку на карте с местоположением площадки, на котором находится оборудование. Предназначено для повышение информативного обеспечения сотрудников фирм проката оборудования.

Целью данной работы будет создание системы, которая позволит создавать заказы на аренду оборудования с расчетом стоимости и выбором дат, позволять пользователю и администраторам просматривать местонахождение оборудования.

Для достижения данной цели в процессе выполнения работы необходимо:

- исследование предприятия и бизнес-процессов;
- составление характеристики предметной области;
- выбор необходимых технологий, формирование требований к веб-приложению;
- анализ бизнес-процессов и построение архитектуры данных;
- описание экономических параметров разработки и внедрения;
- описание реализации конфигурации и интерфейса подсистемы;
- описание работы пользователя с подсистемой.

1 Функционал программного продукта

Одной из основных функций приложения ввести учет оборудования, которое находится на складе. А это значит, что основная система – это система работы с складом, вокруг которой будет строиться дополнительные системы заказов и отслеживания местоположения.

1.1 Функционал работы со складом

Схема функций, связанных с работой со складом представлена на рисунке 1.1.

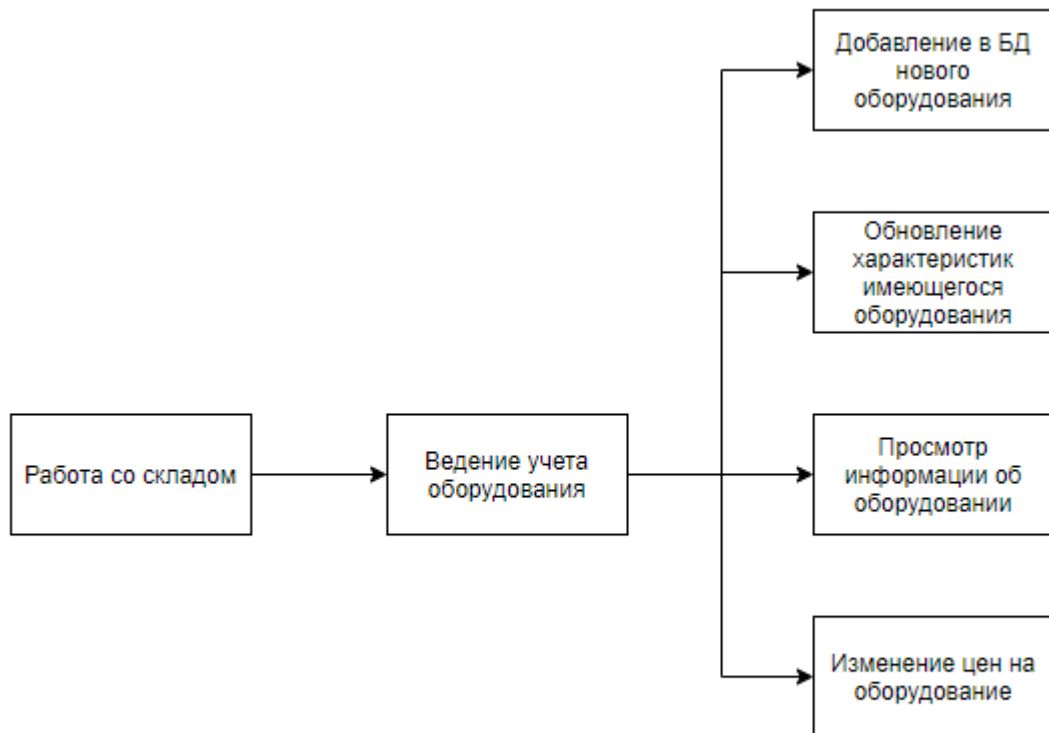


Рисунок 1.1 – Схема функций работы со складом

Как видно из рисунка 1.1, основная работа со складом является ввод и вывод информации об оборудовании его цене и характеристиках.

Так же помимо ввода и вывода этой информации, также важной функцией является изменение цены аренды товара.

Рассмотрим функции, связанные со складом:

1. Добавление в БД нового оборудования: необходимо иметь возможность пополнять склад оборудованием с указанием цены приобретения, цены аренды и самое главное характеристиками, которыми обладает оборудование;

2. Обновление характеристик имеющегося оборудования: система должна давать возможность администраторам изменять ключевые характеристики у оборудования;

3. Просмотр информации об оборудовании: эта функция должна предоставлять пользователю удобные фильтры для просмотра оборудования, уже имеющегося на складе;

4. Изменение цен на оборудование: система должна давать возможность изменять цену аренды оборудования;

1.2 Функционал работы с заказами на аренду

Схема функций, связанных с работой с заказами клиента представлена на рисунке 1.2.

При работе с заказами ведется список пользователей, которые производят заказы. В данном исполнении деление лиц на физические и юридические не ведется, так как функционал, заключающийся на юридических различиях этих групп, не являлся приоритетным.

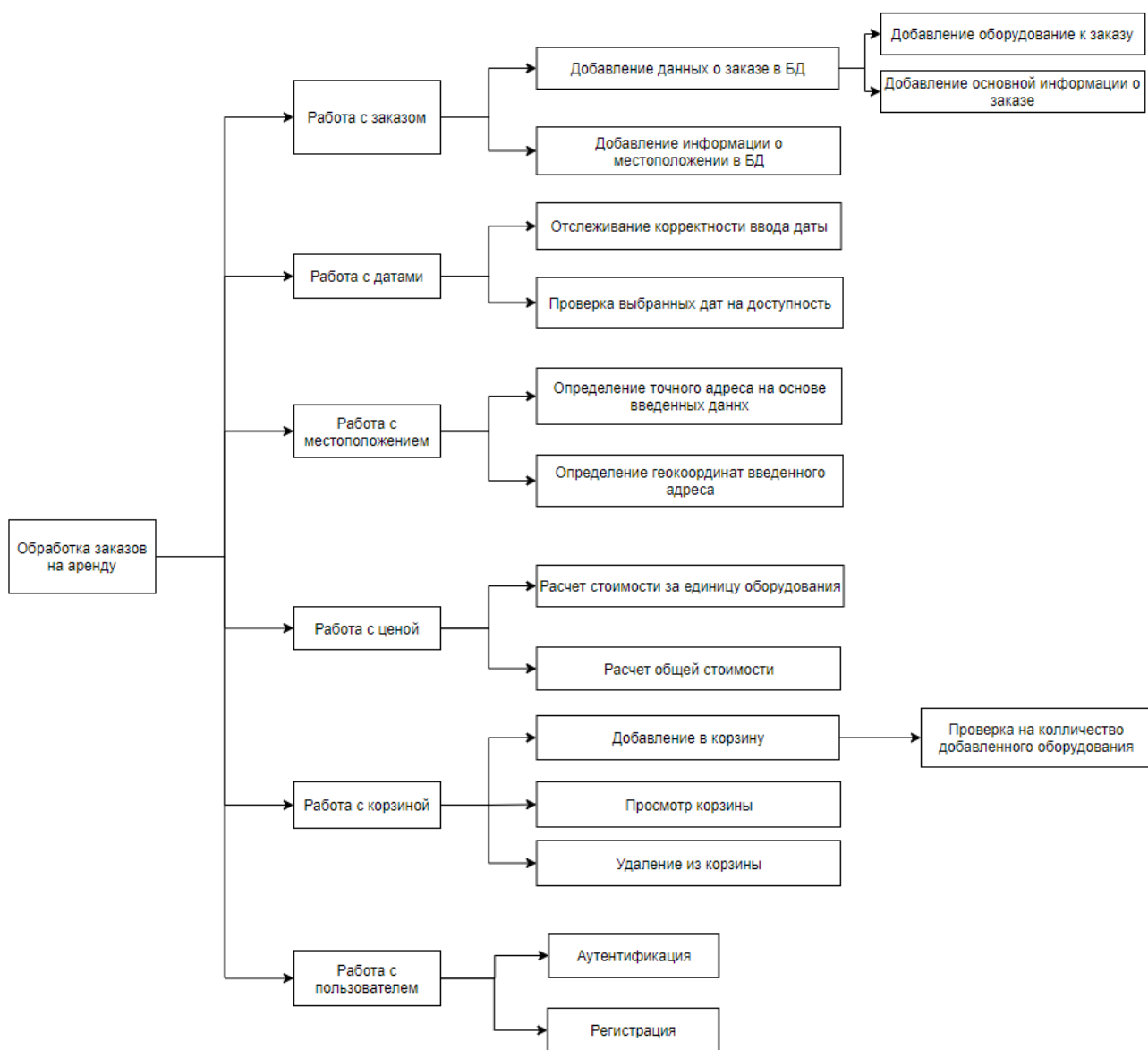


Рисунок 1.2 – Схема функций работы с заказами на аренду

Рассмотрим функционал работы с заказами. К каждому заказу добавляется информация о лице, который сделал заказ. Пользователь регистрируется, вводя свои имя, фамилию, номер телефона, пароль и почту, которая используется как логин при аутентификации. Если пользователь уже зарегистрирован, то он проходит процесс аутентификации.

Далее пользователь должен выбрать оборудование, которое он хочет забронировать и добавить его в корзину. Перед добавлением в корзину происходит проверка, в ходе которой выясняется, есть ли выбранное оборудование еще на складе, и если нет, то пользователю выходит соответствующее предупреждение. Пользователь так же может увидеть, что за товары у него находятся в корзине,

перейдя на соответствующую страницу. На это же странице пользователь может удалить товар из корзины.

Собрав все необходимое оборудование в корзине, пользователь должен ввести дату начала и конца аренды в соответствующие окна. В этих окнах будет информация о уже занятых датах на оборудование в корзине.

Далее пользователь должен ввести адрес, куда необходима доставка заказа. Пользователь должен ввести адрес и подтвердить его, для того чтобы система смогла получить необходимые геокоординаты для отображения адреса на карте.

Перед подтверждением заказа пользователь сможет ознакомиться с рассчитанной системой, на основе введенных дат, стоимостью.

После подтверждения пользователя, что все верно, заказ добавляется в базу данных, к заказу прикрепляются выбранное оборудование, и добавляется информация о адресе заказа.

1.3 Функционал работы с геолокацией

Функционал работы с геолокацией представлен на рисунке 1.3.

При разработке функций геолокации было встречено ряд трудностей, состоящий в основном из закрытости доступа к API самых известных систем работ с картами: Яндекс карт, google maps и 2gis. В связи с этим было решено ограничить функционал работы с картами отображением места на карте.

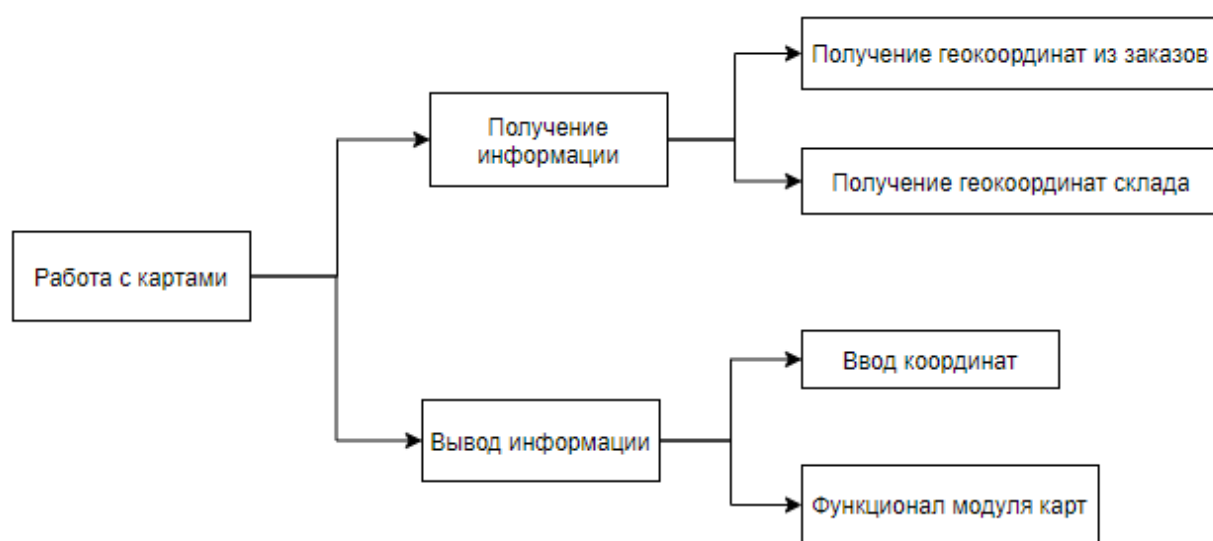


Рисунок 1.3 – Схема функций работы с геолокацией

Рассмотрим функционал работы с геолокацией. В базе данных содержится информация о местоположении каждой единицы оборудования, места куда его нужно доставить по заказу и местоположение склада.

Пользователь и администратор имеют разный уровень доступа к информации в базе данных, так пользователь может увидеть местоположение места куда отправится заказ. У администратора данных больше. Он может отслеживать местоположение каждого товара на специальной странице и изменять его, меняя состояние товара. После изменения состояние страницы необходимо перезагрузить для корректного отображения данных на карте.

Кроме того, сама карта имеет базовый функционал в виде изменения масштаба и перемещения. При этом на месте, которое отмечено как текущее местоположение будет стоять маркер, который нельзя убрать при этом взаимодействие с картой.

2 Выбор средств разработки

Программный продукт решено реализовывать в формате веб-приложения, так как это наиболее распространенный и простой способ взаимодействия пользователя и системы. Один из самых распространенных языков для web-программирования является JavaScript. Кроме того, с его помощью можно реализовать и front, и back части приложения. Именно поэтому он был выбран в качестве основного языка программирования.

JavaScript – мультипарадигменный язык программирования. Поддерживает объектно-ориентированный, императивный и функциональный стили. Является реализацией спецификации ECMAScript.

Основные архитектурные черты: динамическая типизация, слабая типизация, автоматическое управление памятью, прототипное программирование, функции как объекты первого класса.

Среди множества библиотек для создания frontend части приложения была выбрана библиотека ReactJS. Преимуществами данной библиотеки можно назвать простоту в изучении и использовании, а также наличие виртуальной DOM, которая упорядочивает документы HTML и XHTML, что позволяет веб-браузерам быстрее обрабатывать элементы веб-приложений.

Для написания серверной части приложения была выбрана библиотека Node.js так как эта библиотека прошла проверку временем, обладает обширным функционалом и может интегрировать в себя огромную библиотеку модулей JavaScript.

3 Реализация программного продукта, базы данных

3.1 Выбор системы управления базами данных

Для реализации базы данных была выбрана СУБД PostgreSQL. Она легко интегрируется в серверную библиотеку Node.js, построена на языке SQL, что делает ее простой в использовании.

PostgreSQL – свободная объектно-реляционная система управления базами данных, основанная на языке SQL.

СУБД отличается высокой надёжностью и хорошей производительностью. PostgreSQL поддерживает транзакции (ACID), репликация реализована встроенными механизмами. При этом система расширяемая – можно создавать свои типы данных и индексов, а также расширять поведение при помощи языков программирования.

Сильными сторонами PostgreSQL считаются:

- Высокопроизводительные и надёжные механизмы транзакций и репликации;
- Расширяемая система встроенных языков программирования: в стандартной поставке поддерживаются PL/pgSQL, PL/Perl, PL/Python и PL/Tcl; дополнительно можно использовать PL/Java, PL/PHP, PL/Py, PL/R, PL/Ruby, PL/Scheme, PL/sh и PL/V8, а также имеется поддержка загрузки модулей расширения на языке C;
- Наследование;
- Возможность индексирования геометрических (в частности, географических) объектов и наличие базирующегося на ней расширения PostGIS;
- Встроенная поддержка слабоструктурированных данных в формате JSON с возможностью их индексации;
- Расширяемость (возможность создавать новые типы данных, типы индексов, языки программирования, модули расширения, подключать любые внешние источники данных).

В разработке простых сайтов PostgreSQL используется несколько реже, чем MySQL/MariaDB, но всё же эта пара с заметным отрывом опережает по частоте

использования остальные системы управления базами данных. При этом в разработке сложных сайтов и веб-приложений PostgreSQL опережает по использованию MySQL и MariaDB. Большинство фреймворков (например, Ruby on Rails, Yii, Symfony, Django) поддерживают использование PostgreSQL в разработке.

3.2 Разработка модели базы данных

Для простоты взаимодействия сервера и СУБД, было принято решение, под каждый элемент данных выделить свою таблицу. В итоге получилось 7 таблиц.

На рисунке 3.1 изображена ERD диаграмма базы данных.

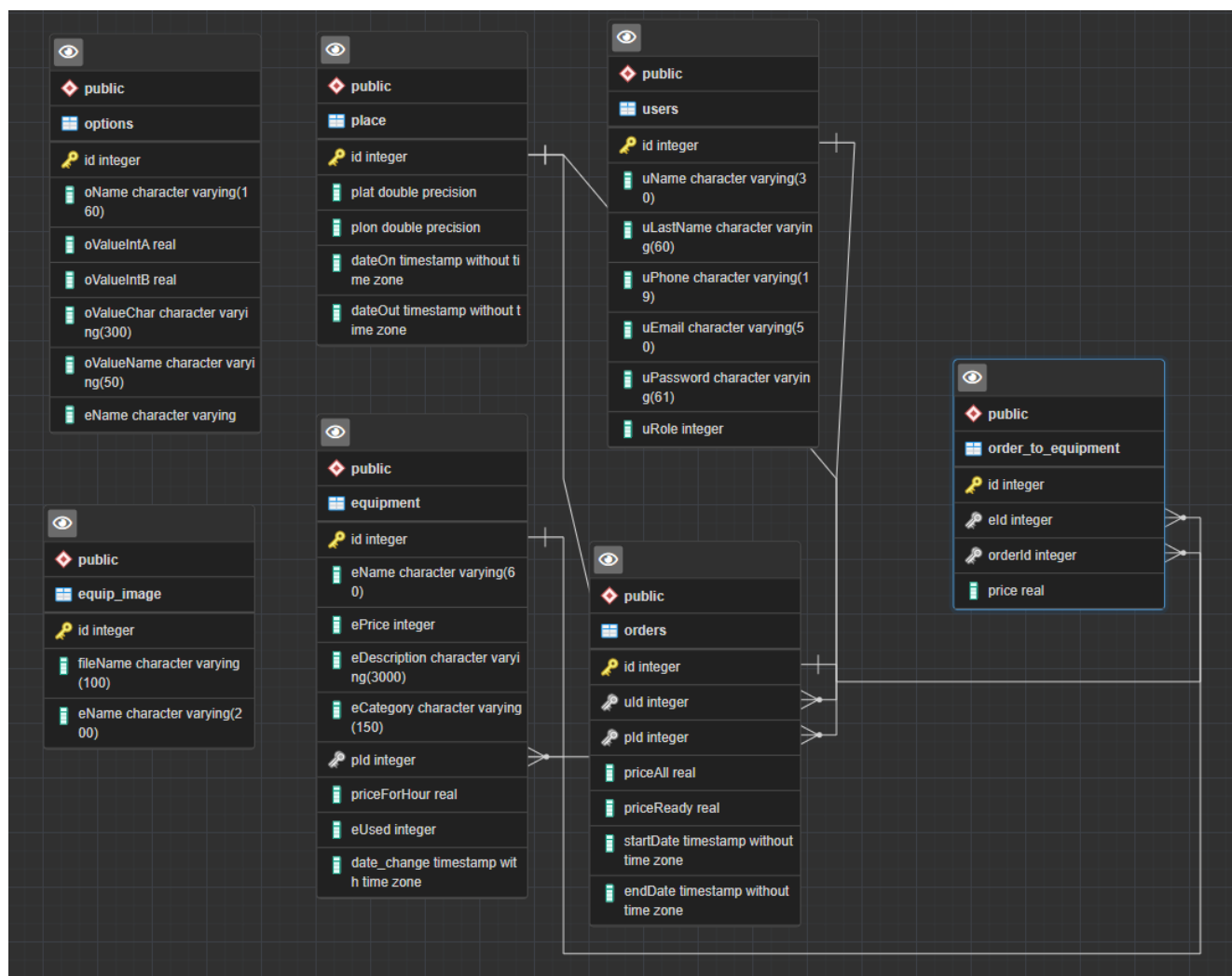


Рисунок 3.1 – ERD диаграмма базы данных

3.3 Разработка веб-приложения

Разрабатываемое приложение состоит из двух частей: frontend (клиентская часть) и backend (серверная часть). Затем прописывается связь между этими частями и получается рабочее приложение.

3.3.1 Клиентская часть приложения

В ходе разработки клиентской части приложения создаются страницы, на которых в дальнейшем будет отображаться информация, взятая из серверной части.

После создания приложения общее количество страниц стало равно 6. Каждая страница содержит в себе функциональные модули для получения, обработки и обменом данными с серверной частью. Например, на рисунке 3.2 изображена страница, где для пользователей выводятся доступные для заказа товары.

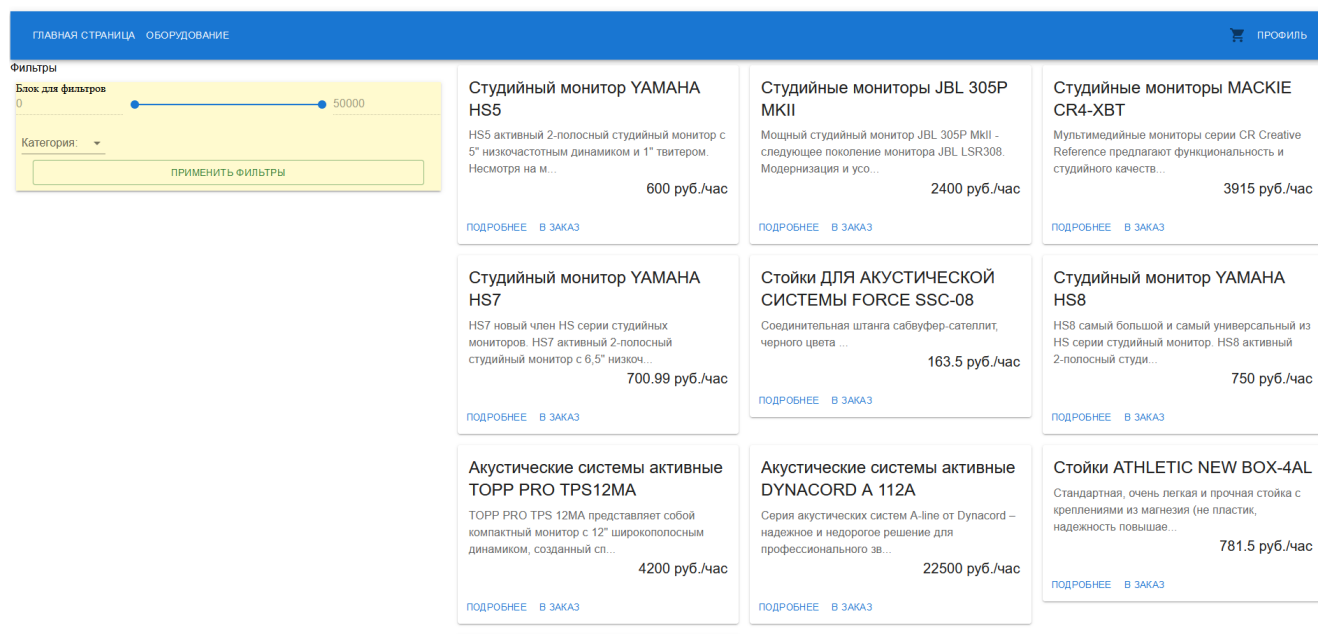


Рисунок 3.2 – Страница для заказа товаров.

Рассмотрим функционал этой страницы подробнее. При загрузке страницы происходит несколько запросов на сервер для получения оборудования, чтобы отобразить его на странице, и для получения всех категорий оборудования, для использования в фильтрах.

После того как были получены данные об оборудовании эта информация передается в модуль, который отвечает за отрисовку блоков с конкретным товаром. В этом блоке выводятся наименование товара, краткое описание, цену аренды за

час времени, и две кнопки: одна отвечает за переход на страницу с описанием конкретного оборудования, а вторая за добавления товара в корзину.

Так же на странице есть модуль, отвечающий за фильтры. При передвижении ползунок меняется диапазон цен, в котором должны находиться товары и выпадающий список с категориями оборудования. При нажатии на кнопку «Применить фильтр» происходит запрос к серверу для получения нового списка оборудования, который подходит под выбранные фильтры.

3.3.2 Серверная часть приложения

При разработке серверной части приложения пишутся обработчики для запросов с клиентской части приложения и запросы к базе данных с первичной обработкой полученных данных.

Возьмем примером запросы, которые приходят со страницы на рисунке 3.2.

Запрос на получение всех оборудования при загрузке страницы представлен на рисунке 3.3

```
25 // */api/item/getItems
26 router.get("/getItems", async (req, res) => {
27   try {
28     const itemsArray = await db.query("SELECT * FROM equipment");
29     const items = new Map();
30     let minPrice = itemsArray.rows[0].priceForHour;
31     let maxPrice = itemsArray.rows[0].priceForHour;
32     for (let i = 0; i < itemsArray.rowCount; i++) {
33       if (minPrice > itemsArray.rows[i].priceForHour) {
34         minPrice = itemsArray.rows[i].priceForHour;
35       }
36       if (maxPrice < itemsArray.rows[i].priceForHour) {
37         maxPrice = itemsArray.rows[i].priceForHour;
38       }
39       if (items.has(itemsArray.rows[i].eName)) {
40         if (
41           items.get(itemsArray.rows[i].eName).eUsed >= itemsArray.rows[i].eUsed
42         ) {
43           items.set(itemsArray.rows[i].eName, itemsArray.rows[i]);
44         }
45       } else {
46         items.set(itemsArray.rows[i].eName, itemsArray.rows[i]);
47       }
48     }
49     let array = Array.from(items, ([name, value]) => ({ name, value }));
50     res.status(201).json({ items: array, maxPrice, minPrice });
51   } catch (e) {
52     res.status(400).json({ message: e.message });
53   }
54 });
```

Рисунок 3.3 – Фрагмент кода получения оборудования.

Когда приходит данный запрос на сервер, происходит получение данных из базы данных хранящиеся в таблице «equipment», после получения оборудование записывается в словарь, для избегания множественного вывода одного и того же товара на странице, также параллельно высчитываются максимальные и минимальные ценники оборудования, для того чтобы в дальнейшем использовать их в фильтрах.

4 Тестирование

Для тестирования было выбрано функциональное тестирование.

Функциональное испытание – это тип тестирования программного обеспечения, который проверяет программную систему на соответствие функциональным требованиям/спецификациям. Цель функциональных тестов состоит в том, чтобы проверить каждую функцию программного приложения, предоставляя соответствующий ввод, проверяя выход в соответствии с функциональными требованиями.

Функциональное тестирование в основном включает тестирование черного ящика и не касается исходного кода приложения. Это тестирование проверяет пользовательский интерфейс, API, базу данных, безопасность, связь клиент / сервер и другие функциональные возможности тестируемого приложения.

Данные тестирования представлены в таблицах 4.1 и 4.2.

Таблица 4.1 – Тестирование основных функций клиентской части приложения

Полномочия	Описание проверки	Ожидаемый результат	Статус проверки
Не зарегистрированный	Регистрация без введенных данных	Ошибка регистрации введите данные.	Пройдено
Не зарегистрированный	Регистрация с введенными данными	Пользователь успешно создан	Пройдено
Пользователь/администратор/не зарегистрированный	Применение всех фильтров	Успешное применение и изменение списка товаров	Пройдено
Пользователь/администратор/не зарегистрированный	Добавление товаров в корзину	Успешное добавление товара в корзину пока он есть в наличии	Пройдено
Не зарегистрированный	Попытка создать заказ	Предупреждение о том, что пользователь не зарегистрирован	Пройдено
Пользователь/администратор	Попытка создать заказ без введенных дат или адреса	Заказ не создан пользователь уведомлен	Пройдено

Продолжение таблицы 4.1

Пользователь	Попытка зайти на страницу для администратора	Не возможность зайти на страницу из-за низкой роли	Пройдено
Пользователь/ администратор	Изменения личных данных и пароля	Корректное изменение данных	Пройдено
Администратор	Добавление нового оборудования	Корректное добавление в БД	Пройдено

Таблица 4.2 – Тестирование основных функций серверной части приложения

Полномочия	Описание проверки	Ожидаемый результат	Статус проверки
Не зарегистрированный	Возможность зарегистрироваться с пустыми полями	Ошибка регистрации введите данные.	Пройдено
Не зарегистрированный	Регистрация с введенными данными	Пользователь успешно добавлен в базу данных и на клиент выслан токен доступа	Пройдено
Пользователь/ администратор/не зарегистрированный	Получение товаров с фильтрами и без	Успешное получение данных из Базы данных, запись в словарь и отправка клиенту	Пройдено
Пользователь/ администратор	Добавление оборудования	Успешное добавление в базу данных оборудование и его характеристик	Пройдено
Пользователь/ администратор	Получение геокоординат из полученного адреса	Успешное получение геокоординат и запись их в базу данных	Пройдено

5 Инструкция пользователя

5.1 Основные операции с оборудованием

5.1.1 Бронирование оборудования

Для брони оборудования необходимо добавить его в корзину на странице с оборудованием (рисунок 5.1), затем перейти на страницу корзины (рисунок 5.2) и нажать кнопку «Забронировать». В открывшейся форме (рисунок 5.3), ввести необходимые данные и подтвердить бронь.

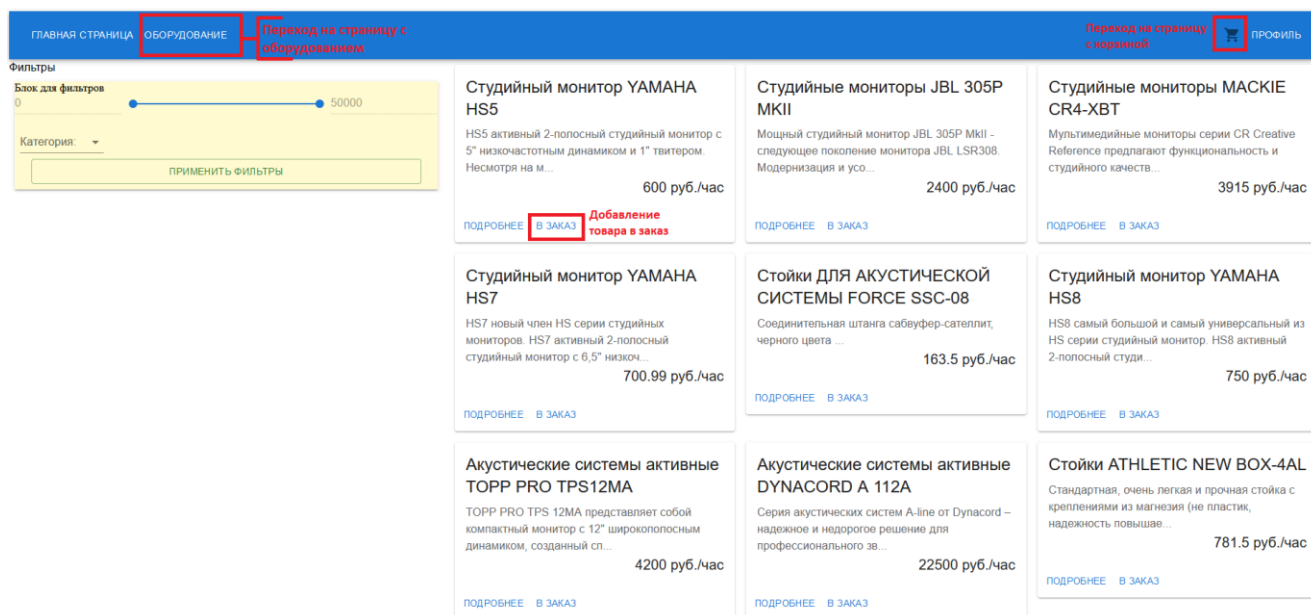


Рисунок 5.1 – Страница для заказа товаров.

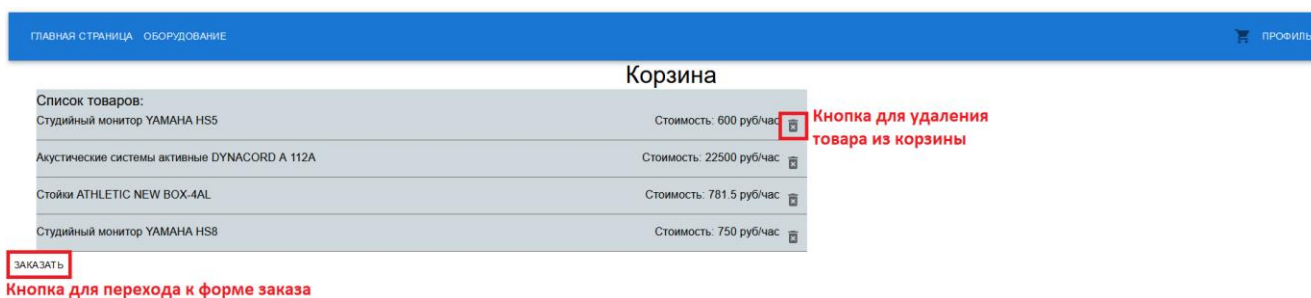


Рисунок 5.2 – Страница корзины.

Заказ **Поля для ввода промежутка дат**

Выберите промежуток аренды

30-05-2022 06:00

02-06-2022, 16:00

Введите адрес, куда будет производиться доставка

Адрес: Проспект Вернадского 76

[ПОДТВЕРДИТЬ АДРЕС](#)

Внимательно проверьте адрес зафиксированный системой

Адрес: г Москва, пр-кт Вернадского, д 76

[ПОДТВЕРДИТЬ ВРЕМЯ И МЕСТО](#)

Подтверждение дат и адреса

В поле вводится адрес, затем с помощью кнопки он фиксируется в системе

Введите/Подтвердите ваши персональные данные

Имя: Олег

Фамилия: Лебедев

Номер телефона: +7(965)288-12-86

Почта: lebedevoleg72@gmail.com

[ПОДТВЕРДИТЬ ДАННЫЕ](#)

Подтверждение данных пользователя

Счет:

Наименование	Цена/час	Итого
Студийный монитор YAMAHA HS5	600 руб/час	49200 руб
Акустические системы активные DYNACORD A 112A	22500 руб/час	1845000 руб
Стойки ATHLETIC NEW BOX-4AL	781.5 руб/час	64083 руб
Студийный монитор YAMAHA HS8	750 руб/час	61500 руб

Итого: 2019783

К оплате авансом идет 15% от всей стоимости

К оплате: 302967.45

Если сумма аренды вышла меньше 6 тыс. руб, то доставка оплачивается дополнительно 1500 руб

Подтверждение заказа

[ПОДТВЕРДИТЬ ЗАЯВКУ](#)

Рисунок 5.3 – Форма бронирования.

5.1.2 Просмотр данных о заказе

Для зарегистрированных пользователей, которые уже сделали заказ, есть возможность посмотреть детали заказа. Необходимо посетить страницу профиля (рисунок 5.4) и перейти в вкладку заказы.

ГЛАВНАЯ СТРАНИЦА ОБОРУДОВАНИЕ

Страница пользователя

[ПРОФИЛЬ](#) [ЗАКАЗЫ](#) [СМЕНА ПАРОЛЯ](#) **Вкладки**

Переход на страницу профиля

ПРОФИЛЬ

Мой аккаунт

Выйти

Информация о заказах

Номер	Адрес	Даты	Оплачено	Вся сумма
7	3, Ходынский бульвар, Ходынское поле, Хорошёвский район, Москва, Центральный федеральный округ, 125252, Россия	12-05-2022 03:00 - 19-05-2022 15:30	113715	758100

3, Ходынский бульвар, Ходынское поле, Хорошёвский район, Москва, Центральный федеральный округ, 125252, Россия

Сергиевская слобода

Метрополитен

ЖК Тонус

Как добраться API Яндекс.Услуги

Рисунок 5.4 – Страница профиля

5.1.3 Добавление нового оборудования

Для добавления оборудования в систему, нужно зайти на сервисную страницу (рисунок 5.5), предварительно зайдя под аккаунтом администратора.

Затем открыть форму добавления оборудования (рисунок 5.6, рисунок 5.7), заполнить все поля и подтвердить добавление.

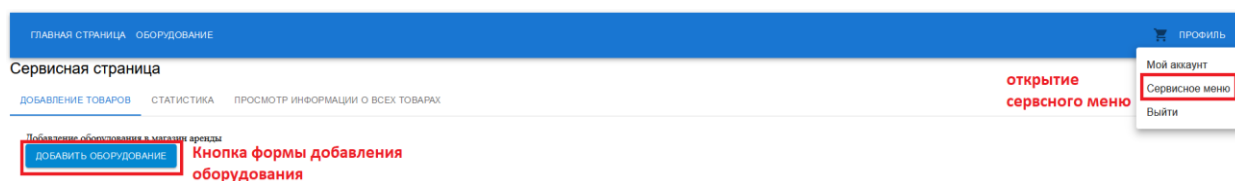


Рисунок 5.5 – Сервисное меню

The screenshot shows the 'Добавление оборудования' form. It has two steps: '1. Добавление основных параметров' (active) and '2. Добавление характеристик'. The first step contains several input fields: 'Выберите категорию:' (with a dropdown), 'Введите название товара:' (with a text input), 'Введите цену за которую куплен товар:' (with a text input), 'Введите цену аренды за час:' (with a text input), 'Введите Описание товара:' (with a text area), and 'Введите количество товара:' (with a text input). There are two buttons: 'ЗАГРУЗИТЬ ФОТО' (highlighted with a red box and labeled 'Поле для загрузки фотографий') and 'ПЕРЕЙТИ К ЗАПОЛНЕНИЮ ХАРАКТЕРИСТИК' (highlighted with a red box and labeled 'Кнопка перехода к следующей стадии заполнения данных').

Рисунок 5.6 – Форма добавления оборудования. Часть 1

The screenshot shows the 'Добавление оборудования' form, specifically the '2. Добавление характеристик' step. The title 'Изменение характеристик' is displayed. Below it are several input fields for technical specifications: 'Частотный диапазон:' (with a text input), 'Выходная мощность:' (with a text input), 'Входы/Выходы:' (with a text input), 'Габариты (Ширина/Высота/Глубина):' (with a text input), 'Габариты в заводской упаковке:' (with a text input), 'Вес без упаковки:' (with a text input), and 'Вес в упаковке:' (with a text input). At the bottom left, there is a 'СОХРАНИТЬ ТОВАРЫ' button (highlighted with a red box and labeled 'Кнопка сохранения оборудования').

Рисунок 5.7 – Форма добавления оборудования. Часть 2

Заключение

В ходе выполнения данной практики было разработано программное средство, которое способно выводить отметку на карте с текущим местоположением арендуемого оборудования. Так же средство обладает функционалом создания заказов на аренду и расчета его стоимости.

Для создания системы была составлена характеристика предметной области, Выбраны и проанализированы технологии, с помощью которых можно создавать данное средство. Была составлена структура базы данных и реализовано веб-приложение, состоящее из клиентской и серверной части.

Список литературы

1. Баулина К.В. Имитационное моделирование процесса сдачи в аренду объектов технопарка //Форум молодых ученых. – 2018. – №. 11-1. – С. 184-188.
2. Семёнов Е.С. Особенности проектирования веб-ориентированных систем для предприятий прокатного бизнеса // Образование. Наука. Производство. – 2018. – С. 1857-1860.
3. Потовиченко М. А., Шатилов Ю. Ю. РАЗРАБОТКА КЛИЕНТСКОЙ ЧАСТИ ОДНОСТРАНИЧНОГО WEB-ПРИЛОЖЕНИЯ С ИСПОЛЬЗОВАНИЕМ БИБЛИОТЕКИ REACT // Научное обозрение. Технические науки. – 2020. – №. 1. – С. 39-43.
4. Хэррон Д. Node. js Разработка серверных веб-приложений на JavaScript. – Litres, 2022. – С. 12-19.
5. Выходцев Н.А. Разработка информационной системы аренды спецтехники коммерческой организации. – 2016.
6. Пивень А. А., Скорин Ю. И. Тестирование программного обеспечения // Системи обробки інформації. – 2012. – №. 4 (1). – С. 56-58.
7. Drake J. D., Worsley J. C. Practical PostgreSQL. – " O'Reilly Media, Inc.", 2002.
8. DaData, API сервиса [Электронный ресурс]: URL: <https://dadata.ru/api/>

9. Документация к PostgreSQL 14.3 [Электронный ресурс]: URL: <https://postgrespro.ru/docs/postgresql/14/index>

10. Node.js v17.9.0 documentation [Электронный ресурс]: URL: <https://nodejs.org/docs/latest-v17.x/api/>

11. ReactJS документация [Электронный ресурс]: URL: <https://ru.reactjs.org/docs/getting-started.html>