МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ПОВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет **Информатики и вычислительной техники**

**Кафедра информатики и системного программирования**

**КУРСОВОЙ ПРОЕКТ (РАБОТА)**

По дисциплине «Базы данных»

На тему «Система для управления чайным магазином»

Выполнил обучающийся гр. ПС-31

Кутимский Р. В.

(Фамилия .И.О.)

Проверил: Доцент кафедры ИиСП, к.т.н. Лучинин З.С.

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_

Дата \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Йошкар-Ола

2024 г

# Содержание

[Введение 3](#_Toc188735575)

[1. Анализ программных технологий для разработки web-сервиса 5](#_Toc188735576)

[1.1. Backend технологии 5](#_Toc188735577)

[1.2. Базы данных 6](#_Toc188735578)

[1.3. Frontend технологии 7](#_Toc188735579)

[2. Подготовительные работы для разработки 9](#_Toc188735580)

[2.1. Выбор ферймворков и библиотек 9](#_Toc188735581)

[2.2. Архитектура приложения 10](#_Toc188735582)

[2.3. Реализация общения между клиентом и сервером 14](#_Toc188735583)

[3. Реализация web-сервиса 16](#_Toc188735584)

[4. Литература 17](#_Toc188735585)

# Введение

Web-сервис — это программа, выполняющая определенные действия по запросам пользователей через интернет, без необходимости скачивания и установки на локальный компьютер. Программа физически размещается на сервере (сайте) в сети, пользователь работает с ней через браузер, а вся обработка информации проводится серверным процессором.

Такие системы серьезно облегчают повседневную жизнь. Одни являются близкими родственниками обычных компьютерных программ, например, графические редакторы в интернете повторяют возможности Photoshop. Другая группа систем совсем не имеет аналогов и не может быть представлена в ином виде, к примеру, email рассылки. К ним же относятся различные специализированные проекты - инструменты для вебмастеров, такие как аудиты сайтов, рекламные биржи, сервисы учета и аналитики и т. д. Третьи заменяют походы в различные учреждения, например, онлайн сервисы, позволяющие оплачивать коммунальные услуги, штрафы, государственные пошлины и налоги. От этих платежей никогда не убежать, и возможность оплатить их через интернет экономит время.

Актуальность курсового проекта состоит в том, что в настоящее время нет сайтов по данной тематике, либо они предоставляют пользователю минимум информации, поэтому зачастую пользователю необходимо пользоваться несколькими сайтами, что не всегда удобно.

Цель – разработать систему для управления чайным магазином.

Задачи:

1. Проанализировать существующие технологии для разработки системы.
2. Рассмотреть составляющие компоненты для создания сайта.
3. Выбрать необходимые фреймворки и библиотеки.
4. Продумать архитектуру проекта.
5. Постепенно реализовать составляющие системы, чтобы в итоге получился качественный инструмент.

Практическая значимость: владелец чайного магазина может управлять всеми аспектами своего бизнеса в одном месте в любое время и с любого устройства, поддерживающего выход в сеть. Сможет провести анализ продаж за необходимый период и оптимизировать свои бизнес-процессы.

# Анализ программных технологий для разработки web-сервиса

# Backend технологии

Backend разработка предполагает использование инструментов, фреймворков и других средств для разработки самого движка сайта. Движок всегда составляется из нескольких базовых компонентов, которые неразрывно связаны с сервером, базами данных, корнем сайта, а также сторонними и внутренними ресурсами.

* Node.js – платформа серверного типа, рассматривая которую нельзя не отметить тот факт, что она удовлетворяет большинство требований разработчиков и базируется на популярном языке программирования JavaScript. Она имеет немаловажную особенность – движок V8.

На этом движке базируется большинство всемирно известных проектов – например, браузер Chrome и другие браузерные программы, разработанные на его основе. На данной платформе именно благодаря вышеуказанному фреймворку можно написать код, который будет выполняться на сервере. Этим определяется высокая ценность платформы для программистов.

* Go (или Golang) – это язык программирования, разработанный Google, который сочетает в себе простоту, производительность и эффективность. Go идеально подходит для создания высоконагруженных backend-систем благодаря своей высокой скорости выполнения и низкому потреблению ресурсов. Он часто используется для разработки микросервисов, API и распределенных систем. Go также предоставляет встроенную поддержку многопоточности (горутины), что делает его отличным выбором для задач, требующих параллельной обработки данных.
* Flutter – это платформа с открытым исходным кодом, построенная на языке Dart, который быстро набирает популярность. Ее основные функции – это выполнение кода точно в срок, рендеринг на ходу без промежуточного слоя и виджеты, специфичные для дизайна, для взаимодействия с машинным кодом. Все это обеспечивает быструю разработку кроссплатформенных приложений, которые работают не только на iOS и Android – Flutter также охватывает Windows, Linux, Mac и Google Fuchsia.
* C# – это объектно-ориентированный язык программирования, разработанный компанией Microsoft. Он является частью платформы .NET и используется для создания разнообразных приложений, включая веб-приложения, настольные приложения и мобильные приложения. C# сочетает в себе простоту и мощь, предоставляя разработчикам широкий набор инструментов для создания высокопроизводительных и надежных приложений. Одной из ключевых особенностей C# является его интеграция с .NET Framework, что позволяет легко использовать библиотеки и компоненты, разработанные для этой платформы. Кроме того, C# поддерживает асинхронное программирование, что делает его отличным выбором для разработки приложений, требующих высокой производительности и масштабируемости.

# Базы данных

* MySQL работает на Linux, Windows, OSX, FreeBSD и Solaris. Можно начать работать с бесплатным сервером, а затем перейти на коммерческую версию. Эта система управления базами данных использует стандартную форму SQL. Утилиты для проектирования таблиц имеют интуитивно понятный интерфейс. MySQL поддерживает до 50 миллионов строк в таблице. Поддерживает секционирование и репликацию, а также Xpath и хранимые процедуры, триггеры и представления.
* Microsoft SQL Server – самая популярная коммерческая СУБД. Она привязана к Windows. И графический интерфейс, и программное обеспечение основаны на командах. Поддерживает SQL, непроцедурные, нечувствительные к регистру и общие языки баз данных.
* PostgreSQL – масштабируемая объектно-реляционная база данных, работающая на Linux, Windows, OSX и некоторых других системах. В PostgreSQL 10 есть такие функции, как логическая репликация, декларативное разбиение таблиц, улучшенные параллельные запросы
* MongoDB – cамая популярная NoSQL система управления базами данных. Лучше всего подходит для динамических запросов и определения индексов. Гибкая структура, которую можно модифицировать и расширять. Поддерживает Linux, OSX и Windows, но размер БД ограничен 2,5 ГБ в 32-битных системах. Использует платформы хранения MMAPv1 и WiredTiger.

# Frontend технологии

Frontend разработчик превращает дизайн-макет в страницу, которая открывается и работает в браузере. Также он программирует логику поведения визуальных элементов, получает данные от бэкенда, организует их предоставление пользователю. Он может собирать данные от пользователя, программировать лёгкую обработку этих данных или отправку на бэкенд для дальнейшей обработки.

* React – современный React с открытым исходным кодом отличается своей виртуальной объектной моделью документов (DOM), которая предлагает исключительные функциональные возможности. React – идеальная платформа для тех, кто ожидает большой трафик и нуждается в стабильной платформе для его поддержания.
* Vue один из самых популярных в настоящее время фронтенд фреймворков. Он прост и понятен, в сравнении с запудренным Angular. Кроме малых размеров, его основными преимуществами являются визуальный DOM, компонентный подход и двустороннее связывание данных. Vue универсален, многозадачен и может легко обрабатывать как простые, так и динамические процессы.
* Angular – единственный из списка фреймворков, основанный на TypeScript. Официально запущенный в 2016 году, Angular был разработан компанией Google для преодоления разрыва между растущими технологическими требованиями и традиционными концепциями, что в итоге дало свои результаты.

# Подготовительные работы для разработки

# Выбор ферймворков и библиотек

Основу составляют четыре технологии: PostgreSQL, Node.js (NestJS), React (MUI).

PostgreSQL — это продвинутая, открытая объектно-реляционная система управления базами данных (СУБД), поддерживающая широкий спектр типов данных и расширенные функции SQL. Она известна своей надежностью, высокой производительностью и гибкостью, что делает её популярным выбором для малых и крупных приложений. PostgreSQL поддерживает такие расширенные типы данных, как JSON, XML и hstore, а также полнотекстовый поиск и пространственные данные с расширением PostGIS.

Node.js — это среда выполнения, позволяющая выполнять JavaScript-код на серверной стороне. Она использует модель событийного, неблокирующего ввода-вывода, что делает её легкой и эффективной, идеально подходящей для приложений, обрабатывающих большие объемы данных в режиме реального времени. NestJS — это прогрессивный фреймворк для Node.js, который строится на принципах модульности, обеспечивая структуру для создания масштабируемых и поддерживаемых серверных приложений. NestJS использует TypeScript, что повышает её надежность и устойчивость.

React — это JavaScript-библиотека для создания динамических пользовательских интерфейсов. Она позволяет разработчикам создавать крупные веб-приложения, которые могут изменять данные без перезагрузки страницы. Виртуальный DOM React улучшает скорость и производительность веб-приложений. Material-UI (MUI) — это популярная библиотека компонентов для React, реализующая материал-дизайн от Google и предоставляющая полный набор компонентов, упрощающих создание эстетически привлекательных и высокофункциональных пользовательских интерфейсов.

# Архитектура приложения

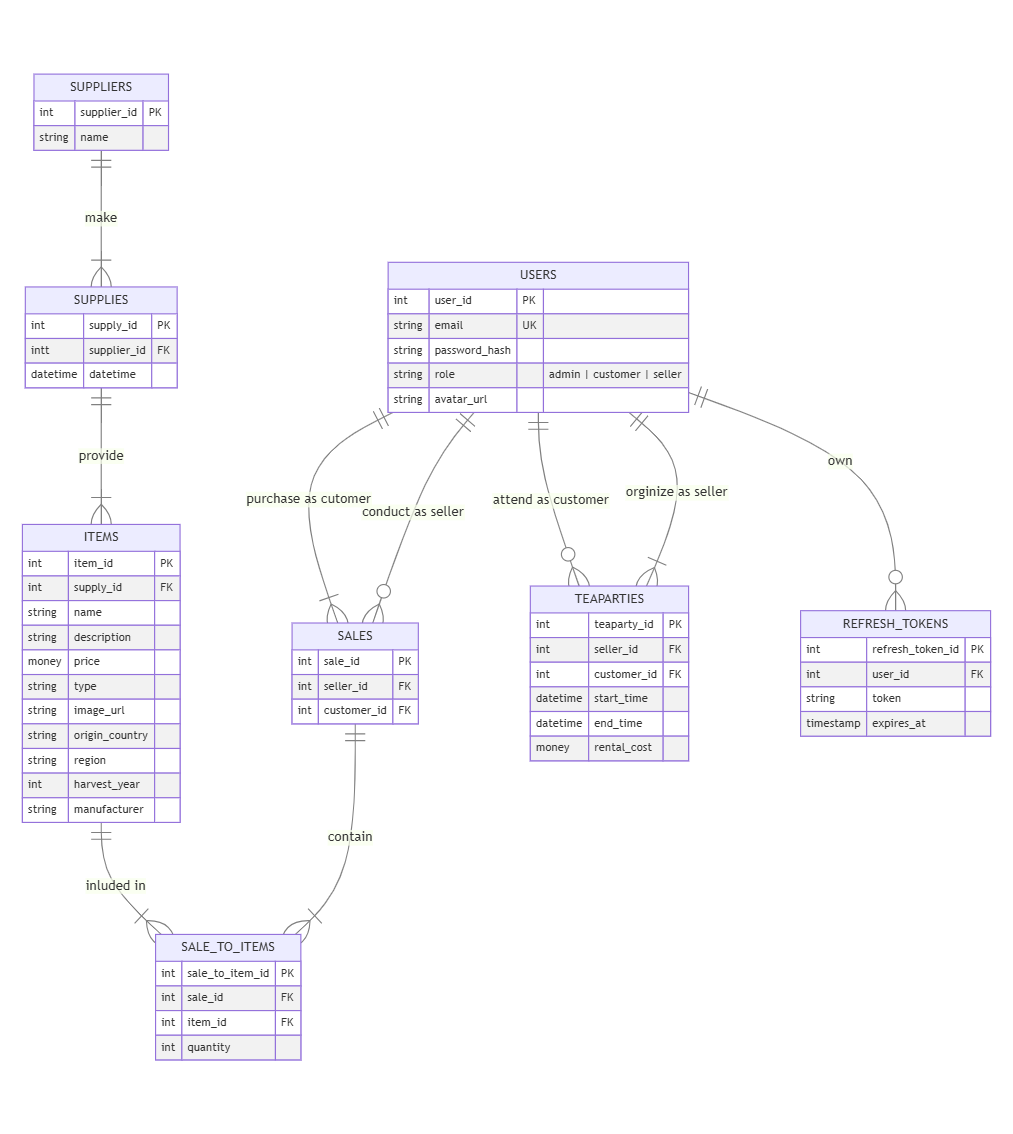


Рисунок 1 – ER-диаграмма

1. Описание сущности Item



1. Описание сущности SaleToItem



1. Описание сущности Sale



1. Описание сущности User



1. Описание сущности RefreshToken



# Реализация общения между клиентом и сервером

Чтобы поддержать соответствие отправляемых и принимаемых данных, я завёл контракты – это общие модели данных, которые используются как на сервере, так и на клиенте.

1. Frontend



1. Backend



# Реализация web-сервиса

Основные этапы разработки web-сервиса включают создание инфраструктуры, обеспечение функциональности и настройку деплоя.

Для управления доступом и обеспечения безопасности пользователей используется JSON Web Token (JWT). Пользователь регистрируется, логинится, и сервер выдает ему JWT, который используется для доступа к защищенным ресурсам.

Основные функции для работы с каталогом товаров включают пагинацию (разбиение списка товаров на страницы), поиск по ключевым словам, фильтрацию товаров по различным параметрам (например, тип, год сбора, производитель) и сортировку по различным критериям (например, по цене или популярности).

Администраторская панель предоставляет доступ ко всем функциям управления магазина: добавление и редактирование товаров, управление заказами, пользователями.

Каталог показывает все доступные товары с возможностью их добавления в корзину. Корзина отображает выбранные товары и позволяет оформить заказ. На странице пользователя отображается личная информация, история заказов и настройки профиля.

Для хранения изображений товаров используется S3. Картинки загружаются на S3, и ссылки на них сохраняются в базе данных магазина.

Для деплоя используется Nginx в качестве обратного прокси-сервера, а Docker Compose для создания и управления контейнерами с приложением и его зависимостями. Это обеспечивает легкость масштабирования и управления сервисом.

# Литература

1. Documentation // NestJS - A progressive Node.js framework. URL: https://docs.nestjs.com/ (дата обращения: 25.08.2024).
2. Quick Start // React. URL: https://react.dev/learn (дата обращения: 20.09.2024).
3. React components that implement Material Design // Material UI. URL: https://mui.com/material-ui/ (дата обращения: 20.09.2024).