Министерство образования и науки Российской Федерации

Муромский институт (филиал)

федерального государственного бюджетного образовательного учреждения

высшего профессионального образования

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

Факультет ФИТР

Кафедра ПИн

КУРСОВАЯ РАБОТА

по дисциплине Системы управления базами данных

Тема Сервис доставки грузов

Руководитель

Колпаков А.А.

(оценка

(подпись) (дата)

Члены комиссии Студент ПИНз-120

(группа)

Чернышев А.Е.

(подпись) (дата)

(подпись) (дата) (подпись) (дата)

Муром 2024 г.

Министерство науки и высшего образования Российской Федерации

**Муромский институт (филиал)**

федерального государственного бюджетного образовательного учреждения высшего образования

**«Владимирский государственный университет**

**имени Александра Григорьевича и Николая Григорьевича Столетовых»**

**(МИ ВлГУ)**

# Факультет ИТР

# «УТВЕРЖДАЮ»

# Зав. кафедрой ПИн \_\_\_\_\_\_\_\_\_\_\_А.Л. Жизняков

(подпись)

# « \_\_ » \_\_\_\_\_\_\_\_ 2023 г.

**ЗАДАНИЕ**

**На курсовой проект по курсу**  *Системы управления базами данных.*

Студент  *Чернышев А.Е.*

1. Тема проекта  *Сайт библиотеки .*

2. Сроки сдачи студентом законченного проекта  *«20» \_\_05\_\_ 2023 г.*

3. Исходные данные к проекту

*1. Локальный сервер (Apache, PHP, MySQL, и т.д.) и программная оболочка;*

|  |
| --- |
| *2. Создать базу данных текстов, информации об изображениях и разделах;* |
| *3. Использовать загрузку и выгрузку информации из базы данных на страницу* |
| *сайта, при помощи языка PHP и/или JavaScript;* |
| *4. Динамическое изменение внешнего вида и/или информации на странице сайта* |
| *5. Обязательно использование комментариев к функциям динамического* |
| *ввода/вывода* |
| *6. На сайте должно быть не менее 5 страниц.* |
| *7. БД должна содержать не менее 20 записей.* |
|  |
| *На сайте должна быть информация о библиотеке, список книг, список карточек* |
| *читателей. Сотрудник может осуществлять резервирование книги.* |
|  |

4. Содержание расчетно‑пояснительной записки (перечень подлежащих разработке вопросов)

*Аннотация (на двух языках)*

|  |
| --- |
| *Введение* |
| *1. Анализ технического задания* |
| *2. Описание языка PHP и JavaScript* |
| *3. Выбор и обоснование архитектуры системы* |
| *4. Выбор и обоснование алгоритмов* |
| *5. Описание программы* |
| *Заключение* |
| *Список используемой литературы* |

5. Перечень графического материала (с точным указанием обязательных чертежей и графиков)

*структурная или функциональная схема сайта*

|  |
| --- |
| *описание структуры базы данных* |
| *текст шаблонов и статических страниц с комментариями* |
|  |

6. Рекомендуемая литература

|  |
| --- |
| *1. Флойд, К. С. Введение в программирование на PHP5 : учебное пособие / К. С. Флойд. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 280 c. — ISBN 978-5-4497-0886-1. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/101998.html (дата обращения: 22.09.2022). — Режим доступа: для авторизир. пользователей* |
| *2. Титов, В. А. Разработка WEB-сайта средствами языка HTML : учебное пособие / В. А. Титов, Г. И. Пещеров. — Москва : Институт мировых цивилизаций, 2018. — 184 c. — ISBN 978-5-9500469-3-3. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/80643.html (дата обращения: 04.02.2023). — Режим доступа: для авторизир. пользователей* |
| *3. Беликова, С. А. Основы HTML и CSS: проектирование и дизайн веб-сайтов : учебное пособие по курсу «Web-разработка» / С. А. Беликова, А. Н. Беликов. — Ростов-на-Дону, Таганрог : Издательство Южного федерального университета, 2020. — 174 c. — ISBN 978-5-9275-3435-7. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/100186.html (дата обращения: 04.02.2023). — Режим доступа: для авторизир. пользователей* |

7. Дата выдачи задания  *04.02.2023*

8. Календарный график работы над проектом (на весь период проектирования, с указанием сроков выполнения и трудоемкости отдельных этапов)

|  |
| --- |
| *Введение, анализ ТЗ, описание языка* 10%, 5 нед |
| *Выбор и обоснование архитектуры системы и алгоритмов* 20%, 6 нед. |
| *Программирование* 60%, 10 нед |
| *Тестирование* 80%, 12 нед. |
| *Оформление пояснительной записки* 100%, 15 нед. |
|  |

**Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

(подпись)

Задание принял к исполнению (дата)

**Подпись студента \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

В данной курсовой работе представлена разработка сайта Библиотеки. Для разработки были использованы технологии PHP, HTML, CSS, JavaScript.

This course work presents the development of the Library website. PHP, HTML, CSS, JavaScript technologies were used for development

Введение

Бурное развитие информационных технологий обусловило необходимость выработки новых подходов к решению проблем автоматизации деятельности различных организаций, служб, предприятий. Работа таких организаций связана с накоплением большого количества информации, в частности, большие объёмы информации необходимо обрабатывать различным службам, фиксирующим информацию об абонентах телефонной сети.

В недавнем прошлом, информацию такого рода хранили в картотеках, используя «бумажные» технологии. И, разумеется, они занимали довольно большие объёмы. Вся информация заносилась вручную на бумажные карточки, что создавало большие неудобства, затрачивалось большое количество времени и, естественно, работа такого вида сама по себе была очень кропотливой. При необходимости узнать информацию требовалась затратить время. Сама картотека изнашивалась по ходу работы, то есть через некоторое время бумага могла утратить своё былое качество.

Сейчас же, в век компьютерных технологий и больших прогрессов в этой сфере, на смену неудобного прошлого пришли очень удобные, надёжные, быстродействующие компьютерные системы, но при этом часто возникают ситуации, когда трудно осуществить быстрый отбор нужной информации. Немаловажен вопрос надежности хранения и конфиденциальности различных сведений. Для решения таких задач используется специальное программное обеспечение, часто объединенное в крупные информационно-справочные системы.

Информационно-справочные системы предназначены для управления различными информационными и справочными массивами, в число которых входят базы данных. Эти системы организовывают определенный интерфейс пользователя с функциями обработки информации самой системы. Набор возможностей таких систем обычно включает в себя коррекцию структуры используемого объекта (инкапсулирующего информацию), работа с его составляющими - дополнение, редактирование, удаление (в случае, когда роль такого объекта выполняет база данных, это - записи), простой просмотр, поиск элементов. Также необходимо обеспечить для такой системы и функции надежного постоянного хранения информации. Программная реализация таких программных продуктов конкретизирует абстрактные операции в конкретные процедуры и функции (или же объекты) и зависит полностью от возможностей языка программирования, поставленной задачи и искусства программиста.

Таким образом, автоматизация процесса обработка различной информации, а особенно картотеки библиотеки, является нужным и перспективным процессом.

Цель курсовой работы – разработка программы для сервиса доставки грузов, который будет обеспечивать удобный интерфейс для взаимодействия с заказами на доставку, упростить поиск заказов и связанных с ними грузов, предоставлять пользователям информацию о заказах и статусе их выполнения.

1 Анализ технического задания

* 1. Описание предметной области

Предметная область данной курсовой работы – служба доставки грузов. Служба доставки грузов оперирует заказами на доставку и грузами, переданными для доставки. Разрабатываемый сервис должен представлять собой систему, позволяющую вести учет осуществляемых доставок и позволять взаимодействовать с заказами на доставку. Основной задачей разработки приложения является обеспечение эффективного взаимодействия с пользователями, предоставление информации о заказах, организация контроля доставок.

Для систем выполняющих роль агрегирования данных неотъемлемой частью является наличие базы данных, в качестве центрального звена. В настоящее время выбор баз данных огромен. Большой популярностью среди реляционных баз данных пользуются: MySQL, PostgreSQL, СУБД компании REDSOFT.

Для разработки клиент-серверного взаимодействия используют большое количество различных комбинаций языков программирования и технологий. В рамках данного проектирования приложение будет представлять собой web-приложение. В современных web-приложениях происходит разделение на клиентскую и серверную часть приложения. Для клиентских приложений разработано большое количество фреймворков: React, Angular, Svelt, Solid, Vue, Nuxt, Nest и т.д. Серверную часть приложений на данный момент можно реализовать практически на всех известых языках программирования: Java и Spring, C# и ASP.Net, python и Django, FastAPI, javascript и express, next. В целом прослеживается тенденция к универсализации языков программирования и технологий, написанных для этих языков. Поэтому выбор технологий для написания того или иного проекта будет зависеть от совершенно различных факторов, начиная от личных предпочтений компании разработчика или компании заказчика, до возможности нанять компетентных программистов, разбирающихся в данной конкретной технологии, и обоснования применения той или иной технологии для осуществления поставленной задачи, с учетом решения вопросов нагрузки на сервис, количества одновременных запросов, или, там где это необходимо, быстродействия системы.

1.2 Формирования требований к программе

На основании произведенного анализа предметной области, сформированы требования к разрабатываемой программе:

1. Главная страница сайта является информационной страницей, отвечающая за представление пользователям информацию о сервисе.
2. Страница авторизации, представляющая собой форму для ввода данных для авторизации пользователя.
3. Главная страница личного кабинета, предоставляющая историю заказов. Данная страница должна иметь возможность поиска заказов по параметрам: отправитель, получатель, пункт отправки, пункт получения, выполненные и заказы в процессе доставки.
4. Страница информации о заказе, предоставляющая полностью информацию об отправителе, получателе заказа, а также информацию о грузах и статусе исполнения.
5. Страница с списком пунктов выдачи, включает в себя сам список и механизм фильтрации и поиска.
6. Страница создания/редактирования пункта выдачи.
7. Страница просмотра списка пользователей, включающая в себя список пользователей и механизмы поиска и фильтрации.
8. Страница создания/редактирования информации о пользователе.

1.3 Анализ системных требований

Приложение будет состоять из двух частей: клиентское приложение (nuxtjs) и серверное (node + expressjs). Преимущество использования данной связки nuxt + express является наличие единой кодовой базы, что позволяет программисту разрабатывать полноценное приложение без смены контекста языка программирования. В качестве базы данных будет использоваться MySQL, как достаточно простая и при этом надежная база данных, предоставляющая весь необходимый функционал для приложения.

Использование связки nuxt + express, работающих на nodejs, позволит приложение быть минимально зависимым от выбранной платформы для запуска приложения. Основным условием для запуска будет служить возможность установки на операционной системе nodejs и базы данных MySQL.

Для того чтобы сделать приложение еще менее зависимым от операционной системы, приложение будет иметь разработанный docker-compose, позволяющий запускать приложение на любой операционной системе, поддерживающей docker. Использование Docker позволит быть заказчику уверенным в том, что не зависимо от используемой операционной системы приложение будет вести себя одинаково, так как докер позволяет явно указать окружение, в котором будет выполняться приложение.

Минимальные же системные требование, предъявляемые к системе, для удовлетворения характеристик быстродействия следующие:

- Процессор с частотой 1ГГц или быстрее, или система на кристалле SoC;

- ОЗУ 1Гб для 32-разрядных систем или 2 Гб для 64-разрядных систем;

- Место на жестком диске от 8 Гб;

- Доступ к сети интернет с пропускной способностью 10Мвьит в сек или быстрее.

Поскольку сервис представляет собой Web-приложение, то для потребителя будет достаточно иметь устройство, имеющее выход в интернет и поддерживающее открытие web-страниц с использованием javascript.

2. Разработка моделей данных

При проектировании базы данных необходимо провести анализ информационных потоков, и уже на основе этого анализа построить концептуальную модель. Это значительно облегчит процесс программной реализации.

В данной работе при построении модели данных предметной области используется модель «сущность-связь». Сущность – это объект любой природы, данные о котором хранятся в базе данных. Данные о сущности хранятся в отношении. Каждая сущность имеет свой набор атрибутов. Атрибуты представляют собой свойства, характеризующие сущность.

Исходя из анализа предметной области была разработана концептуальная схема базы данных (рисунок 1). Из которой видно, что в приложении есть основные сущности: пользователь, заказ, склад и груз. Каждой сущности присущ свой набор атрибутов, характеризующий ее.

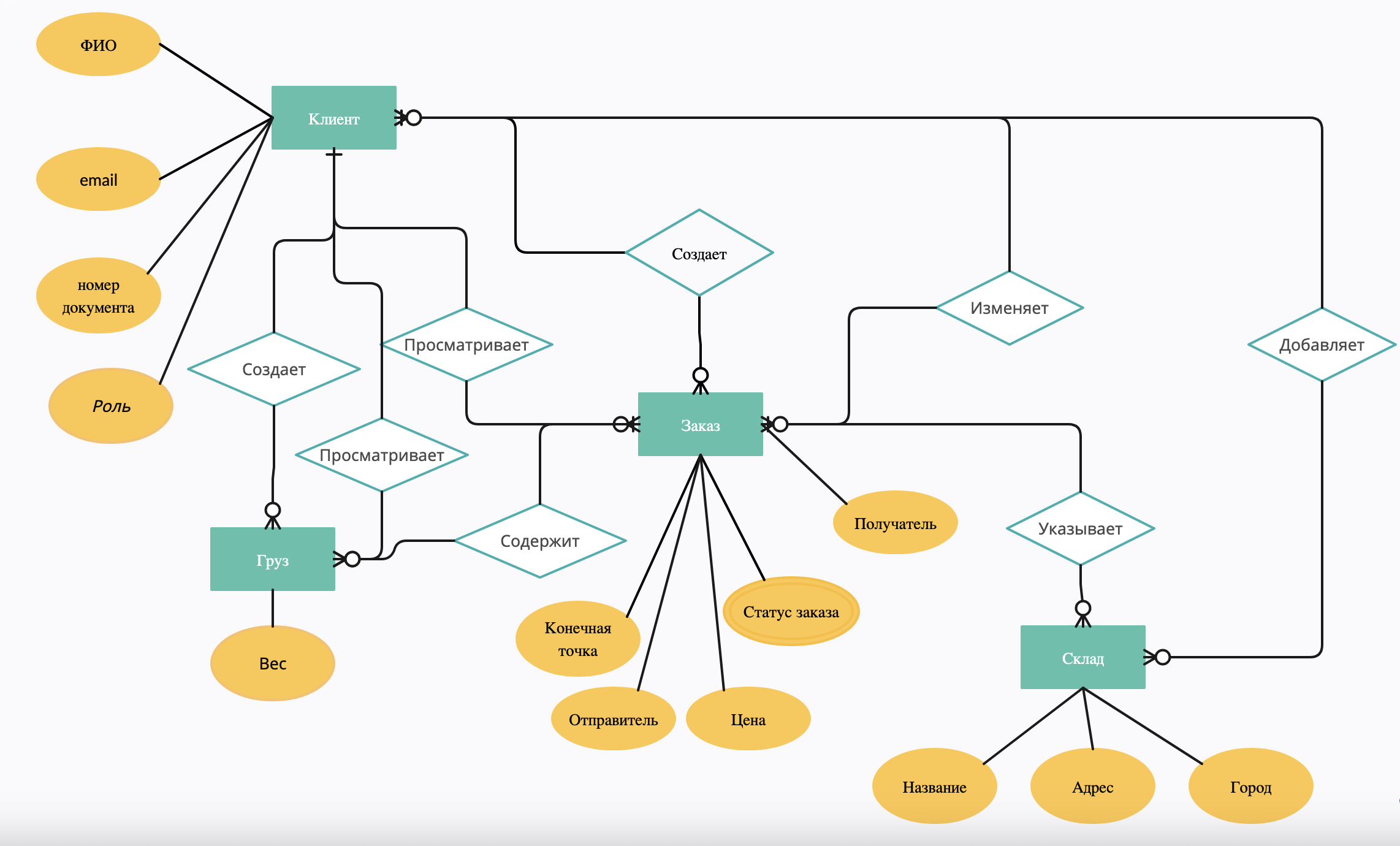


Рисунок 1 – Концептуальная схема.

Используя разработанную концептуальную модель базы данных, можно разработать логическую модель. С помощью логической модели данных можно визуализировать операционные или транзакционные процессы на диаграмме взаимоотношения между сущностями. Логические модели данных определяют, как работают и взаимодействуют объекты данных, понятным для заинтересованных представителей бизнеса способом. Разработка логической модели производится независимо от реальной базы данных, в которой позже будут развернуты данные.