

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

**«МИРЭА – Российский технологический университет»**

**РТУ МИРЭА**

**Институт информационных технологий (ИТ)**

**Кафедра инструментального и прикладного программного обеспечения (ИиППО)**

**КУРСОВАЯ РАБОТА**

по дисциплине: Разработка серверных частей интернет-ресурсов

по профилю: Разработка и дизайн компьютерных игр и мультимедийных приложений

направления профессиональной подготовки: 09.03.04 «Программная инженерия»

Тема: Серверная часть веб-приложения «Магазин сладостей»

Студент: Быкова С.Г.

Группа: ИКБО-20-20

Работа представлена к защите ХХ.12.2022 (дата) /Быкова С.Г./

(подпись и ф.и.о. студента)

Руководитель: к.т.н., старший преподаватель, Синицын Анатолий Васильевич

Работа представлена к защите ХХ.12.2022 (дата) /Синицын А.В./

(подпись и ф.и.о. рук-ля)

Оценка по итогам защиты:

/ Синицын А.В. ,ст. п., доц., к.т.н./

/ , /

(подписи, дата, ф.и.о., должность, звание, уч. степень двух преподавателей, принявших защиту)

**М. РТУ МИРЭА. 2022 г.**

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

**Институт информационных технологий (ИТ)**

**Кафедра инструментального и прикладного программного обеспечения (ИиППО)**

**ЗАДАНИЕ**

**на выполнение курсовой работы**

по дисциплине: Разработка серверных частей интернет-ресурсов

по профилю: Разработка и дизайн компьютерных игр и мультимедийных приложений

направления профессиональной подготовки: Программная инженерия (09.03.04)

Студент: Быкова Софья Георгиевна

Группа: ИКБО-20-20

Срок представления к защите: 09.12.2022

Руководитель: Синицын Анатолий Васильевич, к.т.н., старший преподаватель

**Тема:** Серверная часть веб-приложения «Магазин сладостей»

**Исходные данные:** используемые технологии: HTML5, CSS3, Java, Jetbrains Intellij IDEA, SQL, наличие: межстраничной навигации, внешнего вида страниц, соответствующего современным стандартам веб-разработки, использование паттерна проектирования (MVC, Clear Architecture, DDD). Нормативный документ: инструкция по организации и проведению курсового проектирования СМКО МИРЭА 7.5.1/04.И.05-18.

**Перечень вопросов, подлежащих разработке, и обязательного графического материала:** 1. Провести анализ предметной области разрабатываемого веб-приложения. 2. Обосновать выбор технологий разработки веб-приложения. 3. Разработать архитектуру веб-приложения на основе выбранного паттерна проектирования. 4. Реализовать слой серверной логики веб-приложения с применением выбранной технологии. 5. Реализовать слой логики базы данных. 6. Разработать слой клиентского представления веб-приложения 7. Создать презентацию по выполненной курсовой работе.

Руководителем произведён инструктаж по технике безопасности, противопожарной технике и правилам внутреннего распорядка.

Зав. кафедрой ИиППО: \_\_\_\_\_\_\_\_\_\_\_/Р. Г. Болбаков/, «\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_2022 г.

Задание на КР выдал: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/А. В. Синицын/, «\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_2022 г.

Задание на КР получил: \_\_\_\_\_\_\_\_\_\_\_/С. Г. Быкова/, «\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_2022 г.

**АННОТАЦИЯ**

Отчет содержит: 31 страницу, 18 рисунков, 1 листинг и 8 информационных источников.

Курсовая работа посвящена разработке интернет-ресурса на тему «Магазин сладостей». Во время ее разработки были созданы 12 страниц формата «.html». Были использованы технологии HTML5, CSS3, JavaScript, Java, Spring Boot, MySQL, Intellij IDEA для разработки данного веб-приложения.

Были реализованы слои логики работы приложения, взаимодействия с базой данных и клиентского представления.

**СОДЕРЖАНИЕ**

[АННОТАЦИЯ 3](#_Toc130231728)

[Перечень сокращений 5](#_Toc130231729)

[ВВЕДЕНИЕ 6](#_Toc130231730)

[1. ОБЩИЕ СВЕДЕНИЯ 8](#_Toc130231731)

[1.1. Обозначение и наименование интернет-ресурса 8](#_Toc130231732)

[1.2. Языки и технологии, с помощью которых реализован интернет-ресурс 8](#_Toc130231733)

[2. ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ 9](#_Toc130231734)

[3. ОСНОВНАЯ ЧАСТЬ РАБОТЫ 10](#_Toc130231735)

[3.1. Анализ предметной области 10](#_Toc130231736)

[3.2. Выбор технологий 12](#_Toc130231737)

[3.3. Разработка архитектуры приложения 14](#_Toc130231738)

[3.4. Реализация слоя серверной логики веб-приложения 17](#_Toc130231739)

[3.5. Реализация слоя логики базы данных 18](#_Toc130231740)

[3.6. Разработка слоя клиентского представления веб-приложения 23](#_Toc130231741)

[3.7. Подготовка презентации выполненной курсовой работы 29](#_Toc130231742)

[ЗАКЛЮЧЕНИЕ 30](#_Toc130231743)

**Перечень сокращений**

|  |  |
| --- | --- |
| HTML | ­– Hypertext Markup Language (язык гипертекстовой разметки) |
| CSS | – Cascading Style Sheets (каскадные таблицы стилей) |
| JS | – Java Script (язык программирования) |
| MVC | – Model View Controller |
| XML | – Extensible Markup Language (Расширяемый Язык Разметки) |
| JPA | – Java Persistence API |
| БД/DB | – База данных (Database) |

**ВВЕДЕНИЕ**

Интернет-магазин – это сайт, предназначенный для организации продаж различных товаров и услуг. С развитием информационных технологий создание сайта для магазина приобрело большую популярность. Это объясняется наличием особых условий, упрощающих и улучшающих жизнь как для покупателей, так и для предпринимателей. К примеру, отсутствие необходимости находиться в магазине для выбора или приобретения товара.

Ежегодно люди отмечают такие события как годовщины, дни рождения, международные праздники, личные достижения. Практически на каждом таком событии на столе находятся кондитерские изделия, ведь они выступают в роли высоко эстетической продукции, призванной формировать празднично-радостное настроение, а также многим приходятся по вкусу, потому что их разнообразие очень велико. Актуальность выбранной темы обосновывается ростом числа людей, заинтересованных в покупке кондитерских изделий. Рост спроса же вызван широким спектром возможных сочетаний вкусов и декора.

Целью курсовой работы является разработка, тестирование и отладка серверной части веб-приложения на тему «Магазин сладостей» с применением технологий HTML5, CSS3, Java, Jetbrains Intellij IDEA[1], SQL с организацией межстраничной навигации, внешним видом страниц, соответствующего современным стандартам веб-разработки и с использованием одного из паттернов проектирования (MVC, Clear Architecture, DDD).

Для достижения поставленной цели необходимым является:

* Провести анализ предметной области разрабатываемого веб-приложения;
* Обосновать выбор технологий разработки веб-приложения;
* Разработать архитектуру веб-приложения на основе выбранного паттерна проектирования;
* Реализовать слой серверной логики веб-приложения с применением выбранной технологии;
* Реализовать слой логики базы данных;
* Разработать слой клиентского представления веб-приложения;
* Подготовить презентацию выполненной курсовой работы.

В данной курсовой работе объектом является само веб-приложение.

Предметом исследования является тематика ресурса.

В работе были использованы методы анализа, синтеза.

Работа состоит из аннотации, введения, содержания, трёх основных глав, заключения и списка информационных источников.

**ОБЩИЕ СВЕДЕНИЯ**

**1.1. Обозначение и наименование интернет-ресурса**

Тема данного веб-приложения – «Магазин сладостей». В качестве наименования разработанного продукта выступает выдуманное название магазина – «SONYA’SUGAR».

Прикладное программное обеспечение, необходимое для разработки и функционирования интернет-ресурса

Прикладное программное обеспечение – это программа или совокупность программ, предназначенная для решения определенных пользовательских задач.

Для разработки и функционирования интернет-ресурса использовался текстовый редактор Visual Studio Code[2], интегрированная среда разработки Jetbrains Intellij IDEA.

Дополнительно был использован онлайн сервис Figma[3] для создания макета страниц сайта и Adobe Illustrator[4] для создания некоторых элементов внешнего вида.

**1.2. Языки и технологии, с помощью которых реализован интернет-ресурс**

Интернет-ресурс был реализован на стандартизированном языке разметки документов во Всемирной паутине – HTML. Он является универсальным языком разработки и подходит для решения разнообразных задач.

Также для оформления внешнего вида страниц был использован CSS – формальный язык описания внешнего вида веб-страницы. Главной ролью этой технологии является расширение возможностей документа.

Для описания логики работы приложения использовался язык Java.

Веб-приложений написано при помощи Spring Boot [8] - популярного фреймворка для создания веб-приложений с использованием языка программирования Java.

**2.** **ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ**

Разработанное веб-приложение имеет условно коммерческий характер, поэтому его структура аналогична структуре любого интернет-магазина. Программный продукт предназначен для ознакомления с ассортиментом магазина сладостей, выбора желаемой продукции и оформления заказа.

Веб-приложение содержит информацию об имеющихся в магазине товаров, возможных способах доставки и оплаты, а также позволяет покупателю сформировать корзину и оформить заказ.

**3. ОСНОВНАЯ ЧАСТЬ РАБОТЫ**

**3.1. Анализ предметной области**

Для успешного составления информационной базы исследования и для проектирования пользовательского интерфейса был проведен анализ предметной области разрабатываемого веб-приложения на тему «Магазин сладостей» – анализ сайтов потенциальных конкурентов на рынке.

Для понимания, какого рода контент будет содержать веб-приложение, были использованы метод наблюдения и метод сравнения, посещены такие ресурсы, как [mdshow.ru](https://mdshow.ru) [5] (рисунок 3.1.1), confaelshop.ru [6] (рисунок 3.1.2), sweet-lavka.ru [7] (рисунок 3.1.3).

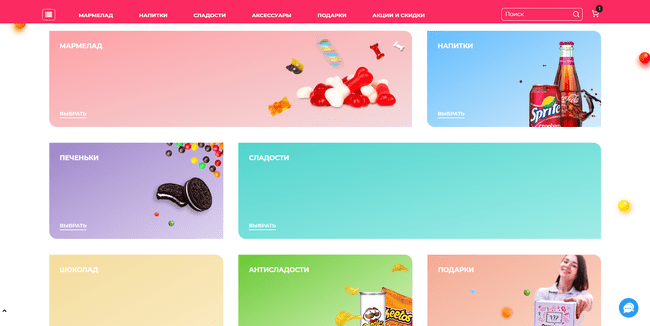


Рисунок 3.1.1 – Сайт [mdshow.ru](https://mdshow.ru)

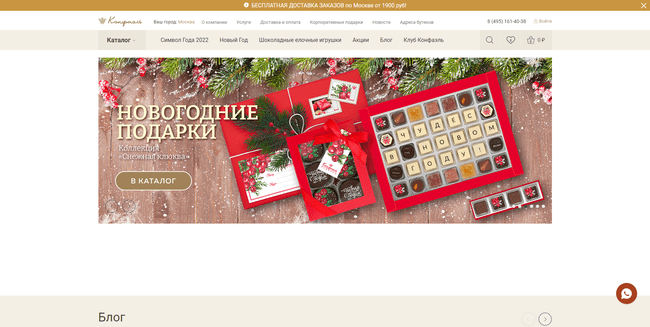


Рисунок 3.1.2 – Сайт confaelshop.ru

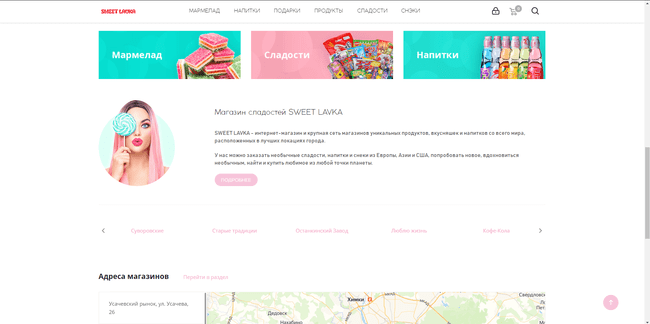


Рисунок 3.1.3 – Сайт sweet-lavka.ru

В ходе исследования было определено следующее:

* со сладостями больше всего ассоциируются розовый и коричневый цвета, вне зависимости от внешнего вида кондитерских изделий;
* необходимо провести сбор информации о видах кондитерских изделий для более точного распределения их по категориям;
* веб-приложение должно включать в себя такие разделы как «Корзина», «Главная», «Оформление заказа», «Мои заказы» и непосредственно сам шаблон раздел с товарами;
* веб-приложение может включать разделы «Каталог», «Избранное».

После сбора информации было принято решение разделить категории на 6 основных подкатегорий: торты, печенье, вафли, шоколад, пончики и кексы – некоторые из которых будут включать в себя деления на конкретные подразделы.

Помимо навигационного меню каждый из разделов имеет свою особую структуру, описание которой для различных разделов приведено ниже.

* Страница «Избранное» должна содержать набор товаров, понравившихся пользователю. В данном разделе должна иметься возможность убрать товар из любимых. Также в некоторых веб-приложениях пользователю может предлагаться отправиться на поиски новых вкусностей для дальнейшего добавления.
* Страница «Корзина» отображает пользователю его ещё не сформированный заказ: товары, их количество, стоимость продукта в данной количестве и общая стоимость. Из «Корзины» должна быть возможность оформить заказ.
* Страница «Оформление заказа» служит связующим элементом между выбором товаров и подготовки к их получению, так как содержит особенности заказа, не связанные с товарами, а именно: способ получения, способ оплаты, данные заказчика, данные для получения и оплаты.
* Страница «Мои заказы» должна содержать все сделанные пользователем заказы. Это могут быть как все заказы (выполненные, в процессе выполнения, отменённые), так и конкретные, например, только выполненные. Я считаю, что приятнее видеть всю историю заказов. Также на такой странице у заказа может располагаться действие его повтора – переход в корзину с тем же набором товаров и при подтверждении готовности оформления теми же данными заказчика, получения и прочее на случай, если пользователь захочет что-то изменить.
* Страница «Каталог» содержит список всех категорий с соответствующими разделами. Данная страница может отсутствовать, если всё её содержимое и так отображено в навигационном меню.
* Страница «Главная» представляет собой визитную карточку магазина, а также может отображать изменения в ассортименте, новинки, скидки и акции.

**3.2. Выбор технологий**

В данной работе использовался следующий ряд технологий:

* Основные технологии вёрстки сайтов – HTML, CSS, JS. Данные технологии используются повсеместно и позволяют гибко и просто настраивать внешний вид интернет-ресурса.
* Intellij IDEA. Данная среда разработки предоставляет удобный интерфейс и достаточно широкий спектр поддерживаемых языков (встроены по умолчанию)
* Язык программирования Java. Является надёжным средством для создания множества сервисов и приложений, а также изучался курсами ранее.
* Фреймворк Spring Boot. Данный фреймворк популярен в области разработки веб-приложений на языке Java. Среди его преимуществ можно отметить упрощённую конфигурацию.
* MySQL. Популярная свободная реляционная система управления базами данных, созданная корпорацией Oracle. Является простой и удобной в использовании.
* Паттерн программирования MVC. Идеальный паттерн для использования со Spring boot, обеспечивает разграниченность области ответственности (модель, контроллер, база данных).

Spring Boot позволяет добавлять в проект фреймворки, имеющие интеграцию с ним, с помощью систем автоматизации сборки. Для создания данного интернет-ресурса использовался следующий ряд компонентов:

* Система автоматизации сборки Maven. Настройка происходит в .pom формате (подмножество XML);
* Thymeleaf. Средства шаблонизации, поддерживающее следующий ряд языков: HTML, CSS, JS, XML;
* Spring Boot Security. Фреймворк для настройки безопасности, регистрации и авторизации;
* Lombok. Процессор аннотаций, который работает во время компиляции для добавления кодов в классы;
* JPA. Спецификация, позволяющая хранить Java-объекты в реляционной базе данных;
* Mail. Фреймворк для отправки сообщений по почте;
* MySQL. Фреймворк для работы с базой данных MySQL;
* Flyway MySQL. Средство для создания миграций на основе ранее созданных описаний классов с помощью Lombok и Java.

**3.3. Разработка архитектуры приложения**

Model, View, Controller (MVC) — паттерн программирования, разделяющий архитектуру приложения на три модуля. Данный шаблон позволяет изменять каждый компонент независимо друг от друга для простой разработки и поддержки веб-приложений (рис. 3.3.1).

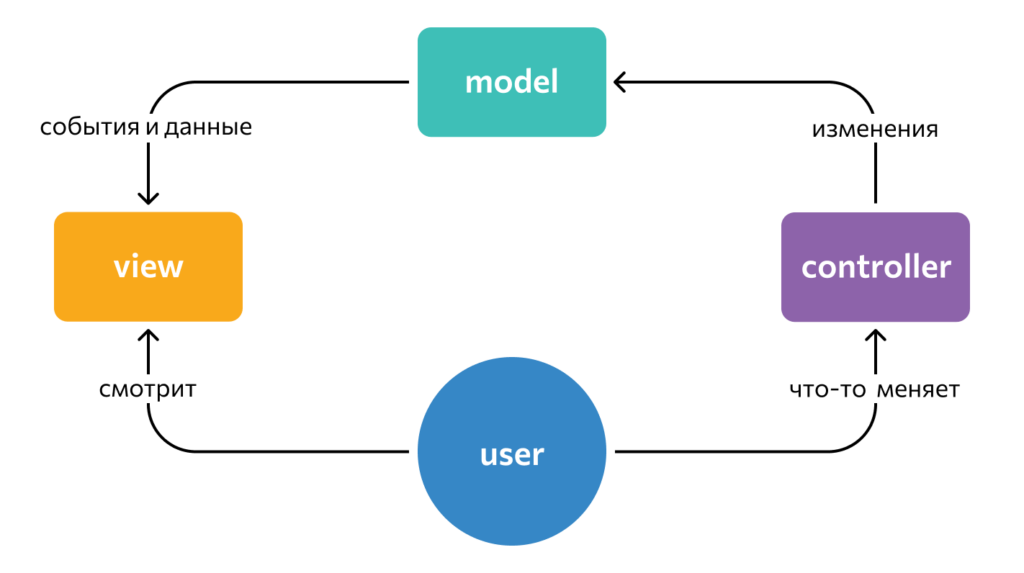


Рис. 3.3.1 – Схема паттерна MVC

В приложении будет присутствовать физическое разграничение модули модели, представления и контроллера. Это выглядит следующим образом (рис. 3.3.2):

* контроллеры хранятся в пакете controller в папке java;
* модель передаётся контроллерам для изменения;
* логика обработки этих изменений (логика моделей) содержится в пакете service;
* представления (статические и шаблонные) хранятся в папке ресурсов (resources).

Также присутствует и другое физическое разграничение – аннотации и особые виды передачи. Контроллеры обязательно помечаются аннотацией @Controller, представления передаются как строка – название представления, а модель – отдельный тип Model или карта Map со строковыми ключами и значениями-объектами.

В данной работе имеется:

* Представление «Регистрация»;
* Представление «Вход»;
* Представление «Главная страница»;
* Представление «Меню товаров»;
* Представление «Меню секции товаров»;
* Представление «Товар»;
* Представление «Корзина»;
* Представление «Оформление заказа»;
* Представление «Настройки»;
* Представление «Изменение данных»;
* Контроллер меню;
* Контроллер товаров;
* Контроллер профиля пользователя;
* Контроллер регистрации;
* Контроллер корзины;
* Общий контроллер;
* Модель обработки корзины;
* Модель обработки товаров;
* Модель обработки почты;
* Модель обработки пользователя.
* База данных содержит следующий ряд сущностей:
* Товар;
* Тип товара;
* Подтип товара;
* Ингредиенты;
* Список ингредиентов товара;
* Заказ пользователя;
* Пользователь;
* Роли;
* Тип доставки;
* Список типов доставки;
* Список товаров заказа;
* Статус заказа.

На рисунке 3.3.3 представлена физическая модель базы данных.

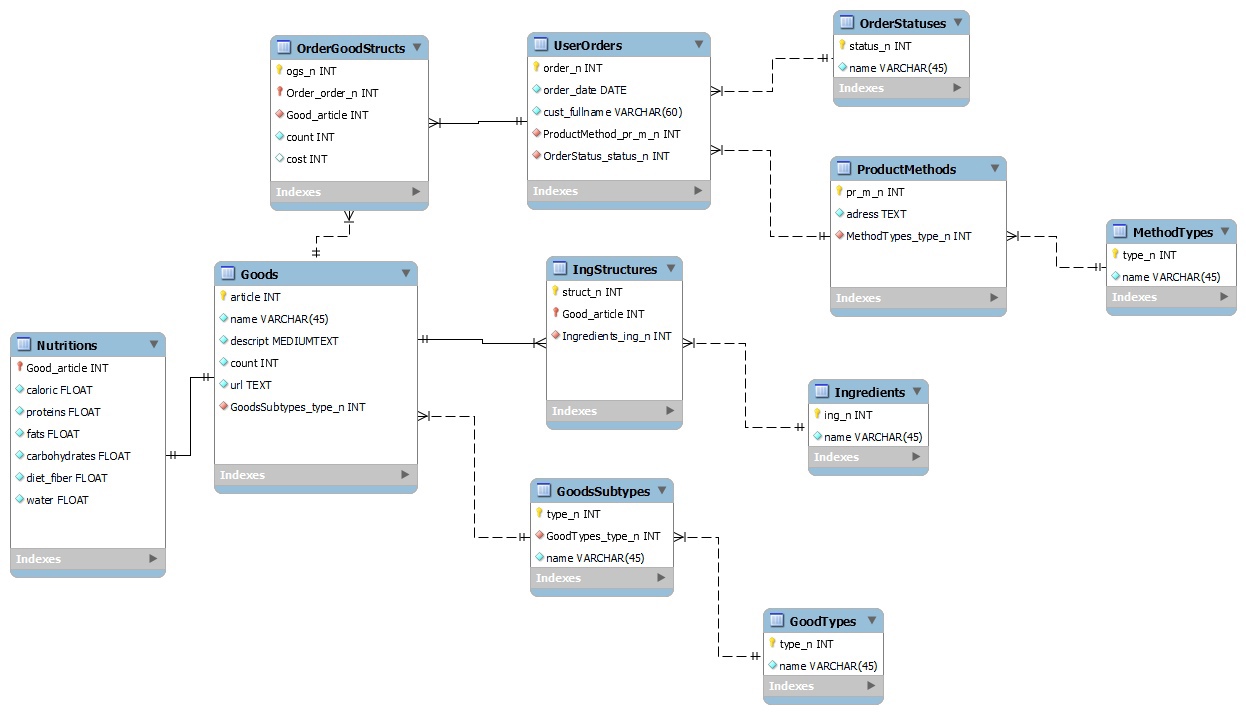


Рис. 3.3.3 – Физическая модель БД

**3.4. Реализация слоя серверной логики веб-приложения**

Организация правильной иерархии пакетов и папок является важным этапом на стадии разработки логики приложения.

Конфигурация приложения размещается в пакете config. Всего в проекте предусматривается четыре объекта конфигурации: MVC, безопасность, почта и менеджер аутентификации (рис. 3.4.1).

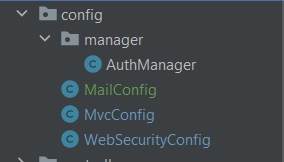


Рис. 3.4.1 – Конфигурации веб-приложения в пакете config

MVC конфигурация позволяет регистрировать контроллеры, которые не созданы вручную.

WebSecurity позволяет настроить права доступа для авторизованных и неавторизованных пользователей.

Mail даёт возможность настроить параметры рассылки.

AuthManager предоставляет возможность аутентифицировать пользователя.

Контроллеры располагаются в пакете controller (рис. 3.4.2). Они отвечают за общение модели и представлений. В данном проекте один контроллер чаще всего отвечает за взаимодействие с несколькими представлениями.

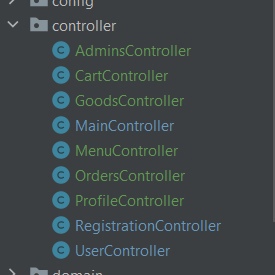


Рис. 3.4.2 – Список контроллеров в пакете controller

Модели находятся в пакете service (рис. 3.4.3).

В названиях моделей и контроллеров содержится информация о том, с какими представлениями они связаны. Например, CartService служит для обработки корзины, а значит связано с представлением cart через контроллер CartController.

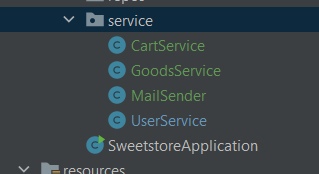


Рис. 3.4.3 – Список моделей в пакете service

**3.5. Реализация слоя логики базы данных**

Корректная работа базы данных очень важна для веб-приложений, поэтому нужно грамотно настраивать связи и взаимодействия между таблицами.

Ниже представлено краткое описание содержимого таблиц и их область применения (рис. 3.5.1).

* Таблица «user\_statuses» содержит наименование статусов заказов пользователя. Является информационной;
* Таблица «user\_orders» содержит пользователя, номер и статус его заказа и информацию о способе доставки;
* Таблица «ingredients» содержит список ингредиентов, которые могут присутствовать в составе товара;
* Таблица «ing\_structures» содержит информацию о находящихся ингредиентах в товарах: какой и где;
* Таблица «goods» содержит информацию о товарах;
* Таблица «order\_good\_structs» содержит информацию о находящихся товарах и их количестве в корзинах пользователей;
* Таблица «method\_types» содержит информацию о типах доставки. Является информационной;
* Таблица «product\_methods» содержит информацию об адресе и методе доставки;
* Таблица «good\_types» содержит название типов товаров. Является информационной;
* Таблица «good\_subtypes» содержит название подтипов товаров и их родителей. Является информационной.
* Таблица «usr» содержит информацию о пользователе;
* Таблица «user\_role» содержит информации об правах пользователя (его роли).

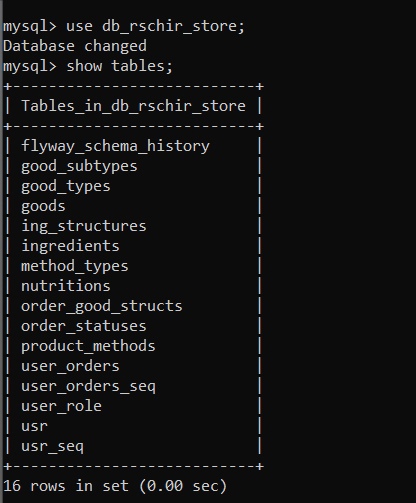


Рис. 3.5.1 – Таблицы в базе данных

За данные таблицы в коде отвечают следующие классы: UserStatuses, UserOrders, ProductMethods, MethodTypes, GoodTypes, GoodSubtypes, Goods, Ingredients, IngStructures, OrderGoodStructs, User, UserRole (рис. 3.5.2).



Рис. 3.5.2 – Java-сущности в пакете domain

Доступ к базе данных со стороны приложения предоставляется за счёт CRUD-репозиториев (рис. 3.5.3) – по одному на каждую сущность – и свойствам, установленным в application.properties (рис. 3.5.4).

CRUD-репозиторий – это интерфейс, содержащий методы для доступа к данным в базе данных. Он позволяет создавать методы на основе специфики названий без внутренней реализации или создавать нативные запросы – написанные самостоятельно.

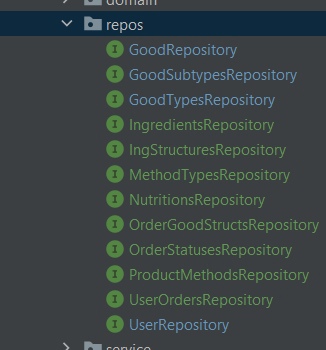


Рис. 3.5.3 – CRUD-репозитории в пакете repos

В свойствах приложения содержится основная информация для его работы и отладки.

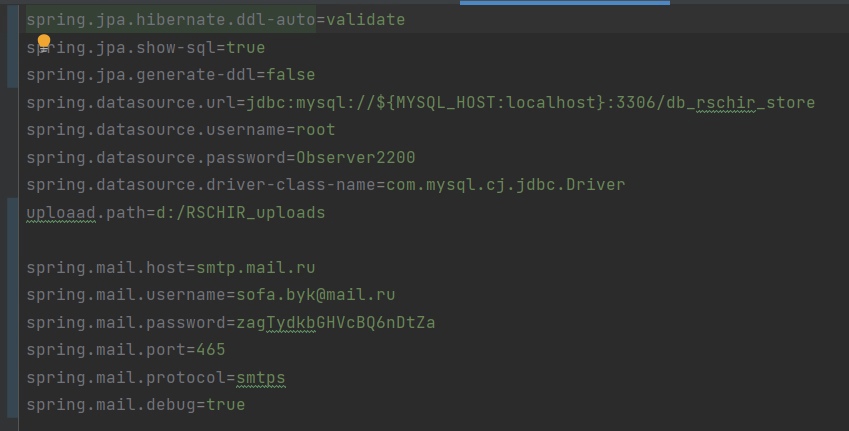


Рис. 3.5.4 – содержимое файла application.properties

Доступ к данным осуществляется при помощи моделей. Ниже представлен листинг 3.5.1 с отрывком кода модели GoodsService для обновления данных ингредиентов.

Листинг 3.5.1 – Код обновления данных ингредиентов

public void updateIngredients(String oldIds, String oldNames, String newNames) {

String[] newIngNames = newNames.split(",");

List<String> allIngsNames = ingsRep.findAllName();

List<Ingredients> willUpdateIngs = getIngredientsFromIdAndName(oldIds, oldNames);

if (willUpdateIngs != null && willUpdateIngs.size() == newIngNames.length) {

for (int i = 0; i < newIngNames.length; ++i) {

if (!allIngsNames.contains(newIngNames[i])) {

Ingredients ing = willUpdateIngs.get(i);

ing.setName(newIngNames[i]);

willUpdateIngs.set(i, ing);

}

}

ingsRep.saveAll(willUpdateIngs);

}

}

**3.6. Разработка слоя клиентского представления веб-приложения**

Слой клиентского представления – это то, через что пользователь взаимодействует с нашим приложением и где видит результат своих действий.

При разработке представлений была использована информация, полученная в результате анализа интернет-ресурсов конкурентов.

Внешний вид прототипов представлений представлен на рисунках 3.6.1–3.6.7.

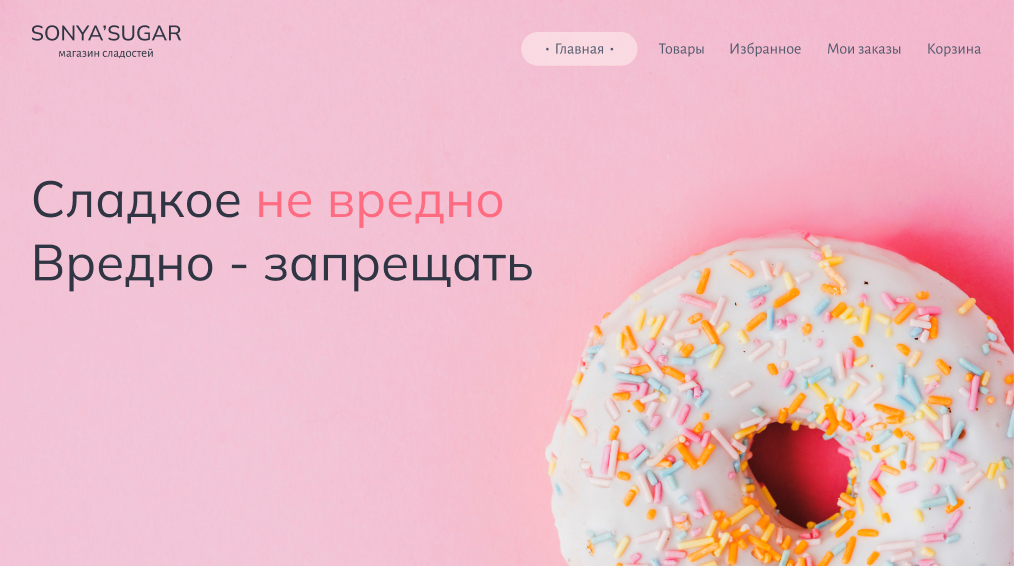


Рис 3.6.1 – Главный экран

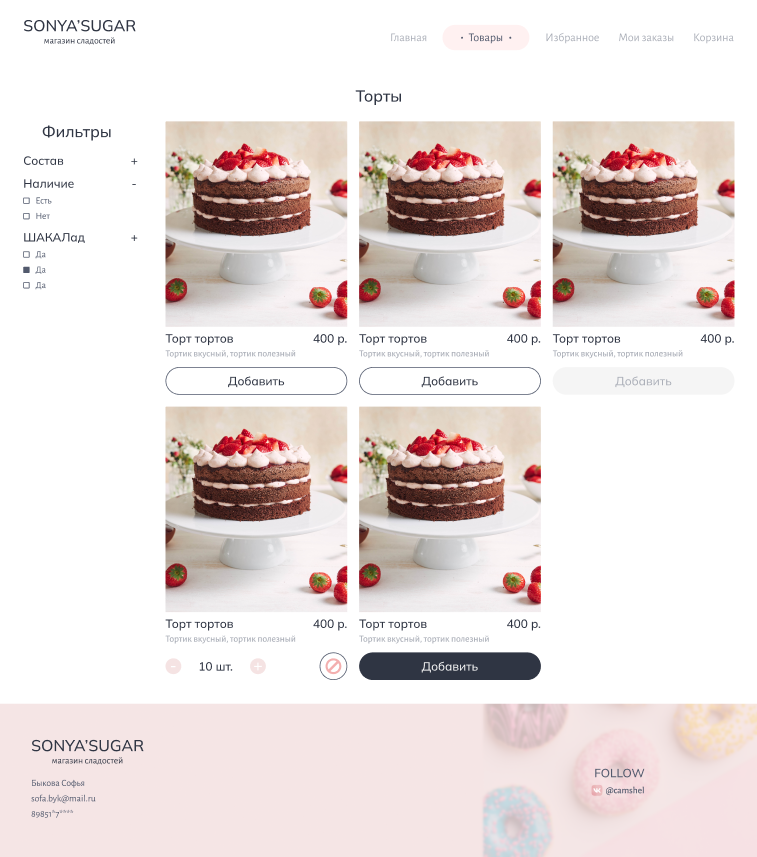


Рис. 3.6.2 – Страница секции товаров

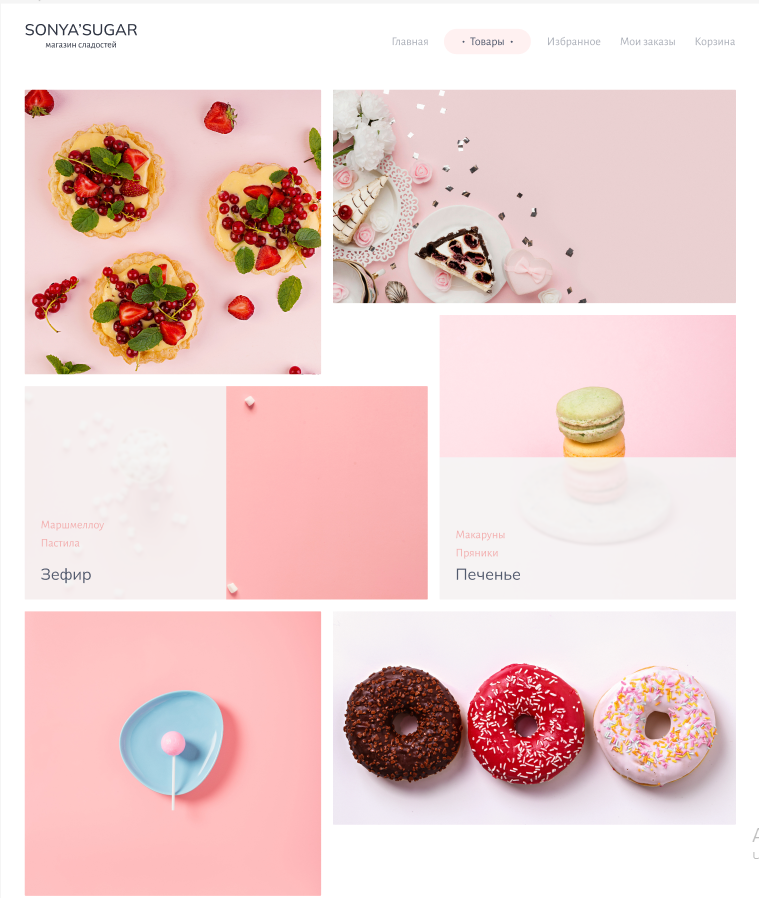


Рис. 3.6.3 – Страница меню категорий

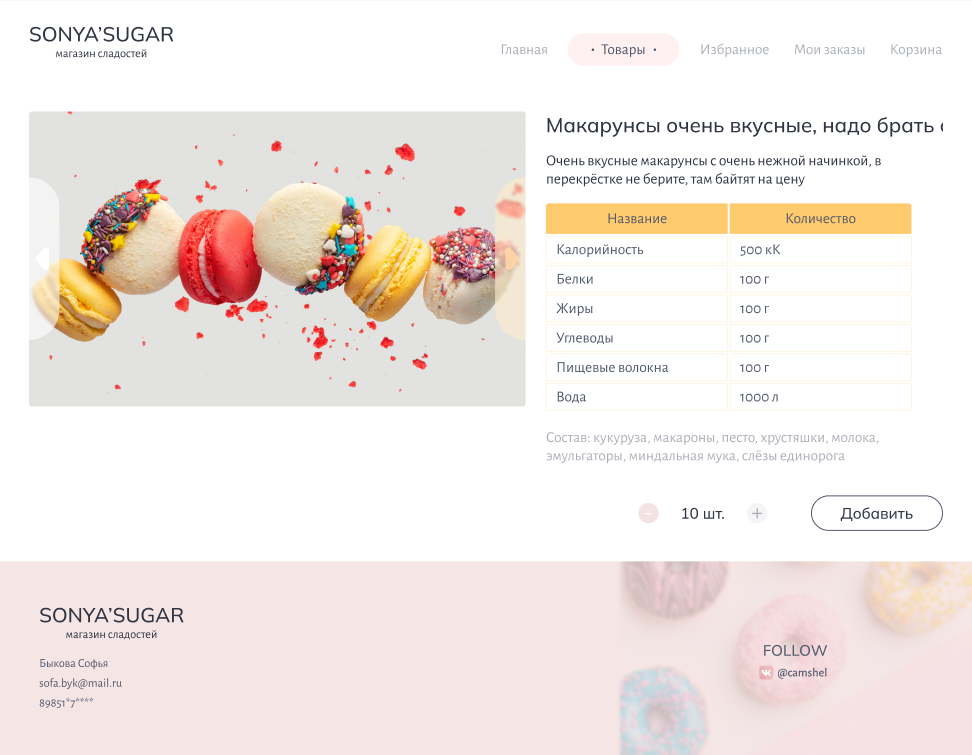


Рис. 3.6.4 – Страница товара

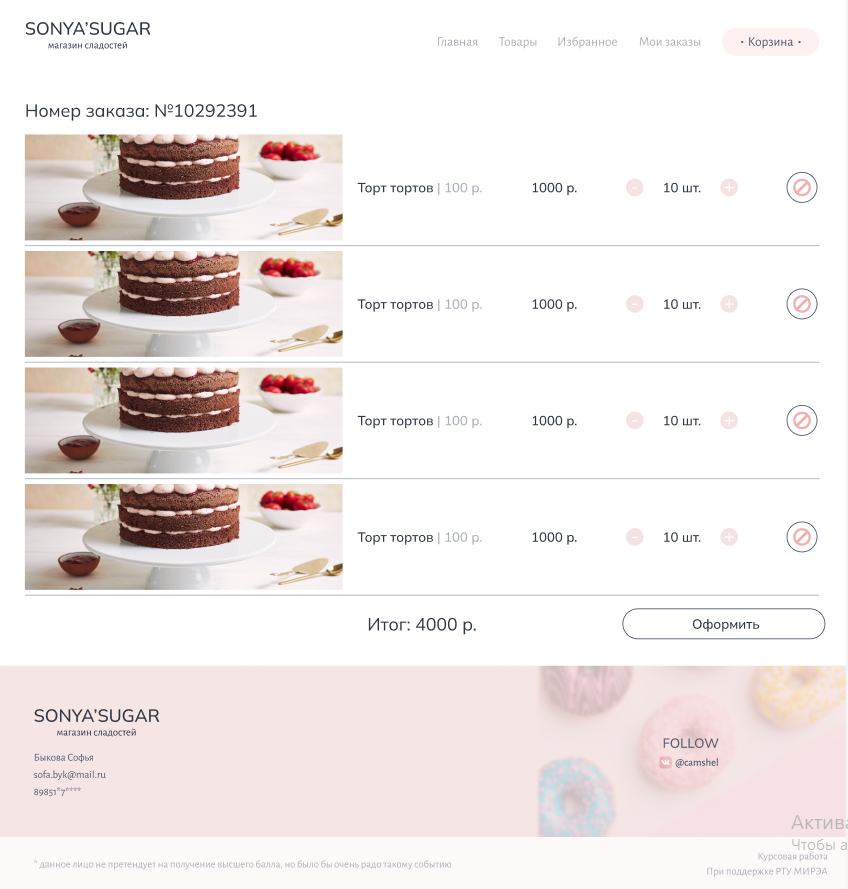


Рис. 3.6.5 – Страница корзины

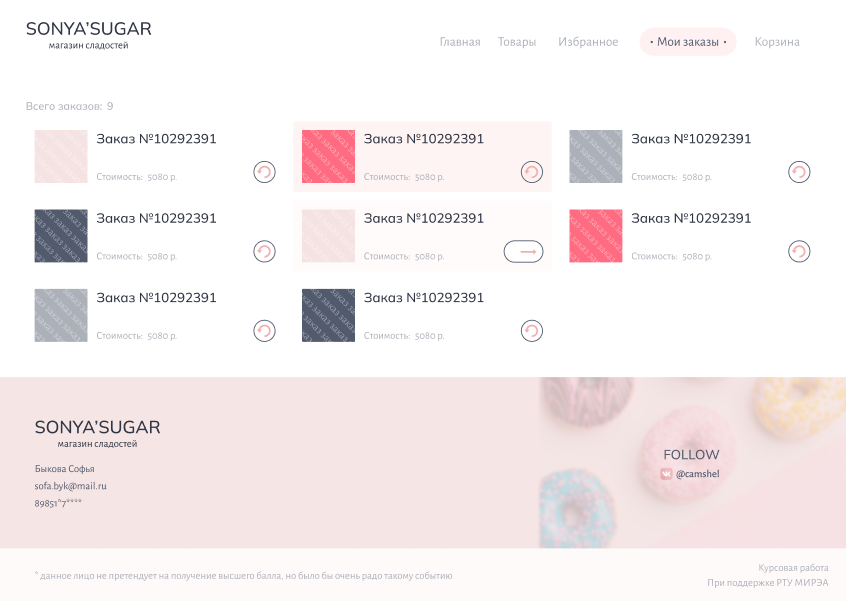


Рис. 3.6.6 – Страница заказов

Внешний вид и количество представлений в веб-приложении претерпели некоторые изменения.

С помощью Thymeleaf страницы шаблонизированы, что даёт возможность сократить количество рукописного кода.

**3.7. Подготовка презентации выполненной курсовой работы**

Данный этап не представлен в работе, так как не имеет прямого отношения к работе, а отражает самое важное из процесса создания веб-приложения.

**ЗАКЛЮЧЕНИЕ**

В ходе выполнения данной курсовой работы был разработан интернет-ресурс на тему «Магазин сладостей». Во время разработки были изучены технологии Spring Boot, Java, MVC.

Было разработано 12 шаблонных страниц, проработано их содержание, расположение элементов, логика переходов и наполнения динамическим содержимым. Была продумана система взаимодействия моделей, контроллеров и представлений друг с другом.

Был разработан понятный для использования интерфейс при помощи языка CSS3.

Была произведена настройка конфигураций работы приложения.

Были разработаны три основных слоя приложения: логика раоты приложения, взаимодействие с БД и клиентское представление.

Вся информация о моем проекте располагается на платформе GitHub в моем личном репозитории по адресу [Plintus-bit/Kursovaya-RSCHIR (github.com)](https://github.com/Plintus-bit/Kursovaya-RSCHIR).

**СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ**

1. Jetbrains Intellij IDEA [Электронный ресурс] / Developed with drive and IntelliJ IDEA – И.: JetBrains, [200?] – Электрон. дан. – 2000-2022 JetBrains s.r.o. – Режим доступа: <https://www.jetbrains.com/idea/> – Загл. с экрана;
2. Visual Studio Code [Электронный ресурс] / Code editing. Redefined – И.: Microsoft, 2015 – Электрон. дан. – 1972-2022 Microsoft – Режим доступа: <https://code.visualstudio.com/> – Загл. с экрана;
3. Figma [Электронный ресурс] / Figma: the collaborative interface design tool – Д.: 2012 – Электрон. дан. – Режим доступа: <https://www.figma.com/?fuid=> – Загл. с экрана;
4. Adobe Illustrator [Электронный ресурс] / Приложение для создания графики – Д.: 1985 – Электрон. дан. – Режим доступа: <https://www.adobe.com/ru/products/illustrator.html> – Загл. с экрана;
5. Мармелад Шоу [Электронный ресурс] / Мармелад Шоу – магазин необычных сладостей в Москве – М.: МармеладШоу, 2016-2021 ИП Амбарцумян Микаэл Геворкович – Электрон. дан. – Режим доступа: [mdshow.ru/](https://mdshow.ru/) – Загл. с экрана;
6. Конфаэль [Электронный ресурс] / Конфаэль | Фирменный интернет-магазин шоколадных подарков в Москве – М.: 2001 Группа компаний «Конфаэль» – Электрон. дан. – Режим доступа: <https://confaelshop.ru/> – Загл. с экрана;
7. Сладкая лавка [Электронный ресурс] / Магазин сладостей из Европы и Сша Москве – Sweet Lavka – М.: 2022 Интернет-магазин SWEET LAVKA – Электрон. дан. – Режим доступа: [sweet-lavka.ru](https://www.sweet-lavka.ru/) – Загл. с экрана.
8. Spring Boot [Электронный ресурс] / Spring Boot – М.: 2023 – Электрон. дан. – Режим доступа: <https://spring.io/projects/spring-boot> – Загл. с экрана;