

МИНОБРНАУКИ РОССИИ

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**

**«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «ВГУ»)**

Факультет *Компьютерных наук*

Кафедра *программирования и информационных технологий*

Продуктовый онлайн-магазин «GoodFood»

Курсовой проект

09.03.04 Программная инженерия

Преподаватель _____ В.С. Тарасов __.__.20__

Обучающийся _____ Я.В. Лазарев, 3 курс, д/о

Обучающийся _____ М.А. Мотенко, 3 курс, д/о

Обучающийся _____ Н.А. Трофимов, 3 курс, д/о

Воронеж 2021

Содержание

Содержание	2
Введение	4
1 Постановка задачи	4
1.1 Цель	4
1.2 Сфера использования	5
1.3 Требования	5
1.4 Задачи	5
2 Анализ предметной области	6
2.1 Целевая аудитория	6
2.2 Пользователи системы	7
2.3 Анализ существующих решений	7
2.4 Анализ задачи	11
2.4.1 Варианты использования приложения	11
2.4.2 Взаимодействие компонентов системы	13
2.4.3 Варианты состояния системы	16
2.4.4 Варианты действий в системе	17
2.4.5 Развертывание приложения	18
2.4.6 Диаграмма классов	19
2.4.7 Диаграмма объектов	19
2.4.8 IDEF0	20
3 Архитектура приложения	22
4 Проектная часть	25
4.1 Интерфейс приложения пользователя	25
4.1.1 Страница регистрации	25
4.1.2 Страница авторизации	25
4.1.3 Главная страница	26
4.1.4 Личный кабинет	26
4.1.5 Страница правильного питания	29
4.1.6 Товары	30
4.1.7 Корзина	31
4.1.8 Оформление заказа	31
5 Тестирование	33

5.1 Дымовое тестирование	33
Заключение	35
Список используемой литературы	36

Введение

На данный момент всё больше людей предпочитают пользоваться услугами online супермаркетов. Их главным преимуществом является возможность выбирать, оплачивать и получать товары, не выходя из дома. Именно для реализации этих целей и создаётся сайт «GoodFood».

Также большой популярностью пользуется система правильного питания, позволяющая людям худеть и следить за своим здоровьем, подсчитывая количество калорий в съеденной пищи. Поэтому, помимо экономии времени на поход в магазин, наш сервис должен помогать людям, придерживающимся правил здорового питания, контролировать количество калорий в заказе.

1 Постановка задачи

Проект «GoodFood» является продуктовым онлайн-магазином, предназначенным для пользователей, желающих приобрести продукты, не выходя из дома.

1.1 Цель

Создать сервис, который предоставит всем желающим:

- Возможность заказать продукты онлайн;
- Простую навигацию по сайту;
- Помощь в выборе продуктов с помощью рекомендательной системы;
- Возможность подсчета суточной нормы калорий каждого зарегистрированного пользователя;
- Возможность регулирования количества калорий в заказе каждого зарегистрированного пользователя;

1.2 Сфера использования

Повседневная жизнь.

1.3 Требования

Разрабатываемый сайт должен предоставлять пользователям следующие возможности:

1. Регистрация и авторизация;
2. Редактирование профиля пользователя;
3. Введения дополнительных личных данные, позволяющих сайту рассчитать суточную норму калорий;
4. Просмотр товаров и добавления в корзину с регулированием их количества;
5. Оформления заказа с оплатой на сайте;
6. Ознакомление со статьёй о правильном питании и перехода в режим регулирования калорий.

1.4 Задачи

1. Провести анализ рынка с целью выявления достоинств и недостатков схожих по функционалу систем;
2. Спроектировать систему с учетом информации, полученной ранее в ходе анализа;
3. Разработать базу данных;
4. Разработать Back-end часть приложения;
5. Разработать Front-end часть приложения;
6. Создать связь между Front-end и Back-end частями приложения;
7. Создать профиль пользователя;
8. Создать профиль администратора;
9. Описать процесс разработки и результат.

2 Анализ предметной области

2.1 Целевая аудитория

Сайт продуктового магазина нацелен на всех людей. Человек любого возраста может просматривать страницы супермаркета, т.к. ни одна из них не будет содержать информацию с возрастными ограничениями. Согласно проведенному опросу, в котором участвовало 100 человек, 30% из них предпочитают заказывать продукты онлайн, ещё 25% хотели бы иметь такую возможность, но из-за недоступности сервисов в их городах воспользоваться такими услугами невозможно. Остальные 45% людей предпочитают посещать магазины лично.

Мы решили узнать, по какой причине 45% опрошенных не заказывают продукты онлайн, смотреть Рис.1



Рисунок 1 - Основные причины отказа от онлайн супермаркетов

Также стоит заметить, что 15% опрошенных придерживаются правил здорового питания. Они отметили, что после похода в магазин им приходится вводить в свои приложения для подсчёта калорий данные о купленных продуктах, чтобы просмотреть калории в них и рассчитать, какое количество допустимо для употребления.

Таким образом, наше веб-приложение ориентировано на большинство пользователей, предпочитающих заказывать продукты, не выходя из дома.

При этом мы ориентируем наш сайт на правильное питания, предоставляя пользователям возможность при желании воспользоваться функцией подсчёта калорий прямо на сайте, не прибегая к использованию дополнительных приложений.

2.2 Пользователи системы

Для взаимодействия с разрабатываемой системой выделяют следующие типы пользователей: гость, пользователь, администратор.

Гость не предоставляет системе никаких личных данных.

Пользователь (авторизованный в системе) обладает следующими параметрами:

- Имя и фамилия пользователя;
- Электронная почта;
- Пароль.

После перехода в систему подсчёта калорий будут добавлены следующие параметры:

- Вес
- Пол
- Возраст

Администратор (авторизованный в системе работник магазина) обладает следующими параметрами:

- Имя и Фамилия;
- Электронная почта;
- Пароль;

2.3 Анализ существующих решений

Проведя анализ нескольких супермаркетов, мы выявили основные проблемы подобных сервисов. Основные недостатки:

- Нет возможности в любой момент перейти к другой категории товаров. Необходимо подниматься на верх страницы, чтобы выбрать другие продукты, смотреть Рис.2 (Сайт магазина <https://av.ru/>)

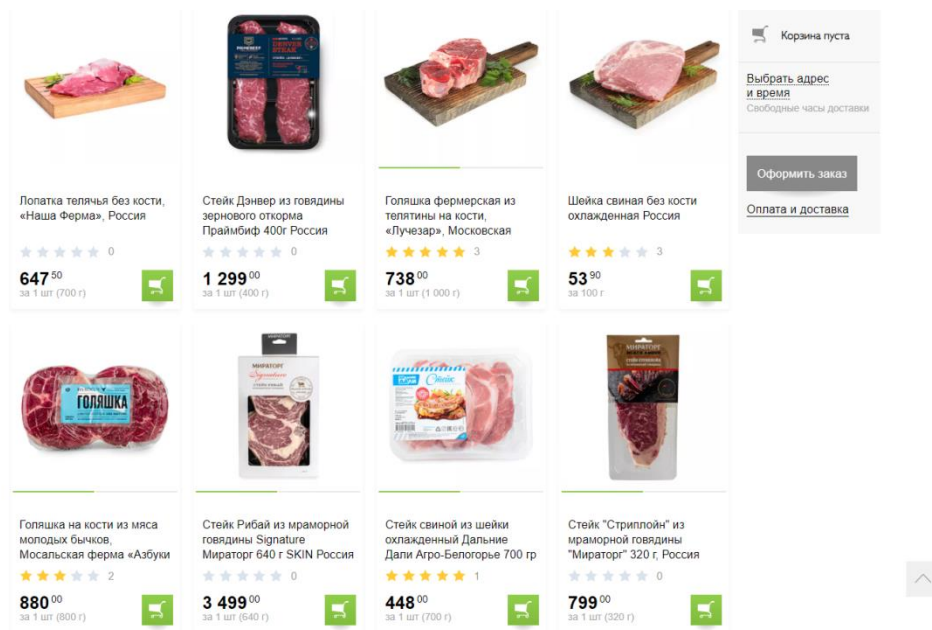


Рисунок 2 - Страница с товарами магазина «Азбука Вкуса»

- Нет возможности в фильтре задать диапазон максимальной и минимальной цены товаров, смотреть Рис.3 и Рис.4 (Сайты магазинов <https://www.auchan.ru/>, <https://vkusvill.ru/>)

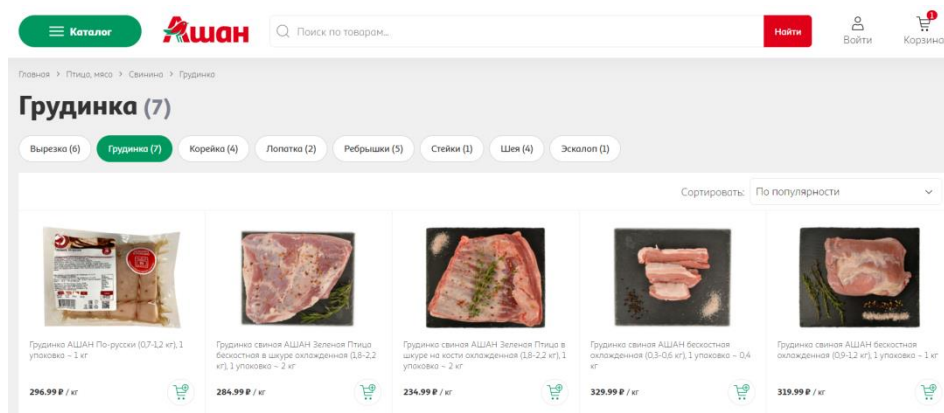


Рисунок 3 - Страница с товарами магазина «Ашан»

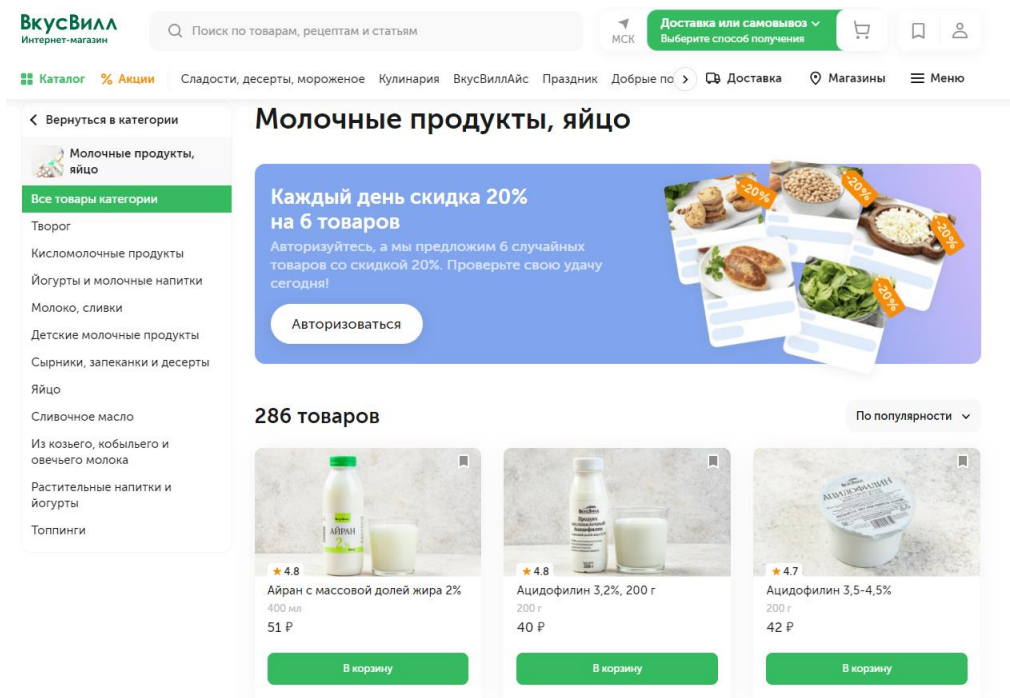


Рисунок 4 - Страница с товарами магазина «ВкусВилл»

- Есть каталоги с продуктами, информация для покупателей о магазине, но нет товаров. Невозможно определить цену продуктов по категориям. (Например, <https://www.okmarket.ru/>) После поиска в Интернете, выяснилось, что для товаров существует отдельный сайт (<https://www.okeydostavka.ru/msk/catalog>). Явных ссылок или кнопок на этот сервис на главном сайте не обнаружилось.
- Реклама и брошюры вылезают посередине всех товаров, что сильно мешает во время выбора продуктов, смотреть Рис.5(Сайт магазина <https://lenta.com/>).

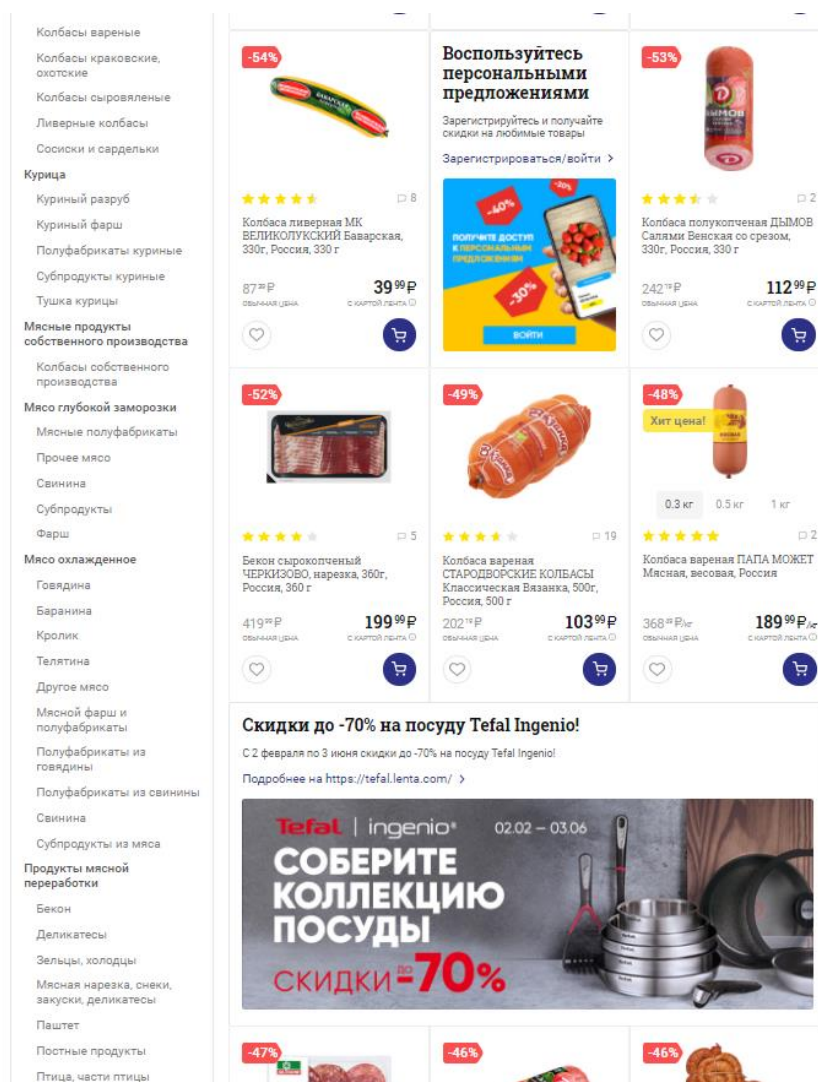


Рисунок 5 - Страница товаров магазина «Лента»

- В настоящее время мало супермаркетов, направленных на правильное питание с подсчётом калорий. Популярностью пользуются отдельные сервисы, в которые пользователи могут вводить данные о еде и узнавать количество калорий в ней. (Например, <https://supercalorizator.ru/> или <https://www.calc.ru/kalkulyator-kalorii.html>)

В результате анализа существующих решений был сформирован перечень основных недостатков аналогичных веб-приложений, а также продуманы варианты их устранения:

Недостаток	Решение
Нет возможности перейти к другим товарам, не поднимаясь на верх страницы	Добавить «Товары» в шапку сайта, которая не должна пропадать при пролистывании страницы
Слишком навязчивая реклама при просмотре товаров	Неиспользование рекламных баннеров, которые перекрывают активные элементы страницы или слишком отвлекают пользователя от основного контента
Нет возможности установить диапазон цен	Реализовать возможность регулирования максимальной и минимальной цены

Таблица 1 - Недостатки аналогов и варианты их устранения

2.4 Анализ задачи

2.4.1 Варианты использования приложения

Диаграмма прецедентов представлена на Рис. 6:

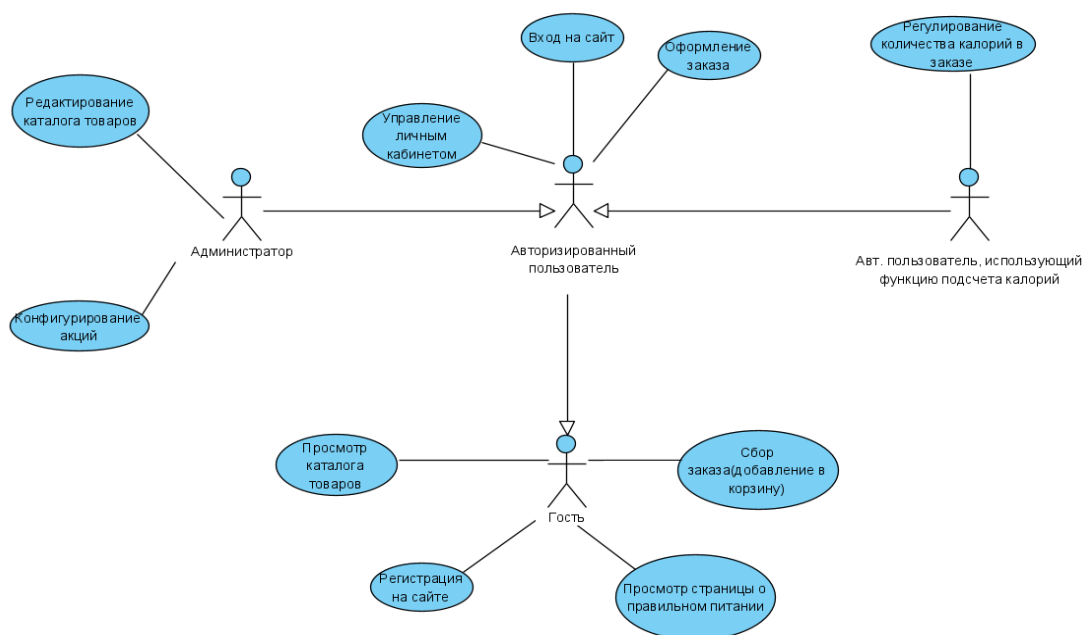


Рисунок 6 - Диаграмма прецедентов

Гость (неавторизованный в системе пользователь) имеет следующие возможности:

- Возможность просмотра всех страниц сайта, кроме Личного кабинета;

- Возможность добавления товаров в корзину;
- Возможность регистрации на сайте.

В процессе взаимодействия с системой пользователь имеет следующие возможности:

- Возможность просмотра всех страниц;
- Возможность редактирования личных данных;
- Возможность добавления товаров в корзину;
- Возможность оформления заказа;
- Возможность оплаты заказа на сайте;
- Возможность видеть количество калорий каждого продукта, суммарное количество калорий в корзине и каждого продукта в отдельности.

Пользователь, подключенный к системе подсчета калорий, обладает теми же возможностями, что и обычный пользователь, и дополнительной возможностью видеть предупреждающее сообщение в корзине, в случае превышения своей суточной нормы калорий.

В процессе взаимодействия с системой администратор имеет те же возможности, что и пользователь, и несколько дополнительных:

- Возможность добавлять, изменять и удалять товары;
- Возможность конфигурирования акций (добавление, удаление, изменение).

Отметим, что администратор, хоть и может, просматривая каталог, добавлять товары в корзину, функция оформления заказа для него отключена.

Опишем конкретно 3 сценария использования приложения.

Первый сценарий – сценарий незарегистрированного пользователя. Гость, попадая на сайт, может просматривать каталог товаров и добавлять товары в корзину. Затем он, вероятнее всего, захочет оформить заказ, однако система потребует от гостя, чтобы он зарегистрировался. Это необходимо

для того, чтобы полностью оформить заказ (внести имя клиента, адрес доставки и способ оплаты). Здесь часть пользователей может покинуть сайт из-за того, что не желает проходить регистрацию

Второй сценарий – сценарий зарегистрированного пользователя. Ему доступно оформление заказов. Некоторая группа пользователей может прекратить оформление заказов по той причине, что не желает производить оплату на сайте (магазин поддерживает только безналичную оплату).

Третий сценарий – пользователь, который включил функцию подсчета калорий. Он может при оформлении заказа увидеть сообщение о том, что его суточная норма потребления калорий может быть превышена, и перед оплатой изменить состав заказа, тем самым регулируя количество калорий в нём. Однако, конверсия по этому сценарию уменьшится, так как не всем пользователям требуется такая функция.

2.4.2 Взаимодействие компонентов системы

Рассмотрим диаграммы последовательностей и взаимодействий с системой.

Ниже представлена диаграмма(Рис.7), отражающая процесс регистрации гостя на сайте:

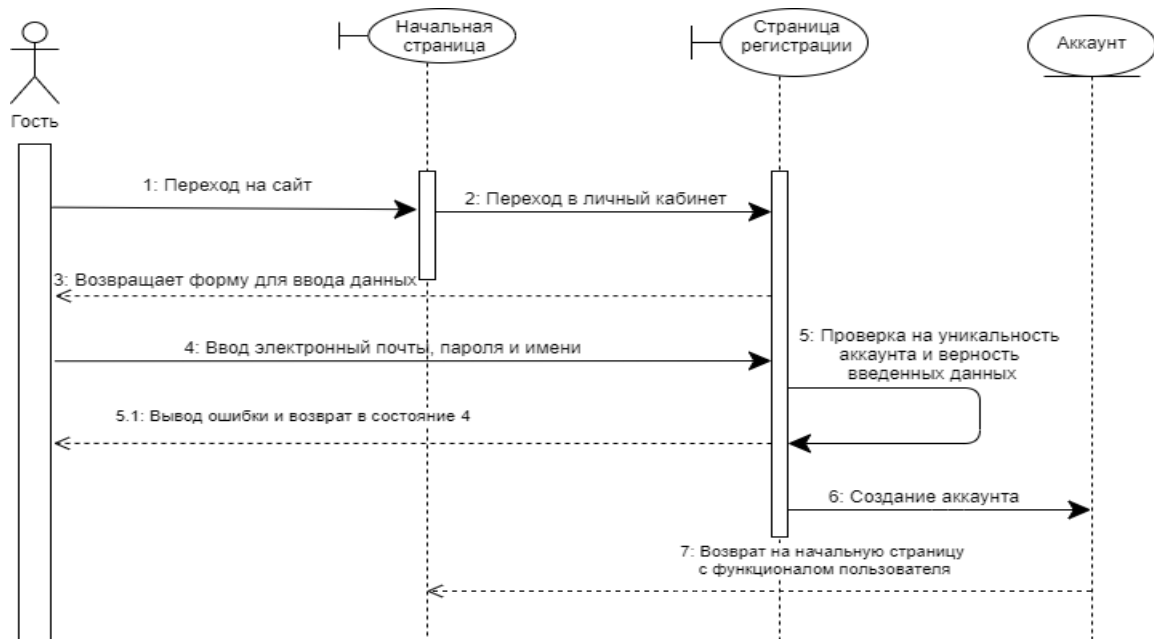
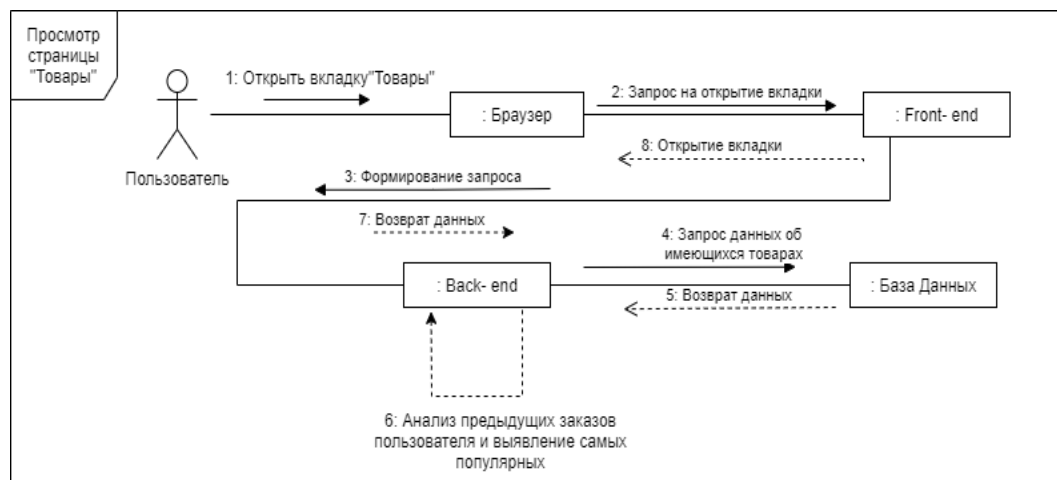


Рисунок 7 - Диаграмма взаимодействия «Регистрация гостя»

Гостю необходимо посетить главную страницу приложения, заполнить форму со своими личными данными, дождаться проверки на верность и уникальность введенных данных. После всех проведенных действий гость становится пользователем и имеет доступ ко всему функционалу, предоставленному пользователю.

Рассмотрим теперь взаимодействие пользователя «Просмотр товаров» (Рис.8)



Рассмотрим все варианты последовательностей пользователя и системы (Рис. 8) и гостя с системой (Рис.9)

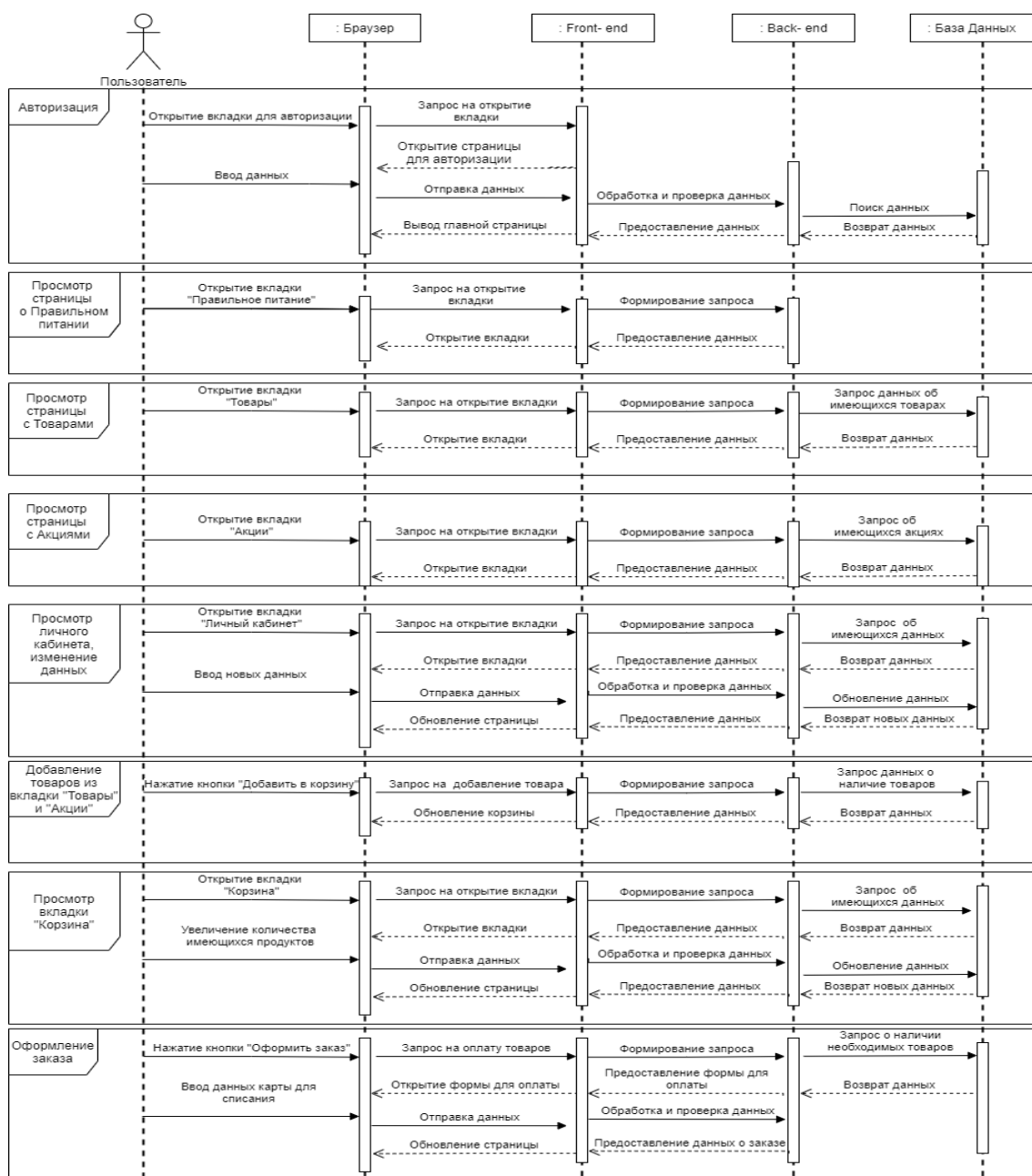


Рисунок 8 - Диаграмма последовательности для пользователя

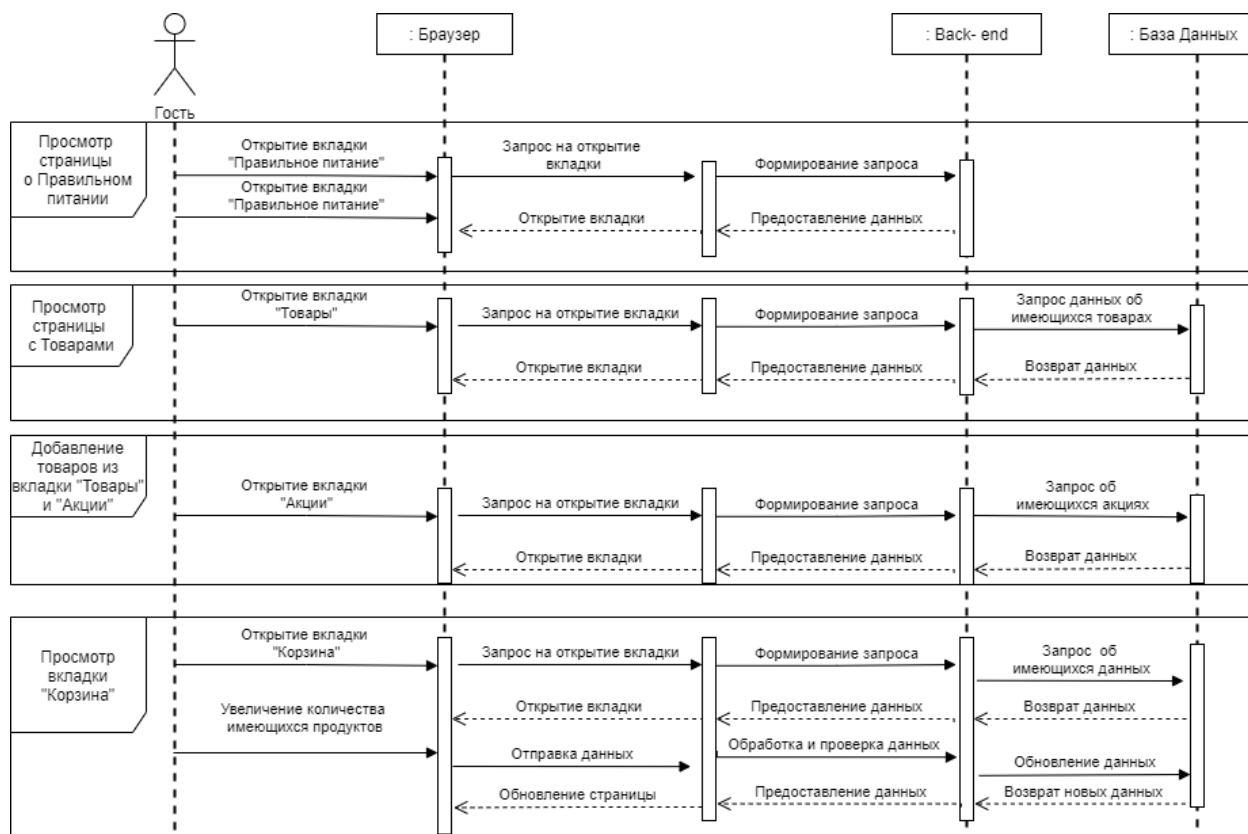


Рисунок 9 - Диаграмма последовательности для гостя

2.4.3 Варианты состояния системы

Данная диаграмма (Рис.10) отражает все состояния, которые может пройти гость и пользователь при просмотре всех страниц приложения и оформлении заказа.

Запуск системы возможен при переходе по ссылке на сайт. Всем посетителям сайта доступны возможности гостя. Если у посетителя есть аккаунт , то необходима лишь авторизация для возможности оформления заказа ; если же аккаунта нет, то необходима сначала регистрация.

После авторизации в системе пользователь может перейти на все страницы приложения, добавлять товары, формировать заказ и оплачивать его. После перехода в «Корзину» пользователь может проверить выбранные продукты и перейти к оплате.

личного кабинета, изменить личные данные и выйти из системы.

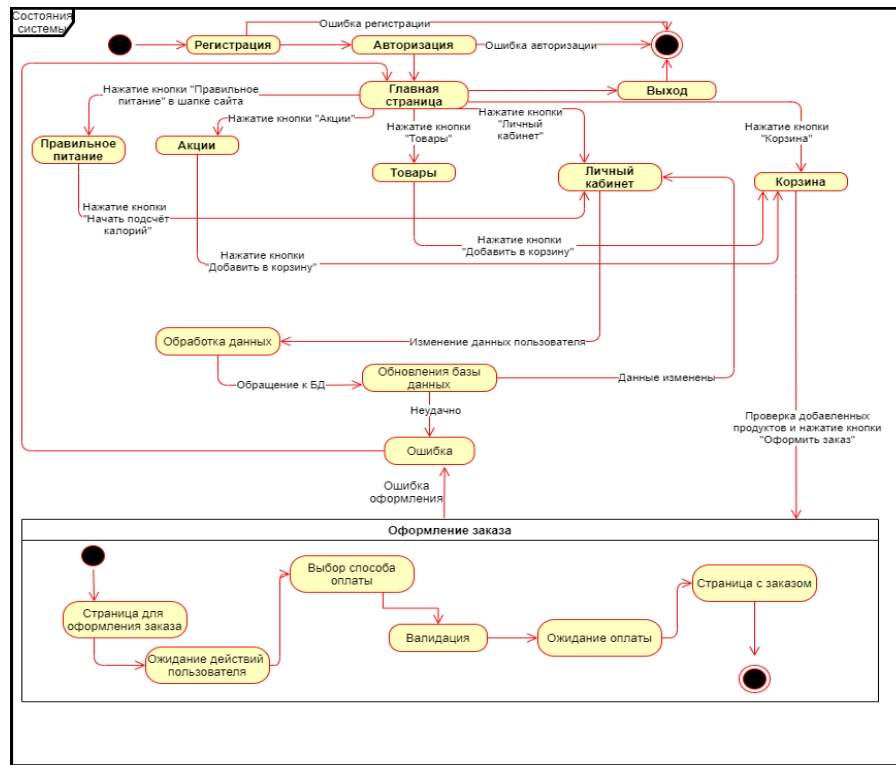


Рисунок 10 - Диаграмма состояний

2.4.4 Варианты действий в системе

Ниже представленная диаграмма (Рис.11) дает возможность увидеть все варианты действий пользователя при оформлении заказа в системе.

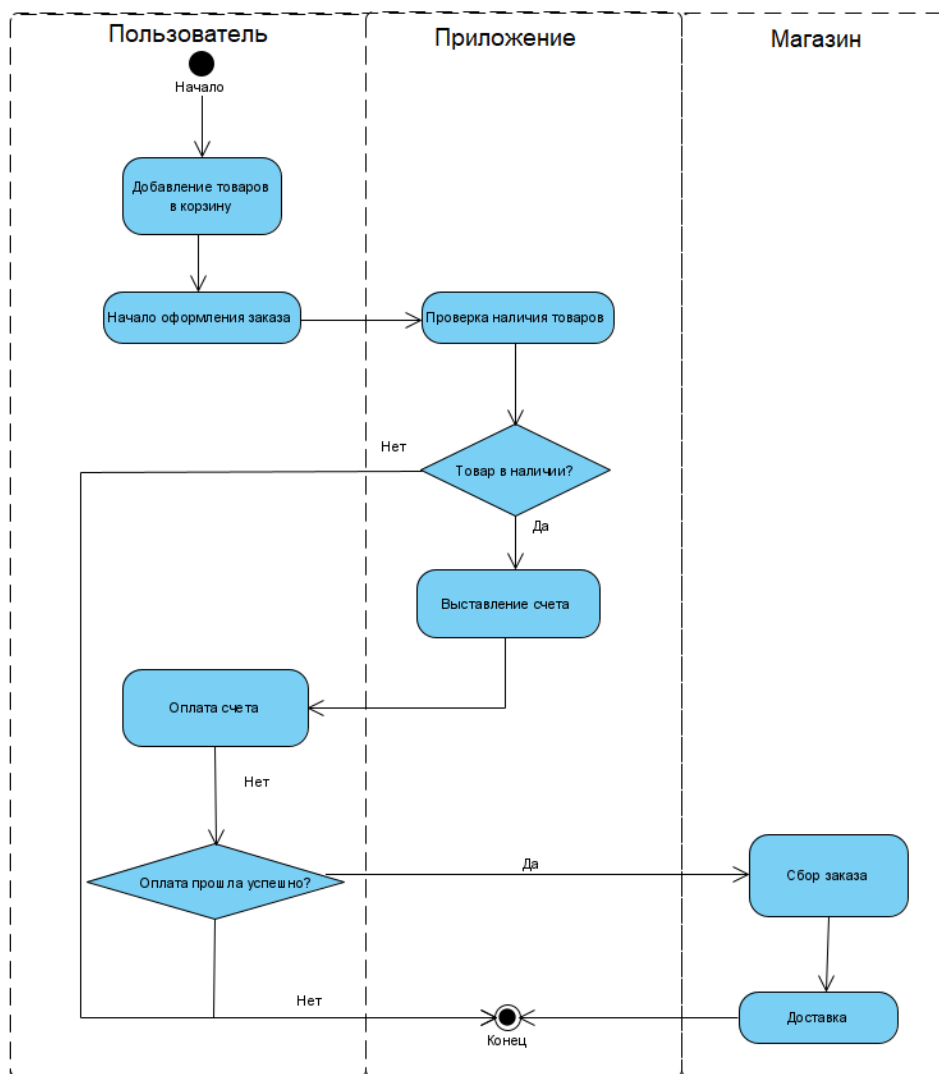


Рисунок 11 - Диаграмма активностей оформления заказа

2.4.5 Развертывание приложения

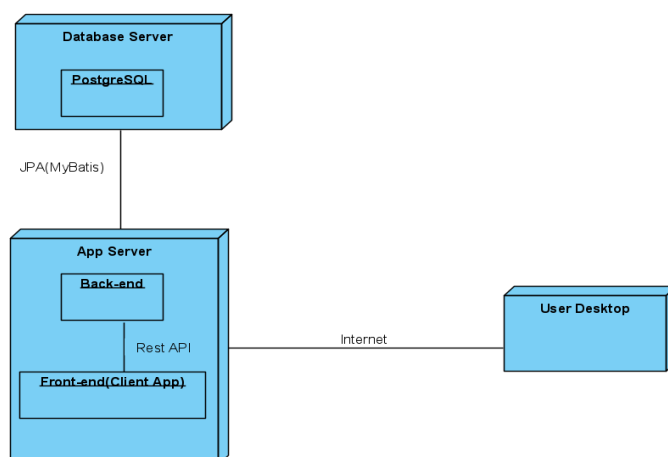


Рисунок 12 - Диаграмма развёртывания

Данная диаграмма (Рис.12) показывает архитектуру исполнения системы, включая такие узлы, как программные среды исполнения, а также промежуточное программное обеспечение, соединяющее их.

2.4.6 Диаграмма классов

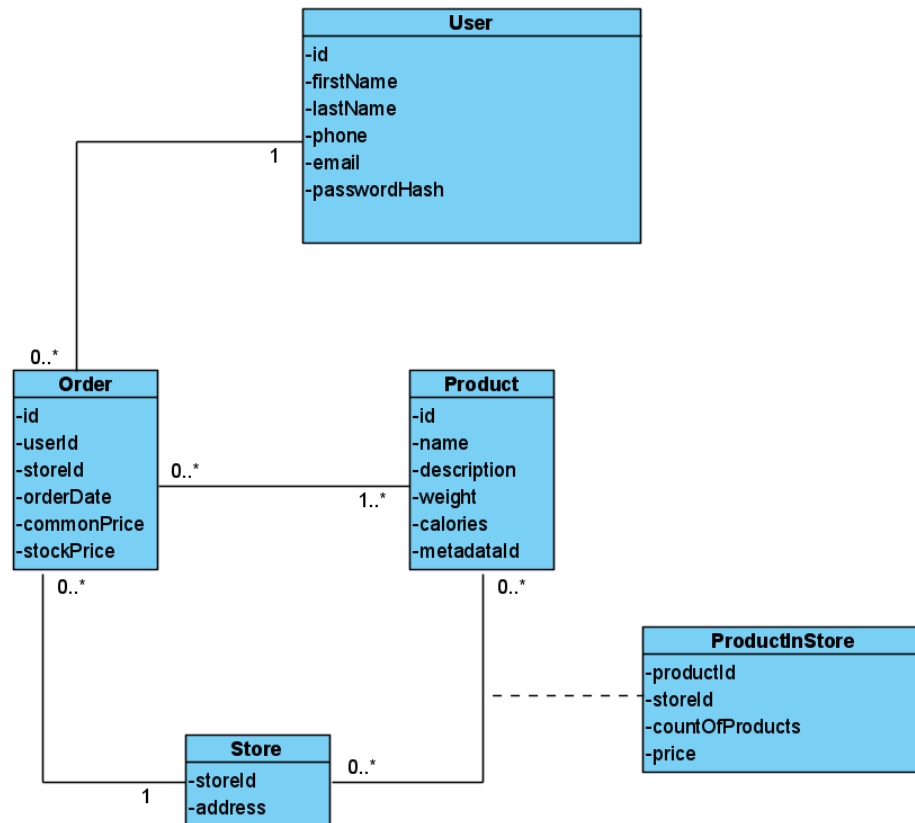


Рисунок 13 - Диаграмма классов

Диаграмма классов (Рис.13) отображает все типы классов системы и различного рода статические связи, которые существуют между ними.

2.4.7 Диаграмма объектов

Диаграмма объектов предназначена для демонстрации совокупности моделируемых объектов и связей между ними в фиксированный момент времени. Рассмотрим момент (Рис.14), когда в системе есть два пользователя, которые выбирают товары и формируют заказы. Причём товар «Яблоко» попадает в оба заказа.

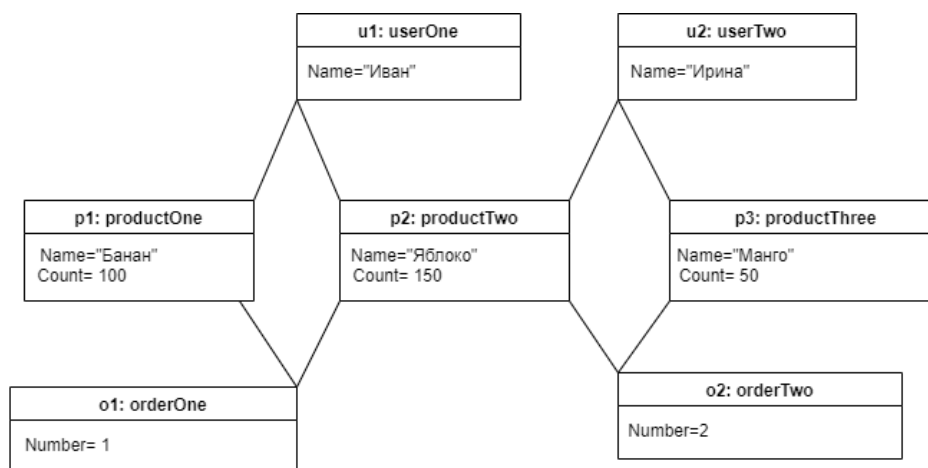
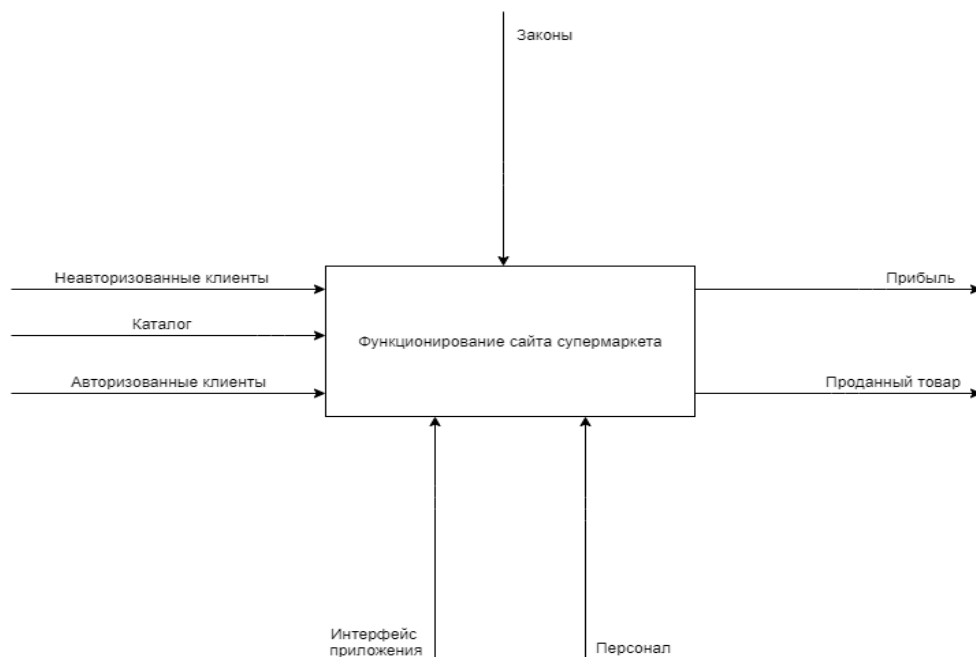


Рисунок 14 - Диаграмма объектов

2.4.8 IDEF0

Контекстная диаграмма системы функционирования сайта супермаркета (или диаграмма верхнего уровня) является вершиной древовидной структуры диаграмм и показывает назначение системы (или основную функцию) и ее взаимодействие с внешней средой. Для супермаркета «GoodFood» контекстная диаграмма представлена на Рисунке 15.



A-0

Рисунок 1 - Контекстная диаграмма

Диаграммы декомпозиции, получающиеся при разбиении контекстной диаграммы на крупные подсистемы, предназначены для детализации функций и описывают каждую подсистему и взаимодействие с другими подсистемами. Рассмотрим диаграмму декомпозиции первого уровня (Рис. 16), демонстрирующую заказ продуктов пользователем, который изначально был гостем.

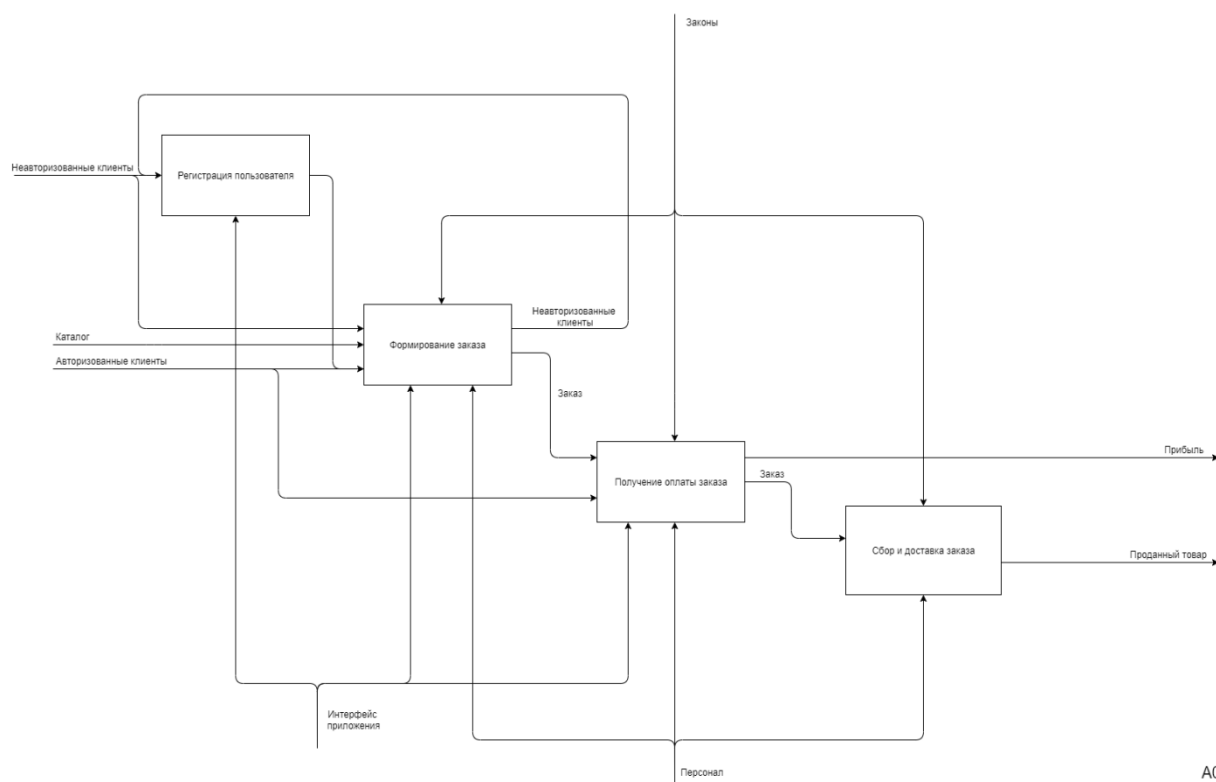


Рисунок 2 - Диаграмма декомпозиции первого уровня

3 Архитектура приложения

В качестве средств реализации приложения были выбраны следующие технологии:

Back-end:

- Java + Spring Boot Framework;

Front-end:

- Angular Framework;
- HTML 5;
- CSS;
- JavaScript;
- TypeScript;

База данных:

- PostgreSQL.

Технология Spring MVC, Spring Framework с Spring Boot и Angular - продуктивный и привлекательный стек для разработки небольших веб-приложений, в особенности таких, где требуется интенсивно работать с формами.

Фреймворк Spring MVC обеспечивает архитектуру паттерна (Модель — Отображение (далее — Вид) — Контроллер) при помощи слабо связанных готовых компонентов. Паттерн MVC разделяет логику ввода, бизнес-логику и логику UI, обеспечивая при этом свободную связь между ними. Такая технология позволяет избегать написания многих аннотаций и конфигураций xml, и обеспечивает взаимодействие с системами Spring JDBC, Spring Security и др.

Spring Framework предлагает не только такие функции, как внедрение зависимостей или обработка транзакций, но также выступает в качестве основы для других фреймворков Spring. Лучшим примером для этого является Spring Boot. Spring Boot использует Spring Framework в качестве

своей основы, упрощая зависимости Spring и запуская приложения прямо из командной строки. Также он не требует наличия внешнего контейнера приложений. Spring Boot помогает контролировать компоненты приложения и настраивает их извне. Благодаря таким функциям, как автоконфигурация, Spring Boot избавляет от написания лишнего кода и помогает избежать ненужной настройки.

AngularTS - это фреймворк, построенный на основе HTML и JavaScript, двух технологий, давно используемых в веб-разработке. Он позволяет использовать привычные редакторы и расширения для браузеров. Версия Angular CLI (Command Line Interface) стандартизирует структуру, позволяет создать сущности внутри приложения, а также автоматизировать его сборку.

На фреймворках разрабатываются довольно большие и сложные сайты с уникальным функционалом. Это значительно быстрее и дешевле, чем на чистом языке, но при этом такое решение позволяет разрабатывать действительно сложные вещи и оптимизировать все это под нагрузки. Кроме того, это почти всегда более безопасно, чем любая коробочная CMS.

Существует множество языков программирования, шаблонов и фреймворков. Но, благодаря многим функциям, упомянутым выше, Spring Framework и AngularTS является отличным выбором для нашего проекта.

В качестве СУБД была выбрана PostgreSQL в силу открытого доступа и высокой производительности, а также за счет поддержки данной СУБД schema-less данных, такие как JSON. PostgreSQL поддерживает интеграцию на различные платформы, взаимодействие с большинством языков программирования. PostgreSQL имеет преимущество перед другими DBMS, когда необходимо перенести базу данных из одной ОС в другую или в случае, когда реализация приложения производится на различных ОС.

Специфика клиентской стороны веб-приложения требует отладки кода во всех популярных браузерах. Наше приложение разрабатывалось под

браузер Google Chrome. Этот браузер, основан на WebKit, обладает встроенным инструментом разработки Web Inspector, который очень хорошо развит и позволяет выполнять отладку JavaScript кода.

Общение между Front-end и Back-end происходит по средством REST API, а обмен информации происходит с помощью передачи данных в формате JSON.

4 Проектная часть

4.1 Интерфейс приложения пользователя

4.1.1 Страница регистрации

Страница для регистрации новых пользователей представлена на Рисунке 17. Для регистрации гостю необходимо ввести свои данные в следующие поля:

- Имя;
- Фамилия;
- Телефон;
- Электронная почта;
- Пароль.

GoodFood Главная Правильное питание Товары Акции Воронеж

Регистрация

Имя
Имя

Фамилия
Фамилия

Телефон
+7 (000) 000-00-00

Эл. почта
email@example.com

Password
Password

Repeat password
Password

Sign up

Уже есть аккаунт? [Вход](#)

Рисунок 17 - Страница регистрации

4.1.2 Страница авторизации

На Рисунке 18 представлена страница авторизации.

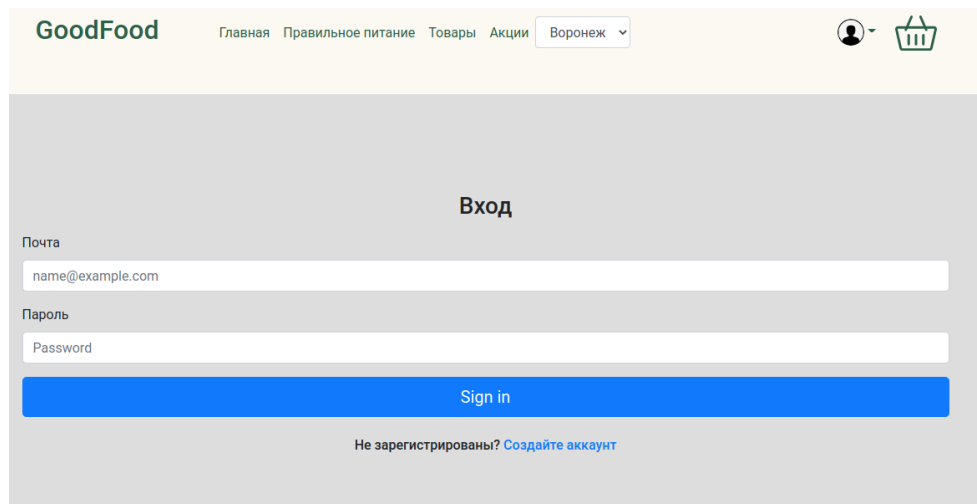


Рисунок 18 - Страница авторизации

4.1.3 Главная страница

Главная страница приложения представлена на Рисунке 19.

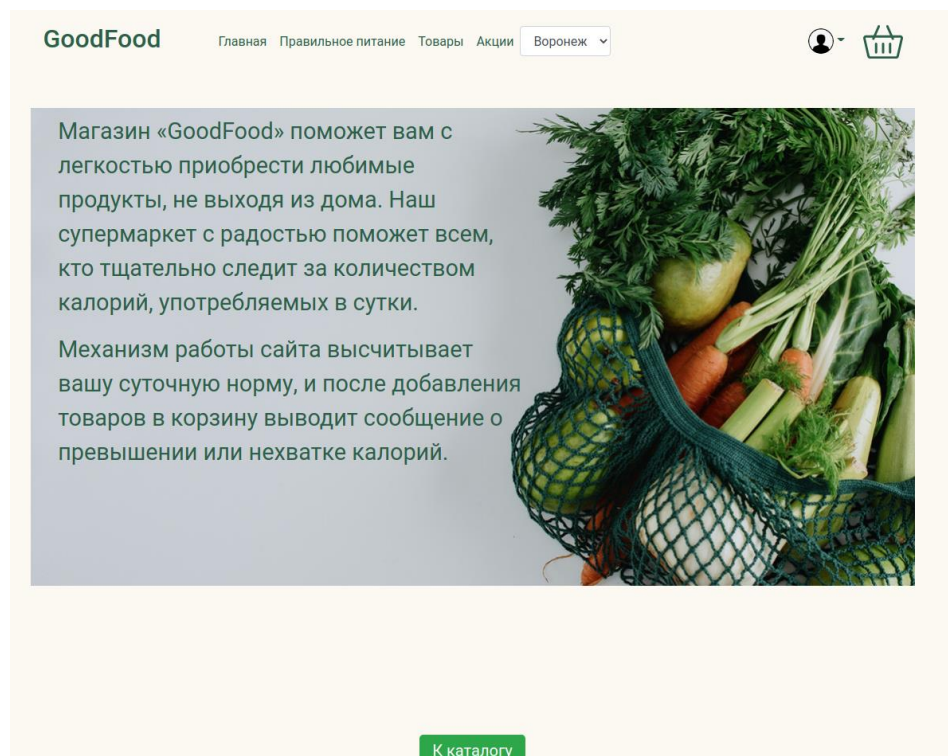


Рисунок 19 - Главная страница

4.1.4 Личный кабинет

Личный кабинет состоит из 5 основных разделов:

- Мои заказы (Рисунок 20);

- Адреса для доставки (Рисунок 21) с возможностью добавления новых адресов (Рисунок 22);
- Мои карты (Рисунок 23);
- Мои калории (Рисунок 24);
- Мои данные (Рисунок 25).

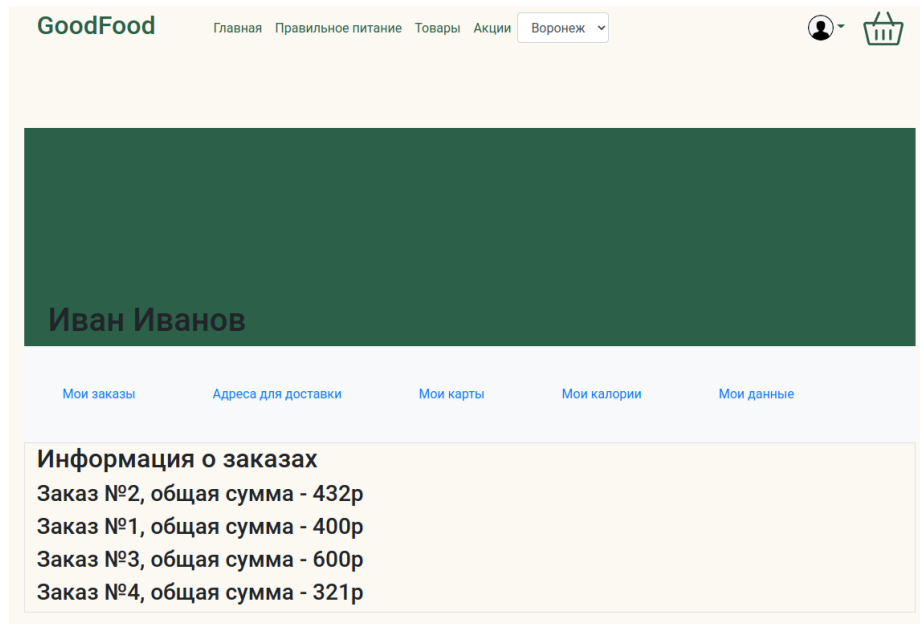


Рисунок 20 - Вкладка «Мои заказы»

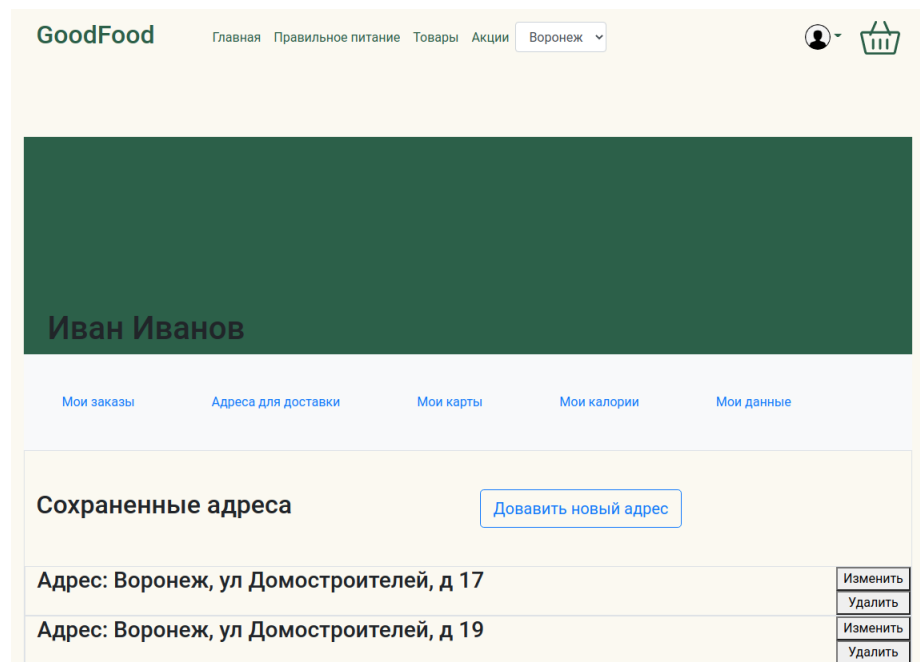


Рисунок 21 - Вкладка «Адреса для доставки»

Добавить новый адрес

✕

Адрес

Добавить/Изменить

Рисунок 22 - Окошко добавления адреса

GoodFood

[Главная](#) [Правильное питание](#) [Товары](#) [Акции](#)

Воронеж ▾

👤

🛒

Иван Иванов

[Мои заказы](#) [Адреса для доставки](#) [Мои карты](#) [Мои калории](#) [Мои данные](#)

Сохраненные карты

Добавить

**** * 4444

09/21

Изменить

Удалить

Рисунок 23 - Вкладка «Мои карты»

Рисунок 24 - Вкладка «Мои калории»

Рисунок 25 - Вкладка «Мои данные»

4.1.5 Страница правильного питания

Страница правильного питания (Рисунок 26) содержит статью об основах здорового питания и кнопку для перехода в раздел «Мои калории» (Рисунок 24).

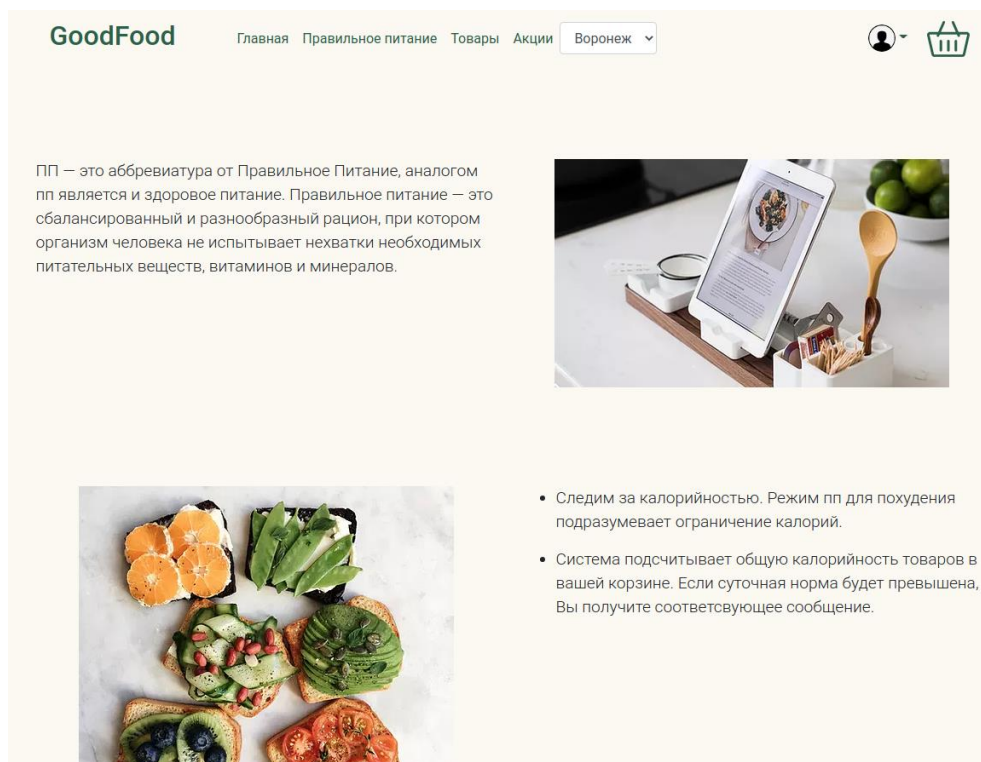


Рисунок 26 - Страница Правильного питания

4.1.6 Товары

Страница «Товары» (Рисунок 27) отражает товары, их цены, а также содержит фильтр по категориям и ценам.

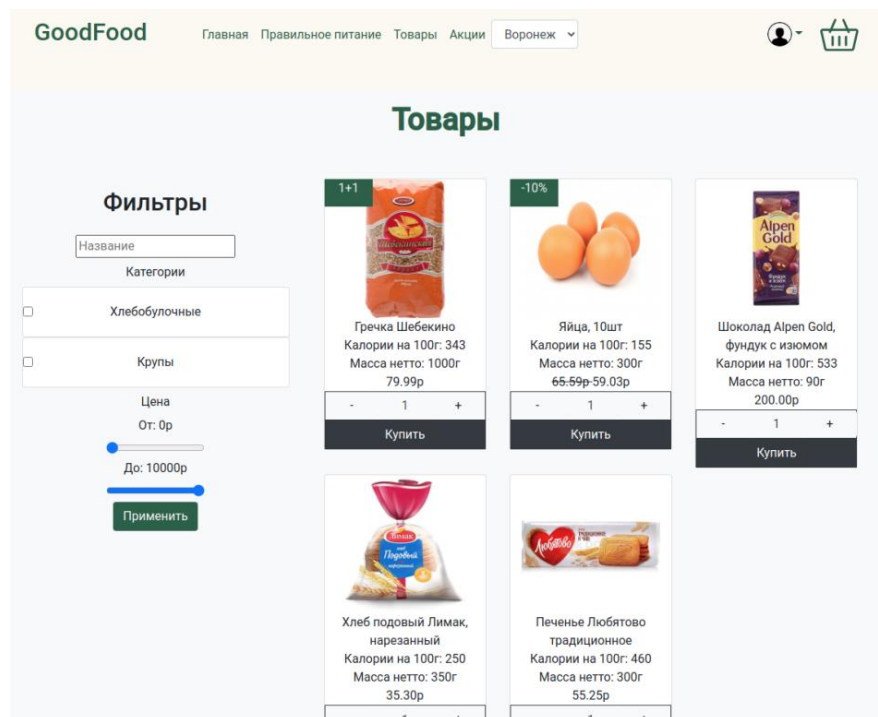


Рисунок 27 - Страница «Товары»

4.1.7 Корзина

После нажатия кнопки «Купить» под продуктом на странице с товарами или акциями у пользователя в корзине появляется данный продукт или его количество увеличивается. Все добавленные товары хранятся в «Корзине» (Рисунок 29) , в которую можно попасть из шапки сайта.

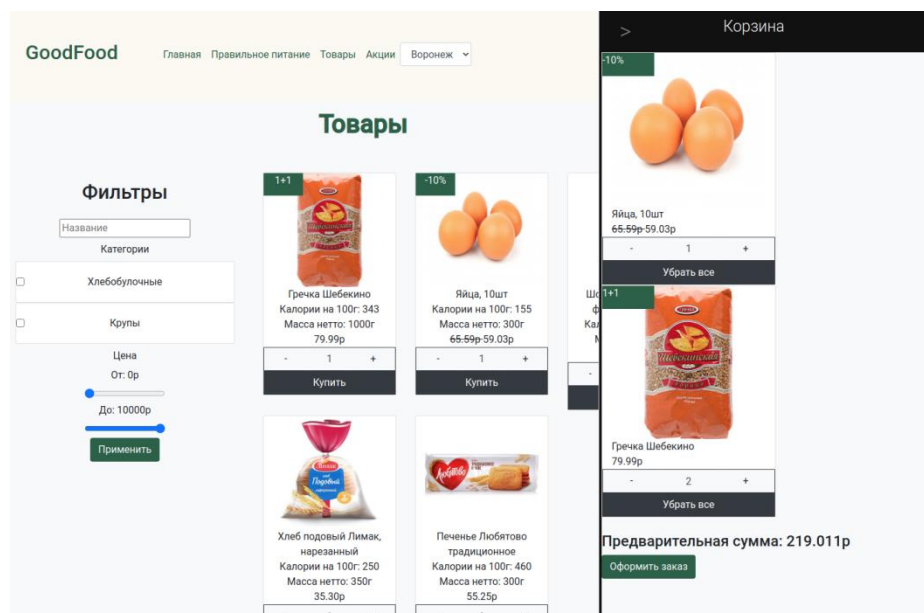


Рисунок 28 - Открытие корзины

4.1.8 Оформление заказа

После добавления необходимых товаров в корзину пользователь может перейти к оформлению заказа (Рисунок 30), нажав в корзине на кнопку «Оформить заказ», и его оплате (Рисунок 31).

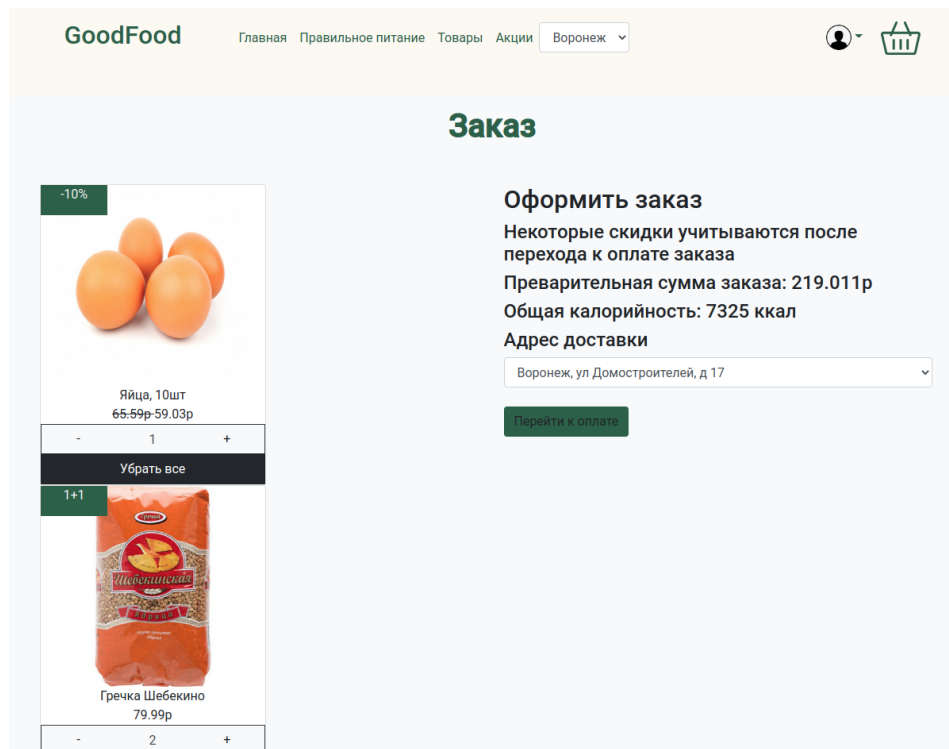


Рисунок 29 - Оформление заказа

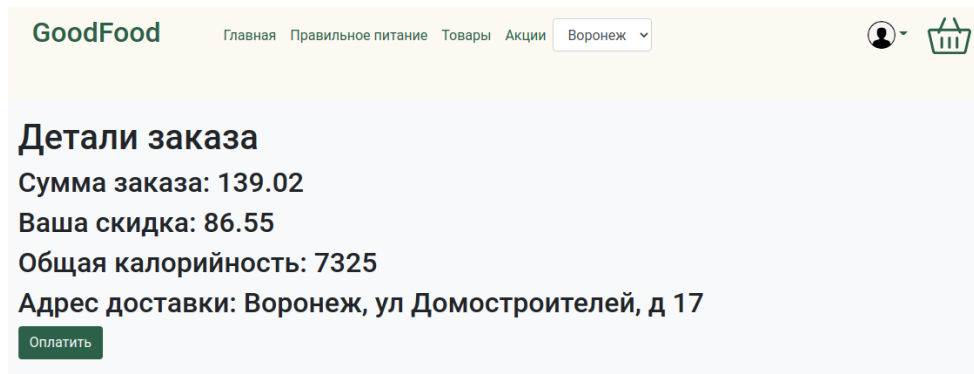


Рисунок 30 - Оплата заказа

5 Тестирование

5.1 Дымовое тестирование

В процессе тестирования была проверена работа приложения по следующим сценариям:

1. Регистрация пользователя;
2. Авторизация пользователя;
3. Изменение личных данных пользователя;
4. Добавление дополнительных данных для подсчёта калорий;
5. Добавление нового адреса доставки;
6. Переход на страницу с товарами;
7. Использование фильтра по цене;
8. Использование фильтра по категории;
9. Добавление товара в корзину;
10. Переход на страницу правильного питания;
11. Переход в корзину;
12. Оформление заказа;
13. Переход на главную страницу.

Дымовое тестирование проводилось ручным способом, на развернутом сервере и открытом в браузере Google Chrome версии 91.0.4472.106.

Результаты, полученные в ходе тестирования, представлены в таблице 2.

Таблица 2. – Результаты дымового тестирования

Сценарий	Результат
Регистрация пользователя	Пройден
Авторизация пользователя	Пройден
Изменение личных данных пользователя	Пройден
Добавление дополнительных данных для подсчёта калорий	Пройден
Добавление нового адреса доставки	Пройден
Переход на страницу с товарами	Пройден
Использование фильтра по цене	Пройден
Использование фильтра по категории	Пройден
Добавление товара в корзину	Пройден
Переход на страницу правильного питания	Пройден
Переход в корзину	Пройден
Оформление заказа	Пройден
Переход на главную страницу	Пройден

По итогам дымового тестирования было установлено, что приложение проходит все основные сценарии.

Заключение

В результате было разработано веб-приложения для заказа продуктов онлайн и возможностью контроля калорий в заказе.

В процессе работы выполнены следующие задачи:

1. Проведен анализ рынка с целью выявления недостатков подобных сервисов;
2. Спроектирована система с учетом информации, полученной в ходе анализа;
3. Разработана база данных;
4. Разработана Back-end часть приложения;
5. Разработана Front-end часть приложения;
6. Создана связь между Front-end и Back-end;
7. Создан профиль пользователя;
8. Создан профиль администратора;
9. Описан процесс разработки и результат.

Приложение «GoodFood» отвечает всем заявленным требованиям.

Список используемой литературы

1. Сравнение баз данных: сайт. – URL: <https://proglib.io/p/databases-2019> (дата обращения: 01.03.2021). – Текст: электронный.
2. Введение в Angular: сайт. – URL: <https://metanit.com/web/angular2/1.1.php> (дата обращения: 10.03.2021). – Текст: электронный.
3. Статья о правильном питании с подсчётом калорий : сайт. – URL: <https://growfood.pro/blog/bzhu/pitanie-s-podschetom-kalorij/> (дата обращения: 28.02.2021). – Текст: электронный.
4. Диаграммы UML: сайт. – URL: <https://evergreens.com.ua/ru/articles/uml-diagrams> (дата обращения: 20.04.2021). – Текст: электронный.
5. Spring Boot Reference Documentation : сайт. – URL: <https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/> (дата обращения: 25.04.2021). – Текст: электронный.