

**ОТЧЁТ**  
**ПО УЧЕБНОЙ ПРАКТИКЕ ПО РАЗРАБОТКЕ И**  
**СОПРОВОЖДЕНИЮ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

Учащегося \_\_\_\_\_ 3 \_\_\_\_\_ курса, группы \_\_\_\_\_ ПЗТ-41  
специальности 2 - 40 01 01 «Программное обеспечение информационных  
технологий» \_\_\_\_\_

Место прохождения практики ООО «Эдельсофт» \_\_\_\_\_

Тема проекта: «Разработка интернет ресурса для оптовой продажи зерновых  
культур» \_\_\_\_\_

Ссылка на проект: <https://github.com/midland73/Prima> \_\_\_\_\_

Данные для входа: email: johnmergich@gmail.com, пароль: 106071220065648m! \_\_\_\_\_

Учётная запись пользователя имя БД: prima, пароль: 106071220065648m! \_\_\_\_\_

Ссылка на UX-прототипы: [https://www.figma.com/design/USTDPyvhsOuJdDGpWN6Blr/UX\(Prima\)](https://www.figma.com/design/USTDPyvhsOuJdDGpWN6Blr/UX(Prima)) \_\_\_\_\_

Ссылка на UI-прототипы: [https://www.figma.com/design/HrUx1cKoTPFJ0XZv5fs4oO/UI\(Prima\)](https://www.figma.com/design/HrUx1cKoTPFJ0XZv5fs4oO/UI(Prima)) \_\_\_\_\_

Выполнил \_\_\_\_\_ А.С. Ганисевский  
(инициалы, фамилия)

Руководитель практики \_\_\_\_\_ Л.Н. Радкевич  
от предприятия М.П. (инициалы, фамилия)

Руководитель практики \_\_\_\_\_ О.И. Кизер  
от колледжа (инициалы, фамилия)

## Содержание

Введение.....	4
1 Анализ задачи .....	6
1.1 Постановка задачи .....	6
1.1.1 Организационно-экономическая сущность задачи .....	6
1.1.2 Функциональные требования.....	7
1.1.3 Описание исходной (входной) информации.....	7
1.1.4 Описание результатной (выходной) информации.....	7
1.1.5 Описание используемой условно-постоянной информации .....	8
1.1.6 Нефункциональные (эксплуатационные) требования .....	8
1.2 Диаграмма вариантов использования.....	9
1.3 Выбор стратегии разработки и модели жизненного цикла.....	10
1.4 Инструменты разработки.....	13
1.5 Разработка плана работы над проектом .....	14
2 Проектирование задачи .....	16
2.1 Разработка структуры сайта, системы меню, навигации.....	16
2.2 Разработка UML-диаграмм .....	16
3 Реализация .....	18
3.1 Создание сайта.....	18
3.2 Настройка функциональности и дизайна .....	20
3.3 Редактирование страниц.....	20
3.4 Настройка header и footer .....	20
3.5 Редактирование главной страницы сайта .....	21
4 Тестирование .....	22
4.1 Тест-кейсы.....	22
4.2 Результаты тестирования.....	25
5 Руководство пользователя .....	26

					УП ТРПО 2-40 01 01.33.41.04.24 ПЗ			
Изм.	Лист	№ докум.	Подпись	Дата				
Разраб.		Ганисевский			Разработка интернет-ресурса для оптовой продажи зерновых культур	Лит.	Лист	Листов
Пров.		Кизер					2	51
Н. контр.								
Утв.								

5.1 Общие сведения о программном продукте .....	26
5.2 Запуск программы .....	27
5.3 Инструкция по работе с программой.....	28
5.4 Завершение работы с программой .....	30
5.5 Использование системы справочной информации .....	30
Заключение .....	32
Список использованных источников .....	34
Приложение А Диаграмма вариантов использования .....	35
Приложение Б Диаграмма Ганта .....	37
Приложение В Структура сайта.....	39
Приложение Г Модель данных .....	41
Приложение Д Диаграмма классов.....	43
Приложение Е Диаграмма последовательности .....	45
Приложение Ж UX, UI-макеты .....	47

## Введение

Тема проекта – разработка интернет-ресурса для оптовой продажи зерновых культур. Основной задачей является создание простой и удобной платформы, где поставщики и покупатели смогут взаимодействовать для заключения сделок. Ресурс позволит участникам рынка быстро находить необходимую информацию о товарах и контрагентах.

Проект направлен на решение базовых задач оптовой торговли зерновыми культурами, таких как предоставление актуальной информации о продукции, упрощение поиска партнеров и минимизация времени на оформление сделок. Интернет-ресурс должен стать инструментом, который сделает взаимодействие участников процесса более удобным и понятным.

Тема актуальна в рамках курсового проекта, так как разработка такого ресурса позволяет применить на практике знания по веб-программированию, проектированию интерфейсов и работе с базами данных.

Описание разделов пояснительной записки:

Первый подраздел, озаглавленный «Постановка задачи» состоит из определении требований к системе, формировании цели и задач проекта. Уточнении функциональных и нефункциональных требований. Диаграмма вариантов использования – графическое представление взаимодействия пользователей с системой через диаграммы вариантов использования. В выбор стратегии разработки и модели жизненного цикла входит обоснование выбора подхода к разработке. Описание технологий, языков программирования, платформ и сред разработки, которые будут использоваться расположено в подразделе «Инструменты разработки». Разработка плана работы над проектом – создание плана-графика выполнения задач с указанием этапов разработки, сроков и ответственных лиц.

Разработка структуры сайта, системы меню, навигации – планирование информационной архитектуры сайта: структура страниц, логика меню, навигационные цепочки. Разработка UML-диаграмм – подготовка диаграмм UML: диаграммы классов, последовательностей, взаимодействия и активности.

Руководство программиста – разработка инструкций для программистов, включающих описание установки, настройки и модификации кода. Подраздел «Диаграмма компонентов» включает создание диаграммы компонентов для отображения архитектуры системы и взаимодействия между её частями.

Тестирование – проведение тестирования системы, включая описание методологий, этапов и устранение выявленных ошибок с документированием результатов.

Руководство пользователя – разработка инструкций для конечных пользователей, охватывающих регистрацию, оформление заказа и использование личного кабинета.

Заключение – обобщение проделанной работы, достижение поставленных целей и оценка перспектив развития проекта.

					УП ТРПО 2-40 01 01.33.41.04.24 ПЗ	Лист
						5
Изм	Лист	№ докум.	Подпись	Дата		

## **1. Анализ задачи**

### **1.1 Постановка задачи**

#### **1.1.1 Организационно-экономическая сущность задачи**

Основная задача проекта заключается в разработке интернет-ресурса для оптовой продажи зерновых культур.

Создание удобного и функционального программного продукта для оптовой торговли зерновыми культурами направлено на повышение эффективности бизнеса в аграрной сфере. Такой продукт должен обеспечивать интуитивно понятный интерфейс, который упростит процесс поиска и заказа продукции, предоставит быстрый доступ к актуальной информации о зерновых культурах, ценах, наличии на складе и характеристиках товаров. Программный продукт предназначен для следующих объектов и пользователей:

- Поставщики зерновых культур (фермерские хозяйства, аграрные компании), которым необходима платформа для демонстрации своих товаров и поиска покупателей;
- Покупатели (предприятия, закупающие зерно оптом, посредники), которым требуется простой доступ к информации о продукции и возможность быстрой связи с продавцами;
- Администраторы ресурса, ответственные за поддержку работоспособности системы и управление пользовательскими аккаунтами.

Программный продукт предполагает регулярное использование и должен быть рассчитан на ежедневную востребованность. Пользователи, включая поставщиков и покупателей, смогут размещать и искать предложения в режиме реального времени, что обеспечит оперативность и гибкость взаимодействий. Источники и способы получения данных:

- Данные о продукции предоставляются поставщиками через их личные кабинеты;
- Информация о покупателях и поставщиках (контактные данные, история сделок) поступает в процессе регистрации и взаимодействия на платформе.

Также платформа может взаимодействовать с платежными системами для упрощения расчетов между покупателями и поставщиками.

Существует несколько платформ, которые предоставляют услуги для торговли сельскохозяйственными товарами, включая зерновые культуры. Примером таких решений являются:

- GrainTrade – специализированная платформа для торговли зерном, где поставщики и покупатели могут находить друг друга и заключать сделки;
- АгроТорги – онлайн-платформа для торговли сельскохозяйственной продукцией, которая предоставляет информацию о предложениях от различных поставщиков.

					УП ТРПО 2-40 01 01.33.41.04.24 ПЗ	Лист
Изм	Лист	Недокум.	Подпись	Дата		6

### 1.1.2 Функциональные требования

Разрабатываемый программный продукт для оптовой продажи зерновых культур должен включать следующие основные функции и задачи:

- Регистрация и управление аккаунтами пользователей: возможность регистрации поставщиков и покупателей на платформе. Разграничение доступа для различных типов пользователей (поставщик, покупатель, администратор);
- Создание и управление каталогом продукции – поставщики должны иметь возможность добавлять и редактировать информацию о зерновых культурах: наименование, описание, фотографии, цена за единицу, количество в наличии;
- Оформление и управление заказами – возможность оформления заказов покупателями с указанием количества, условий доставки и оплаты.

### 1.1.3 Описание исходной (входной) информации

Перечень исходной(входной) информации:

- Данные о продукции (зерновые культуры): наименование, описание, цена, количество, регион, фотографии, сертификаты качества.

Документ в виде карточки товара с полями для ввода данных. Пример заполнения:

- Наименование: Пшеница озимая;
- Описание: Пшеница 3 класса, урожай 2024 года;
- Цена: 15 000 руб./тонна;
- Количество: 100 тонн;
- Регион: Гомельская область.

Перечень пользователей исходной информации:

- Поставщики – для загрузки информации о продукции;
- Покупатели – для оформления заказов и выбора товаров;
- Администраторы – для управления данными пользователей и контроля над процессами.

### 1.1.4 Описание результатной (выходной) информации

Перечень результатной(выходной) информации:

- Отчеты о заказах: информация о всех заказах, оформленных на платформе: количество, сумма, статус;
- Форма представления: представлен отчетом в виде таблицы или сводки. Пример:
  - Заказ: Пшеница озимая, 50 тонн;

					УП ТРПО 2-40 01 01.33.41.04.24 ПЗ	Лист
						7
Изм	Лист	№докум.	Подпись	Дата		

- Сумма: 750 000 руб.;
- Статус: Оплачено.

Периодичность и сроки представления – сроки представления зависят от типа отчета: для ежедневных отчетов сроки составляют несколько минут, для ежемесячных – отчет будет готов в течение первого рабочего дня следующего месяца.

Перечень пользователей результатной информации:

- Администраторы – для анализа работы платформы, проверки статистики и выполнения административных задач.
- Поставщики и покупатели – для получения отчетности по своим сделкам и заказам.

### 1.1.5 Описание используемой условно-постоянной информации

Перечень условно-постоянной информации:

- Классификатор зерновых культур – включает список всех типов зерновых культур, представленных на платформе (пшеница, ячмень, кукуруза, овес и т.д.). Форма представления: таблица с полями для ввода наименования культуры, ее характеристики, описание, категория.
- Справочник статусов заказов – перечень всех возможных статусов заказов, таких как «Ожидает подтверждения», «Подтвержден», «Отправлен», «Доставлен», «Закрыт». Форма представления: таблица с наименованиями статусов и их описанием.
- Классификатор единиц измерения – включает перечень всех единиц измерения, используемых для описания количества зерновых культур (тонны, килограммы, мешки и т.д.). Форма представления: таблица с наименованиями единиц измерения и их кодами.
- Справочник статусов заказов – перечень всех возможных статусов заказов, таких как «Ожидает подтверждения», «Подтвержден», «Отправлен», «Доставлен», «Закрыт». Форма представления: таблица с наименованиями статусов и их описанием.

### 1.1.6 Нефункциональные (эксплуатационные) требования

- Требования к применению

Качество пользовательского интерфейса. Интерфейс должен быть интуитивно понятным и удобным для пользователей всех уровней, включая поставщиков и покупателей. Дизайн должен быть адаптивным и работать на различных устройствах. Все элементы интерфейса (кнопки, формы, ссылки)

					УП ТРПО 2-40 01 01.33.41.04.24 ПЗ	Лист
Изм	Лист	Недокум.	Подпись	Дата		8



должны быть четко обозначены и иметь соответствующие подсказки или всплывающие окна для пояснений. Платформа должна поддерживать локализацию на несколько языков для расширения охвата;

– Требования к производительности

Эффективность использования ресурсов. Программный продукт должен быть оптимизирован для работы с большими объемами данных (базы данных с информацией о сотнях пользователей и тысячах товаров). Использование серверных ресурсов должно быть минимизировано;

– Требования к реализации

Используемые стандарты и языки программирования. В качестве основного языка программирования должен использоваться JavaScript для серверной и клиентской разработки соответственно. Для фронтенда рекомендуется использовать HTML, CSS. Для базы данных предпочтительны MySQL для реляционных данных. Операционная среда: Windows;

– Требования к надежности

Частота и воздействие сбоев. Программный продукт должен иметь высокую доступность. В случае сбоя в системе, платформа должна быть способна восстанавливать данные и продолжать работу без потери критической информации. Возможности восстановления после сбоев;

– Требования к интерфейсу

Внешние сущности и взаимодействие с ними. Программный продукт должен взаимодействовать с внешними платежными системами для проведения транзакций. Все взаимодействия с внешними сервисами должны быть логируемыми с возможностью мониторинга и анализа состояния.

## 1.2 Диаграмма вариантов использования

Гость – незарегистрированный пользователь, который может просматривать товары на сайте и искать их по названию. Связь «include» используется для взаимодействия с функцией «Ввод названия».

Покупатель – зарегистрированный пользователь с доступом к оформлению заказов, оплате товаров, истории заказов, личному кабинету и настройкам. Связь «include» обеспечивает связь между этапами, например, «Заполнение формы» в процессе оформления заказа.

Продавец – пользователь, управляющий объявлениями. Имеет доступ к добавлению и редактированию объявлений, истории объявлений, личному кабинету и настройкам. Связь «include» связывает личный кабинет с историей объявлений.

Администратор – пользователь с правами управления системой. Может мониторить систему, управлять пользователями и объявлениями, изменять настройки. Связь «include» интегрирует управление функциями в общую структуру.

Виды отношений: «Include» используется для обязательных частей функций, таких как «Заполнение формы» при заказе. «Extend» указывает на дополнительные опции, например, «Отправка формы продавцу». Разработанная диаграмма вариантов использования представлена в приложении А.

### 1.3 Выбор стратегии разработки и модели жизненного цикла

Выбор модели жизненного цикла на основе характеристик требований был представлен в таблице 1.

Таблица 1 – Выбор модели жизненного цикла на основе характеристик требований

№ критерия	Критерии категории требований	Каскадная	V-образная	RAD	Инкрементная	Быстрого прототипирования	Эволюционная
1	2	3	4	5	6	7	8
1.	Являются ли требования к проекту легко определяемыми и реализуемыми?	Да	Да	Да	Нет	Нет	Нет
2.	Могут ли требования быть сформулированы в начале ЖЦ?	Да	Да	Да	Да	Нет	Нет
3.	Часто ли будут изменяться требования на протяжении ЖЦ?	Нет	Нет	Нет	Нет	Да	Да
4.	Нужно ли демонстрировать требования с целью их определения?	Нет	Нет	Да	Нет	Да	Да
5.	Требуется ли проверка концепции программного средства или системы?	Нет	Нет	Да	Нет	Да	Да
6.	Будут ли требования изменяться или уточняться с ростом сложности системы (программного средства) в ЖЦ?	Нет	Нет	Нет	Да	Да	Да
7.	Нужно ли реализовать основные требования на ранних этапах разработки?	Нет	Нет	Да	Да	Да	Да

Вычисления: 5 за каскадную, 5 за V-образную, 5 за RAD, 6 за инкрементную, 2 за быстрого прототипирования и 2 за эволюционную.

Итог: На основе результатов заполнения табл. 1 подходящей является инкрементная модель.

Таблица 2 содержит информацию о выборе модели жизненного цикла на основе характеристик команды разработчиков.

Таблица 2 – Выбор модели жизненного цикла на основе характеристик команды разработчиков

№ критерия	Критерии категории команды разработчиков проекта	Каскадная	V-образная	RAD	Инкрементная	Быстрого прототипирования	Эволюционная
1	2	3	4	5	6	7	8
1.	Являются ли проблемы предметной области проекта новыми для большинства разработчиков?	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	Да	Да
2.	Являются ли инструментальные средства, используемые в проекте, новыми для большинства разработчиков?	Да	Да	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	Да
3.	Изменяются ли роли участников проекта на протяжении ЖЦ?	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	Да	Да	Да
4.	Является ли структура процесса разработки более значимой для разработчиков, чем гибкость?	<u>Да</u>	<u>Да</u>	Нет	<u>Да</u>	Нет	Нет
5.	Важна ли легкость распределения человеческих ресурсов проекта?	<u>Да</u>	<u>Да</u>	<u>Да</u>	<u>Да</u>	Нет	Нет
6.	Приемлет ли команда разработчиков оценки, проверки, стадии разработки?	<u>Да</u>	<u>Да</u>	Нет	<u>Да</u>	<u>Да</u>	<u>Да</u>

Вычисления: 5 за каскадную, 5 за V-образную, 4 за RAD, 5 за инкрементную, 2 за быстрого прототипирования и 1 за эволюционную.

Итог: На основе результатов заполнения табл. 2 подходящим являются каскадная, V-образная и инкрементная модели.

В таблице 3 представлен выбор жизненного цикла на основе характеристик коллектива пользователей.

Таблица 3 – Выбор модели жизненного цикла на основе характеристик коллектива пользователей

№ критерия	Критерии категории коллектива пользователей	Каскадная	V-образная	RAD	Инкрементная	Быстрого прототипирования	Эволюционная
1	2	3	4	5	6	7	8
1.	Будет ли присутствие пользователей ограничено в ЖЦ разработки?	Да	Да	<u>Нет</u>	Да	<u>Нет</u>	Да
2.	Будут ли пользователи оценивать текущее состояние программного продукта (системы) в процессе разработки?	Нет	Нет	Нет	<u>Да</u>	<u>Да</u>	<u>Да</u>
3.	Будут ли пользователи вовлечены во все фазы ЖЦ разработки?	Нет	Нет	<u>Да</u>	Нет	<u>Да</u>	Нет
4.	Будет ли заказчик отслеживать ход выполнения проекта?	Нет	Нет	Нет	Нет	<u>Да</u>	<u>Да</u>

Вычисления: 0 за каскадную, 0 за V-образную, 2 за RAD, 1 за инкрементную, 4 за быстрого прототипирования и 2 за эволюционную.

Итог: На основе результатов заполнения табл. 3 подходящей является модель быстрого проектирования.

Выбор жизненного цикла на основе характеристик типа проектов и рисков представлен в таблице 4.

Таблица 4 – Выбор модели жизненного цикла на основе характеристик типа проектов и рисков

№ критерия	Критерии категории типов проекта и рисков	Каскадная	V-образная	RAD	Инкрементная	Быстрого прототипирования	Эволюционная
1	2	3	4	5	6	7	8
1.	Разрабатывается ли в проекте продукт нового для организации направления?	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	Да	Да	Да
2.	Будет ли проект являться расширением существующей системы?	Да	Да	Да	Да	<u>Нет</u>	<u>Нет</u>
3.	Будет ли проект крупно- или среднemasштабным?	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	Да	Да	Да
4.	Ожидается ли длительная эксплуатация продукта?	<u>Да</u>	<u>Да</u>	Нет	<u>Да</u>	Нет	<u>Да</u>
5.	Необходим ли высокий уровень надежности продукта проекта?	Нет	<u>Да</u>	Нет	<u>Да</u>	Нет	<u>Да</u>
6.	Предполагается ли эволюция продукта проекта в течение ЖЦ?	Нет	Нет	Нет	<u>Да</u>	<u>Да</u>	<u>Да</u>
7.	Велика ли вероятность изменения системы (продукта) на этапе сопровождения?	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	Да	Да	Да
8.	Является ли график сжатым?	Нет	Нет	<u>Да</u>	<u>Да</u>	<u>Да</u>	<u>Да</u>
9.	Предполагается ли повторное использование компонентов?	Нет	Нет	<u>Да</u>	<u>Да</u>	<u>Да</u>	<u>Да</u>
10.	Являются ли достаточными ресурсы (время, деньги, инструменты, персонал)?	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	<u>Нет</u>	Да	Да

Вычисления: 5 за каскадную, 6 за V-образную, 6 за RAD, 6 за инкрементную, 4 за быстрого прототипирования и 6 за эволюционную.

Итог: На основе результатов заполнения табл. 4 подходящей является V-образная модель, RAD модель, инкрементная и эволюционная модель.

Общий итог: в итоге заполнения табл. 1 – 4 наиболее подходящей является инкрементная модель.

## 1.4 Инструменты разработки

Разработка проекта будет осуществляться с использованием современного набора технологий и инструментов, что позволит обеспечить высокую производительность, гибкость и стабильность веб-ресурса на всех этапах его создания. Основные технологии:

- HTML (HyperText Markup Language) – язык разметки, который используется для создания структуры веб-страниц. Он является фундаментом любого интернет-ресурса и обеспечивает логичное и понятное расположение всех элементов, включая текстовый контент, изображения, ссылки и медиафайлы;

- CSS (Cascading Style Sheets) – каскадные таблицы стилей, отвечающие за визуальное оформление и стилизацию страниц. Использование CSS позволяет создать привлекательный и адаптивный интерфейс, который будет корректно отображаться на устройствах с разными разрешениями экрана;

- JavaScript – язык программирования, применяемый для добавления интерактивности и динамического поведения на веб-страницах. JavaScript позволяет реализовать обработку пользовательских действий, анимацию, всплывающие окна и другие элементы, повышающие удобство и функциональность сайта.

Для написания и отладки кода будет использоваться Microsoft Visual Studio – мощная и современная интегрированная среда разработки. Она отличается широким спектром функциональных возможностей, включая автодополнение кода, мощные инструменты отладки и поддержку множества языков программирования. Это позволяет значительно ускорить процесс разработки и минимизировать вероятность ошибок в коде. Дополнительные инструменты:

- OpenServer – локальная серверная платформа, включающая в себя сервер Apache, интерпретатор PHP, базу данных MySQL и другие модули. OpenServer позволяет разрабатывать и тестировать веб-приложения в условиях, приближенных к реальному серверному окружению. Благодаря этому можно проверить работу сайта на локальном компьютере, не подключаясь к удалённому хостингу.

- DRAW.IO – бесплатный и удобный веб-инструмент для создания диаграмм и визуализаций. Он поддерживает создание UML-диаграмм, ER-моделей, блок-схем и других графических схем, которые помогут наглядно представить архитектуру и процессы проекта. Визуализация данных играет важную роль как на этапе проектирования, так и при подготовке итоговой документации.

- Microsoft Office Word – инструмент для подготовки текстовой документации, технических отчётов и инструкций. Его использование

					УП ТРПО 2-40 01 01.33.41.04.24 ПЗ	Лист
Изм	Лист	Недокум.	Подпись	Дата		14

обеспечивает высокое качество оформления документов, структурирование текста и удобство дальнейшего редактирования.

– phpMyAdmin – веб-интерфейс для управления базами данных MySQL и MariaDB. Инструмент предоставляет удобный доступ к функциям системы управления базами данных через браузер. С его помощью можно создавать, редактировать и удалять таблицы, выполнять SQL-запросы, а также управлять пользователями и настройками БД.

Разработка проекта будет выполняться на современном и производительном компьютере, который обеспечит высокую скорость выполнения задач, удобство работы с графикой и тестирования ресурсоёмких элементов сайта:

- Операционная система: Windows 11;
- Процессор: AMD Ryzen 9 7945HX;
- Оперативная память: 16 GB;
- Жесткий диск: 1 TB;
- Видеокарта: NVIDIA RTX 4060.

### 1.5 Разработка плана работы над проектом

Диаграмма Ганта — это инструмент управления проектами, предназначенный для визуализации плана и контроля выполнения задач во времени. Её основное назначение заключается в упрощении планирования и управления проектом, позволяя определить задачи, их последовательность, продолжительность и сроки выполнения. Она помогает следить за текущим статусом выполнения, сравнивать фактический прогресс с запланированным, а также выявлять отклонения.

Разработанная диаграмма вариантов использования представлена в приложении Б

## 2 Проектирование задачи

### 2.1 Разработка структуры сайта, системы меню, навигации

Диаграммы структуры сайта и навигации предназначены для визуального представления архитектуры веб-ресурса и организации его элементов. Их основное назначение заключается в упрощении планирования и разработки сайта, создании логичной и интуитивной навигации для пользователей, а также в улучшении взаимодействия между командами разработчиков, дизайнеров и заказчиков.

Эти диаграммы помогают четко представить иерархию разделов и страниц, что позволяет оптимально распределить контент и обеспечить удобный доступ к ключевой информации. Они также способствуют выявлению возможных проблем в навигации, таких как избыточная сложность или дублирование разделов, и помогают их устранить еще на этапе проектирования.

Кроме того, такие диаграммы играют важную роль в создании положительного пользовательского опыта, поскольку четкая структура и понятные пути навигации делают взаимодействие с сайтом более удобным и эффективным. Это особенно важно для повышения удовлетворенности пользователей и достижения их целей, будь то поиск информации, размещение заказа или взаимодействие с сервисами. Данные диаграммы представлены в приложении В

### 2.2 Разработка UML-диаграмм

Модель данных — это абстрактное представление структуры и взаимосвязей данных в системе. Ее основное назначение — организовать данные, определить правила их хранения, обработки и взаимодействия между компонентами системы. Модель данных обеспечивает целостность информации, помогает эффективно управлять ею и интегрировать различные системы. Она служит основой для разработки, поддержки и масштабирования информационных систем, облегчая понимание и использование данных всеми участниками проекта. Созданная модель данных представлена в приложении Г.

Диаграмма классов — это один из видов диаграмм в языке UML, используемый для моделирования структуры системы. Она отображает классы, их атрибуты и методы, а также взаимосвязи между ними. Основная цель диаграммы классов — представить объекты и их взаимодействия внутри программной системы. На диаграмме классов можно увидеть, какие данные и функции принадлежат каждому классу, а также как классы связаны между собой через ассоциации, наследование или агрегацию. Созданная диаграмма вариантов использования представлена в приложении Д.



Диаграмма последовательности в UML описывает взаимодействие объектов системы, отображая порядок сообщений, которые передаются между ними, и их выполнение во времени. Каждый объект изображается вертикальной линией, которая представляет его жизненную линию, показывая его активность в процессе выполнения сценария. Сообщения между объектами представлены горизонтальными стрелками, которые могут быть синхронными (ожидающими ответа) или асинхронными (не ожидающими ответа). Время на диаграмме идет сверху вниз, где верхняя часть представляет собой более ранние события, а нижняя — более поздние. Активность объектов в процессе выполнения операций отображается с помощью прямоугольников, размещенных на их жизненных линиях. Сообщения возврата показывают результат выполнения операции и идут в обратном направлении. В диаграмме также могут быть использованы условные и циклические структуры для обозначения различных вариантов исполнения. Данная диаграмма вариантов использования представлена в приложении Е.

### 3. Реализация

#### 3.1 Создание сайта

Для начала необходимо подготовить окружение. Скачайте и установите OpenServer с официального сайта. После установки запустите сервер и настройте параметры: выберите версию PHP, совместимую с последней версией WordPress, настройте MySQL для базы данных и выберите веб-сервер (Apache или Nginx). Далее откройте phpMyAdmin через интерфейс OpenServer и создайте новую базу данных с кодировкой utf8mb4\_general\_ci, например, под названием site\_db. После подготовки базы данных скачайте последнюю версию WordPress с wordpress.org и распакуйте файлы в директорию domains OpenServer, например, в папку C:\OpenServer\domains\mywebsite.local. Запустите сайт по адресу <http://mywebsite.local> в браузере и следуйте инструкциям установщика WordPress. На этапе конфигурации укажите название базы данных, имя пользователя (по умолчанию root) и пароль (оставьте пустым, если не меняли настройки).

Процесс установки программного продукта включает несколько ключевых шагов, которые обеспечивают полноценную настройку и успешный запуск сайта. Для достижения корректной работы и стабильности веб-приложения важно следовать каждому этапу и соблюдать все рекомендации, указанные в данном разделе. Здесь подробно описаны шаги инсталляции, начиная с подготовки окружения до финального тестирования системы, а также приведен список возможных ошибок, которые могут возникнуть, и способы их устранения. Для этого нужно подготовить следующие инструменты и программное обеспечение:

- OpenServer – локальная серверная платформа, которая включает в себя сервер Apache, систему управления базами данных MySQL и язык программирования PHP. Этот инструмент позволяет создавать локальную серверную среду для разработки и тестирования веб-приложений;
- WordPress – популярная система управления контентом, которая служит основой для создания и администрирования веб-сайта. Перед установкой необходимо скачать последнюю версию WordPress с официального сайта;
- phpMyAdmin – веб-интерфейс для управления базами данных MySQL, который интегрирован в OpenServer и позволяет легко создавать и редактировать базы данных для веб-приложений.

Для начала необходимо скачать и установить OpenServer. После скачивания архива с официального сайта, распакуйте его в подходящую директорию (например, на диск C в папку C:\OpenServer). Затем запустите OpenServer и убедитесь, что серверные модули Apache, MySQL и PHP активированы. На этом этапе могут возникнуть следующие сообщения:

					УП ТРПО 2-40 01 01.33.41.04.24 ПЗ	Лист
Изм	Лист	Недокум.	Подпись	Дата		18

– «Порт 80 занят другим процессом»– это означает, что другой процесс (например, Skype или IIS) использует порт, необходимый для работы веб-сервера. Чтобы решить эту проблему, нужно либо завершить данный процесс, либо изменить настройки OpenServer, указав другой порт;

– «Ошибка запуска MySQL»– если MySQL не запускается, нужно проверить настройки OpenServer и убедиться, что файлы конфигурации корректны.

После того как OpenServer настроен и запущен, можно переходить к настройке базы данных через phpMyAdmin. Для этого откройте браузер и перейдите по адресу <http://localhost/phpmyadmin>. Войдите в систему под стандартными учётными данными (логин: root, пароль: пустой) и создайте новую базу данных, например, wp\_project.

После того как OpenServer установлен, скачайте последнюю версию WordPress с официального сайта. Распакуйте файлы в директорию domains внутри папки с OpenServer (например, C:\OpenServer\domains\project.local). Для того чтобы начать установку, откройте браузер и перейдите по адресу <http://project.local>. На экране появится мастер установки WordPress. На этом этапе вам будет предложено указать параметры подключения к базе данных:

– Имя базы данных: wp\_project (или любое другое имя, которое вы указали при создании базы данных);

– Логин: root;

– Пароль: оставьте пустым (по умолчанию пароль не установлен для пользователя root).

Если возникает ошибка соединения с базой данных, проверьте правильность введённых данных и убедитесь, что база данных была создана в phpMyAdmin. В случае появления сообщения о том, что файл конфигурации wp-config.php не найден, используйте шаблон wp-config-sample.php, чтобы создать файл конфигурации вручную, указав в нём правильные данные для подключения к базе данных.

После настройки базы данных, в процессе установки нужно ввести имя сайта, выбрать язык, а также указать имя администратора, пароль и электронную почту. После успешного завершения установки появится сообщение «WordPress успешно установлен», и вы сможете войти в административную панель вашего сайта, чтобы начать работу с контентом. После успешной установки WordPress, следующим шагом является установка и настройка необходимых плагинов. Для этого зайдите в административную панель WordPress, перейдите в раздел «Плагины» → «Добавить новый». После активации каждого плагина нужно настроить его в соответствии с потребностями проекта. Например, в WooCommerce нужно настроить валюту, способы оплаты и доставки.

После установки и настройки всех компонентов необходимо провести финальное тестирование сайта. Проверяется корректность работы всех страниц, функциональность интернет-магазина, правильность отображения содержимого на разных языках, а также работа шорткодов и виджетов.

### 3.2 Настройка функциональности и дизайна

После установки WordPress начинается этап настройки дизайна и добавления функциональности. В первую очередь выберите и установите тему для сайта. Это можно сделать через раздел Внешний вид → Темы. Выберите тему из репозитория WordPress или загрузите свою кастомную тему в формате .zip. Активируйте выбранную тему. Для расширения возможностей сайта установите следующие плагины:

- Elementor — визуальный конструктор страниц с поддержкой drag-and-drop редактора;
- Elementor Header & Footer Builder — позволяет настраивать хедер и футер через Elementor;
- WooCommerce — плагин для создания интернет-магазина. Он добавляет возможность управления товарами, заказами и оплатами;
- Login/Signup Popup — плагин для создания личных кабинетов пользователей, авторизации и регистрации;

Для установки плагинов перейдите в раздел Плагины → Добавить новый, введите название плагина в поиске, установите и активируйте его.

### 3.3 Редактирование страниц

Добавление и редактирование страниц осуществляется через административный раздел WordPress. В разделе Страницы → Добавить новую создайте необходимые страницы (например, «О нас», «Контакты», «Главная» и т.д.). Для редактирования страниц используйте Elementor. В редакторе Elementor можно создавать страницы, добавляя и редактируя виджеты: текстовые блоки, изображения, кнопки и другие элементы. Структурируйте страницу с помощью секций и колонок, настройте стили блоков, включая шрифты, цвета и отступы. Для каждой страницы определите структуру и контент, необходимый для пользователей

### 3.4 Настройка Header и Footer

Для создания хедера и футера был использован плагин Elementor Header & Footer Builder. В разделе Внешний вид → Header Footer Builder создайте отдельные

шаблоны для верхнего (Header) и нижнего (Footer) блоков сайта. В хедере разместите логотип, основное меню навигации и контактную информацию. В футере добавьте ссылки на важные страницы, контактные данные и иконки социальных сетей. Редактирование этих блоков осуществляется через интерфейс Elementor аналогично работе с обычными страницами. После сохранения шаблонов хедер и футер автоматически отобразятся на всех страницах сайта.

### 3.5 Редактирование главной страницы сайта

Главная страница вашего сайта — это первое, что видит посетитель. Она должна быть одновременно информативной, эстетически привлекательной и удобной в использовании. Эффективная главная страница помогает удержать внимание пользователя, формирует положительное впечатление о вашем бизнесе и способствует конверсии.

Для редактирования главной страницы откройте её в конструкторе Elementor через раздел «Страницы → Все страницы». Создайте структуру, которая будет отвечать задачам вашего проекта, следуя этим рекомендациям. Верхний блок главной страницы — это визуальный центр и отправная точка для пользователя. Разместите здесь крупный, привлекающий внимание заголовок, который чётко передаёт суть вашего предложения или миссию компании. Под заголовком разместите баннер с изображением, которое визуально дополняет текст.

Используйте виджеты «Заголовок» и «Изображение», чтобы легко настроить элементы. Информационные блоки помогают донести основные услуги или преимущества вашего бизнеса. Разделите страницу на секции с короткими текстами, небольшими иконками и тематическими изображениями.

Используйте виджеты «Иконка» и «Текст», чтобы создать визуально привлекательный и легко читаемый раздел. Если на вашем сайте используется WooCommerce, добавьте блок с товарами.

Для обеспечения удобного взаимодействия с сайтом добавьте функционал личного кабинета и регистрации. Используйте блоки из плагина Login/Signup Рорир, чтобы пользователи могли создавать аккаунты, входить в систему, просматривать историю заказов и управлять своими данными.

Главная страница должна отлично выглядеть и на компьютерах, и на мобильных устройствах. Настройте элементы через вкладку «Продвинутый → Адаптивность» в Elementor, чтобы ваш сайт был одинаково удобен для пользователей смартфонов и планшетов. Для достижения наилучшего результата используйте качественные изображения, лаконичный и понятный текст, а также элементы дизайна, которые гармонируют с общей стилистикой сайта. Добавьте кнопки с призывами к действию, такие как «Подробнее», «Связаться», «Купить».

4. Тестирование

4.1 Тест-кейсы

В таблице 5 приведены тест-кейсы

Таблица 5 – Тест-кейсы

Идентификатор	Приоритет	Заглавие и шаги выполнения	Ожидаемый результат	Фактический результат	Статус
1	2	3	4	5	6
T_01	Крайне высокий	<b>Регистрация покупателя</b> 1. Открыть страницу регистрации. 2. Ввести корректные данные в поля: Email, Имя, Фамилия, Пароль, Подтверждение пароля. 3. Нажать кнопку «Продолжить».	1.Открытие страницы регистрации 2. Ввод корректных данных в поля: Email, Имя, Фамилия, Пароль, Подтверждение пароля. 3. Регистрация завершена успешно, покупатель получает уведомление о регистрации и может войти в личный кабинет	Регистрация прошла успешно, покупатель может войти в личный кабинет	Пройдено
T_02	Высокий	<b>Размещение объявления продавцом</b> 1. Войти в аккаунт продавца. 2. Перейти в раздел «Разместить объявление». 3. Заполнить все обязательные поля: название товара, описание, цена и т.д. 4. Нажать «Опубликовать».	1. Вход в аккаунт продавца. 2. Переход в раздел «Разместить объявление». 3. Заполнение обязательных полей. 4.Объявление появляется в списке товаров, доступных для покупки.	Объявление успешно размещено	Пройдено
T_03	Высокий	<b>Поиск товара</b> 1. Открыть главную страницу. 2. Ввести запрос в строку поиска (например, «яблоки»). 3. Нажать кнопку «Поиск».	1. Открытие главной страницы. 2. Ввод запроса в строку поиска. 3. Отображаются товары, соответствующие запросу.	Товары, соответствующие запросу, отображаются	Пройдено
T_04	Крайне высокий	<b>Оформление заказа покупателем</b> 1. Войти в аккаунт покупателя. 2. Выбрать товар и добавить его в корзину. 3. Перейти в корзину и оформить заказ. 4. Ввести данные для доставки и подтвердить покупку.	1. Вход в аккаунт покупателя. 2. Выбор и добавление товара в корзину. 3. Переход в корзину и оформление заказа. 4. Заказ успешно оформлен, покупатель получает уведомление.	Заказ оформлен, уведомление получено	Пройдено
T_05	Низкий	<b>Регистрация с некорректным Email</b> 1. Открыть страницу регистрации. 2. Ввести некорректный Email (например, «invalidemail.com»). 3. Заполнить остальные поля и нажать «Продолжить».	1. Открытие страницы регистрации. 2. Ввод некорректного Email. 3. Появляется сообщение об ошибке с просьбой ввести корректный Email.	Сообщение об ошибке отображено	Пройдено



Продолжение таблицы 5

1	2	3	4	5	6
T_13	Низкий	<b>Проверка фильтрации товаров по категориям</b> 1. Открыть страницу каталога товаров. 2. Выбрать категорию товаров. 3. Проверить, что отображаются только товары из выбранной категории.	1. Открытие страницы каталога товаров. 2. Выбор категории товаров. 3. Отображаются только товары из выбранной категории.	Товары фильтруются корректно по категориям	Пройдено
T_14	Низкий	<b>Попытка обновить информацию о товаре без обязательных полей</b> 1. Войти в аккаунт продавца. 2. Перейти в раздел «Мои товары». 3. Выбрать товар для редактирования. 4. Удалить обязательные поля (например, цену или описание). 5. Нажать «Сохранить изменения».	1. Вход в аккаунт продавца. 2. Переход в раздел «Мои товары». 3. Выбор товара для редактирования. 4. Удаление всех обязательных полей. 5. Появляется сообщение об ошибке, что обязательные поля не могут быть пустыми.	Ошибка отображена, товар не обновлён	Пройдено
T_15	Высокий	<b>Проверка процесса оформления заказа с выбором способа доставки</b> 1. Добавить товар в корзину. 2. Перейти в корзину. 3. Выбрать способ доставки. 4. Оформить заказ.	1. Добавление товара в корзину. 2. Переход в корзину. 3. Выбрать способ доставки. 4. Способ доставки выбран, заказ оформляется с указанным способом доставки.	Способ доставки выбран, заказ оформлен	Пройдено
T_16	Высокий	<b>Проверка перехода в справочную систему</b> 1. Открыть главную страницу сайта. 2. Нажать на кнопку «Справка» в меню навигации. 3. Дождаться загрузки новой страницы.	1. Открытие главной страницы сайта. 2. Нажатие кнопку «Справка» в меню навигации. 3. Пользователь перенаправляется на страницу справочной системы. Отображается раздел с общей информацией.	Переход выполнен успешно, страница справочной системы загрузилась, элементы отображаются корректно.	Пройдено

УП ТРПО 2-40 01 01.33.41.04.24 ПЗ



## 4.2 Результаты тестирования

Результаты тестирования подтвердили, что веб-приложение для оптовой торговли зерновыми культурами работает стабильно и соответствует заявленным требованиям. Были проверены ключевые модули: регистрация, авторизация, работа с объявлениями, поиск и фильтрация товаров, оформление заказов и справочная система. Расписание работ над проектом представлено в таблице 6.

Таблица 6 – Расписание работ над проектом

Имя	Дата	Деятельность	Продолжительность,ч
Ганисевский Александр	11.12.2024	Разработка тестов	3
Ганисевский Александр	13.12.2024	Тестирование web-сайта	2
Ганисевский Александр	15.12.2024	Составление отчетов о найденных дефектах	2
Ганисевский Александр	16.12.2024	Исправление найденных ошибок	3
Ганисевский Александр	17.12.2024	Проведение регрессионного тестирования	2
Ганисевский Александр	18.12.2024	Составление отчета о результатах тестирования	2

На основании тестирования можно заключить, что приложение полностью готово к эксплуатации. Все выявленные в процессе тестирования дефекты были устранены. Система демонстрирует стабильность, удобство и соответствует современным стандартам качества, что делает её надёжным инструментом для пользователей. Статистика по всем дефектам представлена в таблице 7.

Таблица 7 – Статистика по всем дефектам

Статус	Количество	Важность			
		Низкая	Средняя	Высокая	Критическая
Найдено	0	0	0	0	0
Исправлено	0	0	0	0	0
Проверено	0	0	0	0	
Открыто заново	0	0	0	0	0
Отклонено	0	0	0	0	0

## 5. Руководство пользователя

### 5.1 Общие сведения о программном продукте

Разрабатываемый программный продукт представляет собой интернет-магазин, построенный на платформе WordPress с использованием локального сервера OpenServer на этапе разработки и тестирования. Сайт предназначен для организации онлайн-продаж товаров и предлагает пользователям удобную платформу для выбора, заказа и покупки продукции. Назначение приложения заключается в предоставлении возможности продавать товары через интернет в автоматизированном режиме. Благодаря интуитивно понятному интерфейсу клиенты могут просматривать товары, добавлять их в корзину, оформлять заказы и получать подтверждения. Администраторы получают доступ к управлению ассортиментом, обработке заказов и аналитике продаж через панель управления WordPress. Управление и продажу товаров через WooCommerce. Основные возможности программы включают:

- Регистрацию и авторизацию пользователей с помощью Login/Signup Popup, что обеспечивает доступ к личным кабинетам;
- Личный кабинет пользователя: С помощью Login/Signup Popup пользователи могут зарегистрироваться, войти в свой профиль, управлять своими заказами;
- Управление товарами и заказами: Реализовано на основе WooCommerce, что позволяет добавлять товары, устанавливать цены, управлять категориями и отслеживать статус заказов;
- Шорткоды для функциональных элементов: Широкое использование шорткодов позволяет внедрять дополнительные динамические элементы, такие как формы, слайдеры, кнопки и другие блоки, без необходимости написания кода.

Основные характеристики программного продукта включают простоту управления контентом, возможность быстрой кастомизации дизайна и расширение функционала за счет плагинов и шорткодов. Область применения продукта охватывает малый и средний бизнес, который занимается оптовой и розничной продажей товаров. Веб-сайт может использоваться для магазинов, маркетплейсов, сервисных компаний и информационных порталов. Периодичность использования предполагает регулярное обновление информации: добавление новых товаров, управление заказами и взаимодействие с клиентами через личные кабинеты. Среда функционирования включает работу на веб-сервере (Apache/Nginx), с базой данных MySQL, управляемой через phpMyAdmin. Для корректной работы сайта необходимо:

- 512 МБ оперативной памяти или выше;

					УП ТРПО 2-40 01 01.33.41.04.24 ПЗ	Лист
Изм	Лист	Недокум.	Подпись	Дата		26

- Объем внешней памяти от 1 ГБ, в зависимости от количества контента и медиафайлов;
- Поддержка PHP версии 7.4 и выше.

Для пользователей сайта требуется устройство с экраном разрешением не менее 1280x720 пикселей и подключение к интернету. Периферийное оборудование, такое как клавиатура и мышь, необходимо для полноценного взаимодействия с интерфейсом сайта, особенно для администраторов и менеджеров, работающих с контентом и заказами.

## 5.2 Запуск программы

Запуск программы осуществляется через веб-браузер на локальном сервере, который настроен с использованием OpenServer. Для начала необходимо открыть папку OpenServer на вашем компьютере и запустить исполняемый файл Open Server.exe. После этого в системном трее появится значок OpenServer, где нужно выбрать команду «Запустить сервер». При успешном запуске сервера на экране появится уведомление в соответствии с рисунком 1. После запуска сервера необходимо открыть веб-браузер и в адресной строке ввести локальный адрес сайта, например, <http://project.local>. Главная страница сайта успешно загрузится, предоставляя доступ к основным функциям. Пользователь сможет взаимодействовать с интерфейсом сайта и использовать его функционал в соответствии со своим уровнем доступа (рисунок 2).

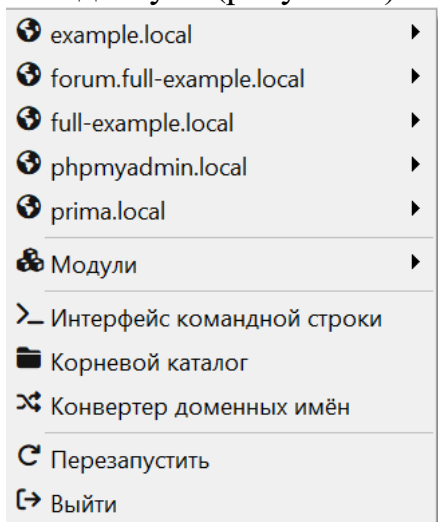


Рисунок 1 - Меню OpenServer

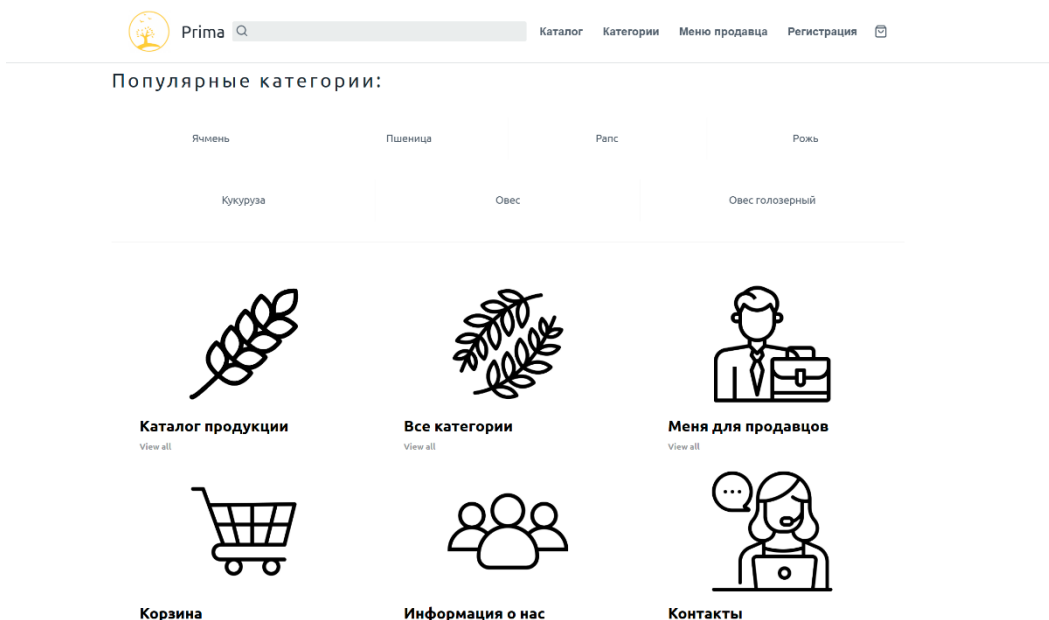



Рисунок 2 - Главное меню сайта

### 5.3 Инструкции по работе с программой

Программа предлагает пользователю удобный и интуитивно понятный интерфейс. Основные действия включают просмотр каталога, работу с личным кабинетом и управление языками интерфейса. На главной странице располагаются ключевые разделы: Каталог товаров, Личный кабинет и языковой переключатель. Для доступа к каталогу пользователь может выбрать соответствующий пункт в меню или нажать на кнопку на главной странице. При добавлении товара в корзину программа отобразит уведомление о том, что товар успешно добавлен. Для регистрации и авторизации необходимо перейти в раздел «Личный кабинет» (рисунок 3). Новые пользователи могут зарегистрироваться, заполнив форму с именем пользователя, электронной почтой и паролем. После успешной авторизации программа выводит приветственное сообщение с именем пользователя. В верхнем меню можно выбрать нужный язык, после чего интерфейс сайта обновится и отобразится на выбранном языке. Для администратора предусмотрена возможность редактирования страниц через панель управления WordPress (рисунок 4). Необходимо открыть административную панель по адресу <http://project.local/wp-admin>, выбрать нужную страницу и запустить редактор Elementor для внесения изменений.



**Alex Brooks**  
[Просмотр профиля](#)

**Учётная запись**

Имя пользователя

Имя

Фамилия

Email Address

Обновить учётную запись

- Учётная запись
- Изменить пароль
- Безопасность
- Удалить учётную запись

Рисунок 3 – Личный кабинет

- Консоль
- Записи
- Медиафайлы
- Страницы**
- Все страницы
- Добавить страницу
- Команды
- Комментарии
- Ultimate Member
- Marketplace
- WooCommerce
- Товары
- Платежи
- Аналитика
- Маркетинг

**Страницы** [Добавить стр.](#)

Включена опция "Членство - Л пользователи могли регистри страницу wp-login.php открыто

Do you enjoy Orbit Fox? Become  
[Sure, I would love to help.](#)

Все (17) | Опубликованные (17) |

Действия

- ☐ Заголовок
- ☐ Home — Главная страни
- ☐ Store Manager — Страни
- ☐ Войти — UM Войти

Рисунок 4 – Панель администратора

## 5.4 Завершение работы с программой

Завершение работы с программой начинается с выхода из личного кабинета. Пользователю рекомендуется нажать на кнопку «Выйти» в меню, чтобы завершить текущую сессию (рисунок 5). Это обеспечивает безопасность данных и корректное завершение работы с программой. Для остановки локального сервера необходимо щёлкнуть правой кнопкой мыши по значку OpenServer в системном трее и выбрать команду «Выйти» (рисунок 6). После этого на экране появится уведомление о том, что сервер успешно остановлен. Завершив работу с сервером, можно закрыть все окна браузера и другие программы, связанные с работой сайта. Некорректное завершение работы, например, принудительное закрытие OpenServer, может вызвать ошибки при следующем запуске. Рекомендуется следовать описанным инструкциям для стабильной работы программы.

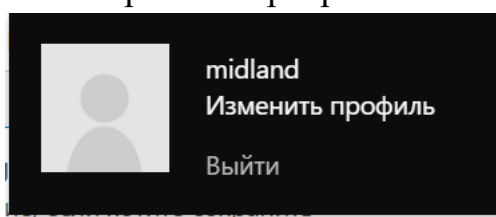


Рисунок 5 – Выход из личного кабинета

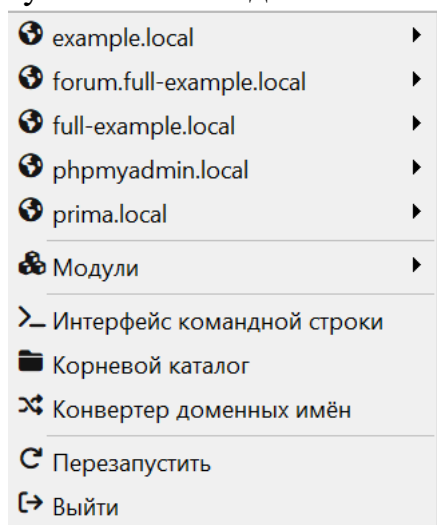


Рисунок 6 – Остановка локального сервера

## 5.5 Использование системы справочной информации

Система справочной информации предназначена для оказания помощи пользователям в процессе работы с программой. Справка доступна через специальный раздел на главной странице сайта под названием «Справка» (рисунок

7). В разделе собраны текстовые инструкции и ответы на популярные вопросы пользователей, что позволяет быстро найти нужную информацию.

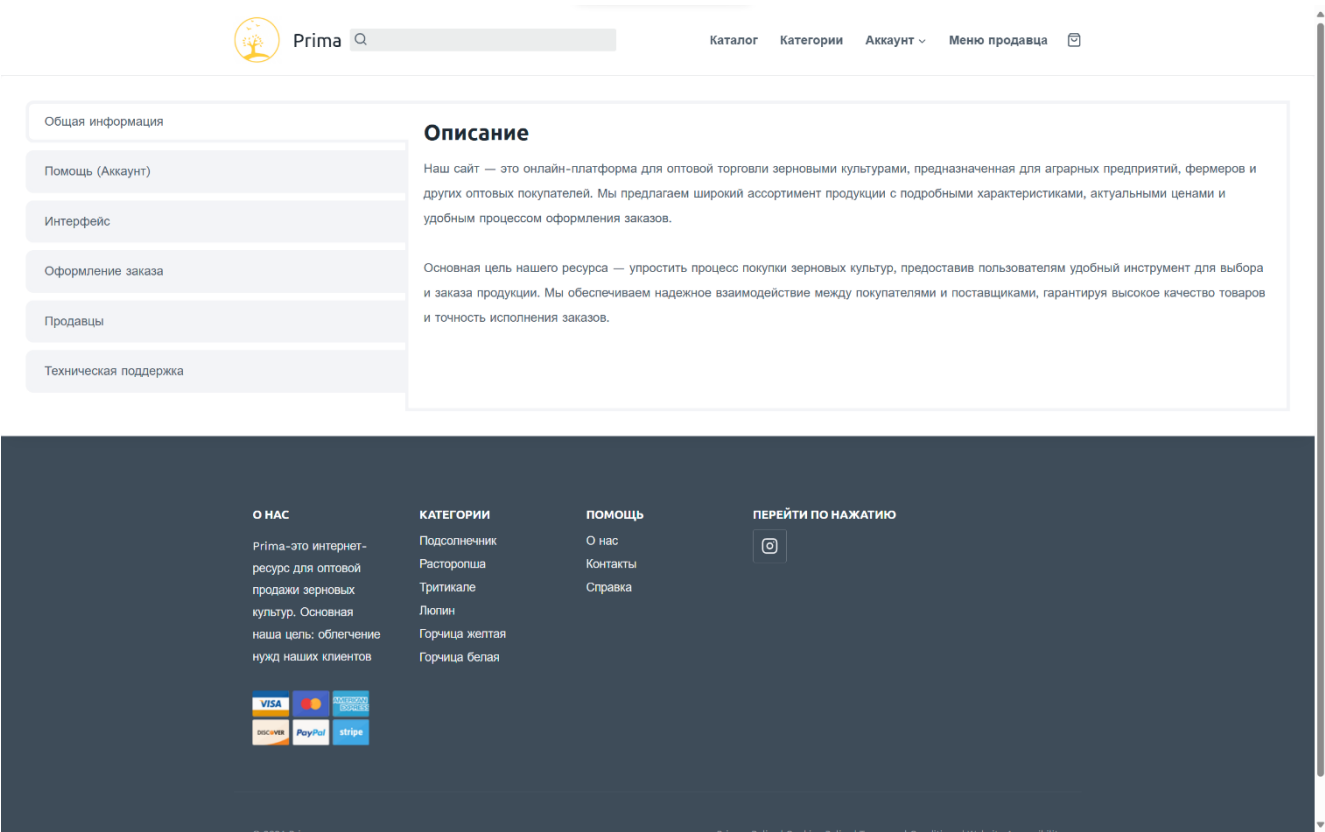


Рисунок 7 – Меню справки

## Заключение

В ходе выполнения работы был разработан и внедрён программный продукт — интернет-ресурс на базе CMS WordPress с использованием современных технологий и инструментов, включая WooCommerce, Login/Signup Popup, а также конструктор страниц Elementor. Для создания визуально привлекательного и функционального интерфейса была использована тема Kadence, которая предоставляет гибкие возможности настройки дизайна и повышает удобство работы с сайтом.

В процессе разработки особое внимание было уделено качеству пользовательского опыта, интеграции необходимых плагинов и настройке интерфейса. Благодаря использованию Kadence, удалось создать современный, адаптивный и интуитивно понятный сайт, который соответствует требованиям пользователей и обеспечивает стабильную работу на различных устройствах. Дополнительно была подключена справочная система, которая предоставляет пользователям доступ к инструкциям, ответам на часто задаваемые вопросы и другой полезной информации. Это значительно облегчает процесс взаимодействия с сайтом и позволяет пользователям быстро находить решения возникающих вопросов.

Результаты тестирования продемонстрировали высокий уровень готовности программного продукта к эксплуатации. Все ключевые функции, такие как:

- просмотр каталога товаров;
- регистрация и авторизация пользователей;
- переключение языков интерфейса;
- взаимодействие с базой данных;
- работа со справочной системой.

Были проверены и успешно функционируют без критических ошибок. Сайт обеспечивает стабильную работу, удобный доступ к информации и соответствует современным требованиям к производительности и безопасности. Ключевыми преимуществами разработанного продукта являются:

- Гибкость и адаптивность: благодаря теме Kadence и плагину Elementor, сайт легко адаптируется под нужды пользователей и обладает современным дизайном;
- Функциональность: интеграция WooCommerce и Login/Signup Popup позволила реализовать управление товарами, заказами и пользователями, что критически важно для коммерческого использования;
- Интерактивность и поддержка: подключение справочной системы помогает пользователям эффективно решать возникающие проблемы и улучшает общее восприятие продукта.

					УП ТРПО 2-40 01 01.33.41.04.24 ПЗ	Лист
Изм	Лист	Недокум.	Подпись	Дата		32



Таким образом, можно сделать вывод, что поставленные задачи были успешно выполнены, а цели работы достигнуты. Разработанный программный продукт обладает высокой степенью готовности к эксплуатации и может быть использован в реальных условиях для решения задач бизнеса и пользователей. Благодаря использованию современных инструментов и тщательному тестированию, сайт демонстрирует стабильность, функциональность и готовность к дальнейшему развитию и масштабированию.

					УП ТРПО 2-40 01 01.33.41.04.24 ПЗ	Лист
Изм	Лист	№ докум.	Подпись	Дата		33

## Список использованных источников

Википедия [Электронный ресурс]. – Электронные данные. – Режим доступа: [Википедия — свободная энциклопедия](#)

Зерновой портал Центрального Черноземья [Электронный ресурс]. – Электронные данные. – Режим доступа: [Зерновой портал Центрального Черноземья | Стандарты | Классификация зерновых культур](#)

Брестский государственный университет имени А.С. Пушкина [Электронный ресурс]: Зерновые культуры. – Электронные данные. – Режим доступа: [bosh\\_24.04.2020.docx](#)

Reddit [Электронный ресурс]. – Электронные данные. – Режим доступа: [Reddit - Dive into anything](#)

Гродненский государственный университет имени Янки Купалы [Электронный ресурс]. – Электронные данные. – Режим доступа: [Глава 2. ЗЕРНОВЫЕ КУЛЬТУРЫ](#)

**Приложение А**  
**Диаграмма вариантов использования**

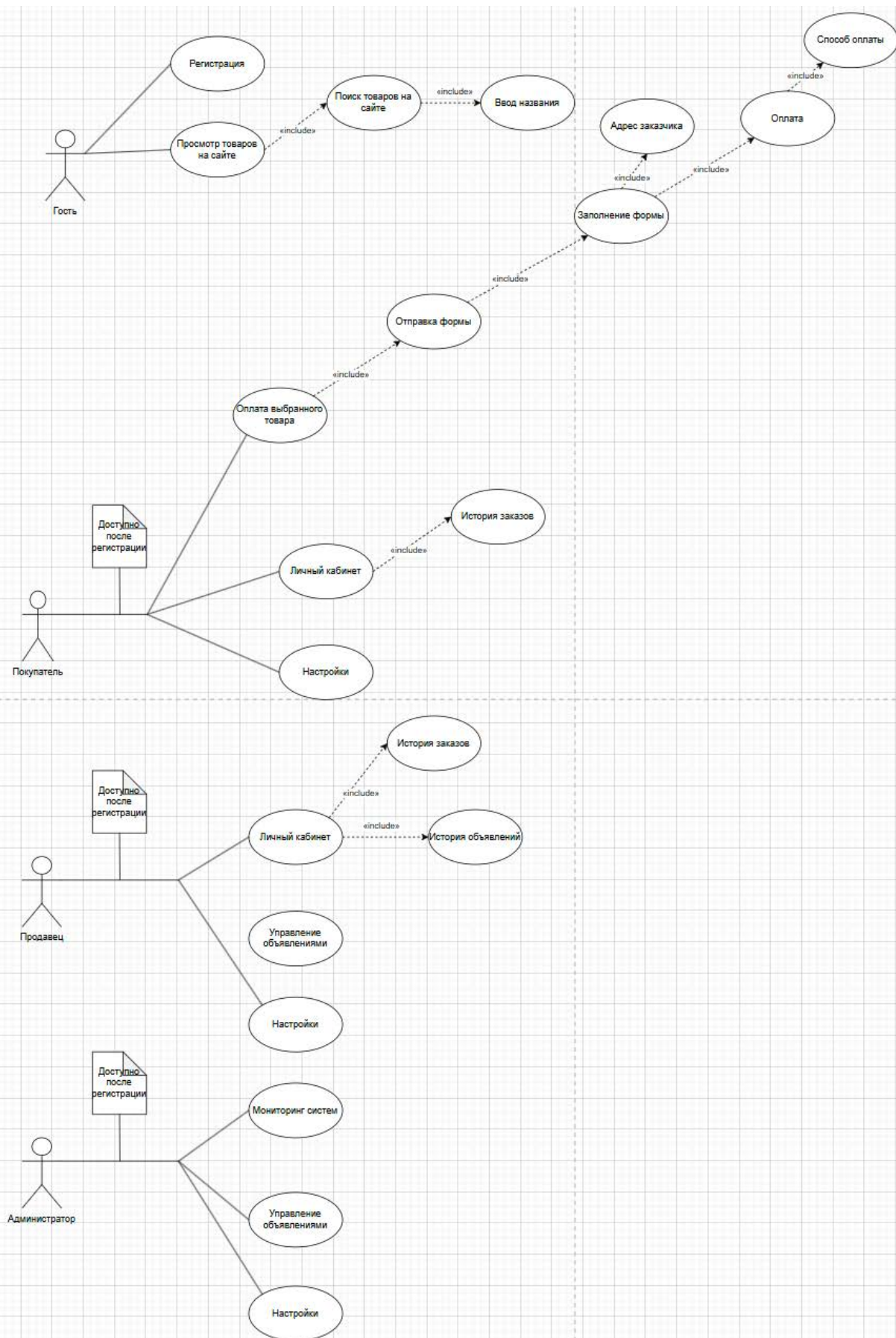


Рисунок А.1 – Диаграмма вариантов использования

**Приложение Б**  
Диаграмма Ганта



## **Приложение В**

### Структура сайта



Рисунок В.2 – Структура главного меню



## **Приложение Г**

### **Модель данных**

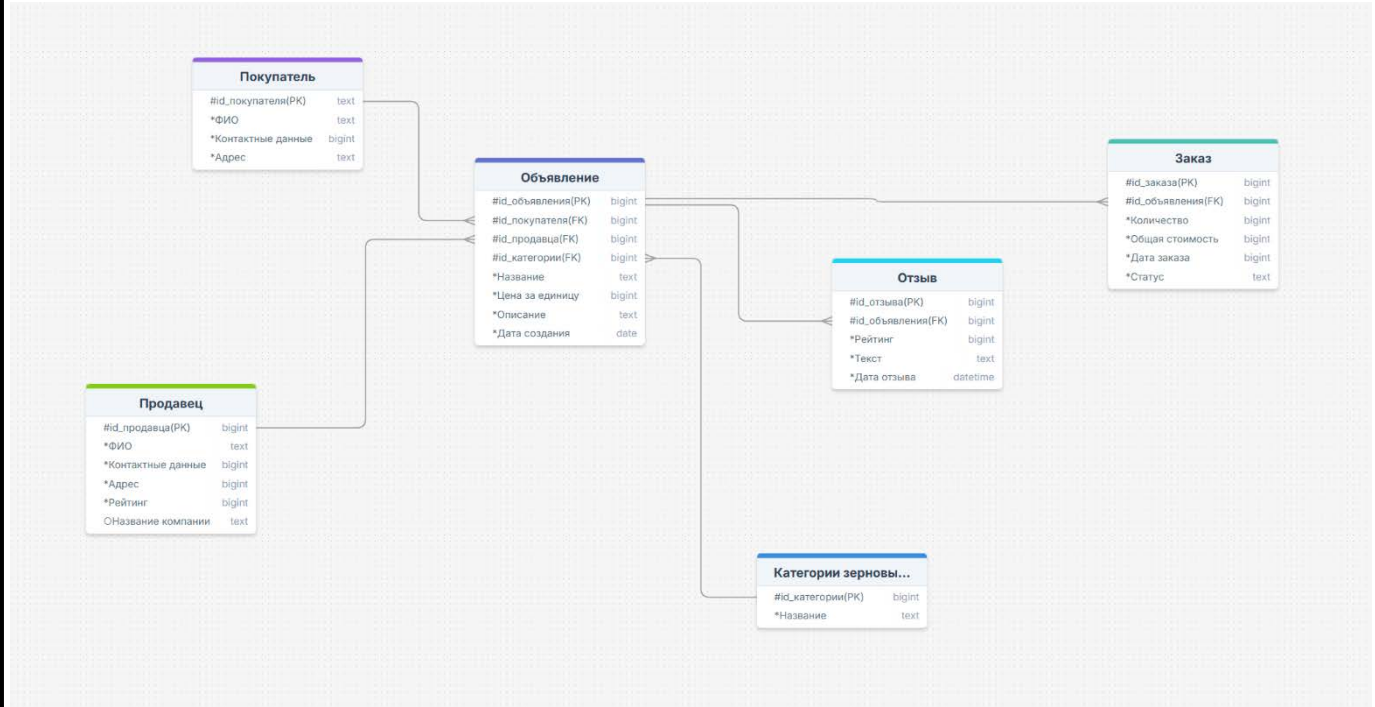


Рисунок Г.1 – Модель данных

**Приложение Д**  
**Диаграмма классов**

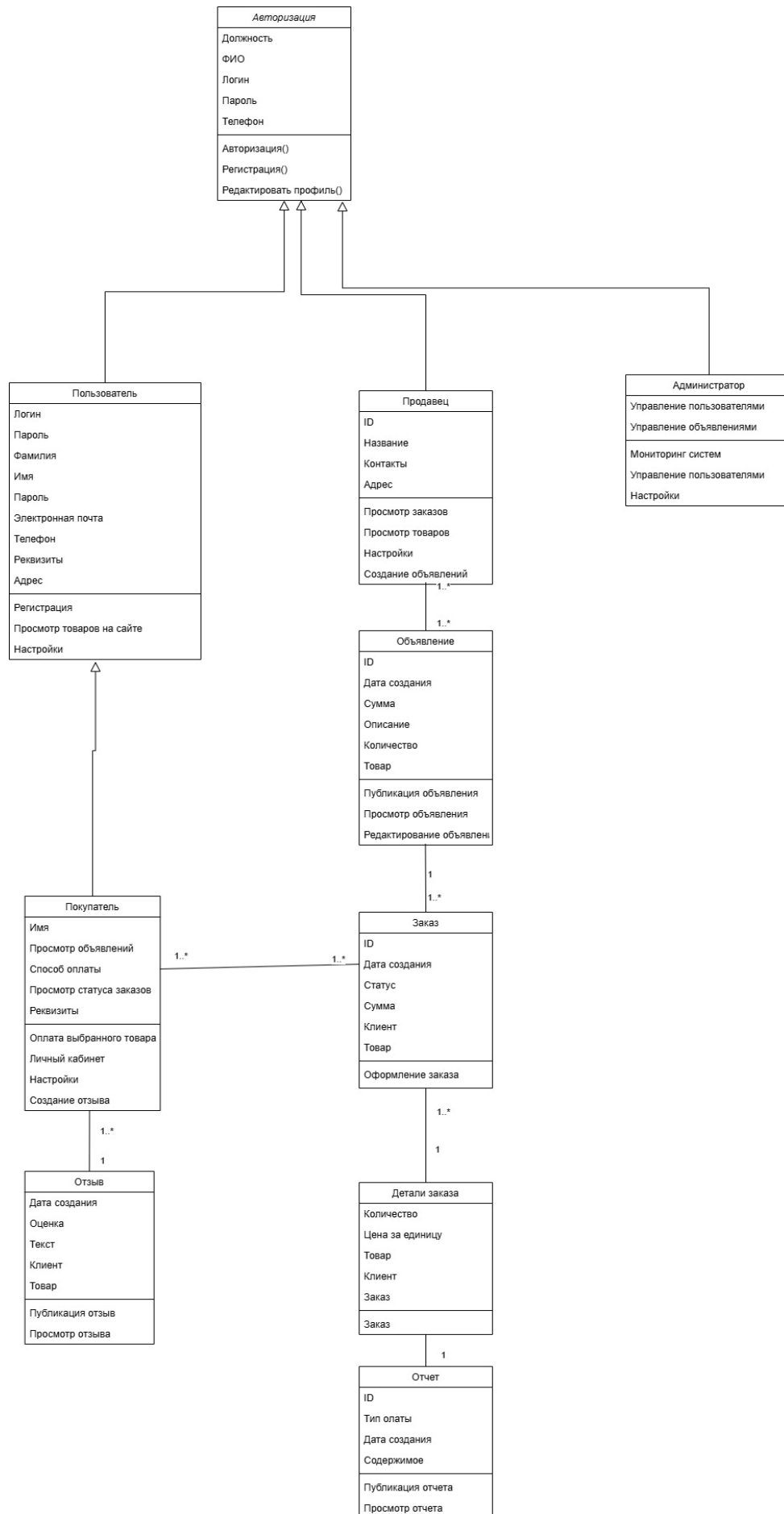


Рисунок Д.1 – Диаграмма классов

**Приложение Е**  
Диаграмма последовательности

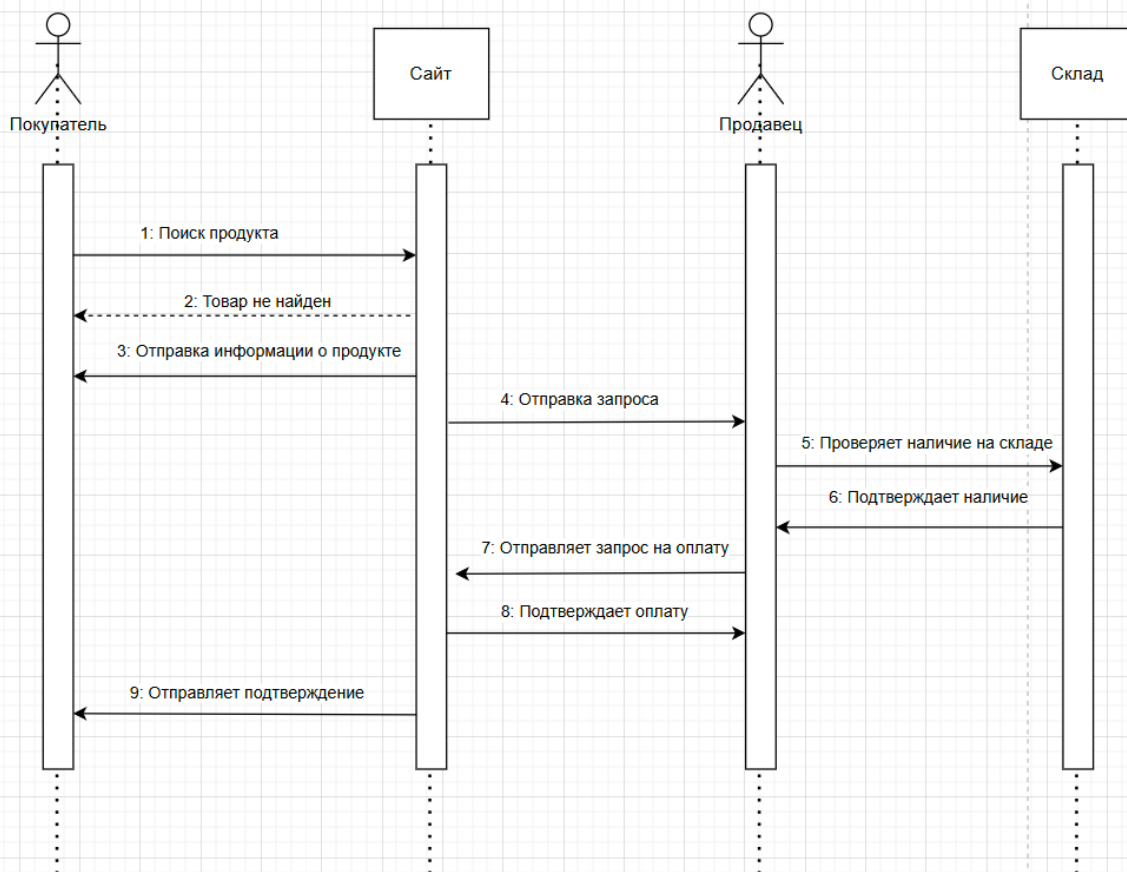


Рисунок Е.1 – Диаграмма последовательности

**Приложение Ж**  
UX, UI макеты

Desktop - 1

✕ Prima

### Регистрация

Фамилия и Имя:

Email:

Пароль:

Назад Далее

Рисунок Ж.1 – Страница регистрации

Desktop - 2

✕ Prima

### Вход

Эл. почта:

Пароль:

Назад Вход

Рисунок Ж.2 – Страница входа

					УП 2-40 01 01.33.41.04.24 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		48





Рисунок Ж.3 – Главная страница (UX)

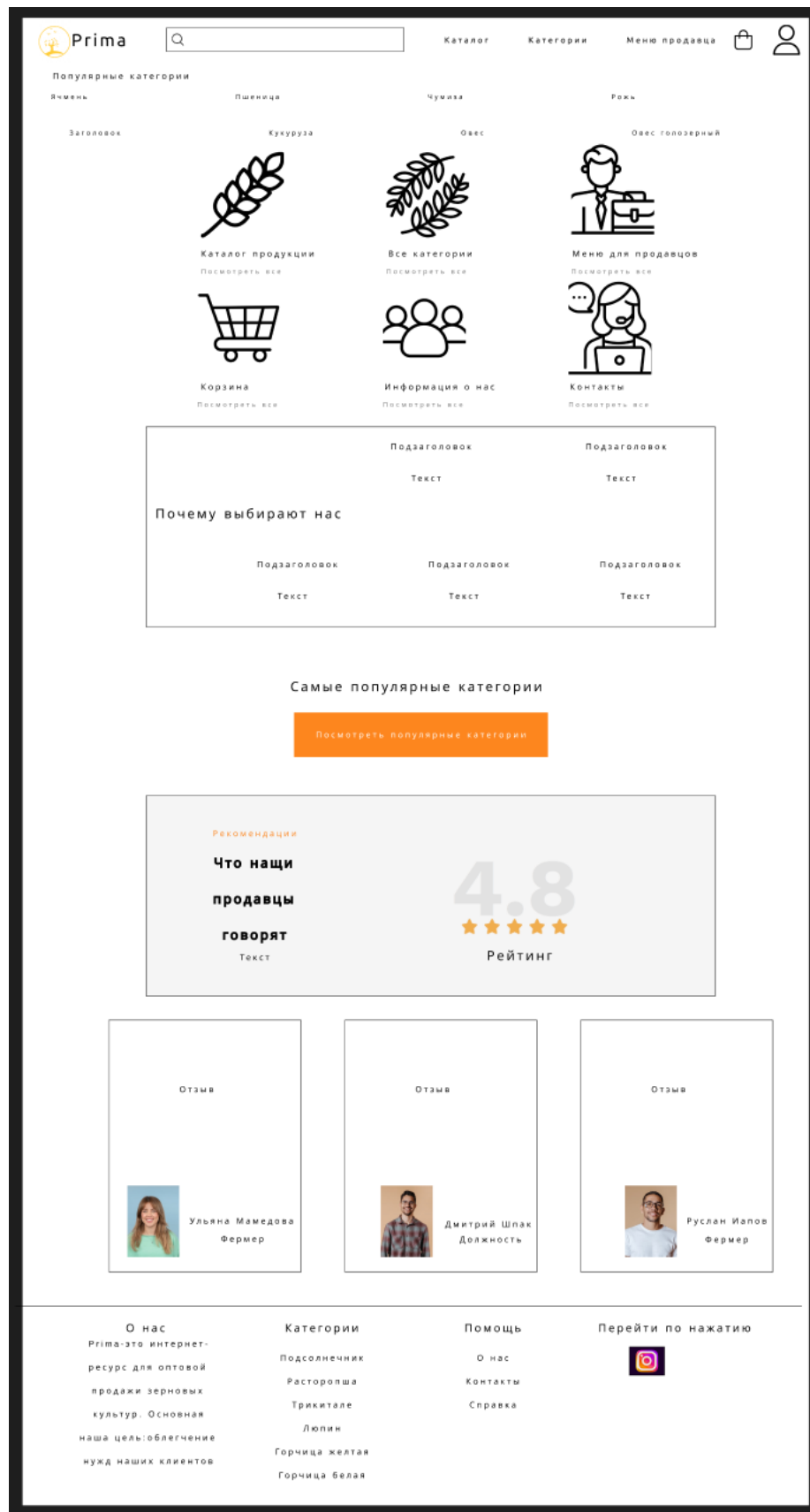


Рисунок Ж.4 – Главная страница (UI)