

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования

**«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

**ЛЕСОСИБИРСКИЙ ПЕДАГОГИЧЕСКИЙ ИНСТИТУТ –  
филиал Сибирского федерального университета**

Высшей математики, информатики, экономики и естествознания  
кафедра

УТВЕРЖДАЮ

Заведующий кафедрой

 Л.Н. Храмова

подпись      инициалы, фамилия

« 14 » 06 2024 г.

## БАКАЛАВРСКАЯ РАБОТА

09.03.02 Информационные системы и технологии  
код-наименование направления

РАЗРАБОТКА WEB-САЙТА ЦВЕТОЧНЫЙ МАГАЗИН

Руководитель

 14.06.2024

подпись, дата

доктор технических наук.

должность, ученая степень

А.П. Мохирев  
инициалы, фамилия

Выпускник

 14.06.2024

подпись, дата

В.В. Шиверёв  
инициалы, фамилия

Нормоконтролер

 14.06.2024

подпись, дата

Е.В. Киргизова  
инициалы, фамилия

Лесосибирск 2024

## РЕФЕРАТ

Выпускная квалификационная работа по теме «Разработка Web-сайта цветочный магазин» содержит 56 страниц текстового документа, 7 иллюстраций, 4 таблицы, 41 использованный источник.

Цель исследования – обосновать и разработать web-приложение для осуществления онлайн заказов покупки цветов.

Объект исследования – web-приложение для осуществления онлайн заказов покупки цветов.

Предмет исследования – процесс создания и сопровождения web-приложения для осуществления онлайн заказов при покупке цветов.

Для достижения поставленной цели необходимо решить следующие задачи:

- на основе анализа учебной и технической документации по теме исследования, определить инструментальные средства для разработки web-приложения и их характеристики, языки программирования и области их применения;
- рассмотреть структуру и создание пользовательского интерфейса web-приложения для осуществления онлайн заказов покупки цветов;
- разработать web-приложение для осуществления онлайн заказов для покупки цветов.

Во время разработки web-приложения для осуществления онлайн заказов для покупки цветов изучены основы таких языков программирования, как JavaScript, PHP, а также язык разметки HTML и язык стилей CSS.

## СОДЕРЖАНИЕ

Введение.....	3
1 Теоретические основы создания цветочного web-приложения для создания онлайн заказов.....	6
1.1 Анализ существующих web-приложений цветочного магазина .....	8
1.2 Отличия web-приложения от web-сайта .....	10
1.3 Языки программирования для разработки .....	13
1.4 Моделирование архитектуры сайта.....	14
2 Разработка web-приложения для осуществления онлайн заказов.....	18
2.1 Проектирование дизайна интерфейса для web-приложения .....	18
2.2 Разработка базы данных .....	22
2.3 Анализ целевой аудитории.....	25
2.4 Разработка серверной части .....	31
Заключение .....	35
Список использованных источников .....	36
Приложение А Компонент Flower Catalog .....	40
Приложение Б Компонент React Flower Cart Address Delivery.....	43
Приложение В Компонент React Flower Cart Alert Message .....	45
Приложение Г Компонент React Flower Cart Sum Info .....	47
Приложение Д Компонент Flower Cart Summary в React .....	49
Приложение Е Компонент Confirm Page в React.....	52
Приложение Ж Компонент Confirm Page Order в React .....	54

## ВВЕДЕНИЕ

В современном мире, с бурным развитием интернет-технологий, онлайн-торговля уверенно завоевывает все новые рынки. Практически любая компания, будь то крупная сеть магазинов или небольшой магазинчик, может вести коммерческую деятельность и совершать продажи через интернет.

Интернет не только предоставил бизнесу новые возможности, но и значительно снизил затраты на проведение торговых операций и сделок. Онлайн-магазины превратились в крупную и значимую отрасль не только онлайн-бизнеса, но и розничной торговли в целом.

Несмотря на высокую конкуренцию в торговле цветами, спрос на них остается стабильным. Флористический бизнес, как и многие другие, успешно адаптировался к новым реалиям.

Создание интернет-магазина может стать как самостоятельным проектом, так и эффективным инструментом для расширения уже существующего цветочного салона.

Интернет-магазин – это не просто сайт, а полноценный инструмент для развития и масштабирования бизнеса. Он позволяет не только увеличить продажи, но и повысить узнаваемость бренда, улучшить качество обслуживания клиентов и выйти на новый уровень развития.

Интернет-магазины получают большую прибыль не только за счёт продажи самих цветов, но также и за счёт доставки, также огромную роль играет отсутствие проблем с возвратом товара, что характерно для традиционного магазина.

Уже большое количество регионов страны оснащены большим количеством, успешно работающих онлайн цветочных магазинов. Большой спрос на онлайн покупку обоснован в связи с постоянной востребованностью услуги доставки букетов, что экономит время потребителя. Заказ на онлайн-ресурсе можно оформить в любое удобное время из любой точки, а также выбрать удобное место и время доставки.

Цель исследования – обосновать и разработать цветочный web-магазин для онлайн заказов.

Объектом исследования – разработка web-магазина по продаже цветов. Для достижения поставленной цели необходимо решить следующие задачи:

- на основе анализа учебной и технической документации по теме исследования определить инструменты, используемые для разработки цветочного веб-магазина, их характеристики, языки программирования и области их применения.

- рассмотреть структуру и создание пользовательского интерфейса цветочного web-магазина для создания онлайн заказов;

- разработать цветочный web-магазин.

Теоретические методы исследования: Изучение и анализ существующих учебных и научных источников, связанных с темой исследования, для выявления текущих тенденций, методов и результатов. Обобщение полученных данных и проведение сравнительного анализа для выявления сходств и различий в подходах, методах и результатах предыдущих исследований.

Эмпирические методы исследования: разработка концепции и структуры программного продукта, создание моделей и прототипов для дальнейшей реализации. Техническая поддержка и обслуживание программного продукта на всех этапах его жизненного цикла, включая обновление, исправление ошибок и улучшение функциональности. Проверка программного продукта на соответствие заданным требованиям, выявление и устранение дефектов, оценка его надежности, производительности и удобства использования.

Структура работы – работа состоит из введения, двух глав, заключения, приложений и списка использованных источников, включающих в себя 41 наименование. Результаты работы представлены в 7 иллюстрациях, 4 таблицах. Общий объем работы – 56 страниц.

## **1 Теоретические аспекты разработки цветочного web-приложения для создания онлайн заказов**

Для создания web-приложения цветочного магазина следует учитывать теоретические аспекты, которые эффективны для разработки удобного и надёжного конечного продукта.

Первым шагом следует выделить анализ предметной области потребностей пользователей, предпочтение клиентов, а также анализ конкурентных предприятий направленных на онлайн продажу цветов.

Вторым шагом следует выделить составление списка основных функций приложения, таких как каталог товаров, корзина пользователя, система онлайн-оплаты, доставка или самовывоз товара, учёт и управление заказами и отзывы клиентов.

Третий шаг разработки включает в себя Frontend разработку (создание пользовательского интерфейса). Для его создания чаще всего используют такие языки программирования, как HTML, CSS, и JavaScript [1].

Четвёртый шаг разработки включает в себя тестирование реализации Frontend разработки, а именно: usability тестирование интерфейса.

Usability, тестирование интерфейса – это тестирование пользовательского интерфейса для учёта принципов удобства использования адаптивного дизайна и корректного отображения на различных устройствах [36].

Пятым шагом служит Backend-разработка, а именно определение как будет работать серверная часть приложения, как будет проходить взаимодействие с базой данных, как будет проходить обработка запросов пользователей, а также Backend-разработка включает в себя обеспечение безопасности данных клиентов. Backend-разработка включает в себя выбор языка программирования, такие как Node.js, Python, Ruby или PHP, определение системы управления базой данных и выбор базы данных (например, MySQL, Postgre SQL, Mongo DB) [32].

База данных служит хранилищем данных: о товарах, заказах, клиентах, и их транзакциях, а также база данных создаётся для оптимизации производительности серверной части web-приложения и достигается посредством нормализации данных, создание таблиц, связей и индексов [18]. Обеспечение безопасности данных клиентов достигается посредством реализации методов защиты пользовательских данных методом шифрование, аутентификация и авторизация. Так для обеспечения безопасности помимо вышесказанных могут использоваться SQL-инъекции, XSS (межсайтовый скриптинг) и CSRF (межсайтовая подделка запросов) [9].

Следующим шагом обеспечения безопасности служит безопасность интеграции с внешними платёжными сервисами. Обеспечение безопасности платежей следует интегрировать через популярные платёжные системы такие как Sber Pay и Mir Pay. После создания Backend-разработки следует провести функциональное тестирование web-приложения. Функциональное тестирование, включает в себя проверку работы всех перечисленных выше функций, а именно полный процесс оформления заказа, оформление платежа, а также проверку работоспособности управление профилем пользователя [16].

Следующий вид тестирования, который следует применить после разработки, нагрузочное тестирование, которое включает в себя проверку производительности web-приложения при различных уровнях нагрузки на web-приложение. После завершения и вывода web-приложения на рынок следует делать регулярное обновление и исправление ошибок, которое повлечёт за собой улучшение функциональности web-приложения. Также при выводе web-приложения на рынок следует обеспечить круглосуточную поддержку пользователей, решение проблем и трудностей, выявляемых с помощью отзывов и предложений, за счёт обратной связи от клиентов, но постоянная поддержка web-приложения также позволит осуществить мониторинг работы для дальнейшего улучшения приложения.

## **1.1 Анализ существующих web-приложений цветочного магазина**

В параграфе 1 одним из приведённых пунктов служит анализ уже существующих решений, который позволит создать ключевые аспекты для создания web-приложения.

Так как в современном мире уже совсем не новым является данное направление именно тщательный анализ уже существующих решений критически важный этап создания web-приложения.

Выделяя основные аспекты web-приложения следует учитывать пользовательский уже имеющийся опыт. Для этого на конкурентном рынке подберём уже существующие и функционирующие способы оформления онлайн заказов цветов и обратим внимания на следующие:

Пользовательский интерфейс web-приложения. Он позволяет оценить простоту работы и использования интерфейса web-приложения.

Пользовательский опыт web-приложения. Он позволит проверить удобство пользования навигацией по web-приложению, скорости загрузки web-приложения.

Функциональные особенности web-приложения. Они позволяют определить, какие функции предлагает web-приложение и насколько они соответствуют ожиданию пользователя.

Производительные особенности web-приложения. Они позволяют выявить скорость загрузки web-приложения, а также его стабильную работу и оптимизацию на всех устройствах.

Мобильная адаптация web-приложения. Она позволяет выявить корректную работу web-приложения на различных устройствах.

На основе приведенных факторов сделаем анализ существующих решений по онлайн рынку города Лесосибирска. Онлайн рынок продажи цветов города Лесосибирск состоит из малого количества магазинов. Из них 2 магазина: магазин цветочный и магазин Feel flora.

Таблица 1 – Анализ цветочных магазинов

<b>Параметр</b>	<b>Цветочный</b>	<b>Feel Flora</b>
Удобство использования	Интуитивная навигация, простота оформления заказа	Простой интерфейс, но сложный процесс оформления заказа
Мобильная адаптация	Адаптивный дизайн для мобильных устройств	Полная мобильная адаптация, плавный UX
Быстрая загрузка	Быстрая загрузка страниц	Быстрая загрузка, оптимизация изображений
Подробные описания товаров с качественными фотографиями	Подробные описания и качественные фотографии	Подробные описания, высокое качество фотографий
Разнообразные способы оплаты и быстрая доставка	Несколько способов оплаты, быстрая доставка	Многочисленные способы оплаты, возможность срочной доставки
Наличие отзывов и рейтингов товаров	Отзывы на сайте, высокие оценки	Отзывы и рейтинги на сайте, клиентская система оценок
Слабые стороны	Ограниченный функционал фильтрации	Сложный процесс оформления заказа, мало уникальных функций
Уникальные функции	Индивидуальная сборка букетов, праздничные упаковки	Праздничные напоминания, бонусы для постоянных клиентов
Современный дизайн	Современный, визуально привлекательный дизайн	Современный, стильный дизайн
Удобство использования	Высокое удобство использования, минимальное количество шагов для оформления заказа	Высокое удобство использования, но сложный процесс оформления заказа
Анализ функций	Широкий ассортимент, фильтры по категориям	Широкий ассортимент, подробные описания
Обратная связь	Онлайн-чат, форма обратной связи	Онлайн-чат, форма обратной связи, обратный звонок

Исходя из таблицы 1 можно заметить, что проанализированные web-приложения цветочных магазинов города Лесосибирска, демонстрируют высокий уровень проработки пользовательского интерфейса, обеспечивают широкий функционал и высокую производительность, а также надежную защиту данных пользователей. Для успешного создания web-приложения следует учесть все приведённые аспекты.

## 1.2 Отличия web-приложения от web-сайта

Начиная разработку web-приложения следует разобраться в отличиях и преимуществах web-приложения и web-сайта. В современном интернет-пространстве, где торговля стремительно развивается, создание web-приложения – это оптимальное решение для многих бизнес-процессов. Проект web-приложения для продажи онлайн цветов не станет исключением.

Web-приложение – это программное обеспечение, которое работает на веб-сервере и доступно пользователям через интернет браузер.

Главными плюсами web-приложения являются: Web-приложение не требует установку, так как оно находится в сети интернет, его не нужно скачивать и устанавливать на устройство, для корректного пользования им, так как доступ к нему осуществляется через любой доступный браузер, на компьютере или телефоне.

Следующим плюсом web-приложения является его доступность. Web-приложение доступно из любой точки мира при наличии доступа к сети интернет.

Одним из главных плюсов web-приложения является низкие затраты на его разработку и обслуживание, так как разработка и поддержка веб-приложений обычно обходится дешевле, чем разработка и поддержка традиционных настольных приложений. Из-за этого web-приложение легко можно обновить, так как обновления веб-приложения происходят на сервере, что не требует от пользователей никаких действий. Также то что web-приложение находится на сервере даёт нам ещё один плюс web-приложение можно легко масштабировать для поддержки большого количества пользователей.

Web-сайт это просто совокупность связанных между собой web-страниц, которые доступны в сети интернет. Web-сайты как правило содержат просто информацию такие как текст, изображения, видео и аудио и чаще всего их называют сайт визитка, который демонстрирует информацию о том и ином

товаре. Web-сайт, как правило, предназначен для предоставления информации или продвижения продуктов и услуг [8].

Web-сайт обеспечивает динамический контент, который предоставляет пользователям доступ к постоянно обновляемой информации. Web-сайт обеспечивает удобный интерфейс для пользователей, который позволяет легко просматривать каталог товаров, но так же стоит отметить, что разработка web-сайта требует значительных финансовых и временных затрат. Включая оплату его доменного имени, хостинга, разработку дизайна, тестирование. Исходя из приведённых выше аргументов web-приложение станет мощным инструментом для онлайн бизнеса по продаже цветов [9].

В параграфе 1 уже были приведены некоторые языки программирования для разработки web-приложения, но объединяет все из них редакторы кода. В таблице 2 представим некоторые параметры сравнения двух кроссплатформенных компиляторов Visual Studio Code и IntelliJ IDEA.

Таблица 2 – Сравнение компиляторов

<b>Критерий</b>	<b>Visual Studio Code</b>	<b>IntelliJ IDEA</b>
Тип	Легкий текстовый редактор с расширениями для IDE	Интегрированная среда разработки (IDE)
Платформа	Windows, macOS, Linux	Windows, macOS, Linux
Цена	Бесплатный	Бесплатная Community Edition, платная Ultimate Edition
Языки программирования	Поддерживает множество языков программирования через расширения	Поддерживает множество языков программирования из коробки
Функции	Подсветка синтаксиса, автозаполнение кода, отладка, Git-интеграция, расширения для выполнения различных задач	Подсветка синтаксиса, автозаполнение кода, рефакторинг, отладка, профилирование, Git-интеграция, расширения для выполнения различных задач

Visual Studio Code – это бесплатный редактор исходного кода, разработанный компанией Microsoft, исходя из таблицы 1 преобладающий над IntelliJ IDEA.

Основным плюсом для создания web-приложения в Visual Studio Code является именно поддержка множества языков программирования, интеграция с различными языками программирования, как отмечено в таблице 2 происходит через расширения [32].

Следующим плюсом для создания в Visual Studio Code для разработки является огромное количество людей, работающих в нём, это обеспечивает большое количество обучающих уроков и решений различных задач.

Так же не маловажно отметить, что в Visual Studio Code есть интегрированный в него терминал, который позволит создавать web-приложение и запускать его сразу через командную строку и видеть все изменения в реальном времени. Это гораздо упростит нам работу с инструментами сборки проекта [32].

Интересное решение сохранения кода с помощью git, Visual Studio Code имеет встроенную в него интеграцию с git что позволяет легко сохранить проект и управлять версиями проекта, выполнять коммиты и слияния веток проекта. Этот немаловажный аспект позволит не потерять проект полностью если во время разработки произойдёт сбой или нарушение [34].

После выбора компилятора стоит задуматься и о других решениях, которые помогут создать качественное web-приложение, одним из таких приложений является Figma.

Figma – это инструмент, который помогает разработчикам создать первичный дизайн интерфейсов и прототипов страниц web-приложения. Figma инструмент, который доступен каждому и широко используется для разработки макетов и удобен для работы в команде разработчиков, так как одним из главных особенностей является совместная работа в реальном времени, несколько дизайнеров могут работать над одним проектом, что может сильно ускорить разработку web-приложения.

Также стоит отметить, что Figma позволяет создавать интерактивные прототипы, которые можно сразу же протестировать и продемонстрировать заказчику проекта и демонстративно показать разработчикам за что отвечает каждый из элементов системы web-приложения.

В Figma можно повторно использовать компоненты и стили это позволит поддерживать консистентность в дизайне и ускорить процесс создания интерфейсов.

Следующим на очереди стоит выбор базы данных для web-приложения, MySQL это система, которая позволит управлять базой данных с открытым кодом, и которая широко используется для хранения управления данными с web-приложения. Главным плюсом MySQL следует отметить оптимизированность, что позволит добиться высокой производительности web-приложения, именно этот показатель доказывает, что MySQL подходит для создания web-приложения [15].

Также для создания web-приложения потребуется локальный сервер для этого можно использовать MAMP, который позволяет разрабатывать и тестировать web-приложения на локальном компьютере. Одним из основных особенностей приложения MAMP является простота его настройки и установки при помощи apache. Также MAMP поддерживает возможность переключения между версиями php, что полезно для тестирования совместимости приложений. MAMP доступен для всех версий MacOS и Windows, что позволит использовать web-приложение на различных операционных системах[7].

### **1.3 Языки программирования для разработки**

В параграфе 1 были обозначены некоторые языки программирования для разработки. Так для разработки комплексной задачи, а именно разработка web-приложения для продажи цветов, нужно использовать различные языки программирования, каждый из которых играет важную роль при создании функционального и привлекательно web-приложения.

Так одним из основных языков программирования для разработчиков является HTML, который прославляет собой фундамент для любой web-страницы, определяя её структуру и содержание[8].

Первым шагом для создания web-страницы является определение элементов этой страницы, такие как заголовки, абзацы, изображения, которые будут основным аргументом в web-приложении по продаже цветов.

Следующим по значимости является язык программирования CSS который отвечает за привлекательность и стиль страницы, созданной при помощи вышеупомянутого языка программирования HTML. CSS помогает определить цвета и шрифты размеры и расположение элементов[6].

Так же CSS помогает сделать web-приложение адаптивным и корректно отображающимся на различных устройствах. CSS позволяет разработчикам распределить на блоки стили оформления страницы и использовать единый стиль оформления для нескольких блоков сразу [16].

Следующий язык программирования является JavaScript который взаимодействует вместе с CSS и добавляет интерактивность к web-странице, делая её наиболее функциональной и с интересным для пользователя дизайном и анимацией [9]. Также JavaScript позволяет обеспечить работу форм, кнопок и других элементов управления, обновление контента без перезагрузки страницы, обработка событий пользователя таких, как например клик мыши или нажатие клавиши [3].

Следующим языком программирования служит серверный язык программирования PHP который обеспечивает работу с базой данных, упомянутой выше, обработку данных из формы, полученной с помощью языка программирования JavaScript и обеспечения логики работы самого web-приложения [12].

#### **1.4 Моделирование архитектуры сайта**

Джим Арлоу и Ила Нойштадт в своей книге утверждают, что Uml диаграммы – это графический язык для визуализации, спецификации, конструирования и документирования компонентов и аспектов программных

систем. UML диаграммы являются ключевым инструментом в объектно-ориентированном анализе и дизайне, обеспечивая стандартизированный способ описания архитектуры и поведения системы.

Uml диаграммы представляют графические части системы, что облегчает понимание и коммутацию среди членов команды разработчиков. В параграфе 1.3 была прокомментирована работы Figma которая облегчает работу дизайнеров web-приложения, uml диаграммы идут от обратного, они показывают, как сама система взаимодействует с пользователем. Она точно описывает требования и функциональность системы. Uml диаграммы создают для точных и подробных и подробных спецификаций, которые используются для реализации системы.

Uml диаграммы служат инструментом для проектирования системы и помогают разработчикам планировать взаимодействие системы между модулями, компонентами и моделировать поведение системы.

Uml диаграммы можно разделить на некоторое количество видов, каждый из которых предназначен для визуализации различных видов систем.

1. Диаграммы классов: показывает структуру системы на уровне классов, включая их атрибуты, методы и отношения между ними.

2. Диаграммы объектов: отображает экземпляры классов и их отношения в конкретный момент времени.

3. Диаграммы прецедентов (Use Case): описывает функциональность системы с точки зрения взаимодействия с пользователями.

4. Диаграммы последовательностей: показывает взаимодействие объектов в хронологическом порядке.

5. Диаграммы коопераций: отображает организацию объектов и их взаимодействие.

6. Диаграммы состояний: моделирует возможные состояния объектов и переходы между этими состояниями.

7. Диаграммы активностей: показывает потоки работ или бизнес-процессы в системе.

8. Диаграммы компонентов: описывает физические компоненты системы и их зависимости.

9. Диаграммы развертывания: моделирует физическое расположение компонентов на аппаратных узлах.

В соответствии с вышеизложенным создадим диаграмму последовательности для цветочного интернет-магазина, с помощью UML. Эта диаграмма представлена на рисунке 2, и она отображает взаимодействие и последовательность выполнения операций в системе.

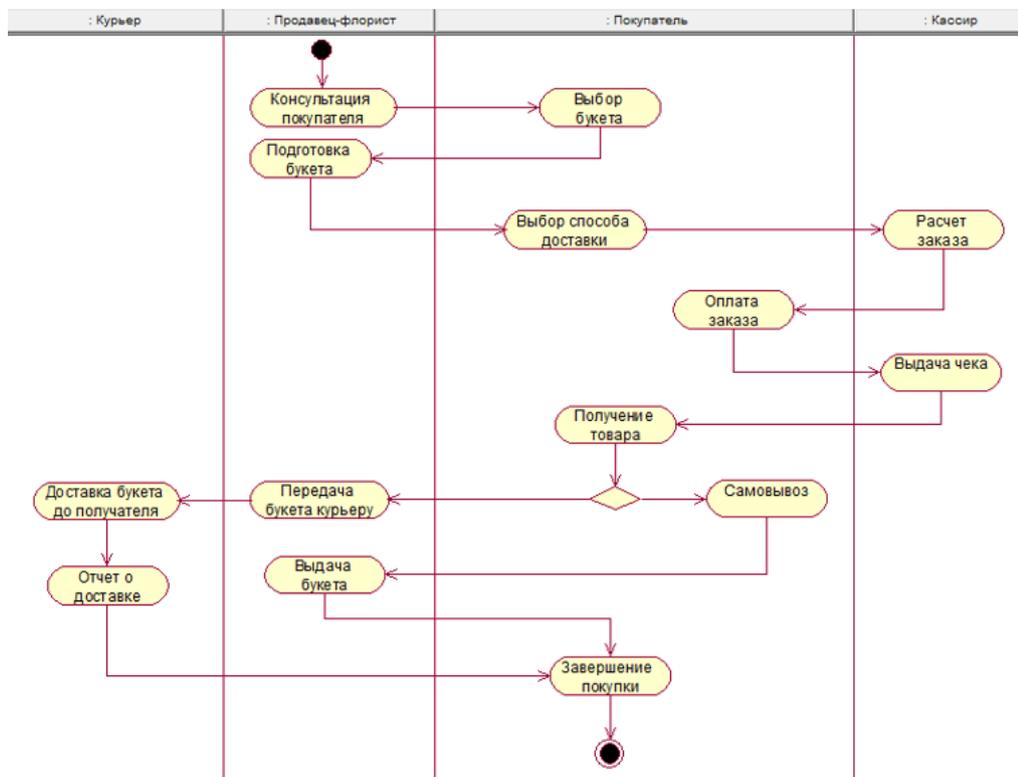


Рисунок 1 – Диаграмма деятельности процесса работы цветочного салона

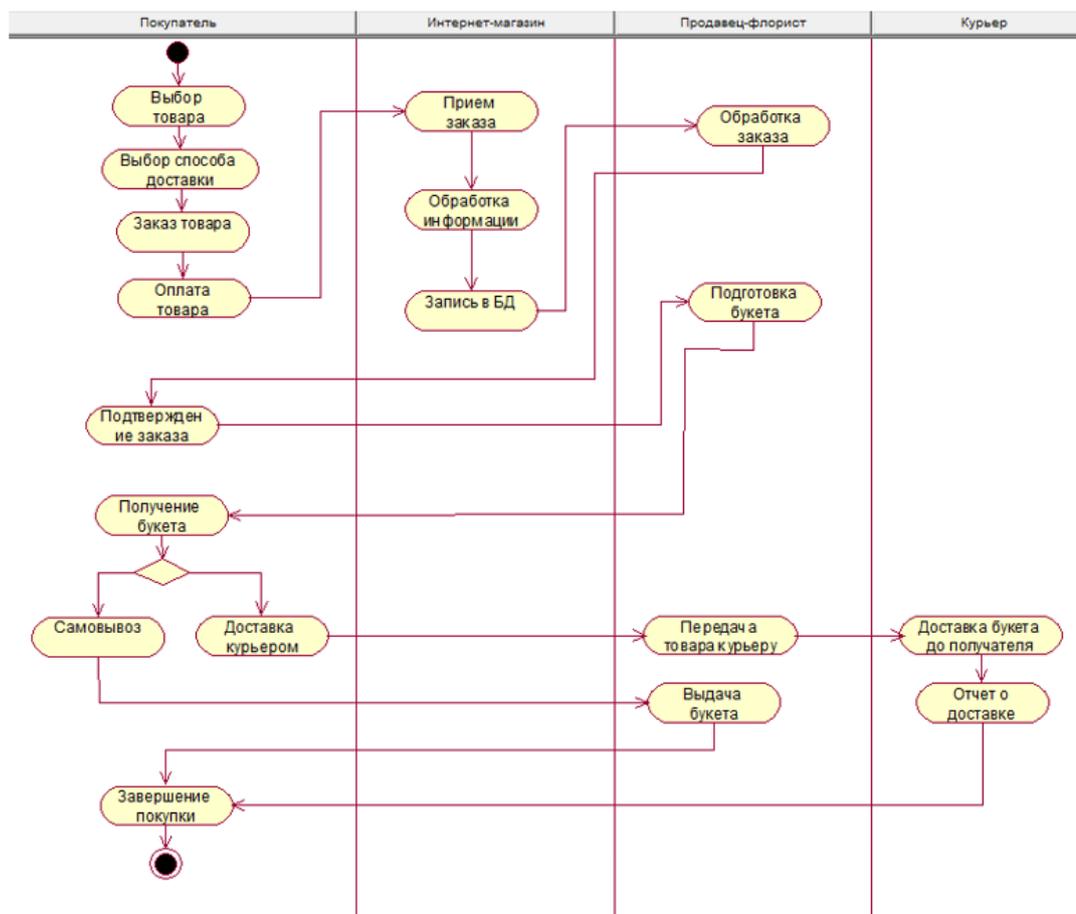


Рисунок 2 – Диаграмма последовательности процесса работы

Проведена постройка UML диаграмм деятельности процесса работы и диаграмма последовательности процесса работы, цветочного магазина для понятия отображения взаимодействия операций в системе.

## **2 Разработка web-приложения для осуществления онлайн заказов**

### **2.1 Проектирование дизайна интерфейса для web-приложения**

Для перехода к дизайну интерфейса обратимся к параграфу 1.1, где был рассмотрен анализ существующих уже решений из него можно выделить элементы, которые играют важную роль для проектирования дизайна интерфейса web-приложения.

Что бы создать актуальный дизайн web-приложения следует учитывать следующие параметры к визуальному оформлению использовать привлекательные цветовые схемы, качественная типографика и элементы дизайна, усиливающие восприятие бренда.

Интеграция лучших практик разрабатываемый макет главной страницы должен сочетает в себе лучшие практики веб-дизайна, такие как:

- использование контрастных цветов и шрифтов для улучшения читаемости.

- применение визуальных иерархий для акцентирования внимания на ключевых элементах.

- создание отзывчивого дизайна, адаптирующегося к различным устройствам.

Для проектирования web-приложения по продаже цветов создадим интерфейс главной страницы, который обеспечивает удобство использования, эстетическую привлекательность и высокую функциональность. Разработанный макет главной страницы является основой для дальнейшего развития и совершенствования приложения.

Особое внимание уделялось обеспечению адаптивного дизайна, чтобы страница корректно отображалась на устройствах с различными экранами. Были оптимизированы изображения и использованы современные технологии для быстрой загрузки страницы. Принципы доступности также учитывались, чтобы сделать страницу удобной для пользователей с ограниченными

ВОЗМОЖНОСТЯМИ.

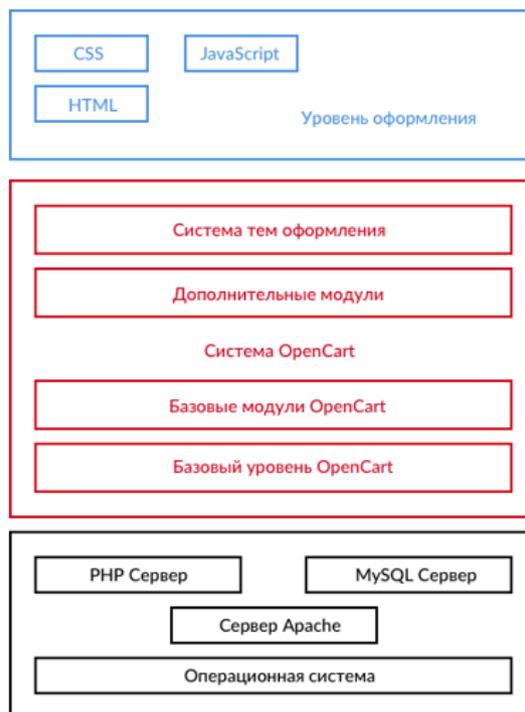


Рисунок 3 – Структура системы управления сайтом

После создания макета главной страницы, необходимо создать дизайн и сделать оформление макета.

Для создания яркой и привлекательной главной страницы следует использовать яркий цвета. В книге «Сила цвета» Морт Уолкер описывает психологию цвета и говорит, что для создания комфортной и не раздражающей обстановки лучше использовать яркие постельные цвета, так, к примеру для создания были выбраны цвета: голубой, белый и ярко-зелёный.

1. Фиолетовый цвет главной страницы символизирует роскошь, креативность и мудрость. Хорош для сайтов, связанных с модой и искусством.

2. Белый цвет часто ассоциируется с чистотой и невинностью. Этот цвет используется в различных культурах для символизации чистоты души, невинности или начала нового этапа.

3. Ярко зелёный цвет ассоциируется с природой, жизнью, ростом и плодородием. Яркий зелёный может усиливать это восприятие, добавляя энергию и динамику.

Рисунок 4 наглядно демонстрирует достигнутый результат.

С помощью грамотного подхода к оформлению удалось создать главную страницу, которая не только эстетически привлекательна, но и обеспечивает удобство использования и эффективность работы веб-приложения.

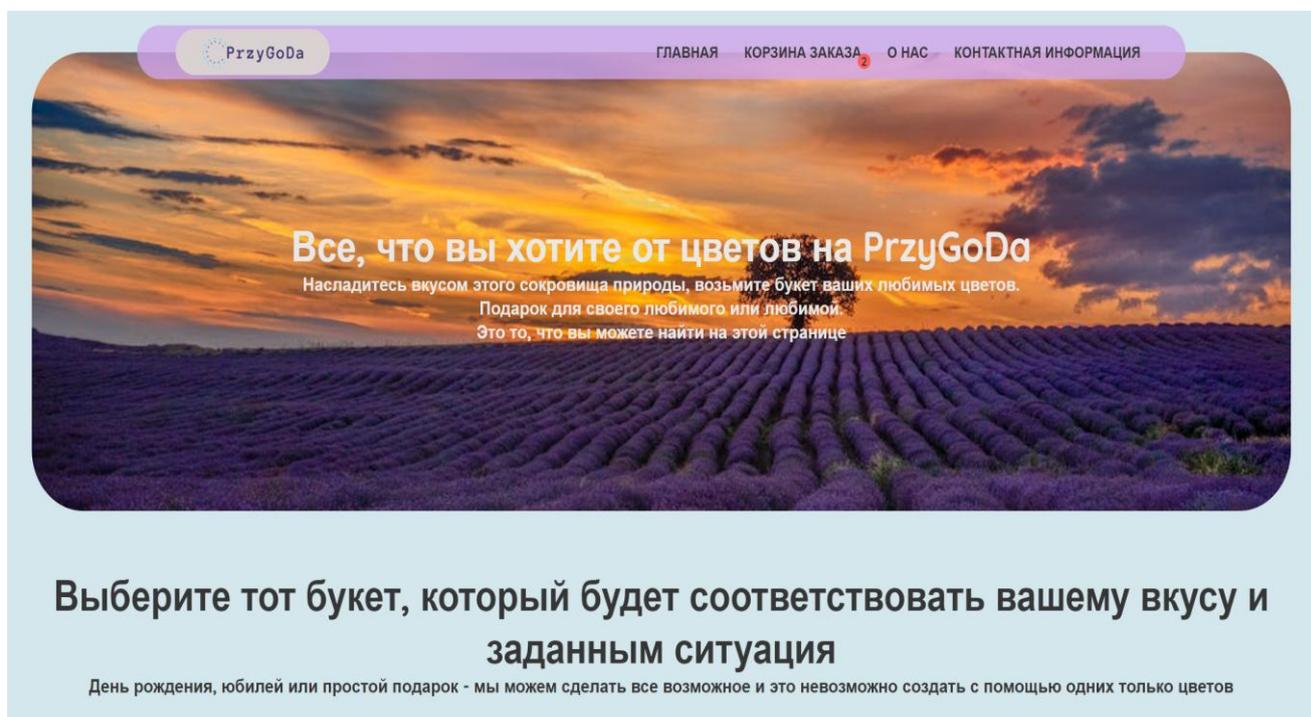


Рисунок 4 – Дизайн главной страницы

Следующей страница, продемонстрированная на рисунке 5, служит «корзина».

Страница должна быть выполнена в том же дизайне что и главная страница сайта, что означает что цветовая палитра остаётся та же.

Функциональные блоки страницы отображают полный заказ клиента

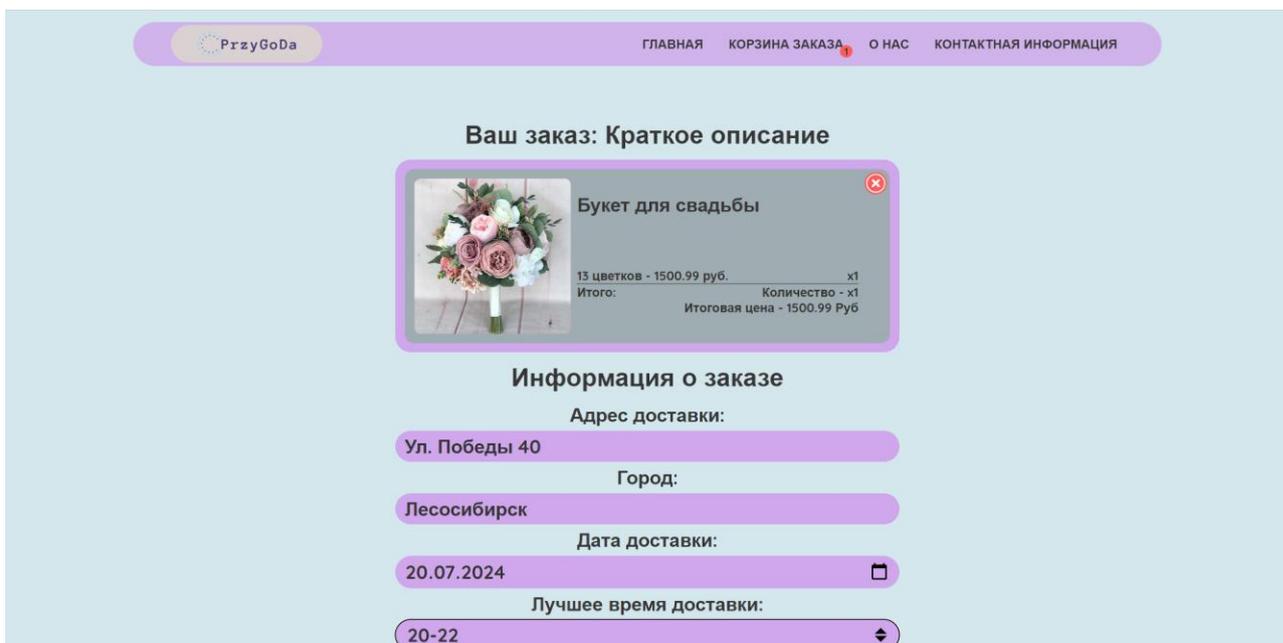


Рисунок 5 – Корзина заказов

Страница «О нас» для web-приложения, занимающегося продажей цветов, создана тщательным подходом, чтобы представить магазин в лучшем свете и вызвать доверие у потенциальных клиентов.

Страница достаточно информативна, привлекательна и легко читаемая.

На этой странице клиент может узнать всю интересующую его информацию о магазине такую как:

- информация о магазине;
- информация о доставке;
- существующие программы лояльности;
- кэшбек.

Приведённые выше пункты странице позволил разделить страницу на логические блоки, чтобы пользователи могли легко найти нужную информацию.

Создание страницы «О нас» для веб-приложения по продаже цветов продемонстрированная на рисунке 6 помогает установить доверие и привлечь внимание потенциальных клиентов. Эффективно спроектированная страница должна быть информативной, привлекательной и структурированной [37].

## О НАШЕЙ КОМПАНИИ

Добро пожаловать в PrzyGoDa, где вы сможете найти изысканные цветочные композиции и искренние подарки. С момента своего основания компания PrzyGoDa, страстно любящая цветы и стремящаяся к исключительному обслуживанию клиентов, распространяет радость и красоту с помощью языка цветов.

В PrzyGoDa мы считаем, что каждое событие заслуживает того, чтобы отпраздновать его с идеальным цветочным оформлением. Будь то если вы ищете яркий букет, чтобы украсить чей-то день, романтическую композицию, чтобы выразить свою любовь, или элегантное украшение для особого мероприятия, наша талантливая команда флористов-дизайнеров посвятит себя созданию потрясающие композиции, которые передадут суть ваших эмоций.

С момента своего основания компания PrzyGoDa установила прочные отношения с местными цветоводами, гарантируя, что мы поставляйте нашим клиентам только самые свежие и качественные цветы. Мы очень гордимся тем, что поддерживаем устойчивые методы и тесное сотрудничество с нашими поставщиками для продвижения экологически чистых и социально ориентированных продуктов. ответственные подходы в цветочной индустрии.

## ИНФОРМАЦИЯ О ДОСТАВКЕ

В PrzyGoDa мы понимаем важность своевременной и надежной доставки, когда речь идет о том, чтобы дарить радость и делать моменты особенными. Мы стремимся к тому, чтобы ваши цветочные композиции и подарки были доставлены с максимальной точностью забота и эффективность, создающие незабываемые впечатления как для отправителей, так и для получателей.

Варианты доставки:

1. Местная доставка: Мы предлагаем услуги местной доставки нашим уважаемым клиентам в пределах указанного радиуса нашего присутствия. цветочный магазин. Наша специализированная команда доставки гордится тем, что выполняет ваши

Рисунок 6 – Созданная страница «О нас»

## 2.2 Разработка базы данных

Создание базы данных для онлайн магазина цветов потребовало тщательного подхода к проектированию структуры данных, чтобы обеспечить её функциональность, эффективность и удобство использования. База данных поддерживает основные функции магазина такие как: управление товарами и заказами, пользователями и их взаимодействиями.

Определение требований – это важный этап в любом проекте, будь то разработка веб-сайта, программного обеспечения или бизнес-процесса.

Цель этого этапа – чётко сформулировать и задокументировать все потребности заинтересованных сторон, чтобы конечный продукт соответствовал их ожиданиям.

Сколько пользователей будет одновременно пользоваться магазином? Количество пользователей – это не статическая величина.

Оно может меняться в зависимости от различных факторов, таких как сезонность, рекламные кампании и изменения в поведении пользователей.

## Прототипирование базы данных для магазина показано на рисунке 7

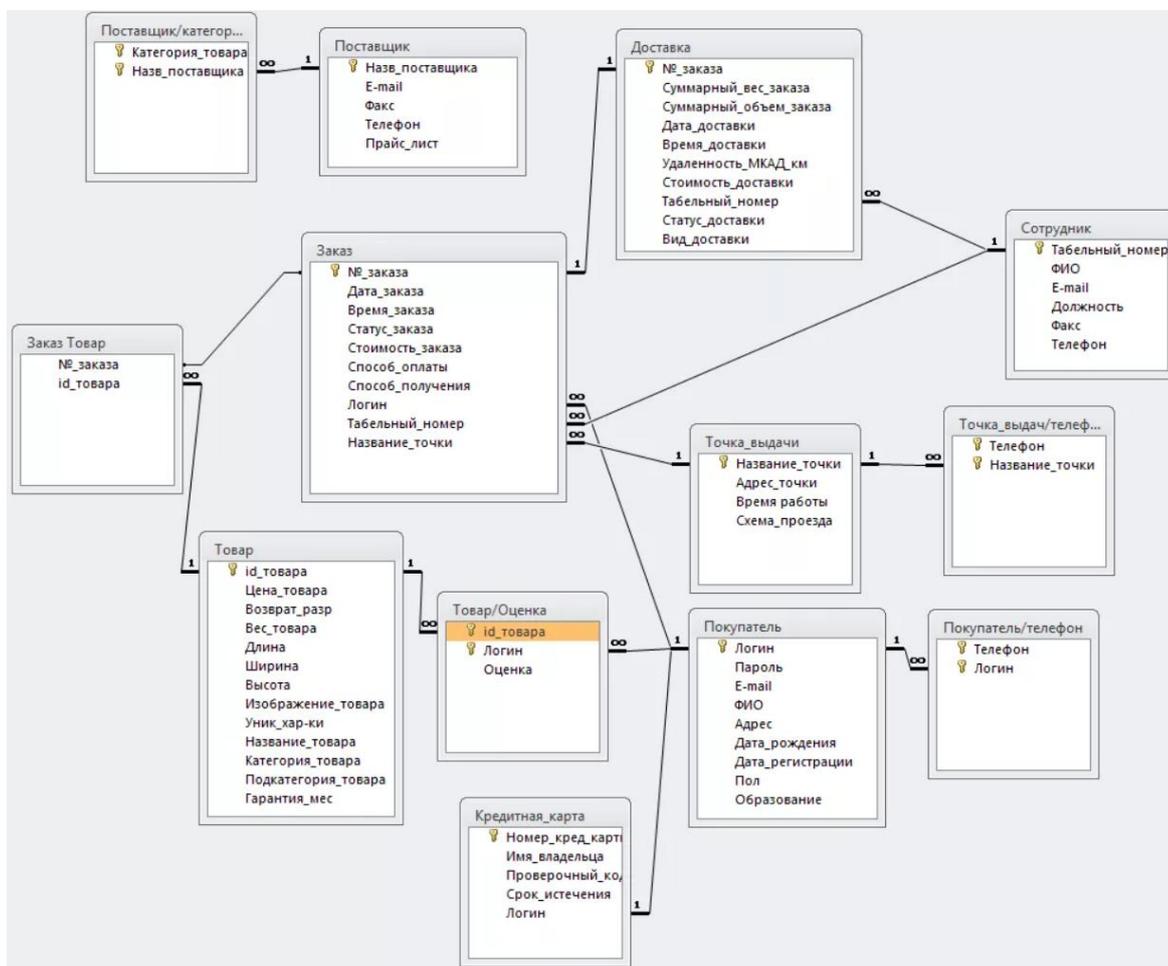


Рисунок 7 – База данных для онлайн магазина

Разработка базы данных для онлайн-магазина – это важный этап, который необходимо было тщательно спланировать, чтобы обеспечить эффективное хранение и управление.

### а) Товары:

- 1) Товарный код: Уникальный идентификатор товара.
- 2) Название товара: Название товара.
- 3) Описание товара: Подробное описание товара.
- 4) Цена товара: Цена товара.

### б) Изображение товара: Фотография товара.

- 1) Категория товара: Категория, к которой относится товар.

- 2) Количество в наличии: Количество товара на складе.
- 3) Вес товара: Вес товара.
- 4) Размеры товара: Размеры товара (длина, ширина, высота).
- 5) Производитель товара: Производитель товара.

в) Категории:

- 1) Код категории: Уникальный идентификатор категории.
- 2) Название категории: Название категории.
- 3) Описание категории: Описание категории.
- 4) Родительская категория: ID категории, которая является родительской для данной категории (для многоуровневой структуры категорий).

г) Заказы:

1. Номер заказа: Уникальный идентификатор заказа.
2. Дата заказа: Дата оформления заказа.
3. Время заказа: Время оформления заказа.
4. Статус заказа: Статус заказа (например, «Новый», «Обрабатывается», «Отправлен», «Выполнен»).
5. Способ оплаты: Способ оплаты заказа (например, «Наличными», «Кредитной картой», «Электронными деньгами»).
6. Способ доставки: Способ доставки заказа (например, «Самовывоз», «Курьерская доставка»).

д) Товары в заказе:

- 1) Номер заказа: Уникальный идентификатор заказа.
- 2) Товарный код: Уникальный идентификатор товара.
- 3) Количество товара: Количество данного товара в заказе.
- 4) Цена товара в заказе: Цена товара в момент оформления заказа.

е) Пользователи:

- 1) ID пользователя: Уникальный идентификатор пользователя.
- 2) ФИО пользователя: ФИО пользователя.
- 3) Email пользователя: Email пользователя.
- 4) Телефон пользователя: Телефон пользователя.

5) Пароль пользователя: Зашифрованный пароль пользователя.

6) Дата регистрации: Дата регистрации пользователя.

7) История заказов: История заказов пользователя.

Обеспечить безопасность базы данных: Используйте надежные пароли, шифрование данных и другие меры безопасности.

Регулярно резервировать базу данных: это позволит вам восстановить данные в случае их потери или повреждения.

Оптимизировать базу данных: это позволит повысить ее производительность.

### **2.3 Анализ целевой аудитории**

Перед разработкой интернет-магазина цветов в Лесосибирске необходимо провести детальный анализ целевой аудитории. Это позволит компании сформировать эффективную стратегию развития, ориентированную на потребности потенциальных клиентов.

Особенности цветочного бизнеса в Лесосибирске:

Высокая стоимость: цветы, особенно на рынке Лесосибирска, стоят дороже из-за своей редкости и уникальности.

Цель анализа состоит в определении целевой аудитории и ее сегментов, а также в изучение потребностей, предпочтений и факторов, влияющих на принятие решения о покупке. Выявление барьеров и скрытых страхов потенциальных клиентов.

Анализ поисковых запросов: Изучение запросов в поисковых системах для понимания потребностей пользователей.

Анализ целевой аудитории позволит онлайн-магазину цветов в Лесосибирске:

Сформировать точное представление о своих потенциальных клиентах.

Разработать маркетинговую стратегию, ориентированную на потребности

целевой аудитории. Повысить эффективность рекламных кампаний. Увеличить продажи и прибыль.

Для более точного определения целевой аудитории необходимо разделить ее на четыре группы представленные в таблице 3, основываясь на критериях рынка.

Таблица 3 – Критерии рынка

<b>Критерий</b>	<b>Описание</b>	<b>Примеры сегментов</b>
Демографический	Возраст, пол, семейное положение, уровень дохода, образование, род занятий	- Молодые профессионалы (25-35 лет) с высоким доходом - Семьи с маленькими детьми Пенсионеры
Географический	Регион проживания, тип населенного пункта, климат	- Жители крупных мегаполисов - Жители небольших городов - Дачники
Психографические	Личностные ценности, образ жизни, интересы, стиль поведения	- Люди, ценящие роскошь и эксклюзивность - Эко-активисты - Романтики
Поведенческий	Поводы для покупки, частота покупок, предпочитаемые каналы покупки, отношение к цене	- Клиенты, покупающие цветы для подарков - Клиенты, заказывающие цветы для оформления мероприятий - Клиенты, регулярно пользующиеся услугами онлайн-заказа

Для изучения целевой аудитории интернет-магазина были проанализированы статистические данные, собранные с 1 февраля по 15 апреля 2023 года.

Посетители: на рисунке 8 представлено процентное соотношение мужчин и женщин, посещающих интернет-магазин.

Заказчики: на рисунке 9 показано соотношение мужчин и женщин, совершивших заказ.

## 2. Возрастная структура аудитории:

Рисунок 10 демонстрирует классификацию посетителей по возрастным категориям.

Анализ диаграмм позволяет понять демографические характеристики посетителей и клиентов интернет-магазина.

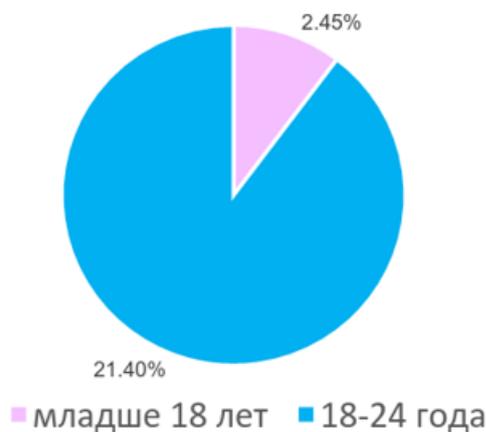


Рисунок 8 – Соотношение мужчин и женщин, посещающих Интернет-магазин

Как видно из диаграммы выше, мужчины и женщины в относительно равной мере интересуются цветочным ассортиментом: доля мужской аудитории составляет всего на 9% больше, чем женской. Однако мужчины совершают заказы в интернет-магазине в 3 раза чаще, чем женщины.

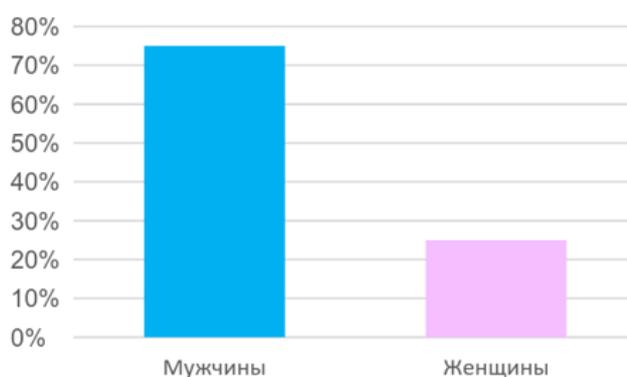


Рисунок 9 – Соотношение мужчин и женщин, совершивших заказ

Исходя из предварительной оценки, можно сделать вывод, что наиболее привлекательной по емкости и экономическому потенциалу является именно

целевая аудитория, состоящая из мужчин.

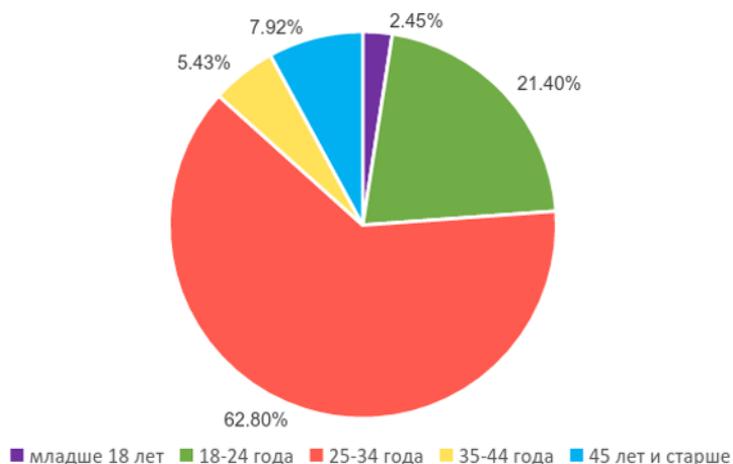


Рисунок 10 – Классификация посетителей по возрастным категориям

Учитывая, что товары интернет-магазина имеют стоимость выше среднего благодаря своей уникальности и отсутствию аналогов в городе Лесосибирск, основными целевыми сегментами будут мужчины старше 18 лет со средним и высоким доходом, которые могут позволить себе предметы роскоши.

Таким образом, анализ позволил определить социально-демографические параметры целевой аудитории:

- пол;
- возраст;
- уровень дохода.

Таблица 4 –Потенциальная аудитория web-приложения

Параметры аудитории	Сегмент аудитории	Примеры
Населенный пункт	Жители г. Лесосибирска	Женщины и мужчины всех возрастов - Люди с разным уровнем дохода Люди, интересующиеся цветами и подарками
	Жители близлежащих населенных пунктов	Женщины и мужчины всех возрастов Люди с разным уровнем дохода Люди, имеющие доступ к интернету и возможность заказать цветы онлайн
Пол	Женщины	Жены, матери, сестры, подруги Женщины, которые хотят сделать подарок себе или другим женщинам Женщины, которые ценят красоту и эстетику
	Мужчины	Мужья, отцы, братья, друзья Мужчины, которые хотят сделать подарок женщине Мужчины, которые хотят произвести впечатление
Возраст	Молодые люди (18-35 лет)	- Студенты Молодые специалисты Молодые семьи
	Взрослые (35-55 лет)	Люди среднего возраста Семьи с детьми Люди, которые ценят практичность и удобство
	Пожилые люди (55+)	- Пенсионеры Люди, которые хотят порадовать своих близких Люди, которые ценят традиции
Уровень дохода	Низкий	Люди, которые ищут бюджетные варианты подарков Люди, которые не могут себе позволить дорогие цветы
	Средний	Люди, которые могут себе позволить купить цветы среднего ценового диапазона Люди, которые хотят сделать красивый и практичный подарок
	Высокий	Люди, которые могут себе позволить купить дорогие цветы Люди, которые хотят сделать эксклюзивный подарок Люди, которые ценят статус
Важные события	День рождения	Рождения близких и друзей Юбилей
	8 марта	Международный женский день Поздравление женщин
	14 февраля	День святого Валентина Поздравление любимых людей
	Свадьба	Свадьба близких и друзей Подарок молодоженам
Мотивация покупки	Сделать подарок	Показать свою любовь и заботу Сделать приятный сюрприз Поздравить с праздником
	Украсить дом	- Создать уютную атмосферу - Поднять настроение - Сделать дом более красивым
	Порадовать себя	- Получить эстетическое удовольствие - Поднять настроение Сделать свою жизнь более яркой

Для успешного развития онлайн-бизнеса по продаже цветов в Лесосибирске необходимо провести тщательный анализ целевой аудитории.

Основная целевая аудитория:

а) Мужчины среднего и высокого достатка:

1) Покупают цветы для женщин в качестве знака внимания.

2) Доминируют среди заказчиков в День Святого Валентина и на 8 марта.

3) Покупают цветы для праздников и торжественных событий.

4) Косвенная целевая аудитория.

б) Женщины:

(хотя и не совершают покупку сами, часто являются инициаторами)

1) Испытывают восторг от цветов.

Сезонность: активность различных целевых аудиторий зависит от времени года.

Дополнительные потенциальные клиенты:

а) Корпоративные заказчики:

1) Дарят цветы своим клиентам, партнерам или сотрудникам.

2) Могут стать важной частью клиентской базы.

Помогают укрепить лояльность и преданность коллег и клиентов.

б) Частные компании:

1) Заинтересованы в цветочном оформлении своих офисов, магазинов или салонов.

Выводы:

а) Портрет конечного клиента:

1) Мужчина с доходом выше среднего.

2) Покупает цветы для женщин.

3) Чувствителен к сезонности.

б) Мотивация к покупке:

1) Знак внимания.

2) Праздники и торжественные события.

- 3) Укрепление лояльности и преданности.
- 4) Эстетическое оформление.
- в) Аспекты для повышения привлекательности:
  - 1) Широкий выбор цветов.
  - 2) Удобный онлайн-заказ.
  - 3) Доставка в день заказа.
  - 4) Акции и скидки.
  - 5) Привлекательное оформление сайта.
  - 6) Позиционирование как надежного партнера.

Знание целевой аудитории позволяет:

- разработать эффективную маркетинговую стратегию.
- сформировать ассортимент товаров и услуг, соответствующий потребностям клиентов.
- создать сайт и другие материалы, ориентированные на целевую аудиторию.
- повысить конверсию и продажи.

Анализ целевой аудитории является важным этапом разработки бизнес-плана онлайн-магазина цветов в Лесосибирске. Полученные данные помогут компании оптимизировать свою работу и добиться успеха на рынке.

## **2.4 Разработка серверной части**

На рисунке 11 представлен внешний вид списка цветов, представленных в web-приложении.

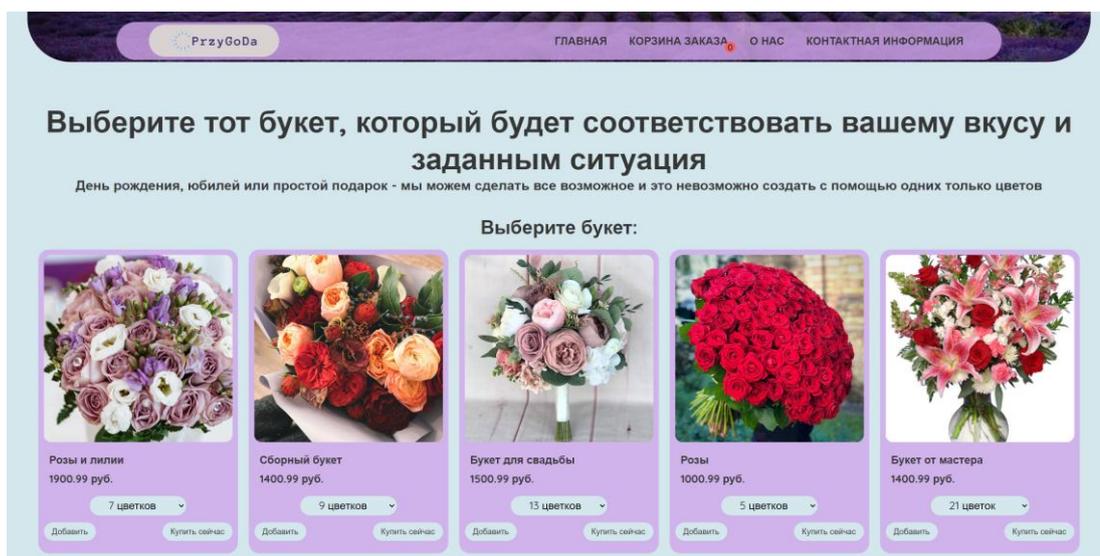


Рисунок 11 – Блоки выбора цветов создаваемые PHP-кодом

Для отображения списка цветов, как показано на рисунке 11, необходимо извлечь данные из базы данных, включая название, фотографию и ID. Для этого был написан PHP-код, который выполняет эту задачу.

Функциональность кода:

Обеспечение возможности навигации по страницам для просмотра большего количества цветов.

Динамическая загрузка и отображение информации из базы данных.

Описание работы кода:

1. Конфигурирование подключения к базе данных:

Указываются параметры для подключения к базе данных MySQL: имя пользователя, пароль, название базы данных, хост и порт.[22]

2. Подключение к базе данных:

Создается соединение с MySQL с помощью функций `mysqli_init()` и `mysqli_real_connect()`.

В случае неудачного подключения выводится сообщение об ошибке, и выполнение кода прекращается.

3. Получение данных из базы данных:

Определяется количество цветов на странице (`limit`).

Из GET-запроса извлекается текущая страница (`page`). Если параметр не

указан, используется значение 1.

Рассчитывается смещение (offset) для выборки данных из базы данных.

Формируется SQL-запрос для выборки данных из таблицы цветов в порядке убывания даты добавления (added\_date) с ограничением по количеству и смещением.

SQL-запрос выполняется с помощью функции `mysqli_query()`.

#### 4. Отображение данных:

Начинается блок отображения цветов с использованием HTML-тега `<div class="thumbnail"></div>`.

В цикле `while` перебираются результаты выборки данных из базы данных.

Отображается блок с изображением, названием и ссылкой на страницу.

#### 5. Кнопки «предыдущая» и «следующая» страница:

Рассчитываются значения предыдущей (`prevPage`) и следующей (`nextPage`) страницы.

Отображаются кнопки для перехода на предыдущую и следующую страницу, если они существуют.

Отображается текущая страница.

Рассчитывается общее количество страниц (`totalPages`) на основе количества элементов на странице.

Отображается блок пагинации с кнопками «предыдущая» и «следующая» страницы (если они доступны) и текущей страницей.

#### 6. Закрытие соединения с базой данных:

Соединение с базой данных закрывается с помощью функции `mysqli_close()`.

Преимущества использования этого подхода:

Удобный интерфейс: Пользователи могут легко просматривать список цветов и переходить по страницам.

Эффективность: Данные из базы данных загружаются динамически, что обеспечивает высокую производительность.

Функциональность: Код позволяет отображать изображения, названия и

ссылки на страницы цветов.

Представленный PHP-код обеспечивает удобный и эффективный способ отображения и навигации по списку цветов на веб-странице, делая пользовательский интерфейс более привлекательным и функциональным.

## ЗАКЛЮЧЕНИЕ

В процессе выполнения выпускной квалификационной работы были тщательно изучены и проанализированы теоретические основы создания и поддержки веб-приложений для онлайн-покупки цветов.

В первой главе рассмотрены существующие решения на рынке города Лесосибирска, что позволило выявить особенности и функциональные возможности для онлайн-заказов цветов. Также были изучены понятие «веб-приложение», выбранные инструменты разработки и области их применения.

Во второй главе было разработано собственное веб-приложение для онлайн-заказов цветов. Спроектирован дизайн интерфейса, учитывающий требования пользователей и современные тенденции в веб-дизайне. Создана база данных для хранения и управления информацией о доступных цветах, количестве заказов и данных клиентов. Разработаны серверная и клиентская части приложения, что позволило реализовать его функционал и взаимодействие с базой данных.

Результатом работы стало полноценное веб-приложение для онлайн-покупки цветов, предоставляющее удобный интерфейс для пользователей. Оно позволяет осуществлять онлайн-заказ, а также хранить данные о клиентах, частоте заказов, комментариях и отзывах клиентов, что способствует улучшению деятельности онлайн-магазина.

Разработанное веб-приложение соответствует современным требованиям в данной сфере и может успешно применяться в реальных условиях.

Таким образом, выполнение данной дипломной работы позволило углубить знания о создании и поддержке веб-приложений для онлайн-покупки цветов, а также приобрести практические навыки разработки, проектирования и реализации. Разработанное приложение может служить основой для дальнейших исследований и разработок в этой области.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Аквино, К. Frontend. Клиентская разработка для профессионалов / К.Аквино. – СПб: Питер, 2017. – 512 с.
2. Андерсон, С. Приманка для пользователей.  
Создаем привлекательный сайт / С. Андерсон. – М.: Питер, 2021. – 793 с.
3. Браун, Э. Изучаем JavaScript. Руководство по созданию современных веб-сайтов / Э. Браун. – Москва: Альфа-книга, 2017. – 368 с.
4. Веллинг, Л. В. Разработка веб-приложений с помощью PHP и MySQL /Л. В. Веллинг, Л. Т. Томпсон. – Москва: Вильямс, 2020. – 768 с.
5. Вора, П. В. Шаблоны проектирования веб-приложений / П. В. Вора. – Москва: Эксмо, 2011. – 870 с.
6. Губин, М. С. PHP 8. Новинки языка и программы для работы с ним / М. С. Губин, Д. Котеров. – Екатеринбург: Издательские Решения, 2020. – 19 с. – ISBN 9785005138330.
7. Дакетт, Д. Основы веб-программирования с использованием HTML / Джон Дакетт. – Москва: Эксмо, 2020. – 239 с.
8. Дакетт, Д. Разработка и дизайн веб-сайтов / Джон Дакетт. – Москва: Эксмо, 2018. – 250 с.
9. Дронов, В.А. HTML5, CSS3 и web2.0. Разработка современных Web-сайтов / В.А. Дронов. – Спб: БХВ-Петербург, 2011. – 416 с.
10. Емелин, Н.В. Способы оценки и анализа конкурентоспособности / Н. В. Емелин. – М: Форсайт, 2018. – 355 с.
11. Ешану, А. А. Отличия веб-приложения от веб-сайта / А. А. Ешану // Современное программирование. – Нижневартовск: НВГУ, 2020. – С. 41-43.
12. Журавлев, А. Е. Корпоративные информационные системы. Администрирование сетевого домена. Учебное пособие для СПО / А. Е. Журавлев, А. В. Макшанов, Л. Н. Тындыкарь. – Санкт-Петербург: Лань, 2021.– 172 с. – ISBN 978-5-8114-8417-1.

13. Ильин, И. В. Базы данных: учебное пособие / И. В. Ильин, О. Ю. Ильяшенко. – Санкт-Петербург: СПбГПУ, 2020. – 96 с.
14. Карпов, Александр. Создание и продвижение сайтов. Непрофессионал для Непрофессионала / Александр Карпов, Джи Ким. – Москва: 2016. – 280 с.
15. Кевин, Янк. PHP и MySQL. От новичка к профессионалу / Янк Кевин. – Москва.: Эксмо, 2018. – 384 с.
16. Кириченко, А. В. Web на практике. CSS, HTML, JavaScript, MySQL, PHP для fullstack-разработчиков / А. В. Кириченко, А. П. Никольский, Е. В. Дубовик. – Санкт-Петербург: Наука и техника, 2021. – 432 с. – ISBN 978-5-94387-271-6.
17. Климов, А. JavaScript на примерах / А. Климов. – СПб: БХВ-Петербург: Эксмо, 2018. – 336 с.
18. Котеров, Дмитрий. PHP 8 / Дмитрий Котеров, Игорь Симдянов. – Москва: БХВ, 2023. – 992 с. – ISBN 978-5-9775-1692-1.
19. Коэн, Исси. Полный справочник по HTML, CSS и JavaScript / Исси Коэн. – Паблшерз: Эксмо, 2017. – 246 с.
20. Локхарт, Джош. Современный PHP. Новые возможности и передовой опыт / Джош Локхарт. – М.: ДМК Пресс, 2016. – 304 с.
21. Лоре, А. Проектирование веб-API / А. Лоре. – Москва: ДМК-Пресс, 2020. – 440 с. – ISBN 978-5-97060-861-6.
22. Мехта, Ч. MySQL 8 для больших данных / Ч. Мехта. – Москва: ДМК Пресс, 2017. – 228 с.
23. Никсон, Р. Создаем динамические веб-сайты с помощью PHP, MySQL и JavaScript / Р. Никсон. – СПб: Питер, 2010. – 768 с.
24. Прохоренок, Н. А. HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера, 5 изд / Н.А. Прохоренок. – СПб.: БХВ-Петербург, 2019. – 912 с.
25. Пьюривал, Сэмми Основы разработки веб-приложений / Сэмми Пьюривал. – Москва: Питер, 2015. – 272 с.
26. Роббинс, Д. Н. Веб-дизайн для начинающих. HTML, CSS, JavaScript и

веб-графика подробнее: <https://www.labyrinth.ru/books/797578/> / Д. Н. Роббинс. – Санкт-Петербург: BHV, 2021. – 956 с.

27. Робсон, Элизабет. Изучаем HTML, XHTML и CSS / Элизабет Робсон, Эрик Фримен. – Питер: -, 2019. – 720 с.

28. Сафронов, М. Разработка веб-приложений с помощью Yii 2 и PHP / М.Сафронов. – Москва: ДМК Пресс, 2015. – 377 с.

29. Ткаченко, В. MySQL по максимуму / В. Ткаченко, П. Зайцев, Ш. Бэрн. – М.: Питер, 2018. – 864 с.

30. Фримен, Э. Изучаем программирование на JavaScript / Э. Фримен, Э. Робсон. – Москва: Эксмо, 2007. – 2010 с.

31. Хоффман, Э. Безопасность веб-приложений. Разведка, защита, нападение / Э. Хоффман. – Санкт-Петербург: Питер, 2020. – 336 с. – ISBN 9785-4461-1786-4.

32. Хэррон, Д. Node.js Разработка серверных веб-приложений на JavaScript / Д. Хэррон. – Москва: ДМК, 2014. – 144 с.

33. Эспозито, Д. Разработка современных веб-приложений. Анализ предметных областей и технологий / Д. Эспозито – Москва: Вильямс, 2017. – 464 с. – ISBN 978-5-9908910-3-6.

34. GitHub//phpmyadmin: сайт. – URL: <https://github.com/phpmyadmin/phpmyadmin/wiki> (дата обращения: 15.05.2024)

35. HTML5BOOK: сайт. – URL: <https://html5book.ru/> (дата обращения: 03.04.2024)

36. PHP: официальный сайт. – URL: <https://www.php.net/> (дата обращения: 15.06.2024)

37. w3schools//PHP tutorial: сайт. – URL: <https://www.w3schools.com/php/default.asp> (дата обращения: 08.05.2024)

38. w3schools//CSS Tutorial: сайт. – URL: <https://www.w3schools.com/css/default.asp> (дата обращения: 22.04.2024)

39. w3schools//HTML Tutorial: сайт. – URL: <https://www.w3schools.com/html/default.asp> (дата обращения: 22.04.2024)

40. w3schools//JavaScript Tutorial: сайт. – URL:

<https://www.w3schools.com/js/default.asp> (дата обращения: 27.04.2024)

41. w3schools//MySQL Tutorial: сайт. – URL:

<https://www.w3schools.com/mysql/default.asp> (дата обращения: 15.05.2024)

## ПРИЛОЖЕНИЕ А

### Компонент Flower Catalog

```
import first from '../img/1.jpeg'
import second from '../img/2.jpeg'
import third from '../img/3.jpg'
import fourth from '../img/4.jpg'
import fiveth from '../img/5.jpg'
const flowers = [
  {
    id: 0,
    img_src: first,
    name: "Розы и лилии",
    prise: {
      option_0: 1900.99,
      option_1: 2100.99,
      option_2: 2300.99,
    },
    options: {
      option_0: "7 цветков",
      option_1: "11 цветков",
      option_2: "15 цветков",
    },
  },
  {
    id: 1,
    img_src: second,
    name: "Сборный букет",
    prise: {
      option_0: 1400.99,
```

```
    option_1: 1800.99,  
    option_2: 2300.99,  
  },  
  options: {  
    option_0: "9 цветков",  
    option_1: "13 цветков",  
    option_2: "17 цветков",  
  },  
},  
{  
  id: 2,  
  img_src: third,  
  name: "Букет для свадьбы",  
  prise: {  
    option_0: 1500.99,  
    option_1: 1600.99,  
    option_2: 1900.99,  
  },  
  options: {  
    option_0: "13 цветков",  
    option_1: "15 цветков",  
    option_2: "17 цветков",  
  },  
},  
{  
  id: 3,  
  img_src: fourth,  
  name: "Розы",  
  prise: {  
    option_0: 1000.99,
```

```
    option_1: 1600.99,  
    option_2: 2200.99,  
    option_3: 2800.99,  
    option_4: 3400.99,  
  },  
  options: {  
    option_0: "5 цветков",  
    option_1: "9 цветков",  
    option_2: "13 цветков",  
    option_3: "17 цветков",  
    option_4: "21 цветок",  
  },  
},  
{  
  id: 4,  
  img_src: fiveth,  
  name: "Букет от мастера",  
  prise: {  
    option_0: 1400.99,  
    option_1: 2000.99,  
    option_2: 2500.99,  
    option_3: 3000.99,  
  },  
  options: {  
    option_0: "21 цветок",  
    option_1: "25 цветков",  
    option_2: "29 цветков",  
    option_3: "33 цветка",  
  },  
}, export default flowers;
```

## ПРИЛОЖЕНИЕ Б

### Компонент React Flower Cart Address Delivery

```
import React, { useContext } from "react";
import { CartItemsContext } from "../../../../../context/CartContext";
import { STATEPROPS } from "../../../../../reducers/formReducer";

export const FlowerCartAddressDelivery = () => {
  const { deliveryInfoReducer, radioReferanse, deliveryInfoDispatch, radioID }
= useContext(CartItemsContext);

  const handleRadioChange = (e) => {
    deliveryInfoDispatch({
      type: STATEPROPS.RADIO_CHANGED,
      radioChange: e.target.value,
    });
  };

  return (
    <form
      id="Flowers__Cart__DeliveryInfo__DeliveryTypeRadio"
      defaultValue={deliveryInfoReducer.typeOfOrder}
      onChange={handleRadioChange}
      ref={radioReferanse}
    >
      <label>
        <input type="radio" name="typeOfOrder" defaultValue="Доставка"
id={`\${radioID}001`} />
        Доставка
      </label>
      <label>
```

```
        <input type="radio" name="typeOfOrder" defaultValue="Самовывоз"  
id={`${radioID}002`} />  
        Самовывоз  
    </label>  
</form>  
);  
};
```

## ПРИЛОЖЕНИЕ В

### Компонент React Flower Cart Alert Message

```
import React, { useContext } from "react";
import { Link } from "react-router-dom";
import { CartItemsContext } from "../../context/CartContext";
export const FlowerCartAlertMessage = () => {
  const { deliveryInfoReducer, cartRender } = useContext(CartItemsContext);
  let fillPoleProver;
  fillPoleProver = Object.entries(deliveryInfoReducer).some((item) => {
    return item[1] === "";
  });
  const AlarmText = () => {
    if (!cartRender.length && fillPoleProver) {
      return (
        <h3>
          Ваша корзина пуста, и вы не заполнили все поля для информации о
          заказе. Пожалуйста, закажите что-нибудь в своей корзине и
          заполните все необходимые поля
        </h3>
      );
    } else if (cartRender.length > 0 && fillPoleProver) {
      return <h3>Вы заполнили не все поля для информации о заказе.
        Пожалуйста, заполните все необходимые поля</h3>;
    } else if (!cartRender.length && !fillPoleProver) {
      return <h3>Ваша корзина пуста. Пожалуйста, закажите что-
        нибудь</h3>;
    }
  };
};
```

```
return (  
  <section id="alarm_message">  
    <h2>Предупреждение</h2>  
    <AlarmText />  
    {!cartRender.length && (  
      <button>  
        <Link to="/Flowers-shop-PrzyGoDa">Главная страница</Link>  
      </button>  
    )}  
  </section>  
);  
};
```

## ПРИЛОЖЕНИЕ Г

### Компонент React Flower Cart Sum Info

```
import React, { useContext } from "react";
import { CartItemsContext } from "../../context/CartContext";
import { CART_RENDER_ACTION_TYPE } from
"../../reducers/cartOrderRender";
export const FlowerCartSumInfo = () => {
  const {
    alarmProver,
    cartValue,
    dispatchRender,
    setCartValue,
    setSmrOpenTrue,
    cartRender,
    deliveryInfoReduser,
    setAlarmProver,
  } = useContext(CartItemsContext);
  let fillPoleProver;

  fillPoleProver = Object.entries(deliveryInfoReduser).some((item) => {
    return item[1] === "";
  });
  const timeOpenComponent = () => {
    if (!alarmProver) {
      setAlarmProver((prev) => !prev);
      setTimeout(() => setAlarmProver((prev) => !prev), 5000);
    }
  };
};
```

```

const toggleButtonConfirmingOrder = () => {
    !cartRender.length || fillPoleProver ? timeOpenComponent() :
setSmrOpenTrue((prev) => !prev);
};

const toggleButtonOrderToZero = () => {
    setCartValue({ cartItem: 0, cartCost: 0 });
    dispatchRender({
        type: CART_RENDER_ACTION_TYPE.ZEROING_CART,
    });
};

return (
    <div className="Flowers__Cart__SumInfo">
        <article>{ cartValue.cartItem === 0 ? `Вы ничего не добавили в корзину`
: `Количество: ${cartValue.cartItem}` }</article>
        <article>Итоговая стоимость:
{ cartValue.cartCost.toFixed(2)}Руб</article>
        <section className="Flowers__Cart__SumInfo--ButtonContainer">
            <button onClick={toggleButtonConfirmingOrder}>Подтвердите
заказ</button>
            <button onClick={toggleButtonOrderToZero}>Отменить заказ</button>
        </section>
    </div>
);
};

```

## ПРИЛОЖЕНИЕ Д

### Компонент Flower Cart Summary в React

```
import React, { useContext } from "react";
import flowers from "../../flowerslist/flowersList";
import { CART_RENDER_ACTION_TYPE } from
"../../reducers/cartOrderRender";
import { CartItemsContext } from "../../context/CartContext";
export const FlowerCartSummary = () => {
  const { cartValue, setCartValue, cartRender, dispatchRender } =
useContext(CartItemsContext);
  return cartRender.map((item) => {
    const quantitySrt = item.quantity;
    const itemSearchSrt = flowers.find((it) => it.id === item.id);
    const nmFlowersSrt = Object.keys(quantitySrt || {});
    let nmbItemsCount = 0;
    let nmbItemCost = 0;
    Object.entries(quantitySrt).forEach((el) => {
      nmbItemsCount += el[1];
      nmbItemCost += el[1] * flowers.find((el) => el.id ===
item.id)?.price[`option_${el[0]}`];
    });
    const onToggleRemoveButton = () => {
      let nmbDeletingItems = 0;
      let nmbDecrisingCost = 0;
      Object.entries(quantitySrt).forEach((el) => {
        nmbDeletingItems += el[1];
        nmbDecrisingCost += el[1] * flowers.find((el) => el.id ===
item.id)?.price[`option_${el[0]}`];
      });
    };
  });
}
```

```

    });
    setCartValue({
      cartItem: cartValue.cartItem - nmbDeletingItems,
      cartCost: cartValue.cartCost - nmbDecrisingCost,
    });
    dispatchRender({
      type: CART_RENDER_ACTION_TYPE.REMOVE_POSITION,
      id: item.id,
    });
  };
  return (
    <section key={item.id} className="Flowers__Cart--
Summary__PositionContainer">
      <img src={itemSearchSrt?.img_src} alt="" />
      <article>
        <span>
          <div>{itemSearchSrt?.name}</div>
        </span>
        <span>
          {nmFlowersSrt.length === 0
            ? null
            : nmFlowersSrt.map((element) => {
              if (quantitySrt[element] === 0) {
                return null;
              } else {
                return (
                  <div key={element} className="Flowers__Cart--
Summary__PositionContainer--Info">
                    <span key={itemSearchSrt?.options[`${element}`]}>
                      {itemSearchSrt?.options[`${element}`]}

```

```

        {" - "}
        {itemSearchSrt?.prise[option_`${element}`]} руб.
    </span>
    <span>x{quantitySrt[element]}</span>
</div>
);
}
}}
<div className="Flowers__Cart--Summary__PositionContainer--
Total">
    <div>
        <span>Итого:</span>
        <span>Количество - x{nmbItemsCount}</span>
    </div>
    <div>Итоговая цена - {nmbItemCost.toFixed(2)} Руб </div>
</div>
</span>
</article>
    <div className="Flowers__Cart--Summary__PositionContainer--
CloseButton" onClick={onToggleRemoveButton}></div>
</section>
);
});
};

```

## ПРИЛОЖЕНИЕ E

### Компонент Confirm Page в React

```
import React, { useState } from "react";
import { CartItemsContext } from "../../context/CartContext";
import { useContext } from "react";
import ConfirmPageOrder from "./ConfirmPageOrder";
export const ConfirmPage = () => {
  const { setSmrOpenTrue, cartValue, cartRender, scrollTopProve,
scrollDownProve, confirmPageRef, dialogPageRef } =
  useContext(CartItemsContext);
  const [isClosing, setIsClosing] = useState(false);
  const toggleButtonClose = () => {
    const refShrt = confirmPageRef.current;
    if (isClosing === false) {
      refShrt.classList.add("closing");
      setIsClosing(true);
      refShrt.addEventListener(
        "animationend",
        () => {
          refShrt.classList.remove("closing");
          setIsClosing(false);
          setSmrOpenTrue((prev) => !prev);
        },
        { once: true }
      );
    }
  };
  const toggleButtonAsseptOrder = () => {
```

```

const refShrt = dialogPageRef.current;
refShrt.showModal();
setSmrOpenTrue((prev) => !prev);
};
return (
<section className="flower-assept" ref={confirmPageRef}>
  <div
    className="flower-assept__container"
    data-length_proove={cartRender.length > 1 ? true : false}
    data-ontop={scrollTopProve}
    data-onbottom={scrollDownProve}
  >
    <h2>Страница подтверждения заказа</h2>
    <ConfirmPageOrder />
    <div>`Цена заказа: ${cartValue.cartCost.toFixed(2)} `</div>
    <div className="flower-assept__container--buttonHolder">
      <button onClick={toggleButtonAsseptOrder}>Подтвердить
заказ</button>
    </div>
    <div className="closeButton" onClick={toggleButtonClose}></div>
  </div>
</section>
);
};

```

## ПРИЛОЖЕНИЕ Ж

### Компонент Confirm Page Order в React

```
import React from "react";
import { CartItemsContext } from "../../context/CartContext";
import flowers from "../../flowerslist/flowersList";
import { useContext } from "react";
const ConfirmPageOrder = () => {
  const { cartRender, setScrollTopProve, setScrollDownProve } =
useContext(CartItemsContext);
  const toggleOrderScroll = (e) => {
    const htmlFontSize = document.querySelector("html");
    const bottomOfScroll =
      parseInt(window.getComputedStyle(htmlFontSize,
null).getPropertyValue("font-size")) *
      20 *
      (cartRender.length - 1);
    e.target.scrollTop <= 20 ? setScrollTopProve(true) :
setScrollTopProve(false);
    e.target.scrollTop >= bottomOfScroll ? setScrollDownProve(true) :
setScrollDownProve(false);
  };
  return (
    <section className="flower-assept__container--order"
onScroll={toggleOrderScroll}>
      {cartRender.map((item) => {
        const quantitySrt = item.quantity;
        const itemSearchSrt = flowers.find((it) => it.id === item.id);
        const nmFlowersSrt = Object.keys(quantitySrt || {});
        let nmbDeletingItems = 0;
```

```

let nmbDecrisingCost = 0;
Object.entries(quantitySrt).forEach((el) => {
  nmbDeletingItems += el[1];
  nmbDecrisingCost += el[1] * flowers.find((el) => el.id ===
item.id)?.prise[`${option}_${el[0]}`];
});
return (
  <section key={item.id} className="flower-assept__container--
order__PositionContainer">
    <img src={itemSearchSrt?.img_src} alt="" />
    <article>
      <h3>{itemSearchSrt?.name}</h3>
      <article>
        {nmFlowersSrt.length === 0
? null
: nmFlowersSrt.map((element) => {
  if (quantitySrt[element] === 0) {
    return null;
  } else {
    return (
      <div key={element} className="flower-assept__container--
order__PositionContainer--Info">
        <span key={itemSearchSrt?.options[`${option}_${element}`]}>
          {itemSearchSrt?.options[`${option}_${element}`]}
          {" - "}
          {itemSearchSrt?.prise[`${option}_${element}`]}$
        </span>
        <span>x{quantitySrt[element]}</span>
      </div>
    );
  }
});

```

```

        }
    }}}

    <div className="flower-assept__container--
order__PositionContainer--Total">
        <h3>Итого:</h3>
        <span>Букеты - x{nmbDeletingItems}</span>
        <div>Итоговая цена - {nmbDecrisingCost.toFixed(2)}    Руб
</div>

    </div>
</article>
</article>
</section>
);
}}
</section>
);
};

```