|  |  |
| --- | --- |
| УТВЕРЖДАЮ | УТВЕРЖДАЮ |
| Компания-исполнитель | Компания-заказчик |
| Руководитель | Руководитель |
| *(ФИО)* | *(ФИО)* |
| «7» мая 2021 г. | «7» мая 2021 г. |

ТЕХНИЧЕСКОЕ ЗАДАНИЕ

на разработку сайта интернет-магазина

для компании «Наименование компании»

|  |  |
| --- | --- |
| СОГЛАСОВАНО | СОГЛАСОВАНО |
| Представитель компании-исполнителя | Представитель компании-заказчика |
| *(должность)* | *(должность)* |
| *(ФИО)* | *(ФИО)* |
| «7» мая 2021 г. | «7» мая 2021 г. |

Минск, 2021

Содержание

[1 Общие сведения 2](#_Toc449717286)

[2 Назначение системы 3](#_Toc449717294)

[3 Требования к системе 4](#_Toc449717300)

[3.1 Функциональные требования 4](#_Toc449717301)

[3.1.1 Описание пользователей системы 4](#_Toc449717302)

[3.1.2 Описание вариантов использования 5](#_Toc449717303)

[3.2 Технические требования 1](#_Toc449717316)2

[3.2.1 Архитектура системы 1](#_Toc449717317)2

[3.2.2 Требования к пользовательскому интерфейсу 1](#_Toc449717318)4

[3.2.3 Требования к безопасности 1](#_Toc449717319)6

[3.2.4 Требования к надежности 1](#_Toc449717319)6

[3.2.5 Требования к языку программирования 1](#_Toc449717319)6

[4 Порядок приема системы 1](#_Toc449717331)7

[4.1 Приемочные тесты 17](#_Toc449717332)

[4.2 Сроки исполнения 1](#_Toc449717333)9

**1 Общие сведения**

Настоящий документ, (далее по тексту - Техническое задание) является приложением к Договору № от 07.05.2021 г.

В Техническом задании приведен полный набор требований к разработке сайта Заказчика (далее по тексту – система) и выполнения смежных работ, связанных с разработкой.

Исполнитель подготовил Техническое задание, которое содержит перечень требований к выполняемым исполнителем работам.

В процессе согласования изменений в Техническом задании могут быть разработаны дополнительные требования, которые прикрепляются к Договору.

По окончании работ Исполнитель обязан предоставить полностью функционирующий сайт и все необходимые данные для доступа к системе управления сайтом (аккаунты, пароли, адреса серверов и т.п.).

2 Назначение системы

Система «Интернет-магазин» представлена в виде многопользовательского веб-приложения – официального представительства компании-заказчика в сети Интернет.

Сайт предназначен для обеспечения компании Заказчика возможностью сбыта продукции посредством сети Интернет, доступом клиента к опубликованной на сайте информации, возможностью отслеживания информации о наличии конкретных товаров, добавления новых Заказчиком.

3 Требования к системе

**3.1 Функциональные требования**

**3.1.1 Описание пользователей системы**

Пользователь системы обладает функциональностью для просмотра товаров и оформления заказов.

Таблица 1 - Описание пользователей системы

|  |  |  |
| --- | --- | --- |
| № | Актер | Описание |
| 1 | Администратор | Пользователь системы, который может редактировать каталог товаров (добавлять товары, изменять наличие товаров), добавлять клиентов-неплательщиков в «черный список». |
| 2 | Клиент | Основной пользователь системы, имеет возможность просматривать каталог товаров, добавлять товары в корзину, оформлять заказ. |
| 3 | Неплательщик | Пользователь системы, находящийся в «черном списке», ограничен в действиях, может только просматривать каталог и оплачивать задолженность. |

**3.1.2 Описание вариантов использования**

В приложении представлены три различных актера, имеющие как схожие виды активности (авторизация, просмотр каталога), так и характерные для каждого актера.

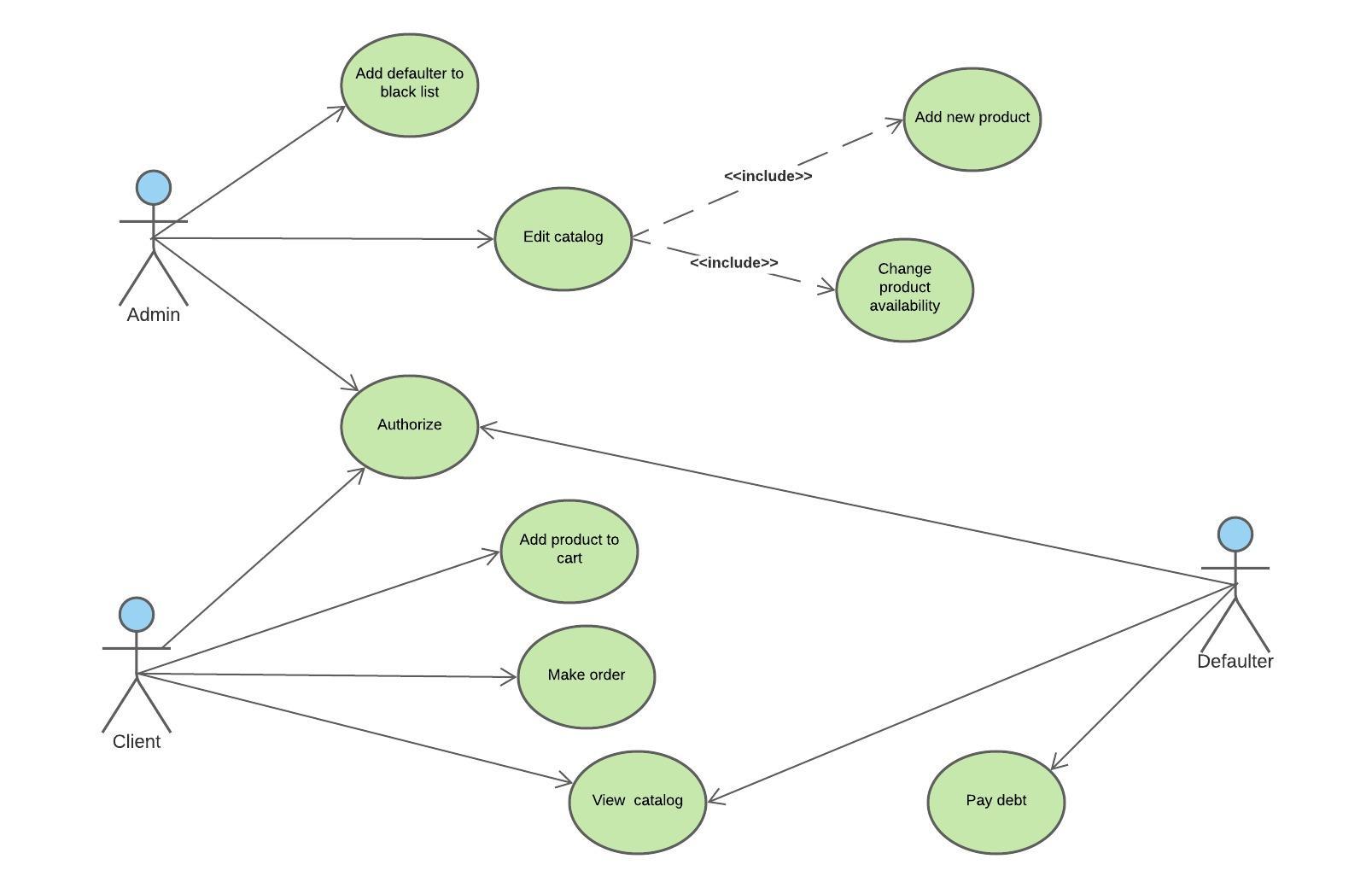


Рисунок 1 - Диаграмма вариантов использования системы

Вариант использования – Оформление заказа

Таблица 2 - ВИ Оформление заказа

|  |  |
| --- | --- |
| Actor | Клиент |
| Goal in context | Оформить заказ. |
| Trigger event | Оформить заказ |
| Pre-Condition | Пользователь авторизован, в корзине имеются товары. |
| Success End Condition | Заказ успешно оформлен. |
| Failed in Condition | Заказ отклонен по указанным причинам. |

Таблица 3 - Основной сценарий ВИ Оформление заказа

|  |  |  |
| --- | --- | --- |
| Step | Actor | Action Description |
| 1 | Клиент | На странице корзины кликает по кнопке оформления заказа. |
| 2 | Приложение | Выдает форму оформления заказа. |
| 3 | Клиент | Вводит/выбирает следующую информацию:   1. Способ доставки 2. Способ оплаты 3. Имя клиента 4. Контактный телефон |
| 4 | Приложение | Проверяет правильность заполнения формы. В случае успешного результата возвращает пользователю сообщение об успешном оформлении заказа. |

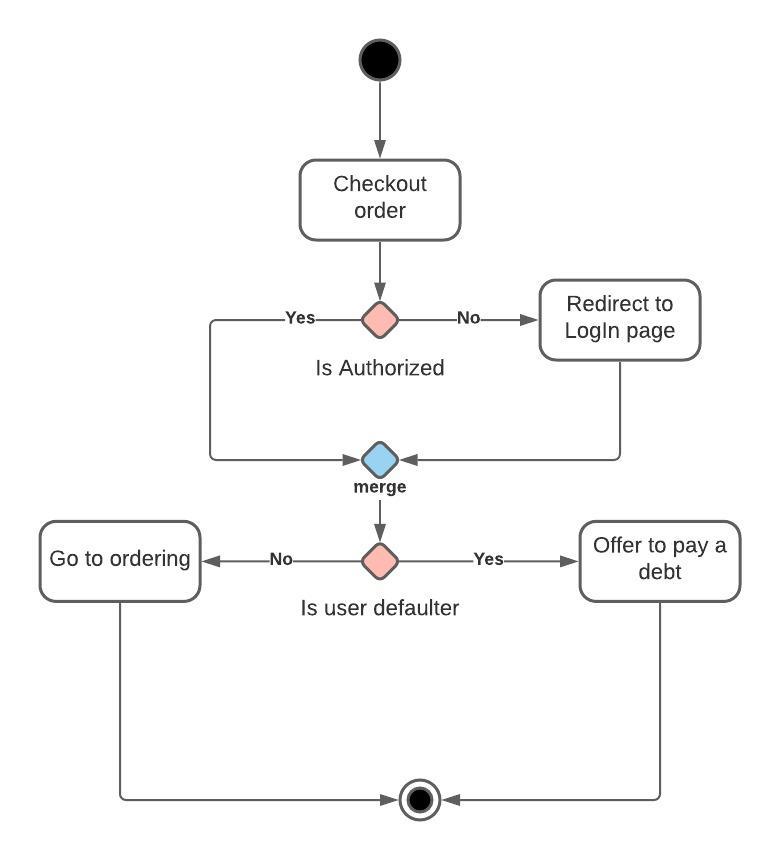


Рисунок 2 - Диаграмма активности для ВИ Оформление заказа

Таблица 4 - Альтернативный сценарий ВИ Оформление заказа

|  |  |  |
| --- | --- | --- |
| Step | Variable | Possible Variations |
| 3а | Обязательные поля формы не заполнены | Приложение указывает, какое поле не заполнено. |
| 3б | Поля формы заполнены неверно | Приложение указывает (сообщением под полем), если контактный телефон не соответствует формату. |

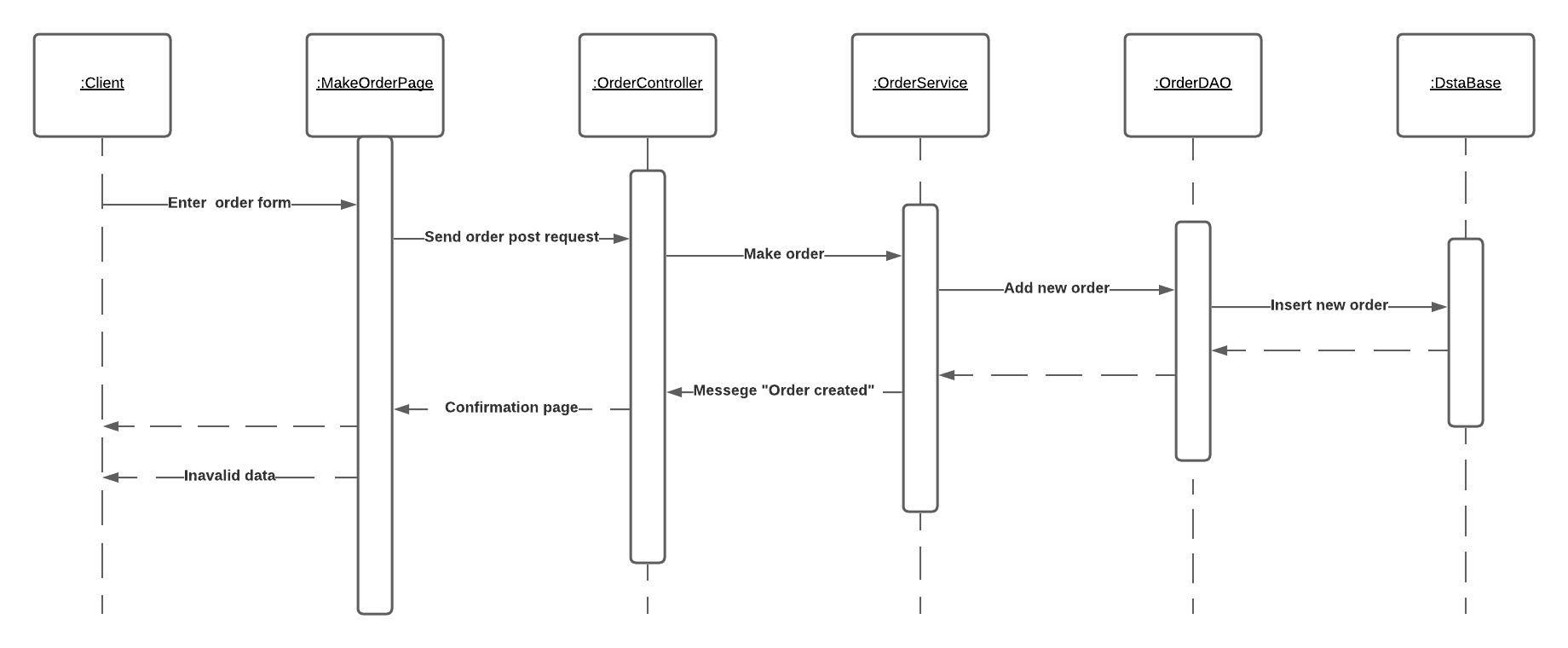


Рисунок 3 - Диаграмма последовательности для ВИ Оформление заказа

Вариант использования – Оплата задолженности

Таблица 5 - ВИ Оплата задолженности

|  |  |
| --- | --- |
| Actor | Неплательщик |
| Goal in context | Оплатить задолженности. |
| Trigger event | Добавить товар в корзину. |
| Pre-Condition | Пользователь авторизован. |
| Success End Condition | Задолженности оплачены, пользователь имеет активность Клиента. |
| Failed in Condition | Оплата отклонена по причинам. |

Таблица 6 - Основной сценарий ВИ Оплата задолженности

|  |  |  |
| --- | --- | --- |
| Step | Actor | Action Description |
| 1 | Неплательщик | Кликает по кнопке «Оплатить задолженность». |
| 2 | Приложение | Выдает форму оплаты долга. |
| 3 | Неплательщик | Вводит следующую информацию:   1. Способ оплаты    1. Если выбрана оплата банковской картой, то вводит реквизиты карты. |
| 4 | Приложение | Проверяет правильность заполнения формы. В случае успешного результата с помощью сервиса оплаты отправляет банковский запрос об оплате (если выбран вариант оплаты картой). |
| 5 | Приложение | Возвращает пользователю сообщение об оплате. Предоставляет возможность осуществлять покупки |

Таблица 7 - Альтернативный сценарий ВИ Оплата задолженности

|  |  |  |
| --- | --- | --- |
| Step | Variable | Possible Variations |
| 3а | Обязательные поля формы не заполнены | Приложение указывает, какое поле не заполнено. |
| 3б | Поля формы заполнены неверно | Приложение указывает (сообщением под полем), если реквизиты карты введены неверно. |
| 4б | Ошибка оплаты | Приложение возвращает форму оплаты с указанием неверных данных, если реквизиты карты введены неверно (например, такой карты не существует/у данного пользователя нет такой карты/неверный код). |

Вариант использования – Добавление товара

Таблица 8 - ВИ Добавление товара

|  |  |
| --- | --- |
| Actor | Администратор |
| Goal in context | Добавить новый товар. |
| Trigger event | Добавить товар в каталог. |
| Pre-Condition | Пользователь авторизован. |
| Success End Condition | Товар добавлен. |
| Failed in Condition | Добавление невозможно по указанным причинам. |

Таблица 9 - Основной сценарий ВИ Добавление товара

|  |  |  |
| --- | --- | --- |
| Step | Actor | Action Description |
| 1 | Администратор | Кликает по кнопке «Добавить новый товар». |
| 2 | Приложение | Выдает форму добавления товара. |
| 3 | Клиент | Вводит следующую информацию:   1. Наименование товара 2. Цену на товар 3. Выбирает наличие товара 4. Описание товара |
| 4 | Приложение | Проверяет правильность заполнения формы. В случае успешного результата перенаправляет администратора на страницу продукта. |

Таблица 10 - Альтернативный сценарий ВИ Добавление товара

|  |  |  |
| --- | --- | --- |
| Step | Variable | Possible Variations |
| 3а | Обязательные поля формы не заполнены | Приложение указывает, какое поле не заполнено. |
| 3б | Поля формы заполнены неверно | Приложение указывает (сообщением под полем), если указана неверная цена. |
| 4б | Такой товар уже существует | Приложение возвращает страницу с сообщением о существовании такого товара. |

**3.2 Технические требования**

**3.2.1 Архитектура системы**

Система «Интернет-магазин» представлена в виде приложения с многослойной архитектурой, построенного по схеме MVC с использованием языка Java. Данные в приложении разделены тремя основными слоями, при этом на данной платформе разработки система имеет промежуточные слои, как на следующей диаграмме.

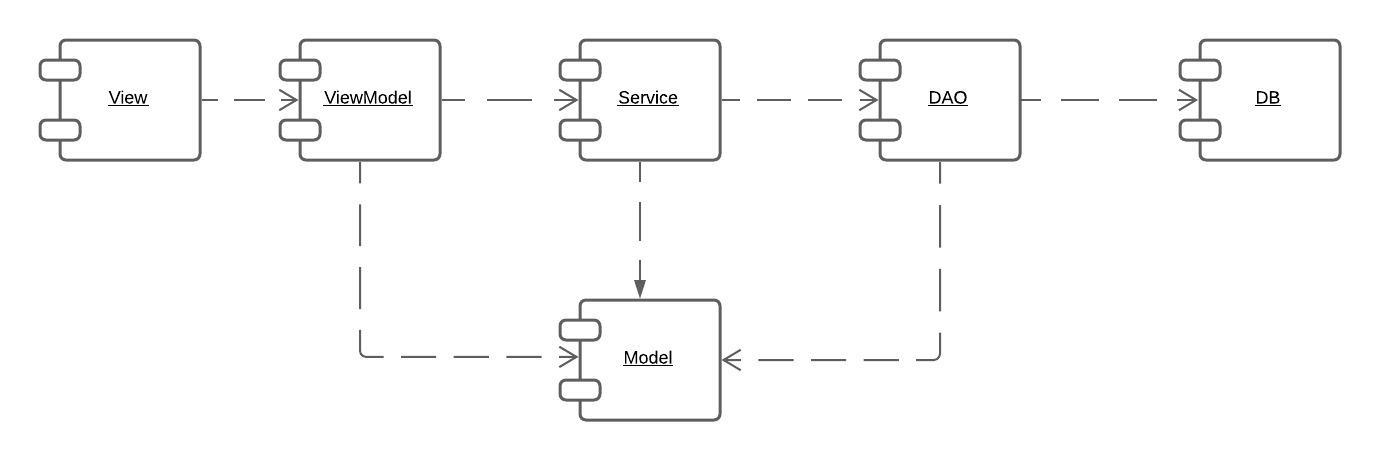


Рисунок 4 - Диаграмма компонентов для системы «Интернет-магазин»

Основные сущности приложения:

* Приложение – система
* Клиент – основной пользователь, который делает заказы.
* Неплательщик – клиент с непогашенным перед компанией-Заказчиком долгом
* Каталог – каталог товаров системы «Интернет-магазин», состоит из продаваемых товаров.
* Товар – продукция компании-Заказчика.
* Корзина – корзина товаров, куда Клиент может добавлять товары для дальнейшего оформления заказа.
* Заказ – заказ с данными о пользователе и статусе заказа.
* Статус заказа – реализован в виде перечисляемого типа, указывает на статус обработки заказа.

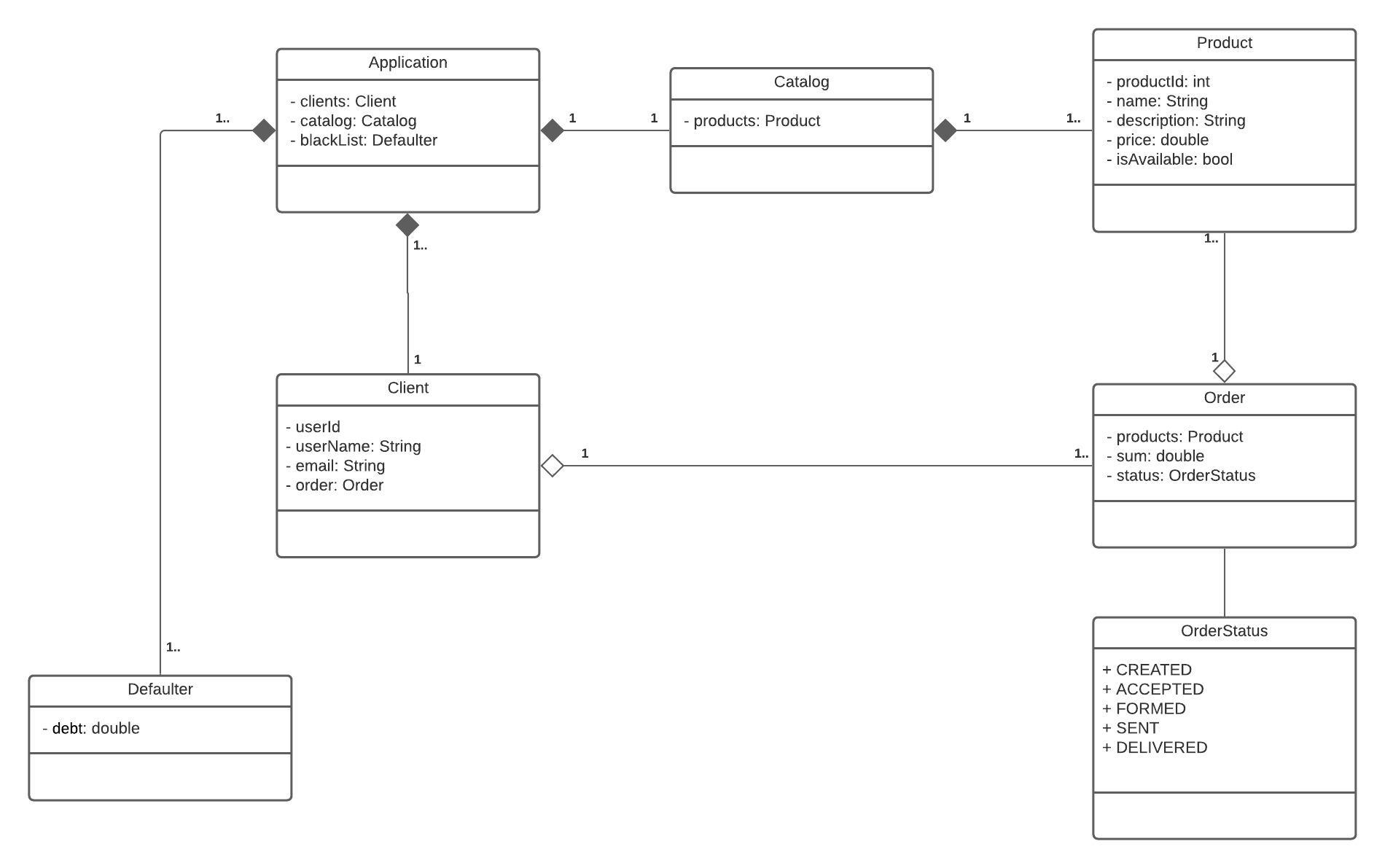


Рисунок 5 - Диаграмма классов для системы «Интернет-магазин»

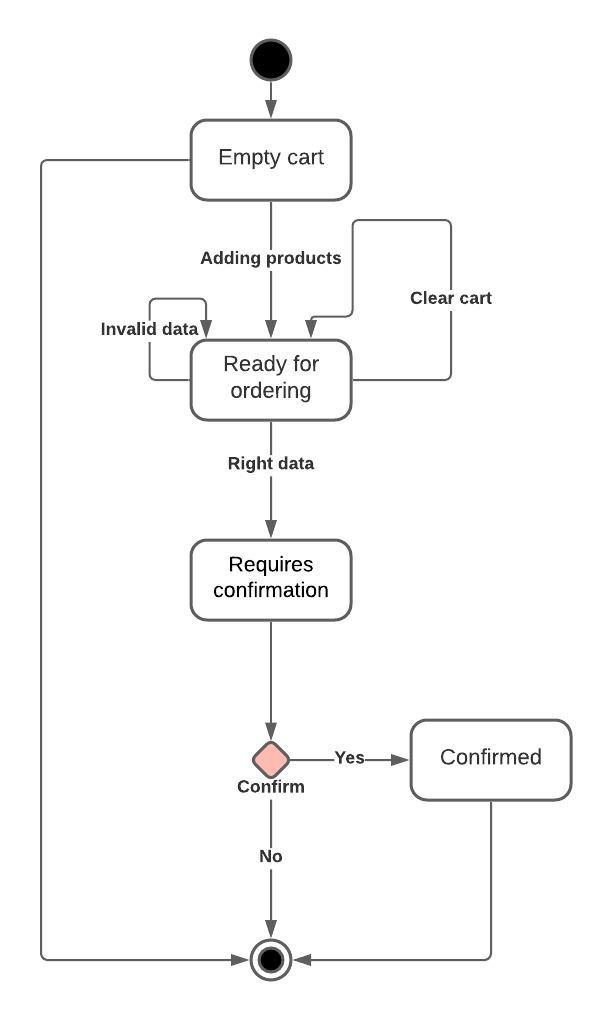


Рисунок 6 - Диаграмма состояния заказа

**3.2.2 Требования к пользовательскому интерфейсу**

Главная страница сайта должна содержать графическую часть, навигационное меню сайта, а также контентную область для того, чтобы посетитель сайта с первой страницы мог получить вводную информацию о продукции компании, а также ознакомиться с новыми продуктами и акциями. Контентная область первой страницы должна делиться на следующие разделы:- вступительная статья о компании со ссылкой, раздел «Контакты»;- краткая контактная информация – наименование, телефон и e-mail компании;- вверху страницы отображаются облегченная навигационная панель, которая обеспечивает переход к основным пунктам меню сайта (Главная, Каталог и т.д.);- поле поиска – предназначено для выполнения полнотекстового поиска по сайту;- поле входа на сайт для зарегистрированных пользователей или регистрации нового пользователя; - поле выбора дополнительной(справочной)информации(«полезная информация»);- поле быстрого перехода к подразделу «новые товары».

Графическая оболочка внутренних страниц должна делиться на следующие разделы:

- графическая шапка (активная, осуществляет переход на главную страницу)

- навигационное меню сайта (навигационная панель 2 обеспечивает переход к основным пунктам меню сайта);

- поле поиска – предназначено для выполнения полнотекстового поиска по сайту;

- навигационная панель по подразделам выбранного раздела сайта;

- поле для отображения контента выбранной страницы сайта;

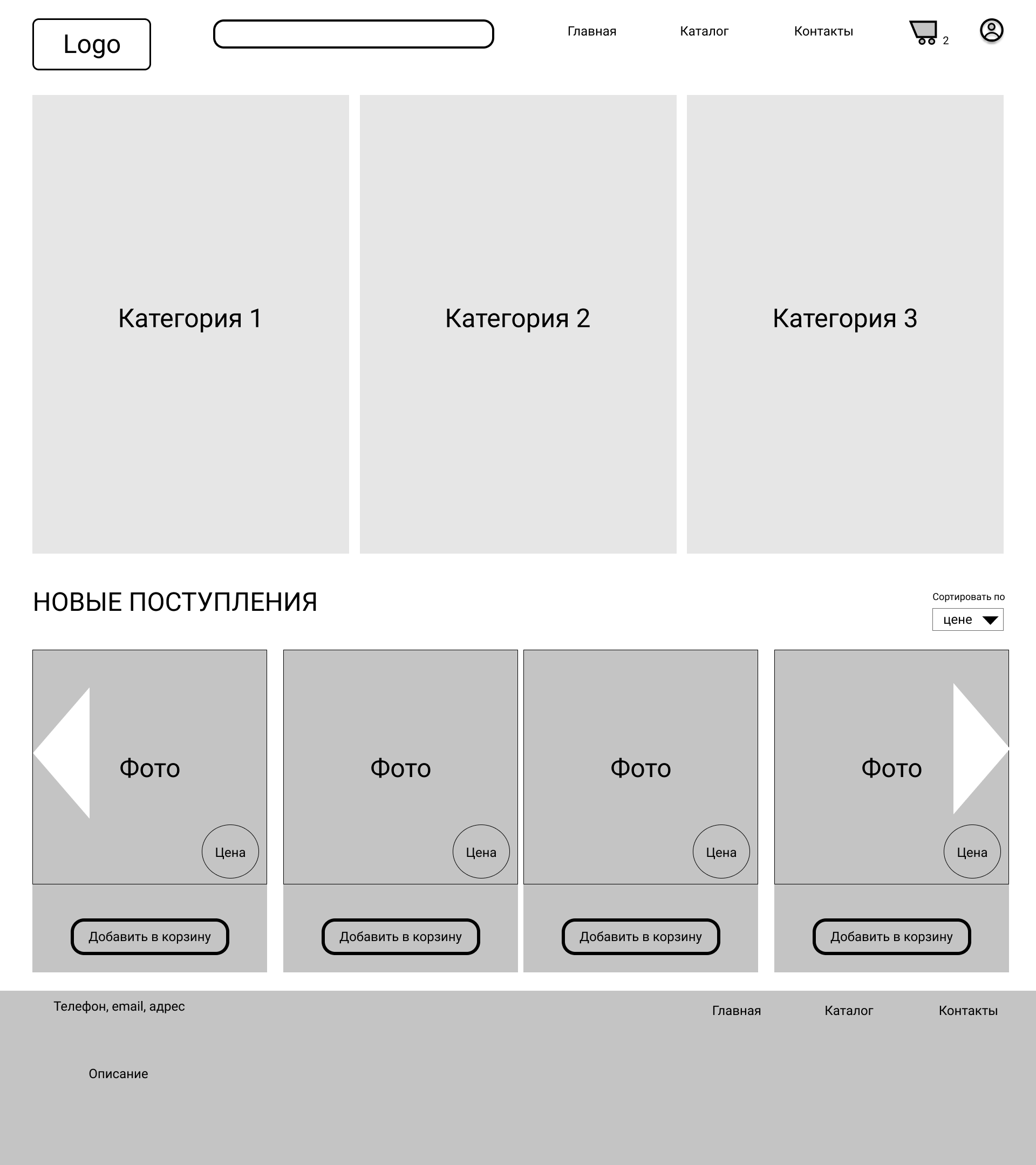


Рисунок 7 - Шаблон UX-дизайна главной страницы сайта

**3.2.3 Требования к безопасности**

Сайт должен предусматривать базовую защиту от основных видов атак: межсайтового скриптинга (XSS), SQL-инъекций, CSRF-уязвимостей.

Требуется защитить исходный код общей части сайта. Требуется разграничение доступа. Пароли пользователей хранятся в зашифрованном виде. Перехват данных на уровне протокола TCP невозможен. На уровне СУБД должно быть реализовано разграничение доступа к данным в БД.

**3.2.4 Требования к надежности**

В случае отказа работы серверной части и последующей недоступности веб-приложения, время восстановления не должно превышать одни рабочие сутки. Необходима автоматическая система бэкапов кода. Резервирование данных осуществляет хостинг-провайдер.

**3.2.5 Требования к языку программирования**

Исходный код системы должен быть написан на языке Java. Front-End часть должна быть разработана с использованием фреймворков React (не ниже версии 17) или Angular (не ниже версии 11).

4 Порядок приема системы

* 1. **Приемочные тесты**

В процессе разработки система должна быть покрыта модульными тестами. Необходимо протестировать следующие пункты: проверка правильности работы главных функций ресурса, корректность внутренних ссылок, проверка пользовательских форм. Включая проверку полей и страниц «авторизация», «регистрация», корректность работы «Добавление товара в корзину». Также необходимо провести тестирование производительности, куда входят стресс-тестирование для определения поведения сайта за пределами ожидаемой рабочей нагрузки, нагрузочное тестирование, проверка на способность работы в течение приемлемого периода, тестирование стабильности.

Таблица 11 - Тестовый сценарий для Оформления заказа

|  |  |
| --- | --- |
| **Test Case Id** | 1 |
| **Test Priority (Low/Medium/High)** | Medium |
| **Module Name** | Making order |
| **Test Title** | Verification of client’s data |
| **Description** | Testing the ordering system |
| **Test Designed date** | 20.05.2021 |
| **Pre-conditions** | Пользователь авторизован, в корзине есть товары |
|  |  |
| **Step** | **Expected Result** |
| 1. В низу корзины пользователь нажимает «Оформить заказ» | Пользователю возвращается страница оформления заказа |
| 1. Пользователь вводит свое имя | В поле ввода имени видно имя пользователя |
| 1. Пользователь вводит контактный телефон | В поле ввода телефона виден телефон верного формата |
| 1. Пользователь выбирает способ оплаты из выпадающего списка | В выпадающем списке выбран способ оплаты |
| 1. Пользователь нажимает кнопку «Оформить заказ» | Пользователю возвращается сообщение «Ваш заказ оформлен. В ближайшее время с Вами свяжется наш сотрудник» |
|  |  |
| **Post-Condition** | В базу данных добавлен новый заказ в статусе Created |

**4.2 Сроки исполнения**

Работы необходимо выполнить с помощью подхода Agile, заказчик оплачивает работу поэтапно. Срок разработки составляет 5 недель без учета корректировки заказчиком этапов работы.

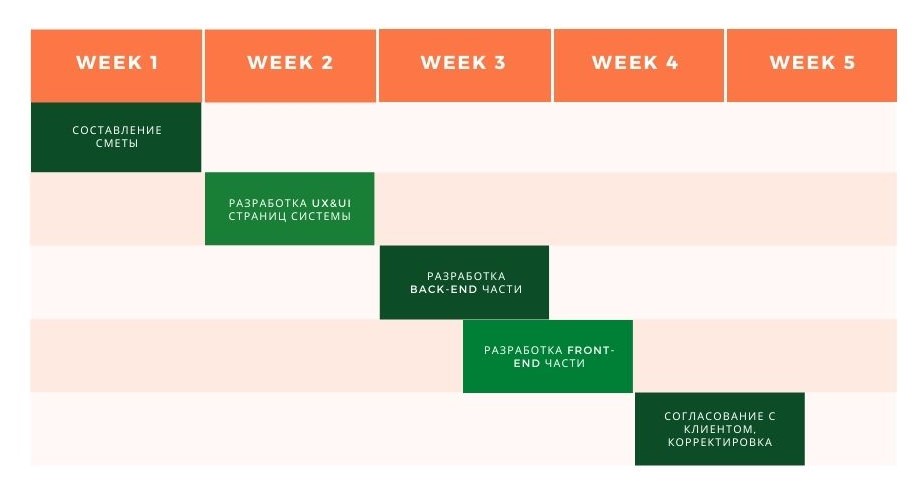


Рисунок 8 - Диаграмма Ганта сроков исполнения работы