# Спецификация требований

# Введение

Документ, который вы читаете, является спецификацией требований к созданию сайта пиццерии. Этот документ был создан для того, чтобы определить требования к дизайну и функциональности сайта. Он предназначен для всех заинтересованных сторон, включая дизайнеров, разработчиков и менеджеров проекта.

### Цель документа:

Целью этого документа является определение назначения и границ проекта, требований к сайту пиццерии.

### Соглашения, принятые в документе:

1. Все требования должны быть описаны ясно и конкретно.
2. Требования должны быть основаны на реальных потребностях бизнеса.
3. Требования не должны противоречить друг другу.
4. Любые изменения в требованиях должны быть обсуждены и согласованы всеми заинтересованными сторонами.

### Используемые нотации:

1. UML

# Описание системы

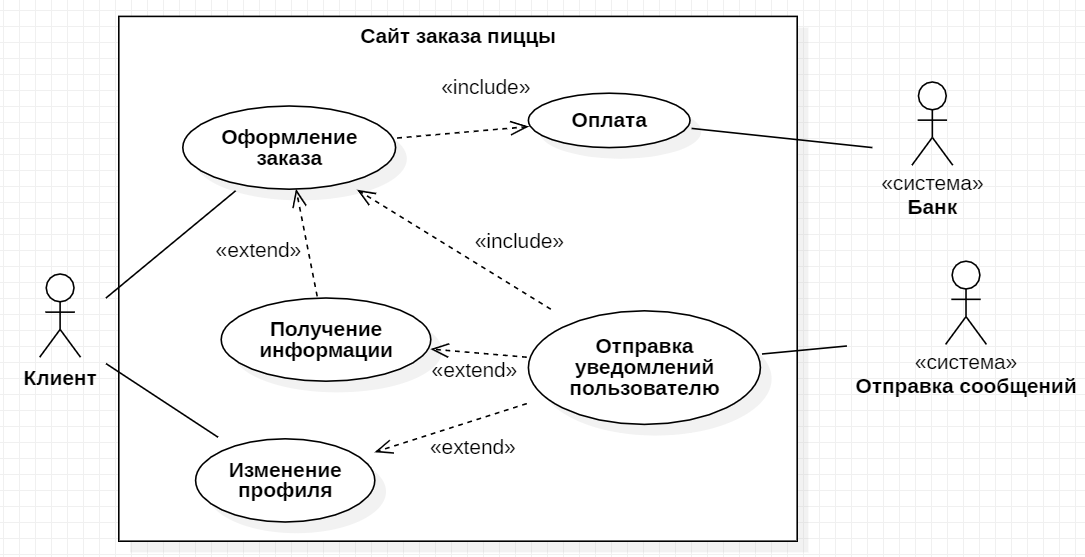
## Назначение

Система предназначена для автоматизации процесса продаж через интернет.

## Границы системы и бизнес-требования

Система должна обеспечить:

* Уменьшение нагрузки колл-центра на 70%.
* Уменьшение среднего времени оформления заказов до 3 минут (не учитывая время выбора товара) без непосредственного участия сотрудников пиццерии.
* Уменьшения максимума времени информирования клиента об изменении статуса заказа до 1 минуты.



## Краткое описание контекста использования

Система взаимодействует с внешними системами:

* Получает данные пользователей из CRM-системы посредством её REST API
* Получает данные заказов, товаров и остатков из ERP-системы посредством файлового обмена из общего FTP-сервера.
* Предоставляет информацию о пользователях и их заказах CRM- и ERP-системами посредством REST API.

## Заинтересованные лица

### Реестр заинтересованных лиц

| **Стейкхолдер** | **Классификация** | | | | **Стратегия взаимодействия** |
| --- | --- | --- | --- | --- | --- |
| **По участию** | **По заинтересованности** | **По влиянию** | |
| **Уровень власти** | **Уровень интереса** |
| Клиенты | Внешний | Сторонник | Низкий | Высокий | Держать в курсе дел |
| Владелец компании | Внутренний | Сторонник | Высокий | Низкий | Поддерживать удовлетворённость |
| Заказчик | Внутренний | Сторонник | Высокий | Высокий | Активно управлять |
| Инвестор | Внутренний | Сторонник | Высокий | Высокий | Активно управлять |
| Отдел маркетинга | Внутренний | Сторонник | Низкий | Высокий | Держать в курсе дел |
| Управляющие пиццериями | Внутренний | Сторонник | Низкий | Низкий | Наблюдать |
| Сотрудники колл-центра | Внутренний | Противник | Низкий | Высокий | Держать в курсе дел |
| Конкуренты | Внешний | Противник | Низкий | Низкий | Наблюдать |
| Куратор проекта | Внутренний | Сторонник | Высокий | Высокий | Активно управлять |
| Руководитель проекта | Внутренний | Сторонник | Высокий | Высокий | Активно управлять |
| Аналитик | Внутренний | Сторонник | Низкий | Высокий | Держать в курсе дел |
| Дизайнер | Внутренний | Сторонник | Низкий | Высокий | Держать в курсе дел |
| Разработчик | Внутренний | Сторонник | Низкий | Высокий | Держать в курсе дел |

### Матрица RACI

| **Задача \ ЗС** | **Руководство компании** | **Куратор проекта** | **Руководитель проекта** | **Аналитик** | **Дизайнер** | **Разработчик** | **Отдел маркетинга** | **Клиенты** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Сбор требований** | **A** | **C** | **R** | **R** |  |  | **C** |  |
| **Анализ требований** |  | **A** | **A** | **R** |  |  |  |  |
| **Планирование проекта** |  | **C** | **R** | **C** |  | **C** |  |  |
| **Создание прототипа и уточнение требований** |  | **A** | **A** | **R** | **R** | **R** |  |  |
| **Проектирование** |  | **C** | **A** | **C** | **C** | **R** |  |  |
| **Разработка** |  | **C** | **A** | **C** | **R** | **R** |  |  |
| **Внедрение** | **A** |  |  | **C** | **C** | **R** |  |  |
| **Использование** | **I** |  |  |  |  |  | **I** | **R** |
| **Поддержка** |  | **A** |  | **C** | **C** | **R** |  |  |

* R — исполнитель;
* A — утверждающий;
* C — консультант;
* I — информируемый.

## Типы пользователей

Типы пользователей:

* Клиенты.
  + Конечные пользователи системы, использующие её для приобретения товаров. Имеют права только на просмотр информации, необходимой для покупки товаров.
* Сотрудники компании.
  + Используют систему для получения заказов от клиентов. Имеют ограниченные права доступа, необходимые для обработки заказов.
* Администраторы.
  + Тип пользователей с полными правами доступа, следящие за работоспособностью системы.

## Основные функции

Функции системы:

1. Демонстрация товаров в наличии.
2. Создание заказа.
3. Редактирование корзины товаров.
4. Добавление, обновление и удаление информации из личного кабинета.
5. Возможность проведения оплаты банковскими картами.
6. Возможность отслеживания статуса заказа.

## Ограничения

Ограничения системы:

* Пользователи будут использовать систему, используя канал связи низкой пропускной способности.
* При разработке системы запрещается использовать open-source-компоненты.
* Система должна хранить данные пользовательских действий в течение 30 дней по требованиям федерального закона.

# Системные требования

## Функциональные требования

### Epic №1 Авторизация в системе

#### User story №1 Авторизация

Я как пользователь хочу авторизоваться на сайте доставки, чтобы заказать пиццу и использовать ранее сохранённую информацию для заказов (телефон, адрес и т.д.)

*Критерии приёмки:*

* Пользователь может войти используя почту и одноразовый код с почты.
* Пользователь может войти используя телефон и одноразовый СМС-код.
* На экране может быть выбор между двумя способами авторизации.

*Функциональные требования:*

* Система должна отправлять одноразовый код авторизации на почту пользователя, когда он авторизуется через почту.
* Система должна принимать одноразовый код авторизации от пользователя, когда он авторизуется.
* Система автоматически различать способ входа (телефон/почта), когда пользователь вводит номер телефона или почту.
* Система должна отправлять одноразовый СМС-код авторизации на телефон пользователя, когда он авторизуется через телефон.
* Система должна предлагать регистрацию, если пользователя с введённой почтой или телефоном нет.

*Сценарий “Авторизация”*

*Цель:* авторизоваться в системе

*Акторы:* пользователь, система

*Основной сценарий:*

1. Пользователь открывает сайт системы и вводит номер телефона в специальное поле для заполнения.
2. Система выводит на странице сайта, что СМС-код отправлен, и отправляет на указанный номер телефона одноразовый СМС-код для авторизации.
3. Пользователь вводит одноразовый СМС-код в свободное поле на сайте системы.
4. Система авторизует пользователя и показывает каталог товаров.

*Авторизация через почту:*

1. Пользователь открывает сайт системы и вводит адрес своей почты в специальное поле для заполнения.
2. Система выводит на странице сайта, что код отправлен на почту, и отправляет на указанную почту одноразовый код для авторизации.
3. Пользователь вводит одноразовый код из почты в свободное поле на сайте системы.
4. Система авторизует пользователя и показывает каталог товаров.

*Ошибка авторизации по телефону:*

1. Пользователь открывает сайт системы и вводит номер телефона в специальное поле для заполнения.
2. Система выводит ошибку и предлагает регистрацию.

*Ошибка авторизации по телефону:*

1. Пользователь открывает сайт системы и вводит почту в специальное поле для заполнения.
2. Система выводит ошибку и предлагает регистрацию.

#### User story №2 Регистрация

Я как пользователь хочу зарегистрироваться на сайте доставки, чтобы заказать пиццу, иметь возможность использовать в будущем сохранять информацию для заказов (телефон, адрес…) и прочее.

*Критерии приёмки:*

* Пользователь может зарегистрироваться в системе только по номеру мобильного телефона.
* Пользователь получает одноразовый СМС-код регистрации.
* Пользователь может авторизоваться в системе после регистрации.

*Функциональные требования:*

* Система должна запрашивать телефон пользователя при регистрации.
* Система должна запрашивать почту пользователя при регистрации.
* Система должна отправлять одноразовый СМС-код регистрации на телефон пользователя, когда он выбрал отправку кода на телефон.
* Система автоматически создаёт новый профиль пользователя при регистрации.
* Система должна отправлять СМС-код о завершении регистрации на номер пользователя.

*Сценарий “Регистрация”*

*Цель:* зарегистрироваться в системе

*Акторы:* пользователь, система

*Основной сценарий:*

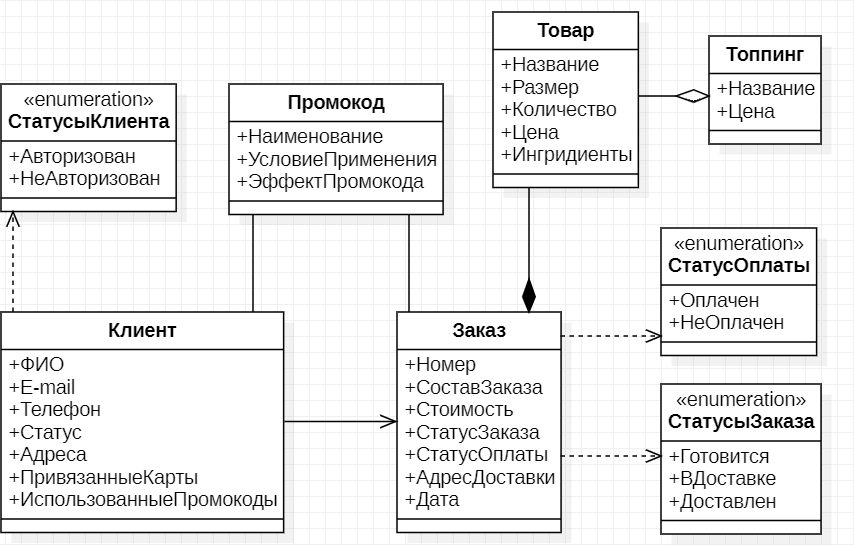
1. Пользователь открывает страницу регистрации на сайте доставки и вводит номер мобильного телефона в специальное поле для заполнения.
2. Система проверяет введённый номер на правильность и отправляет на него СМС-код регистрации.
3. Пользователь вводит полученный СМС-код в свободное поле на сайте.
4. Система создаёт новый профиль пользователя и сохраняет в нём номер телефона.
5. Система отправляет СМС-код о завершении регистрации на номер пользователя.

## Логическая модель данных

Основные сущности и их атрибуты:

* Клиент;
  + Конечный пользователь системы;
* СтатусыКлиента;
  + Авторизован/не авторизован;
* Промокод;
  + Код, предоставляющий бонус клиенту;
* Заказ;
  + Поручение на приготовление и доставку товара, хранящее в себе всю необходимую информацию для удовлетворения потребности клиента;
* Товар;
  + Предмет, представляющий ценность. В контексте проекта - пицца;
* Топпинг;
  + Модификация товара, придающая ему дополнительные свойства;
* СтатусОплаты;
  + Оплачен/не оплачен;
* СтатусыЗаказа;
  + Готовится/в доставке/доставлен.

Атрибуты и связи сущностей представлены на диаграмме классов:



## Программные интерфейсы

API системы сайта пиццы будет основан на архитектуре REST.

### Описание методов и возвращаемых параметров:

#### CRUD-методы коллекции пицц

##### POST

*Запрос*

* Ресурс: /api/v1/pizzas
* Описание: вставка новой конкретной информации о пицце
* Метод: POST
* Параметры запроса:
  + name - название пиццы;
  + price - цена;
  + ingredients - состав
* Авторизация: токен пользователя в заголовке Authorization
* Пример запроса: POST mypizzeria.com/api/v1/pizzas?name=cheese&price=600

*Ответ*

* HTTP-код: 201 Created.

##### GET

*Запрос*

* Ресурс: /api/v1/pizzas
* Описание: получение информации всех о пиццах
* Метод: GET
* Параметры запроса:
  + name - получение названий пицц (0 - включать информацию в ответе, 1 - не включать информацию в ответе);
  + price - получение цен пицц (0 - включать информацию в ответе, 1 - не включать информацию в ответе);
  + ingredients - получение состава (0 - включать информацию в ответе, 1 - не включать информацию в ответе).
* Авторизация: токен пользователя в заголовке Authorization
* Пример запроса: GET mypizzeria.com/api/v1/pizzas?name=0&price=1&ingredients=1

*Ответ*

* HTTP-код: 200 OK.
* Заголовки: Content-Type:application/json;
* Тело: структура данных пицц

| **Параметр** | **Тип** | **Описание** |
| --- | --- | --- |
| pizza-id | Число | Идентификатор пиццы |
| name | Строка | Название пицц |
| price | Число с плавающей запятой | Цена пиццы |
| ingredients | Текст | Состав |

## Пользовательские интерфейсы

Проект Figma:

<https://www.figma.com/file/2CBuwn1Xk9f4pnVF1pE3I9/%D0%A2%D0%B5%D1%81%D1%82?type=design&node-id=0%3A1&t=3LcJ9l2BHpYIhhX8-1>

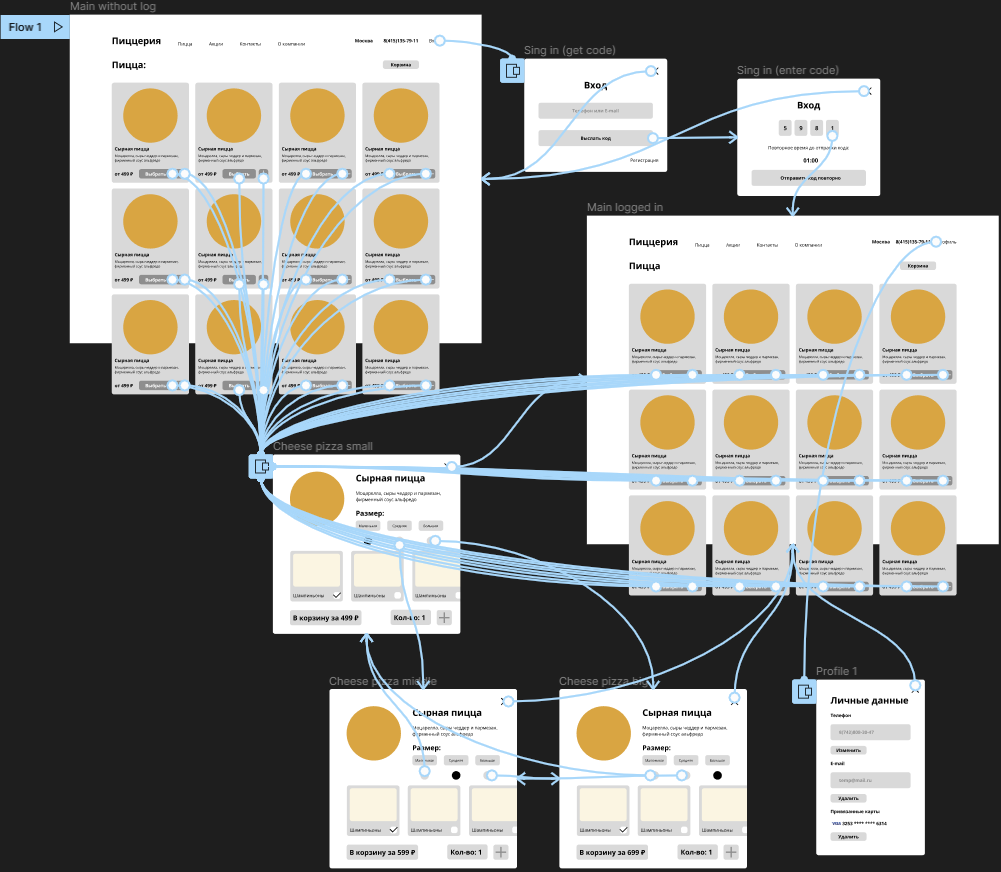
Прототип Figma:

<https://www.figma.com/proto/2CBuwn1Xk9f4pnVF1pE3I9/%D0%A2%D0%B5%D1%81%D1%82?type=design&node-id=5-17&scaling=min-zoom&page-id=0%3A1&starting-point-node-id=5%3A17>

### Таблица связи веб-страниц (всплывающих окон) с требованиями:

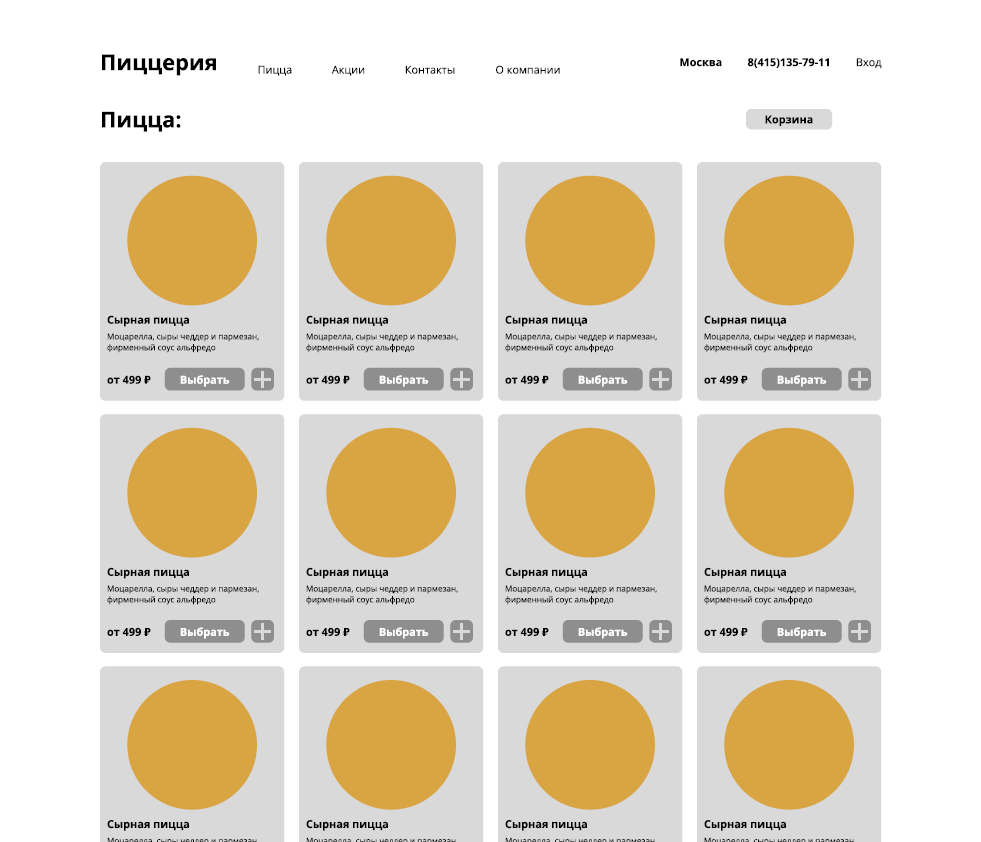
| **Название веб-страницы** | **Выполняемое требование** | |
| --- | --- | --- |
| Главная страница (до входа в профиль) | * (User Story 4) Система должна добавлять единицу товара в корзину, когда пользователь находится в каталоге товаров (не вошёл в меню товара). | |
| Всплывающее окно входа для ввода телефона или почты | * (User Story 1) Система автоматически различать способ входа (телефон/почта), когда пользователь вводит номер телефона или почту. * (User Story 1) Система должна предлагать регистрацию, если пользователя с введённой почтой или телефоном нет. | |
| Всплывающее окно входа для вставки кода | * (User Story 1) Система должна принимать одноразовый код авторизации от пользователя, когда он авторизуется. | |
| Регистрация (ввод данных) | * (User Story 1) Система должна предлагать регистрацию, если пользователя с введённой почтой или телефоном нет. * (User Story 2) Система должна запрашивать телефон пользователя при регистрации. * (User Story 2) Система должна запрашивать почту пользователя при регистрации. | |
| Меню настройки пиццы | * (User Story 4) Система должна добавлять товар с указанными настройками (количество, топпинг) в корзину пользователя из меню товара. | |
| Главная страница (после входа в профиль): | * (User Story 4) Система должна добавлять единицу товара в корзину, когда пользователь находится в каталоге товаров (не вошёл в меню товара). | |
| Профиль - вариант №1 (как всплывающее окно): | * (User Story 3) Система должна позволять пользователю привязывать банковскую карту к профилю для оплаты. * (User Story 3) Система должна позволять пользователю отвязывать банковскую карту к профилю для оплаты. | |
| Профиль - вариант №2 (как всплывающее веб-страница): | * (User Story 3) Система должна позволять пользователю привязывать банковскую карту к профилю для оплаты. * (User Story 3) Система должна позволять пользователю отвязывать банковскую карту к профилю для оплаты. | |
| Профиль - вариант №3 (как всплывающее веб-страница): | * (User Story 3) Система должна позволять пользователю привязывать банковскую карту к профилю для оплаты. * (User Story 3) Система должна позволять пользователю отвязывать банковскую карту к профилю для оплаты. | |

### Карта сайта:



### Прототипы:

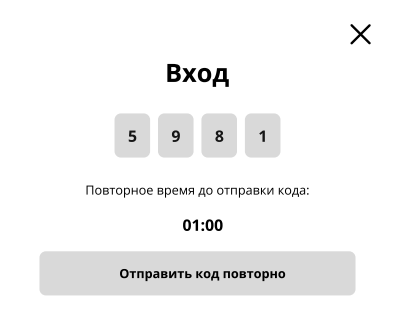
#### Главная страница (до входа в профиль):



#### Всплывающее окно входа для ввода телефона или почты:



#### Всплывающее окно входа для вставки кода:



## Удобство использования

Цвета и шрифты пользовательского интерфейса веб-сайта должны соответствовать требованиям стандарта по инклюзивному дизайну WCAG 2.0

## Атрибуты качества

Атрибуты качества системы:

* Система должна обрабатывать 95% запросов от пользователей не дольше 1 секунду.
* Система должна обеспечивать доступность на уровне 99,9% времени работы.