**ОНЛАЙН-СЕРВИС ЗАКАЗА ПИЦЦЫ**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**требования к программному обеспечению**

|  |  |
| --- | --- |
| Резюме: | |
| Автор документа: | Ушкова А. С. |
| Рассмотрено: |  |
| Версия: | 1.0.3 |
| Дата последнего изменения: | 02.08.2022 |

**Журнал изменений**

|  |  |  |  |
| --- | --- | --- | --- |
| **Дата** | **Версия** | **Описания** | **Автор** |
| 07.08.2022 | Черновик | Создан первый черновой вариант | Ушкова А.С. |
|  |  |  |  |

[**Введение**](#_yjdwb3pbxzmn) **4**

[1.1. Назначение](#_n4ffuhp0dyhx) 4

[**Общее описание**](#_n4duq88jhdlc) **5**

[2.1. Общий взгляд на продукт](#_14dk2yjhrks1) 5

[2.2. Классы пользователей](#_9o7nyz2iad0w) 5

[2.3. Операционная среда](#_oolxqbvq3qpx) 5

[**Функции системы**](#_xip74rqzerce) **6**

[3.1. Регистрация пользователей](#_1fxg1xre5pey) 6

[3.1.1. Описание](#_igs76pc0ga62) 6

[3.1.2. Функциональные требования](#_fn4i5tkj6lab) 6

[3.2. Оформление заказа пользователя](#_daahjdkajnq6) 8

[3.2.1. Описание.](#_66hjj026zzop) 8

[3.2.2. Функциональные требования](#_ge31i8e6kp4b) 8

[3.3. Просмотр заказов](#_3j3kcep6vb2s) 10

[3.3.1. Описание](#_rypz5ixje6fx) 10

[3.3.2. Функциональные требования](#_5h72l0as4mv9) 10

[**Требования к данным**](#_i4fbit8h48d) **14**

[4.1. Физическая модель данных](#_pbe3lxrxbwdh) 14

[**Требования к внешним интерфейсам**](#_dvgvcgdw3dqn) **15**

[5.1. Пользовательские интерфейсы](#_7360s6gr842n) 15

[**Приложение А. Словарь терминов**](#_pt4b9taf2thq) **21**

[**Приложение Б. Модели анализа**](#_dv7kuv2qd6d8) **22**

[Use case регистрация пользователя](#_djoz7qaiwdh7) 24

[Use case оформление заказа](#_ce9dvcr94ltx) 26

[Use case просмотр заказов](#_mhd5n1aozvkt) 27

# **Введение**

Цель Спецификации требований к программному обеспечению онлайн-сервиса заказа пиццы состоит в том, чтобы предоставить четкое и последовательное видение Онлайн-сервиса заказа пиццы и перечислить новые Требования для его программного обеспечения.

Этот документ предназначен для обеспечения общего понимания целей программного обеспечения и задачи как команды разработчиков, так и бизнес-экспертов.

Документ должен постоянно обновляться, чтобы отражать функциональные изменения, которые произойдут во время реализации проекта онлайн-сервиса заказа пиццы, так чтобы точное видение того, что было сделано, было доступно в любое время.

## **1.1. Назначение**

Данная спецификация требований относится к нововведениям мобильного приложения для заказы пиццы: регистрация, просмотр заказов и добавление ID пользователя в оформленный заказ. Данный документ адресован разработчикам и тестировщикам.

# **Общее описание**

Данный продукт представляет дополнение к мобильному приложению, предназначенное для заказа пиццы. В нем, после обновления функционала, можно будет зарегистрироваться в личном кабинете и просмотреть список всех ранее оформленных заказов. Это приложение предназначено для клиентов пиццерии.

## **2.1. Общий взгляд на продукт**

Данный продукт является новой версией существующего приложения.

## **2.2. Классы пользователей**

Следующие классы пользователей могут быть определены в рамках онлайн-сервиса заказа пиццы:

**Клиент:** пользователь, который оформляет заказ на доставку пиццы

* **Android-пользователь:** пользователь, который оформляет заказ через Android-приложение.
* **IOS-пользователь:** пользователь, который оформляет заказ через IOS-приложение.

## **2.3. Операционная среда**

Операционную среду для существующего приложения следует оставить такой, какая она есть в настоящее время. Ниже перечислены операционные среды для новых дополнений продукта:

|  |  |
| --- | --- |
| **ID требований** | **Требования** |
| ОС-1 | Приложение по заказам будет работать со следующими версиями операционных систем:   * Android 8 и выше; * IOS 14.8.1 и выше; |

# **Функции системы**

## **3.1. Регистрация пользователей**

## **3.1.1. Описание**

Данная функция осуществляет регистрацию пользователей в личном кабинете. Приоритет: высокий.

## **3.1.2. Функциональные требования**

Пользователь заполняет в мобильном приложении все поля при регистрации: *фамилия, имя, отчество, номер телефона, логин и пароль.* Функция эти данные считывает и передает их на сервер. Сервер проверяет, есть ли в бд уже данный пользователь. Если нет, то присваивает ему уникальный идентификатор, записывает данные в БД (пароль при этом передается и записывается в зашифрованном виде) и отправляет приложению ответ о том, что пользователь успешно зарегистрирован. В противном случае, система выдаст страницу с ошибкой.

Если пользователь введет некорректные данные система также выдаст страницу об ошибке.

**Запрос/Ответ**

|  |  |
| --- | --- |
| Запрос | Ответ |
| POST /user  "lastName": "Ivanov",  "firstName": "Ivan",  "middleName": "Ivanovich",  "phone": "89278353639",  "login": "ivanov@gmail.com",  "password": "5994471abb01112afcc18159f6cc74b4f511b99806da59b3caf5a9c173cacfc5"  } | HTTP/1.1 201 Created  {  "user\_id": 101  } |

**Входные параметры**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Параметр** | **Тип данных** | **Обязательность** | **Описание** | **Варианты значений** |
| 1 | lastName | string | + | Фамилия пользователя | Ivanov |
| 2 | firstName | string | + | Имя пользователя | Ivan |
| 3 | middleName | string | - | Отчество пользователя | Ivanovich |
| 4 | phone | string | + | Номер телефона пользователя | 89278353639 |
| 5 | login | string | + | Логин пользователя, который используется при авторизации | ivanov@gmail.com |
| 6 | password | string | + | Пароль пользователя, который используется при авторизации | 12345 |

**Выходные параметры**

Положительный ответ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Параметр** | **Тип данных** | **Обязательность** | **Описание** | **Варианты значений** |
| 1 | user\_id | int | + | Уникальный идентификатор пользователя | 101 |

Ответ с ошибками

|  |  |
| --- | --- |
| **Код** | **Описание** |
| 400 | Некорректный запрос |

## **3.2. Оформление заказа пользователя**

## **3.2.1. Описание.**

Данная функция позволяет пользователю сделать заказ пиццы в мобильном приложении. Приоритет: высокий.

## **3.2.2. Функциональные требования**

Заказ может сделать только зарегистрированный и авторизованный пользователь. Он выбирает все позиции, которые хочет добавить в заказ. Далее нажимает на кнопку “Оформить заказ”. В этот момент функция берет данные заказа и уникальный идентификатор пользователя (благодаря ему, система должна понять, какому пользователю принадлежит текущий заказ.) и передает их серверу. Сервер обрабатывает эти данные, добавляет их в БД и присваивает заказу уникальный идентификатор. В случае сбоя, сервер возвращает страницу с ошибкой.

**Запрос/Ответ**

|  |  |
| --- | --- |
| **Запрос** | **Ответ** |
| POST /orders  {  "crust": "original",  "toppings":  [  "cheese",  "pepperoni",  "garlic"  ],  "castomer\_id": 101  } | HTTP/1.1 201 Created  {  "order\_id":5  } |

**Входные параметры**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Параметр** | **Тип данных** | **Обязательность** | **Описание** | **Варианты значений** |
| 1 | crust | string | + | Тесто для пиццы (корочка) | “thin” или “thick” |
| 2 | toppings | array | - | Начинка для пиццы | если не указано – по умолчанию только “cheese”. Доступные варианты:  "cheese", "pepperoni", "garlic" |
| 3 | castomer\_id | int | + | Id пользователя, сделавший заказ | 101 |

**Выходные параметры**

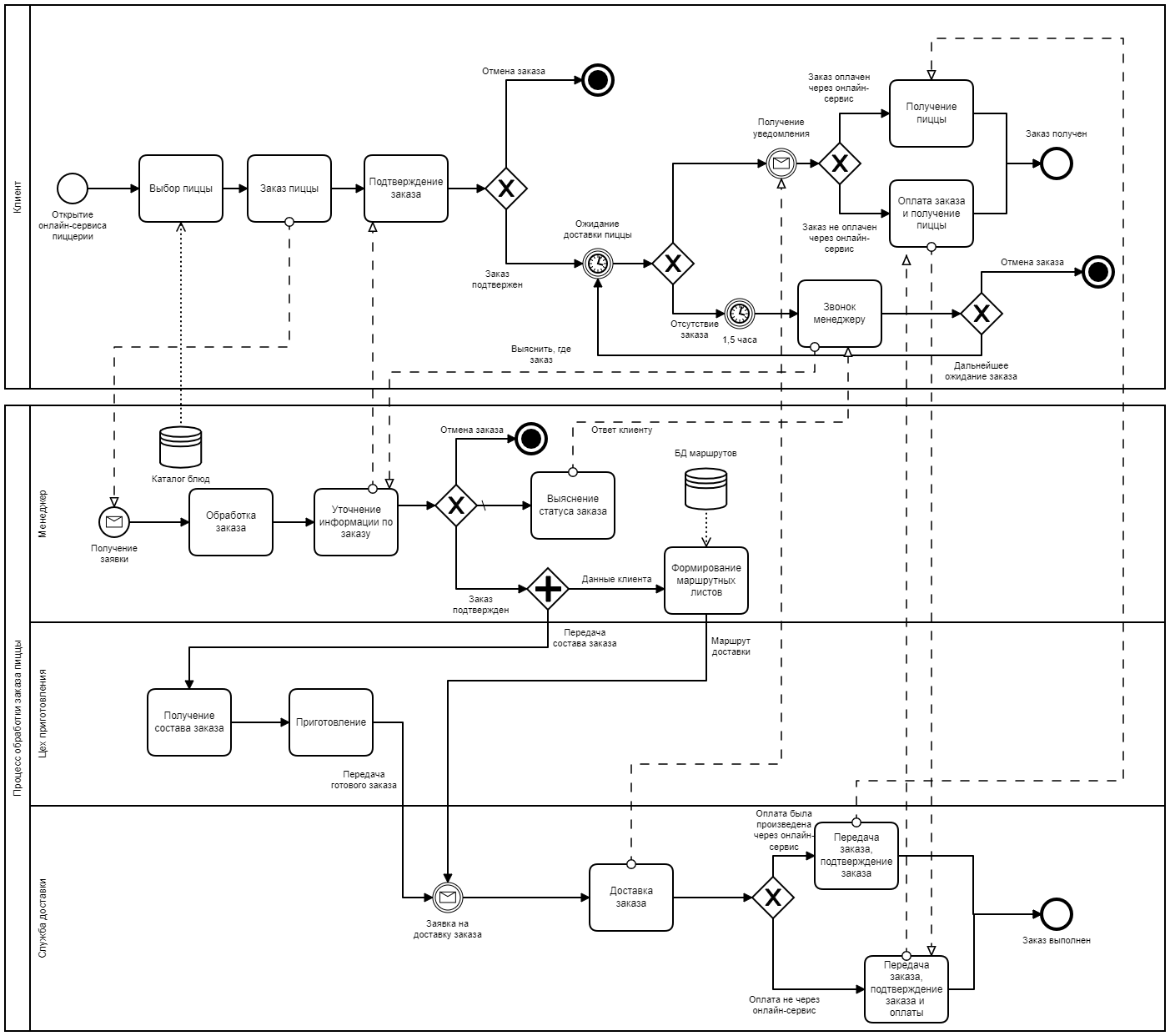
Положительный ответ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Параметр** | **Тип данных** | **Обязательность** | **Описание** | **Варианты значений** |
| 1 | order\_id | int | + | Уникальный идентификатор заказа | 5 |

Ответ с ошибками

|  |  |
| --- | --- |
| **Код** | **Описание** |
| 400 | Некорректный запрос |

Ниже приведен BPMN-процесс заказа пиццы:



## **3.3. Просмотр заказов**

## **3.3.1. Описание**

С помощью данной функции пользователю можно просмотреть все ранее сделанные заказы (история заказов). Приоритет: высокий.

## **3.3.2. Функциональные требования**

Свою историю заказов может посмотреть только зарегистрированный и авторизованный пользователь. Он в мобильном приложении переходит в личный кабинет, и там ему выпадает список его заказов.

Функция берет уникальный идентификатор пользователя и отправляет его на сервер. Сервер проверяет, существует ли данный пользователь в БД. Если существует, сервер ищет в БД все заказы по ID пользователя. Все найденные заказы отправляются назад в приложение в виде списка. Если ничего не будет найдено, сервер возвращает NULL, и приложение выдает клиенту, что клиент не сделал ни одного заказа.

В случае сбоя, сервер возвращает страницу с ошибкой.

**Запрос/Ответ**

|  |  |
| --- | --- |
| **Запрос** | **Ответ** |
| POST /getOrders/<castomer\_id>  {  "user\_id": 101  } | HTTP/1.1 200 Success  {  {  "crust": "original",  "toppings": [  "cheese",  "pepperoni",  "garlic"  ]  },  {  "crust": "original",  "toppings": [  "pepperoni",  "garlic"  ]  }  } |

**Входные параметры**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Параметр** | **Тип данных** | **Обязательность** | **Описание** | **Варианты значений** |
| 1 | user\_id | int | + | Уникальный идентификатор пользователя | 101 |

**Выходные параметры**

Положительный ответ

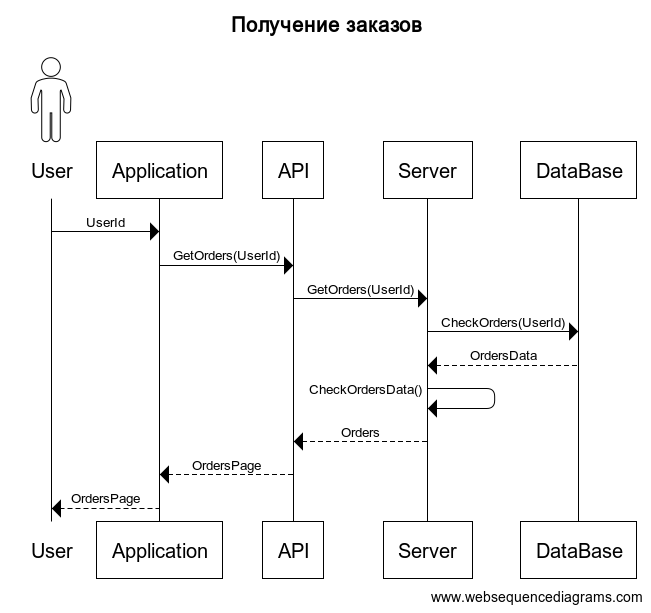
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **№** | **Параметр** | **Тип данных** | **Обязательность** | **Описание** | **Варианты значений** |
| 1 | crust | string | + | Тесто для пиццы (корочка) | “thin” или “thick” |
| 2 | toppings | array | - | Начинка для пиццы | если не указано – по умолчанию только “cheese”. Доступные варианты:  "cheese", "pepperoni", "garlic" |

Ответ с ошибками

|  |  |
| --- | --- |
| **Код** | **Описание** |
| 400 | Некорректный запрос |

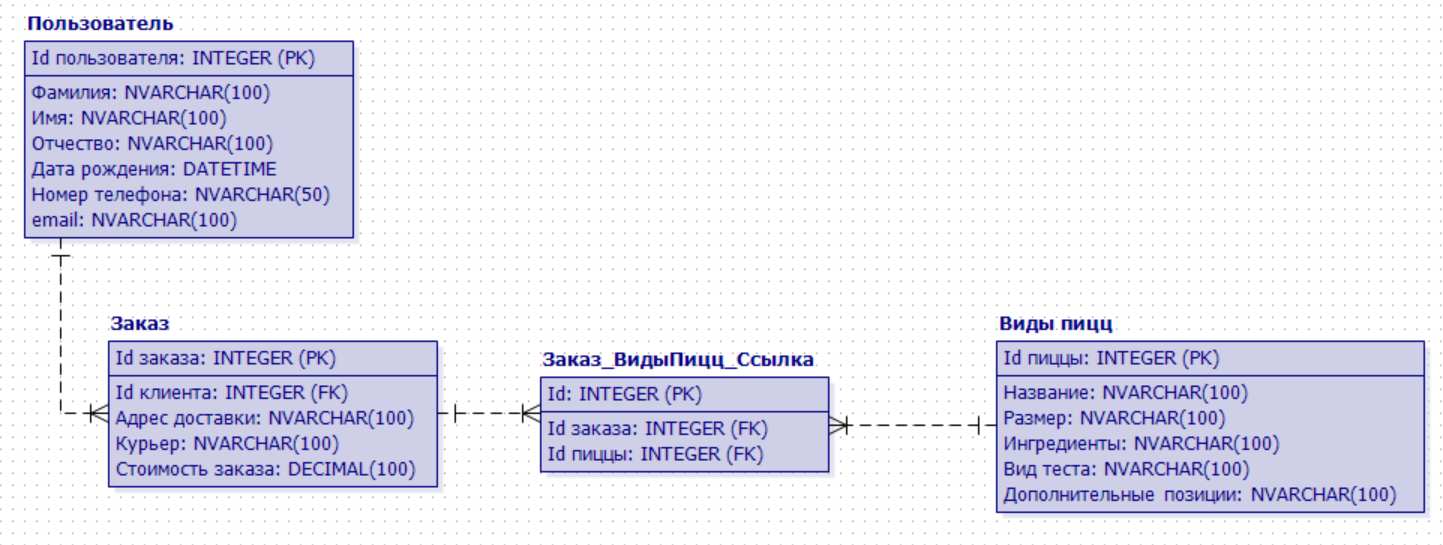
**Описание интеграции**

## 



# **Требования к данным**

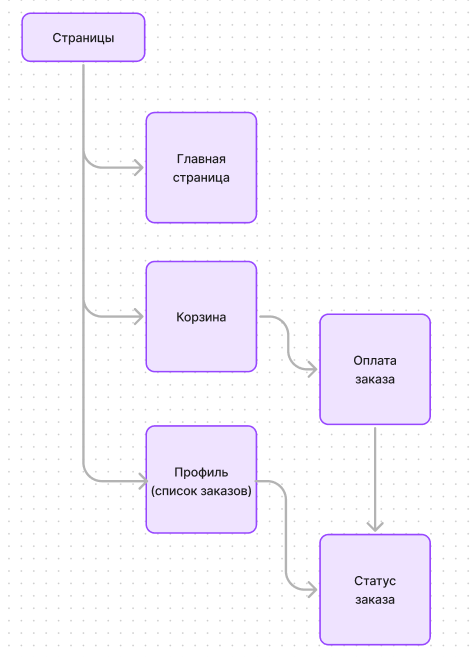
## **4.1. Физическая модель данных**



# **Требования к внешним интерфейсам**

## **5.1. Пользовательские интерфейсы**

Карта переходов:

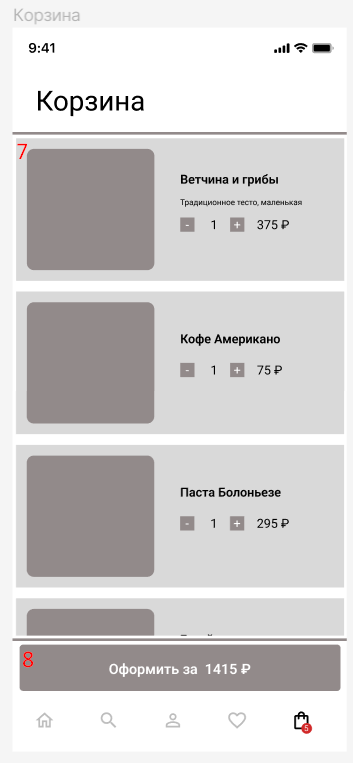
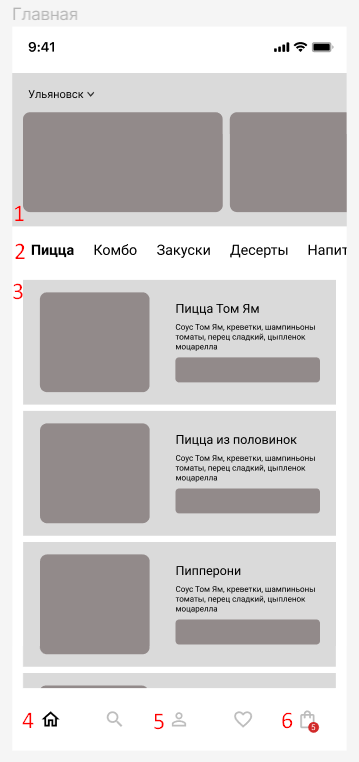


На экране есть кнопки быстрого перехода. Это главная страница, корзина и профиль клиента. Через страницу “Корзина” клиент может оказаться на странице “Оплата заказа”, если тот решит оплатить свой заказ. Совершив покупку, приложение перебрасывает клиента на страницу со статусом заказа. Также на эту страницу можно перейти следующим образом: Профиль->”нажать на конкретный заказ из списка”->Статус заказа. Подробно можно рассмотреть в прототипе приложения.

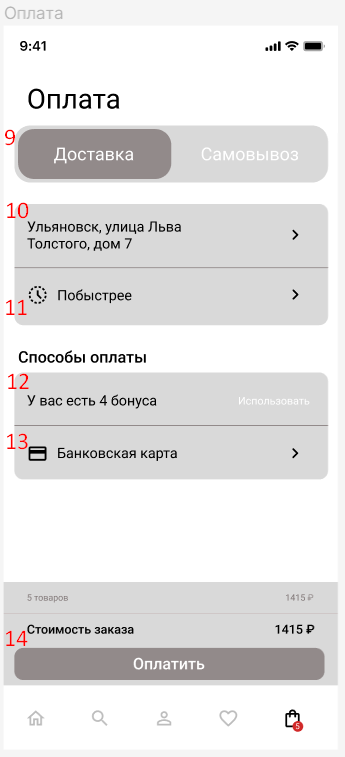
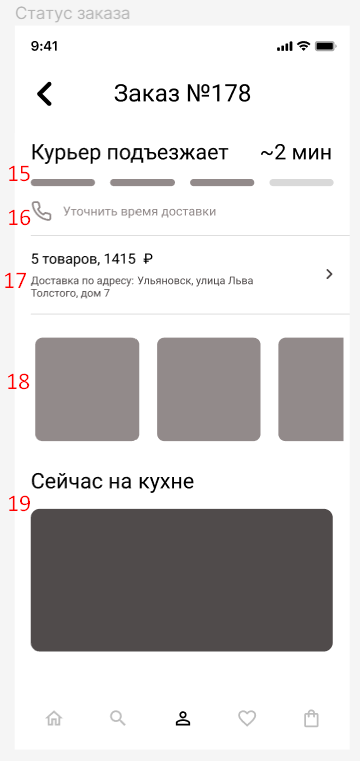
**Кликабельный прототип:**

[https://www.figma.com/file/LeGS6DEaG0JNdbv626cnzE/Заказ-пиццы?node-id=0%3A1](https://www.figma.com/file/LeGS6DEaG0JNdbv626cnzE/%D0%97%D0%B0%D0%BA%D0%B0%D0%B7-%D0%BF%D0%B8%D1%86%D1%86%D1%8B?node-id=0%3A1)

**Элементы макета**

Главная страница Корзина

Оформление заказа Заказ



Личный кабинет

# 

# 

# 

# 

# 

**Описание элементов макета**

|  |  |  |  |
| --- | --- | --- | --- |
| **№ элемента** | **Назначение** | **Его поведение** | **Используемые данные** |
| 1 | Баннеры с рекламой | В этом блоке можно поменять город.  Реклама горизонтально прокручивается.  При смене города, меняются все акции и данные о пиццериях, которые соответствуют и находятся в этом городе. | Данные соответствующие конкретным городам |
| 2 | Перечень позиций меню | Горизонтальная прокрутка.  При нажатии на позицию, меню проматывается на необходимый продукт. | Вид позиции |
| 3 | Меню | Вертикальная прокрутка.  При нажатии на конкретный продукт, система откроет страницу, на которой можно сделать выбор дополнительных параметров и добавить продукт в корзину. | Данные о продукте |
| 4 | Главная страница | Переход на главную страницу | Город, акции, позиции меню и само меню |
| 5 | Личный кабинет | Переход в личный кабинет клиента | Данные клиента и его заказов |
| 6 | Корзина | Перехов в корзину | Данные о продуктах, которые были добавлены в корзину |
| 7 | Продукты, добавленные в корзину | Вертикальная прокрутка. Можно изменить количество или убрать из корзины.. | Данные о продукте |
| 8 | Оформление заказа | Переход на страницу с оформлением заказа | Данные о заказе |
| 9 | Выбор доставки | Правное горизонтальное переключение.  При изменении выбора доставки, меняются ее вид и условия. | Вид и условия доставки |
| 10 | Адрес | Доставка: адрес, куда нужно доставить заказ.  Самовывоз: адрес, откуда нужно забрать заказ.  При нажатии на стрелочку адрес можно изменить. | Геолокация |
| 11 | Условия заказа | При нажатии на стрелочку условия можно изменить.  И в доставке и в самовывозе условия одинаковые. | Условия заказа |
| 12 | Бонусы клиента | Их можно потратить нажав “Использовать”. Если клиент их использует, то из суммы заказа вычитается определенная сумма. | Количество бонусов из личного кабинета |
| 13 | Способ оплаты | Можно изменить способ оплаты, нажав на стрелочку. | Способ оплаты и его условия |
| 14 | Оплата заказа | Нажав на кнопку “Оплатить” сервис открывает страницу, где клиент оплачивает свой заказ. | Способ оплаты |
| 15 | Статус заказа | Каждая полосочка отвечает за свой этап. Когда заказ находится на конкретном этапе, полосочка загружается. Когда этап завершается, полосочка закрашивается полностью. | Статус заказа |
| 16 | Связь с оператором | При нажатии, открывается страница с контактами операторов, по которым можно позвонить и сделать уточнения по заказу | Контакты операторов |
| 17 | Заказ | Показывает сумму заказа и адрес доставки | Данные заказа |
| 18 | Позиции заказа | Перечень позиций заказа. Горизонтальная прокрутка. | Данные заказа |
| 19 | Прямой эфир с кухни | Показывает прямой эфир с кухни заведения. |  |
| 20 | Уведомления | Уведомления, которые пришли пользователю | Данные пользователя или его заказа |
| 21 | Перечень заказов | Представлены все текущие заказы, и те, которые пользователь когда либо делал. При нажатии на заказ, произойдет переход на страницу с его статусом.  Вертикальная прокрутка | Данные пользователя и заказов. |

# 

# 

# 

# 

# 

# 

# **Приложение А. Словарь терминов**

**БД** - база данных

**IOS** - мобильная операционная система для смартфонов, электронных планшетов, носимых проигрывателей, разрабатываемая и выпускаемая американской компанией Apple.

# **Приложение Б. Модели анализа**

**Анкета-опросник**

*Примечание от автора: знаю, что ее здесь быть не должно, но решила, что лучше вставить сюда, чем скидывать отдельным документом.*

|  |  |
| --- | --- |
| **Вопрос** | **Подчеркнуть нужное** |
| Реализация приложения | Web-приложение  Мобильное приложение  Десктопное приложение |
| Какие сервисы приложения обязательно необходимы | Оплата онлайн  Поиск  Отзывы  Корзина  Личный кабинет  Оценка\рейтинг продукта  Сбор статистики  Чат с онлайн-ассистентом |
| Языковые версии приложения | Русский  Русский + иностранный |
| Какой компанией планируется доставка? | Собственная доставка  Доставка смежных компаний |
| Наполненность приложения | Одностраничный  Многостраничный  Без анимации  Анимация  Спокойная цветовая гамма  Яркая цветовая гамма  Цветовая гамма компании |
| Принцип работы заведения | Только доставка  Ресторан/кафе + доставка |
| Количество заведений | Одно заведение  Сеть заведений |
| Способ приема платежей | Банковские карты  Электронные деньги  Наличные деньги  Деньги со счета мобильного телефона  Платежные системы  Банковские переводы |
| Необходим личный кабинет для курьеров? | Да  Нет |
| Самовывоз заказа | Можно  Нельзя |
| **Вопрос** | **Ответ** |
| Укажите конкурентные преимущества Вашего товара / услуги |  |
| По какому принципу должна рассчитываться стоимость доставки заказа |  |

## **Use case регистрация пользователя**

|  |  |  |
| --- | --- | --- |
| **Актор** | **Роль** | **Описание** |
| Клиент | Пользователь | Человек очень любящий пиццу |
| Мобильное приложение пиццерии | ПО (система) | Разрабатываемая система |

**Сценарий - Регистрация клиента в мобильном приложении пиццерии**

|  |  |
| --- | --- |
| **Описание** | Регистрация клиента. Для осуществления данной процедуры, пользователь не должен быть уже зарегистрирован в системе |
| **Действующие лица** | Пользователь, ПО |
| **Предварительные условия** | Пользователь не зарегистрирован |
| **Выходные условия** | Пользователь зарегистрирован |
| **Триггер** | Пользователь нажал “Регистрация” |

**Основной сценарий**

|  |  |  |
| --- | --- | --- |
| **Шаг** | **Действие** | **Ожидаемый результат** |
| 1 | Пользователь нажал “Регистрация” | Система открывает форму для регистрации |
| 2 | Пользователь вводит свои личные данные |  |
| 3 | Пользователь нажал “Зарегистрироваться” |  |
| 4 | Система проверяет, не был ли данный пользователь зарегистрирован ранее | Не был |
| 5 | Система добавляет данные пользователя в БД |  |
| 6 | Система сообщает пользователю об успешной регистрации | Пользователю доступен личный кабинет |

**Альтернативный сценарий**

|  |  |  |
| --- | --- | --- |
| **Шаг** | **Действие** | **Ожидаемый результат** |
| 1 | Пользователь нажал “Регистрация” | Система открывает форму для регистрации |
| 2 | Пользователь вводит свои личные данные |  |
| 3 | Пользователь нажал “Зарегистрироваться” |  |
| 4 | Система проверяет, не был ли данный пользователь зарегистрирован ранее | Был |
| 5 | Система прерывает процедуру регистрации |  |
| 6 | Система сообщает пользователю о том, что он уже зарегистрирован | Пользователю не доступен личный кабинет |

## **Use case оформление заказа**

|  |  |  |
| --- | --- | --- |
| **Актор** | **Роль** | **Описание** |
| Клиент | Пользователь | Человек очень любящий пиццу |
| Мобильное приложение пиццерии | ПО (система) | Разрабатываемая система |

**Сценарий - Оформление заказа в мобильном приложении пиццерии**

|  |  |
| --- | --- |
| **Описание** | Оформление заказа. Для осуществления данного действия пользователь должен быть зарегистрирован и авторизован. |
| **Действующие лица** | Пользователь, ПО |
| **Предварительные условия** | Пользователь авторизован |
| **Выходные условия** | Заказ оформлен |
| **Триггер** | Пользователь добавил пиццу в корзину |

**Основной сценарий**

|  |  |  |
| --- | --- | --- |
| **Шаг** | **Действие** | **Ожидаемый результат** |
| 1 | Пользователь добавил пиццу в корзину, указал параметры (тесто, дополнительные начинки) | Система поместила товар в корзину |
| 2 | Пользователь нажал “Подтвердить заказ” | Система создаёт сущность “Заказ”, открывает форму с адресом |
| 3 | Пользователь заполняет адрес для доставки. | Система сохраняет адрес |
| 4 | Пользователь оплачивает заказ | Деньги переведены |
| 5 | Система сообщает пользователю об успешном принятии заказа. | Система передает заказ на кухню |

**Альтернативный сценарий**

|  |  |  |
| --- | --- | --- |
| **Шаг** | **Действие** | **Ожидаемый результат** |
| 1 | Пользователь добавил пиццу в корзину, указал параметры (тесто, дополнительные начинки) | Система поместила товар в корзину |
| 2 | Пользователь нажал “Подтвердить заказ” | Система создаёт сущность “Заказ”, открывает форму с адресом |
| 3 | Пользователь заполняет адрес для доставки. | Система сохраняет адрес |
| 4 | Пользователь не оплачивает заказ | Деньги не переведены |
| 5 | Система не оформляет заказ |  |

## **Use case просмотр заказов**

|  |  |  |
| --- | --- | --- |
| **Актор** | **Роль** | **Описание** |
| Клиент | Пользователь | Человек очень любящий пиццу |
| Мобильное приложение пиццерии | ПО (система) | Разрабатываемая система |

**Сценарий - Просмотр заказов в мобильном приложении пиццерии**

|  |  |
| --- | --- |
| **Описание** | Просмотр заказов. Для осуществления данного действия пользователь должен быть зарегистрирован и авторизован. |
| **Действующие лица** | Пользователь, ПО |
| **Предварительные условия** | Пользователь авторизован |
| **Выходные условия** | Предоставлен список заказов |
| **Триггер** | Пользователь перешел в личный кабинет |

**Основной сценарий**

|  |  |  |
| --- | --- | --- |
| **Шаг** | **Действие** | **Ожидаемый результат** |
| 1 | Пользователь перешел в личный кабинет | Система открывает страницу личного кабинета |
| 2 | Система считывает ID пользователя | ID пользователя определен |
| 3 | Система ищет по ID все заказы данного пользователя | Заказы найдены |
| 4 | Система предоставляет список всех заказов данного пользователя |  |

**Альтернативный сценарий**

|  |  |  |
| --- | --- | --- |
| **Шаг** | **Действие** | **Ожидаемый результат** |
| 1 | Пользователь перешел в личный кабинет | Система открывает страницу личного кабинета |
| 2 | Система считывает ID пользователя | ID пользователя определен |
| 3 | Система ищет по ID все заказы данного пользователя | Заказы не найдены |
| 4 | Система сообщает, что заказов нет |  |