# Информация о проекте

Интернет-магазин музыкальных инструментов

# User stories

## Покупка е2е. Критичный сценарий

Я как клиент

Хочу авторизоваться

Тогда я смогу пользоваться системой

Я как клиент

Хочу получить список товаров

Тогда я смогу найти нужный мне

Я как клиент

Хочу искать товары по параметрам

Тогда я смогу найти нужный мне быстрее

Я как клиент

Хочу посмотреть детальное описание товара

Тогда я смогу найти нужный мне

Я как клиент

Хочу добавить/удалить товар в корзину

Тогда я смогу сформировать заказ

Я как клиент

Хочу получить список товаров в корзине

Тогда я смогу его проанализировать

Я как клиент

Хочу получить варианты доставки

Тогда я смогу выбрать подходящий

Я как клиент

Хочу указать адрес доставки

Тогда я смогу указать удобный для меня адрес

Я как клиент

Хочу оформить заказ

Тогда я смогу получить товар

Я как клиент

Хочу оплатить заказ  
Тогда мой заказ будет окончательно оформлен

Я как клиент

Хочу получить подтверждение заказа на email

Тогда я буду знать, что мой заказ оформлен

## Просмотр и изменение данных профиля

Я как клиент

Хочу иметь возможность поменять данные своего профиля (кроме логина и пароля)

Я как клиент

Хочу иметь возможность посмотреть данные своего профиля

Я как клиент

Хочу иметь возможность посмотреть историю заказов и их статус

## Добавление нового продукта администратором

Я как администратор  
Хочу иметь возможность создать новый товар

Я как клиент  
Хочу иметь возможность купить новый продукт

## Просмотр и изменение заявок на доставку

Я как курьер  
Хочу посмотреть список всех прикрепленных ко мне заявок на доставку

Я как курьер  
Хочу изменить статус заявки на В процессе

Я как курьер  
Хочу изменить статус заявки на Выполнена

Я как клиент   
Хочу получать оповещения о смене статуса заказа

## ~~Просмотр клиентов и заказов CSR?~~

# Мокапы

Где рисовать?

# Системные действия

## Покупка е2е. Критичный сценарий

Клиент авторизовывается

Клиент получает список товаров

Клиент получает товары по параметрам

Клиент получает детальное описание товара

Клиент добавляет/удаляет товар в корзину

Клиент получает список товаров в корзине

Клиент получает варианты доставки

Клиент выбирает вариант доставки

Клиент указывает адрес доставки

Клиент оформляет заказ

Клиент оплачивает заказ

Приложение резервирует товар на складе

Приложение отправляет подтверждение заказа на email Клиента

Приложение передает заказ в службу доставки

## Просмотр и изменение данных профиля

Клиент авторизовывается

Клиент получает данные о своем профиле  
Клиент получает данные о своих заказах  
Клиент меняет данные своего профиля

## Добавление нового продукта администратором

Администратор авторизовывается  
Администратор добавляет новый продукт  
Клиент авторизовывается

Клиент находит новый продукт  
Клиент добавляет новый продукт в корзину

## Просмотр и изменение заявок на доставку

Курьер авторизовывается  
Курьер получает список прикрепленных к нему заявок  
Курьер меняет статус заявки  
Клиент получает оповещение на email

# Модель на основе ООП

**Основные сущности:**

Пользователь (Клиент, Администратор, Курьер, Профиль пользователя)

Товар

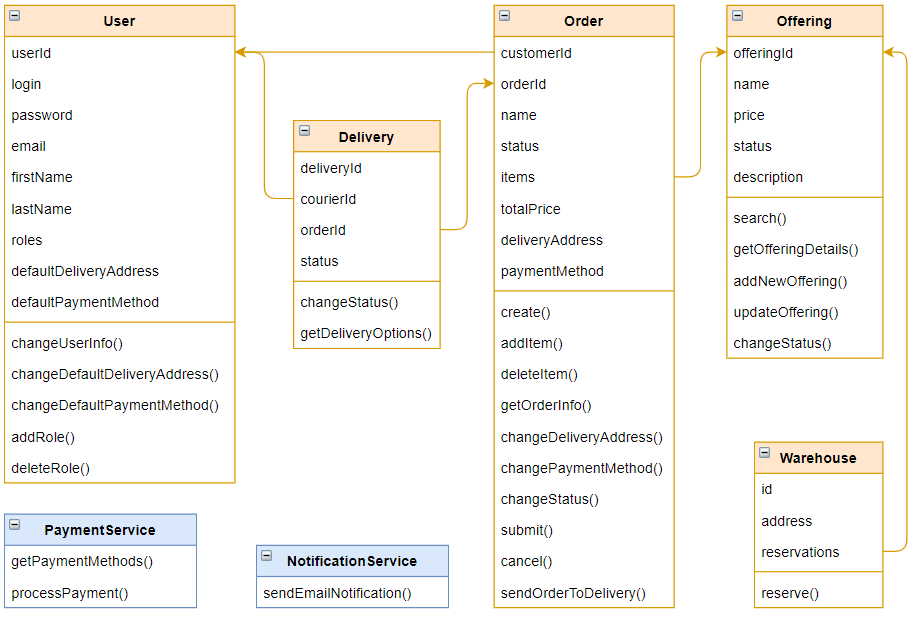
Заказ (Корзина)

Служба доставки (Доставка, Заявка на доставку)

Склад

Платежный сервис (Оплата)

Оповещение (на email)



**Сервисы:**

UserManagement

OrderManagement

OfferingCatalog

WarehouseManagement

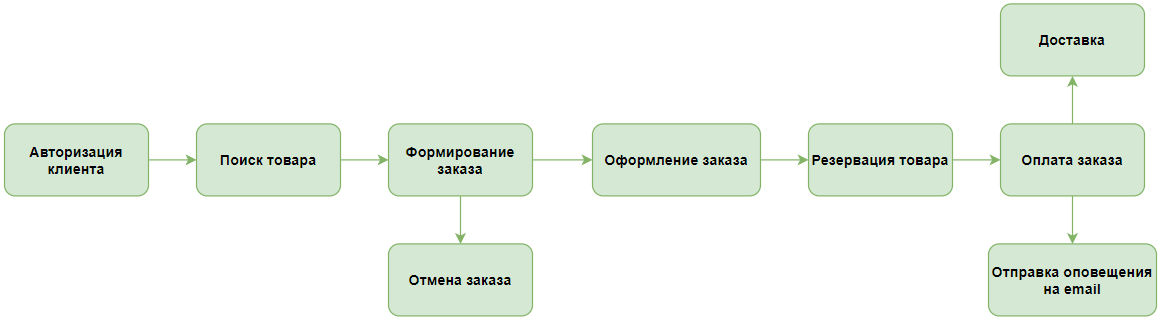
DeliveryService

PaymentService

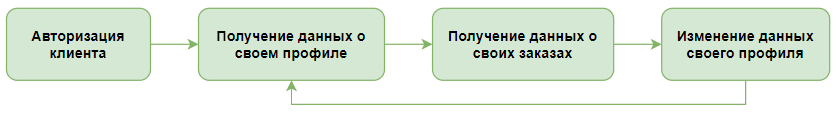
NotificationService

# Функциональная модель

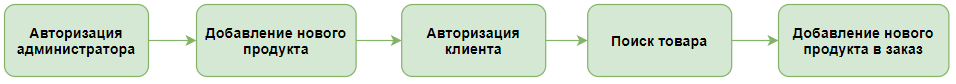
Покупка е2е



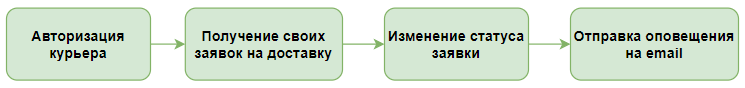
Просмотр и изменение данных профиля пользователя



Добавление нового продукта



Просмотр и изменение заявок на доставку



# Сервисы

Дальше работаем с моделью на основе ООП

**Сервисы:**

UserManagement

OrderManagement

OfferingCatalog

WarehouseManagement

DeliveryService

PaymentService

NotificationService

# Системные действия в виде API

**Пользователь**

POST /api/v1/users

GET /api/v1/users/{id}/details

PUT /api/v1/users/{id}/details

**Заказ**

GET /api/v1/orders/{id}

GET /api/v1/orders?userId=123

POST /api/v1/orders

POST /api/v1/orders{id}/changeStatus

POST /api/v1/orders/{id}/items

DELETE /api/v1/orders/{id}/items

PUT /api/v1/orders/{id}

**Товар**

GET /api/v1/offerings/search?query=OfferingName123

GET /api/v1/offerings/{id}

POST /api/v1/offerings

PUT /api/v1/offerings/{id}

POST /api/v1/offerings/{id}/changeStatus

**Склад**

POST /api/v1/reserve

**Оплата**

GET /api/v1/paymentMethods

POST /api/v1/pay

**Доставка**

POST /api/v1/deliveryTasks

GET /api/v1/deliveryTasks/{id}

POST /api/v1/deliveryTasks/{id}/changeStatus

GET /api/v1/deliveryOptions

GET /api/v1/deliveryTasks?courierId=123

**Оповещение**

POST /api/v1/notification

# AuthenticationService

TODO

# UserManagement

Запросы:

|  |  |  |
| --- | --- | --- |
|  | Request | Response |
| Получить информацию о пользователе | GET  /api/v1/users/{userId}/details | {  "id":"123",  "email":"alex@gmail.com",  "firstName":"Alex",  "lastName":"Xela",  "defaultAddress":"Voronezh, Lenina 1",  "defaultPaymentMethod":"Credit Card"  } |

Команды:

|  |  |  |
| --- | --- | --- |
|  | Request | Response |
| Создать нового пользователя | POST  /api/v1/users | todo |
| Изменить информацию о пользователе (  email  firstName  lastName  roles  defaultDeliveryAddress  defaultPaymentMethod  ) | PUT  /api/v1/users/{userId}/details | ToDO |

События:

Зависимости:

Вопросы:

# OrderManagement

Запросы:

|  |  |  |
| --- | --- | --- |
|  | Request | Response |
| Получить информацию о заказе | GET  /api/v1/orders/{orderId} | {  "id":"123",  "name":"Order#123",  "totalPrice":555.22,  "status":"In progress",  "deliveryAddress":"Voronezh, Lenina 1",  "paymentMethod": "CreditCard",  "items":[  {  "id":"123",  "offeringId":"123",  "price":222.22  }  ]  } |
| Получить все заказы определенного клиента | GET  /api/v1/orders?userId={userId} | ToDo |

Команды:

|  |  |  |
| --- | --- | --- |
|  | Request | Response |
| Создать заказ | POST  /api/v1/orders | {  "id":"123",  "name":"Order#123",  "totalPrice":0,  "status":"In progress",  "deliveryAddress":"Voronezh, Lenina 1",  "paymentMethod": " Credit Card ",  "items":[]  } |
| Добавить товар | POST  /api/v1/orders/{orderId}/items  {  "offeringId":"123"  } | {  ...,  "items":[  {  "id":"123",  "offeringId":"123",  "price":222.22  }  ]  } |
| Удалить товар | DELETE  /api/v1/orders/{orderId}/items/{itemId} |  |
| Обновить адрес доставки | PUT  /api/v1/orders/{orderId}  {  "deliveryAddress":"Voronezh, Lenina 1"  } | {  "id":"123",  "name":"Order#123",  "totalPrice":0,  "status":"In progress",  "deliveryAddress":"Voronezh, Lenina 1",  "paymentMethod": " Credit Card ",  "items":[]  } |
| Обновить метод оплаты | PUT  /api/v1/orders/{orderId}  {  "paymentMethod": "Credit Card"  } | {  "id":"123",  "name":"Order#123",  "totalPrice":0,  "status":"In progress",  "deliveryAddress":"Voronezh, Lenina 1",  "paymentMethod": " Credit Card ",  "items":[]  } |
| Обновить статус заказа | POST  /api/v1/orders/{orderId}/changeStatus  {  "status":"Submitted"  } | {  "id":"123",  "name":"Order#123",  "totalPrice":0,  "status":"Submitted ",  "deliveryAddress":"Voronezh, Lenina 1",  "paymentMethod": " Credit Card ",  "items":[]  } |

События:

|  |
| --- |
| Смена статуса заказа: Оформление, Отмена |

Зависимости:

|  |  |
| --- | --- |
| ~~UserManagement~~ | ~~Получение адреса доставки и метода оплаты по умолчанию~~ |
| OfferingCatalog | Поучение деталей о товаре. Цена товара |
|  |  |

Вопросы:

|  |
| --- |
| ~~Добавить более явные методы для смены статуса заказа?~~ |
| ~~Нужно ли ходить отдельно за дефолтными значениями в юзер менеджмент? – Нет. Пусть приходят при инициализации ордера~~ |
| ~~Если заказ бросили? Удалять? – Нет. Пусть хранится история пока что~~ |

# OfferingCatalog

Запросы:

|  |  |  |
| --- | --- | --- |
|  | Request | Response |
| Получить товар | GET  /api/v1/offerings/{offeringId} | {  "id":"123",  "name":"Offering Name",  "price":222.22,  "description":"Some Text",  “available”: 12  } |
| Поиск товара по параметрам | GET  /api/v1/offerings/search  ?query=OfferingName123 | [  {  "id":"123",  "name":"Offering Name",  "price":222.22,  "description":"Some Text"  },  {  ...  }  ] |

Команды:

|  |  |  |
| --- | --- | --- |
|  | Request | Response |
|  | POST /api/v1/offerings | TodO |
|  | PUT /api/v1/offerings/{offeringId} | TodO |
|  | POST /api/v1/offerings/{offeringId}/changeStatus | ToDo |

События:

Зависимости:

|  |  |
| --- | --- |
| WarehouseManagement | Слушает события на изменения количества доступных товаров |

Вопросы:

|  |
| --- |
| Поиск через эластик? |
| ~~Проверять наличие товара на складе?~~ |

# WarehouseManagement

Запросы:

Команды:

События:

|  |
| --- |
| Изменение количества доступных товаров |

Зависимости:

|  |  |
| --- | --- |
| OrderManagement | Слушает события изменения статуса заказа. Во время оформления заказа резервирует товар на складе |

Вопросы:

# PaymentService

Запросы:

|  |  |  |
| --- | --- | --- |
|  | Request | Response |
| Получение доступных методов оплаты | GET  /api/v1/paymentMethods | [  {  "id":"123",  "name":"Credit Card",  "description":"Some Text"  }  ] |

Команды:

События:

|  |
| --- |
| Заказ оплачен |

Зависимости:

Вопросы:

|  |
| --- |
| ~~Как именно выглядит способ оплаты? Это типы? Кредитка? Что и где хранится? -Типы~~ |

# DeliveryService

Запросы:

|  |  |  |
| --- | --- | --- |
|  | Request | Response |
| Получение доступных методов доставки | GET  /api/v1/deliveryOptions | [  {  "id":"123",  "name":"Courier",  "description":"Some Text"  }  ] |
|  | GET  /api/v1/deliveryTasks/{taskId} | ToDo |
|  | GET /api/v1/deliveryTasks?courierId={ courierId } | ToDo |

Команды:

|  |  |  |
| --- | --- | --- |
|  | Request | Response |
|  | POST  /api/v1/deliveryTasks/{id}/changeStatus | ToDo |

События:

Зависимости:

|  |  |
| --- | --- |
| ~~PaymentService~~ | ~~Слушает статус оплаты. После оплаты передает заказ в доставку.~~ |
| OrderManagement | Получает детальную информацию по заказу. Слушает изменение статуса заказа. После оплаты передает заказ в доставку |
| UserManagement | Получает детальную информацию по заказчику |

Вопросы:

|  |
| --- |
| ~~Какие типы доставки? Надо ли определить список? – Пока что только доставка домой курьером. В теории: Самовывоз~~ |

# NotificationService

Запросы:

Команды:

События:

Зависимости:

|  |  |
| --- | --- |
| ~~PaymentService~~ | ~~Слушает изменение статуса заказа. Когда оплачен – отправляет оповещение~~ |
| OrderManagement | Получает детальную информацию по заказу Слушает изменение статуса заказа. Когда оплачен – отправляет оповещение |
| UserManagement | Получает детальную информацию по заказчику. Как минимум его email |

Вопросы:

# Схема взаимодействия сервисов

