Описание API

**1. API Регистрация / Авторизация**

МС решает задачи:

а) Создания нового пользователя

б) Вход по логину и паролю

в) Выход из системы

г) Обновление токена

2. API Уведомления решает задачи:

а) Уведомления о статусах заказов

б) Платёжные уведомления

в) Системные уведомления (Например, уведомления о блокировке учётной записи)

3. API Заказы решает задачи:

а) Создание заказа

б) Получение администратором всех заказов конкретного пользователя

в) Получение администратором конкретного заказа

г) Обновление администратором заказа

д) Изменение статуса и отмена заказа

4. API Платежи решает задачи:

а) Проведение оплаты заказа

б) Получение администратором информации о статусе платежа

в) Получение истории платежей конкретного клиента

**API Регистрация / Авторизация**

**Метод POST/auth/register (Регистрация нового пользователя)**

Request:

{

"email": "user@mail.ru",

"password": "0000string",

"fname": "Петр Петров",

"phone": "89110005556"

}

Response:

200 ok: { "status": "success", "message": "User created successfully" }

401 unathorized { "status": "error", "message": "Invalid email or password" }

403 conflict { "status": "error", "message": "Duplicate user for email user@mail.ru"}

**Метод POST/auth/login (Авторизация пользователя)**

Request:

{

“email”: “[user@mail.ru](mailto:user@mail.ru)”,

“password”: “0000string”

}

Response:

200 ok: {

"status": "success",

"access\_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6...",

"refresh\_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6..."

}

401 unathorized {

"status": "error",

"message": "Invalid email or password"

}

403 Forbidden {

"status": "error",

"message": "Account blocked"

}

Метод POST/auth/refresh (Обновление токена)

Request:

{

"refresh\_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6..."

}

Response:

200 OK: {

"status": "success",

"access\_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6...",

"expires\_in": 3600

}

401 Unauthorized: { "status": "error", "message": "Invalid or expired refresh token" }

**Метод POST/auth/logout (Выход из системы)**

Request:

{

"refresh\_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6..."

}

Response:

200 OK: { "status": "success", "message": "Logout successful" }

400 Bad Request: { "status": "error", "message": "Invalid token or user already logged out" }

**API Уведомления**

Метод POST/notification/send (Отправка уведомления администратором)

Request:

{

"client\_id": "11",

"title": "Заявка №1 обновлена",

"message": "Ваш заказ доставлен",

"type": "status"

}

Response:

200 ok: { "status": "success", "message": "Notification sent" }

Метод GET/notification/user/{id} (Получение администратором всех уведомлений конкретного пользователя)

Response:

[

{

"id": "1",

"title": "Платёж получен",

"message": "Оплата №102 успешно обработана",

"type": "payment",

"created\_at": "13.10.2025-13:45"

},

{

"id": "2",

"title": "Заявка №1 обновлена",

"message": "Ваш заказ доставлен",

"type": "status",

"created\_at": "13.10.2025-15:00"

} }

**API Платежи**

а) Проведение оплаты заказа

б) Получение администратором информации о статусе платежа

в) Получение истории платежей конкретного клиента

Метод POST/payments (Оплата заказа)

Request {

“amount”: 1250.0,

“payment\_method”: “card”,

“date”: “15.10.2025T12:34:56”

“card\_number”: “4111 1111 1111 1111"

}

Response:

200 ok: { “status”: “success”, “message”: ”payment completed successfully” }

400 Bad Request: { “status”: “error”, “message”: “incorrect payment data” }

402 Payment required: { “status”: “failed”, “message”: “Not enough money” }

500 server error: { “status”: “error”, “message”: “Error processing payment” }

Метод GET/payments/{id\_pay}/status (Получение администратором информации о статусе платежа “pending”, “success”, “failed”)

Response {

“id\_pay”: “001”,

“amount”: “1250.0”,

“payment\_method”: “card”,

“date”: “15.10.2025T12:34:56”,

“card\_number”: “\*\*\*\* \*\*\*\* \*\*\*\* 1111”,

“status”: “success”

}

Метод GET/client/{id}/payments (Получение истории платежей конкретного клиента)

200 OK

Response {

“status”: “success”,

“message”: “Payments history recieved”,

“payments”: [

{

“id\_pay”: “001”,

“amount”: “1250.0”,

“payment\_method”: “card”,

“date”: “15.10.2025T12:34:56”,

“card\_number”: “\*\*\*\* \*\*\*\* \*\*\*\* 1111”,

“status”: “success”

},

{

“id\_pay”: “002”,

“amount”: “980.0”,

“payment\_method”: “card”,

“date”: “03.10.2025T15:27:56”,

“card\_number”: “\*\*\*\* \*\*\*\* \*\*\*\* 1111”,

“status”: “failed”

}

]

}

**API Заказы**

Метод POST/requests (Создание заказа)

Request {

"req\_id": "42abcdfdf-fd98-4hda-fhk-0bcsvse",

"name": "Zakaz",

"status": "created",

"created\_at": "2022-07-01T15:00:00"

}

Response {

"req\_id": "42abcdfdf-fd98-4hda-fhk-0bcsvse",

"name": "Zakaz",

"status": "created",

"created\_at": "2022-07-01T15:00:00"

}

Метод GET/requests/user/{id} (Получение всех заказов конкретного пользователя)

Response {

{

"status": "success",

"message": "Client’s requests list is recieved",

"requests": [

{

"req\_id": "001",

“id\_pay”: “001”,

“date”: “15.10.2025T14:26:12”,

"address": "г. Владивосток, ул. Пограничная, 15",

"status": "delivered"

},

{

“req\_id”: “002”,

“id\_pay”: “002”,

“date”: “13.10.2025T19:18:44”,

“status”: “created”;

“address”: “г. Владивосток, ул. Державина, 21”

}

]

}

**Метод GET/requests/{id}/user/{id} (Получение данных конкретного заказа пользователя)**

Response:

{

"req\_id": "002",

"id\_pay": "002",

"date": "2025-10-13T19:18:44",

"address": "г. Владивосток, ул. Окатовая 25",

"status": "created"

}

Метод PUT/requests/{id} (Обновление заказа)

Request:

{

"address": "г. Владивосток, ул. Ленинская, 12",

"delivery\_date": "2025-10-17T14:00:00"

}

Response:

{

"status": "success",

"message": "Request updated successfully"

}

**Метод PATCH/requests/{id}/status (Изменение статуса и отмена заказа ("created", "approved", "being\_delivered", "delivered", "canceled"))**

Request:

{

"status": "canceled"

}

Response:

{

"status": "success",

"message": "Request status updated to 'canceled'"

}