# API Overview:

- Основна частина посилань:

<https://api-dev-selina.d.yeducoders.com/api>

- Формат відповідей:

{

"message": "",

"body": { …},

"isError": false | true

}

Message – це повідомлення, що описує суть відповіді. Відображає опис помилок, або інформацію про успіх виконання

Body – дані, які повертає запит до апі

isError – вказує, чи було виконано запит успішно

- Формування запиту:  
Документація поділена на модулі, на початку кожного модуля, вказано його шлях. Цей шлях потрібно додавати до основного посилання, а потім, додавати ще шлях, що вказаний біля конкретного запиту.

Приклад:

Модуль: Auth;

Endpoint: /auth;

Запит: Registrer;

Endpoint: /register;

Повний шлях запиту: https://api-dev-selina.d.yeducoders.com/api/auth/register

# Authentication:

Тип: Bearer Token

Щоб отримати доступ до API, що мають певні обмеження залежно від ролі користувача, або його авторизованості на сайті, ви повинні включити заголовок авторизації разом із токеном:

Authorization: Bearer YOUR\_ACCESS\_TOKEN

# Unique types:

- email – параметр email є обов'язковим і не перевищує 255 символів. Також параметр має формат електронної пошти.

- password – параметр має довжину від 8 до 255 символів.

**Endpoints:**

# AUTH

**Endpoint:** /auth

**Description:** Використовується користувачем по відношенню до свого акаунта.

## Register

**Endpoint:** /register

**Method:** POST

**Description:** реєструє користувача на платформі

**Request Headers:**

Content-Type: application/json

**Request Body:**

{

"name": "string",

"email": "email",

"password": "password",

"acceptedTermCondition": "true",

}

**Response body-field:** {}

**Status Code:** 201 Created

**Errors:**

- 400 Bad Request - недійсні дані

- 409 Conflict - пошта була зареєстрована раніше

- 403 Forbidden - авторизований користувач намагається зареєструвати новий обліковий запис

## Verify Email

**Endpoint:** /verify-email

**Method:** POST

**Description:** підтвердження електронної пошти користувача за допомогою токену

**Request Headers:**

Content-Type: application/json

**Request Body:**

{

"email": "email",

"token": "string"

}

**Response body-field:** {}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - невірний токен або електронна пошта

- 404 Not Found - користувач не знайдений

## Login

**Endpoint:** /login

**Method:** POST

**Description:** аутентифікація користувача за допомогою його електронної пошти та паролю

**Request Headers:**

Content-Type: application/json

**Request Body:**

{

"email": "email",

"password": "string",

"rememberMe": "boolean" // не обов'язкове поле

}

**Response body-field:**

1) Якщо аутентифікація успішна і двофакторна аутентифікація не потрібна:

{

"user": {

"id": "string",

"name": "string",

"email": "email",

// решта полів користувача

},

"authToken": "string",

"userId": "string",

"needCode": false,

"canSendCodeByPhone": "boolean",

"needRegularViewInfoForm": "boolean"

}

2) Якщо потрібна двофакторна аутентифікація, але телефон підтверджено:

{

"user": {

"email": "email",

"phone": "string",

"id": "string",

"name": "string"

},

"needCode": true,

"codeSent": false

}

3) Якщо потрібна двофакторна аутентифікація і телефон не підтверджено:

{

"user": {

"email": "email",

"phone": "string",

"id": "string",

"name": "string"

},

"needCode": true,

"codeSent": true

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - Невірні дані запиту

- 401 Unauthorized - Неправильний email або пароль

- 403 Forbidden - Акаунт не підтверджено або заблоковано

## Two-Factor Authentication Code Generation

**Endpoint:** /generate-two-factor-code

**Method:** POST

**Description:** генерація коду двофакторної аутентифікації для користувача

**Request Headers:**

Content-Type: application/json

**Request Body:**

{

"email": "email",

"password": "password",

"type": "string", // "phone" або "email"

}

**Response body-field:** {}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - невірний токен або електронна пошта

- 404 Not Found - користувач не знайдений

## Verify Two-Factor Authentication Code

**Endpoint:** / check-two-factor-code

**Method:** POST

**Description:** перевірка коду двофакторної аутентифікації для користувача

**Request Headers:**

Content-Type: application/json

**Request Body:**

{

"code": "string",

"type": "string", // "phone" або "email"

"id": "string",

"rememberMe": "boolean" // не обов'язкове поле

}

**Response body-field:**

{

"user": {

"id": "string",

"name": "string",

"email": "email",

// додаткові поля користувача

},

"authToken": "string",

"userId": "string",

"needRegularViewInfoForm": "boolean"

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - код не дійсний або невірний

- 401 Unauthorized - неправильний email або пароль

## My Info

**Endpoint:** /my-info

**Method:** POST

**Description:** отримання інформації про користувача

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:** {}

**Response body-field:**

{

"user": {

"id": "string",

"name": "string",

"email": "email",

"phone": "string",

// додаткові поля користувача

}

}

**Status Code:** 200 OK

**Errors:**

- 401 Unauthorized - відсутній або недійсний токен доступу

## Save Profile

**Endpoint:** /save-profile

**Method:** POST

**Description:** збереження інформації про користувача

**Request Headers:**

Content-Type: multipart/form-data

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"name": "string",

"phone": "string",

"contactDetails": "string",

"briefBio": "string",

"placeWork": "string",

"facebookUrl": "string",

"instagramUrl": "string",

"linkedinUrl": "string",

"twitterUrl": "string",

"photo": "file" // обов'язкове поле, якщо є файл

}

**Response body-field:**

{

"user": {

"id": "string",

"name": "string",

"email": "email",

"phone": "string",

// додаткові поля користувача

}

}

**Status Code:** 200 OK

**Errors:**

- 401 Unauthorized - відсутній або недійсний токен доступу

## Update My Password

**Endpoint:** /update-my-password

**Method:** POST

**Description:** зміна пароля користувача

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"currentPassword": "string",

"newPassword": "string"

}

**Response body-field:** {}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - невірні дані запиту

- 401 Unauthorized - неправильний поточний пароль

## Save My Documents

**Endpoint:** /save-my-documents

**Method:** POST

**Description:** збереження документів користувача

**Request Headers:**

Content-Type: multipart/form-data

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

- userPhoto: файл зображення профілю (не обов'язкове поле)

- documentFront: файл передньої сторони документа (не обов'язкове поле)

- documentBack: файл задньої сторони документа (не обов'язкове поле)

**Response body-field:**

{

"documents": {

"userPhoto": "string",

"documentFront": "string",

"documentBack": "string"

}

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - невірні дані запиту або файл перевищує допустимий розмір

- 401 Unauthorized - відсутній або недійсний токен доступу

## Reset Password Send

**Endpoint:** /reset-password-send

**Method:** POST

**Description:** надсилає електронного листа для скидання пароля користувача

**Request Headers:**

Content-Type: application/json

**Request Body:**

{

"email": "email"

}

**Response body-field:** {}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - немає користувача з вказаною електронною поштою

- 400 Bad Request - користувач може увійти в обліковий запис тільки через Google або Facebook

- 401 Unauthorized - відсутній або недійсний токен доступу

## Reset Password

**Endpoint:** /reset-password

**Method:** POST

**Description:** оновлює пароль користувача за допомогою токена скидання пароля

**Request Headers:**

Content-Type: application/json

**Request Body:**

{

"token": "string",

"password": "password"

}

**Response body-field:** {}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - токен недійсний

- 400 Bad Request - новий пароль не може бути таким же, як поточний

## Generate My Phone Code

**Endpoint:** /generate-my-phone-code

**Method:** POST

**Description:** генерує код для перевірки телефону користувача та надсилає його на збережений номер телефону

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:** {}

**Response body-field:** {}

**Status Code:** 200 OK

**Errors:**

- 404 Not Found - не знайдено користувача

- 400 Bad Request - у вашому обліковому записі не збережено номер телефону

- 401 Unauthorized - відсутній або недійсний токен доступу

## Generate Email Code

**Endpoint:** /generate-my-email-code

**Method:** POST

**Description:** генерує код для перевірки електронної пошти користувача та надсилає його на вказану електронну пошту

**Request Headers:**

Content-Type: application/json

**Request Body:**

{

"email": "email"

}

**Response body-field:** {}

**Status Code:** 200 OK

**Errors:**

- 409 Conflict - електронна пошта не знайдена

## Check My Phone Code

**Endpoint:** /check-my-phone-code

**Method:** POST

**Description:** перевіряє код підтвердження телефону та активує номер телефону користувача

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"code": "string"

}

**Response body-field:** {}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - недійсний код OTP

- 401 Unauthorized - відсутній або недійсний токен доступу

## Change Two-Factor Auth

**Endpoint:** /change-two-factor-auth

**Method:** POST

**Description:** активує або деактивує двофакторну аутентифікацію для користувача

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"id": "string",

"twoFactorAuthentication": true | false

}

**Response body-field:** {}

**Status Code:** 200 OK

**Errors:**

- 401 Unauthorized - відсутній або недійсний токен доступу

## Auth By Provider

**Endpoint:** /auth-by-provider

**Method:** POST

**Description:** авторизація користувача через сторонні сервіси (Google або Facebook)

**Request Headers:**

Content-Type: application/json

**Request Body:**

{

"name": "string" //ім'я користувача від провайдера,

"email": "email" //електронна пошта користувача від провайдера,

"token": "string" //токен доступу від провайдера (Google або Facebook),

"provider": "string" //назва провайдера (Google або Facebook)

}

**Response body-field:**

{

"needRegularViewInfoForm": "boolean",

"authToken": "string",

"userId": "string"

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - недійсний токен

- 401 Unauthorized - ваша обліковка заблокована. Для отримання додаткової -інформації зверніться до адміністратора

- 404 Not Found - не знайдено користувача з наданою електронною поштою

# LISTINGS

**Endpoint:** /listings

**Description:** взаємодія з інструментами, як власними, так і чужими

## List

**Endpoint:** /list

**Method:** POST

**Description:** повертає список лістингів на основі фільтрів та локації користувача

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN – не обов’язковий

**Request Body:**

{

"filter": "string", // необов'язкове поле

"itemsPerPage": "number", // необов'язкове поле, за замовчуванням 20

"order": "string", // необов'язкове поле

"orderType": "string", // необов'язкове поле, за замовчуванням "desc"

"page": "number", // необов'язкове поле, за замовчуванням 1

"cities": ["string"], // необов'язкове поле

"categories": ["string"], // необов'язкове поле

"distance": "number", // необов'язкове поле

"minPrice": "number", // необов'язкове поле

"maxPrice": "number", // необов'язкове поле

"lat": "number", // необов'язкове поле

"lng": "number", // необов'язкове поле

"othersCategories": "boolean", // необов'язкове поле

"searchCity": "string", // необов'язкове поле

"searchCategory": "string", // необов'язкове поле

"searchListing": "string", // необов'язкове поле

"clientTime": "string", // необов'язкове поле

"fromTime": "string", // необов'язкове поле

"toTime": "string" // необов'язкове поле

}

**Response body-field:**

{

"items": [/\* список інструментів \*/],

"options": {/\* дані, на основі яких було зроблено пошук \*/},

"countItems": "number",

"canSendCreateNotifyRequest": "boolean"

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - невірні дані запиту або невірний формат параметрів

## List By Owner

**Endpoint:** /owner-list

**Method:** POST

**Description:** повертає список лістингів, що належать певному власнику, на основі фільтрів та локації користувача

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN – не обов’язковий

**Request Body:**

{

"ownerId": "integer",

"filter": "string", // необов'язкове поле

"itemsPerPage": "number", // необов'язкове поле, за замовчуванням 20

"order": "string", // необов'язкове поле

"orderType": "string", // необов'язкове поле, за замовчуванням "desc"

"page": "number", // необов'язкове поле, за замовчуванням 1

"cities": ["string"], // необов'язкове поле

"categories": ["string"], // необов'язкове поле

"distance": "number", // необов'язкове поле

"minPrice": "number", // необов'язкове поле

"maxPrice": "number", // необов'язкове поле

"lat": "number", // необов'язкове поле

"lng": "number", // необов'язкове поле

"othersCategories": "boolean", // необов'язкове поле

"searchCity": "string", // необов'язкове поле

"searchCategory": "string", // необов'язкове поле

"searchListing": "string", // необов'язкове поле

"clientTime": "string", // необов'язкове поле

"fromTime": "string", // необов'язкове поле

"toTime": "string" // необов'язкове поле

}

**Response body-field:**

{

"items": [/\* список інструментів \*/],

"options": {/\* дані, на основі яких було зроблено пошук \*/},

"countItems": "number",

"canSendCreateNotifyRequest": "boolean"

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - невірні дані запиту або невірний формат параметрів

## Session User List

**Endpoint:** /user-list

**Method:** POST

**Description:** повертає список інструментів поточного користувача з можливістю фільтрації за статусом

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"status": "string", // необов'язкове поле, за замовчуванням "all"

"filter": "string", // необов'язкове поле

"itemsPerPage": "number", // необов'язкове поле, за замовчуванням 20

"orderType": "string" // необов'язкове поле, значення "desc" або "asc"

}

**Response body-field:**

{

"items": [/\* список інструментів \*/],

"options": {/\* дані, на основі яких було зроблено пошук \*/},

"countItems": "number",

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - невірні дані запиту або невірний формат параметрів

## Creating New Listing

**Endpoint:** /create

**Method:** POST

**Description:** створення нового лістингу

**Request Headers:**

Content-Type: multipart/form-data

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"address": "string",

"name": "string",

"categoryId": "number", // необов'язкове поле, за замовчуванням null

"compensationCost": "number",

"countStoredItems": "number",

"description": "string", // необов'язкове поле, за замовчуванням ""

"postcode": "string",

"pricePerDay": "number",

"rentalLat": "number",

"rentalLng": "number",

"rentalRadius": "number",

"keyWords": "string", // необов'язкове поле, за замовчуванням ""

"minRentalDays": "number", // необов'язкове поле, за замовчуванням null

"city": "string",

"otherCategory": "string", // необов'язкове поле, за замовчуванням null

"backgroundPhoto": "file",

"listingImages": ["file"]

}

**Response body-field:**

{

"listing": {

"address": "string",

"name": "string",

"categoryId": "number",

"compensationCost": "number",

"countStoredItems": "number",

"description": "string",

"postcode": "string",

"pricePerDay": "number",

"rentalLat": "number",

"rentalLng": "number",

"rentalRadius": "number",

"keyWords": "string",

"minRentalDays": "number",

"city": "string",

"backgroundPhoto": "string",

"listingImages": ["object"],

"id": "number",

"listingId": "number",

"userVerified": "boolean"

},

"createdVerifiedRequest": "boolean"

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - невірні дані запиту або файл перевищує допустимий розмір

- 401 Unauthorized - відсутній або недійсний токен доступу

## Updating Listing

**Endpoint:** /update

**Method:** POST

**Description:** оновлення існуючого лістингу

**Request Headers:**

Content-Type: multipart/form-data

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"id": "number",

"address": "string",

"name": "string",

"categoryId": "number",

"compensationCost": "number",

"countStoredItems": "number",

"description": "string",

"postcode": "string",

"pricePerDay": "number",

"rentalLat": "number",

"rentalLng": "number",

"rentalRadius": "number",

"keyWords": "string",

"minRentalDays": "number", // необов'язкове поле, за замовчуванням null

"city": "string",

"active": "boolean", // необов'язкове поле, за замовчуванням true

"otherCategory": "string",

"backgroundPhoto": "file", // необов'язкове поле

"listingImages": ["file"] // необов'язкове поле

}

**Response body-field:**

{

"listing": {

"address": "string",

"name": "string",

"categoryId": "number",

"compensationCost": "number",

"countStoredItems": "number",

"description": "string",

"postcode": "string",

"pricePerDay": "number",

"rentalLat": "number",

"rentalLng": "number",

"rentalRadius": "number",

"keyWords": "string",

"minRentalDays": "number",

"city": "string",

"backgroundPhoto": "string",

"listingImages": ["object"],

"id": "number",

"listingId": "number",

"userVerified": "boolean"

},

"createdVerifiedRequest": "boolean"

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - невірні дані запиту або файл перевищує допустимий розмір

- 401 Unauthorized - відсутній або недійсний токен доступу

- 409 Conflict - список має незавершене бронювання або замовлення

- 403 Forbidden - доступ до цього інструменту заборонено

## Change Listing Active Status

**Endpoint:** /change-active

**Method:** POST

**Description:** зміна активного статусу списку

**Request Headers:**

Content-Type: multipart/form-data

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"id": "number"

}

**Response body-field:**

{

"listing": {

"id": "number",

"active": "boolean"

}

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - невірні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

- 409 Conflict - список має незавершене бронювання або замовлення

- 403 Forbidden - доступ до цього інструменту заборонено

## Change Listing Favorite Status

**Endpoint:** /change-favorite

**Method:** POST

**Description:** позначення, або скасування позначення, лістингу, як “улюблений”

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"listingId": "number"

}

**Response body-field:**

{

"isFavorite": "boolean"

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - невірні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

# LISTING APPROVAL REQUESTS

**Endpoint:** /listing-approval-requests

**Description:** запити адміністраторам на підтвердження валідності лістингів

## Create Listing Approval Request

**Endpoint:** /create

**Method:** POST

**Description:** створення запиту на затвердження інструменту

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"listingId": "number"

}

**Response body-field:**

{

"id": "number"

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - запит вже був надісланий раніше. Зачекайте відповіді адміністратора

- 403 Forbidden - доступ до цього інструменту заборонено

- 401 Unauthorized - відсутній або недійсний токен доступу

# ORDERS

**Endpoint:** /orders

**Description:** Робота з ордерами користувачів

## Create Order

**Endpoint:** /create

**Method:** POST

**Description:** створення нового замовлення

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"pricePerDay": "number",

"startDate": "string", // формат YYYY-MM-DD

"endDate": "string", // формат YYYY-MM-DD

"listingId": "number",

"feeActive": "boolean",

"message": "string" // необов'язкове поле

}

**Response body-field:**

{

"id": "number" // ID створеного замовлення

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту або вибрана дата недоступна для бронювання

- 401 Unauthorized - відсутній або недійсний токен доступу

## Extend Order

**Endpoint:** /extend

**Method:** POST

**Description:** продовження існуючого замовлення

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"pricePerDay": "number",

"startDate": "string", // формат YYYY-MM-DD

"endDate": "string", // формат YYYY-MM-DD

"listingId": "number",

"feeActive": "boolean",

"message": "string", // необов'язкове поле

"parentOrderId": "number"

}

**Response body-field:**

{

"id": "number",

"chatMessage": "string",

"opponent": {

"id": "number",

"name": "string",

"email": "string",

// додаткові поля користувача

}

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту або вибрана дата недоступна для бронювання

- 401 Unauthorized - відсутній або недійсний токен доступу

- 404 Not Found - попереднє замовлення не знайдено

- 409 Conflict - власник товару повинен підтвердити ваше попереднє прохання про продовження

- 403 Forbidden - доступу до цього замовлення немає

## Get Full Order By ID

**Endpoint:** /get-full-by-id/:id

**Method:** GET

**Description:** отримання повної інформації про замовлення за ID

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Parameters:**

- id (number) - ID замовлення

**Response body-field:**

{

"id": "number",

"tenantId": "number",

"ownerId": "number",

"status": "string",

"pricePerDay": "number",

"startDate": "string",

"endDate": "string",

"listingId": "number",

"feeActive": "boolean",

"message": "string",

"payedInfo": {

// додаткові поля

},

"extendOrders": [

{

"id": "number",

"pricePerDay": "number",

"startDate": "string",

"endDate": "string",

// додаткові поля

}

],

// додаткові поля замовлення

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

- 404 Not Found - замовлення не знайдено або користувач не має доступу до нього

- 403 Forbidden - доступу до цього замовлення немає

## Session User Order List

**Endpoint:** /order-list

**Method:** POST

**Description:** отримання списку замовлень

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"type": "string", // 'tenant' для орендарів або 'owner' для власників

"filter": "string", // фільтр для замовлень

"start": "number", // початок пагінації

"count": "number", // кількість замовлень для отримання

"clientTime": "string", // час клієнта

"fromTime": "string", // (не обов'язкове) час початку

"toTime": "string" // (не обов'язкове) час закінчення

}

**Response body-field:**

{

"items": [

{

"id": "string", // ідентифікатор замовлення

"listingId": "string", // ідентифікатор списку

"tenantId": "string", // ідентифікатор орендаря

"ownerId": "string", // ідентифікатор власника

"status": "string", // статус замовлення

"offerPricePerDay": "number", // ціна за день

"offerStartDate": "string", // дата початку

"offerEndDate": "string", // дата закінчення

"conflictOrders": [

{

"id": "string", // ідентифікатор конфліктного замовлення

"status": "string" // статус конфліктного замовлення

}

],

"blockedDates": [

{

"startDate": "string", // дата початку блокування

"endDate": "string" // дата закінчення блокування

}

],

"paymentInfo": {

"amount": "number", // сума платежу

"status": "string" // статус платежу

},

"canFastCancelPayed": "boolean", // чи можна швидко скасувати оплачений

"canFinalization": "boolean", // чи можна завершити

"extendOrders": [

{

"id": "string", // ідентифікатор продовженого замовлення

"status": "string" // статус продовженого замовлення

}

]

}

],

"options": {

"type": "string", // 'tenant' або 'owner'

"filter": "string", // фільтр

"start": "number", // початок пагінації

"count": "number" // кількість замовлень

},

"countItems": "number" // загальна кількість замовлень

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

## Accept Booking

**Endpoint:** /accept-booking

**Method:** POST

**Description:** підтвердження бронювання

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"id": "number"

}

**Response body-field:**

{

"chatMessage": {

// дані повідомлення чату

},

"status": "string"

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

- 403 Forbidden - доступу до цього замовлення немає

- 404 Not Found - замовлення не знайдено

- 409 Data Conflict - конфлікт замовлень

## Reject Booking

**Endpoint:** /reject-booking

**Method:** POST

**Description:** відхилення бронювання

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"id": "number"

}

**Response body-field:**

{

"chatMessage": {

// дані повідомлення чату

},

"status": "string",

"cancelStatus": "string"

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

- 403 Forbidden - доступ заборонено

- 404 Not Found - замовлення не знайдено

- 409 Conflict - неможливо виконати операцію для поточного статусу замовлення

## Reject Booking With Page Props

**Endpoint:** /reject-booking-with-page-props

**Method:** POST

**Description:** відхилення замовлення з оновленням інформації про замовлення та списку замовлень

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"id": "string", // ідентифікатор замовлення

"type": "string", // 'tenant' для орендарів або 'owner' для власників

"filter": "string", // фільтр для замовлень

"start": "number", // початок пагінації

"count": "number", // кількість замовлень для отримання

"clientTime": "string", // час клієнта

"fromTime": "string", // (не обов'язкове) час початку

"toTime": "string" // (не обов'язкове) час закінчення

}

**Response body-field:**

{

"orderUpdateResult": {

"chatMessage": "string", // повідомлення чату

"status": "string", // новий статус замовлення

"cancelStatus": "string" // новий статус скасування

},

"orderListResult": {

"items": [

{

"id": "string", // ідентифікатор замовлення

"listingId": "string", // ідентифікатор списку

"tenantId": "string", // ідентифікатор орендаря

"ownerId": "string", // ідентифікатор власника

"status": "string", // статус замовлення

"offerPricePerDay": "number", // ціна за день

"offerStartDate": "string", // дата початку

"offerEndDate": "string", // дата закінчення

"conflictOrders": [

{

"id": "string", // ідентифікатор конфліктного замовлення

"status": "string" // статус конфліктного замовлення

}

],

"blockedDates": [

{

"startDate": "string", // дата початку блокування

"endDate": "string" // дата закінчення блокування

}

],

"paymentInfo": {

"amount": "number", // сума платежу

"status": "string" // статус платежу

},

"canFastCancelPayed": "boolean", // чи можна швидко скасувати оплачений

"canFinalization": "boolean", // Чи можна завершити

"extendOrders": [

{

"id": "string", // ідентифікатор продовженого замовлення

"status": "string" // статус продовженого замовлення

}

]

}

],

"options": {

"type": "string", // 'tenant' або 'owner'

"filter": "string", // фільтр

"start": "number", // початок пагінації

"count": "number" // кількість замовлень

},

"countItems": "number" // загальна кількість замовлень

}

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

- 403 Forbidden – доступ до ордера відсутній

- 404 Not Found - замовлення не знайдено

- 409 Conflict - неможливо виконати операцію для поточного статусу замовлення

## PayPal Order Paid

**Endpoint:** /paypal-order-payed

**Method:** POST

**Description:** обробка підтвердження платежу через PayPal для замовлення

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"orderId": "string", // Ідентифікатор замовлення від PayPal

"type": "string" // Тип платежу PayPal, "paypal" | "credit-card"

}

**Response body-field:**

{

"chatMessage": {

// дані повідомлення чату

},

"orderPart": {

"id": "string", // ідентифікатор замовлення

"status": "string" // новий статус замовлення

}

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

- 403 Forbidden – доступ до ордера відсутній

- 404 Not Found - замовлення не знайдено

- 409 Conflict - конфлікт платежів або статусів

## Unpaid Order Transaction Bank Transfer

**Endpoint:** /unpaid-order-transaction-bank-transfer

**Method:** POST

**Description:** обробка підтвердження непогашеного платежу за допомогою банківського переказу

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"orderId": "string",

"proof": "file" // обов'язкове поле

}

**Response body-field:**

{

"transactionId": "string",

"type": "string"

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

- 403 Forbidden - доступ до ордера відсутній

- 404 Not Found - замовлення не знайдено

- 409 Conflict - конфлікт з інформацією про платіж

## Approve Client Got Listing

**Endpoint:** /approve-client-got-listing

**Method:** POST

**Description:** підтвердження отримання об'єкта оренди клієнтом та обробка подальших платежів

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"token": "string", // токен підтвердження орендодавця

"defectDescription": "string" // опис дефектів (опційно)

}

**Response body-field:**

{

"chatMessage": {

// дані повідомлення чату

},

"orderPart": {

"id": "string", // ідентифікатор замовлення

"status": "string" // новий статус замовлення

},

"qrCode": "string" // згенерований QR-код

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

- 403 Forbidden - доступ до ордера відсутній

- 404 Not Found - замовлення не знайдено

- 409 Conflict - конфлікт із поточним статусом замовлення

## Finish Order

**Endpoint:** /finished-by-owner

**Method:** POST

**Description:** завершення замовлення

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"token": "string", // токен замовлення для власника

"defectDescription": "string" // опис дефектів (необов'язково)

}

**Response body-field:**

{

"chatMessage": {

// дані повідомлення чату

},

"orderPart": {

"id": "string", // ідентифікатор замовлення

"status": "string" // новий статус замовлення

}

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

- 403 Forbidden - доступ до замовлення відсутній

- 404 Not Found - замовлення не знайдено

- 409 Conflict - конфлікт завершення або статусу

## Full Cancel Payed

**Endpoint:** /full-cancel-payed

**Method:** POST

**Description:** повне скасування оплаченої броні

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"id": "string", // ідентифікатор замовлення

"receiptType": "string", // тип платежу, "bank-transfer" | "paypal"

"paypalId": "string", // ідентифікатор PayPal (для типу "paypal", необов'язково)

"cardNumber": "string" // номер картки (для типу "bank-transfer", необов'язково)

}

**Response body-field:**

{

"chatMessage": {

// дані повідомлення чату

},

"orderPart": {

"id": "string", // ідентифікатор замовлення

"cancelStatus": "string" // новий статус скасування замовлення

}

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

- 403 Forbidden - доступ до замовлення відсутній

- 404 Not Found - замовлення не знайдено

- 409 Conflict - конфлікт скасування або статусу

## Full Cancel Payed With Page Props

**Endpoint:** /full-cancel-payed-with-page-props

**Method:** POST

**Description:** повне скасування оплаченої броні з оновленням даних сторінки

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"id": "string", // ідентифікатор замовлення

" receiptType": "string", // тип платежу, "bank-transfer" | "paypal"

"paypalId": "string", // ідентифікатор PayPal (для типу "paypal", необов'язково)

"cardNumber": "string" // номер картки (для типу "bank-transfer", необов'язково),

"type": "string", // 'tenant' для орендарів або 'owner' для власників

"filter": "string", // фільтр для замовлень

"start": "number", // початок пагінації

"count": "number", // кількість замовлень для отримання

"clientTime": "string", // час клієнта

"fromTime": "string", // (не обов'язкове) час початку

"toTime": "string" // (не обов'язкове) час закінчення

}

**Response body-field:**

{

"orderUpdateResult": {

"chatMessage": "string", // повідомлення чату

"status": "string", // новий статус замовлення

"cancelStatus": "string" // новий статус скасування

},

"orderListResult": {

"items": [

{

"id": "string", // ідентифікатор замовлення

"listingId": "string", // ідентифікатор списку

"tenantId": "string", // ідентифікатор орендаря

"ownerId": "string", // ідентифікатор власника

"status": "string", // статус замовлення

"offerPricePerDay": "number", // ціна за день

"offerStartDate": "string", // дата початку

"offerEndDate": "string", // дата закінчення

"conflictOrders": [

{

"id": "string", // ідентифікатор конфліктного замовлення

"status": "string" // статус конфліктного замовлення

}

],

"blockedDates": [

{

"startDate": "string", // дата початку блокування

"endDate": "string" // дата закінчення блокування

}

],

"paymentInfo": {

"amount": "number", // сума платежу

"status": "string" // статус платежу

},

"canFastCancelPayed": "boolean", // чи можна швидко скасувати оплачений

"canFinalization": "boolean", // Чи можна завершити

"extendOrders": [

{

"id": "string", // ідентифікатор продовженого замовлення

"status": "string" // статус продовженого замовлення

}

]

}

],

"options": {

"type": "string", // 'tenant' або 'owner'

"filter": "string", // фільтр

"start": "number", // початок пагінації

"count": "number" // кількість замовлень

},

"countItems": "number" // загальна кількість замовлень

}

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

- 403 Forbidden - доступ до замовлення відсутній

- 404 Not Found - замовлення не знайдено

- 409 Conflict - конфлікт скасування або статусу

## Full Cancel

**Endpoint:** /full-cancel

**Method:** POST

**Description:** повне скасування замовлення

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"id": "string" // ідентифікатор замовлення

}

**Response body-field:**

{

"chatMessage": {

// дані повідомлення чату

},

"orderPart": {

"cancelStatus": "string", // новий статус скасування

"id": "string" // ідентифікатор замовлення

}

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

- 403 Forbidden - доступ до замовлення відсутній

- 404 Not Found - замовлення не знайдено

- 409 Conflict - конфлікт скасування або статусу

## Full Cancel With Page Props

**Endpoint:** /full-cancel-with-page-props

**Method:** POST

**Description:** повне скасування замовлення з оновленням даних сторінки

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"id": "string", // ідентифікатор замовлення

"type": "string", // 'tenant' для орендарів або 'owner' для власників

"filter": "string", // фільтр для замовлень

"start": "number", // початок пагінації

"count": "number", // кількість замовлень для отримання

"clientTime": "string", // час клієнта

"fromTime": "string", // (не обов'язкове) час початку

"toTime": "string" // (не обов'язкове) час закінчення

}

**Response body-field:**

{

"orderUpdateResult": {

"chatMessage": "string", // повідомлення чату

"status": "string", // новий статус замовлення

"cancelStatus": "string" // новий статус скасування

},

"orderListResult": {

"items": [

{

"id": "string", // ідентифікатор замовлення

"listingId": "string", // ідентифікатор списку

"tenantId": "string", // ідентифікатор орендаря

"ownerId": "string", // ідентифікатор власника

"status": "string", // статус замовлення

"offerPricePerDay": "number", // ціна за день

"offerStartDate": "string", // дата початку

"offerEndDate": "string", // дата закінчення

"conflictOrders": [

{

"id": "string", // ідентифікатор конфліктного замовлення

"status": "string" // статус конфліктного замовлення

}

],

"blockedDates": [

{

"startDate": "string", // дата початку блокування

"endDate": "string" // дата закінчення блокування

}

],

"paymentInfo": {

"amount": "number", // сума платежу

"status": "string" // статус платежу

},

"canFastCancelPayed": "boolean", // чи можна швидко скасувати оплачений

"canFinalization": "boolean", // Чи можна завершити

"extendOrders": [

{

"id": "string", // ідентифікатор продовженого замовлення

"status": "string" // статус продовженого замовлення

}

]

}

],

"options": {

"type": "string", // 'tenant' або 'owner'

"filter": "string", // фільтр

"start": "number", // початок пагінації

"count": "number" // кількість замовлень

},

"countItems": "number" // загальна кількість замовлень

}

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

- 403 Forbidden - доступ до замовлення відсутній

- 404 Not Found - замовлення не знайдено

- 409 Conflict - конфлікт скасування або статусу

## Generate Invoice Pdf

**Endpoint:** /invoice-pdf/:id

**Method:** POST

**Description:** повертає інвойс у форматі pdf

**Request Headers:**

Content-Type: application/pdf

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Parameters:**

{

id (string) - ідентифікатор замовлення

}

**Response Body:** PDF файл рахунку

**Status Code:** 200 OK

**Errors:**

- 404 Not Found - замовлення не знайдено

- 401 Unauthorized - відсутній або недійсний токен доступу

- 403 Forbidden - доступ до замовлення заборонено

# ORDER UPDATE REQUESTS

**Endpoint:** /order-update-requests

**Description:** пропозиції до зміни замовлення

## Create Update Request

**Endpoint:** /create

**Method:** POST

**Description:** створення нової пропозиції

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"orderId": "string", // ідентифікатор замовлення

"newStartDate": "string", // нова дата початку оренди (ISO 8601)

"newEndDate": "string", // нова дата закінчення оренди (ISO 8601)

"newPricePerDay": "number" // нова ціна за день оренди

}

**Response body-field:**

{

"request": {

"id": "string", // ідентифікатор запиту на оновлення

"orderId": "string", // ідентифікатор замовлення

"newStartDate": "string", // нова дата початку оренди (ISO 8601)

"newEndDate": "string", // нова дата закінчення оренди (ISO 8601)

"newPricePerDay": "number", // нова ціна за день оренди

"senderId": "string", // ідентифікатор відправника

"fee": "number" // комісія

},

"chatMessage": {

// дані повідомлення чату

}

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

- 403 Forbidden - доступ до замовлення відсутній

- 404 Not Found - замовлення не знайдено

- 409 Conflict - конфлікт з іншими замовленнями або запитами

# DISPUTES

**Endpoint:** /disputes

**Description:** суперечки до замовлень

## Create Dispute

**Endpoint:** /create

**Method:** POST

**Description:** створення нової пропозиції

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"orderId": "string", // ідентифікатор замовлення

"type": "string", // тип суперечки

"description": "string" // опис суперечки

}

**Response body-field:**

{

"chatMessage": {

// дані повідомлення чату

},

"orderPart": {

"id": "string", // ідентифікатор замовлення

"disputeId": "string", // ідентифікатор суперечки

"disputeStatus": "string", // статус суперечки

"disputeType": "string", // тип суперечки

"disputeDescription": "string", // опис суперечки

"disputeChatId": "string" // ідентифікатор чату суперечки

}

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

- 403 Forbidden - доступ до замовлення відсутній

- 404 Not Found - замовлення не знайдено

# RECIPIENT PAYMENTS

**Endpoint:** /recipient-payments

**Description:** платежі, в результаті яких користувач отримав кошти

## Payment List

**Endpoint:** /list

**Method:** POST

**Description:** отримує список платежів для користувача

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"order": "string", // необов'язкове поле

"orderType": "string", // необов'язкове поле, за замовчуванням "desc"

"itemsPerPage": "number", // необов'язкове поле

"filter": "string", // необов'язкове поле

"start": "number", // необов'язкове поле

}

**Response body-field:**

{

"items": [

{

"id": "string", // ідентифікатор платежу

"amount": "number", // сума платежу

"date": "string", // дата платежу

"status": "string", // статус платежу

"type": "string" // тип отриманого платежу

}

],

options: {

"filter": "string",

"order": "string",

"orderType": "string",

"start": "number",

"count": "number",

"page": "number",

"totalPages: "number",

},

"countItems": "number"

}

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

# SENDER PAYMENTS

**Endpoint:** /sender-payments

**Description:** платежі, в результаті яких користувач отримав кошти

## Payment List

**Endpoint:** /list

**Method:** POST

**Description:** отримує список платежів для користувача

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"type": "string", // тип платежу, наприклад "all" або конкретний тип

"status": "string", // необов'язкове поле, статус платежу, наприклад "all" або конкретний статус

"filter": "string", // необов'язкове поле, фільтр для платежів

"order": "string", // необов'язкове поле, порядок сортування

"orderType": "string", // необов'язкове поле, тип сортування (наприклад, "asc" або "desc"), за замовчуванням "desc"

"itemsPerPage": "number", // необов'язкове поле, кількість елементів на сторінці

"start": "number" // необов'язкове поле, номер сторінки

}

**Response body-field:**

{

"items": [

{

"id": "string",

"amount": "number",

"date": "string",

"status": "string",

"type": "string"

}

],

"options": {

"filter": "string",

"order": "string",

"orderType": "string",

"start": "number",

"count": "number",

"page": "number",

"totalPages": "number",

},

"countItems": "number",

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

## PayPal Create Order

**Endpoint:** /paypal-create-order

**Method:** POST

**Description:** створює нове замовлення в PayPal для платежу за замовленням

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"amount": "number", // сума платежу

"orderId": "string" // ідентифікатор замовлення

}

**Response body-field:**

{

"id": "string", // ідентифікатор PayPal замовлення

"status": "string", // статус PayPal замовлення

"approvalUrl": "string" // URL для переходу на сторінку оплати PayPal

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

- 409 Conflict - конфлікт з поточним статусом замовлення

## Generate Invoice PDF

**Endpoint:** /invoice-pdf/:id

**Method:** POST

**Description:** створює нове замовлення в PayPal для платежу за замовленням

**Request Headers:**

Content-Type: application/pdf

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Parameters:**

id (URL параметр) - ідентифікатор платежу

**Response Body:** PDF файл рахунку

**Status Code:** 200 OK

**Errors:**

- 404 Not Found - платіж не знайдено або користувач не є платником платежу

- 401 Unauthorized - відсутній або недійсний токен доступу

# COMMENTS

**Endpoint:** /comments

**Description:** коментарі до власників та орендарів

## Create Owner Review

**Endpoint:** /create-owner-review

**Method:** POST

**Description:** створює відгук власника і коментар до списку для зазначеного замовлення

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"userCommentInfo": {

"description": "string", // опис коментаря

"quality": "number", // якість (оцінка)

"listingAccuracy": "number", // точність опису (оцінка)

"utility": "number", // корисність (оцінка)

"condition": "number", // стан (оцінка)

"performance": "number", // виконання (оцінка)

"location": "number" // місце розташування (оцінка)

},

"listingCommentInfo": {

"description": "string", // опис коментаря

"punctuality": "number", // пунктуальність (оцінка)

"communication": "number", // комунікація (оцінка)

"flexibility": "number", // гнучкість (оцінка)

"reliability": "number", // надійність (оцінка)

"kindness": "number", // доброта (оцінка)

"generalExperience": "number" // загальне враження (оцінка)

},

"orderId": "string" // ідентифікатор замовлення

}

**Response body-field:**

{

"ownerCommentId": "string", // ідентифікатор коментаря власника

"listingCommentId": "string" // ідентифікатор коментаря до списку

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

- 403 Forbidden - замовлення вже має коментар власнику або коментар до інструменту

## Create Tenant Review

**Endpoint:** /create-tenant-review

**Method:** POST

**Description:** створює відгук власника і коментар до списку для зазначеного замовлення

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"userCommentInfo": {

"description": "string", // опис коментаря

"quality": "number", // якість (оцінка)

"listingAccuracy": "number", // точність опису (оцінка)

"utility": "number", // корисність (оцінка)

"condition": "number", // стан (оцінка)

"performance": "number", // виконання (оцінка)

"location": "number" // місце розташування (оцінка)

},

"orderId": "string" // ідентифікатор замовлення

}

**Response body-field:**

{

" tenantCommentId": "string", // ідентифікатор коментаря орендаря

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

- 403 Forbidden - замовлення вже має коментар орендарю

# CHATS

**Endpoint:** /comments

**Description:** інформація про чати користувачів та їх вміст

## Get Chat List

**Endpoint:** /chat-list

**Method:** POST

**Description:** отримує список чатів для користувача

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"id": "string", // необов'язкове поле: ідентифікатор чату для перевірки доступу

"chatType": "string", // необов'язкове поле: тип чату (наприклад, "order")

"chatFilter": "string", // необов'язкове поле: фільтр для чату

"lastChatId": "string" // необов'язкове поле: ідентифікатор останнього чату для пагінації

}

**Response body-field:**

{

"chats": [

{

"id": "string", // ідентифікатор чату

"messageCreatedAt": "string", // дата і час створення останнього повідомлення

"entityType": "string", // тип сутності (наприклад, "order")

// інші поля чату

}

],

"chatsCanShowMore": "boolean", // чи є ще чати для показу

"options": {

"chatType": "string" // тип чату

},

"dopInfo": {} // додаткова інформація, якщо є

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

- 404 Not Found - чат не знайдено або відсутній доступ до чату

## Get Chat Message List

**Endpoint:** /chat-message-list

**Method:** POST

**Description:** отримує список повідомлень для чату

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"id": "string", // ідентифікатор чату

"lastMessageId": "string" // необов'язкове поле: ідентифікатор останнього повідомлення для пагінації

}

**Response body-field:**

{

"messages": [

{

"id": "string", // ідентифікатор повідомлення

"createdAt": "string", // дата і час створення повідомлення

"senderId": "string", // ідентифікатор відправника

"content": "string" // зміст повідомлення

// інші поля повідомлення

}

],

"messagesCanShowMore": "boolean", // чи є ще повідомлення для показу

"options": {} // додаткові параметри, якщо є

}

**Status Code:** 200 OK

**Errors:**

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

- 404 Not Found - чат не знайдено або відсутній доступ до чату

**Get Chat Base Info**

**Endpoint:** /chat-base-info

**Method:** POST

**Description:** отримує базову інформацію про чат, включаючи список повідомлень та додаткову інформацію про чат

**Request Headers:**

Content-Type: application/json

Authorization: Bearer YOUR\_ACCESS\_TOKEN

**Request Body:**

{

"id": "string", // ідентифікатор чату

"lastMessageId": "string" // необов'язкове поле: ідентифікатор останнього повідомлення для пагінації

}

**Response body-field:**

{

"messages": [

{

"id": "string", // ідентифікатор повідомлення

"createdAt": "string", // дата і час створення повідомлення

"senderId": "string", // ідентифікатор відправника

"content": "string" // зміст повідомлення

// інші поля повідомлення

}

],

"messagesCanShowMore": "boolean", // чи є ще повідомлення для показу

"options": {}, // додаткові параметри для повідомлень

"entity": {

// інформація про сутність чату (наприклад, замовлення чи суперечка)

},

"dopEntityInfo": {

// додаткова інформація про сутність

"tenantBaseCommission": "number", // базова комісія орендаря

"bankInfo": {

// інформація про банківський рахунок

}

}

}

**Status Code:** 200 OK

**Errors:**

- 200 OK - базову інформацію про чат успішно отримано

- 400 Bad Request - некоректні дані запиту

- 401 Unauthorized - відсутній або недійсний токен доступу

- 403 Forbidden - відсутній доступ до чату

- 404 Not Found - чат не знайдено