

# User Service

## Overview

User Service - это микросервис, который отвечает за управление аккаунтами пользователей и предоставляет другим микросервисам их учётные данные

## Tags

### Account

Эндпоинты для управления своим аккаунтом и просмотра аккаунтов других пользователей

### Admin

Эндпоинты для администрирования

### System

Эндпоинты для манипулирования учетными данными пользователей другими микросервисами

## Paths

### ***GET* /api/v0/users/{username}** Получения профиля пользователя

Ищет пользователя по имени и в случае успеха возвращает его профильные данные

#### Parameters

Type	Name	Description	Schema
path	username <i>required</i>	Имя пользователя	string

#### Responses

Code	Description	Links
200	Профиль успешно найден  <i>Content</i> application/json	No Links

Code	Description	Links
404	<p>Пользователь не найден</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"User not found."}</pre>	No Links
500	<p>Ошибка на сервере</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"Internal server error."}</pre>	No Links

## **PATCH** /api/v0/users/update-username **Изменение имени пользователя**

Проверяет свободно ли новое имя пользователя и в случае успеха изменяет имя пользователя на новое

### *Responses*

Code	Description	Links
200	Имя пользователя успешно изменено	No Links
409	<p>Имя пользователя занято</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"Username already exists."}</pre>	No Links

Code	Description	Links
500	<p>Ошибка на сервере</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"Internal server error."}</pre>	No Links

Type	Name	Scopes
oauth2	bearerAuth	UPDATE_USERNAME

## **PATCH** /api/v0/users/update-email **Изменение email**

Проверяет свободен ли новый email и в случае успеха отправляет на новый email письмо для подтверждения

### Responses

Code	Description	Links
202	Процесс изменения email начат	No Links
409	<p>Email занят</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"Email already exists."}</pre>	No Links
500	<p>Ошибка на сервере</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"Internal server error."}</pre>	No Links

Type	Name	Scopes
oauth2	bearerAuth	UPDATE_EMAIL

# GET /api/v0/users/update-email/confirm

## Подтверждение изменения email

Проверяет корректность токена и свободен ли новый email и в случае успеха изменяет email

### Parameters

Type	Name	Description	Schema
query	token <i>required</i>	Токен	string

### Responses

Code	Description	Links
200	Email успешно изменен	No Links
400	Токен некорректный или просроченный  <i>Content</i> <b>application/json</b> <i>Example</i> <div><pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"The confirmation token is invalid or has expired."}</pre></div>	No Links
409	Email занят  <i>Content</i> <b>application/json</b> <i>Example</i> <div><pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"Email already exists."}</pre></div>	No Links
500	Ошибка на сервере  <i>Content</i> <b>application/json</b> <i>Example</i> <div><pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"Internal server error."}</pre></div>	No Links

Type	Name	Scopes
oauth2	bearerAuth	UPDATE_EMAIL

# PATCH /api/v0/users/update-password Изменение пароля

Проверяет совпадение старого пароля и корректность нового и в случае успеха изменяет пароль на новый

### Responses

Code	Description	Links
200	Пароль успешно изменен	No Links
400	Неправильный старый пароль; Некорректный новый пароль  <i>Content</i> <b>application/json</b> <b>Неправильный старый пароль</b>  <i>Example</i> <div><pre>{"created_at": "2024-03-27T03:26:19.385Z", "message": "Old password is incorrect."}</pre></div> <b>Некорректный новый пароль</b>  <i>Example</i> <div><pre>{"created_at": "2024-03-27T03:26:19.385Z", "message": "Validation error:", "errors": {"newPassword": "The password must contain upper and lowercase Latin letters, a number, and a special character."}}</pre></div>	No Links
500	Ошибка на сервере  <i>Content</i> <b>application/json</b>  <i>Example</i> <div><pre>{"created_at": "2024-03-27T03:26:19.385Z", "message": "Internal server error."}</pre></div>	No Links

Type	Name	Scopes
oauth2	bearerAuth	UPDATE_PASSWORD

## DELETE /api/v0/users Удаление своего аккаунта

Удаляет все учётные данные пользователя из системы

### Responses

Code	Description	Links
204	Аккаунт успешно удален	No Links
500	Ошибка на сервере  <i>Content</i> <b>application/json</b> <i>Example</i> <pre>{"created_at": "2024-03-27T03:26:19.385Z", "message": "Internal server error."}</pre>	No Links

Type	Name	Scopes
oauth2	bearerAuth	DELETE_ACCOUNT

## DELETE /api/v0/users/admin/{username} Блокировка аккаунта

Ищет пользователя по имени и в случае успеха блокирует

### Parameters

Type	Name	Description	Schema
path	<b>username</b> <i>required</i>	Имя пользователя	string

### Responses

Code	Description	Links
204	Пользователь успешно удален	No Links
404	Пользователь не найден  <i>Content</i> <b>application/json</b> <i>Example</i> <pre>{"created_at": "2024-03-27T03:26:19.385Z", "message": "User not found."}</pre>	No Links

Code	Description	Links
500	<p>Ошибка на сервере</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"Internal server error."}</pre>	No Links

Type	Name	Scopes
oauth2	bearerAuth	DELETE_USER_ACCOUNT

## **PATCH** **/api/v0/users/admin/ban/{username}** Блокировка аккаунта

Ищет пользователя по имени и в случае успеха блокирует

### Parameters

Type	Name	Description	Schema
path	<b>username</b> <i>required</i>	Имя пользователя	string

### Responses

Code	Description	Links
200	Пользователь успешно заблокирован	No Links
404	<p>Пользователь не найден</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"User not found."}</pre>	No Links

Code	Description	Links
500	<p>Ошибка на сервере</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"Internal server error."}</pre>	No Links

Type	Name	Scopes
oauth2	bearerAuth	BAN_ACCOUNT

## **PATCH** /api/v0/users/admin/unban/{username}

### Разблокировка аккаунта

Ищет пользователя по имени и в случае успеха разблокирует

#### Parameters

Type	Name	Description	Schema
path	username <i>required</i>	Имя пользователя	string

#### Responses

Code	Description	Links
200	Пользователь успешно разблокирован	No Links
404	<p>Пользователь не найден</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"User not found."}</pre>	No Links



Code	Description	Links
500	<p>Ошибка на сервере</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"Internal server error."}</pre>	No Links

Type	Name	Scopes
oauth2	bearerAuth	UNBAN_ACCOUNT

## **GET /sys/v0/users** Получение учетных данных

Ищет учетные данные по логину или по имени пользователя и email и в случае успеха возвращает их

### Parameters

Type	Name	Description	Schema
query	<b>login</b> <i>optional</i>	Логин	string
query	<b>email</b> <i>optional</i>	Email	string
query	<b>username</b> <i>optional</i>	Имя пользователя	string

### Responses

Code	Description	Links
200	<p>Пользователь успешно найден</p> <p><i>Content</i></p> <p><b>application/json</b></p>	No Links
400	<p>Пропущен параметр запроса</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"Missing request parameter."}</pre>	No Links

Code	Description	Links
404	<p>Пользователь не найден</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"User not found."}</pre>	No Links
500	<p>Ошибка на сервере</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"Internal server error."}</pre>	No Links

## **POST /sys/v0/users** Сохранение новых учетных данных

Проверяет имя пользователя, email и пароль и в случае успеха сохраняет учетные данные в базу

### *Responses*

Code	Description	Links
201	Пользователь успешно создан	No Links
400	<p>Некорректные входные данные</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"Validation error:","errors":{"username":"The username must start with a letter and contain only Latin letters, numbers and underscores."}}</pre>	No Links

Code	Description	Links
409	<p>Имя пользователя или адрес электронной почты заняты</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"Username already exists."}</pre>	No Links
500	<p>Ошибка на сервере</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"Internal server error."}</pre>	No Links

## **POST /sys/v0/users/validate-password** Проверка пароля

Ищет учетные данные по логину и в случае успеха проверяет пароли

### *Responses*

Code	Description	Links
200	Пароль успешно проверен	No Links

Code	Description	Links
400	<p>Некорректные входные данные; Неправильный пароль</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><b>Некорректные входные данные</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"Validation error:","errors":{"login":"The login must be not null."}}</pre> <p><b>Неправильный пароль</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"Password is incorrect."}</pre>	No Links
404	<p>Пользователь не найден</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"User not found."}</pre>	No Links
500	<p>Ошибка на сервере</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"Internal server error."}</pre>	No Links

## **PATCH /sys/v0/users/reset-password** Сброс пароля

Ищет учетные данные по логину и в случае успеха изменяет пароль на новый

### Responses

Code	Description	Links
200	Пароль успешно сброшен	No Links

Code	Description	Links
400	<p>Некорректные входные данные</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"Validation error:","errors":{"username":"The username must be not null."}}</pre>	No Links
404	<p>Пользователь не найден</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"User not found."}</pre>	No Links
500	<p>Ошибка на сервере</p> <p><i>Content</i></p> <p><b>application/json</b></p> <p><i>Example</i></p> <pre>{"created_at":"2024-03-27T03:26:19.385Z","message":"Internal server error."}</pre>	No Links

# Components

## Schemas

### ErrorDto

#### Properties

Name	Description	Schema
message <i>required</i>	Сообщение ошибки	string
errors <i>optional</i>	Название поля и его ошибки	object
created_at <i>required</i>	Время создания	string (date-time)

## CredentialsInputDto

### Properties

Name	Description	Schema
username <i>required</i>	Имя пользователя	string
email <i>required</i>	Email	string
password <i>required</i>	Пароль	string

## ValidatePasswordDto

### Properties

Name	Description	Schema
login <i>required</i>	ЛОГИН	string
password <i>required</i>	Пароль	string

## ResetPasswordDto

### Properties

Name	Description	Schema
login <i>required</i>	ЛОГИН	string
password <i>required</i>	Пароль	string

## UpdateUsernameDto

### Properties

Name	Description	Schema
username <i>required</i>	Имя пользователя	string

## UpdatePasswordDto

### Properties

Name	Description	Schema
oldPassword <i>required</i>	Старый пароль	string
newPassword <i>required</i>	Новый пароль	string

## UpdateEmailDto

### Properties

Name	Description	Schema
email <i>required</i>	Email	string

## CredentialsOutputDto

### Properties

Name	Description	Schema
username <i>required</i>	Имя пользователя	string
email <i>required</i>	Email	string
authorities <i>required</i>	Полномочия <i>unique items</i>	< string > array
is_enabled <i>required</i>	Заблокирован ли пользователь	boolean

## ProfileDto

### Properties

Name	Description	Schema
username <i>required</i>	Имя пользователя	string
email <i>required</i>	Email	string
is_enabled <i>required</i>	Заблокирован ли пользователь	boolean