

Описание проекта Стой Монтаж

Авторизация

Для авторизации запроса надо передать заголовок

Authorization: Bearer {token}

Пользователи (users)

Базово пользователи делятся на 2 группы

1. Пользователь - сотрудник
2. Пользователь - администратор системы

Администраторы системы

Администраторы системы - пользователи, для которых установлен флаг `is_system_admin` в базе данных.

Данные пользователи обладают полными правами во всей системе, они могут добавлять, изменять, удалять любую сущность в системе.

⚠️ Важно! Данные пользователи не являются сотрудниками, их нельзя добавить в проект

Эти пользователи нужны по большей части для разработчиков системы и людей, которые будут заниматься ее поддержкой

Сотрудники

Пользователи у которых установлен флаг `is_employee` в базе данных. Могут быть добавлены в проекты как сотрудники. Имеют доступ к разным функциям системы в зависимости от своей роли

Таблица в БД

Таблица называется `users` и содержит следующие колонки

Название поля	Тип поля	Примечание
<code>id</code>	<code>int</code>	Ключ
<code>first_name</code>	<code>varchar(255)</code>	Имя пользователя, обязательное
<code>second_name</code>	<code>varchar(255)</code>	Отчество
<code>last_name</code>	<code>varchar(255)</code>	Фамилия, обязательное
<code>birth_date</code>	<code>date</code>	Дата рождения, обязательное
<code>gender</code>	<code>varchar(255)</code>	Пол, обязательное
<code>email</code>	<code>varchar(255)</code>	Электронная почта

Название поля	Тип поля	Примечание
phone	varchar(255)	Номер телефона
password	varchar(255)	Пароль (хеш), обязательное
role	varchar(255)	Роль. Доступные значения: пользователь, Главный Инженер проекта (ГИП), Бухгалтер, Сметчик, Бригадир, обязательное, по умолчанию - пользователь
avatar_id	int	id аватара пользователя. По дизайну они есть, добавил на будущее
is_employee	boolean	Является ли пользователь сотрудником, обязательное
employment_date	date	Дата приема на работу, обязательное для сотрудников
employee_status	varchar(255)	Статус сотрудника - простой текст, обязательное для сотрудников
is_dismissed	boolean	Уволен ли сотрудник
dismissal_date	date	Дата увольнения
position	varchar(255)	Должность - простой текст, обязательное для сотрудников
rate_per_hour	double	Ставка в час, обязательное для сотрудников
work_schedule	varchar(255)	Рабочий график - простой текст, обязательное для сотрудников
is_system_admin	boolean	Является ли пользователь админом системы, обязательное
deleted	boolean	Удален ли пользователь
deleted_at	timestamp	Дата-время удаления, автоматически заполняется
created_at	timestamp	Дата-время создания, автоматически заполняется
updated_at	timestamp	Дата-время изменения, автоматически заполняется

⚠ Флаги `is_employee` и `is_system_admin` не могут быть равны `true` одновременно

Методы

Методы, которые связаны с пользователями

Адрес	Метод	Параметры	Описание
/api/auth/login	POST	{ "email": " org@com.net ", "password": "super-secret" }	Авторизация пользователя. Возвращает модель пользователя, токен доступа

Адрес	Метод	Параметры	Описание
/api/auth/register	POST	{ "first_name": "first_name", "second_name": "second_name", "last_name": "last_name", "birth_date": "2020-01-01", "gender": "male", "email": " org@com.net ", "phone": "+79999999999", "password": "super-secret", "role": "Главный Инженер проекта (ГИП)", "is_employee": true, "is_system_admin": false, "employment_date": "2021-01-01", "position": "Самый главный", "employee_status": "active", "rate_per_hour": 3000.00, "work_schedule": "5/2" }	Добавление нового пользователя. Для создания пользователя требуется авторизация Добавлять новых пользователей могут: Администраторы системы Пользователи с ролью Главный Инженер проекта (ГИП) Возвращает, модель пользователя, токен доступа
/api/auth/me	GET		Получение данных об авторизованном пользователе. Возвращает модель пользователя
/api/users	GET		Возвращает список пользователей, требуется авторизация
/api/users/{id}	GET		Возвращает конкретного пользователя, требуется авторизация

Проекты

Проекты - таблица в БД и сущность в системе

В таблице ниже поля таблицы projects

Название поля	Тип поля	Примечание
id	int	Ключ
name	varchar(255)	Название проекта (обязательное)
start_date	date	Дата начала проекта

Название поля	Тип поля	Примечание
end_date	date	Дата окончания проекта
project_manager_id	int	id пользователя - бригадир по проекту." Только пользователи с ролью "Бригадир" Обязательное
budget	int	Размер ФОТ
address	text	Адрес
description	text	Описание проекта
status	varchar(255)	Статус проекта - простой текст, по умолчанию "В работе"
is_archived	boolean	Отметка архивности проекта
is_deleted	boolean	Отметка удаленности проекта
deleted_at	timestamp	Дата-время удаления проекта, автоматически заполняется
created_at	timestamp	Дата-время создания проекта, автоматически заполняется
updated_at	timestamp	Дата-время изменения проекта, автоматически заполняется

Методы

Методы, которые связаны с проектами

Адрес	Метод	Параметры	Описание
/api/projects	GET		Получение списка проектов, требуется авторизация
/api/projects/{id}	GET		Получение проекта, требуется авторизация
/api/projects	POST	{ "name": "Тестовый проект", "project_manager_id": 2, "start_date": "2025-10-01", "end_date": "2025-12-01", "budget": 1000000.00, "address": "Москва, ул. Снежная, д. 16, к. 4", "description": "Это тестовый проект, который создается из Postman", "status": "В работе", "employee_ids": [2,3] }	Создание проекта, требуется авторизация, права уровня админа систему или ГИП. Возвращает модель проекта

Адрес	Метод	Параметры	Описание
/api/projects/{id}	PATCH	{ "name": "Тестовый проект", "project_manager_id": 2, "start_date": "2025-10-01", "end_date": "2025-12-01", "budget": 1000000.00, "address": "Москва, ул. Снежная, д. 16, к. 4", "description": "Это тестовый проект, который создается из Postman", "status": "В работе", "employee_ids": [1,2,3] }	Редактирование проекта, требуется авторизация, права уровня администратора системы или ГИП
/api/projects/{id}	DELETE		Удаление проекта, требуется авторизация
/api/projects/{id}/archive	POST		Изменение статуса "архивности" проекта
/api/projects/{id}/nomenclature/{nomenclatureId}/add	POST	{ "start_amount": 100 }	Добавление номенклатуры в проект
/api/projects/{id}/nomenclature/{nomenclatureId}/remove	POST		Удаление номенклатуры из проекта (не изменение)

Номенклатура

Таблица nomenclature, одноименная модель - библиотека номенклатуры.

Таблица

Название поля	Тип поля	Примечание
id	int	Ключ
name	varchar(255)	Название
unit	varchar(255)	Единицы изменения
price	decimal	Цена
description	text	Описание
is_deleted	boolean	Флаг удаления
deleted_at	timestamp	Дата-время удаления
created_at	timestamp	Дата-время создания
updated_at	timestamp	Дата-время изменения

Методы

Адрес	Метод	Параметры	Описание
/api/nomenclature	GET		Возвращает список номенклатуры
/api/nomenclature/{id}	GET		Получение номенклатуры по ИД
/api/nomenclature	POST	{ "name": "Вода", "unit": "Литр", "price": 10, "description": "Это простая техническая вода по 10 рублей за литр. Вдруг кому-то надо?" }	Добавление номенклатуры. Права - ГИП и Сметчик
/api/nomenclature/{id}	PATCH	{ "name": "Вода", "unit": "Литр", "price": 10, "description": "Это простая техническая вода по 10 рублей за литр. Вдруг кому-то надо?" }	Изменение номенклатуры
/api/nomenclature/{id}	DELETE		Удаление номенклатуры
/api/nomenclature/import	POST	Form-data: file - файл mode - режим, возможные значения: demo, import	Импорт из файла. Поддерживается CSV, XLS, XLSX. Обязательные колонки: Номенклатура, Ед. измерения, Кол-во. Примеры доступны по адресу /test.csv /test.xlsx /test.xls