

API Documentation

API Documentation

This document lists the main REST API endpoints exposed by the backend.

Authentication

- POST /api/auth/login - Authenticate a user.
 - Request: { "email": "demo@example.com", "password": "Password123!" }
 - Response: { "token": "<jwt>" }
- POST /api/auth/register - Register a new user.
 - Request: { "email": "new@user.com", "password": "NewPass123!", "name": "New User" }

Users

- GET /api/users - List users (requires auth)
- GET /api/users/{userId} - Get user details

Boards

- POST /api/boards - Create a board
 - Request: { "name": "Board name" }
- GET /api/boards - List boards

Task Lists

- POST /api/task-lists - Create a task list
 - Request: { "name": "To Do", "boardId": 1 }
- GET /api/task-lists/board/{boardId} - Lists for a board
- PUT /api/task-lists/{id} - Update a list
- DELETE /api/task-lists/{id} - Delete a list

Tasks

- POST /api/tasks - Create task
 - Request: { "title": "Do X", "description": "Details", "listId": 1 }
- GET /api/tasks/list/{listId} - Tasks in a list
- PUT /api/tasks/{taskId}/assign - Assign user to task
- PUT /api/tasks/{taskId}/reorder - Reorder within list
- PUT /api/tasks/{taskId}/move - Move task between lists
- PUT /api/tasks/{taskId} - Update task
- DELETE /api/tasks/{taskId} - Delete task
- GET /api/tasks/paged - Paged tasks
- GET /api/tasks/search - Search tasks

WebSocket

- The backend exposes WebSocket endpoints configured under the `websocket` package. The frontend `s

Notes

- All endpoints that modify or expose user-specific data require Authorization header: `Authorization`
- Response schemas vary; consult source controllers for exact DTO fields.