



WorkSense API (Simple)

1.0.0 OAS 3.0

Basic API Documentation for the WorkSense Project Management Platform

Note: All API routes have the following prefixes:

- New API routes: `/api/v1/...`
- Legacy routes: `/...` (root path)

Please use the new API routes when possible as legacy routes may be deprecated in future versions.

Servers

`http://localhost:5050/api/v1 - Development Server (New API)`

Authorize



Authentication

User authentication operations (Login, Register, Get Self)



GET `/users/{id}` Get a user by ID (protected route)



Parameters

Try it out

Name	Description
------	-------------

id * required	ID of the user to retrieve
----------------------	----------------------------

string (path)	<input type="text" value="id"/>
-------------------------	---------------------------------

Responses

Code	Description	Links
------	-------------	-------

200	User details	<i>No links</i>
-----	--------------	-----------------

Code	Description	Links
	<p>Media type</p> <p>application/json</p> <p>Controls Accept header.</p>	
401	Unauthorized - User is not authenticated	No links
404	User not found	No links

POST	/login Authenticate a user and get a token	^
Parameters		Try it out
No parameters		
Request body <small>required</small>		application/json
<p>Example Value Schema</p> <pre>{ "email": "user@example.com", "password": "string" }</pre>		
Responses		
Code	Description	Links
200	Login successful	No links
	<p>Media type</p> <p>application/json</p> <p>Controls Accept header.</p>	
	<p>Example Value Schema</p>	

Code	Description	Links
	<pre>{ "token": "string", "user": { "id": "string", "name": "string", "email": "string", "role": "string" } }</pre>	
400	Invalid request data	No links
401	Invalid credentials	No links

GET	/protected Test protected route access																						
Parameters		Try it out																					
No parameters																							
Responses																							
<table border="1"> <thead> <tr> <th>Code</th><th>Description</th><th>Links</th></tr> </thead> <tbody> <tr> <td>200</td><td>Protected route access successful</td><td>No links</td></tr> <tr> <td colspan="2"> Media type application/json </td><td></td></tr> <tr> <td colspan="3">Controls Accept header.</td></tr> <tr> <td colspan="2"> Example Value Schema </td><td></td></tr> <tr> <td colspan="2"> <pre>{ "message": "Protected route access successful" }</pre> </td><td></td></tr> <tr> <td>401</td><td>Unauthorized - User is not authenticated</td><td>No links</td></tr> </tbody> </table>			Code	Description	Links	200	Protected route access successful	No links	Media type application/json			Controls Accept header.			Example Value Schema			<pre>{ "message": "Protected route access successful" }</pre>			401	Unauthorized - User is not authenticated	No links
Code	Description	Links																					
200	Protected route access successful	No links																					
Media type application/json																							
Controls Accept header.																							
Example Value Schema																							
<pre>{ "message": "Protected route access successful" }</pre>																							
401	Unauthorized - User is not authenticated	No links																					

PUT	/me Update current user's profile	
Parameters		Try it out

No parameters

Request body required**application/json****Example Value** Schema

```
{
  "email": "string",
  "firstName": "string",
  "lastName": "string",
  "gender": 0,
  "country": "string",
  "nickName": "string",
  "timezone": 0,
  "pfp": "string"
}
```

Responses

Code	Description	Links
200	Profile updated successfully	No links
400	Invalid request data	No links
401	Unauthorized - User is not authenticated	No links

Platform Admin Platform-level administration (Managing roles/permissions) ^GET /admin/permissions List all available permissions in the platform 🔒 ^

Retrieve a list of all available permissions that can be assigned to project roles. Permissions are categorized by their function (Admin, Backlog, General, etc.).

Parameters**Try it out**

No parameters

Responses

Code	Description	Links
200	List of available permissions	No links
	<p>Media type</p> <div style="border: 2px solid #ccc; padding: 2px; display: inline-block;">application/json</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{ "id": "delete:project", "key": "delete:project", "description": "Delete the entire project (use with caution).", "category": "Admin" }, { "id": "edit:backlog", "key": "edit:backlog", "description": "Create, edit, delete backlog items (stories, bugs, etc.)", "category": "Backlog" }, { "id": "edit:project", "key": "edit:project", "description": "Edit project name, description, and context.", "category": "General" }, { "id": "edit:sprint_items", "key": "edit:sprint_items", "description": "Add/remove items from sprints.", "category": "Sprint Management" }, { "id": "manage:members", "key": "manage:members", "description": "Manage project members and permissions." }]</pre>	
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User is not a platform admin	No links

GET /admin/projectRoles List all project roles



Retrieve a list of all available project roles, including both system-defined roles (Developer, Product Owner, Scrum Master, Viewer) and custom roles created by platform admins. Each role includes its assigned permissions that determine what actions members can perform.

Parameters

[Try it out](#)

No parameters

Responses

Code	Description	Links
200	List of project roles	No links
	<p>Media type</p> <div style="border: 2px solid green; padding: 2px; display: inline-block;">application/json</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{ "id": "developer", "name": "Developer", "permissions": ["view:project", "edit:backlog", "edit:sprint_items", "move:sprint_item"] }, { "id": "product-owner", "name": "Product Owner", "permissions": ["view:project", "edit:backlog", "manage:sprints", "manage:members", "edit:sprint_items", "move:sprint_item", "edit:project"] }, { "id": "scrum-master", "name": "Scrum Master", "permissions": [...] }]</pre>	
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User is not a platform admin	No links

POST /admin/projectRoles Create a new project role



Create a new custom project role with specified permissions. The role name must be unique and permissions must be valid permission keys.

Parameters

Try it out

No parameters

Request body required**application/json****Examples:****Basic role with minimal permissions****Example Value** Schema

```
{
  "name": "Basic Developer",
  "permissions": [
    "view:project",
    "edit:backlog"
  ]
}
```

Responses

Code	Description	Links
201	Project role created successfully	<i>No links</i>
	Media type	
	application/json	
	Controls Accept header.	
	Example Value Schema	
	<pre>{ "id": "YCT8iHm9sooTiFkxaBDU", "name": "Custom Developer", "permissions": ["view:project", "edit:backlog"] }</pre>	
400	Invalid request data - Invalid role name or permissions	<i>No links</i>
401	Unauthorized - User is not authenticated	<i>No links</i>
403	Forbidden - User is not a platform admin	<i>No links</i>

PUT /admin/projectRoles/{roleId} Update an existing project role

**Parameters****Try it out**

Name	Description
roleId * required string (path)	ID of the role to update <input type="text" value="roleId"/>

Request body required	<input type="text" value="application/json"/>
-----------------------	---

Example Value	Schema
{ "name": "string", "permissions": ["string"] }	

Responses		
Code	Description	Links

200	Project role updated successfully	<i>No links</i>
Media type <input type="text" value="application/json"/> Controls Accept header.		
400	Invalid request data	<i>No links</i>
401	Unauthorized - User is not authenticated	<i>No links</i>
403	Forbidden - User is not a platform admin	<i>No links</i>
404	Role not found	<i>No links</i>

DELETE /admin/projectRoles/{roleId} Delete a project role



Delete a custom project role. System-defined roles (developer, product-owner, scrum-master, viewer) cannot be deleted. Only roles created through this API can be removed.

Parameters

[Try it out](#)

Name	Description
------	-------------

roleId * required string (path)	ID of the role to delete <i>Example : YCT8iHm9sooTiFkxaBDU</i>
--	---

Responses

Code	Description	Links
200	Project role deleted successfully	<i>No links</i>
	Media type	
	application/json	
	Controls Accept header.	
	Example Value Schema	
	<pre>{ "message": "Project role deleted successfully", "roleId": "YCT8iHm9sooTiFkxaBDU" }</pre>	
400	Bad Request - Cannot delete system-defined role	<i>No links</i>
401	Unauthorized - User is not authenticated	<i>No links</i>
403	Forbidden - User is not a platform admin	<i>No links</i>
404	Role not found	<i>No links</i>

PUT /users/{id} Update a user (admin only)



Parameters

[Try it out](#)

Name	Description
------	-------------

id * required ID of the user to update

string
(path)

Request body required

application/json

Example Value Schema

```
{  
  "name": "string",  
  "email": "string",  
  "password": "string"  
}
```

Responses

Code	Description	Links
200	User updated successfully	No links
	Media type	
	application/json	
	Controls Accept header.	
	Example Value Schema	
	<pre>{ "id": "string", "name": "string", "email": "string", "updatedAt": "2025-05-04T09:06:05.861Z" }</pre>	
400	Invalid request data	No links
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User does not have admin privileges	No links
404	User not found	No links

DELETE /users/{id} Delete a user (admin only)

Parameters

Name **Description**

id * required string (path)	ID of the user to delete id
--	------------------------------------

Responses

Code	Description	Links
200	User deleted successfully	No links
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User does not have admin privileges	No links
404	User not found	No links

GET /users Get all users (admin only)

Parameters

No parameters

Responses

Code	Description	Links
200	List of all users Media type application/json	No links

Controls Accept header.

Code	Description	Links
	Example Value Schema	
<pre>[{ "id": 0, "email": "string", "firstName": "string", "lastName": "string", "platformRole": "string", "pfp": "string" }]</pre>		
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User does not have admin privileges	No links
500	Server error	No links

POST /users Create a new user (admin only) 🔒 ⌂

Parameters	Try it out	
No parameters		
Request body required	application/json	
Example Value Schema		
<pre>{ "name": "string", "email": "user@example.com", "password": "string" }</pre>		
Responses		
Code	Description	Links
201	User created successfully	No links
Media type application/json		
Controls Accept header.		

Code	Description	Links
	Example Value Schema	
	<pre>{ "id": "string", "name": "string", "email": "string", "createdAt": "2025-05-04T09:06:05.873Z" }</pre>	
400	Invalid request data	No links
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User does not have admin privileges	No links
409	Email already in use	No links

Projects

Project management operations including creation, updates, and member management



GET	/projects	List all projects for the authenticated user			
Parameters		Try it out			
No parameters					
Responses					
Code	Description	Links			
200	List of user's projects	No links			
Media type					
<div style="border: 2px solid #ccc; padding: 2px; display: inline-block;">application/json</div>					
Controls Accept header.					
Example Value Schema					
<pre>[{ "id": "proj_abc123", "name": "changedName Platform Enhancements", "description": "There is none because i changed it", "ownerId": 16, "context": { "techStack": [...] } }]</pre>					

Code	Description	Links
	<pre> "React"], "objectives": "" }, "createdAt": { "_seconds": 1745715508, "_nanoseconds": 392000000 }, "updatedAt": { "_seconds": 1745788678, "_nanoseconds": 523000000 } }] </pre>	
401	Unauthorized - User is not authenticated	No links

POST /projects Create a new project  

Parameters

Try it out

No parameters

Request body required

application/json

Example Value Schema

```
{
  "name": "Inventory Management System",
  "description": "Develop a new system to track warehouse inventory.",
  "context": {
    "techStack": [
      "React",
      "Node.js",
      "PostgreSQL",
      "Firebase Auth"
    ],
    "objectives": "Reduce stock discrepancies by 50% within 6 months."
  }
}
```

Responses

Code	Description	Links
201	Project created successfully	No links
	Media type	

Code	Description	Links
	<p>application/json</p> <p>Controls Accept header.</p>	
	<p>Example Value Schema</p> <pre>{ "id": "echRFU49i8xDz0ymJ0u9", "name": "Inventory Management System", "description": "Develop a new system to track warehouse inventory.", "ownerId": 16, "context": { "techStack": ["React", "Node.js", "PostgreSQL", "Firebase Auth"], "objectives": "Reduce stock discrepancies by 50% within 6 months." }, "createdAt": { "_seconds": 1745807479, "_nanoseconds": 15000000 } }</pre>	
400	Invalid request data	No links
401	Unauthorized - User is not authenticated	No links

GET /projects/{projectId} Get project details by ID 🔒 ⌂	
Parameters	Try it out
Name	Description
projectId * required	ID of the project
string (path)	<input type="text" value="projectId"/>
Responses	
Code	Description
200	Project details <p>Media type</p>
	No links

Code	Description	Links
	<p>application/json</p> <p>Controls Accept header.</p>	
	<p>Example Value Schema</p> <pre>{ "id": "proj_abc123", "name": "changedName Platform Enhancements", "description": "There is none because i changed it", "ownerId": 16, "context": { "techStack": ["React"], "objectives": "" }, "createdAt": { "_seconds": 1745715508, "_nanoseconds": 392000000 }, "updatedAt": { "_seconds": 1745788678, "_nanoseconds": 523000000 } }</pre>	
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User is not a member of the project	No links
404	Project not found	No links

PUT /projects/{projectId} Update project details 🔒 ⌂

Parameters	Try it out
Name	Description
projectId * required	ID of the project
string (path)	<input type="text" value="projectId"/>
Request body required	application/json
Example Value Schema	
{ "name": "string", "description": "string", "context": { "techStack": [}}	

```

        "string"
    ],
    "objectives": "string"
}
}

```

Responses

Code	Description	Links
200	Project updated successfully	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "id": "proj_abc123", "name": "changedName Platform Enhancements", "description": "There is none because i changed it", "ownerId": 16, "context": { "techStack": ["React"], "objectives": "" }, "createdAt": { "_seconds": 1745715508, "_nanoseconds": 392000000 }, "updatedAt": { "_seconds": 1745788678, "_nanoseconds": 523000000 } }</pre>	
400	Invalid request data	No links
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User does not have permission to edit the project	No links
404	Project not found	No links

DELETE /projects/{projectId} Delete a project



Try it out

Parameters

Name	Description	
projectId * required string (path)	ID of the project <input type="text" value="projectId"/>	

Responses

Code	Description	Links
200	Project deleted successfully	No links
	Media type application/json	
	Controls Accept header.	
	Example Value Schema	
	{ "message": "string", "projectId": "string" }	
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User does not have permission to delete the project	No links
404	Project not found	No links

Project Members

Operations for managing project members and their roles



GET /projects/{projectId}/members List all members of a project



Parameters

Try it out

Name	Description
projectId * required string (path)	ID of the project <i>Example : proj_abc123</i> <input type="text" value="proj_abc123"/>

Responses

Code	Description	Links
200	List of project members	<i>No links</i>
	Media type	
	<input type="text" value="application/json"/>	
	Controls Accept header.	
	Example Value Schema	
	<pre>[{ "userId": 16, "projectRoleId": "product-owner", "joinedAt": { "_seconds": 1745715722, "_nanoseconds": 754000000 } }]</pre>	
401	Unauthorized - User is not authenticated	<i>No links</i>
403	Forbidden - User is not a member of the project	<i>No links</i>
404	Project not found	<i>No links</i>
500	Internal server error	<i>No links</i>

POST /projects/{projectId}/members Add a new member to the project



Add a new member to the project with a specific role.

The user making the request must have the `manage:members` permission for the project. The `userId` must exist in the SQL database and the `projectId` must exist in the Firestore `/projectRoles` collection.

Parameters**Try it out**

Name	Description
------	-------------

projectId * required
string
(path)
ID of the project
Example : proj_abc123

proj_abc123

Request body required

application/json

Example Value Schema

```
{
  "userId": 102,
  "projectRoleId": "developer"
}
```

Responses

Code	Description	Links
201	Member added successfully	No links
	Media type	
	application/json	
	Controls Accept header.	
	Example Value Schema	
	<pre>{ "userId": 102, "projectRoleId": "developer", "joinedAt": { "_seconds": 1745715722, "_nanoseconds": 754000000 } }</pre>	
400	Invalid request data or project role not found	No links
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User does not have permission to manage members	No links

Code	Description	Links
404	User not found in SQL database	No links
409	User is already a member of the project	No links
500	Database connection error	No links

GET	/projects/{projectId}/members-detail List all members of a project with their email addresses	
------------	--	--

Parameters

Try it out

Name	Description
projectId * required string (path)	ID of the project <i>Example : proj_abc123</i> <input type="text" value="proj_abc123"/>

Responses

Code	Description	Links
200	List of project members with email addresses	No links

Media type

application/json

Controls Accept header.

Example Value Schema

```
[
  {
    "userId": 16,
    "projectRoleId": "product-owner",
    "joinedAt": {
      "_seconds": 1745715722,
      "_nanoseconds": 754000000
    },
    "name": "John Doe",
    "email": "john.doe@example.com",
    "lastLogin": 1745715722,
    "nickname": "johndoe",
    "profilePicture": "https://example.com/profile.jpg"
  }
]
```

Code	Description	Links
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User is not a member of the project	No links
404	Project not found	No links
500	Internal server error or database connection error	No links

PUT /projects/{projectId}/members/{userId} Update a member's role in the project



Update the role of an existing project member.

The user making the request must have the `manage:members` permission for the project. The `projectRoleId` must exist in the Firestore `/projectRoles` collection.

Parameters

[Try it out](#)

Name Description

projectId * required
string
(path)
Example : proj_abc123
proj_abc123

userId * required
integer
(path)
Example : 102
102

Request body required

application/json

Example Value Schema

```
{
  "projectId": "proj_abc123"
}
```

Responses

Code	Description	Links
200	Member role updated successfully	No links
	<p>Media type</p> <div style="border: 2px solid green; padding: 2px; display: inline-block;">application/json</div> <p>Controls Accept header.</p>	
	<p>Example Value Schema</p> <pre>{ "userId": 102, "projectRoleId": "developer", "updatedAt": { "_seconds": 1745715722, "_nanoseconds": 754000000 } }</pre>	
400	Invalid request data or project role not found	No links
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User does not have permission to manage members	No links
404	Member not found	No links
500	Database connection error	No links

DELETE /projects/{projectId}/members/{userId} Remove a member from the project



Remove a user from the project members.

The user making the request must have the **manage:members** permission for the project. The project owner cannot be removed from the project.

Parameters

[Try it out](#)

Name	Description
projectId * required string (path)	ID of the project <i>Example : proj_abc123</i> <input type="text" value="proj_abc123"/>
userId * required integer (path)	ID of the user to remove from the project <i>Example : 102</i>

Name**Description**

102

Responses**Code****Description****Links**

200

Member removed successfully

No links

Media type

application/json

Controls Accept header.

Example Value Schema

```
{
  "message": "Member removed successfully",
  "userId": 102,
  "projectId": "proj_abc123"
}
```

401

Unauthorized - User is not authenticated

No links

403

Forbidden - User does not have permission to manage members or trying to remove project owner

No links

404

Project or member not found

No links

500

Database connection error

*No links***Sprints** Sprint management operations within projects**POST**

/projects/{projectId}/sprints Create a new sprint

**Parameters****Try it out****Name****Description****projectId** * requiredstring
(path)

ID of the project

projectId

Request body required**application/json****Example Value** Schema

```
{
  "name": "string",
  "goal": "string",
  "startDate": "2025-05-04T09:06:05.916Z",
  "endDate": "2025-05-04T09:06:05.916Z"
}
```

Responses

Code	Description	Links
201	Sprint created successfully	<i>No links</i>
	<p>Media type</p> <div style="border: 2px solid green; padding: 2px; display: inline-block;">application/json</div> <p>Controls Accept header.</p> <p>Example Value <small>Schema</small></p> <pre>{ "id": "string", "projectId": "string", "name": "string", "goal": "string", "startDate": { "_seconds": 0, "_nanoseconds": 0 }, "endDate": { "_seconds": 0, "_nanoseconds": 0 }, "status": "Active", "createdAt": { "_seconds": 0, "_nanoseconds": 0 }, "updatedAt": { "_seconds": 0, "_nanoseconds": 0 } }</pre>	
400	Invalid request data (missing dates, invalid date format, or start date after end date)	<i>No links</i>
401	Unauthorized - User is not authenticated	<i>No links</i>

Code	Description	Links															
403	Forbidden - User does not have permission to create sprints	No links															
GET	/projects/{projectId}/sprints Get all sprints for a project																
Parameters		Try it out															
<table> <thead> <tr> <th>Name</th><th>Description</th><th></th></tr> </thead> <tbody> <tr> <td>projectId * required</td><td>ID of the project</td><td></td></tr> <tr> <td>string (path)</td><td>projectId</td><td></td></tr> <tr> <td>status</td><td>Filter sprints by status</td><td></td></tr> <tr> <td>array<string> (query)</td><td>Available values : Active, Planned, Completed</td><td> <div style="border: 1px solid #ccc; padding: 5px; width: fit-content;"> -- Active Planned Completed </div> </td></tr> </tbody> </table>			Name	Description		projectId * required	ID of the project		string (path)	projectId		status	Filter sprints by status		array<string> (query)	Available values : Active, Planned, Completed	<div style="border: 1px solid #ccc; padding: 5px; width: fit-content;"> -- Active Planned Completed </div>
Name	Description																
projectId * required	ID of the project																
string (path)	projectId																
status	Filter sprints by status																
array<string> (query)	Available values : Active, Planned, Completed	<div style="border: 1px solid #ccc; padding: 5px; width: fit-content;"> -- Active Planned Completed </div>															
Responses																	
Code	Description	Links															
200	List of sprints	No links															
<p>Media type</p> <div style="border: 2px solid green; padding: 2px; display: inline-block;">application/json</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>[{ "id": "string", "projectId": "string", "name": "string", "goal": "string", "startDate": { "_seconds": 0, "_nanoseconds": 0 }, "endDate": { "_seconds": 0, "_nanoseconds": 0 } }]</pre>																	

Code	Description	Links
	<pre> }, "status": "Active", "createdAt": { "_seconds": 0, "_nanoseconds": 0 }, "updatedAt": { "_seconds": 0, "_nanoseconds": 0 } }]</pre>	
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User is not a member of the project	No links

GET /projects/{projectId}/sprints/{sprintId} Get a specific sprint by ID



^

Parameters

Try it out

Name	Description
projectId * required	ID of the project string (path)
sprintId * required	ID of the sprint string (path)

Responses

Code	Description	Links
200	Sprint details Media type application/json Controls Accept header. Example Value Schema <pre>{ "id": "string", "projectId": "string", "name": "string", "startAt": "string", "endAt": "string", "status": "Active", "createdAt": { "_seconds": 0, "_nanoseconds": 0 }, "updatedAt": { "_seconds": 0, "_nanoseconds": 0 } }</pre>	No links

Code	Description	Links
	<pre> "name": "string", "goal": "string", "startDate": { "_seconds": 0, "_nanoseconds": 0 }, "endDate": { "_seconds": 0, "_nanoseconds": 0 }, "status": "Active", "createdAt": { "_seconds": 0, "_nanoseconds": 0 }, "updatedAt": { "_seconds": 0, "_nanoseconds": 0 } } } </pre>	
400	Bad Request - sprintId parameter is required	No links
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User is not a member of the project	No links
404	Sprint not found or sprint does not belong to the specified project	No links

PATCH /projects/{projectId}/sprints/{sprintId}/status Update the status of a sprint  

Parameters Try it out

Name	Description
projectId * required	ID of the project string (path)
sprintId * required	ID of the sprint string (path)

Request body required application/json

Example Value Schema

```
{
    "status": "Active"
}
```

Responses

Code	Description	Links
200	Sprint status updated successfully	No links
	<p>Media type</p> <div style="border: 2px solid green; padding: 2px; display: inline-block;"> application/json </div> <p>Controls Accept header.</p>	
	<p>Example Value Schema</p> <pre>{ "id": "string", "projectId": "string", "name": "string", "goal": "string", "startDate": { "_seconds": 0, "_nanoseconds": 0 }, "endDate": { "_seconds": 0, "_nanoseconds": 0 }, "status": "Active", "createdAt": { "_seconds": 0, "_nanoseconds": 0 }, "updatedAt": { "_seconds": 0, "_nanoseconds": 0 } }</pre>	
400	Invalid request data (invalid status) or another sprint is already active	No links
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User does not have permission to update sprint status	No links
404	Sprint not found or sprint does not belong to the specified project	No links

POST

/projects/{projectId}/sprints/{sprintId}/tasks Add a task to a sprint



Parameters

[Try it out](#)

Name	Description
projectId * required	ID of the project string (path)
sprintId * required	ID of the sprint string (path)
Request body required	application/json

Example Value Schema

```
{
  "type": "string",
  "originalId": "string",
  "originalType": "string"
}
```

Responses

Code	Description	Links
201	Task added to sprint successfully Media type application/json Controls Accept header.	No links

Example Value Schema


```
{
      "id": "string",
      "projectId": "string",
      "sprintId": "string",
      "backlogId": "string",
      "title": "string",
      "status": "todo",
      "priority": "string",
      "type": "string",
      "assignees": [
        {
          "id": "string",
          "name": "string",
          "avatarUrl": "string"
        }
      ],
      "subtasksCompleted": 0,
      "subtasksTotal": 0
    }
```

Code	Description	Links
	<pre>"subtasksTotal": 0, "coverImageUrl": "string", "startDate": "2025-05-04T09:06:05.949Z", "endDate": "2025-05-04T09:06:05.949Z", "commentsCount": 0, "linksCount": 0, "description": "string", "order": 0, "createdAt": { "_seconds": 0, "_nanoseconds": 0 }</pre>	
400	Invalid request data	No links
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User does not have permission to add tasks to sprint	No links
404	Sprint not found	No links

GET /projects/{projectId}/sprints/{sprintId}/tasks Get all tasks in a sprint 🔒 ⌂

Parameters ▼ Try it out

Name	Description
projectId * required	ID of the project string (path)
sprintId * required	ID of the sprint string (path)

Responses

Code	Description	Links
200	List of tasks in the sprint Media type application/json Controls Accept header. Example Value Schema	No links

Code	Description	Links
	<pre>[{ "id": "string", "projectId": "string", "sprintId": "string", "backlogId": "string", "title": "string", "status": "todo", "priority": "string", "type": "string", "assignees": [{ "id": "string", "name": "string", "avatarUrl": "string" }], "subtasksCompleted": 0, "subtasksTotal": 0, "coverImageUrl": "string", "startDate": "2025-05-04T09:06:05.959Z", "endDate": "2025-05-04T09:06:05.959Z", "commentsCount": 0, "linksCount": 0, "description": "string", "order": 0, "createdAt": { "_seconds": 0 } }]</pre>	
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User is not a member of the project	No links
404	Sprint not found	No links

POST /projectss/{projectId}/sprints/{sprintId}/complete  

Parameters Try it out

Name	Description
projectId * required string (path)	projectId
sprintId * required string (path)	sprintId

Responses

Code	Description	Links
200	Sprint marcado como completado exitosamente	No links
	Media type	
	application/json	
	Controls Accept header.	
	Example Value Schema	
	<pre>{ "message": "string", "completedAt": {}, "completionMetrics": { "totalItems": 0, "completedItems": 0, "itemsByStatus": {} } }</pre>	
400	El sprint ya está completado	No links
404	Sprint no encontrado	No links

Tasks Task management operations within projects

^

POST /projects/{projectId}/sprints/{sprintId}/tasks Add a task to a sprint  

Parameters 

Name	Description
projectId * required	ID of the project
string (path)	projectId
sprintId * required	ID of the sprint
string (path)	sprintId

Request body required 

Example Value Schema

```
{
  "type": "string",
}
```

```

    "originalId": "string",
    "originalType": "string"
}

```

Responses

Code	Description	Links
201	Task added to sprint successfully	No links
	<p>Media type</p> <div style="border: 2px solid green; padding: 2px; display: inline-block;">application/json</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "id": "string", "projectId": "string", "sprintId": "string", "backlogId": "string", "title": "string", "status": "todo", "priority": "string", "type": "string", "assignees": [{ "id": "string", "name": "string", "avatarUrl": "string" }], "subtasksCompleted": 0, "subtasksTotal": 0, "coverImageUrl": "string", "startDate": "2025-05-04T09:06:05.974Z", "endDate": "2025-05-04T09:06:05.974Z", "commentsCount": 0, "linksCount": 0, "description": "string", "order": 0, "createdAt": { "_seconds": 0, "_nanoseconds": 0 } }</pre>	
400	Invalid request data	No links
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User does not have permission to add tasks to sprint	No links
404	Sprint not found	No links

GET /projects/{projectId}/sprints/{sprintId}/tasks Get all tasks in a sprint



Parameters**Try it out**

Name	Description
projectId * required string (path)	ID of the project <input type="text" value="projectId"/>
sprintId * required string (path)	ID of the sprint <input type="text" value="sprintId"/>

Responses

Code	Description	Links
200	List of tasks in the sprint	No links

Media type

Controls Accept header.

Example Value [Schema](#)

```
[  
  {  
    "id": "string",  
    "projectId": "string",  
    "sprintId": "string",  
    "backlogId": "string",  
    "title": "string",  
    "status": "todo",  
    "priority": "string",  
    "type": "string",  
    "assignees": [  
      {  
        "id": "string",  
        "name": "string",  
        "avatarUrl": "string"  
      }  
    ],  
    "subtasksCompleted": 0,  
    "subtasksTotal": 0,  
    "coverImageUrl": "string",  
    "startDate": "2025-05-04T09:06:05.982Z",  
    "endDate": "2025-05-04T09:06:05.982Z",  
    "commentsCount": 0,  
    "linksCount": 0,  
    "description": "string",  
    "order": 0,  
    "createdAt": {  
      "_seconds": 0,  
      "nanoseconds": 0  
    },  
    "modifiedAt": {  
      "_seconds": 0,  
      "nanoseconds": 0  
    }  
  }]
```

Code	Description	Links
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User is not a member of the project	No links
404	Sprint not found	No links

GET	/projects/{projectId}/tasks/{taskId}	Get a specific task by ID	 
-----	--------------------------------------	---------------------------	---

Parameters	Try it out
------------	------------

Name	Description
projectId * required	ID of the project string (path)
taskId * required	ID of the task string (path)

Responses

Code	Description	Links
200	Task details Media type application/json Controls Accept header.	No links

Example Value Schema

```
{
  "id": "string",
  "projectId": "string",
  "sprintId": "string",
  "backlogId": "string",
  "title": "string",
  "status": "todo",
  "priority": "string",
  "type": "string",
  "assignees": [
    {
      "id": "string",
      "name": "string"
    }
  ],
  "dueDate": "string"
}
```

Code	Description	Links
	<pre> "name": "string", "avatarUrl": "string" },], "subtasksCompleted": 0, "subtasksTotal": 0, "coverImageUrl": "string", "startDate": "2025-05-04T09:06:05.993Z", "endDate": "2025-05-04T09:06:05.993Z", "commentsCount": 0, "linksCount": 0, "description": "string", "order": 0, "createdAt": { "_seconds": 0, </pre>	
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User is not a member of the project	No links
404	Task not found	No links

PATCH /projects/{projectId}/tasks/{taskId} Update a specific task

Try it out

Parameters

Name	Description
projectId * required	ID of the project string (path)
taskId * required	ID of the task string (path)

Request body required

application/json

Example Value Schema

```
{
  "status": "todo",
  "sprintAssigneeId": 0,
  "order": 0
}
```

Responses

Code	Description	Links
200	Task updated successfully	No links
	<p>Media type</p> <div style="border: 1px solid #ccc; padding: 2px; display: inline-block;">application/json</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "id": "string", "projectId": "string", "sprintId": "string", "backlogId": "string", "title": "string", "status": "todo", "priority": "string", "type": "string", "assignees": [{ "id": "string", "name": "string", "avatarUrl": "string" }], "subtasksCompleted": 0, "subtasksTotal": 0, "coverImageUrl": "string", "startDate": "2025-05-04T09:06:06.005Z", "endDate": "2025-05-04T09:06:06.005Z", "commentsCount": 0, "linksCount": 0, "description": "string", "order": 0, "createdAt": { "_seconds": 0, "_nanoseconds": 0 } }</pre>	
400	Invalid request data	No links
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User does not have permission to edit tasks	No links
404	Task not found	No links

DELETE /projects/{projectId}/tasks/{taskId} Delete a specific task  

Parameters

Try it out

Name	Description
------	-------------

projectId * required	ID of the project
-----------------------------	-------------------

Name	Description
string (path)	projectId
taskId * required	ID of the task
string (path)	taskId

Responses

Code	Description	Links
200	Task deleted successfully	No links
	Media type	
	application/json	
	Controls Accept header.	
	Example Value Schema	
	{ "message": "string", "taskId": "string" }	
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User does not have permission to delete tasks	No links
404	Task not found	No links

PATCH /projects/{projectId}/tasks/{taskId}/status Update the status of a specific task



Parameters

[Try it out](#)

Name	Description
projectId * required	ID of the project
string (path)	projectId

Name	Description	
taskId * required string (path)	ID of the task <input type="text" value="taskId"/>	
Request body required	application/json	
Example Value Schema		
{ "status": "todo" }		
Responses		
Code	Description	Links
200	Task status updated successfully	No links
	Media type	
	application/json	
	Controls Accept header.	
	Example Value Schema	
	{ "message": "string", "taskId": "string", "status": "string" }	
400	Invalid request data	No links
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User does not have permission to update task status	No links
404	Task not found	No links

Backlog

Backlog management operations within projects



AI Module

 AI-powered content generation features**POST**`/{projectId}/ai/generate-epic`

Genera sugerencias de épicas basadas en la descripción del proyecto



Parameters

Try it out**Name****Description****projectId** * required

ID del proyecto

string

(path)

projectId

Responses

Code**Description****Links**

200

Lista de épicas sugeridas

No links

Media type

application/json

Controls Accept header.

Example Value Schema

```
{  
  "epics": [  
    {  
      "name": "string",  
      "description": "string",  
      "priority": "lowest"  
    }  
  ]  
}
```

401

No autorizado - Token inválido o expirado

No links

404

Proyecto no encontrado

No links

502

Error al comunicarse con el servicio de IA

No links

POST`/{projectId}/ai/confirm-epics` Confirma y persiste las épicas sugeridas en el proyecto

Parameters

Try it out

Name	Description	
projectId * required string (path)	ID del proyecto projectId	
Request body required	application/json	
Example Value Schema		
<pre>{ "epics": [{ "name": "string", "description": "string", "priority": "lowest" }] }</pre>		
Responses		
Code	Description	Links
201	Épicas guardadas exitosamente	No links
Media type application/json		
Controls Accept header.		
Example Value Schema		
<pre>{ "message": "Epics saved" }</pre>		
400	Datos de entrada inválidos	No links
401	No autorizado - Token inválido o expirado	No links
404	Proyecto no encontrado	No links
409	Conflicto - Las épicas ya existen en el proyecto	No links

POST

/{projectId}/ai/stories/generate-stories

Genera sugerencias de historias de usuario para una épica



Parameters

Try it out

Name

Description

projectId * required

string

(path)

ID del proyecto

projectId

Request body required

application/json

ID de la épica para la cual generar historias

Example Value Schema

```
{
  "epicId": "6wqDIVRt1UdvG610yU08"
}
```

Responses

Code

Description

Links

200

Lista de historias sugeridas

No links

Media type

application/json

Controls Accept header.

Example Value Schema

```
{
  "stories": [
    {
      "name": "string",
      "description": "string",
      "priority": "low",
      "assignees": [
        {}
      ]
    }
  ]
}
```

Code	Description	Links
400	epicId faltante o inválido	No links
401	No autorizado – token inválido o expirado	No links
404	Proyecto o épica no encontrada	No links
502	Error al comunicarse con el servicio de IA	No links

POST	/{{projectId}}/ai/stories/confirm-stories Confirma y persiste las historias sugeridas en el backlog	
------	--	--

Parameters	Try it out
------------	----------------------------

Name	Description
projectId * required <small>string (path)</small>	ID del proyecto <input type="text" value="projectId"/>

Request body	application/json
--------------	----------------------------------

Épica padre y lista de historias a guardar
Example Value Schema

{ "epicId": "6wqDIVRt1UdvG610yU08", "stories": [{ "name": "string", "description": "string", "priority": "low" }] }
--

Responses

Code	Description	Links
201	Historias guardadas exitosamente Media type	No links

Code	Description	Links
	application/json Controls Accept header. Example Value Schema <pre>{ "message": "Stories saved" }</pre>	
400	epicId o stories faltantes / formato inválido	No links
401	No autorizado – token inválido o expirado	No links
404	Proyecto o épica no encontrada	No links
409	Conflicto – Al menos una historia ya existe en el backlog	No links

Backlog Items Backlog item management operations



POST [/projects/{projectId}/backlog/items](#) Create a new backlog item

Create a new backlog item (epic, story, bug, tech task, or knowledge item) in the project. The type field determines which additional fields are required/optional.

Parameters
Try it out

Name	Description
projectId * required string (path)	ID of the project <i>Example : proj_abc123</i> <input type="text" value="proj_abc123"/>

Request body required **application/json**

Example Value Schema

```
{
  "type": "story",
  "title": "User Login with Email/Password",
  "description": "As a user, I want to log in using my email and password.",
  "priority": "high",
  "status": "planned"
}
```

```

    "status": "todo",
    "epicId": "epic_001",
    "storyPoints": 5,
    "linkedStoryId": "story_001",
    "severity": "minor",
    "content": "Decided on JWT-based auth with 1-hour expiry.",
    "tags": [
        "auth",
        "decision"
    ]
}

```

Responses

Code	Description	Links
201	Backlog item created successfully	No links
	<p>Media type</p> <div style="border: 2px solid green; padding: 2px; display: inline-block;">application/json</div> <p>Controls Accept header.</p> <p>Example Value Schema</p> <pre>{ "id": "epic_001", "projectId": "proj_abc123", "type": "epic", "title": "User Authentication Module", "description": "Implement complete user login, registration, and profile management.", "priority": "high", "status": "new", "reporterId": "101", "assigneeId": null, "createdAt": { "_seconds": 1745792338, "_nanoseconds": 635000000 }, "updatedAt": { "_seconds": 1745792352, "_nanoseconds": 683000000 }, "linkedItems": null }</pre>	
400	Invalid request data	No links
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User does not have permission to edit the backlog	No links
404	Project not found	No links

GET /projects/{projectId}/backlog/items List all backlog items for a project



Retrieve all backlog items (epics, stories, bugs, tech tasks, and knowledge items) for a project. Items are returned in a flat array, with each item containing its specific type and related fields.

Parameters

[Try it out](#)

Name	Description
projectId * required string (path)	ID of the project <i>Example : proj_abc123</i> <input type="text" value="proj_abc123"/>
type string (query)	Filter by item type <i>Available values : epic, story, bug, techTask, knowledge</i> <i>Example : story</i> <input type="text" value="story"/>
status string (query)	Filter by item status <i>Available values : new, todo, refining, in-progress, review, done, active</i> <i>Example : todo</i> <input type="text" value="todo"/>

Responses

Code	Description	Links
200	List of backlog items Media type <input type="text" value="application/json"/> Controls Accept header. Example Value Schema <pre>[{ "id": "epic_001", "projectId": "proj_abc123", "type": "epic", "title": "User Authentication Module", "description": "Implement complete user login, registration, and profile management.", "priority": "high", "status": "new", "dueDate": null, "lastUpdated": "2025-04-05T10:00:00Z" }]</pre>	No links

Code	Description	Links
	<pre> "status": "new", "reporterId": "101", "assigneeId": null, "createdAt": { "_seconds": 1745792338, "_nanoseconds": 635000000 }, "updatedAt": { "_seconds": 1745792352, "_nanoseconds": 683000000 }, "linkedItems": null }, { "id": "story_001", "projectId": "proj_abc123", "type": "story", "title": "User Login with Email/Password", "description": "As a user, I want to log in using my email and na </pre>	
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User is not a member of the project	No links
404	Project not found	No links

GET /projects/{projectId}/backlog/items/{itemId} Get a specific backlog item by ID

Parameters

[Try it out](#)

Name	Description
projectId * required	ID of the project string (path)
itemId * required	ID of the backlog item string (path)
type * required	Type of the backlog item Available values : epic, story, bug, techTask, knowledge string (query)

Responses

Code	Description	Links
200	Backlog item details	No links
	Media type	
	application/json	
	Controls Accept header.	
	Example Value Schema	
	<pre>{ "id": "epic_001", "projectId": "proj_abc123", "type": "epic", "title": "User Authentication Module", "description": "Implement complete user login, registration, and profile management.", "priority": "high", "status": "new", "reporterId": "101", "assigneeId": null, "createdAt": { "_seconds": 1745792338, "_nanoseconds": 635000000 }, "updatedAt": { "_seconds": 1745792352, "_nanoseconds": 683000000 }, "linkedItems": null }</pre>	
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User is not a member of the project	No links
404	Project or backlog item not found	No links

PUT /projects/{projectId}/backlog/items/{itemId} Update a backlog item  

DELETE /projects/{projectId}/backlog/items/{itemId} Delete a backlog item  

Delete a backlog item from the project. Note: Deleting an epic will not automatically delete its child stories.

Parameters

Try it out

Name	Description
------	-------------

projectId * required ID of the project
string *Example : proj_abc123*

Name	Description
(path)	proj_abc123
itemId * required string	ID of the backlog item to delete <i>Example : story_001</i>
(path)	story_001
type * required string (query)	Type of the backlog item <i>Available values : epic, story, bug, techTask, knowledge</i> epic

Responses

Code	Description	Links
200	Backlog item deleted successfully	No links
	Media type	
	application/json	
	Controls Accept header.	
	Example Value Schema	
	{ "message": "Backlog item deleted successfully", "itemId": "story_001", "projectId": "proj_abc123" }	
401	Unauthorized - User is not authenticated	No links
403	Forbidden - User does not have permission to edit the backlog	No links
404	Project or backlog item not found	No links

Schemas

