My API 1.0.0

[Base URL: /]

Authorize

Project

GET /api/Project/ Retrieve all projects

Parameters
No parameters

Responses

Code	Description	Links
200	Success	No links

POST /api/Project/ Create a project.

Parameters
No parameters

Responses

Code	Description	Links
200	Success	No links

PUT /api/Project/ Update a project.

Parameters
No parameters

Responses

Code	Description	Links
200	Success	No links

GET /api/Project/{id} Retrieve a project by id.

Parameters

Name	Description
id (integer) (path)	id of the project

Responses

Code	Description	Links
200	Success	No links

DELETE /api/Project/{id} Delete a project by id.

Parameters

Name	Description
id (integer) (path)	id of the project

Responses

Code	Description	Links
200	Success	No links

PATCH /api/Project/{id} Update a project.

Parameters

Name	Description
id (integer) (path)	id of the project
id (integer) (body)	0

Response Body

application/+patch

Curl

```
curl -X PATCH "https://localhost:7001/api/Project/0" -H "accept: text/plain" -H "content-type: application/+patch" -d "0"
```

Request URL

https://localhost:7001/api/Project/0

Response

Code	Description	Links
200	Success	No links

POST /api/Project/{id} Patch a project

Parameters

Name	Description
id (integer) (path)	id of the project
id (integer) (body)	0

Response Body

application/+patch

Curl

```
curl -X POST "https://localhost:7001/api/Project/0" -H "accept: text/plain" -H "content-type: application/+patch" -d "0"
```

Request URL

https://localhost:7001/api/Project/0

Response

Code	Description	Links
200	Success	No links

GET /api/Project/group/{id} Retrieve all projects in a group id

Parameters

Name	Description
id (integer) (path)	id of the group
groupId (query)	groupId

Responses

Code	Description	Links
200	Success	No links

DELETE /api/Project/group/{id} Delete all projects in a group id

Parameters

Name	Description
id (integer) (path)	id of the group
groupId (query)	groupId

Responses

Code	Description	Links
200	Success	No links

Task

GET /api/Task/ Retrieve all tasks

Parameters
No parameters

Responses

Code	Description	Links
200	Success	No links

DELETE /api/Task/ Delete all tasks.

Parameters
No parameters

Responses

Code	Description	Links
200	Success	No links

POST /api/Task/ Create a task

Parameters
No parameters

Response Body

application/+patch

Curl

```
curl -X POST "https://localhost:7001/api/Task/" -H "accept: text/plain" -H "content-type: application/+patch" -d "0"
```

Request URL

https://localhost:7001/api/Task/

Response

Code	Description	Links
200	Success	No links

GET /api/Task/{id} Retrieve a task by id.

Parameters

Name	Description
id (integer) (path)	The ID of the task to retrieve.

Responses

Code	Description	Links
200	Success	No links

DELETE /api/Task/{id} Delete a task by ID.

Parameters

Name	Description
id (integer) (path)	The ID of the task to delete.

Responses

Code	Description	Links
200	Success	No links

PUT /api/Task/{id} Update a task

Parameters

Name	Description
id (integer) (path)	The ID of the task to update.

Response Body

application/+patch

The task to update.

Example Value | Schema

```
{
  "id": 0,
  "name": "string",
  "description": "string",
  "isCompleted": true,
  "dueDate": "2025-08-19T00:00:00.000Z",
  "projectId": 0
}
```

Responses

Code	Description	Links
200	Success	No links

PATCH /api/Task/{id} Update a task by ID.

Parameters

Name	Description
id (integer) (path)	The ID of the task to update.

Response Body

application/+patch

The updated task data.

Example Value | Schema

```
{
  "id": 0,
  "name": "string",
  "description": "string",
  "isCompleted": true,
  "dueDate": "2025-08-19T00:00:00.000Z",
  "projectId": 0
}
```

Responses

Code	Description	Links
200	Success	No links

GET /api/Task/user/{userId}/{id} Retrieve all tasks by user ID

Parameters

Name	Description
userId (integer) (path)	The ID of the user whose tasks to retrieve.
id (integer) (path)	id

Responses

Code	Description	Links
200	Success	No links

DELETE /api/Task/user/{userId}/{id} Delete all tasks by user ID

Parameters

Name	Description
userId (integer) (path)	The ID of the user whose tasks to delete.
id (integer) (path)	id

Responses

Code	Description	Links
200	Success	No links

User

GET /api/User/ Retrieve all users

Parameters
No parameters

Responses

Code	Description	Links
200	Success	No links

DELETE /api/User/ Delete all users

Parameters
No parameters

Responses

Code	Description	Links
200	Success	No links

GET /api/User/{id} Retrieve a user by ID

Parameters

Name	Description
id (integer) (path)	The ID of the user to retrieve.

Responses

Code	Description	Links
200	Success	No links

DELETE /api/User/{id} Delete a user by ID.

Parameters

Name	Description
id (integer) (path)	The ID of the user to delete.

Responses

Code	Description	Links
200	Success	No links

PUT /api/User/{id} Update a user by ID.

Parameters

Name	Description
id (integer) (path)	The ID of the user to update.

Response Body

application/+patch

The updated user data.

Example Value | Schema

```
{
  "id": 0,
  "firstName": "string",
  "lastName": "string",
  "email": "string",
  "password": "string",
  "isAdmin": true,
  "isActive": true,
  "lastLoginDate": "2025-08-19T00:00:00.000Z",
  "createdAt": "2025-08-19T00:00:00.000Z",
  "updatedAt": "2025-08-19T00:00:00.000Z"
}
```

Responses

Code	Description	Links
200	Success	No links

PATCH /api/User/{id} Partially updates a user by ID.

Parameters

Name	Description
id (integer) (path)	The ID of the user to update.

Response Body

application/+patch

The updated user data.

Example Value | Schema

```
{
  "id": 0,
  "firstName": "string",
  "lastName": "string",
  "email": "string",
  "password": "string",
  "isAdmin": true,
  "isActive": true,
  "lastLoginDate": "2025-08-19T00:00:00.000Z",
  "createdAt": "2025-08-19T00:00:00.000Z",
  "updatedAt": "2025-08-19T00:00:00.000Z"
}
```

Responses

Code	Description	Links
200	Success	No links

POST /api/User/{id}/register Registers a new user.

Parameters

Name	Description
id (integer) (path)	id

Response Body

application/+patch

The new user data.

Example Value | Schema

```
{
  "id": 0,
  "firstName": "string",
  "lastName": "string",
  "email": "string",
  "password": "string",
  "isAdmin": true,
  "isActive": true,
  "lastLoginDate": "2025-08-19T00:00:00.000Z",
  "createdAt": "2025-08-19T00:00:00.000Z",
  "updatedAt": "2025-08-19T00:00:00.000Z"
}
```

Responses

Code	Description	Links
200	Success	No links

POST /api/User/login Login User

Parameters
No parameters

Response Body

application/+patch

The new user data.

Example Value | Schema

```
{
  "id": 0,
  "firstName": "string",
  "lastName": "string",
  "email": "string",
  "password": "string",
  "isAdmin": true,
  "isActive": true,
  "lastLoginDate": "2025-08-19T00:00:00.000Z",
  "createdAt": "2025-08-19T00:00:00.000Z",
  "updatedAt": "2025-08-19T00:00:00.000Z"
}
```

Responses

Code	Description	Links
200	Success	No links

Schemas

Project >

```
{
  id: integer($int32)
  name: string
  description: string | null
  isDeleted: boolean
  groupId: integer($int32)
}
```

ProjectWithTasksDto >

```
{
  id: integer($int32)
  name: string
  description: string | null
  isDeleted: boolean
  groupId: integer($int32)
  tasks: [ ... ]
}
```

Task >

```
{
  id: integer($int32)
  name: string
  description: string | null
  isCompleted: boolean
  dueDate: integer($int64) | null
  projectId: integer($int32)
}
```

TaskWithUsersDto >

```
{
  id: integer($int32)
  name: string
  description: string | null
  isCompleted: boolean
  dueDate: integer($int64) | null
  projectId: integer($int32)
  users: [ ... ]
}
```

User >

```
{
  id: integer($int32)
  firstName: string
  lastName: string
  email: string
  password: string
  isAdmin: boolean
  isActive: boolean
  lastLoginDate: integer($int64) | null
  createdAt: integer($int64)
  updatedAt: integer($int64)
}
```

UserWithTasksDto >

```
{
  id: integer($int32)
  firstName: string
  lastName: string
  email: string
  password: string
  isAdmin: boolean
  isActive: boolean
  lastLoginDate: integer($int64) | null
  createdAt: integer($int64)
  updatedAt: integer($int64)
}
```