



Guía básica para la realización del proyecto con OpenProject

Equipo formado por:

Jonathan David Signes Falcó

Lucas Barabaschi Hryniuk

Roly Alcoser Rocha

Índice

Guía básica para la realización del proyecto con OpenProject.....	1
PASO 1 – Documentación inicial sobre OpenProject.....	3
1 - Parte básica.....	3
Para qué sirve OpenProject.....	3
Funcionamiento general de la plataforma.....	3
Acceso al entorno gráfico en la plataforma web.....	3
2 - Parte técnica.....	5
Funcionamiento de la API REST.....	5
Formato y estructura de la base de datos.....	5
Motor de base de datos utilizado (por defecto: PostgreSQL).....	5
Cómo desplegar OpenProject en local (este paso se puede realizar con el método que prefiera cada grupo).....	5
PASO 2 – Despliegue local de OpenProject.....	6
PASO 3 – Introducción de datos.....	7
Mediante el entorno gráfico.....	7
PASO 4 – Práctica con peticiones.....	9
1. CRUD Básico de Proyectos.....	9
1.1. Lista de todos los proyectos.....	9
1.2. Crea un proyecto llamado “Proyecto de Prueba”.....	10
1.3. Obtén los detalles del proyecto que acabas de crear.....	11
1.4. Cambia el nombre del proyecto creado en el punto anterior a “Proyecto Editado”.....	12
1.5. Elimina el proyecto creado.....	13
1.6. EXTRA - Lista todos los usuarios de Open Project.....	13
2. Consultas Ordenadas.....	14
2.1. Lista los proyectos ordenados por fecha de creación (de viejo a nuevo).....	14
2.2. Lista los proyectos ordenados por fecha de edición (de nuevo a viejo).....	15
2.3. Lista los proyectos ordenados por orden alfabético.....	16
3. Consultas con Filtros.....	17
3.1. Crea un proyecto llamado “Proyecto 1”.....	17
3.2. Lista todos los proyectos llamados “Proyecto 1”.....	17
3.3. Crea un par de tareas.....	18
3.4. Lista todas las tareas activas.....	19
3.5. Lista todas las tareas creadas desde antes del 30 de mayo de 2025.....	21
3.6. Lista todas las tareas inactivas creadas después del 20 de mayo.....	22
PASO 5 – Repositorio en GitHub.....	23
PASO 6 – Desarrollo de la página web.....	23

PASO 1 – Documentación inicial sobre OpenProject

1 - Parte básica

Para qué sirve OpenProject

OpenProject es un software de gestión de proyectos de código abierto para la gestión de proyectos tanto clásico.

Permite controlar fácilmente todas las actividades del proyecto y organizar proyectos, tareas, errores o riesgos, así como asignar responsabilidades, realizar un seguimiento de las fechas de vencimiento, ver el historial de cambios o configurar los flujos de trabajo individuales.

Funcionamiento general de la plataforma

Estas son algunas de las funciones que tiene OpenProject para la colaboración de proyectos:

- Cartera de proyectos
- Planificación de proyectos
- Gestión de tareas
- Agile, Kanban y Scrum
- Seguimiento de tiempo
- Colaboración en equipo
- Hoja de ruta del producto
- Flujos de trabajo

Acceso al entorno gráfico en la plataforma web

En este caso, para poder acceder a la web de Open Project, hemos creado una cuenta temporal para acceder a la web por un periodo de 14 días.

Resumen | practicasjlir.openproj...

practicasjlir.openproject.com/projects/demo-project/

Demo project

+

OpenProject

14 días restantes

Comprar ahora

Buscar en practicasjlir.op...

RA

Resumen

Paquetes de trabajo

Diagramas de Gantt

Planificadores de ...

Tableros

Reuniones

Wiki

Noticias

Miembros

Configuración del ...

14 días restantes

Comprar ahora

Demo project / Resumen

Resumen

GETTING STARTED

We are glad you joined! We suggest to try a few things to get started in OpenProject.

Discover the most important features with our [Guided Tour](#).

Try the following steps:

1. Invite new members to your project: → Go to [Members](#) in the project navigation.

2. View the work in your project: → Go to [Work packages](#) in the project navigation.

3. Create a new work package: → Go to [Work packages](#) → [Create](#).

4. Create and update a project plan: → Go to [Project plan](#) in the project navigation.

5. Activate further modules: → Go to [Project settings](#) → [Modules](#).

6. Complete your tasks in the project: → Go to [Work packages](#) → [Tasks](#).

Here you will find our [User Guides](#).

Please let us know if you have any questions or need support. Contact us: [support\[at\]openproject.com](mailto:support[at]openproject.com).

ESTADO DEL PROYECTO

SEGÚN LO PREVISTO

All tasks are on schedule. The people involved know their tasks. The system is completely set up.

DESCRIPCIÓN DEL PROYECTO

This is a short summary of the goals of this demo project.

MIEMBROS

Project admin

2 - Parte técnica

Funcionamiento de la API REST

Los tokens API permiten a las aplicaciones de terceros comunicarse con esta instancia de OpenProject a través de las API REST.

La APIv3 es una API REST de hipermedia, abreviatura de «Hipermedia como motor del estado de la aplicación» (HATEOAS). Esto significa que cada punto final de esta API tendrá enlaces a otros recursos o acciones definidos en el cuerpo resultante.

Estos recursos y acciones relacionados para cualquier recurso serán sensibles al contexto. Por ejemplo, solo se representan las acciones que el usuario autenticado puede realizar. Esto permite identificar dinámicamente las acciones que el usuario podría realizar ante una respuesta dada.

Formato y estructura de la base de datos

OpenProject fue originalmente derivado de Redmine, y mantiene una estructura de base de datos relacional bastante tradicional. Las tablas más relevantes para este proyecto de prácticas son:

- projects
- users
- roles, members, memberships
- work_packages / issues
- statuses
- time_entries
- others

Motor de base de datos utilizado (por defecto: PostgreSQL)

Hemos utilizado PostgreSQL, la base de datos ya integrada en la imagen de OpenProject (all-in-on).

Cómo desplegar OpenProject en local (este paso se puede realizar con el método que prefiera cada grupo)

Lo desplegamos con el docker utilizando comandos en nuestro terminal (comandos que comentaremos en el paso 2). Tendremos seleccionado un puerto, como el 8080, y gracias a él podremos acceder a dicho proyecto con nuestro navegador a través de ese puerto.

PASO 2 – Despliegue local de OpenProject

Para empezar debemos crear una carpeta para nuestro proyecto e incluir dentro las dos carpetas de “pgdata” y “assets”. A continuación dependiendo de si usamos windows o linux en el terminal debemos escribir los siguientes comandos. En mi caso he utilizado Windows, por lo que he usado el siguiente comando de Powershell para generar mi string de 32 bits que se usa más adelante:

```
$bytes = New-Object 'System.Byte[]' 32
```

```
[System.Security.Cryptography.RandomNumberGenerator]::Create().GetBytes($bytes)
```

```
$slave = [Convert]::ToBase64String($bytes)
```

```
$slave
```

Y he obtenido el siguiente resultado: “pe3sry6JO/MF8Z8k+AeJujjTP0dN8Yt5aqHTAioCS4”

Después de realizar todo esto, tenemos que escribir el siguiente comando con el String que hemos obtenido:

```
docker run -d -p 8081:80 --name openproject `
-e OPENPROJECT_HOST_NAME=localhost:8081 `
-e OPENPROJECT_HTTPS=false `
-e
OPENPROJECT_SECRET_KEY_BASE=2rgFbZSnP0hQnAgCMixt1eV7ShBYFPPzz0gjDhT91C
0 `
-v C:\proyecto-local\pgdata:/var/proyecto-local/pgdata `
-v C:\proyecto-local\assets:/var/proyecto-local/assets `
openproject/openproject:15
```

Ahora ya tenemos creada nuestra instancia local de OpenProject, así que entramos al navegador y buscamos: <http://localhost:8081>. Aquí nos aparece usuario y contraseña que ponemos admin admin y listo, después la cambiamos. Y así tendríamos nuestro despliegue local de OpenProject.

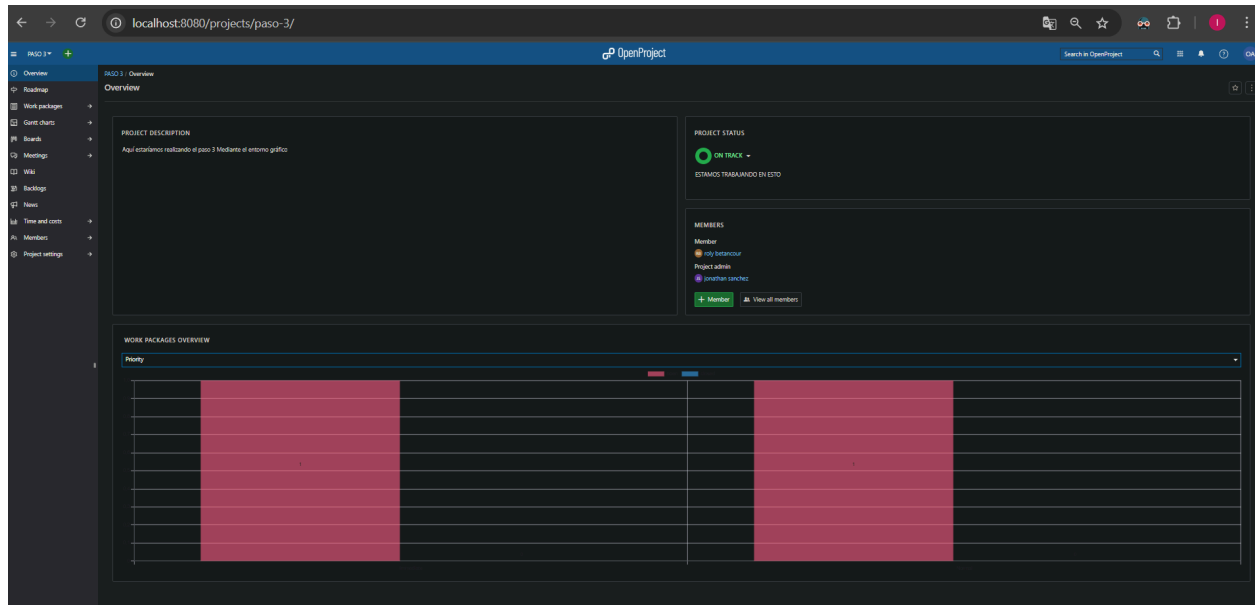
Al final, si queremos apagarlo lo hacemos desde el programa del docker o escribimos en el terminal.

```
docker stop openproject
```

PASO 3 – Introducción de datos

Mediante el entorno gráfico

Esta sería la primera página del proyecto donde vemos lo añadido, desde los miembros que hemos invitado/creado las tareas según su prioridad y el título junto a la descripción del trabajo.



Estas serían las tareas con un mayor detalle para ver ya que le podemos añadir el tiempo que llevan, la prioridad...

localhost:8080/projects/paso-3/work_packages/37/activity

OpenProject

Search in OpenProject

PASO 3

Work packages

Set parent +

TASK Terminar paso 3

In progress #37: Created by OpenProject Admin. Last updated on 05/20/2025 9:33 AM.

Hay que termina el paso 3 de las actividades y documentarlo.

PEOPLE

Assignee: roly betancour Accountable: -

ESTIMATES AND PROGRESS

Work	2h	Remaining work	1h
% Complete	50%	Spent time	0h

DETAILS

Priority: Immediate Date: 05/20/2025 - 06/09/2025

Category: - Version: 1

COSTS

Spent units	-	Labor costs	-
Unit costs	-	Overall costs	-

ACTIVITY FILES RELATIONS WATCHERS (1) MEETINGS

Show everything Newest at the bottom

OpenProject Admin created this on 05/20/2025 08:37 AM #1

OpenProject Admin 05/20/2025 09:33 AM #2

Finish date set to 06/09/2025

Start date set to 05/20/2025

Duration set to 15 days

Add a comment. Type @ to notify people.

Por último de forma gráfica también hemos añadido una tabla donde puedes ver las tareas en general ya que la hemos creado y añadido ahí esas mismas tareas.

localhost:8080/projects/paso-3/boards/10

OpenProject

Search in OpenProject

PASO 3

Boards

primera tabla

Successful update.

Filter 0

Unnamed list

Unnamed list + Add list to board

In progress #38 - PASO 3 TASK Realizar documentación May 20, 2025 - Jun 12, 2025

In progress #37 - PASO 3 TASK Terminar paso 3 May 20, 2025 - Jun 09, 2025

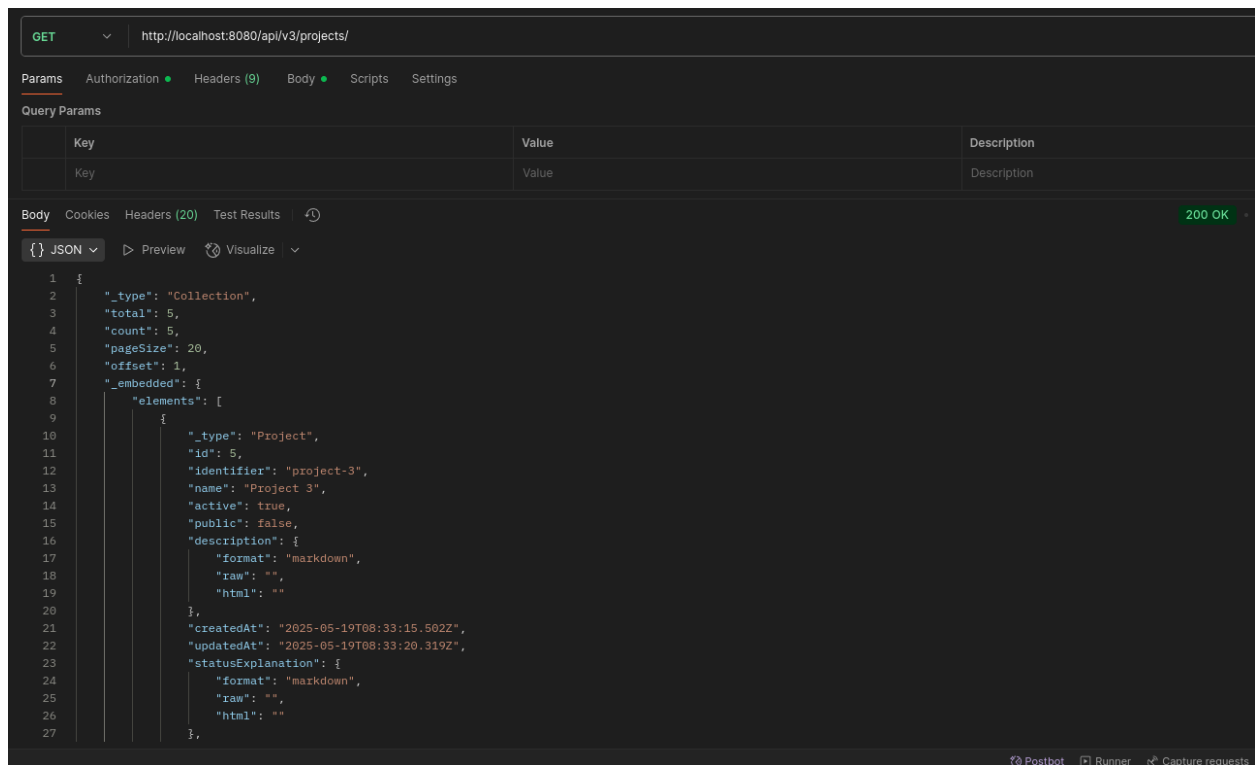
Y este sería el proyecto realizado a través de un entorno gráfico.

PASO 4 – Práctica con peticiones

1. CRUD Básico de Proyectos

Debes realizar todas las prácticas mediante dos métodos, llamada a la API desde Postman y consulta referida en SQL.

1.1. Lista de todos los proyectos



GET <http://localhost:8080/api/v3/projects/>

Params Authorization Headers (9) Body Scripts Settings

Query Params

Key	Value	Description
Key	Value	Description

Body Cookies Headers (20) Test Results 200 OK

JSON Preview Visualize

```
1 {
2   "_type": "Collection",
3   "total": 5,
4   "count": 5,
5   "pageSize": 20,
6   "offset": 1,
7   "_embedded": {
8     "elements": [
9       {
10        "_type": "Project",
11        "id": 5,
12        "identifier": "project-3",
13        "name": "Project 3",
14        "active": true,
15        "public": false,
16        "description": {
17          "format": "markdown",
18          "raw": "",
19          "html": ""
20        },
21        "createdAt": "2025-05-19T08:33:15.502Z",
22        "updatedAt": "2025-05-19T08:33:20.319Z",
23        "statusExplanation": {
24          "format": "markdown",
25          "raw": "",
26          "html": ""
27        }
28      }
29    ]
30  }
```

openProject3 select from projects,									
id	name	active	templated	status_code	description	public	parent_id	created_at	updated_at
lft	rgt					status_explanation	settings		identifier
1	Demo project	t	f	0	This is a short summary of the goals of this demo project.	t		2025-05-19 08:14:32.635018+00	2025-05-19 08:32:46.093321+00
4	Project 2	t	f	3	All tasks are on schedule. The people involved know their tasks. The system is completely set up.	f	1	2025-05-19 08:32:46.064551+00	2025-05-19 08:32:46.093321+00
2	Scrum project	t	f	0	This is a short summary of the goals of this demo Scrum project.	t		2025-05-19 08:14:34.475971+00	2025-05-20 08:22:52.245052+00
5	Prueba 1	t	f	0	All tasks are on schedule. The people involved know their tasks. The system is completely set up.	f	2	2025-05-19 08:31:28.548059+00	2025-05-20 08:22:52.245052+00
6	Project 3	t	f	4		f	3	2025-05-19 08:33:15.502413+00	2025-05-20 08:22:52.245052+00
7	Project 3	t	f	4		f	3	2025-05-19 08:33:15.502413+00	2025-05-20 08:22:52.245052+00

(5 rows)

1.2. Crea un proyecto llamado “Proyecto de Prueba”

The screenshot shows a REST client interface with a POST request to `http://localhost:8080/api/v3/projects`. The request body is a JSON object: `{ "identifier": "proyecto-prueba", "name": "Proyecto de prueba" }`. The response body is a JSON object: `{ "_type": "Project", "id": 6, "identifier": "proyecto-prueba", "name": "Proyecto de prueba", "active": true, "public": false, "description": { "format": "markdown", "raw": "", "html": "" }, "createdAt": "2025-05-20T08:14:07.141Z", "updatedAt": "2025-05-20T08:14:07.245Z", "statusExplanation": { "format": "markdown", "raw": "", "html": "" } }`.

```
1 {
2   "identifier": "proyecto-prueba",
3   "name": "Proyecto de prueba"
4 }
5 }
```

Body Cookies Headers (20) Test Results

{ } JSON Preview Visualize

```
1 {
2   "_type": "Project",
3   "id": 6,
4   "identifier": "proyecto-prueba",
5   "name": "Proyecto de prueba",
6   "active": true,
7   "public": false,
8   "description": {
9     "format": "markdown",
10    "raw": "",
11    "html": ""
12  },
13   "createdAt": "2025-05-20T08:14:07.141Z",
14   "updatedAt": "2025-05-20T08:14:07.245Z",
15   "statusExplanation": {
16     "format": "markdown",
17     "raw": "",
18     "html": ""
19   },
20 }
```

```
openproject=# insert into projects (identifier,name) values ('proyecto-prueba','Proyecto de Prueba');
INSERT 0 1
```

1.3. Obtén los detalles del proyecto que acabas de crear

The screenshot shows a REST client interface with a PATCH request to `http://localhost:8080/api/v3/projects/proyecto-prueba`. The request body is a JSON object:

```
1 {
2   "identifier": "proyecto-editado",
3   "name": "Proyecto Editado"
4 }
5
```

The response is also in JSON format, showing the updated project details:

```
1 {
2   "_type": "Project",
3   "id": 6,
4   "identifier": "proyecto-editado",
5   "name": "Proyecto Editado",
6   "active": true,
7   "public": false,
8   "description": {
9     "format": "markdown",
10    "raw": "",
11    "html": ""
12  },
13   "createdAt": "2025-05-20T08:14:07.141Z",
14   "updatedAt": "2025-05-20T08:21:01.953Z",
15   "statusExplanation": {
16     "format": "markdown",
17     "raw": "",
18     "html": ""
19   },
20 }
```

Below the REST client, a terminal window shows the SQL query and the result of the database update:

```
openproject=# select * from projects where name='Proyecto de Prueba';
 id |      name      | description | public | parent_id | created_at | updated_at | identifier | lft | rgt | active | templated | status_code | status_explanation | settings
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
  8 | Proyecto de Prueba |           | t      |          | 2025-05-20T08:14:07.141Z | 2025-05-20T08:21:01.953Z | proyecto-prueba |    |    | t      | f          | 200         | {}                | {}
(1 row)
```

1.4. Cambia el nombre del proyecto creado en el punto anterior a “Proyecto Editado”

OverviewPATCH http://localhost:8080, [CONFLICT] New Enviro

http://localhost:8080/api/v3/projects/proyecto-prueba

PATCHhttp://localhost:8080/api/v3/projects/proyecto-prueba

ParamsAuthorizationHeaders (9)BodyScriptsSettings

noneform-datax-www-form-urlencodedrawbinaryGraphQLJSON

```
1 {
2   "identifier": "proyecto-editado",
3   "name": "Proyecto Editado"
4 }
5 }
```

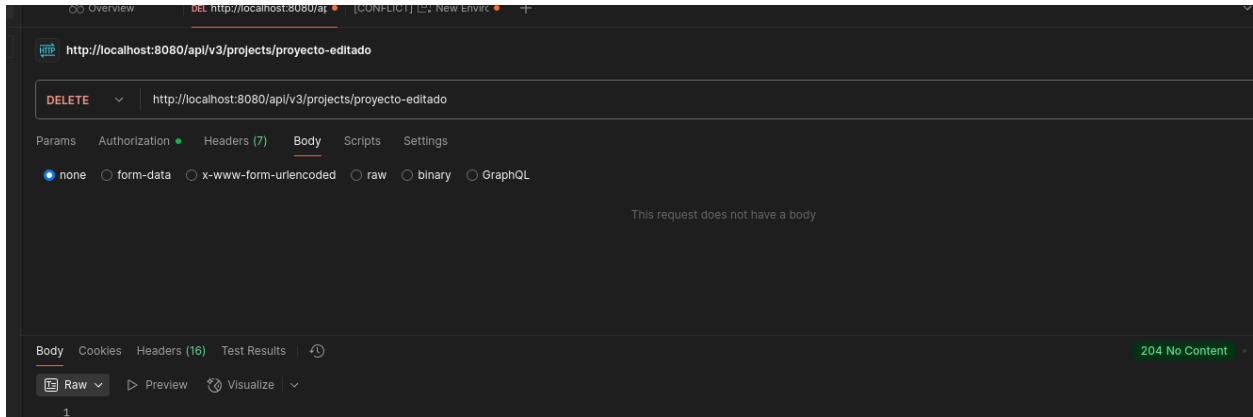
BodyCookiesHeaders (20)Test Results

{ } JSONPreviewVisualize

```
1 {
2   "_type": "Project",
3   "id": 6,
4   "identifier": "proyecto-editado",
5   "name": "Proyecto Editado",
6   "active": true,
7   "public": false,
8   "description": {
9     "format": "markdown",
10    "raw": "",
11    "html": ""
12  },
13   "createdAt": "2025-05-20T08:14:07.141Z",
14   "updatedAt": "2025-05-20T08:21:01.953Z",
15   "statusExplanation": {
16     "format": "markdown",
17     "raw": "",
18     "html": ""
19   },
20   "_links": {
21     "self": {
22       "href": "/api/v3/projects/6",
23       "name": "Proyecto Editado"
24     }
25   }
26 }
```

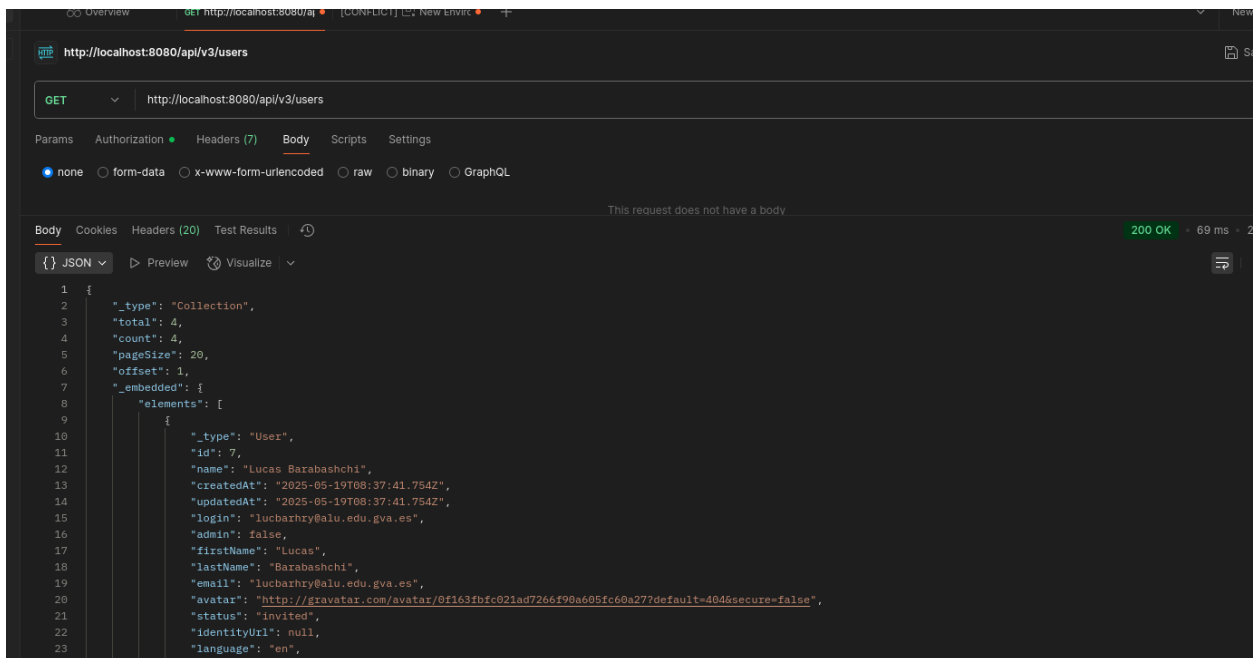
```
openproject=# update projects set name='Proyecto Editado' where identifier='proyecto-prueba';
UPDATE 1
openproject=# select * from projects where name='Proyecto Editado';
 id |      name      | description | public | parent_id | created_at | updated_at | identifier | lft | rgt | active | templated | status_code | status_explanation | settings
-----+-----+-----+-----+-----+-----+-----+-----+----+----+-----+-----+-----+-----+-----+-----
  8 | Proyecto Editado |           | t      |          |           |           | proyecto-prueba |    |    | t      | f          |           |                   | {}
(1 row)
```

1.5. Elimina el proyecto creado



```
openproject=# delete from projects where name='Proyecto Editado';
DELETE 1
openproject=# select * from projects where name='Proyecto Editado';
 id | name | description | public | parent_id | created_at | updated_at | identifier | lft | rgt | active | templated | status_code | status_explanation | settings
-----+-----+-----+-----+-----+-----+-----+-----+----+----+-----+-----+-----+-----+-----+-----
(0 rows)
```

1.6. EXTRA - Lista todos los usuarios de Open Project



OpenProject= select * from projects order by created_at;																				
id		name				description		public		parent_id		created_at		updated_at		identifier				
		lft		rgt		active		templated		status_code				status_explanation						

1		Demo project				This is a short summary of the goals of this demo project.		t				2025-05-19 08:14:32.635018+00		2025-05-19 08:32:46.093321+00		demo-project				

ct	5		10		t		f		0		All tasks are on schedule. The people involved know their tasks. The system is completely set up.		2025-05-19 08:12:34.475971+00		2025-05-20 08:22:52.245052+00		your-scrum-proje			
	3		Prueba 1						f		f		2		2025-05-19 08:31:28.548059+00		2025-05-20 08:22:52.245052+00		prueba-1	
	4		Project 2						f		f		1		2025-05-19 08:32:46.064551+00		2025-05-19 08:32:46.093321+00		project-2	
	2		3		f		3		f		f		3		2025-05-19 08:33:15.502413+00		2025-05-20 08:22:52.245052+00		project-3	
5		Project 3						f		f		3		2025-05-19 08:33:15.502413+00		2025-05-20 08:22:52.245052+00		project-3		
7		8		f		4		f				{}								
(5 rows)																				

2.2. Lista los proyectos ordenados por fecha de edición (de nuevo a viejo)

GET http://localhost:8080/api/v3/projects?sortBy=[["updatedAt","desc"]]

Params Authorization Headers (7) Body Scripts Settings

none form-data x-www-form-urlencoded raw binary GraphQL

This request does not have a body

200 OK 29 ms

```
{
  "_type": "Collection",
  "total": 5,
  "count": 5,
  "pageSize": 20,
  "offset": 1,
  "_embedded": {
    "elements": [
      {
        "_type": "Project",
        "id": 5,
        "identifier": "project-3",
        "name": "Project 3",
        "active": true,
        "public": false,
        "description": {
          "format": "markdown",
          "raw": "",
          "html": ""
        }
      }
    ]
  }
}
```

```
openproject=# select * from projects order by updated_at desc;
 id | name | lft | rgt | active | templated | status_code | description | public | parent_id | status_explanation | created_at | updated_at | identifier
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
  2 | Scrum project | This is a short summary of the goals of this demo Scrum project. | t |  |  | 0 | All tasks are on schedule. The people involved know their tasks. The system is completely set up. | f |  |  | 2025-05-19 08:14:34.475971+00 | 2025-05-20 08:22:52.245052+00 | your-scrum-proje
  3 | Prueba 1 |  | f |  |  | 0 |  | f |  |  | 2025-05-19 08:31:28.548059+00 | 2025-05-20 08:22:52.245052+00 | prueba-1
  5 | Project 3 |  | f |  |  | 0 |  | f |  |  | 2025-05-19 08:33:15.502413+00 | 2025-05-20 08:22:52.245052+00 | project-3
  7 | Demo project | This is a short summary of the goals of this demo project. | t |  |  | 0 | All tasks are on schedule. The people involved know their tasks. The system is completely set up. | f |  |  | 2025-05-19 08:14:32.635018+00 | 2025-05-19 08:32:46.093321+00 | demo-project
  4 | Project 2 |  | f |  |  | 0 |  | f |  |  | 2025-05-19 08:32:46.064551+00 | 2025-05-19 08:32:46.093321+00 | project-2
(5 rows)
```

2.3. Lista los proyectos ordenados por orden alfabético

http://localhost:8080/api/v3/projects?sortBy=[{"name","asc"]

GET http://localhost:8080/api/v3/projects?sortBy=[{"name","asc"]

Params Authorization Headers (7) Body Scripts Settings

none form-data x-www-form-urlencoded raw binary GraphQL

This request does not have a body

Body Cookies Headers (20) Test Results Visualize

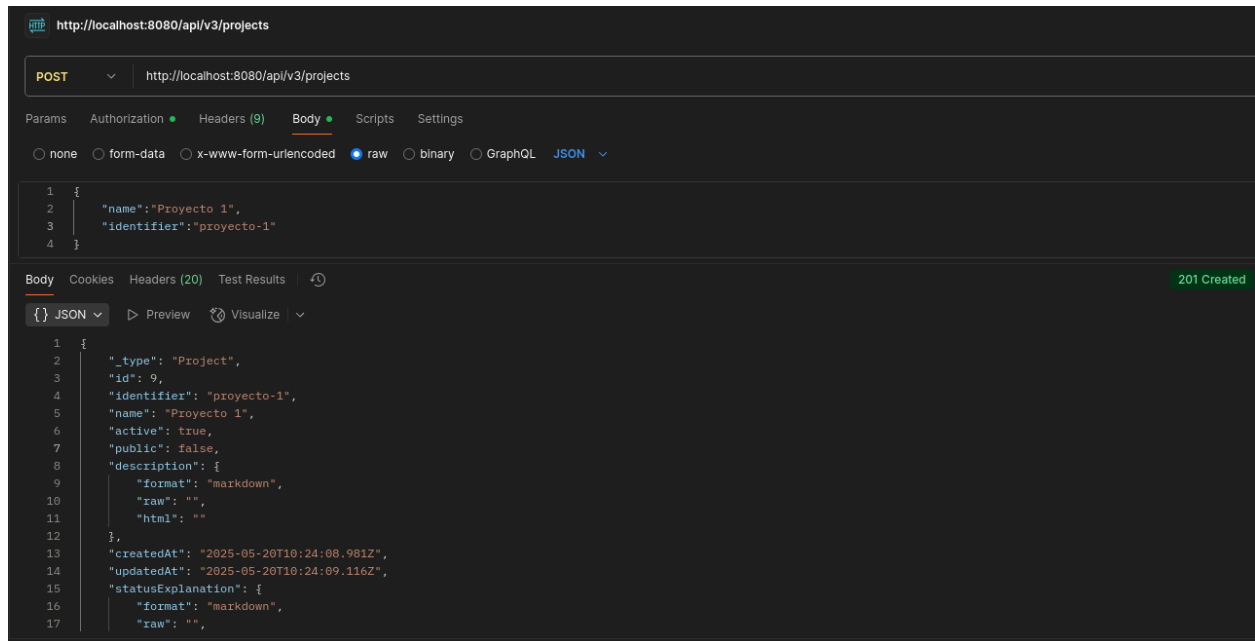
JSON

```
1 {
2   "_type": "Collection",
3   "total": 5,
4   "count": 5,
5   "pageSize": 20,
6   "offset": 1,
7   "_embedded": {
8     "elements": [
9       {
10        "_type": "Project",
11        "id": 5,
12        "identifier": "project-3",
13        "name": "Project 3",
14        "active": true,
15        "public": false,
16        "description": {
17          "format": "markdown",
18          "raw": "",
19          "html": ""
20        },
21        "createdAt": "2025-05-10T08:33:15.502Z"
```

```
openproject=# select * from projects order by name;
 id | name | lft | rgt | active | templated | status_code | description | public | parent_id | status_explanation | created_at | updated_at | identifier
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----
 1 | Demo project | 1 | 4 | t | f | 0 | This is a short summary of the goals of this demo project. | t |  |  | 2025-05-19 08:14:32.635018+00 | 2025-05-19 08:32:46.093321+00 | demo-project
 2 | Project 2 | 2 | 3 | t | f | 3 | All tasks are on schedule. The people involved know their tasks. The system is completely set up. | f | 1 |  | 2025-05-19 08:32:46.064551+00 | 2025-05-19 08:32:46.093321+00 | project-2
 3 | Project 3 | 7 | 8 | t | f | 4 |  | f | 3 |  | 2025-05-19 08:33:15.502413+00 | 2025-05-20 08:22:52.245052+00 | project-3
 4 | Prueba 1 | 6 | 9 | t | f | 0 |  | f | 2 |  | 2025-05-19 08:31:28.548059+00 | 2025-05-20 08:22:52.245052+00 | prueba-1
 5 | Scrum project | 5 | 10 | t | f | 0 | This is a short summary of the goals of this demo Scrum project. | t |  |  | 2025-05-19 08:14:34.475971+00 | 2025-05-20 08:22:52.245052+00 | your-scrum-proje
(5 rows)
```


3. Consultas con Filtros

3.1. Crea un proyecto llamado “Proyecto 1”



REST client interface showing a POST request to `http://localhost:8080/api/v3/projects`. The request body is a JSON object:

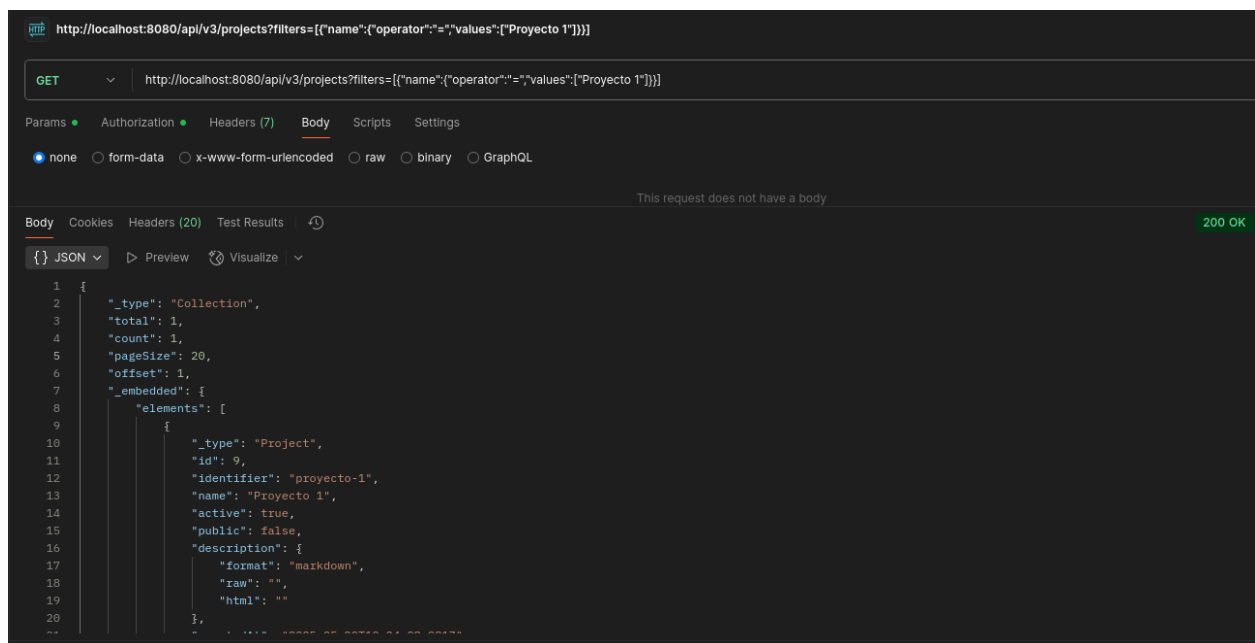
```
1 {
2   "name": "Proyecto 1",
3   "identifier": "proyecto-1"
4 }
```

The response is a 201 Created status. The response body is a JSON object:

```
1 {
2   "_type": "Project",
3   "id": 9,
4   "identifier": "proyecto-1",
5   "name": "Proyecto 1",
6   "active": true,
7   "public": false,
8   "description": {
9     "format": "markdown",
10    "raw": "",
11    "html": ""
12  },
13   "createdAt": "2025-05-20T10:24:08.981Z",
14   "updatedAt": "2025-05-20T10:24:09.116Z",
15   "statusExplanation": {
16     "format": "markdown",
17     "raw": "",
18     "html": ""
19   }
20 }
```

```
openproject=# insert into projects(name,identifier) values ('Proyecto 1','proyecto-1');
INSERT 0 1
```

3.2. Lista todos los proyectos llamados “Proyecto 1”



REST client interface showing a GET request to `http://localhost:8080/api/v3/projects?filters=[{"name":{"operator":"=","values":["Proyecto 1"]}}]`. The response is a 200 OK status. The response body is a JSON object:

```
1 {
2   "_type": "Collection",
3   "total": 1,
4   "count": 1,
5   "pageSize": 20,
6   "offset": 1,
7   "_embedded": {
8     "elements": [
9       {
10        "_type": "Project",
11        "id": 9,
12        "identifier": "proyecto-1",
13        "name": "Proyecto 1",
14        "active": true,
15        "public": false,
16        "description": {
17          "format": "markdown",
18          "raw": "",
19          "html": ""
20        }
21      }
22    ]
23  }
24 }
```

```

openproject=# select * from projects where name='Proyecto 1';
-[ RECORD 1 ]-----+-----
id                | 11
name              | Proyecto 1
description       |
public           | t
parent_id        |
created_at       |
updated_at       |
identifier        | proyecto-1
lft              |
rgt              |
active           | t
templated        | f
status_code      |
status_explanation |
settings         | {}

```

3.3. Crea un par de tareas

The screenshot shows a REST client interface with a POST request to `http://localhost:8080/api/v3/projects/proyecto-1/work_packages`. The request body is a JSON object: `{ "subject": "Creación 1" }`. The response status is `201 Created`. The response body is a JSON object with the following structure:

```

{
  "href": "/api/v3/work_packages/39/summarized_costs_by_type",
  "subject": "Creación 1",
  "description": {
    "format": "markdown",
    "raw": "",
    "html": ""
  },
  "scheduleManually": true,
  "startDate": null,
  "dueDate": null,
  ...
}

```

http://localhost:8080/api/v3/projects/proyecto-1/work_packages

POST http://localhost:8080/api/v3/projects/proyecto-1/work_packages

Params Authorization Headers (9) Body Scripts Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL JSON

```
1 {
2   "subject": "Creación 2"
3 }
```

Body Cookies Headers (20) Test Results 201 Created

JSON Preview Visualize

```
210     "href": "/api/v3/work_packages/40/summarized_costs_by_type"
211   }
212 }
213 }
214 },
215 "_type": "WorkPackage",
216 "id": 40,
217 "lockVersion": 0,
218 "subject": "Creación 2",
219 "description": {
220   "format": "markdown",
221   "raw": "",
222   "html": ""
223 },
224 "scheduleManually": true,
225 "startDate": null,
226 "dueDate": null,
---
```

```
openproject=# insert into work_packages(type_id, project_id,subject,status_id,author_id) values (1,11,'Tarea Prueba 1',1,4);
INSERT 0 1
openproject=# insert into work_packages(type_id, project_id,subject,status_id,author_id) values (1,11,'Tarea Prueba 2',1,4);
INSERT 0 1
openproject=# select * from work_packages where project_id=11;
-[ RECORD 1 ]-----+-----
id              | 41
type_id         | 1
project_id      | 11
subject         | Tarea Prueba 1
description     |
due_date        |
category_id     |
status_id       | 1
assigned to id  |
```

3.4. Lista todas las tareas activas

The screenshot shows a REST client interface with a GET request to `http://localhost:8080/api/v3/work_packages?filters=[{"status":{"operator":"=","values":["1","7","13"]}}]`. The response is a JSON object with the following structure:

```
{
  "category": {
    "href": null
  },
  "type": {
    "href": "/api/v3/types/3",
    "title": "Summary task"
  },
  "priority": {
    "href": "/api/v3/priorities/8",
    "title": "Normal"
  },
  "project": {
    "href": "/api/v3/projects/1",
    "title": "Demo project"
  },
  "status": {
    "href": "/api/v3/statuses/7",
    "title": "In progress"
  },
  "author": {
    "href": "/api/v3/users/4",
    "title": "OpenProject Admin"
  },
  "responsible": {
    "href": null
  },
  "assignee": {

```

```
openproject=# select * from work_packages where status_id IN (select id from statuses where is_closed=false);
-[ RECORD 1 ]-----
id              | 37
type_id         | 1
project_id      | 4
subject         | Primera tarea de prueba
description     |
due_date       |
category_id     |
status_id       | 1
assigned_to_id  | 6
priority_id     | 8
version_id      |
author_id       | 4
lock_version    | 0
done_ratio      |
estimated_hours |
created_at      | 2025-05-20 10:34:41.42785+00
updated_at      | 2025-05-20 10:34:41.497072+00
start_date      |
```

3.5. Lista todas las tareas creadas desde antes del 30 de mayo de 2025

Overview GET http://localhost:8080/api/v3/work_packages?filters=[{"createdAt":{"operator":"<>d","values":["0000-01-01T00:00:00.000Z","2025-05-30T00:00:00.000Z"]}}]

GET http://localhost:8080/api/v3/work_packages?filters=[{"createdAt":{"operator":"<>d","values":["0000-01-01T00:00:00.000Z","2025-05-30T00:00:00.000Z"]}}]

Params Authorization Headers (7) Body Scripts Settings

Body Cookies Headers (20) Test Results 200 OK

{ } JSON Preview Visualize

```
1 {
2   "_type": "WorkPackageCollection",
3   "total": 40,
4   "count": 20,
5   "pageSize": 20,
6   "offset": 1,
7   "_embedded": {
8     "elements": [
9       {
10        "_type": "WorkPackage",
11        "id": 1,
12        "lockVersion": 0,
13        "subject": "Start of project",
14        "description": {
15          "format": "markdown",
16          "raw": "",
17          "html": ""
18        },
19        "scheduleManually": true,
20        "date": "2025-05-18",
21        "estimatedTime": null,
22        "derivedEstimatedTime": null,
23        "derivedRemainingTime": null,
24        "ignoreNonWorkingDays": true,
25        "percentageDone": null,
26        "derivedPercentageDone": null,
27        "createdAt": "2025-05-19T08:14:33.140Z",
28        "updatedAt": "2025-05-19T08:14:33.194Z",
29        "_links": {
30          "attachments": {
31            "href": "/api/v3/work_packages/1/attachments"
32          },
33          "addAttachment": {
```

```
openproject=# select * from work_packages where status_id IN (select id from statuses where is_closed=false);
-[ RECORD 1 ]-----
id              | 37
type_id         | 1
project_id      | 4
subject         | Primera tarea de prueba
description      |
due_date        |
category_id     |
status_id       | 1
assigned_to_id  | 6
priority_id     | 8
version_id      |
author_id       | 4
lock_version    | 0
done_ratio      |
estimated_hours |
created_at      | 2025-05-20 10:34:41.42785+00
updated_at      | 2025-05-20 10:34:41.497072+00
start_date      |
```

3.6. Lista todas las tareas inactivas creadas después del 20 de mayo

Overview GET http://localhost:8080/api/v3/work_packages?filters=[{"status":{"operator":"=","values":["12","14"]}}, {"createdAt":{"operator":">","values":["2025-05-20T00:00:00.000Z","9999-12-31T23:59:59.999Z"]}}]

GET http://localhost:8080/api/v3/work_packages?filters=[{"status":{"operator":"=","values":["12","14"]}}, {"createdAt":{"operator":">","values":["2025-05-20T00:00:00.000Z","9999-12-31T23:59:59.999Z"]}}]

Params Authorization Headers (9) Body Scripts Settings

Key	Value	Description
filters	[{"status":{"operator":"=","values":["12","14"]}}, {"createdAt":{"operator":">","values":["2025-05-20T00:00:00.000Z","9999-12-31T23:59:59.999Z"]}}]	
Key	Value	Description

Body Cookies Headers (20) Test Results 200 OK 70

JSON Preview Visualize

```
1 {
2   "_type": "WorkPackageCollection",
3   "total": 1,
4   "count": 1,
5   "pageSize": 20,
6   "offset": 1,
7   "_embedded": {
8     "elements": [
9       {
10        "derivedStartDate": null,
11        "derivedDueDate": null,
12        "spentTime": "PT0S",
13        "laborCosts": "0.00 EUR",
14        "materialCosts": "0.00 EUR",
15        "overallCosts": "0.00 EUR",
16        "_type": "WorkPackage",
17        "id": 39,
18        "lockVersion": 1,
19        "subject": "Creación 1",
20        "description": {
21          "format": "markdown",
22          "raw": "",
23          "html": ""
24        },
25        "scheduleManually": true,
26        "startDate": null,
27        "dueDate": null,
28        "remainingHours": null
29      }
30    ]
31  }
```

```
openproject=# select * from work_packages where created_at>'2025-05-20 00:00:00.000+00' and status_id IN (Select id from statuses where is_closed=true);
-[ RECORD 1 ]-----
id              | 43
type_id         | 1
project_id      | 4
subject         | pruebas
description      |
due_date        |
category_id     |
status_id       | 12
assigned_to_id  |
priority_id     | 8
version_id      |
author_id       | 4
lock_version    | 1
done_ratio      |
estimated_hours |
created_at      | 2025-05-21 09:21:14.271376+00
updated_at      | 2025-05-21 09:21:20.890782+00
start_date      |
responsible_id  |
budget_id       |
position        | 1
story_points    |
remaining_hours  |
```

PASO 5 – Repositorio en GitHub

Hemos creado un repositorio de Github para poder realizar el proyecto así como gestionar las diferentes versiones y los cambios de los diferentes integrantes.

<https://github.com/jonimax11000/autis>

PASO 6 – Desarrollo de la página web

Desarrollar la página web propuesta anteriormente utilizando llamadas a la API de OpenProject. El ejemplo aportado sirve únicamente como referencia, por lo que podéis mejorarlo, personalizar el diseño o añadir funcionalidades adicionales si lo deseáis.