

# Sprint 1

# Objectif

- Finir le backlog
- Connecter le backend à la base de donnée
- Mise en place de Docker et de Github CI
- Authentification des utilisateurs
- Création de projets
- Récupérer la liste des projets de l'utilisateur

# Ce qu'on a fait

- Finir le backlog
- Connecter le backend à la base de donnée
- Mise en place de Docker et de Github CI
- Authentification des utilisateurs (backend)
- Création de projets (backend)

# Finaliser le backlog

US1 - En tant que visiteur, je souhaite m'enregistrer et me connecter afin de gérer mes projets. (Priorité: Haute | Story point: 5)

- T1.1 - Créer la table "User" dans la base de donnée
- T1.2 - Créer les pages d'inscription et de connexion (email, mot de passe).
- T1.3 - Implémenter la logique d'enregistrement et d'authentification
- T1.4 - Permettre la déconnexion de l'utilisateur et la redirection vers la page d'accueil.

# CI

<p>✓ <b>Setup docker and ci</b> CI #5: Pull request <a href="#">#20</a> opened by <a href="#">DamienLubet</a></p>	setup_docker_ci	Nov 3, 9:56 PM GMT+1	...
<p>✓ <b>chore: Add docker setup and Readme for docker launch</b> CI #4: Commit <a href="#">33e0078</a> pushed by <a href="#">DamienLubet</a></p>	setup_docker_ci	Nov 3, 9:52 PM GMT+1	...
<p>✓ <b>correct CI</b> CI #3: Commit <a href="#">9598119</a> pushed by <a href="#">DamienLubet</a></p>	setup_docker_ci	Nov 3, 12:07 PM GMT+1	...
<p>✗ <b>Add CI workflow for backend and frontend tests</b> CI #2: Commit <a href="#">a35323c</a> pushed by <a href="#">DamienLubet</a></p>	setup_docker_ci	Nov 3, 12:02 PM GMT+1	...
<p>✗ <b>Add CI workflow for backend and frontend tests</b> CI #1: Commit <a href="#">e85468f</a> pushed by <a href="#">DamienLubet</a></p>	master	Nov 3, 11:52 AM GMT+1	...

# Authentification - Register

```
### Authentication USERS

### Register
Send Request
POST http://0.0.0.0:8080/api/auth/register
Content-Type: application/json

{
  "username": "test3user",
  "email": "test3@example.com",
  "password": "password123"
}

1  HTTP/1.1 201 Created
2  X-Powered-By: Express
3  Content-Type: application/json; charset=utf-8
4  Content-Length: 58
5  ETag: W/"3a-LMKNdruuDQfMH2Vz8z0Uh+zYKzY"
6  Date: Wed, 05 Nov 2025 12:15:41 GMT
7  Connection: close
8
9  {
10    "success": true,
11    "message": "User registered successfully."
12 }
```

# Authentification - Login

```
### Login
Send Request
POST http://0.0.0.0:8080/api/auth/login
Content-Type: application/json

{
  "email": "test2@example.com",
  "password": "password123"
}
```

```
HTTP/1.1 200 OK
X-Powered-By: Express
Content-Type: application/json; charset=utf-8
Content-Length: 336
ETag: W/"150-9Tb6xoEDDFcYu2rxRoOglodptc"
Date: Wed, 05 Nov 2025 12:19:08 GMT
Connection: close

{
  "success": true,
  "data": {
    "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6IjY5MDI4Yzg0OGI2ZjY1NGU0NDkwMjg3ZiIsImVtYWlsIjoidGVzdDJAZXhbXBsZS5jb20iLCJpYXQiOjE3NjIzMjUxNDgsImV4cCI6MTc2MjQzMjU0OH0.SAJEqg8XOBdfwwwedZ_Yn0aQcOxWUo4lhE93rB888ZE",
    "user": {
      "id": "69028c848b6f654e4490287f",
      "username": "test2user",
      "email": "test2@example.com"
    }
  }
}
```

# Créer un projet

```
### Create Project
Send Request
POST http://0.0.0.0:8080/api/projects
Content-Type: application/json
Authorization: Bearer {{token}}


{
    "name": "Project Test"
}
```

```
HTTP/1.1 201 Created
X-Powered-By: Express
Content-Type: application/json; charset=utf-8
Content-Length: 58
ETag: W/"3a-WFECT4qydpbIansyMsrHL1/cj4A"
Date: Wed, 05 Nov 2025 12:21:27 GMT
Connection: close

{
    "success": true,
    "message": "Project created successfully."
}
```

# Ce qui n'a pas été fait

- Authentification des utilisateurs (frontend)
- Création de projets (frontend)
- Récupérer la liste des projets de l'utilisateur