

**3A - PM2**

**BRODA** Lou  
**DA COSTA CUNHA** Bruno  
**FRANCO** Nicolas  
**LEVADOUX** Tim



**UNIVERSITÉ**  
**Clermont**  
**Auvergne**

**BUT Informatique**



**< Dev Garden >**



# Contexte

N'en avez-vous pas assez de regarder tous les jours vos différentes plateformes de développement collectif telles que GitHub et GitLab, pour connaître les avancés et ce qu'il reste à faire.

Cela semble rapide quand on travaille sur deux ou trois projets et sur seulement deux plateformes. Mais cela devient vite infernal et interminable et encore plus pour la gestion projet.

Si je vous dis que vous pourriez parrainer ou tutorer un projet qui regroupe vos plateformes en une application.

Pour répondre à cette problématique, nous aimerions développer, "DevGarden". Le développement de ce dernier serait conduit par 4 étudiants du BUT Informatiques de Clermont-Ferrand :

- Da Costa Cunha Bruno
- Broda Lou
- Franco Nicolas
- Levadoux Tim

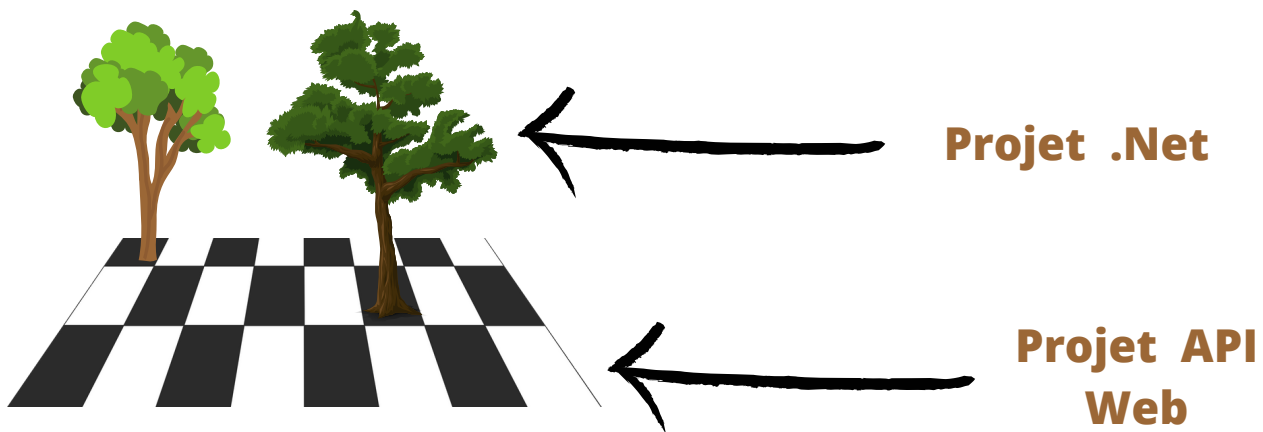
C'est le projet principal et central de leur 3ème et dernière année pour l'obtention de leur diplôme.

Nous vous invitons à lire la suite pour découvrir l'idée derrière DevGarden en espérant vous donnez envie de nous suivre dans le son developpement.



# Présentation du projet

Le but principal de l'application est de rassembler le contenu des différentes plateformes de développement collectif en une seule application. L'application sera représentée sous forme de jardins où chaque jardin représente une plateforme. Dans un jardin, on peut alors choisir un arbre qui représente un projet de cette dernière.



## Exemple : Jardin GitHub

Dès que sélectionne un arbre, on arrive alors dans son espace, c'est-à-dire qu'on rentre dans le projet. C'est ici qu'on retrouve les différentes données présentes dans le projet. Dans l'espace de l'arbre, on retrouve :




- **Une pancarte** qui, lorsque l'on appuie dessus, permet de retrouver les informations générales du projet : participants, langages de programmations, nombre de branches et fichiers, retour CI et qualimétrie de code ...

- **Un épouvantail** qui, lorsque l'on appuie dessus, permet de retrouver la gestion de projet présente telles que le WBS, Kanban, Pert ...



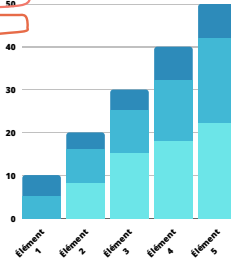
- **Un oiseau** qui, lorsque l'on appuie dessus, affiche le Kanban du projet qu'on peut alors mettre à jour ou ajouter des tickets.
- **Un soleil ou une lune** selon l'heure de la journée où lorsque l'on appuie dessus, on retrouve la gestion de projet présente telles que le WBS, Kanban, Pert ...
- **Un oiseau** qui, lorsque l'on appuie dessus, affiche un agenda avec différentes fonctionnalités telles qu'une Todo List ou des graphismes ...



Mon Projet			
	<b>Dart'agnan</b>	<b>C#</b>	<b>Branch Number</b> <b>3</b>
	<b>LinuxFirst</b>	<b>Java</b>	
<b>CI</b> 	<b>File Number</b> <b>15</b>	<b>Code Smells ..... B</b> <b>Quality Code ..... A</b> <b>Dupplication code .. A</b>	



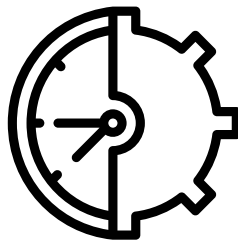
**Dimanche 17 Novembre**



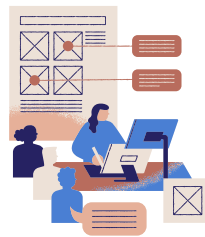
**WBS**

- \_\_\_\_\_
- \_\_\_\_\_
- \_\_\_\_\_
- \_\_\_\_\_

**GANTT**



**PERT**



**1 | TO-DO →**

Schedule publishing via Canva's Content Planner

**2 | IN PROGRESS →**

Finalize a new content release schedule

Collaborate for the writing and design of new posts on Canva

**3 | DONE →**

Perform a content audit

Consolidate recommendations

Brainstorm for next season



# Axes de travail

Nos différents axes de travail sont :

- La récupération des données des plateformes grâce à leur API.
- Développement d'une interface graphique esthétique, agréable ainsi qu'interactive.
- La mise en place d'une base de données pour la connexion au compte ainsi que le stockage des différentes données de configuration...
- La mise à disposition d'outils de gestion de projet comme le WBS, le GANTT ou le Kanban.

Au début, nous utiliserons l'API REST GitHub pour la même plateforme ainsi que l'API Gitlab. Une fois les données récupérées, nous devrons les traiter, les trier ainsi que les centraliser en un type de données pour faciliter leurs utilisations au sein de l'application.

En ce qui concerne l'affichage graphique, le but est à la fois d'avoir une interface facile d'utilisation et intuitive, mais aussi quelque chose de visuellement beau, agréable et ludique, d'où l'idée de verdure.



# Langages de programmation

Pour projet, nous avons pensé à deux langages de programmation pour mobiles :

- **React Native**
- **Flutter**

**React Native** est un framework développé par Meta qui permet de développer des applications en Javascript. Le framework permet le développement d'applications sur Android, IOS, MacOS et Windows. Cependant, le projet communautaire React Native Skya permet de développer sur Linux.

**Flutter** est un framework développé par Google qui permet de développer des applications en Dart. Le framework permet le développement d'applications sur Android, IOS, MacOS, Windows et Linux.

Le tableau ci-dessous représente un comparatif entre flutter et React Native d'après l'article "React Native vs. Flutter: A Quick Analysis of Framework Benefits for App Development in 2023" du blog "Bits and Pieces" datant de juillet 2023



	Flutter	React Native
Performance	Légèrement plus rapide	
Langage de programmation		Plus grandes communautés ...
Architecture	<i>BLoC : simple, rapide, testable</i>	
Installation		Plus rapide : Node Package Manager
UI & développement d'API	UI Toolkit & Natif compatible avec Google	
Réutilisabilité du code	code plus utilisable	
Release	plus souvent mis a jour	
Temps de développement		Beaucoup de ressources disponibles

