

Léonard de Vinci



CNRS



# Octopus

Rapport de stage

Compte rendu réalisé par :  
**THEVARANCHAN** Devamadushan

**Périodes de stage:**  
11/12/2023 .. 22/12/2023  
08/01/2024 .. 09/02/2024

Formation:  
BTS SIO 2eme année

Tuteur de stage:  
Mr Chollet

Contexte : .....	2
Introduction : .....	2
Stage au CNRS:.....	2
Présentaion de l'entreprise : .....	2
Objectif : .....	3
Les Missions : .....	3
a) Mission 1:Machine virtuelle .....	5
b) Crée un dossier partagé.....	5
Mission 2 : Serveur API REST .....	5
a) API REST .....	5
b) Flask .....	6
c) Posteman.....	6
Comment Interroger les APIs des Ecolabs ? .....	6
Routage : .....	7
Explication du code : .....	8
Mission 3 : Base de Données .....	8
Structure de la Base de données : .....	8
Comment Communiquer avec la Base de Donnée ? .....	9
Explication du Code : .....	9
Mission 4 : Serveur Web.....	10
Mission 5 : Serveur Bot.....	15
Problème rencontré .....	15
Annexe.....	15
Conclusion .....	15

## Synthèse :

Le rapport de stage se focalise sur les activités entreprises lors du stage en entreprise de deuxième année de BTS SIO, option SLAM. Dans un premier temps, nous explorerons l'organisation au sein de laquelle le stage s'est déroulé. Ensuite, nous décrirons en détail le projet sur lequel nous avons travaillé. Par la suite, nous examinerons minutieusement les tâches réalisées pendant cette période, qui ont conduit à l'obtention des résultats. En conclusion, nous analyserons les leçons apprises de cette expérience. Le rapport sera complété par un glossaire et des documents annexes en fin de document.

Le stage, d'une durée de 7 semaines, a eu lieu à Saint-Pierre-lès-Nemours (77140), précisément au CNRS (Centre National de la Recherche Scientifique), au sein de l'unité CEREEP – Ecotron IleDeFrance. J'ai collaboré avec un autre étudiant stagiaire du BTS SIO SLAM de Melun : Luca.

Notre objectif principal était de concevoir une interface graphique permettant le contrôle à distance des Cellules climatiques.

Pour ce projet, j'étais chargé de mettre en œuvre à la fois le front-end / back-end des services.

Le travail réalisé durant ce stage a donné des résultats très satisfaisants, ouvrant la voie à une mise en production sur site.

## Entreprise:

### Présentation :

Ce stage a été réalisé au CNRS (Centre National de la Recherche Scientifique), plus précisément au sein de l'unité CEREEP – Ecotron IleDeFrance. Cette Unité d'Appui et de Recherche (UAR), créée en 2008 par l'École normale supérieure et le CNRS (<https://www.cereep.bio.ens.psl.eu/>), est une structure d'accueil et d'accompagnement de la recherche expérimentale en écologie, de l'enseignement et de la formation. Son développement s'est inscrit dans le contexte national de promotion des grands équipements de recherche en écologie par l'Institut "Écologie et Environnement" du CNRS (INEE). Les infrastructures de cette unité sont implantées sur la commune de Saint-Pierre-lès-Nemours en Seine-et-Marne. Le domaine de 78 ha comprend des capacités d'hébergement, un bâtiment de recherche moderne, des espaces naturels et 8 plateaux techniques dédiés à la recherche en écologie expérimentale. Le centre coordonne en particulier l'Ecotron IleDeFrance, un dispositif d'ambition internationale permettant de confiner et de manipuler des organismes et des écosystèmes terrestres et aquatiques dans des conditions hautement contrôlées afin de mesurer leur fonctionnement et leur sensibilité aux changements climatiques. Le centre pilote, la Plateforme expérimentale nationale d'écologie aquatique (PLANAQUA) qui comprend un ensemble unique de dispositifs expérimentaux dédiés à l'étude du fonctionnement écologique des systèmes aquatiques en réponses aux perturbations associées aux activités humaines. Les dispositifs expérimentaux du CEREEP - Ecotron IleDeFrance comprennent tout un ensemble de systèmes terrestres ou aquatiques, que l'on classe généralement en trois grandes catégories en fonction de leur taille et de la complexité des ensembles d'organismes vivants qu'ils peuvent accueillir :

- Les microcosmes sont des systèmes expérimentaux de petite taille (souvent quelques litres ou quelques dm<sup>2</sup>), comme des flacons, des réacteurs, des pots destinés à recevoir des plantes, des terrariums ou des aquariums. De très nombreux facteurs peuvent être contrôlés et les systèmes expérimentaux peuvent être

facilement répliqués à cette échelle d'étude, mais ces systèmes sont souvent assez peu réalistes.

- Les mésocosmes sont des systèmes expérimentaux de quelques m<sup>3</sup> ou m<sup>2</sup>, qui peuvent abriter des ensembles d'organismes plus complexes que les microcosmes, et constituer des habitats plus réalistes pour ces espèces. Les capacités de contrôle de l'environnement sont généralement moindres que dans les microcosmes, même si certains mésocosmes ont été spécifiquement conçus pour contrôler certains paramètres comme la turbulence, la température de l'eau, ou la teneur de l'air en CO<sub>2</sub> ou en ozone.
- Les macrocosmes sont des systèmes expérimentaux de grande taille (généralement plusieurs centaines de m<sup>3</sup> ou m<sup>2</sup>), telles des parcelles de prairies ou des petits plans d'eau. Ils ont l'avantage d'être très proches d'écosystèmes naturels. En revanche, seuls quelques paramètres généraux peuvent être contrôlés à cette échelle.

Le CEREEP - Ecotron IleDeFrance met à disposition ces équipements aux laboratoires extérieurs. Les plateaux techniques sont ouverts aux équipes aux niveaux national et international. Les équipes de recherche accueillies bénéficient des services mis à leur disposition en termes d'espace de travail, d'un accès aux services communs et à la logistique, de l'appui du personnel technique, et des facilités d'hébergement et de restauration de la station. Les plateformes de l'unité sont fortement insérées dans les dispositifs nationaux de plateformes expérimentales comme le ReNSEE (Réseau National des Stations d'Écologie expérimentale du CNRS) et l'infrastructure AnaEE-France coordonnée par le CNRS et par l'INRAE. Elles participent également à plusieurs réseaux internationaux d'infrastructures.

Il existe au total sur le site , 18 cellule climatique :

- 5 ECOLAB comportant 3 cellules.
- 1 ECOCUBE comportant 3 cellules.

Chaque cellule peut générer un climat différent; ce climat comprend plusieurs paramètres : **température, hygrométrie, concentraion de CO<sub>2</sub>**, etc..

Il peut y avoir plusieurs expériences qui se déroulent en parallèle sur différentes cellules climatiques.

### Environnement de travail

Au cours de mon stage, j'étais intégré à l'unité CEREEP - Ecotron Île-de-France, située à Saint-Pierre-lès-Nemours. Cette entité est divisée en deux zones distinctes : la zone haute et la zone basse. La zone haute est la zone principale, c'est ici que la plupart des plateformes expérimentales sont concentrées. C'est également ici que travaille mon maître de stage, Simon CHOLLET, qui

est directeur technique de l'Ecotron.

Chaque semaine, nous avons prévu une réunion avec Simon CHOLLET pour discuter des avancées réalisées, des problèmes rencontrés et des futurs développements envisagés. En plus de cela, ils nous ont fourni une salle de travail où nous pouvions progresser dans notre projet en toute tranquillité, ainsi qu'un ordinateur portable et un badge d'accès.

## Projet:

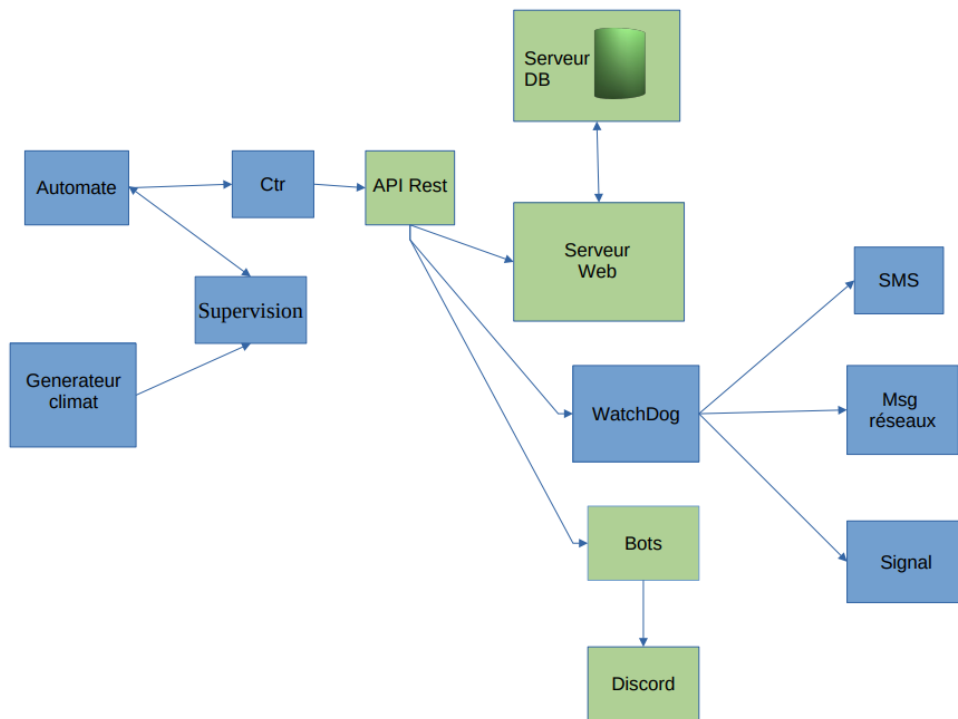
### Présentation:

Le projet Octopus est un projet qui permettra d'avoir au final une vision de l'état des installations techniques de l'Ecotron IDF et dans un premier temps les ECOLAB (cellules climatiques). Actuellement, chaque ECOLAB dispose d'un PC qui permet de dérouler des scénarios climatiques dans chaque cellule.

Ce projet est composé de plusieurs serveurs qui hébergent différents services :

- Serveurs Web.
- Serveurs Base de données.
- Serveurs Bot Discord.
- Serveurs Watchdog.

Le diagramme ci-dessous présente l'architecture système du projet :



Nous sommes intervenus sur les éléments suivants :

- Sereveur API-Rest qui récupère les paramètres en local et les retourne au format JSON, facilitant ainsi l'interrogation et la récupération des données par une autre machine sur le même réseau.
- Serveurs Web qui permet la visualisation en temps réel des températures des cellules climatiques en récupérant les données depuis le serveur API REST. L'administrateur dispose de droits étendus pour une gestion plus complète grâce à le serveur DB.
- Serveurs de base de données pour assurer la sauvegarde complète de toutes les expériences dans chaque cellule, avec un historique détaillé des dates de début et de fin. Ce système de gestion permet à l'administrateur d'attribuer de nouvelles expériences aux cellules en remplaçant l'expérience précédente, et offre des fonctionnalités telles que la consultation, la création, et la modification des expériences, ainsi que la gestion des rôles des utilisateurs.
- Bot Discord pour interroger également le serveur API REST, afin de répondre aux utilisateurs Discord sur les températures en temps réel des cellules climatiques qu'ils souhaitent consulter, ainsi que sur les liens vers nos sites qu'ils désirent obtenir.

## Objectif :

Les objectifs de ce projet sont les suivants :

- Mise en place d'un serveur Web (frontend) permettant la visualisation de l'état des cellules climatiques.
- Mise en place d'un serveur de bases de données (backend) permettant de stocker des données liées aux expériences.
- Mise en place de serveurs avec API REST (backend) pour répondre aux requêtes extérieures.
- Mise en place de clients de types Bot Discord (frontend) pour interroger le serveur.

## Réalisation :

### Téchnologies Utilisées:

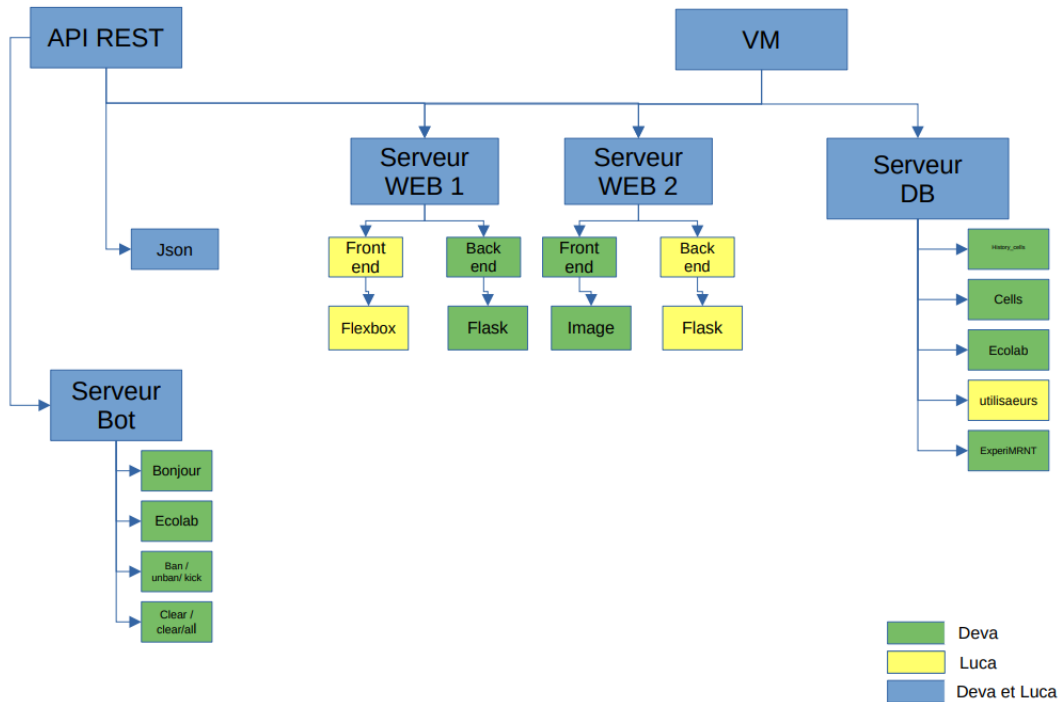
Les différentes technologies utilisées pour la réalisation de ce projet sont les suivantes :

- Langage :
  - Python (ORM,Flask)
  - JavaScript (Js)
  - SQL
- API REST
- Frameworks/Bibliothèque:
  - Flask
  - Bootstrap, CSS, HTML.
- Bases de données:
  - MySQL
- Outils:
  - Postman
  - Git
  - PyCharm

- Visual Studio Code
- VirtualBox
- Postman

### Répartition des taches:

Lors de mon stage, j'avais aussi mon collègue Lucas avec qui j'ai travaillé ensemble sur le projet Octopus. Nous avons réparti les tâches pour progresser dans le projet



Finalement, nous avons créé deux serveurs web, le premier utilisant le Flexbox et le deuxième avec une image pour représenter graphiquement les cellules climatiques.

### Développements:

#### 1 : VM



VM



# VirtualBox

Une machine virtuelle (VM) est un environnement informatique émulé qui fonctionne comme une instance autonome d'un système d'exploitation (OS) au sein d'un système hôte. Elle est créée par un logiciel appelé hyperviseur ou gestionnaire de machines virtuelles.

## *1.a) Installé une Machine virtuelle*

Premièrement, nous avons commencé à installer VirtualBox avec un système d'exploitation Ubuntu.

## *1.b) Crée un dossier partagé*

Deuxièmement nous avons créé un dossier partagé.

Un dossier partagé est un répertoire sur l'hôte physique accessible depuis la machine virtuelle.

Il facilite le transfert de fichiers entre la machine virtuelle et l'hôte, simplifiant ainsi le développement, le test, ou l'accès à des fichiers depuis la VM.

## **2 : Serveur API REST**

Premièrement, nous avons commencé à apprendre les API REST pour faciliter l'échange de données entre le serveur API REST et le serveur web.

## 2.a) API REST

# { REST }

Une API REST repose sur des principes fondamentaux incluant l'architecture orientée ressources avec des identifiants URI, la représentation des données en JSON ou XML, la communication via les méthodes HTTP standard (GET, POST, PUT, DELETE), le principe de communication sans état, et une interface uniforme définissant des conventions pour les URI, les méthodes HTTP et les représentations des ressources.

### Principes de bases

- Architecture Orientée Ressources : Identification des éléments par des URI.
- Représentation des Données : Utilisation de formats standard (JSON, XML) pour représenter les ressources.
- Méthodes HTTP Standard : Utilisation des méthodes GET, POST, PUT, DELETE pour interagir avec les ressources.
- Communication Sans État : Chaque requête contient toutes les informations nécessaires, et le serveur ne garde pas d'état entre les requêtes.
- Interface Uniforme : Définition de conventions pour les URI, les méthodes HTTP et les représentations des ressources.

Dans ce projet, nous avons utilisé uniquement les méthodes HTTP **GET** en format **JSON**

## 2.b) Flask

Deuxièmement, nous avons appris à utiliser Flask pour établir une communication client-serveur



Flask est un framework web en Python. Il simplifie le développement d'applications web en offrant des fonctionnalités de base telles que la gestion des routes, l'utilisation de templates Jinja2 pour le rendu HTML, la manipulation des formulaires, et la gestion des requêtes et réponses HTTP.

### 2.c) Postman

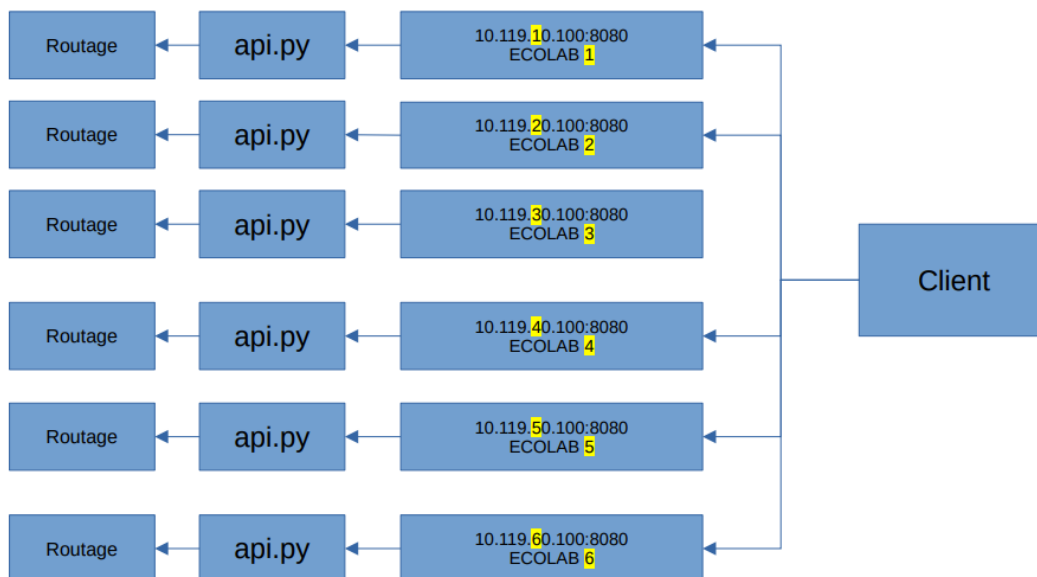
Troisièmement, nous avons utilisé Postman pour tester des requêtes HTTP avec Flask.



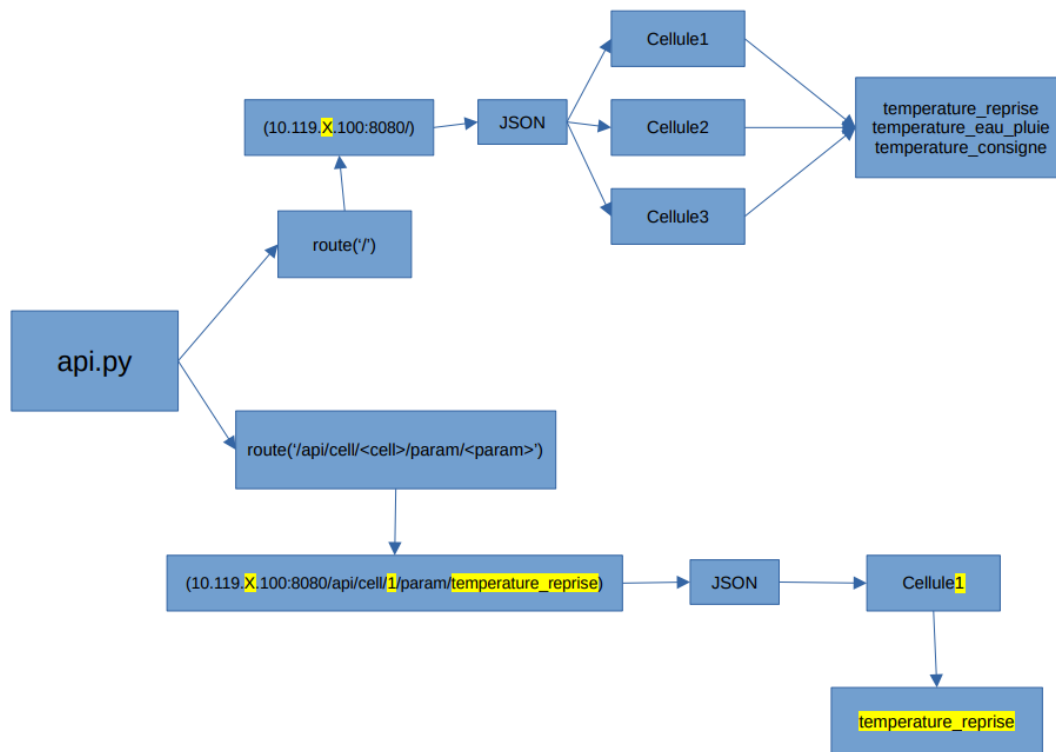
Postman est un outil de collaboration largement utilisé dans le développement d'API. Il permet de créer, tester et documenter des requêtes HTTP, de les organiser dans des collections, et de les regrouper dans des environnements de test.

### Comment Interroger les APIs des Ecolabs ?

Chaque écolab est identifié par une adresse IP. Lorsqu'on interroge l'adresse IP de l'écolab, elle renvoie les paramètres des cellules qu'elle contient.



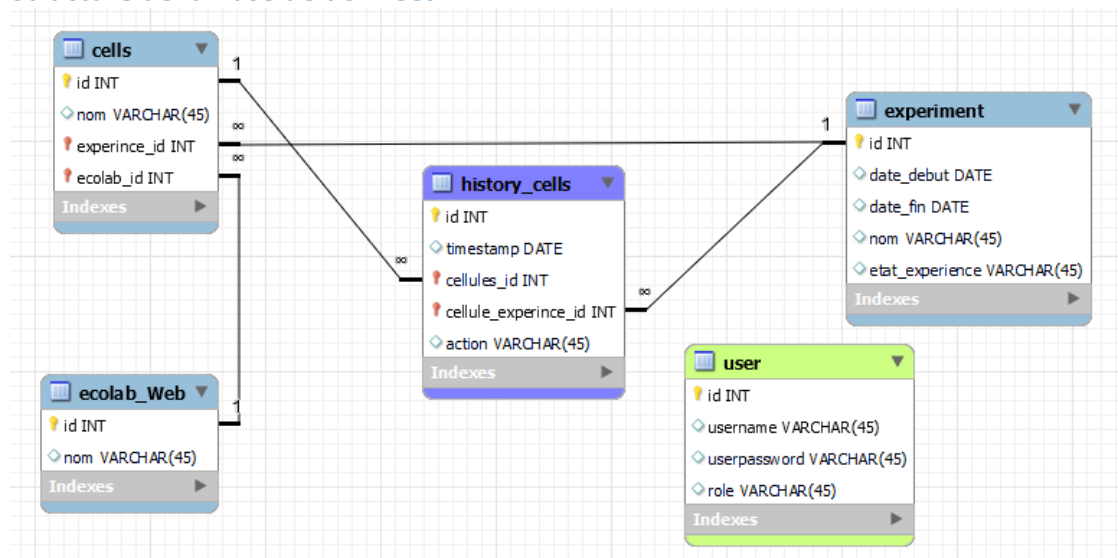
## Routage :



Le fichier **api.py** contient deux routages. Le premier routage à la racine renvoie tous les paramètres de toutes les cellules de l'écolab. Le deuxième routage retourne uniquement le paramètre souhaité d'une seule cellule.

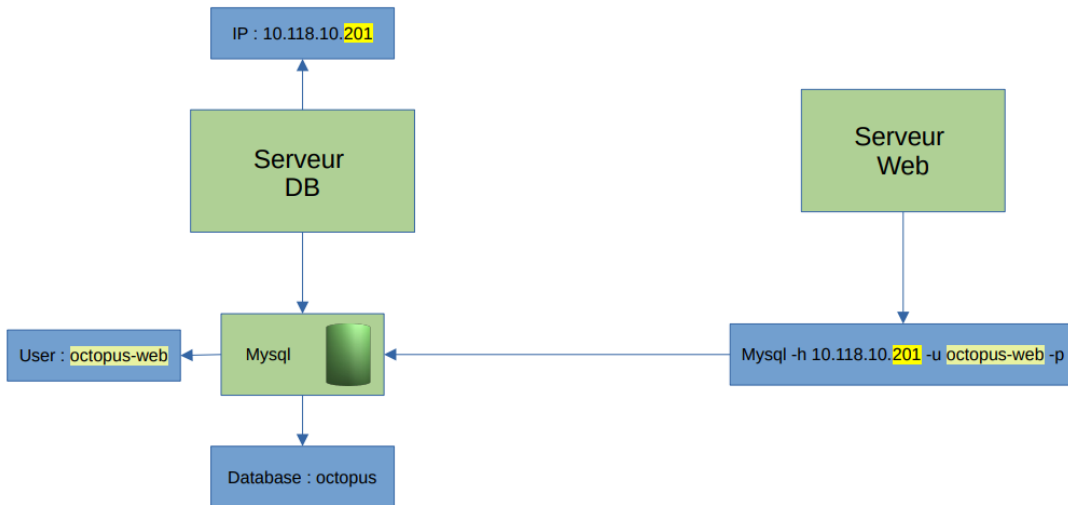
## 3 : Base de Données

### Structure de la Base de données :



### Comment Communiquer avec la Base de Donnée ?

Lorsque vous souhaitez communiquer avec la base de données du serveur DB, il suffit d'ajouter l'adresse IP du serveur DB devant l'utilisateur lorsque vous essayez de vous connecter à la base de données.



### Mission 4 : Serveur Web

#### Partie réalisée par Deva pour le Serveur Web 1

- Les flexbox avec des températures en temps réel (que pour l'ecolab 2) récupérées depuis l'API REST.
- Consulter toutes les expériences, les modifier, ou ajouter une nouvelle expérience à l'aide de la base de données.
- Consulter les détails de chaque cellule pour voir l'historique des expériences de la cellule (la température actuelle uniquement pour l'Ecolab 2) et affecter une nouvelle expérience.
- Les routages nécessaires pour cela.
- Classes : Experiment, Cell, History\_cells, Ecolab\_web , OctopusDB
- Les fonctions nécessaires pour interroger la base de données.

#### Partie réalisée par Luca pour le Serveur Web 1

- Reprise de Flexbox et amélioration.
- la Navbar
- Bouton d'inscription
- Bouton Connexion
- Bouton Utilisateurs
- Footer

- Routage, les fonctions et les classes nécessaires pour la gestion des utilisateurs
- Et également améliorer et modifier mes routages pour les adapter.

#### **Partie réalisée par Deva pour le Serveur Web 2**

- Image responsive pour PC, tablettes, téléphone, en adaptant les boutons pour chaque pixel.
- Les couleurs pour chaque bouton des cellules climatiques : rouge s'il n'y a pas de température, sinon vert.
- La Navbar
- Détails des cellules (en affichant les températures en temps réel)
- Routage pour l'index et détail des cellules

#### **Partie réalisée par Luca pour le Serveur Web 2**

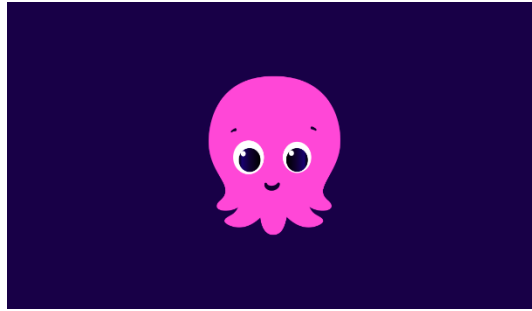
- Bouton d'inscription
- Bouton Connexion
- Bouton Utilisateurs
- Footer
- Tout le backend du serveur web 1 adapté pour ce serveur.

(Annexe 1)

### **Mission 5 : Serveur Bot**

Le serveur bot a été mis en place pour fournir une assistance clientèle instantanée, automatiser les tâches répétitives, engager les utilisateurs avec des informations personnalisées, collecter des données sur les préférences des clients, ainsi que pour offrir des informations sur les cellules climatiques, à propos du CNRS, et bien d'autres sujets encore.

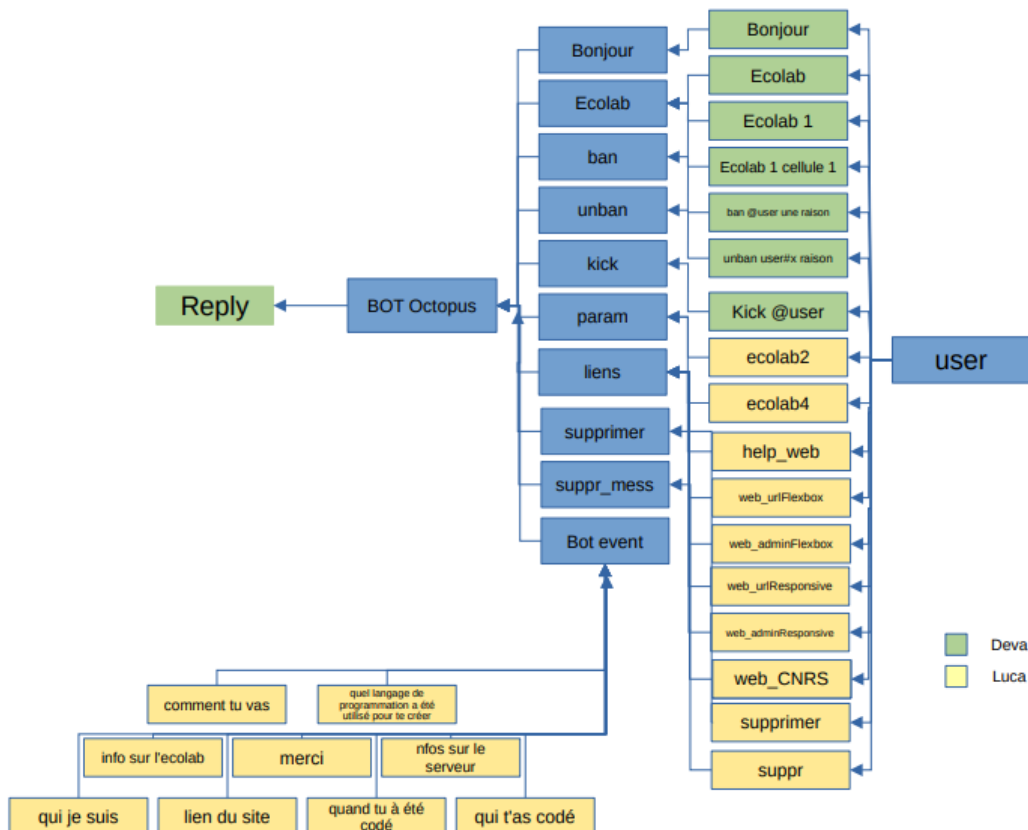
## Bot Octopus:



Le bot Octopus est présent dans le salon API du serveur Octopus sur Discord  
Le bot Octopus a pour but de fournir des informations sur les cellules climatiques, ou sur le CNRS, lorsque un utilisateur de Discord s'y intéresse et demande au bot.

## Les Commandes :

Voici les commandes avec lesquelles on peut interroger le bot:



Voir(1'annexe 2)

## Problème rencontré

Dans ce projet, nous avons été confrontés à plusieurs problèmes qui ont été difficiles à résoudre. Certains d'entre eux n'ont même pas été résolus, et nous allons les examiner un par un.

### L'autonomie

La première problématique à laquelle nous avons été confrontés était celle de devenir autonomes, ce qui diffère considérablement de nos expériences scolaires.

Cependant, grâce à quelques connaissances acquises en cours et à des tutoriels sur YouTube, nous avons réussi à apprendre et à utiliser des outils tels que :

Python, Flask, SQLAlchemy, et les API REST, et autres.

### VM - Dossier Partager

Quand nous avons commencé à travailler sur la machine virtuelle, nous devions créer un dossier partagé pour avoir un espace d'échange entre la machine hôte et la machine virtuelle.

Cependant, après la création du dossier partagé et son association à la machine virtuelle, nous avons rencontré un problème de droits pour créer des fichiers, déplacer des fichiers, des images, etc. sur la machine virtuelle.

Nous avons consulté des forums, des tutoriels sur YouTube, mais nous n'avons pas réussi à résoudre le problème.

### GIT

Pour avoir une sauvegarde à distance, nous avons utilisé Git, mais nous avons rencontré des difficultés pour effectuer des push. Nous avons rencontré des erreurs qui indiquaient que la vérification depuis le terminal n'autorisait pas le push.

Pour résoudre ce problème, nous avons finalement trouvé une solution en créant un token unique pour chaque utilisateur, ce qui a confirmé notre identité et nous a permis d'effectuer le push.

### Base de données

Au commencement du projet, nous n'avons pas eu besoin de créer une base de données.

En effet, au départ du projet, un camarade en BTS SIO option SISR était présent avec nous, et il était chargé d'apprendre et d'utiliser LDAP, puis de



le relier avec notre serveur web pour les vérifications lors de la connexion et sécuriser le serveur web.

Mais malheureusement, cela ne s'est pas déroulé comme prévu, donc nous avons dû résoudre ce problème en créant une base de données nous-mêmes.

### *Communication avec la base de données*

Une fois que j'ai créé la base de données sur un serveur dédié, Lucas a eu du mal à communiquer avec son propre serveur à base de données.

Nous avons commencé à perdre beaucoup de temps dessus et ne trouvions pas de solution.

J'ai donc décidé de prendre en charge la partie front-end pour afficher les expériences et les pages détaillées. C'est pourquoi nous n'avons pas pu respecter la répartition des tâches initiale.

Au final, nous avons fini par trouver une solution avec mon tuteur.

Lorsqu'une personne souhaite communiquer avec la base de données d'un autre serveur, lors de la connexion, elle doit indiquer le nom d'hôte comme ceci :

```
-h 'ip du serveur' -u 'utilisateur' -p.
```

De plus, il ne faut pas oublier qu'au moment de la création de l'utilisateur, il faut créer un utilisateur autre que localhost.

Par exemple : `CREATE USER 'Deva'@'%' IDENTIFIED BY 'sio'.`

Et il faut autoriser le serveur MySQL à permettre la connexion des autres machines. Pour cela, il fallait se rendre dans le fichier MySQL :

- Ouvrir le fichier `my.cnf`
- Ajouter : `Bind-address = 0.0.0.0`

Voir (l'annexe 3)

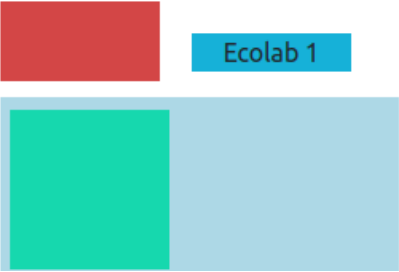
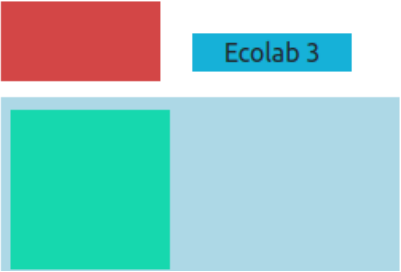
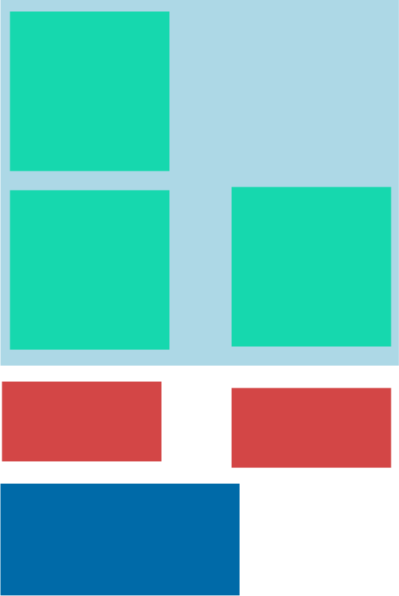
## **Annexes**

(Annexes 1)

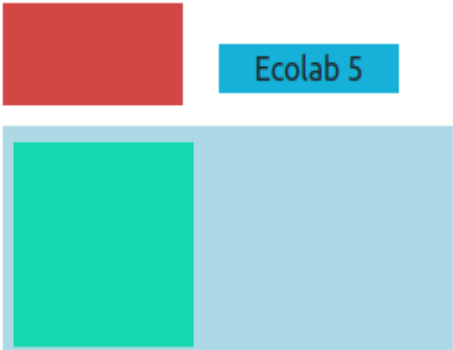
Ordinateur:



Tablette:



Téléphone:



*Ordinateur:*



*Tablette:*



Téléphone:





# Détail de E2C1

Temperature Reprise	18.0
Temperature Eau Pluie	12.0
Temperature Consigne	0





(Annexe 2)

Server Octopus # api 6 février 2024

deva9156 ecolab

@deva9156 ecolab  
Octopus Quel Ecolab voulez-vous voir ?

ECOLAB 1 ECOLAB 2 ECOLAB 3 ECOLAB 4 ECOLAB 5  
ECOLAB 6

Octopus Cliquez pour voir les pièces jointes  
ECOLAB 1

@deva9156 ecolab  
Octopus Quel Cellule de l'Ecolab 1 voulez-vous voir ?

Cellule 1 Cellule 2 Cellule 3

Octopus Cliquez pour voir les pièces jointes  
Cellule 3

@deva9156 ecolab  
Octopus Chargement...

@deva9156 ecolab  
Octopus Quel paramètre de Ecolab 1 Cellule 3, voulez-vous consulter ?

Temperature Reprise Temperature Consigne Temperature Eau Pluie

deva9156 web\_CNRS

Octopus  
CNRS : <https://www.cnrs.fr/fr>  
Facebook: <https://fr-fr.facebook.com/cnrs.fr/>  
Twitter: <https://twitter.com/cnrs?lang=fr&id=20>  
Instagram: <https://www.instagram.com/cnrs/>  
Youtube: <https://www.youtube.com/@CNRS>

Centre national de la recherche scientifique (CNRS)

Centre national de la recherche scientifique (CNRS)

CNRS | Paris  
CNRS, Paris, 206 207 J'aime · 702 en parlent · 1 345 personnes étaient ici. Suivez l'actualité du Centre national de la recherche scientifique (CNRS) <https://cnrs.fr>

CNRS (@CNRS) on X  
Organisme public de recherche pluridisciplinaire, le Centre national de la recherche scientifique c'est 33 000 personnes qui font avancer la connaissance.  
Twitter

Login • Instagram  
Welcome back to Instagram. Sign in to check out what your friends, family & interests have been capturing & sharing around the world.

EN LIGNE - 3  
deva9156  
Octopus  
UPUPIDES

HORS LIGNE - 2  
adithya  
Lucalk



(Annexe 3)

```
#
# This group is read both by the client and the server
# use it for options that affect everything
#
[client-server]
# Port or socket location where to connect
# port = 3306
socket = /run/mysqld/mysqld.sock

# Import all .cnf files from configuration directory
!includedir /etc/mysql/conf.d/
!includedir /etc/mysql/mariadb.conf.d/
#

[mysqld]
bind-address = 0.0.0.0
```

## Conclusion

Durant ce stage, j'ai appris beaucoup de choses intéressantes, telles que la création de bots Discord, la gestion de machines virtuelles, ainsi que le développement de mon autonomie. De plus, nous avons une réunion chaque semaine pour discuter de l'avancement des projets et définir de nouveaux objectifs pour la semaine suivante. En outre, l'environnement de travail était parfait pour maintenir un esprit tranquille.