Dubois Charles, Duchène Nathanaël, Dias Neves Louis-Robbert

**The Louchana High School's problem**

Table des matières:

[Introduction : 3](#_Toc155557047)

[Mise en contexte du besoin client : 3](#_Toc155557048)

[Description détaillée de la solution mise en place : 3](#_Toc155557049)

[Aperçu de l'organisation en tant qu'équipe : 5](#_Toc155557050)

[Gestion du temps et répartition des tâches : 5](#_Toc155557051)

[Portfolio de commandes via GIT : 6](#_Toc155557052)

[Conclusion : 9](#_Toc155557053)

# Introduction :

Une école secondaire nous a contactés afin que nous leur mettions en place un moyen simple, rapide et efficace de gérer les inscriptions dans leurs différentes sections. Mon équipe technique et moi-même nous sommes donc attelés à cette tache afin de leur proposer le service le plus homogène possible.

# Mise en contexte du besoin client :

Notre client (à savoir, la Louchana High School) souhaite une méthode d’inscription dans ses différentes sections plus efficace que celle qu’elle propose actuellement. Les objectifs de base à atteindre sont : que chaque secrétaire doit pouvoir soumettre les inscriptions via un formulaire, validé manuellement par le responsable des inscriptions. Une notification doit alors être reçue par les nouveaux élèves, ainsi qu’aux secrétaires et les informations doivent être encodées dans un fichier Excel. Si l’inscription est refusée, un justificatif devra être fournit.

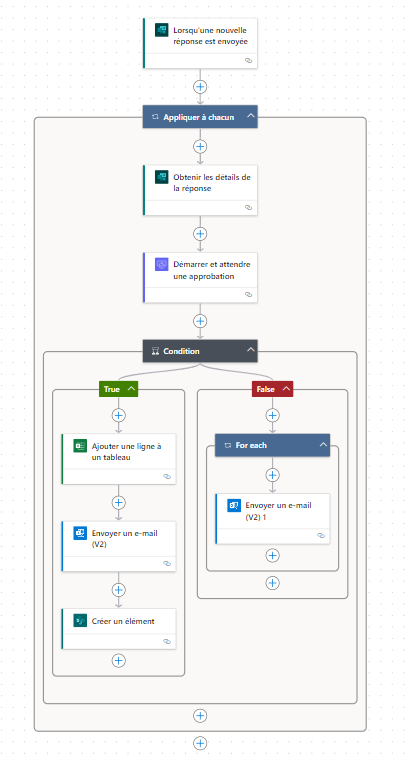
# Description détaillée de la solution mise en place :

Afin de répondre à la demande du client, le responsable en charge de cette étape a utilisé Power Automate pour automatiser le processus d’inscription afin de permettre une meilleure prise en main ainsi qu’une utilisation simplifiée par les secrétaires. Voici comment nous avons procédé.

* Les formulaires d’inscription ou Microsoft Forms :
* Création des formulaires d’inscription via l’outil Microsoft Forms. Formulaires étant accessible aux secrétaires
* Une notification par Courriel aux secrétaires :
* L’une des attentes du client soit que quand une secrétaire soumet une demande d’inscription dans une section via le formulaire, un Email est automatiquement envoyé au responsable des inscriptions afin de lui demander une validation.
* Validation ou refus :
* Dans le mail elle verra les informations du formulaire. Le responsable des inscriptions peut alors cliquer sur l’un deux en fonction de s’il accepte ou non l’inscription.
* Pour les validations :
* Si le responsable des inscriptions valide le formulaire, Power automate ajoute les informations de l’élève automatiquement dans le fichier Excel dédié à cet effet.
* Notification attribuée à l’élève :
* En cas de validation, l’élève reçoit un courriel avec les détails de son inscription
* Envoyer de liste des élèves au professeur le jour de la rentrée :
* Une liste SharePoint qui récupèrera les informations du Excel
* Enverra un mail avec les élèves par classe a leur professeur attitré via une liste a puce
* Pour les refus :
* En cas de refus, l’élève reçoit un email lui expliquant les raisons de ce refus.

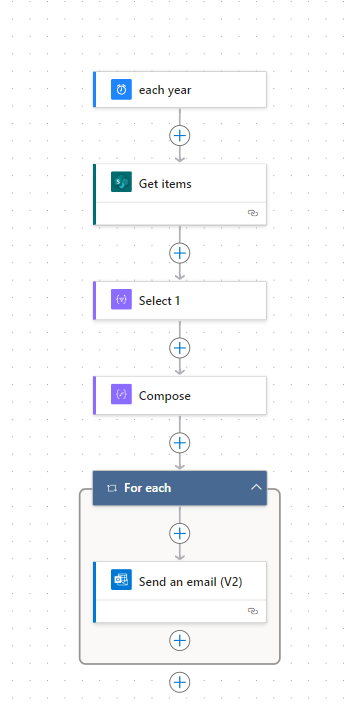
Cette méthodologie permet d’obtenir une inscription simple, efficace et automatisée. Tout ceci en assurant également une communication homogène entre élève et secrétaire. Le responsable des inscriptions a également une meilleure visibilité et compréhension des données grâce à leurs centralisations dans des listes Excel structurées.

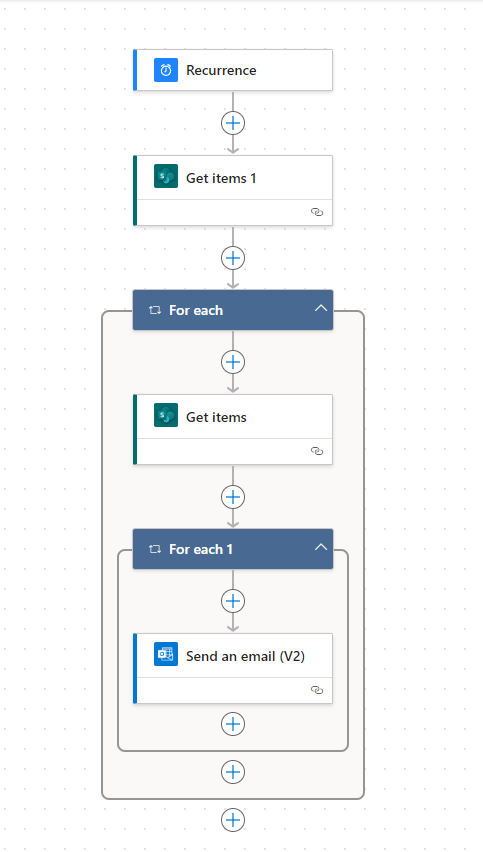
Voici un aperçu du Power Automate :



Pour le dépassement qui est de fournir un email avec la liste des élèves par classe a son prof attitré pour le 1 septembre

Voici quelques essais infructueux :





# Aperçu de l'organisation en tant qu'équipe :

Notre équipe s’est partagé les différentes tâches pour accéder à la demande du client suivant les forces et les faiblesses de chacun. Notre trio s’est composé de Nathanaël, louis et Charles.

Nathanael s’est occupé de l’automatisation grâce à l’outil Power Automate et Microsoft Forms ainsi que faire des retours sur le rapport de projet, et de finaliser la configuration du git-hub.

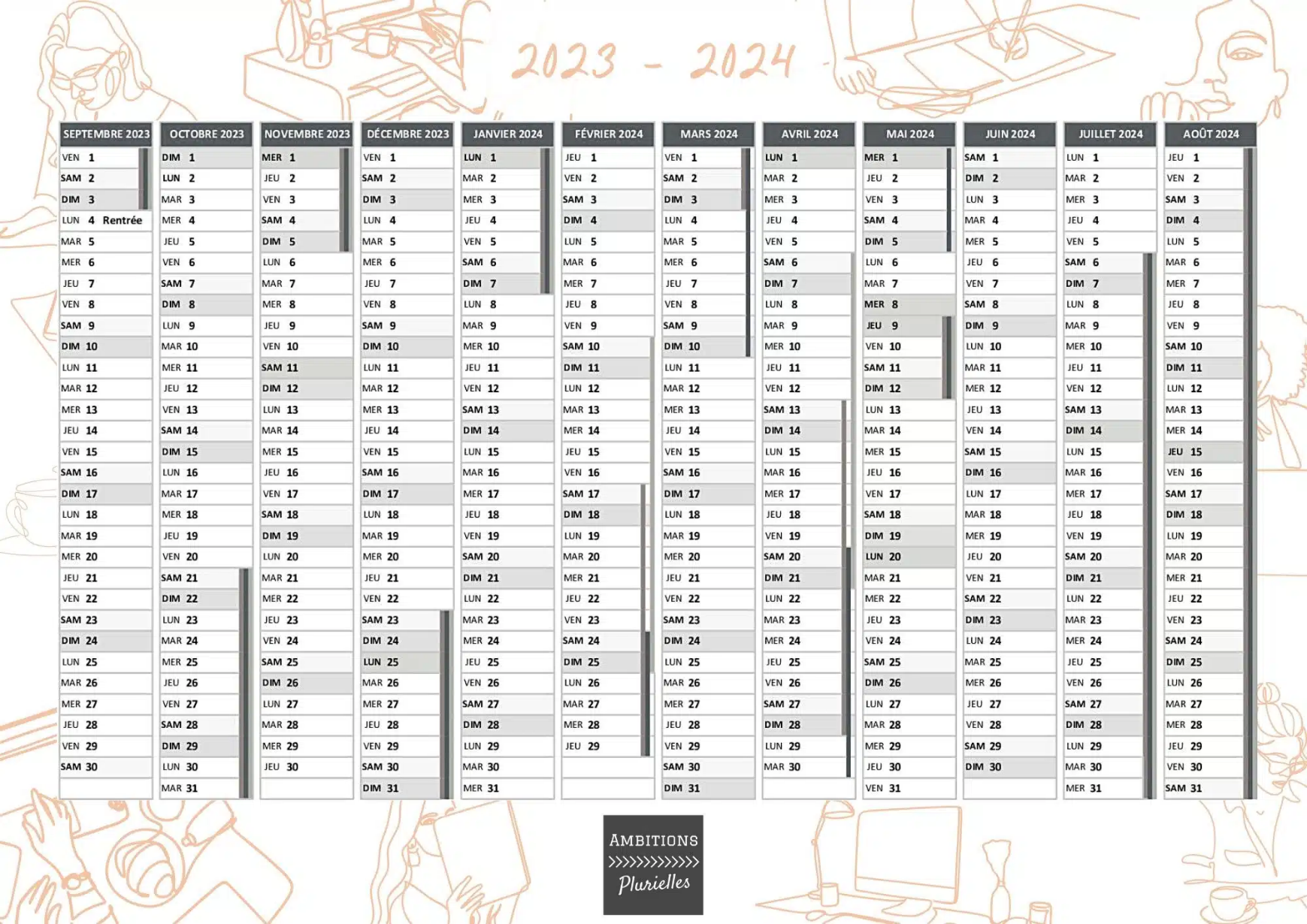
Louis a tenté de configurer le git-hub.

Charles à réaliser le logiciel de gestion de temps via Monday ainsi que le rapport de projet.

Pour ce faire, nous avons travaillé en équipe, que ce soit pour la gestion du temps, selon la disponibilité de chacun, le partage des données via git-hub, mais aussi pour le Power Automate afin de réaliser des tests pour vérifier l’efficacité des automatisations.

Nous avons créé un groupe de messagerie instantanée sur discord afin d'avoir un produit fini qui correspond à nos attentes. On a organisé une réunion en présentiel afin de collaborer sur le projet et obtenir des retours aussi rapidement et aisément que possible.

Établisement d’un calendrier partager pour les rendez vous :

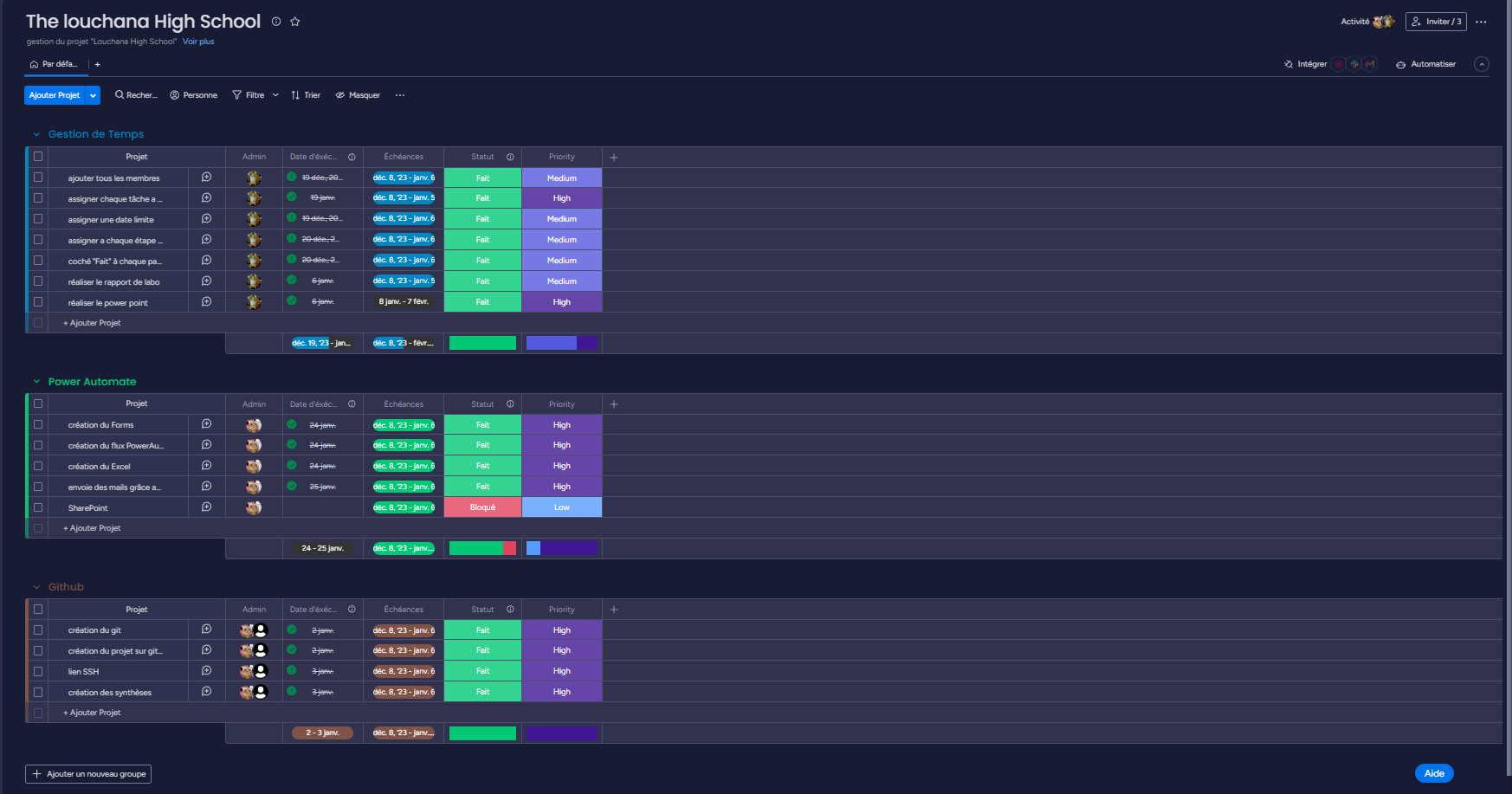
Présentiel

Virtuel

# Gestion du temps et répartition des tâches :

Au tout début du projet, nous avons procéder à quelques réunions virtuel ou présentiel. Une fois toutes les variabilités mises en évidence, nous avons tenté d’estimer le temps que chaque tache nous prendrait tout en restant réalistes. Toutefois, nous avons veillé à laisser un temps supplémentaire pour chaque tache afin de pouvoir relire, corriger certaines erreurs potentielles et donner un temps de sécurité. Le tout est de rester dans les temps.

La répartition des tâches s’est faite de manière plus abstraite. Les personnes ayant le plus de facilités dans un domaine s’en sont chargé. Par exemple, Nathanaël a préféré d’emblée la réalisation du power automate. Louis a préféré git-hub. Charles a préféré s’occuper de l’aspect communication et gestion de projets ainsi que la partie écrite.



# Portfolio de commandes via GIT :

Comment nous avons configurer le GitHub

Voici les commandes nous ayant permis de réaliser le GitHub

A la base nous avons créé un git et fais la configuration de base

Commande globale à faire à l'installation de git

git config --global user.name "Duchene Nathanael"

Dire qui fait les modification mail sur git

Git config --global user.mail "etu55331@student.henallux.be"

Configuration des couleurs sur git en auto

git config --global color.diff auto

git config --global color.satus auto

git config --global color.branch auto

Configuration des d éditeurs de texte de base pour git

git config --global core. editor vim

git config --global merge. tool vimdiff

Ensuite j’ai créé le git

J’ai choisi la localisation de la racine à la base du git sur mon pc je voulais le mettre dans les documents

Cd document

Créer un dossier

Mkdir <nom du dossier>

cd pour rentrer dans fichier créer

cd <nom du dossier>

inistialiser le git

git init

Mettre le fichier dans le git

git add <nom de fichier..extension>

Comment réaliser un commit en local

Il faut modifier un fichier dans le git

Refaire un git add

git commit -m "description"

Corriger un commit

git commit --amend

**Comment annuler un commit**

git commit --amend --no-edit

Annuler un commit

git log

Commit 67251e58d4b794ee28602bf96d457adad55f3d16

Pour voir le status si fichier pas commit

git status

**Supprimer un commit**

Afficher l'historique de commit sur la branche

git log

git reset est une commande pour annuler un commit 3 façons différentes (--soft; --mixed; --hard)

git reset --hard HEAD^ 67251e58d4b794ee28602bf96d457adad55f3d16

Création de branche en via git

Créer une branch

git branch <nom de la branche>

Lister les branch

git branch

Changer de branch

git checkout <nom de la branche>

Fusionner des branch

git merge <nom de la branche>

Supprimer une branch

git branch -d <nom de la branche>

Enregistrer les modifications non indexées pour une utilisation ultérieure.

git stash

Par la suite nous avons créé un lien ssh

**Première configuration de git avec git hub** :

À taper sur git

ssh-keygen -t ed25519 -C "etu55331@student.henallux.be"

cat C:\Users\nathd\.ssh\id\_ed25519.pub | clip

Sur le site git hub

La clé obtenue est à mettre sur le site GitHub dans paramètre et ensuite dans ssh et GPG Keys

Create a new ssh keys et ensuite mettre les donner copier du ssh-keygen

Comment récupérer les donner du GitHub en local :

Commande pour avoir le git sur son pc

git clone https://github.com/etu55331/synth-se.git

Aller dans le dépôt

cd <nom du github en local>

Récupérer la dernière version

git pull orign main

Vérifier si vous avez toutes les branches qui sont présente sur le GitHub

git branch

Comment faire une modification sur le GitHub :

**Effecture un pull request**

git branch <nom de la nouvelle branch>

git checkout <nom de la nouvelle branch>

Mettre le fichier dans le git

git add <nom de fichier.extension>

Faire un commit

git commit -m "description des modif ou fichier"

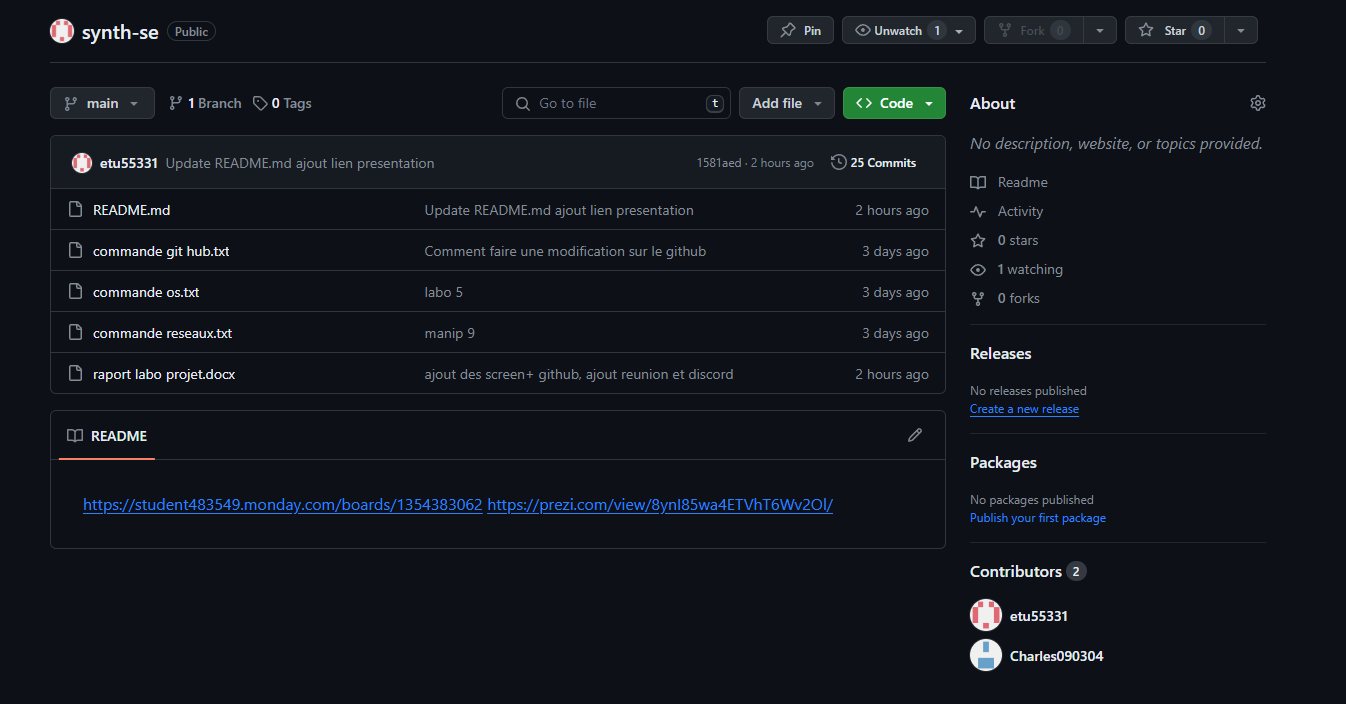
Mettre les fichiers sur git hub

git push

Ensuite ça va vous ouvrir le site github et créer un compare and pull request et ensuite il faudra valider

Le git-Hub contient l’entièreté de nos commandes, c’est avec plaisir que nous vous invitons à consulter le fruit de notre travail pour ce projet.

Et voici le résultat de celui-ci :



Lien du GitHub :

<https://github.com/etu55331/synth-se.git>

# Conclusion :

En conclusion, la demande de notre client qui consiste à faciliter le processus d’inscription a été résolue sans encombre. Chaque méthode que nous avons utilisée est présente dans le rapport afin de garantir une communication transparente.

Cependant, certaines améliorations pourraient être apportées, comme un enregistrement des données dans des listes SharePoint, ainsi qu’un envoi des informations aux professeurs concernés.