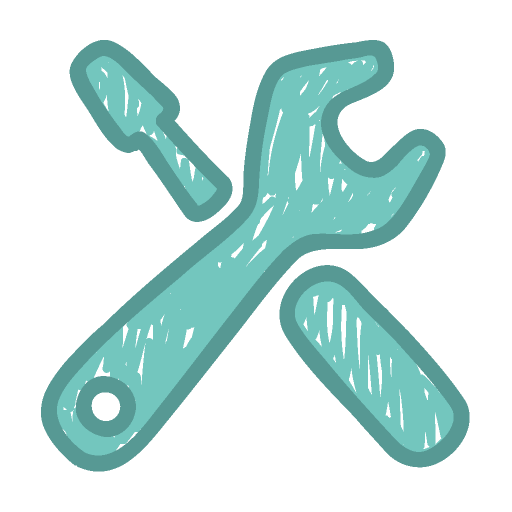
ALEIXO-CARVALHO Enzo  
BONNEAU Martin  
CACOUAULT Yoann  
DEPREISSAT Antonin  
GERARD Kyllian  
VINCENT Maxime

**« Définition des outils »**  
Gestion de projet informatique



Année 2017/2018

IUT Informatique – Université de La Rochelle

# Sommaire

[Sommaire 2](#_Toc517167532)

[Introduction 3](#_Toc517167533)

[Présentation du groupe 4](#_Toc517167534)

[**Redmine** 5](#_Toc517167535)

[Présentation 6](#_Toc517167536)

[Notre utilisation 7](#_Toc517167537)

[**Gitlab** 8](#_Toc517167538)

[Présentation 9](#_Toc517167539)

[Notre utilisation 10](#_Toc517167540)

[Rétrospective – Début de semaine 11](#_Toc517167541)

[Rétrospective – Fin de semaine 12](#_Toc517167542)

[Conclusion 13](#_Toc517167543)

# Introduction

Pour le projet tutoré du semestre 2, nous avons été sollicités par un professeur du département Technique de Commercialisation pour développer une application d’éphéméride permettant aux étudiants de sa promotion de réviser d’une manière plus ludique et donnant envie.

Durant la réalisation de notre projet, l’utilisation de différents outils était indispensable. Ce document recense donc tous les outils que nous avons utilisé.

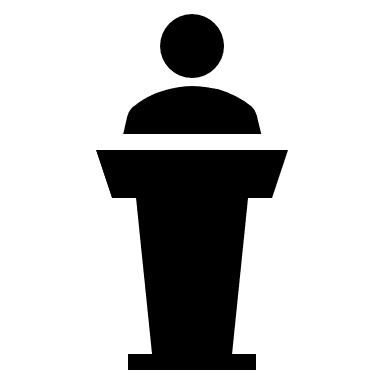
De l’outil de gestion, à l’outil d’organisation, ils vous seront accompagnés d’une présentation, d’une description et enfin d’une explication sur la façon dont nous l’avons utilisé.



# Présentation du groupe

Pour arriver au bout de ce projet, nous étions 6 et nous avons eu la chance de pouvoir rester un groupe entier, contrairement à d’autres groupes dont les membres ont quitté l’IUT.

Voici les membres de l’équipe :

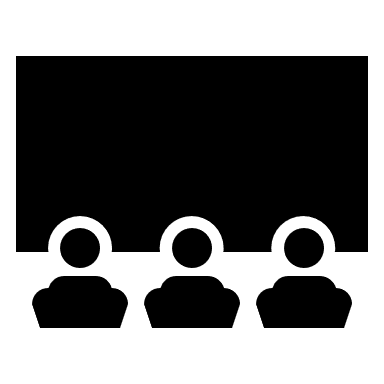


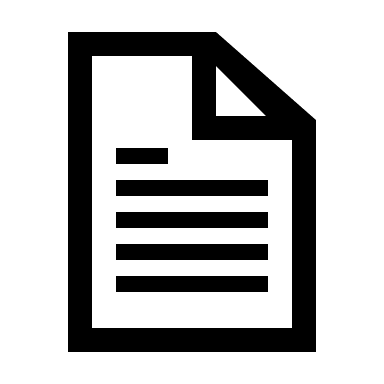
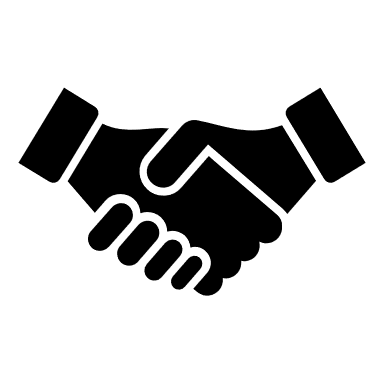
*ALEIXO-CARVALHO Enzo*

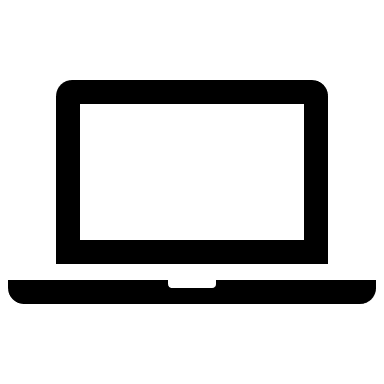
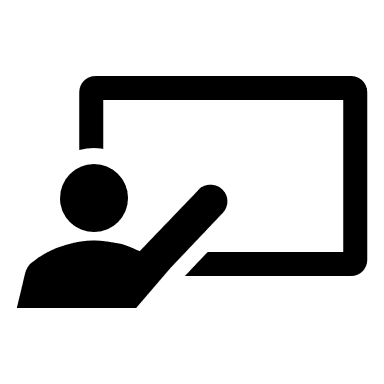
*GÉRARD Kyllian*

*VINCENT Maxime*

*DEPREISSAT Antonin*







*CACOUAULT Yoann*

*BONNEAU Martin*

Nous nous sommes réparti les rôles en fonction de nos compétences :

* Enzo : Leader du projet
* Kyllian : Leader fonctionnel
* Yoann : Greffier + Intégrateur
* Antonin : Leader des interfaces
* Martin : Leader technique
* Maxime : Leader communication

***Leader de projet*** : Le leader c’est celui qui motive l’équipe ! Il vérifie l’avancement du travail général, il assigne les tâches aux membres de l’équipe. Le leader peut éventuellement recadrer un membre de l’équipe si celui-ci fournit un travail insuffisant.

***Leader fonctionnel*** : Le leader fonctionnel gère la documentation du projet (livrables, annexes, etc.). Il s’occupe également de fusionner les différentes parties de codes si besoin. Enfin, il guide l’équipe à travers des plannings d’organisation prévisionnelle.

***Greffier + Intégrateur*** : Il va s’occuper de l’installation logiciel (partie intégrateur) et en tant que greffier il va prendre note des réunions, discussions, il s’occupe cahier des charges et du journal de bord.

***Leader des interfaces*** : Le leader interface comme son nom l’indique va gérer les interfaces de l’application et les contrôleurs associés. Il peut également repartir du travail d’interface à d’autres membres de l’équipe.

***Leader technique*** : Le leader technique va aider et orienter ses partenaires sur le code si ceux-ci sont en difficulté. Il va principalement s’occuper de la Base de Données et de l’architecture du logiciel.

***Leader Communication*** : C’est le parleur de l’équipe, il communique avec le client et les futurs utilisateurs pour pouvoir donner à son équipe les meilleures informations pour le développement de l’application.

# **Redmine**



## Présentation

Redmine est une application WEB qui s’occupe en très grande partie de la gestion de projet en collaboration.

Cet outil permet notamment de gérer son projet et ce de manière totalement automatisée (par exemple, dans une bonne gestion, il est nécessaire d’intégrer un Gantt afin de pouvoir gérer l’aspect temporel du projet, et bien sur Redmine, le Gantt est généré automatiquement en fonction des activités réalisées entrées par le collaborateur).

**Les plus :**

* Simple d’utilisation
* Interactif
* Très complet (gestion des rôles, des collaborateurs, …)

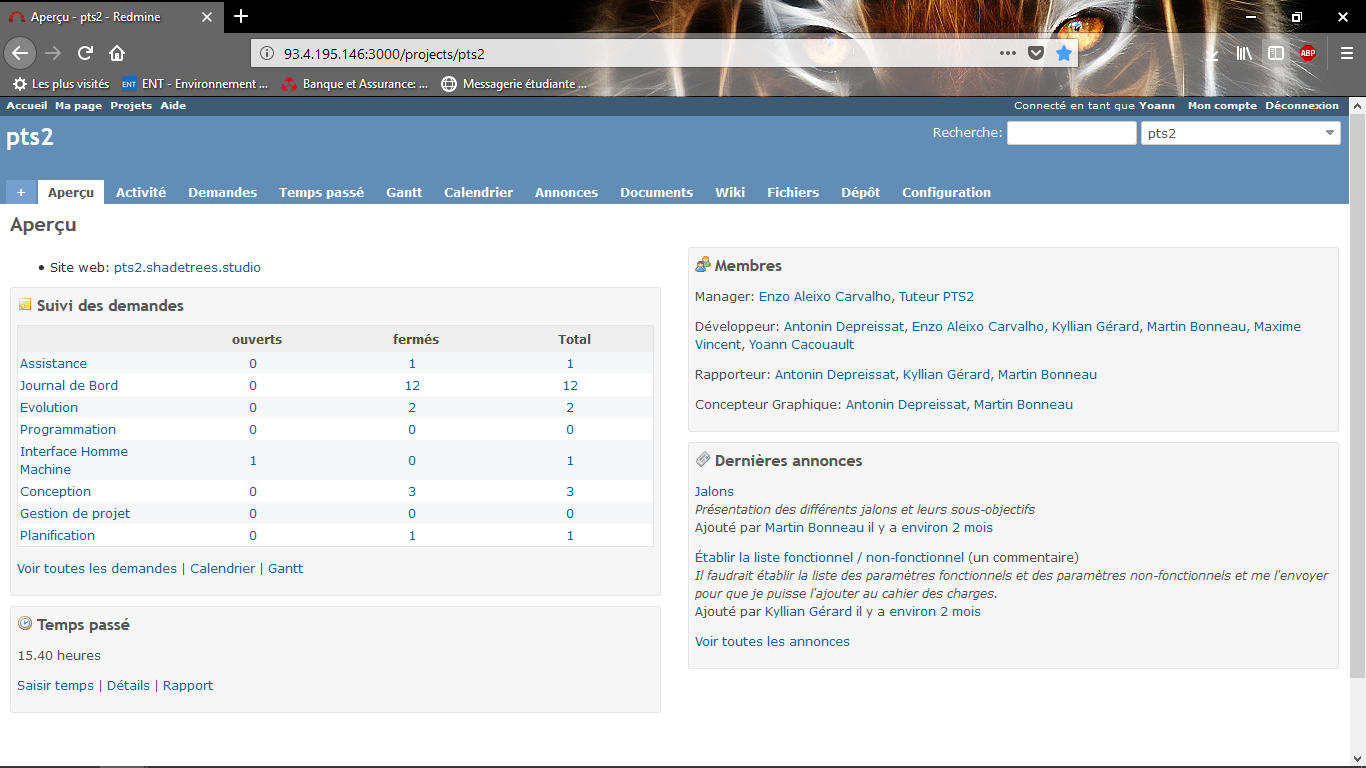
**Les moins :**

* Complexe à installer
* Non disponible sur la connexion sans fil « eduroam »

Martin était celui qui disposait le plus de compétences dans le domaine d’installation d’application WEB. Nous avons donc décidé d’installer Redmine sur sa Raspberry Pi. Cependant, faire ceci était très risqué car de nombreux événements pouvaient survenir et ainsi rendre l’avancée du projet plus complexe (problème avec la Raspberry, soucis liés à l’électricité, …). Nous devions donc prendre toutes les conditions nécessaires au bon maintien de notre projet (sauvegardes régulières, …).

Heureusement, nous avons eu de la chance car aucun de ces événements n’est venu interrompre notre avancée.

Lien de notre Redmine :

<http://redmine.shadetrees.studio/> 

## Notre utilisation

Nous avons utilisé Redmine dans le cadre de notre projet pour de nombreux points. Nous souhaitions utiliser la totalité des capacités de cette applications. Voici donc une liste de ce que nous pouvions faire avec Redmine :

* Générer le Gantt pour avoir un planning prévisionnel concernant les séances futures et le temps que nous devrons passer sur chaque partie.
* Permettre au tuteur de voir l’avancée du projet.
* Mettre en ligne les tableaux de bord à chaque fin de séance.
* Mettre en ligne les différents documents liés à la conception de projet (diagramme de classe, de cas d’utilisation, …)
* Mettre en ligne les différentes versions du cahier des charges.

# **Gitlab**



## Présentation

GitLab est un outil de gestion et de collaboration informatique lié cette fois-ci au code en lui-même.  
GitLab est un dérivé de GitHub qui possède un mode de fonctionnement très complet, mais extrêmement complexe à prendre en main lorsque nous n’en avons jamais entendu parler. Tout réside dans le mot que nous avons cité plus haut qu’est la collaboration. Sur GitLab, nous avons ce que nous appelons des commit (description des modifications apportées au code), des pulls (récupération du code présent sur le serveur, celui-ci ayant subi des modifications des autres collaborateurs) et enfin des push (appliquer les modifications sur le serveur afin que celles-ci soient visibles pour les autres collaborateurs).

**Les plus :**

* Traçabilité extrêmement complète
* Une fois le principe compris, c’est simple d’utilisation
* Efficace
* Possibilité d’étudier le code n’importe ou et de faire des modifications de dernières minutes (simples)
* Intégration simple avec son IDE (Netbeans dans le cadre de ce projet)

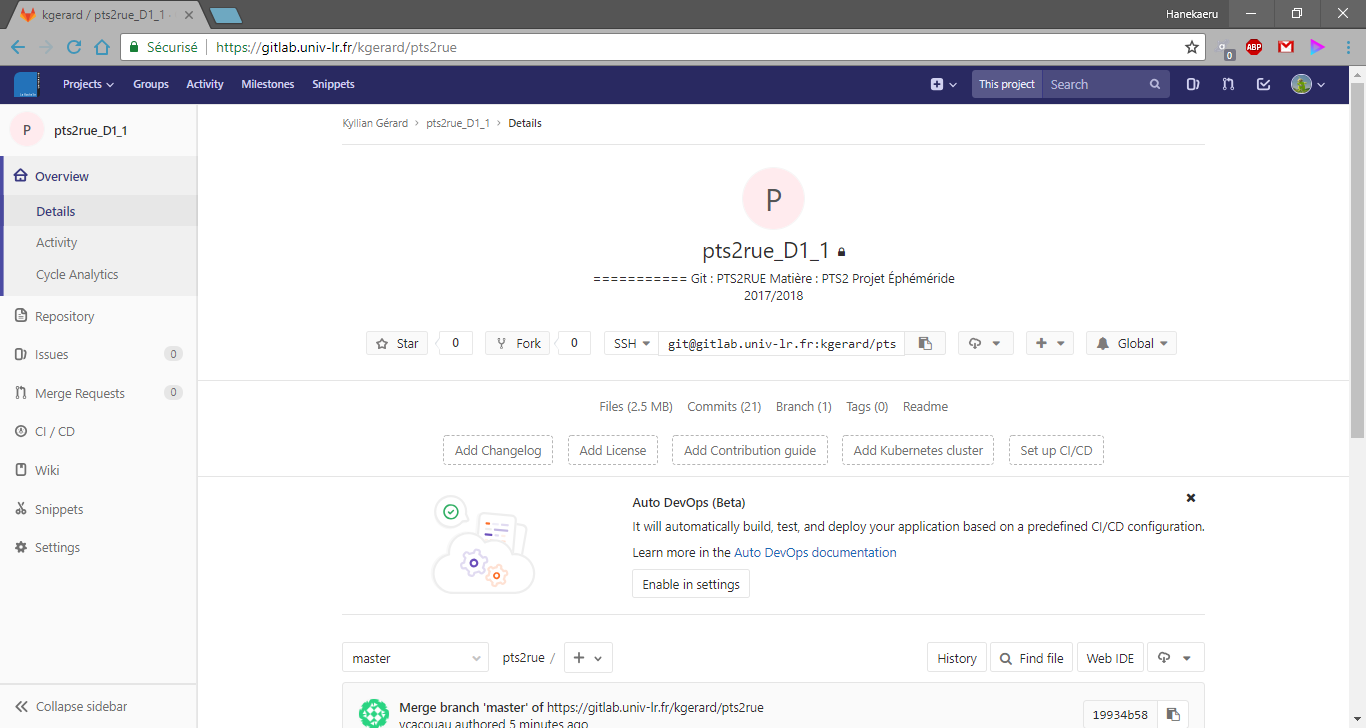
**Les moins :**

* Long à prendre en main (donc perte de temps importante pour expliquer)
* Avec GitBash, possibles erreurs et confusion dans les différentes commandes

Cette fois-ci, nous n’avons pas installé GitLab sur l’une de nos machines personnelles mais bien sur un serveur distant, disponible grâce à notre compte Moodle.

Lien de notre GitLab :

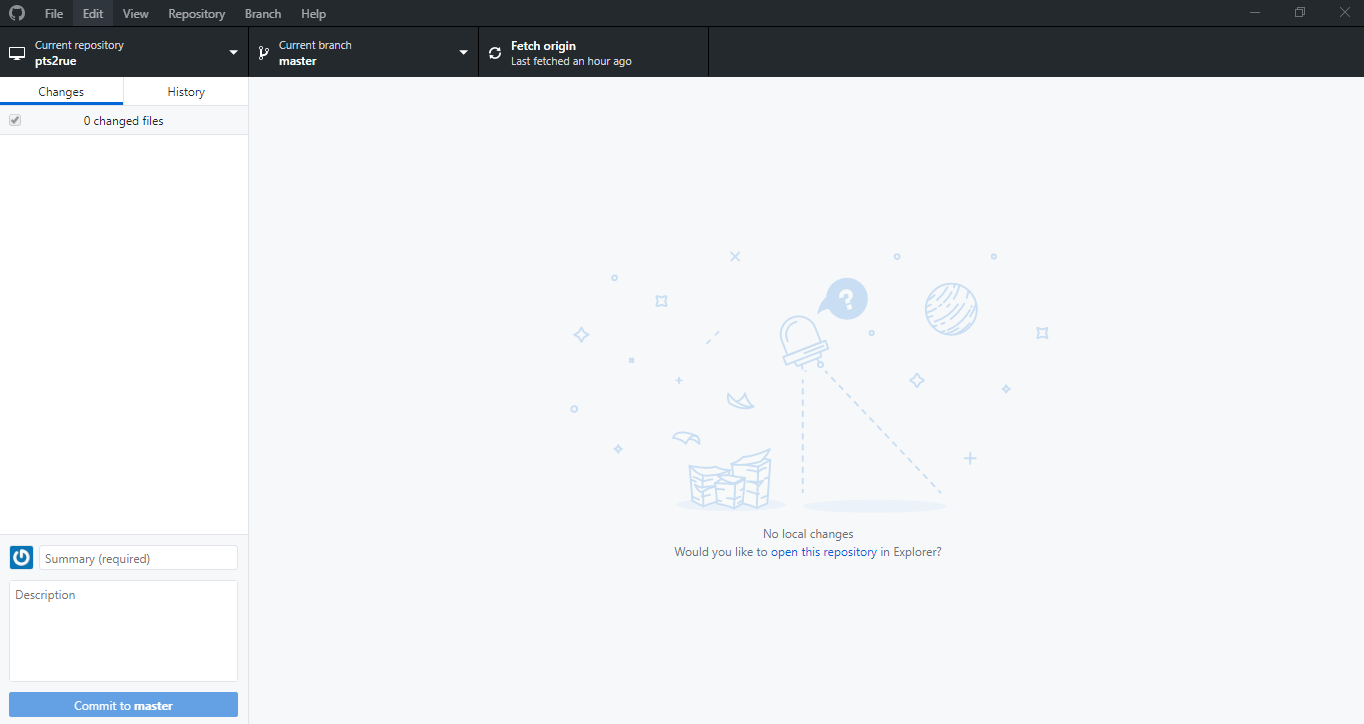
<https://gitlab.univ-lr.fr/kgerard/pts2rue.git>



## Notre utilisation

GitLab est un outil de gestion de version, pour notre projet tutoré il nous a été imposé par les professeurs de la matière.

Il est accessible depuis un terminal mais après avoir eu de nombreuses difficultés à prendre en main les commandes GitBash, nous avons décidé de passer sous Git Desktop (qui est une interface graphique pour Git). Cette interface étant plus simple d’utilisation que le terminal Git, nous avons formé la totalité de l’équipe.



Au début de l’utilisation de cet outil, nous nous sommes fixés des règles concernant l’utilisation de commit (c’est-à-dire éviter les commit inutiles, ne pas hésiter à demander si jamais nous avions besoin d’aide pour la gestion, …).

Nous nous sommes rendu compte que Git est un outil compliqué et qu’il faut se concerter pour l’utiliser sans avoir de problèmes lors des « pull ».

# Rétrospective – Début de semaine

|  |  |
| --- | --- |
| **Ce qui était bien dans le projet** | **Ce qui n’était pas bien dans le projet** |
| Bonne entente au sein du groupe. Bonne cohésion. | Nous avons eu moins de motivation dans certaines périodes. |
| Nous avons pris un maximum d’avance dans la mise en place. | Le Gantt est arrivé assez tardivement, donc nous avions eu un petit souci dans la prévision temporelle du projet. |
| Nous avons des états d’esprits assez semblables. | Nous avions tendance à ne pas donner des conseils lors des soucis et à ne guère donner une surveillance adaptée en fonction des membres. |
| Nous avons travaillé dans une salle à part, nous n’étions donc pas dérangés par tout groupe extérieur. | Nous sommes revenus de très (trop) nombreuses fois sur nos décisions. |
| Nous nous sommes groupés à l’extérieur des cours de PTS2. | Les tensions engendrées par le manque de travail des membres. |
| Nous avons eu l’idée de réaliser des carnets de bord. | Manque de concentration. Nous étions trop facilement distraits. |
| Nous avons fait avec les « moyens du bord », nous avons pu développer un sens de la débrouillardise. |
| **Plan d’action** | |
| Récupérer la motivation : Après l’ajout de nouvelles idées, de nouveaux enjeux et objectifs et la mise en place des idées des membres, la motivation a pu augmenter à nouveau. | |
| Gantt : Nous devions retravailler calmement sur ce qu’il restait à faire et nous pouvions à nouveau | |
| Tensions : Nous avons donc instauré des règles : faire le travail demandé dans les délais impartis. Nous avions la possibilité de demander de l’aide si jamais nous étions perdus mais dans l’ensemble le travail devait être fait pour ne pas prendre de retard et ainsi être retardés dans les autres facteurs. | |
| Anticipation : Prévoir un découpage fonctionnel plus détaillé. | |

# Rétrospective – Fin de semaine

# Conclusion