

Etude de cas thématique

Ce document constitue le rendu du milestone 2 du projet choix d'options

Cliente

Anne LAPUJADE

Tutrice pédagogique

Catherine BARRY

Membres

Tech Lead / Product Owner

Abdelhalim OULAD MANSOUR

Business Analyst

Mamadou Binnet BARRY

Scrum master

Abdoul Aziz TRAORE

Développeur Full stack

Mouhcine BENNANI

Sommaire

1) Introduction	3
2) Description du projet	4
3) Comportement fonctionnel de l'application	6
3.1) Fonctionnalités selon utilisateurs	6
3.2) Illustration	8
4) Conduite de projet	10
5) Présentation activités passées (Milestone 1)	12
6) Présentation des activités (Milestone 2)	14
7) Contact avec la cliente	17
8) Organisation interne au sein de l'équipe	19
9) Prise de recul	22
10) Conclusion	26
11) Annexes	27
11.1) Modèle conceptuel de données	27
11.2) Diagramme de GANTT	28
12) Autres annexes	33

1) Introduction

Cette introduction représente le plan de ce rendu 2 afin que le lecteur soit orienté.

Dans un premier temps on aborde la description du sujet du projet, le comportement fonctionnel attendu de l'application envers les différents utilisateurs et on finit par la présentation de l'arborescence du projet afin de bien illustrer les informations.

Dans la deuxième partie on aborde les activités passées pendant le Milestone 1, en les classifiant plus ou moins en détails, ceci dans le cadre d'avoir un récapitulatif pour arriver aux activités du milestone 2 afin de bien préciser ce qui a été fait durant cette partie.

Dans la troisième partie on décrit le contact et les échanges avec notre cliente ce qui va refléter une idée de la méthode de projet utilisée.

La partie 4 va aborder surtout la conduite de projet qui était utilisée dans la réalisation de ce projet.

La partie 5 va contenir toutes les informations de l'organisation interne au sein de l'équipe.

A la fin de ce document, un emplacement sera dédié à la prise de recul sur la phase analyse et une conclusion.

2) Description du projet

Le projet choix d'options, comme son nom l'indique, consiste à réaliser une application permettant aux étudiants du master MAGE de l'Université de Picardie Jules Verne d'effectuer le choix des unités d'enseignements qu'ils souhaitent suivre au cours de l'année.

L'application comprendra principalement quatre grandes parties :

La première partie, le responsable d'année doit importer un fichier Excel comportant le nom de tous les étudiants et qui servira à la création des comptes étudiants uniquement pour le niveau d'étude Master 1, pour le Master 2 les inscriptions seront gérées par validation du Master 1.

Les responsables d'années ont la responsabilité de renseigner toutes les unités d'enseignements en respectant la structure proposée (nom du bloc, nom de l'UE, nombre de personnes par UE, description de l'UE...).

Pour la seconde partie, les responsables d'années initialisent l'application et ouvrent l'accès aux étudiants en mode lecture et écriture afin qu'ils effectuent leur choix d'options tout en prenant compte des différentes contraintes de gestion sur les unités d'enseignement et la promotion, ce qui permettra de gérer le choix des étudiants par promotion afin que les étudiants ne puissent choisir que les UEs leurs concernant. Chaque initialisation de l'application sera dédiée à une période de choix, à noter que pour le Master 1 nous avons deux périodes de choix d'options ce qui signifie qu'on effectuera l'initialisation deux fois tandis que pour le Master 2 elle s'effectue qu'en début d'année universitaire.

Les étudiants du Master 1 seront scindés en deux groupes lors du second semestre, d'une part les stagiaires et d'autre part les alternants, cette contrainte de gestion reste très importante.

Les responsables d'années doivent préciser la promotion et la spécialité pour laquelle ils effectueront l'initialisation, ce qui permettra uniquement qu'à cette

promotion de pouvoir choisir ces unités d'enseignement et les étudiants seront contraints de réaliser leur choix d'option parmi les UEs renseignées.

La troisième partie consiste à gérer les étudiants afin qu'ils puissent effectuer leur choix d'options, à ce niveau plusieurs autres contraintes sont à prendre en compte telles que les matières optionnelles ou pas, à noter qu'une matière peut être obligatoire pour une spécialité et pas pour l'autre.

Dans cette partie, l'on doit prendre en compte les contraintes de gestion qui ont été mises en place lors de l'initialisation de l'application, ce qui consiste à respecter la date donnée par les responsables d'années.

Tant que la période donnée par l'admin est en cours, les étudiants pourront effectuer tous les différents changements qu'ils veulent parce qu'ils auront l'accès en écriture, mais une fois la période passée, ils ne pourront que consulter et s'ils veulent effectuer des changements, dans ce cas, ils iront voir les responsables d'années qui eux, peuvent modifier les choix.

Pour la dernière partie, elle consiste à créer des groupes dans chaque matière.

La création des groupes par matière prendra en compte plusieurs paramètres: le nombre d'étudiants ayant choisi la matière, le nombre de groupes pour cette matière et aussi le nombre d'étudiants par groupe, ces informations nous permettront d'effectuer une répartition.

3) Comportement fonctionnel de l'application

3.1) Fonctionnalités selon utilisateurs

Rôle responsable d'année :

Au début de l'année universitaire, les responsables d'année se connectent sur l'application et peuvent mettre à jour la maquette de la formation Miage en ajoutant/modifiant/supprimant un parcours et/ou ajoutant/modifiant/supprimant un bloc dans un parcours sélectionné et/ou ajoutant/modifiant/supprimant une UE et préciser si elle est obligatoire/optionnelle, ainsi que le nombre maximal des étudiants qui peuvent choisir cette UE.

Les responsables d'année gèrent l'arrivée des nouveaux étudiants en Master 1 en créant leurs comptes dans l'application via l'importation d'un fichier excel généré par l'application e-candidat.

Les responsables d'année gèrent dans l'application le passage des étudiants du M1 au M2 en les plaçant dans les différents parcours, en supprimant les anciens étudiants ayant validé leurs M2 et en précisant les étudiants redoublants et les matières où ils sont admis/ajournés, tout ceci se fait via l'actualisation des données.

Ainsi qu'ils choisissent une période de choix d'options pour un parcours donné en sélectionnant la date de début et la date de fin, pendant cette période, les étudiants effectueront leurs choix vis-à-vis les matières optionnelles qu'ils veulent suivre au cours de l'année, il y a deux périodes de choix d'options en M1, une période au début de chaque semestre.

Rôle étudiant :

Une fois la période des choix d'options arrivée, les étudiants peuvent accéder à l'application et entrer leurs classement des UEs optionnelles via un formulaire.

Ce formulaire contiendra la liste des UEs avec des champs à côté pour faire entrer l'ordre de préférence de chaque UE par l'étudiant.

Les étudiants peuvent toujours modifier leurs choix tant que cette période est ouverte, une fois que la période est finie, les étudiants ne peuvent plus modifier leurs choix mais ils peuvent toujours les consulter.

Rôles responsable d'année :

Une fois que la période des choix d'options est passée, les responsables d'année récupèrent les choix effectués par les étudiants, ils peuvent valider leurs choix si le nombre d'étudiants choisissant une même UE ne dépasse pas le nombre limite autorisée dans cette UE, sinon le responsable d'année peut retirer manuellement les étudiants d'une UE et les mettre dans l'UE qu'ils ont choisi en 2ème position, si cette UE aussi est pleine, ils seront classés dans leurs 3èmes choix etc..., ceci se fait via un tirage au sort qui se fera par le responsable de l'année.

Les responsables d'année peuvent générer des groupes pour une UE sélectionnée, ainsi ils peuvent modifier ces groupes en basculant les étudiants d'un groupe à l'autre ou supprimer un/des groupes.

Objectif final

Une fois toutes les étapes précédentes réalisées, l'application permettra de générer un fichier excel contenant les choix d'options effectués par les étudiants et les groupes des étudiants dans les UEs.

3.2) Illustration

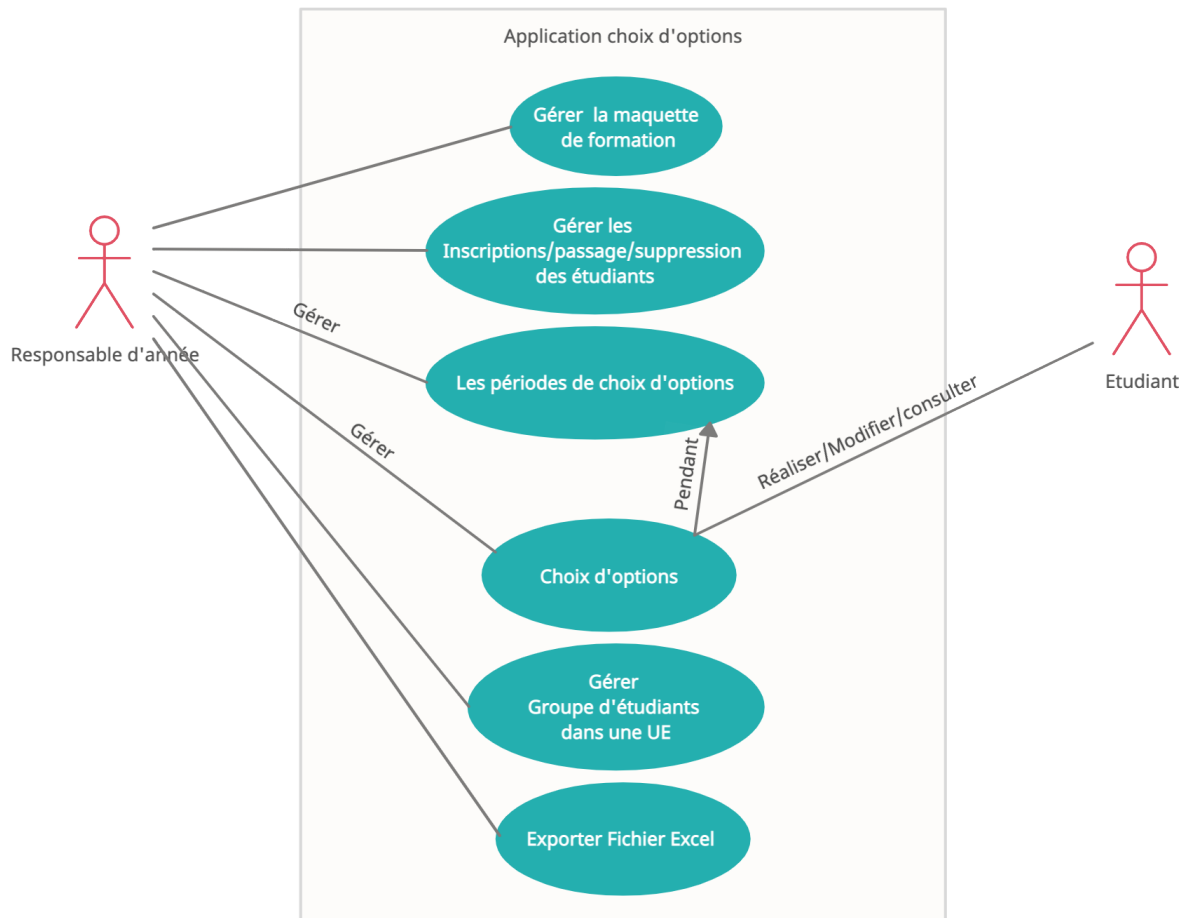


Figure 1 (Diagramme de cas d'utilisation)

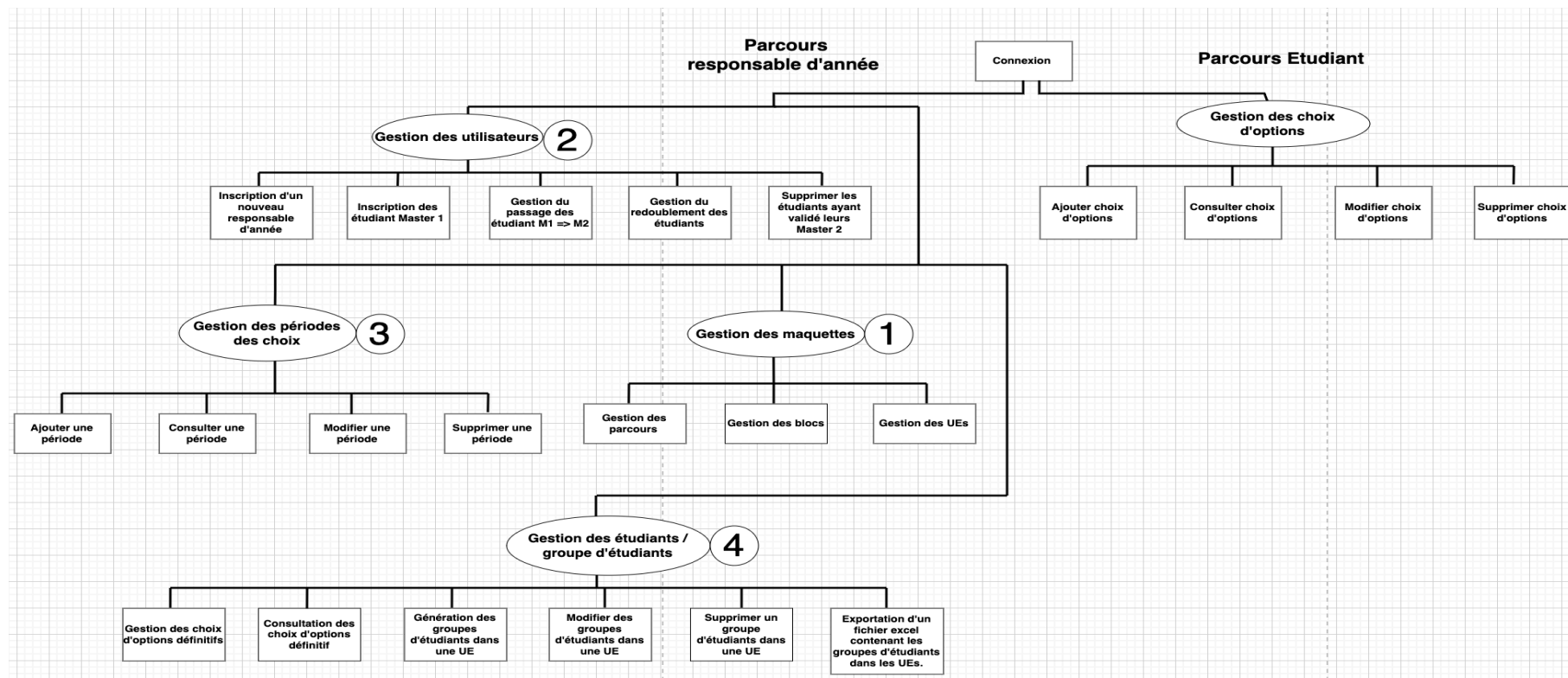


Figure 2 (Arborescence de l'application)

Explication de l'arborescence ci-dessous :

L'arborescence ci-dessus représente les grandes fonctionnalités effectuées par les deux types d'utilisateurs, chaque groupe de fonctionnalité est regroupé sous une forme ovale, il faut suivre les lignes afin de comprendre les liaisons, les fonctionnalités à gauche sont pour l'utilisateur **"responsable d'année"**, et à droite pour l'utilisateur **"étudiant"**.

- Forme rectangle = fonctionnalité.
- Chaque numéro à côté d'un groupe de fonctionnalités représente l'ordre d'exécution de ce dernier (les groupes qui n'ont pas un numéro d'ordre => ils ne sont pas concernés).

4) Conduite de projet

Avant tout, il faut noter qu'on a effectué une phase d'élaboration qui consiste à réaliser une étude de l'existant, cette étude nous a non seulement permis de voir les limites et comprendre l'utilité et l'importance du produit qu'on mettra en place.

Pour la méthode de Gestion de projet, nous étions plus partis pour un début avec la méthode Agile avec l'utilisation d'un backlog comportant toutes les user stories qui seront développées, des différents sprints avec une durée de deux semaines, et toutes les autres cérémonies agiles pour pouvoir réaliser tout le projet, mais on s'est vite rendu compte qu'avec cette méthode à la longue, on risque d'avoir un problème notamment que notre cliente n'était pas très disponible, et nous même n'avions pas le même emploi, vu que parmi nous il y en a qui travaillent, c'est donc compliqué de regrouper tout le monde à chaque fois, enfin qu'il était très difficile d'avoir un rendez-vous de validation ou de sprint planning pour présenter et faire valider les fonctionnalités développées ou à développer avec notre cliente.

Pour éviter ces problèmes, nous nous sommes tournés vers la méthode de gestion de projet "cycle en V" avec une phase descendante, suivie d'une phase ascendante.

Elle repose sur des étapes séquentielles et linéaires, allant de l'analyse des besoins au test d'acceptation. La partie descendante du « V » correspond aux quatre actions de conception et de développement du système, tandis que la partie ascendante reprend les quatre phases d'assurance qualité qui lui sont associées.

Cette méthode est principalement utilisée lorsque le client est indisponible et ne peut par conséquent faire aucun retour permanent sur la situation. Elle est encore utilisée lorsque le client a une vision très précise de ce qu'il souhaite (les besoins, le planning détaillé et l'anticipation des risques). Ce qui est exactement notre cas, c'est pourquoi nous avons décidé de choisir **la méthode cycle en V** pour une première partie.

Mais à la longue, au niveau du développement et des différentes tâches entre nous, on partira nous sur un mixage des deux méthodes (Agile et Cycle en V) pour pouvoir avancer plus facilement et structurer notre travail, notamment avec l'utilisation de Jira pour la gestion de tout le backlog dont les users stories seront placées dans des sprints avec une durée de deux à trois semaines, et à la fin de chaque sprint, on essaiera de faire valider avec la cliente sur Discord si elle est disponible (même si avec elle nous sommes en cycle en V, nous lui avons envoyé des dates prévisionnelles de rencontre).

Sinon, on continuera d'avancer en attendant d'avoir une réunion avec elle, et effectuer quelques modifications si possibles sur certains points afin d'éviter l'effet tunnel du cycle en V surtout que nous n'avons pas d'autres milestones ou de présentation à faire avant la présentation finale, donc nous préférons avoir des réunions de présentations avec la cliente pour éviter toute surprise en février.

Donc pour récapituler, pour la réalisation du projet, nous aurons la méthode Agile et la méthode Cycle en V :

Agile : entre nous les membres du groupe.

Cycle en V : avec la cliente.

5) Présentation activités passées (Milestone 1)

Dans cette partie, nous avons tout ce qui a été fait lors du milestone 1, ce qui aurait dû être fait et ce qui restait à faire.

Ce qui a été fait : (scinder en milestone):

- Le choix de projet parmi les sujets proposés.
 - Réunion entre les membres de l'équipe afin de choisir un sujet du projet thématique qu'on veut faire, le choix s'est porté sur le projet "Choix d'options".
- Définition des rôles des membres du groupe.
 - Réunion entre les membres de l'équipe afin de décider qui fait quoi.
- Premier entretien avec la cliente.
 - Première écoute de la cliente afin d'essayer de comprendre son besoin exact.
- Rédaction du compte rendu de l'entretien.
 - Une synthétisation des idées de la cliente afin de structurer son besoin.
- Analyse du besoin.
 - Réunion entre les membres afin de bien comprendre le besoin et avoir un partage d'idées entre les membres de groupe.
- Analyse de l'existant.
 - Analyse et compréhension de l'existant à partir duquel on va démarrer.
- Identification des ressources.
 - Désigner ce qu'il faut pour la réalisation de ce projet.

- Création d'un environnement de travail :
 - Création du projet sur la plateforme JIRA pour une bonne organisation.
 - Création d'un serveur discord
 - Création d'un google drive.
- Constitution de la charte graphique, des livrables (la police de texte, la couleur, les logos etc...).
- Rédaction d'un premier jet du cahier des charges.
- Introduction du projet.
 - Commencement de la partie technique du projet (mettre en place des premières fonctionnalités).
 - Développement de la fonctionnalité d'authentification.
 - Commencement du développement de la fonctionnalité de création des comptes.
- Planification des fonctionnalités à venir.
- Réalisation de quelques pages de la maquette.

Ce qu'il reste à faire :

- Validation du cahier des charges.
 - Se réunir avec la cliente afin de valider le cahier de charges qu'on a fait.
- Seconde réunion avec notre cliente pour quelques questions (cliente pas disponible => étape non effectuée).
- Réalisation de toutes les pages de la maquette avec les différentes vues.

Ce qui aurait dû être fait :

- Demandez plus de rendez-vous avec la cliente.
 - Ce qui n'était pas possible car on était dans la limite de l'harcèlement.
- Prévoir plus de questions pour avoir plus d'informations.
 - Ceci concerne les cas personnalisés de l'application qu'on n'a pas pu aborder avec la cliente dans les premiers entretiens.
- Prendre des initiatives d'analyse.
 - Certains cas personnalisés nous demandent d'être à la hauteur d'analyser la situation afin de pouvoir répondre.

6) Présentation des activités (Milestone 2)

Dans cette partie le plus grand travail a été effectué, on a fini les tâches dédiées au milestone 1 et effectué les tâches du milestone 2.

Ce qui a été fait :

- Traitement des retours Milestone 1.
 - Lors de la soutenance, on a eu plusieurs retours de la part des profs, ces retours nous ont servi d'informations à utiliser afin d'éviter les mêmes erreurs.
- Réunion avec la cliente pour les différentes parties à modifier
 - Refaire une réunion avec la cliente afin d'avoir des réponses à nos questions (il s'agissait d'une amélioration de l'architecture).
 - Traitement et prise en compte de ces retours.

- Réalisation du modèle conceptuel de données.
 - Un nouveau modèle de données afin de l'utiliser pour désigner l'architecture de l'application.
- Validation du cahier des charges.
 - On a finalement réussi à valider le cahier des charges avec notre cliente, il s'agit d'un cahier des charges final.
- Mise en place de la maquette.
 - Tracer une maquette à suivre pour la partie frontend.
- Mise en place d'une charte graphique
 - Choisir une charte graphique afin d'animer le front avec les couleurs nécessaires.
- Mise en place d'une nouvelle arborescence de l'application
 - Le changement de l'architecture a impliqué un changement dans la totalité de l'arborescence de l'application.
- Réunion Tuteur
 - Se réunir avec notre tutrice afin de lui montrer l'avancement du travail et avoir des retours.
- Traitement des retours tuteurs
 - Prise en compte des remarques faites par la tutrice.
- Réunion Démo /client.
 - Mise en place d'une réunion avec la cliente afin de lui montrer l'état d'avancement du travail et le comportement de l'application.
- Rédaction des différents documents :
 - Rédaction d'un guide utilisateur
 - Rédaction de la documentation technique
 - Rédaction du livrable finale
 - Réalisation du powerpoint de soutenance

Ce qui a été fait en développement:

- Développement des fonctionnalités de la gestion des utilisateurs.
 - Connexion / Déconnexion
 - Changement du mot de passe
 - Inscription d'un nouveau responsable d'année
 - Inscription des étudiants du Master 1
 - Gestion du passage des étudiants du Master 1 au Master 2
 - Gestion des redoublements
 - Suppression des étudiants ayant validé leurs Master 2
- Développement des fonctionnalités de la gestion de la maquette
 - Gestion des parcours
 - Gestion des blocs
 - Gestion des UEs
- Développement des fonctionnalités de la gestion des choix d'options
- Développement des fonctionnalités de la gestion des périodes de choix d'options
- Développement des fonctionnalités de la gestion des étudiants / groupe d'étudiants
 - Gestion des choix d'options définitifs (Ajouter / Supprimer).
 - Gestion des choix d'options définitifs (Modifier).
 - Consultation des choix d'options définitifs.
 - Gestion de la génération des groupes d'étudiants dans une UE.
 - Modifier des groupes d'étudiants dans une UE.
 - Supprimer un groupe d'étudiants dans une UE.
 - Exportation d'un fichier Excel contenant les groupes d'étudiants dans les UEs.

Ce qui aurait dû être fait :

- Plus de réunions avec la tutrice afin de nous permettre de mieux perfectionner le parcours utilisateurs.
- Mieux analyser le besoin et approfondir l'étude de l'existant avant la phase de conception car cela permet de gagner plus de temps.

7) Contact avec la cliente

Lors du milestone 1, nous n'avons eu qu'un seul entretien avec la cliente, et celui-ci nous a permis d'identifier, de cerner les besoins, connaître les différents utilisateurs et l'objectif de l'application.

Nous avons fait un compte rendu de cet entretien accessible au lien suivant : [compte rendu](#).

Après la soutenance du milestone 1, la première des choses que nous avons faite c'est d'envoyer une première version de notre cahier des charges à notre cliente pour une correction.

Ainsi le **13 Décembre** nous avons eu une réunion sur discord, où nous avons discuté de tous les changements qu'on avait à apporter dans le premier jet du cahier des charges.

Le **02 Janvier 2023** nous avons envoyé un second jet du cahier des charges en ayant apporté toutes les modifications demandées lors du précédent entretien et le **09 janvier**, nous avons reçu une réponse de notre cliente qui nous confirmait bien qu'elle validait le cahier des charges qu'on lui avait envoyé et que c'était du bon boulot.

Avec notre cliente, vu son indisponibilité, nous avons beaucoup plus échangé via le webmail de l'ENT ou encore par la plateforme Discord.

Pour finir on a eu rendez-vous avec notre cliente pour le 31/02/2023 afin de lui montrer toutes les fonctionnalités de l'application plus son architecture, et de manière globale, elle était satisfaite du travail effectué.

8) Organisation interne au sein de l'équipe

Nous avons ici la liste de tous les membres du groupe, chacun avec le rôle qu'il joue pour la réalisation de ce projet.

- **Product Owner / Tech Lead** : Abdelhalim Oulad Mansour

Il organise et conduit le projet de bout en bout. Il est la personne à qui il faut rendre compte des modifications apportées, de l'avancement des tâches, etc... Le product owner pourra également communiquer des informations plus précises aux clients. Il travaille en totale collaboration avec toute l'équipe et participe aussi à la réalisation des user stories.

Il impulse le projet en rédigeant les documents techniques de conception. Une fois le projet lancé, il joue un rôle de facilitateur puisqu'il est amené à corriger de nombreux problèmes techniques.

Adresse mail : abdelhalim.oulad.mansour@etud.u-picardie.fr

- **Scrum master** : Abdoul Aziz Traore

Il aide à animer la mêlée (Scrum) pour l'équipe dans son ensemble en s'assurant que le framework Scrum est respecté. Il s'engage à respecter les valeurs et les pratiques Scrum, mais doit aussi rester flexible et ouvert aux possibilités d'amélioration du workflow de l'équipe. Il assiste le product owner au niveau des différents livrables et participe aussi à la réalisation des user stories.

Adresse mail : abdoul.aziz.traore@etud.u-picardie.fr

- **Développeur Full stack** : Mouhcine bannani

Il assiste le tech lead dans la réalisation des fonctionnalités techniques, il réalisera de son rôle de nombreux tickets et travaillera en totale collaboration avec le tech lead pour réaliser toutes les user stories.

Adresse mail : mouhcine.bannani@etud.u-picardie.fr

- **Business Analyst Proxy Product Owner : Mamadou Binnet BARRY**

Il assiste le product owner dans tout son travail, c'est lui qui rédige les spécifications fonctionnelles et les user stories dans Jira, il assure le maintien du backlog.

Adresse mail : mamadou.binnet.barry@etud.u-picardie.fr

Pour la réalisation de ce projet au sein de notre groupe, nous utilisons différents outils nous permettant de communiquer avec tous les autres membres du groupe, de garder une trace de toutes les différentes activités qui ont été menées, nous pouvons citer entre autres :

Jira : est un outil collaboratif permettant de gérer des projets et des tâches en équipe. Il s'agit d'un tableau digital de gestion de projet qui répartit toutes les tâches, sous forme de cartes, au sein de colonnes "à faire", "en cours" et "fait". Ces tâches sont assignées aux différents membres de l'équipe en charge du projet. Les membres du projet ne font que glisser et déposer les cartes correspondantes d'une colonne à l'autre en fonction de comment il avance sur le ticket qui lui est alloué.

Cet outil nous est très bénéfique, car il nous permet de découper nos tâches en sous-tâches et de voir comment évolue le projet comme nous le montre la figure ci-dessous :

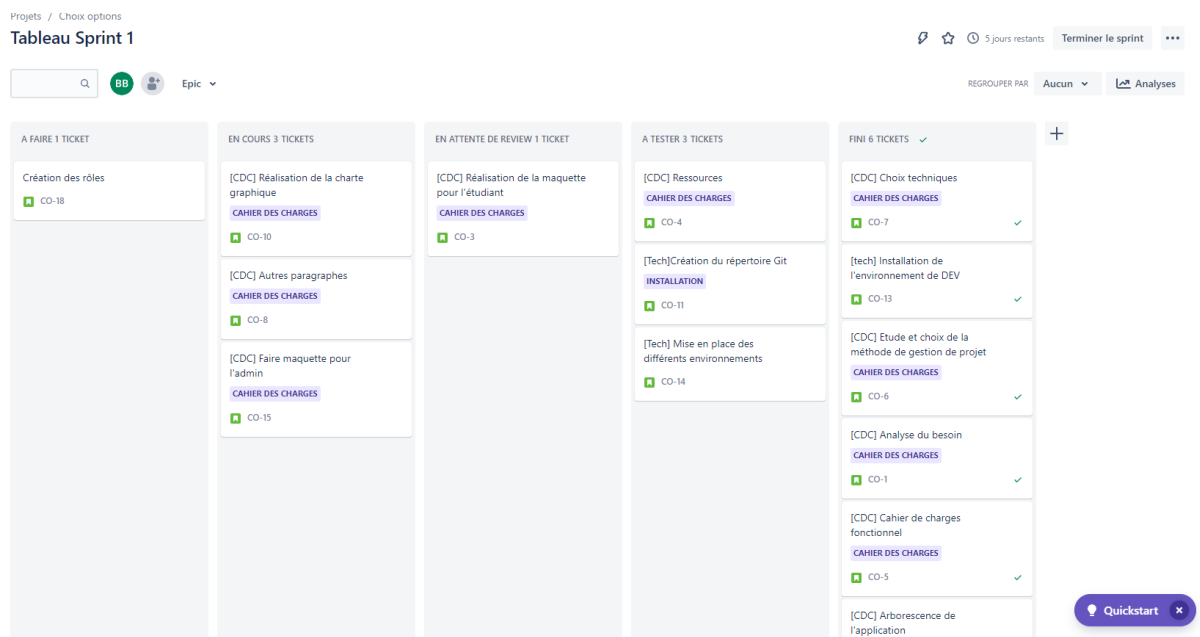


Figure 2 : Avancement du sprint 1 sur Jira lors de la rédaction du cahier des charges.

Nous avons aussi un backlog comportant plusieurs tickets pour justement anticiper les futurs sprints et les user stories.

Drive : Google Drive est un service stockage et de partage de fichiers en ligne. Ce service nous permet de récupérer, modifier et utiliser nos documents à tout moment, n'importe où et même sans connexion. C'est dans ce drive que nous rédigeons tous nos différents livrables.

Nous avons créé un projet avec à l'intérieur tous les documents que nous pouvons échanger entre nous. Google drive nous facilite énormément la rédaction de nos différents livrables, surtout que nous sommes tous à distance.

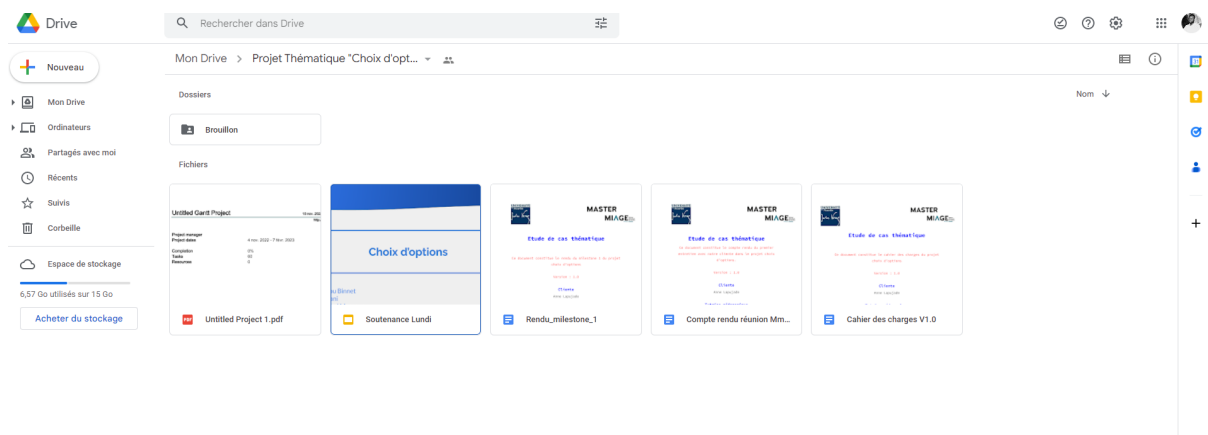


Figure 3 : Image de notre drive avec les différents documents réalisés pour le moment

Discord : C'est grâce à discord que nous organisons toutes les différentes réunions au sein du groupe, que ça soit pour présenter des sprints, résoudre un problème, se planifier pour le prochain sprint ou encore avoir des réunions avec notre cliente et notre tutrice.

Nous avons créé un serveur et disposons de plusieurs salons textuels et vocaux pour pouvoir discuter entre nous.

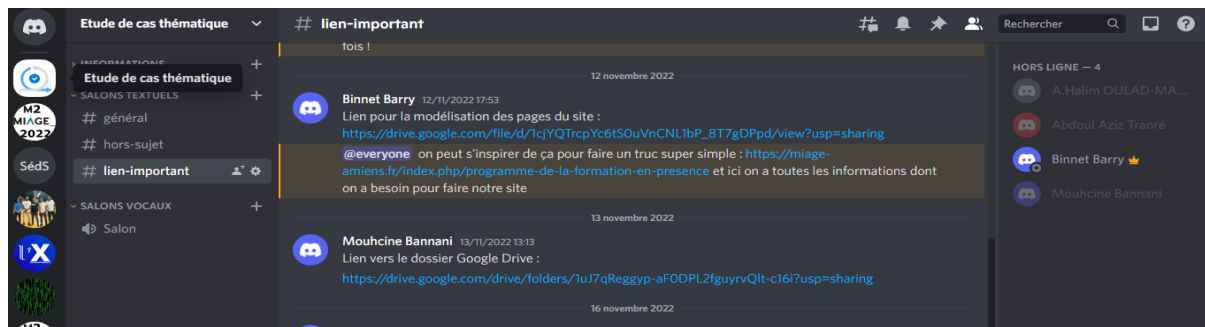


Figure 4 : Image de notre discord avec les différents salons.

9) Prise de recul

A l'issue de cette première phase du projet nous avons souligné plusieurs points importants.

1) Concernant le déroulement de projet en général :

- La différence entre la réalisation d'un projet dans le cadre académique et le cadre professionnel (le cas du projet thématique).
- L'avantage qu'on avait en tant qu'équipe de développement était le choix des technologies, le framework symfony était notre choix technique, les développeurs maîtrisent ce framework et savent avancer sans avoir besoin d'une formation ou montée en compétences.

- En effet lorsque nous réalisons des projets académiques, ces derniers étaient plus guidés et pré-découpés de manière à ce qu'on n'ait pas à réfléchir de l'organisation des différentes étapes, contrairement au projet thématique qui nous a demandé beaucoup d'organisation et de structuration par nous-même en tant qu'équipe de développement et gestion de projet, ainsi qu'on a été amenés à chercher les informations afin de bien répondre aux besoins de notre cliente.
- En matière de choix de méthodes de la gestion de ce projet, il a fallu prendre en compte les contraintes présentes afin de viser un compromis entre efficacité, qualité et rapidité.

2) Concernant la méthode de gestion de projet :

- **Entre les membres de l'équipe :**
 - Étant étudiants de la MAGE bien connaissant les méthodes agiles on a choisi la méthode agile/scrum entre les membres de l'équipe pour assurer la communication permanente et le bon déroulement du développement de l'application, ceci était très bénéfique à l'avancement du projet car la diffusion des informations était très fluide ce qui a impliqué des prises de décisions très rapides.
- **Avec la cliente (Mme LAPUJADE) :**
 - Vu la particularité de notre cliente qui est une professeure à l'université et donc souvent non-disponible, on a été obligé d'adopter la méthode de **Cycle-V** afin qu'on puisse avancer en comptant absolument sur le cahier des charges et sans avoir un suivi avec la cliente.

- Cette méthode a été personnalisée car on avait besoin d'une validation du cahier des charges et donc on a réussi à avoir quelques entretiens avec la cliente afin d'avoir des retours, le cahier des charges est un élément majeur dans la méthode cycle-V afin de bien commencer le développement mais ça nous a pris énormément de temps pour le finir (plus que prévu) et pour cela le développement des fonctionnalités évidentes avait commencé avant la validation final du cahier des charges afin d'éviter la perte de temps et finir en retard.

3) Concernant le développement et la partie technique :

- Les acteurs principaux du développement étaient nos deux développeurs **“Halim et Mouhcine”**.
- L'avantage en matière de temps et d'avancement c'est que ces derniers maîtrisent le framework de Symfony et n'avaient pas besoin d'une formation ou monter de compétences.
- Le début du développement était globalement bien passé, certaines fonctionnalités étaient dédiées à Halim, d'autres à Mouhcine afin d'avancer en même temps et gagner du temps et certaines fonctionnalités étaient à faire par les deux développeurs.
- Pour les fonctionnalités développés par un seul développeur on a utilisé l'outil de versionning Git/Github afin de fusionner le code de chaque développeur avec l'autre et c'était la partie la plus compliquée dans le développement, il s'agit des conflits Git (Deux codes qui ne matchent pas) ainsi que la différence entre les manières de pensée de chaque développeur, ça nous est arrivé de commencer des tâches en développement chaque développeur dans son coin et au moment de la fusion on se retrouve avec des évolutions d'architectures différentes et on a fini à plusieurs reprise par éliminer une évolution d'un des développeurs et qu'il refasse la tâche sur l'évolution de son collègue, cela était tellement frustrant au point ou on a été obligé de changer la manière d'avancement, on a décidé que c'est un seul développeur qui avance et l'autre prendra le relais que quand son collègue a fini la tâche

complètement afin d'éviter les conflits, ceci était efficace mais il a rendu le travail plus long vu que c'est toujours qu'une personne qui avance.

4) Concernant la partie fonctionnelle de l'application :

Pour cette partie, la principale difficulté était de pouvoir rencontrer notre cliente pour pouvoir comprendre son besoin. Ensuite pour la rédaction du cahier des charges nous avons partagé le sommaire et chaque membre a rédigé sa partie puis, nous avons fusionné les parties. Ce qui a provoqué d'énormes redondances dans le texte et plusieurs parties qui ne servaient à rien, ce qui explique que notre première version du cahier des charges n'a pas été validé par la cliente. Pour résoudre ce problème nous nous sommes structurés à nouveau de telle sorte que c'est une seule personne qui rédige les rendus, les autres membres du groupe ne peuvent laisser que des commentaires.

La seconde difficulté était de comprendre tout le besoin fonctionnel de l'application en prenant en compte toutes les différentes possibilités et proposer des solutions, c'est le cas par exemple du la relation bloc - UE dont la remarque nous a été faite par notre tutrice.

10) Conclusion

Comme tout projet, il est primordial de toujours prendre du recul, de choisir la conduite adaptée en fonction du client et de la situation, de bien s'organiser, ainsi qu'il ne faut jamais négliger le temps à consacrer pour les différentes tâches.

Ce document était une présentation et description de toutes les fonctionnalités attendues du produit.

C'était également une description de toutes les ressources et les techniques qui permettront de comprendre le produit et le besoin exprimé par le client.

11) Annexes

11.1) Modèle conceptuel de données

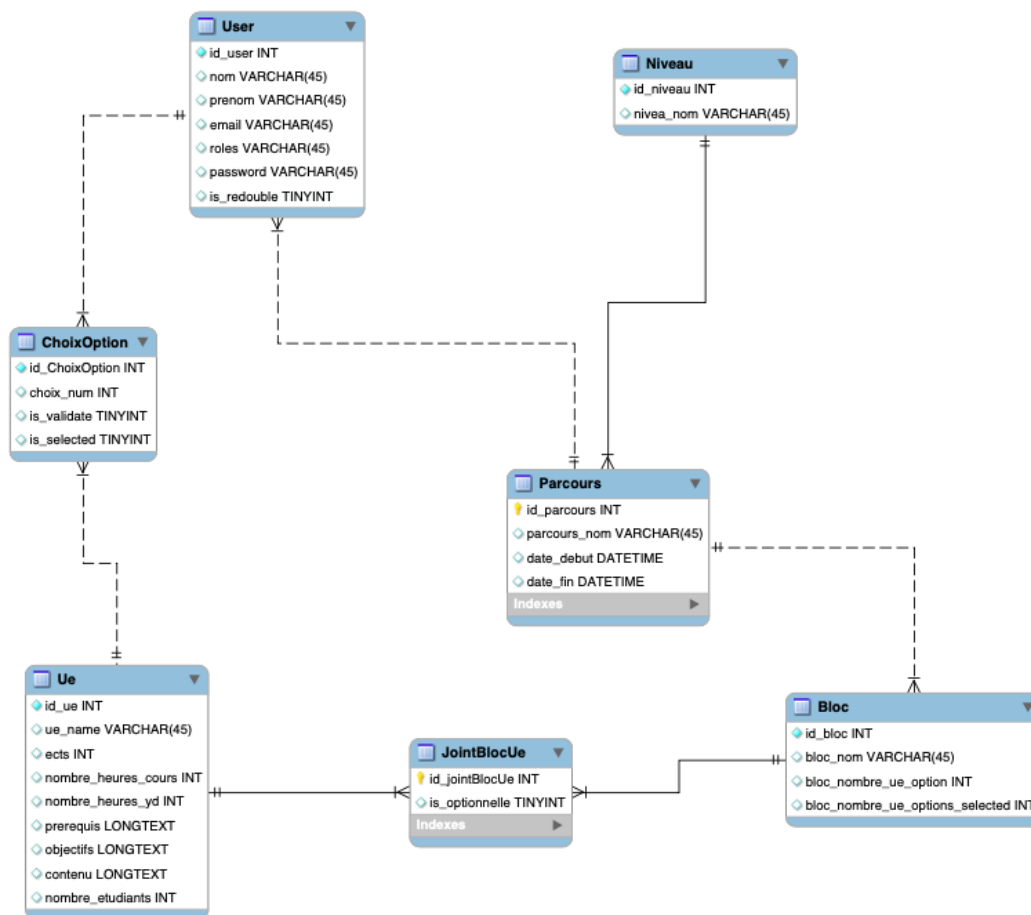


Figure 5 : Modèle conceptuel de données

Remarque lié au MCD

- La cardinalité représentée par une croix est équivalente à “**plusieurs**” et la cardinalité représentée par les 2 petits traits est équivalente à “**un seul**”.
- Il n'y a aucune différence entre les lignes continues et les lignes découpées.

Exemple : Un **User** à plusieurs choix d'options, un choix d'options à un seul **User**.

11.2) Diagramme de GANTT

Fonctionnalité	Nom abrégé de la fonctionnalité	Fonctionnalité précédente
Connexion/Déconnexion	A	/
Changement de mot de passe	B	A
Inscription d'un nouveau responsable de formation	C	B
Inscription des étudiants M1	D	C
Gestion du passage des étudiants Master 1 au Master 2	E	D
Gestion redoublement des étudiants	F	E
Supprimer les étudiants ayant validé leurs M2	G	F
Gestion des parcours	H	G
Gestion des bloc	I	H
Gestion des UEs	J	I
Gestion des choix d'options	K	J
Gestion des périodes des choix d'options	L	K
Gestion des choix d'options définitifs	M	L
Consultation des choix d'options définitifs	N	M
Gestion de la génération des groupes d'étudiants dans une UE	O	N
Modifier des groupes d'étudiants dans une UE	P	O

Supprimer un groupe d'étudiants dans une UE	Q	P
Exportation d'un fichier excel contenant les groupes d'étudiants dans les UEs	R	Q

Remarque : toutes ces fonctionnalités sont de type "Must", elles doivent être faites toutes.

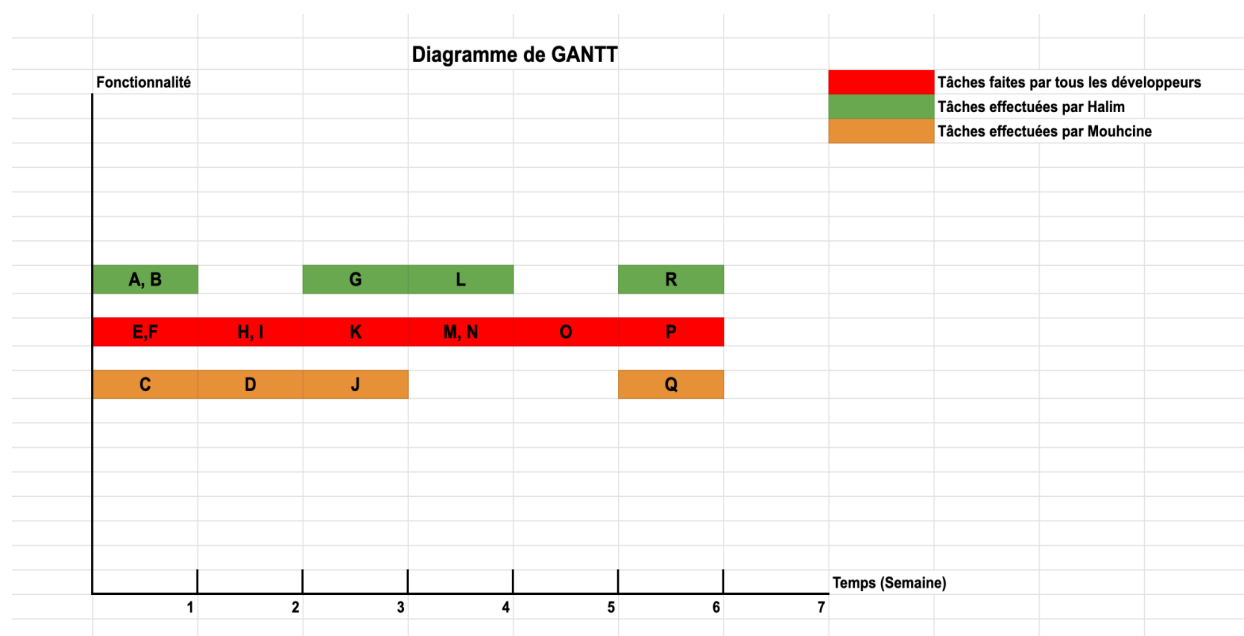


Figure 6 : Diagramme de GANTT (Développement)

Remarque : La semaine 1 est la semaine du **12/12/2022** et la date de fin est le **27/01/2023**.

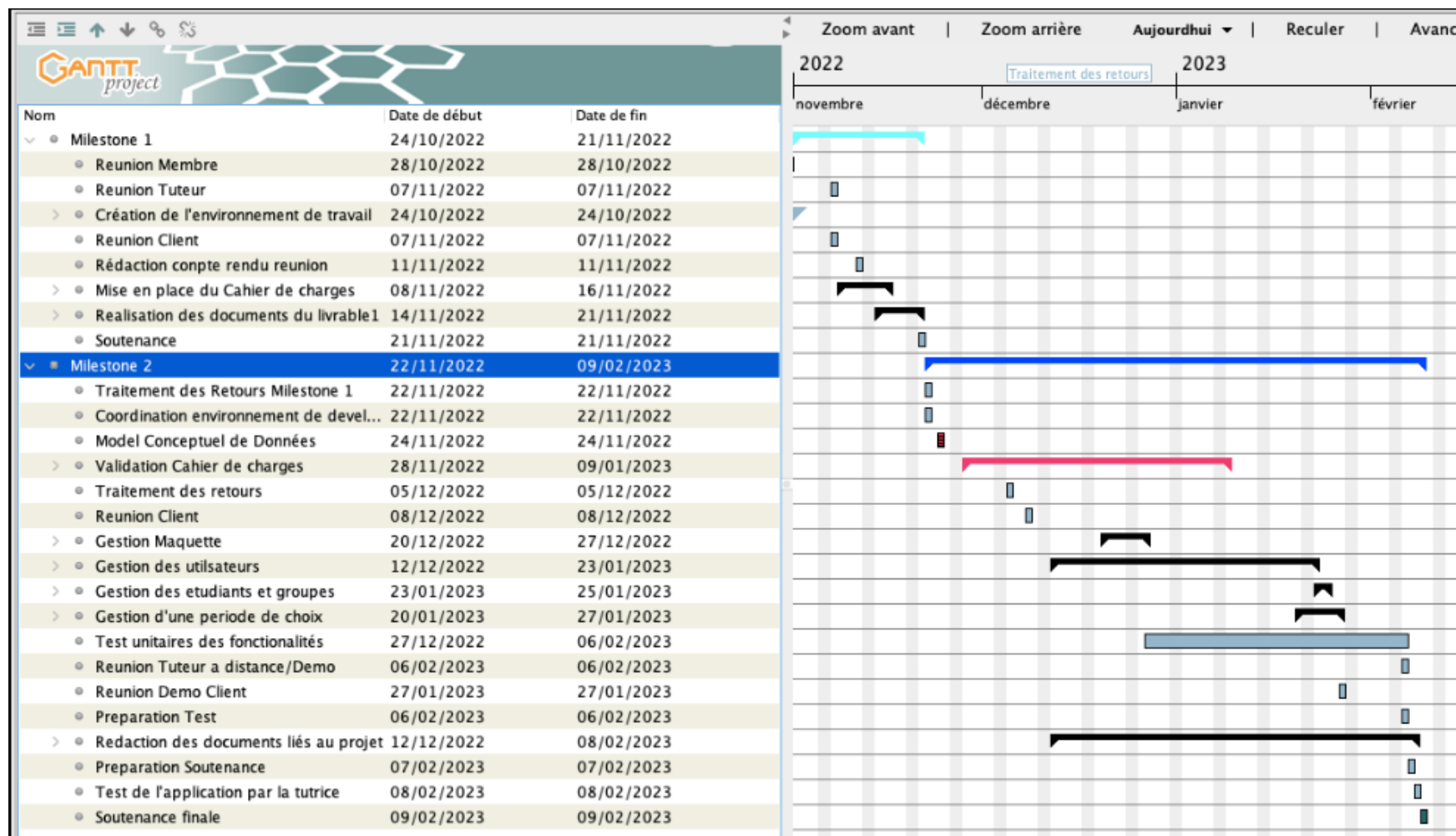






Figure 7 : Diagramme de GANTT général

Dans cette figure nous avons le diagramme de GanTT général sur tout le déroulé du projet, avec le nom des grandes fonctionnalités. Vous pouvez le consulter en intégralité avec plus de détails et le nom de toutes les sous-fonctionnalités en cliquant sur le lien suivant : [diagramme de Gantt version PDF](#)

All sprints

   0 jour restant [Terminer le sprint](#) 

Epic ▾

Sprint ▾

REGROUPER PAR

Aucun ▾


 Analyses

Figure 8 : Capture d'écran d'un sprint sur JIRA

Aperçu d'un sprint lors du développement des fonctionnalités, ce sprint avait pour but le développement des fonctionnalités liées à la gestion des étudiants / groupes d'étudiants

Diagramme de flux cumulé

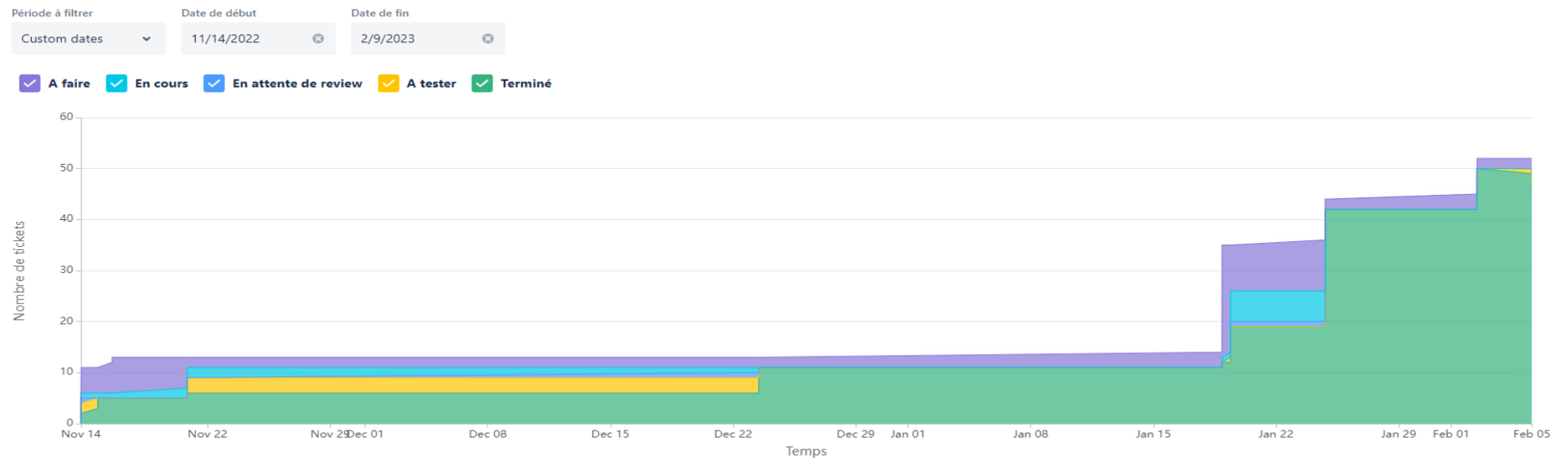
[Comment lire ce rapport](#)

Figure 9 : Diagramme de flux cumulé (Les sprints JIRA)

Ce diagramme de flux cumulé représente l'ensemble de l'évolution du projet, il a été automatiquement généré par Jira. Nous voyons une hausse importante du nombre de tickets passés à Done à partir de mi-janvier qui représente la validation du cahier des charges et donc la possibilité de développer toutes les autres fonctionnalités.

12) Autres annexes

- Cahier des charges (Fichier existe dans le zip).
- Documentation utilisateur (Guide d'utilisation) (Fichier existe dans le zip).
- Documentation technique (Fichier existe dans le zip).
- [Lien vers Jira](#).