Plan projet

Responsable

Julien WATTIER

Contributeurs

Théo DELMAS

Lauric TEYSSEYRE

Pierre-Louis RENON

Approbateurs

Frédéric MIGEON

Gilles LEPINARD

Université Paul Sabatier

Master Informatique 1

Table des matières

1 Objectif du document

Ce document a pour objectif d'exposer les méthodes de management appliquées à ce projet.

2 Contexte et objectifs du projet

2.1 Contexte

FlopEDT! est une application permettant de créer un emploi du temps satisfaisant une série de contraintes basées sur la programmation linéaire. Cependant la gestion actuelle de ces contraintes est relativement technique et peu documentée.

De plus les principaux collaborateurs du projet ont initié un découplage entre le back-end et le front-end, les deux parties étant actuellement gérées par un unique serveur.

2.2 Objectifs

L’objectif est de créer une interface intuitive permettant de rechercher des contraintes à ajouter à la génération d’un emploi du temps, tout en y intégrant la documentation des contraintes, le tout respectant le contexte de découplage front/back de l'application.

Ces demandes sont à l’initiative du principal développeur du projet Pablo SEBAN. Ce projet est donc à façon pour cette même personne.

2.3 Liste des parties prenantes

Les parties prenantes sont listées dans le tableau suivant, qui récapitule également leurs besoins principaux, leurs représentants ainsi que la validation des besoins et la recette.

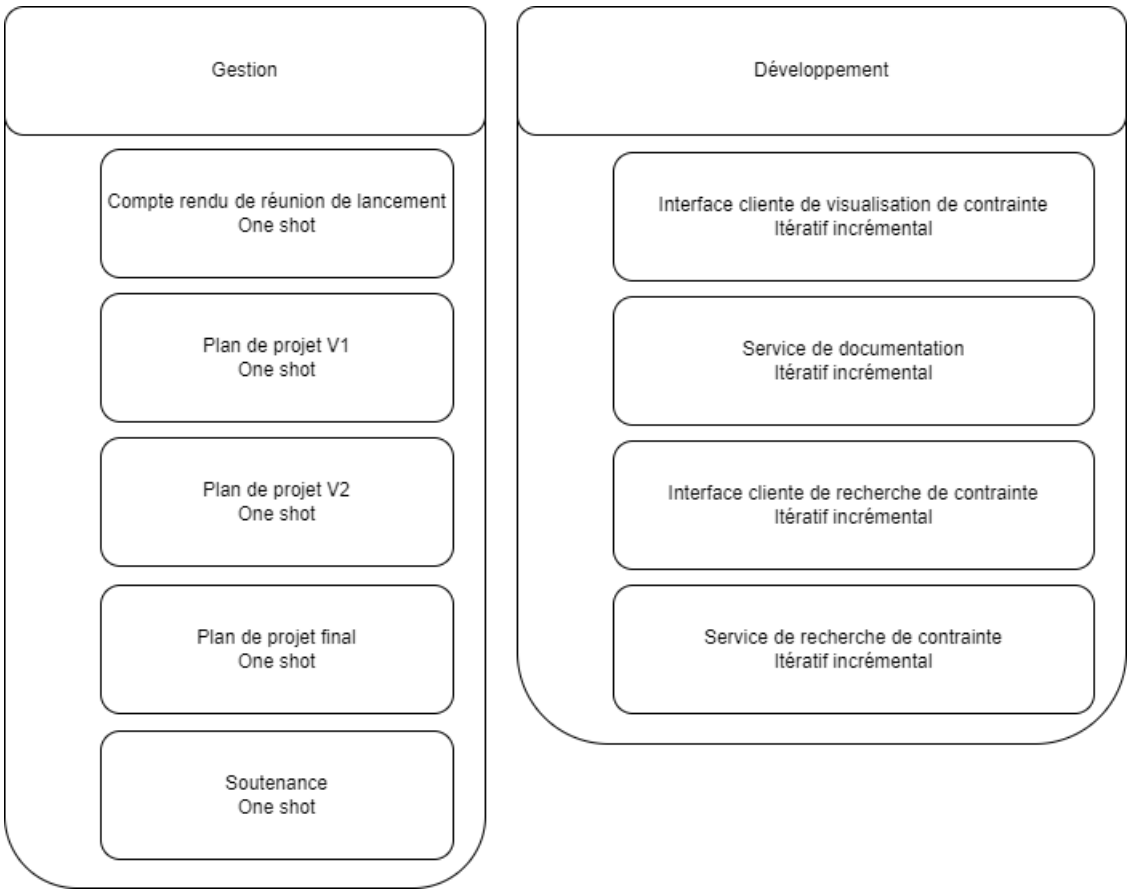
Partie prenante	**Besoins**	**Représentants**	**Validation**	**Recette**
Maître d'ouvrage	Obtenir une interface intuitive pour son produit permettant aux usagers finaux non techniciens de facilement trouver et ajouter des contraintes.	Pablo SEBAN	Voir la section de gestion des besoins	Voir la section de gestion des livrables
Collaborateur du projet	Obtenir une nouvelle version du système maintenable.			
Utilisateur final	Obtenir une nouvelle version du système utilisable avec efficacité.			
Fournisseur	Satisfaire au mieux les demandes du client. Monter en compétence.	Théo DELMAS Lauric TEYSSEYRE Pierre-Louis RENON Julien WATTIER	N'est pas géré par ce projet	
Assistant de maîtrise d'ouvrage	Répondre aux demandes de soutien de la part du maître d'ouvrage dans la gestion de ce projet.	Léo CUSSEAU Florian AZIZEN	N'est pas géré par ce projet	
Professeur	Obtenir des documents de gestion clairs et de qualité.	Frédéric MIGEON Gilles LEPINARD	Défini lors des cours relatifs à l'UE.	Évaluation itérative des documents tous les mois.

Les utilisateurs finaux et contributeurs seront considérés uniquement au travers du client.

3 Référentiel de management de projet

3.1 Product breakdown structure

Le product breakdown structure défini l’ensemble des produits que doit produire ce projet. Pour chaque produit la méthode de réalisation est présentée.

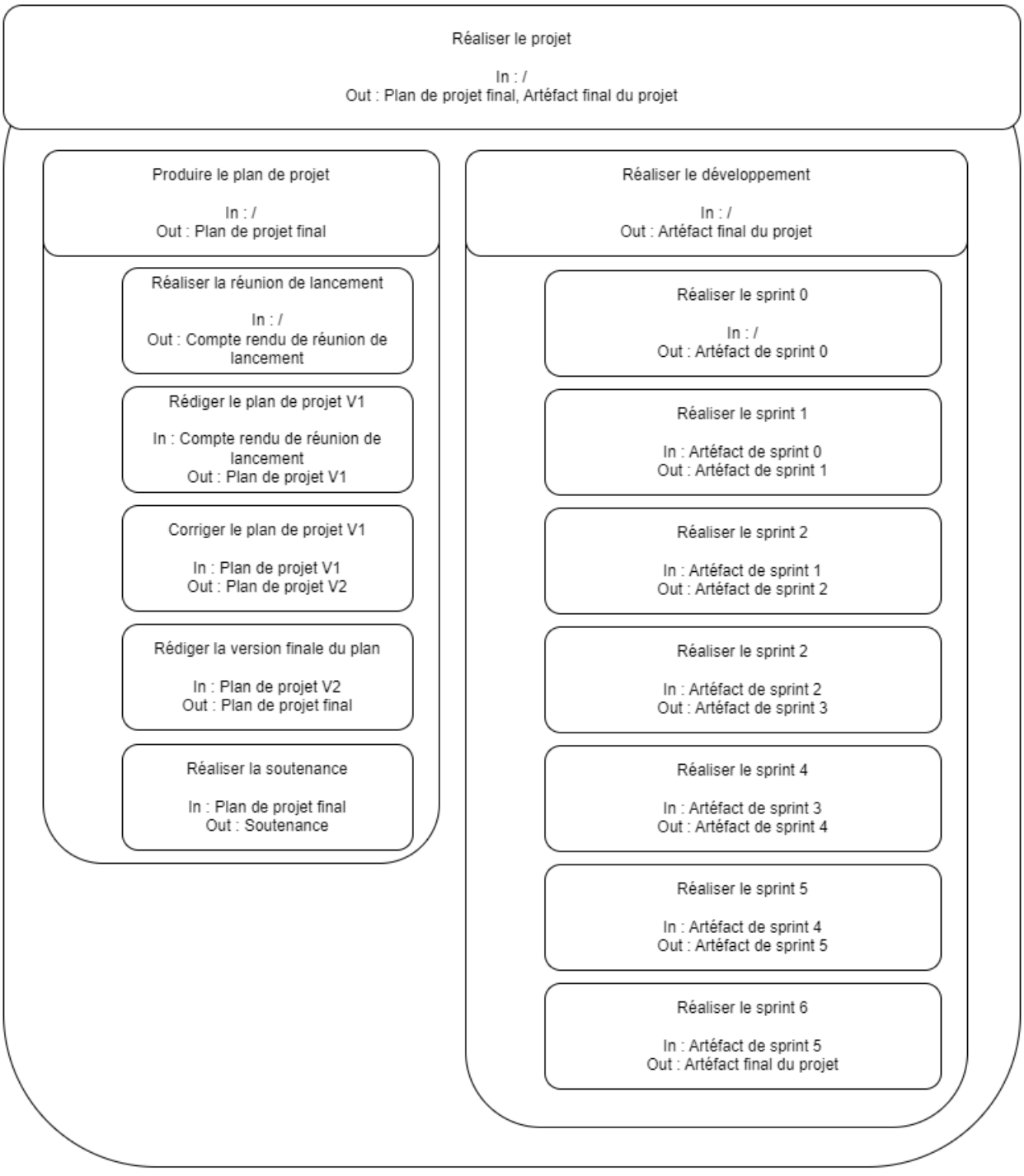


3.2 Activités du projet

3.2.1 Work breakdown structure

Le work breakdown structure défini l'ensemble des activités nécessaires pour réaliser ce projet. Il s'organise en deux parties : La première présente les activités relatives au plan de projet, et la seconde celles au développement, découpées en sprint.

Ces deux grandes parties seront effectuées en parallèle. Cependant le contenu de ces parties sera effectué de manière séquentielle à cause de leur dépendance. Chaque nœud de la structure précise quelles sont ses entrées et ses sorties.



3.2.2 Calendrier des ressources

Ce projet est sans ressources matérielles.

Concernant l'équipe, elle est disponible le jeudi et le vendredi sur la période fixe du 05/01/2023 au 28/04/2023. Les horaires restent souples mais chaque membre devra être disponible au moins 7 heures sur ces deux jours.

L'équipe sera en congé le 02/03/2023 et le 03/03/2023.

Il est également possible qu'elle ne soit pas disponible les deux dernières semaines de la période de projet en fonction des examens universitaires.

3.2.3 Liste des jalons

L'équipe effectue des sprints de deux semaines.

Le premier sprint est cependant plus long afin de préparer l'environnement du projet.

Le tableau suivant présente les différents jalons.

- Fin de sprint 0 (03/02/2023)
- Fin de sprint 1 (17/02/2023)
- Fin de sprint 2 (10/03/2023)
- Fin de sprint 3 (24/03/2023)
- Fin de sprint 4 (07/04/2023)
- Fin de sprint 5 (21/04/2023)
- Fin de sprint 6 (28/04/2023)

Concernant la partie management, les jalons sont relatifs aux dates de rendu du plan projet. Ils sont donc les suivants :

- 17/02/2023 : premier rendu intermédiaire du plan projet.
- 17/03/2023 : second rendu intermédiaire du plan projet.
- 14/04/2023 : rendu final du plan projet.

3.2.4 Prévision de la charge de travail

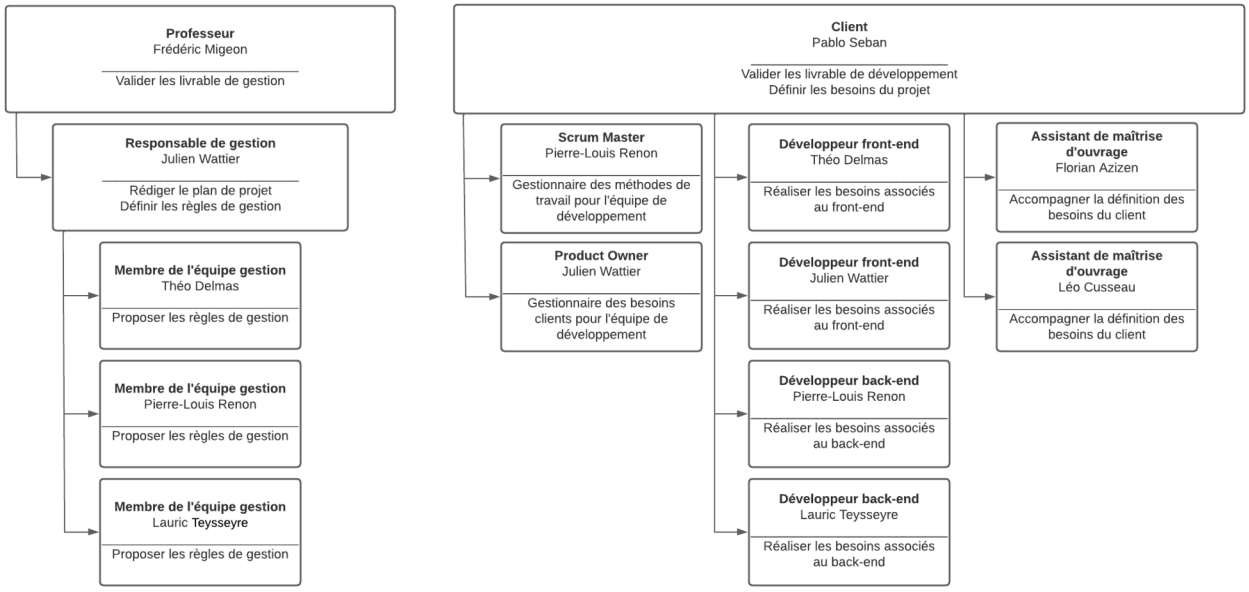
Le haut degré d'inconnu dans ce projet fait qu'aucune prévision de travail (Gantt) n'a été établie car risquant d'être trop éloignée de la réalité.

Pour ce qui concerne les tâches plus « atomiques », faute d'expérience dans le développement et dans la gestion, l'équipe n'appliquera pas de prévision de la charge de travail car cette dernière serait certainement erronée et ferait perdre du temps à l'équipe.

3.3 Rôles et responsabilités

3.3.1 Organizational breakdown structure

Le schéma ci-dessous présente une hiérarchie des différents rôles de ce projet, leurs représentants ainsi qu'une description de chacun.



3.3.2 Matrice RACI

La matrice suivante définie les responsabilités de chaque activité pour chaque rôle. Cette responsabilité est défini selon la codification suivante :

R	A	C	I
Est responsable de la tâche.	Valide ou non la tâche lorsqu'elle est réalisée	Contribue à la réalisation de la tâche	Est informé des avancements et décisions relatives à la tâche

	Professeur	Responsable de gestion	Membre de l'équipe de gestion	Client	Scrum Master	Product owner	Développeur front-end	Développeur back-end	AMO
Réaliser la réunion de lancement	A	R	C	C					C
Rédiger le plan V1	A	R	C						
Corriger le plan V1	A	R	C						
Corriger le plan V2	A	R	C						
Rédiger la version finale du plan	A	R	C						
Réaliser la soutenance	A	R	C						
Réaliser le sprint 0				A	R	R	C	C	I
Réaliser le sprint 1				A	R	R	C	C	I
Réaliser le sprint 2				A	R	R	C	C	I
Réaliser le sprint 3				A	R	R	C	C	I
Réaliser le sprint 4				A	R	R	C	C	I
Réaliser le sprint 5				A	R	R	C	C	I
Réaliser le sprint 6				A	R	R	C	C	I

3.4 Suivi du déroulement

3.4.1 Gestion des décisions

Les décisions relatives aux projet sont traçables dans le registre des décisions.

Chaque décision y est décrite par sa description, qui précise ce qu'elle a modifié dans le plan projet, la justification/le contexte ayant nécessité cette décision ainsi que la date.

Lorsqu'une décision est créée, le plan projet est alors actualisé conformément à la décision et mis à jour sur l'outil de gestion de version associé.

3.4.2 Gestion des actions

Le suivi des actions à réaliser/réalisé est disponible sur le kanban grâce aux listes de tâches qu'il est possible d'adjoindre à un besoin.

3.5 Gestion de la communication

3.5.1 Communication avec les parties prenantes

Les communications avec les professeurs se feront de manière informelle lors de TD dédié à l'UE de projet.

Pour les communications avec le client, elles se feront au travers du discord de FlOpEDT, que ce soit de manière textuelle pour des questions spontanée, ou vocale pour ce qui concerne les revues de sprint. Ces dernières se tenant dans les jours suivant la fin d'un sprint, fonction des disponibilités de chacun.

Les revues de sprint seront enregistrées, puis traitées ultérieurement afin de produire un compte rendu de réunion référencé dans le registre des rapports de réunion.

Le reste des parties prenantes ne sera pas informé de l'avancée du projet, mais pourra le suivre via ce dépôt distant qui trace le plan de projet et les documents liés.

3.5.2 Communication au sein de l'équipe

L'équipe communique au travers d'un serveur discord dédié de manière informelle.

Au début de chaque journée, une daily meeting informelle est organisée pour résumer le travail effectué par chacun lors de la dernière journée, les difficultés rencontrées et le travail prévu sur la journée.

Par la suite les membres sont tous présents en vocal sur ce même serveur, et partage éventuellement leur écran afin de montrer ce qu'ils font.

Enfin, pour faciliter le débugage, l'outil de peer-programming liveshare sera utilisé. Chaque développeur devra lancer une session de partage permettant aux autres membres d'avoir accès à leur code local.

3.6 Gestion de la qualité du management

3.6.1 Qualité des produits

Les différentes versions du plan de projet sont revues par tous les membres de l'équipe avant soumission.

Chaque version du plan est également soumise à une évaluation interne grâce à une check-list fournie par les professeurs. Cette liste conserve les réponses des anciennes versions afin d'évaluer l'évolution du plan.

Enfin, chaque version est revue par les professeurs.

3.6.2 Qualité des activités

L'équipe ne supervise pas les activités de management.

3.7 Gestion des risques et opportunités

3.7.1 Référencement

La liste des risques et opportunités est listée dans le registre des risques et opportunités.

Chaque risque y est décrit par les champs suivants :

- Titre du risque.
- Description du risque.
- Risque d'occurrence : sur une échelle faible - moyen - fort.
- Niveau d'impact : sur une échelle fable - moyen - fort
- Plan d'action : actions à mettre en place si le risque se déclenche ou est sur le point de se déclencher.

S'y ajoute les champs suivants si il y a déclenchement du risque :

- Date de déclenchement.
- Contexte de déclenchement.

3.7.2 Identification des risques

Les risques ont été identifiés lors de l'élaboration de la note de cadrage.

Si, lors de la capture d'un besoin, un risque apparaît comme évident aux fournisseurs, il sera ajouté dans le registre. Mais l'équipe ne dédiera pas particulièrement de temps à l'identification des risques.

3.7.3 Réponse aux risques

En cas de déclenchement d'un risque, l'équipe fournisseur suivra le plan d'action prévu dans le registre. Elle veillera également à remplir les champs associés au déclenchement dans le registre.

Dans le cas de certains risques relativement génériques, l'équipe remplira plusieurs fois les champs de déclenchement.

4 Référentiel de développement de projet

L’équipe applique une méthode de développement hybride qui mixe le SCRUM et le kanban. Dans cette méthode, le travail est segmenté temporellement en sprint afin de ne pas perdre de vue le temps. Cependant elle gère les besoins en kanban. Il n’y a donc pas d’estimation de valeur des user stories pour savoir laquelle développer, ni d’objectif de durée pour réaliser un besoin.

4.1 Gestion des besoins

4.1.1 Référencement

Les besoins sont suivis grâce à ce backlog produit . Celui-ci se décompose en deux parties. La première définit les principaux produits du projet et permet aux parties prenantes de comprendre sur quel produit l'équipe travaille actuellement. Les produits ayant été définis lors de la kick-off meeting.

La seconde est un backlog lié au produit en cours. L'ensemble des besoins y sont retranscrits et décrits grâce à des user stories sur lesquelles l'équipe fournisseur adjoint une liste de tâches à effectuer afin de satisfaire la user story.

Le backlog se divise en 5 colonnes :

- Idée floue : La user story nécessite d'être raffinée. Cette colonne permet de noter informellement les besoins du clients
- À faire : La user story a été définie mais pas choisie pour le sprint en cours.
- En cours : La user story a été choisie pour le sprint en cours.
- A tester : La user story est satisfaite mais nécessite une validation via des tests unitaires et d’intégration, ainsi qu’une approbation de la part du client.
- Ok : La user story a été testée et validée.

Chaque user story est priorisée via sa position dans son couloir, et relativement aux besoins du client. Cette priorisation est définie par le client.

4.1.2 Capture

Une première partie des besoins a été capturé lors de la kick-off meeting et sa préparation.

Par la suite les besoins seront capturés et définis précisément lors des revues de sprint en fonction des remarques du client sur ce qui est présenté.

Si un nouveau besoin fait apparaître un nouveau produit, alors une décision sera émise pour faire changer le référentiel.

Le client défini la nouvelle priorité des besoins à réaliser à chaque sprint.

4.1.3 Validation

Lors des revues de sprint le client valide si la fonctionnalité est suffisamment élaborée pour être considérée comme terminée.

4.1.4 Gestion des changements

Les changements dans la liste des tâches à effectuer pour chaque besoin ne sont pas tracés.

En revanche l'ensemble des besoins annulés sont récupérables dans l'archive du kanban.

Chaque développeur peut manipuler le kanban sans autorisation particulière.

4.2 Gestion du développement

Le développement suivra le workflow git. Il y aura donc une branche maîtresse intégrant les releases (artefact d'un sprint), une branche de développement intégrant les features en cours de développement et des branches de features correspondant au traitement d'un besoin.

4.3 Gestion de la configuration

Les produits de ce projet seront tracés à l'aide de git, sur le dépot officiel de FlopEDT, et en particulier sur la branche catalogDev

Chaque développeur dispose des droits et responsabilités sur les actions qu'il entreprend relativement à ce dépôt.

4.4 Gestion des livrables

Lorsque les fonctionnalités seront considérées comme terminées elles seront intégrées dans la branche maîtresse, ce qui marquera la livraison au client.

L'équipe s'assurera d'intégrer uniquement des fonctionnalités terminées donc fonctionnelles et testées.

4.5 Gestion de la qualité

4.5.1 Front-end

Le code produit est soumis à l'outil Prettier.

Il est également documenté dans la mesure du possible avec JSDoc.

4.5.2 Back-end

4.5.2.1 API Restfull

L'essentiel des développements back-end consiste à produire des API utilisées par le front-end. Ces API seront développées selon la convention RESTFULL.

4.5.3 Revue de code

L’équipe tire profit des fonctionnalités de l’outil de gestion de version (voir gestion de la configuration) en imposant une revue de code avant d’intégrer une fonctionnalité dans la branche de développement. L’intégration doit être validée par au moins deux développeurs différents de celui qui a développé la fonctionnalité.

5 Bilan du projet