

# SAÉ S4 - Plan de management de projet

Équipe pédagogique BUT Informatique de Rodez

# Table des matières

1. Introduction . . . . .	1
1.1. Rôle de l'enseignant tuteur . . . . .	1
1.2. Livrables produit, livrables projet . . . . .	1
1.3. Exigences produit, exigences projet . . . . .	1
2. Gestion de projet agile . . . . .	2
2.1. Itérations et activités . . . . .	2
2.2. Les rôles dans l'équipe . . . . .	2
2.3. Les exigences produit . . . . .	3
2.4. Les réunions . . . . .	4
2.5. La communication . . . . .	5
3. Les outils en soutien à la gestion de projet . . . . .	6
3.1. Git et Github . . . . .	6
3.2. Zenhub . . . . .	6
3.3. Google Drive . . . . .	6
4. Livrables . . . . .	8
4.1. Livrables produit . . . . .	8
4.2. Diaporama de la restitution en anglais du travail sur l'impact environnemental . . . . .	8
4.3. Livrables projet . . . . .	9

# Chapitre 1. Introduction

Ce document décrit l'ensemble des **exigences relatives à la gestion de projet** devant être satisfaites par les équipes étudiantes. Les livrables devant être produits dans le contexte du projet sont explicités.

Dans la suite du document, le terme “**enseignant tuteur**” désigne l'enseignant qui a donné le projet, ou dans le cas d'un projet externe, l'enseignant qui encadre le projet d'un point de vue technique. Le terme “**enseignant gestion projet**” ou “**coach agile**” désigne l'enseignant qui supervise la mise en œuvre du plan de management de projet.

## 1.1. Rôle de l'enseignant tuteur

L'enseignant tuteur est le référent technique du projet. A ce titre, il guide les étudiants dans leurs choix techniques, répond à leurs questions ou les oriente vers des ressources pertinentes. Dans le cas d'un projet interne, il joue aussi le rôle du client. En tant que client, il participe à la réunion Scrum nommée “revue de sprint”. Toutefois, des réunions supplémentaires avec l'enseignant tuteur viennent s'ajouter à celle-ci, à la demande des étudiants et/ou de l'enseignant.

## 1.2. Livrables produit, livrables projet

Dans la suite du document, le terme “produit” désigne la solution conçue par l'équipe pour répondre à la demande du client (le code source du logiciel, le ou les exécutableq, la documentation, etc.) Nous distinguons dans la suite du document les livrables produit des livrables projet. Les livrables produit constituent la solution livrée au client. Les livrables projet sont les livrables générés dans le cadre de l'activité de gestion de projet (compte-rendus de réunion, document de recette, etc.). Ce document vise à détailler la méthode, les livrables produit (dans leurs grandes lignes, les détails étant à préciser avec l'enseignant tuteur) et les livrables projet attendus dans le contexte du projet tutoré.

## 1.3. Exigences produit, exigences projet

De la même manière que nous distinguons les livrables produit des livrables projets, nous distinguons les exigences produits des exigences projets. Les exigences produit sont les exigences que doit satisfaire le produit. Les exigences projet sont les exigences que doit satisfaire le processus de gestion de projet. Les exigences produit spécifient le produit, les exigences projet spécifient le processus. Les exigences produit, dans le cadre de ce projet, sont décrites dans le cahier des charges. Ce document doit être considéré comme le document de spécification des exigences projet. Ce document est traditionnellement appelé le “plan de management de projet”. Dans la suite du document chaque exigence projet est identifiée par un code. La note de conduite de projet sera établie en prenant en considération le respect, ou non respect, de ces exigences essentielles (voir en annexe une grille d'autoévaluation). Ce document, dans le cadre de la démarche d'amélioration continue promue dans Scrum, peut être amené à évoluer dans le temps en fonction de vos avancées sur le projet.

# Chapitre 2. Gestion de projet agile

GEN001 : Le projet doit se dérouler suivant un processus agile inspiré de la méthode Scrum.

## 2.1. Itérations et activités

### 2.1.1. Itérations

Une itération ou sprint est une période durant laquelle l'équipe travaille à la construction du produit en y ajoutant des fonctionnalités.

À la fin de chaque sprint, une version du produit est montrable au client et aux autres parties prenantes afin d'obtenir des retours sur les fonctionnalités qui ont été implantées pendant le sprint. Ces retours permettent d'améliorer le produit. Dans Scrum, la durée des sprints est fixe. Elle est en général comprise entre 2 et 4 semaines. Une exception est faite pour la première période de mise en place du projet : cette période peut être plus longue et est baptisée "sprint 0".

### 2.1.2. Activités

Durant un sprint les membres de l'équipe prennent part aux différentes activités qui permettent de construire le produit : planification, conception, développement et tests définition des exigences produit.

GEN002 : Le sprint 0 ne doit pas excéder 4 semaines

GEN003 : La durée des sprints (hors sprint 0) devrait être de 3 semaines et ne doit en aucun cas excéder les 4 semaines.

## 2.2. Les rôles dans l'équipe

ROL001 : Tous les membres de l'équipe doivent participer de manière équilibrée au développement du produit.

ROL002: L'équipe doit désigner un membre responsable des exigences produits ; ce membre entretient un rapport privilégié avec le client et à ce titre, représente le client au sein de l'équipe ; dans Scrum, ce membre est appelé "Product Owner".

ROL003: L'équipe doit désigner un membre responsable de la bonne mise en œuvre du processus. Dans Scrum, ce membre est appelé "Scrum Master". Certains experts en méthode

agile désignent ce membre «Process Owner».

ROL004: Les rôles de Product Owner et de Scrum Master peuvent changer au cours du projet.

## 2.3. Les exigences produit

Les exigences produits sont composées d'exigences fonctionnelles et d'exigences non fonctionnelles.

### 2.3.1. Le backlog produit

Dans Scrum, l'essentiel des exigences est spécifié sous la forme d'une liste ordonnée par la priorité donnée par le client.

Dans Scrum, cette liste ordonnée est appelée "backlog produit" ou simplement "backlog".

Les exigences fonctionnelles sont les items du backlog correspondant à des fonctionnalités pouvant être décrite du point de vue de l'utilisateur. Dans Scrum, les exigences fonctionnelles sont décrites sous forme de **User Stories**. Ce format sera détaillé dans le cadre de la ressource "Analyse". Parmi les exigences non fonctionnelles, les exigences techniques apparaissent elles aussi dans le backlog.

Quand un item du backlog a été réalisé dans le produit (on parle d'item "fini" dans Scrum), l'item est alors sorti du backlog.

### 2.3.2. La définition de fini

Certaines exigences dites transverses ou omniprésentes (comme par exemple le fait qu'un site Internet soit compatible avec tel ou tel navigateur, le fait que toutes les pages d'un site doivent s'afficher en moins de 3 ms) ne peuvent pas être stockées dans le backlog, car elles ne sont jamais terminées. En effet, à chaque nouvelle fonctionnalité développée, il faudra vérifier que le produit satisfait à ces exigences omniprésentes. On les décrit alors dans un document dédié qui porte le nom de "Définition de fini".

EXI001: Les exigences fonctionnelles et les autres exigences non transverses doivent être gérées à l'aide d'un backlog.

EXI002: Les exigences transverses doivent être spécifiées dans la «définition de fini».

EXI003: Le Product Owner est responsable de l'alimentation du backlog et du document des exigences transverses tout au long du projet.

EXI004: Les User stories doivent respecter le format «En tant que... Je veux... De telle

sorte que...

EXI005: Les User stories doivent être détaillées à l'aide de tests d'acceptation. Les tests d'acceptation peuvent être illustrés par des maquettes basse fidélité.

## 2.4. Les réunions

Les sprints sont rythmés par différentes réunions :

- La réunion de planification : elle ouvre le sprint et permet à l'équipe de déterminer quels sont les items du backlog qu'elle pense pouvoir réaliser pendant le sprint. Cette liste d'items retenue pour le sprint en cours compose le "backlog de sprint". Les users stories devront être valorisées afin de savoir ce que l'on va pouvoir mettre dans le sprint ou pas. Pendant le S3, la valorisation se fera en heures. Au S4, la valorisation se fera en points d'efforts.
- Les "daily scrum" : cette réunion rapide, idéalement quotidienne, vise à partager entre tous les membres du projet ce qui a été fait sur le projet depuis le dernier daily scrum, ce qui est prévu jusqu'au prochain daily scrum et si il y a d'éventuels obstacles à l'évolution du produit. La résolution des problèmes ne doit pas être traitée pendant cette réunion, mais elle permet de prévoir un temps pour la résolution avec les membres de l'équipe les plus pertinents pour le ou les problèmes identifiés.
- La revue de sprint : cette réunion se déroule en fin de sprint pour montrer au client et aux autres parties prenantes intéressées la version du produit obtenue en fin de sprint. Le client peut alors valider les items de backlog finis ou émettre des réserves sur les items non satisfaisants. Les items validés par le client sortent du backlog. Les autres restent dans le backlog et pourront être sélectionnés de nouveau pour le sprint suivant. L'objectif de la revue de sprint est l'amélioration du produit.
- La rétrospective de sprint : cette réunion clôture le sprint et ne concerne que l'équipe de développement ; elle permet à chacun d'exprimer les points forts et les points faibles du processus révélés par le sprint venant de s'écouler. Des actions pour améliorer le processus sont décidées pendant cette réunion. Le plan de management et la définition de fini peuvent être modifiés à la suite d'une rétrospective. L'objectif de la rétrospective de sprint est l'amélioration du processus.

Quand une équipe "junior" est accompagnée par un "coach" agile, ce qui est le cas dans les projets des SAÉ, la réunion d'accompagnement par le coach s'ajoute aux réunions précédentes.

REU001 : Les réunions de sprint doivent être mises en œuvre durant toute la durée du projet.

REU002 : Les daily scrum peuvent se tenir tous les jours et doivent se tenir au moins 2 fois par semaine.

REU003: Toutes les réunions hors daily scrum doivent faire l'objet d'un compte-rendu

de réunion.

REU004 : Pour les réunions daily, tenir un journal dans un document texte (dans le drive) qui contiendra pour chaque réunion de ce type, la date et la liste des éléments qui ont été abordés pendant cette réunion par chaque participant.

## 2.5. La communication

La communication entre les membres de l'équipe, entre l'équipe et les autres parties prenantes s'effectue dans le cadre des réunions prévues par le processus mais aussi au travers de rencontres et d'échanges informels en présentiel ou à distance.

COM001: L'équipe doit mettre en place un outil (tel Slack, Discord, etc.) pour faciliter la communication synchrone ou asynchrone informelle à distance. Dans la suite du document cet outil est désigné "Outil de communication projet".

COM002: Toutes les parties prenantes peuvent communiquer via l'outil de communication projet.

# Chapitre 3. Les outils en soutien à la gestion de projet

Les outils sont indispensables à la mise en œuvre d'un processus. Cette section spécifie les outils à mobiliser pour gérer les livrables à produire.

## 3.1. Git et Github

GIT001 : Git et Github doivent être utilisés pour la gestion de version de code source.

GIT002 : Tous les membres du projet doivent avoir accès au projet comme contributeurs.

GIT003 : Toutes les parties prenantes du projet doivent avoir accès au projet Github a minima en consultation.

GIT004 : Certaines parties prenantes (client, enseignant tuteur) peuvent avoir accès au projet Github en tant que contributeur.

GIT005 : Le workflow <https://githubflow.github.io/>[\_Github flow\_] doit être utilisé pour la gestion des contributions des différents membres du projet. Les autres bonnes pratiques Git comme la bonne utilisation de fichiers ``.gitignore`` doivent être mobilisées.

## 3.2. Zenhub

ZEN001 : Le plugin Zenhub (installable sur les navigateurs les plus courants) doit être utilisé pour la gestion du backlog.

Afin de suivre l'évolution du travail pendant un sprint, on utilise un tableau (board agile ou kanban) permettant d'indiquer visuellement le changement d'état des items du backlog. Zenhub permet aussi de gérer le board agile.

ZEN002 : Le plugin Zenhub doit être utilisé pour la gestion du board agile.

## 3.3. Google Drive



DRI001 : Les documents bureautiques (comptes-rendus, documentation, etc.) des livrables produit ou projet doivent être conçus à l'aide de la suite bureautique Google et stockés sur un Drive attaché à un compte de l'IUT de Rodez appartenant à un des membres de l'équipe.

DRI002 : Ces documents sont regroupés dans un dossier partagé avec l'ensemble des parties prenantes (enseignant tuteur, client, enseignant suivi de projet).

Nom du dossier : SAE XX (No du projet) - Intitulé du projet de la SAE - S4.

# Chapitre 4. Livrables

Cette section récapitule les livrables devant être produits dans le cadre du projet. Certaines exigences projet non encore exprimées sont explicitées.

## 4.1. Livrables produit

### 4.1.1. Audit qualité de l'application

Voir exigence AUD001 du cahier des charges.

## 4.2. Diaporama de la restitution en anglais du travail sur l'impact environnemental

Voir exigence AUD007 du cahier des charges.

### 4.2.1. Les exigences produit

- Le backlog pour les user stories et autres exigences non transverses.
- Le document “Définition de fini”
- Les diagrammes de cas d'utilisation de la ou des applications à développer.

### 4.2.2. Code source

- Le code source hébergé sur Github.

### 4.2.3. Machines virtuelles

- Les images des machines virtuelles exigées par le cahier des charges.



Lieu de stockage reste à définir.

### 4.2.4. La documentation

DOC001 : Documentation utilisateur (peut être multiple, administrateur, utilisateur, ...). À chaque sprint ayant ajouté des fonctionnalités utilisateurs, la documentation utilisateur doit être maintenue à jour.

DOC002 : À chaque sprint, la documentation technique doit être maintenue à jour.

DOC003: La documentation technique doit être composée :

- d'une description de l'architecture technique du projet,
- d'une description des modèles de données utilisés,

- d'une description des technologies utilisées,
- d'une description de la stratégie de tests (taux de couverture du code, répartition tests unitaires/tests intégration) et de la stratégie du maintien de la qualité du code avec PHPStan,
- d'une description de points spécifiques techniques particuliers dépendants du projet (par exemple la sécurité sur un projet web, la présentation d'un algorithme complexe, etc.),
- Et de tous les éléments demandés par l'enseignant tuteur.



La documentation technique sera évaluée sur sa capacité à permettre à un nouvel arrivant sur le projet de comprendre et de s'approprier la solution sur le plan technique.

DOC004: Le mode d'emploi pour la configuration de VirtualBox (si besoin) et l'installation des machines virtuelles.

## 4.3. Livrables projet

### 4.3.1. Adaptation du plan de gestion de projet

PGP001 : Un document décrivant l'adaptation de la gestion de projet (Décrire la méthode, les cadences des réunions, durées des sprint, qui participe, etc..) doit être rédigé en début de période, il peut évoluer au cours du projet.

### 4.3.2. La fiche synthèse du projet

Le fichier README.md à la racine du projet sur Github présente une fiche synthétique du projet.

REA001 : le fichier README à la racine du projet Github doit contenir :

- le nom du projet ;
- la description du projet en 10 ligne maximum ;
- les membres de l'équipe ainsi que la répartition des rôles.
- Le lien vers le board Zenhub
- Le lien vers le dossier Google Drive contenant les documents relatifs au projet.

### 4.3.3. Le board agile

- Le board agile géré via Zenhub.

### 4.3.4. Suivi des temps individuels

ST001 : Un document de suivi des temps individuels (un seul fichier de type sheet avec une feuille par étudiant utilisant le même modèle). Contenu :

- date / tâche / travail réalisé / Catégorie / estimation % de réalisation / temps

passé).

- Temps totalisé par étudiant, par sprint et global sur le semestre (S4) avec graphiques.

#### 4.3.5. Les comptes-rendus de réunions

CR001: Chaque réunion fait l'objet d'un compte-rendu à fournir (nommer les documents avec leur type + date).

CR002: Les comptes-rendus des revues de sprint doivent obligatoirement mentionner les items de backlog validés par le client.

CR003: Les comptes-rendus des revues de sprint doivent obligatoirement être signés par le client.

CR004: Les comptes-rendus des rétrospectives doivent obligatoirement mentionner :

- les événements marquants du sprint.
- les actions retenues pour améliorer les prochains sprints.

#### 4.3.6. Le document de réception client signé

L'approche agile permet une validation de la construction du projet à l'issue de chaque sprint. Néanmoins, une livraison majeure doit faire l'objet d'une réception officielle par le client.

REC001: Le document de réception doit contenir :

- le nom du projet ;
- la date
- la description du projet en 10 lignes maximum ;
- les membres de l'équipe ainsi que la répartition des rôles.
- Le lien vers le projet Github
- Le lien vers le board Zenhub
- Le lien vers le dossier Google Drive contenant les documents relatifs au projet.
- L'intitulé suivant complété par le client : « Je soussigné ..... représentant le client du projet..... atteste réceptionner ce jour ledit projet. »
- Un espace de commentaires pour le client
- Le nom, date et signature du client

#### 4.3.7. Bilans collectifs et individuels

BIL001: À l'issue de chaque livraison majeure un bilan collectif doit être rédigé par l'équipe..

BIL002: À l'issue de chaque livraison majeure un bilan individuel doit être rédigé par chaque membre de l'équipe.