https://www.intechinfo.fr/wp-content/uploads/2015/11/logo-IN-TECH-ecole-informatique.jpg

Projet « Galt »

Version

Historique des révisions

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Version** | **Description** | **Auteur** |
| 14/10/16 | 1.0 | Remplissage de toute les parties | Léo |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Sommaire

[Rôles et responsabilités 4](#_Toc415564780)

[Parties prenantes, rôles et coordonnées 4](#_Toc415564781)

[Organigramme de synthèse 4](#_Toc415564782)

[Modalités de déroulement du projet 5](#_Toc415564783)

[Gestion du board agile 5](#_Toc415564784)

[Evaluation des charges et calendrier cible 5](#_Toc415564785)

[Planification initiale 5](#_Toc415564786)

[Gestion du reporting 6](#_Toc415564787)

[Gestion des relations avec les parties prenantes 7](#_Toc415564788)

[Gestion de la documentation 8](#_Toc415564789)

[Description des livrables 9](#_Toc415564790)

[Règles de validation 9](#_Toc415564791)

# Rôles et responsabilités

## Parties prenantes, rôles et coordonnées

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Nom | Société | Coordonnées | Catégorie | Rôle |
| Thibaud Duval | IN’TECH S4 | tduval@intechinfo.fr | MOE | Chef de projet |
| Antonin Charrier | IN’TECH S4 | acharrier@intechinfo.fr | MOE | Développeur |
| Guillaume Danguin | IN’TECH S4 | danguin@intechinfo.fr | MOE | Développeur |
| Léo  Stéfani | IN’TECH S4 | stefani@intechinfo.fr | MOE | Développeur |
| Antoine Raquillet | IN’TECH | antoine.raquillet@invenietis.com | Enseignent suiveur | Aide à la conduite du projet  Conseils techniques |
| Olivier Spinelli | IN’TECH | olivier.spinelli@invenietis.com | Enseignent suiveur | Aide à la conduite du projet  Conseils techniques |

## Organigramme de synthèse

# Modalités de déroulement du projet

## Gestion du board agile

Le board Trello est à cette adresse : https://trello.com/b/G617Uamk/projet-galt

Le chef de projet ajoute les cartes, y attribue des gens. Chaque personne peut ajouter des checklists à ses cartes et doit modifier l'emplacement selon l'avancement de la tâche qu'il à faire. Le chef de projet s’occupera de le faire évoluer au fur et à mesure des tâches accomplies.

Le board sera constitué d’une multitude de listes comme les tâches à effectuer, la progression des tâches et les tâches accomplies.

Le board possèdera 7 listes de carte.

* A faire au niveau macro : Objectifs d’itération
* A faire au niveau micro : Tâche des jours/semaines à venir
* En cours 0-50% : Tâches en cours d’exécution
* En cours 51-100% : Tâches en fin d’exécution ou à vérifier
* Terminé (et testé) : La tâche a été fini et vérifié/testé
* Documents externes : Regroupe les liens vers des documents externes concernant le projet
* Archives itérations précédentes : Déplacement des itérations effectuées avec leurs tâches dans cette section

## Evaluation des charges et calendrier cible

Le projet comportera 4 itérations :

L’itération 0 est réservée à la création et à la rédaction des documents avant-projets. Elle se finira le 24 octobre et durera 14 jours soit 21 heures (7 séances de 3 heures de PI)

L’itération 1 se terminera le 21 novembre et durera 28 jours soit 57 heures (19 séances de 3 heures de PI).

L’itération 2 se terminera le 20 décembre et durera 29 jours soit 66 heures (22 séances de 3 heures de PI).

L’itération 3 se terminera le 24 janvier et durera 34 jours soit 75 heures (25 séances de 3 heures de PI).

Le projet sera réalisé en 198 heures soit 33 jours (6 heures par jour) et réparti entre les 4 membres du projet ce qui équivaut à 132 jours/homme.

## Planification initiale

Itération 1 :

* Création de la chaine de build qui sera mise à jour à chaque modification
* Mise en place du serveur
* Création de la base de donné

Itération 2 :

* Commencement de la gestion des graphes de dépendance

Itération 3 :

* Finition
* Ajout d’éventuelles fonctionalitées

## Gestion du reporting

L’équipe rendra compte aux suiveurs à travers des réunions lors des différentes itérations pour présenter sous forme de PowerPoint, les objectifs accomplis et leurs fonctionnements, les objectifs restants et l’évolution du planning.

Une réunion sera organisée toutes les semaines à partir du lancement du projet pour rendre compte des possibles difficultés rencontrées et informer de l’avancée du projet par rapport au planning initial.

Le reporting des membres de l’équipe au chef d’équipe se fera de vive voix ou par Skype à chaque accomplissement de tâche.

## **Gestion des relations avec les parties prenantes**

Au sein de l’équipe, la communication se fera à l’oral et par mail. Nous effectuerons des réunions hebdomadaires au sein de l’équipe pour veiller au bon déroulement du projet.

La communication avec les enseignants suiveurs se fera grâce à des réunions hebdomadaires dans lesquelles nous parlerons de l’avancée du projet et une réunion à la fin de chaque itération.

Le chef de projet préparera les réunions qui seront animé par tous les membres de l’équipe. Les comptes rendus et suivis des actions seront effectués par roulement.

## **Gestion de la documentation**

**Règles de nommage :**

* Les comptes rendus de réunions seront nommés comme suit : AAAAMMJJ – Compte Rendu de Réunion
* Tous les caractères sont en minuscules
* Les espaces sont remplacés par des « \_ »
* Pour nommer les fichiers avec une date, écrire sous la forme AAAA.MM.JJ

**Règles de codage :**

* Commentaires et noms en anglais
* Ecrire les champs en camelCasing précédés d’un « \_ » (exemple : \_healPoint)
* Ecrire en PascalCasing les :
  + Namespaces
  + Propriétés
  + Méthodes
  + Événements
* Les noms des méthodes commenceront par un verbe

Le stockage des fichiers se fera grâce à Github à l’adresse :

<https://github.com/Daubiht/Project-Galt.git>

## Description des livrables

A la fin de chaque itération, l’équipe devra être capable de présenter le programme correspondant aux objectifs de l’itération, un diaporama de présentation de l’avancement du projet et des prochains objectifs à atteindre.

## Règles de validation

Si le livrable est un document, la validation se fera après la correction d’au moins un membre et la validation de l’équipe.

Lorsqu’il s’agit d’un programme, des tests unitaires devront être effectués et validés avant de transmettre chaque livrable.

Le projet devra garder une couverture de code de 80% minimum pendant toute la durée du programme.