

Rapport de projet

Whatsup

Présentation

Ce document traite du projet “Whatsup” qui est une salle de discussions (ChatRoom) programmé en Java. Ce projet a été réalisé dans le cadre du cours de génie logiciel. Voici le lien du projet sur GitHub : https://github.com/ork14750/Projet_GL_Chat_Room . Les instructions pour l’utilisation de l’application peuvent se trouve dans la fiche technique (en pièce jointe) ou sur le « read me » dans le GitHub.

1. L’organisation du projet

1.1. Organisation générale

L’équipe est composée de 5 personnes. Nous avons réparti les rôles en fonction des compétences de chacun dans le but d’optimiser le temps de développement du projet. Nous nous sommes appuyés aussi sur une structure type d’une équipe dans le monde de l’entreprise qui travaille sur un projet. Nous avons donc nommé un chef de projet, un manager, un chargé d’affaire et des développeurs :

- D'ALMEIDA Dédé Agripine : Manager
- MWAMBA Nsaka Jean-Jules : Chef de projet
- KOUADIO Olivier Romaric : Chargé d'affaires et assistant chef de projet
- YEHOUENOU Jean-Baptiste : développeur
- KEDE Christophe : développeur

Agripine est chargé de rédiger le cahier des charges ainsi que la démarche choisie pour la réalisation de celui-ci, les « user stories » et « constraint story » et une partie de la fiche technique de l’application.

Jean-Jules doit fournir un diagramme de PERT et le diagramme UML des classes et participer à la réalisation de la fiche technique. Il est également chargé de l’organisation des sprints, des planifications et des comptes rendus des réunions.

Olivier doit réaliser un diagramme de Gantt à l’aide du diagramme de PERT. Il doit également rédiger les autres parties du rapport de projet ainsi qu’une partie de la fiche technique de Whatsup.

Christophe et Jean-Baptiste doivent se répartir les tâches concernant le développement de l’application. Christophe doit également participer à la réalisation du fichier technique de l’application ainsi que le diagramme UML des classes. Quant à Jean-Baptiste, il doit fournir également le diagramme de cas d’utilisation.

1.2 Méthode de travail

La méthode de travail que nous avons choisi d'utiliser est la méthode agile car c'est une manière de travailler que nous avons l'habitude d'utiliser au sein de nos entreprises respectives. Nous avons décidé de faire des sprints tous les deux jours. Tous les deux jours donc nous devons nous retrouver sur microsoft teams afin d'évoquer l'avancement du projet.

2. La spécification

2.1 La démarche

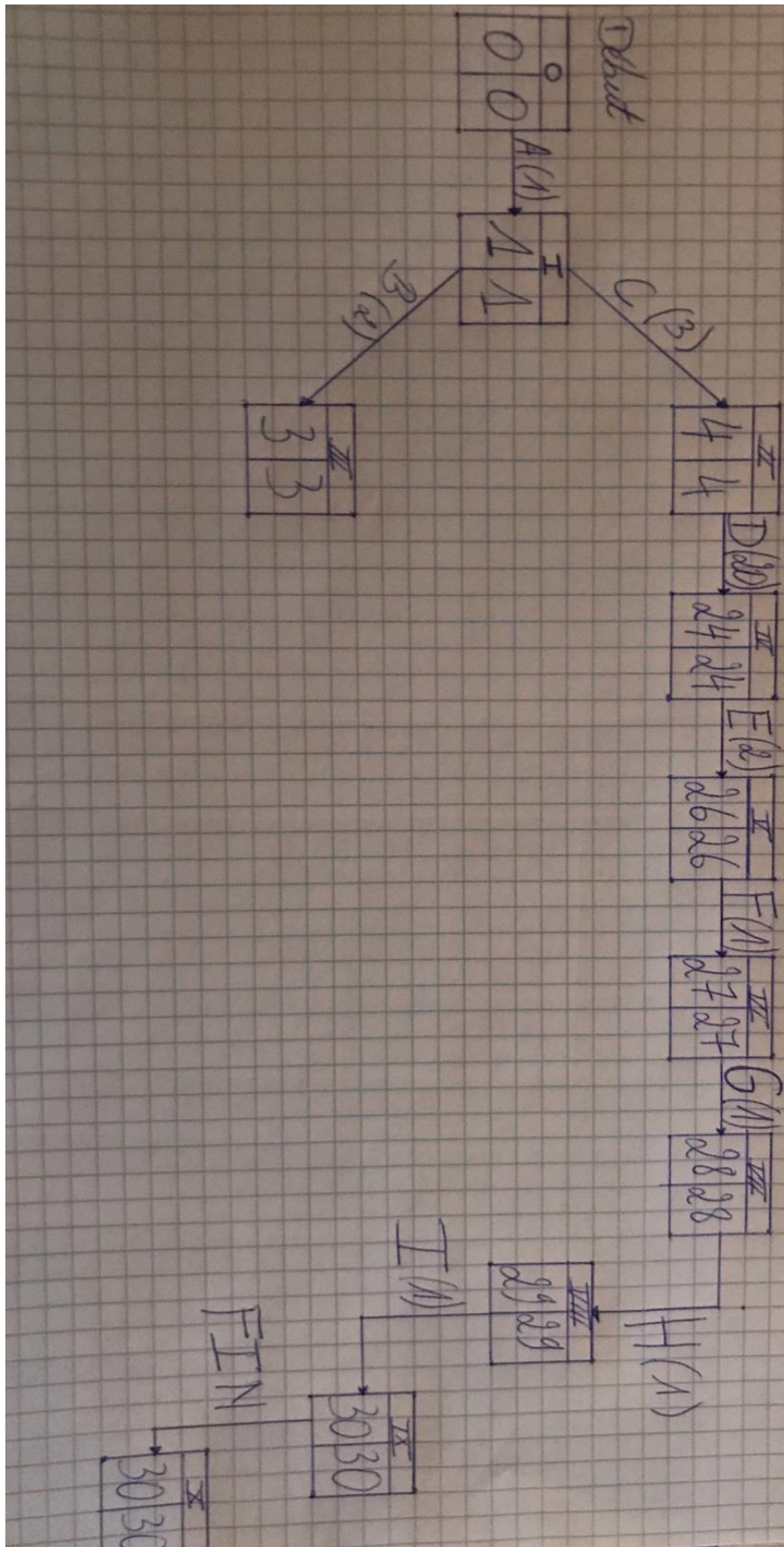
La démarche que nous avons suivi pour produire les exigences et fonctionnalités de notre projet est basée dans un premier temps sur les fonctionnalités qui nous avaient été demandées dans le sujet et dans un deuxième temps sur ce que nous avons pu ajouter à notre application. Nous n'étions pas sur compte tenu du temps de pouvoir ajouter certaines fonctionnalités à notre application donc lors de nos premières réunions nous nous étions convenu sur le fait de faire en priorité les fonctionnalités exigées dans le sujet et en tenant compte du temps qu'il nous restait nous en rajouterons.

Cette liste a donc été faite en collaboration avec nos développeurs en fonction des différentes fonctionnalités de notre application et du temps. (Voir Cahier des charges en pièce jointe).

2.2 Diagramme de PERT

Code	Libellé	Prédécesseurs	Durée
A	Réunion préalable et détermination des tâches respectives	/	1
B	Analyse (Exigences, Définition, types, sources, classification, modélisation, user story,...)	A	2
C	Conception (patrons de conception-Design patterns)	A	3
D	Développement	B	20
E	Intégration	D	2
F	Test (Dire le type de test, vérification et validation)	E	1
G	Build	F	1
H	Beta Test	G	1
I	Livraison/rendu	H	1

PERT



2.3 Diagramme de Gantt

[illegible]

3. USER STORIES & CONSTRAINT STORIES

US1.1)

En tant que User je veux me connecter avec un login et mot de passe afin d'avoir une sécurité sur mon compte

US 1.2)

En tant que User je veux que mes données soient sauvegardées jusqu'à ce que je les supprime afin d'avoir un controle sur la gestion de mes données.

US 1.3)

En tant que User je veux pouvoir créer un groupe, ajouter des collaborateurs à ce groupe, les retirer si besoin et envoyer des messages au sein de ce groupe afin de favoriser une interaction générale dans l'équipe

CS 1.1 (Performance)

Le système est rapide (afficher une page en quelques secondes, envoi et réception de messages fiables et simples) du au fait qu'il est en réseau local

CS 1.2 (Portabilité)

Le système ne fera pas usage de technologies qui rendront difficile le portage Linux ou windows.

CS 1.3 (Capacité)/ Priorité: Must

Le système peut supporter jusqu'à 25 connexions

CS 1.4 (Interoperabilité)

Le système sera écrit en Java

Diffusion non restreinte

CS 1.5 (Interoperabilité)

Toutes les données de configuration seront stockées dans des fichiers XML

Diagramme de cas d'utilisation :

