

Table des matières

1	Cadre du projet	2
1.1	Cadre général du projet :	3
1.2	Etude de l'existant :	3
1.2.1	Problématique	4
1.2.2	Solution	4
1.3	Méthodologie adoptée (SCRUM) :	4
1.3.1	Planification de travail :	4
1.3.2	Pilotage du projet avec SCRUM :	4
1.4	Les rôles dans SCRUMM :	5
1.5	Conclusion :	6
2	Aanalyse et planification	7
2.1	Specification des besoins :	8
2.1.1	Identification des acteurs :	8
2.1.1.1	Les étudiants	8
2.1.1.2	Les formateurs	8
2.1.1.3	L'administration de TEK-UP	8
2.1.2	Les besoins fonctionnels	8
2.1.3	Les besoins non fonctionnels	9
2.2	Conception	10
2.2.1	Diagramme de cas d'utilisation	10
2.2.2	Digramme de classe	10
2.3	Description de la méthode SCRUM	11
2.3.1	Répartition des Releases :	11
2.3.2	Product Backlog :	12
2.4	Environnement de travail	12
2.4.1	Outils de gestion de projet	12
2.4.2	Framework de développement	12
2.4.3	Système de gestion de base de données :	14
2.5	Conclusion :	14

Table des figures

1.1	Cycle de vie de la méthodologie SCRUMM	5
2.1	Diagramme de cas d'utilisation	10
2.2	Diagramme de classe	11
2.3	Logo GitHub	12
2.4	Logo de Trello	12
2.5	Logo Angular.	13
2.6	Logo Node Js.	13
2.7	Logo MongoDB.	14

Liste des tableaux

2.1 Répartition des sprints 11

Introduction générale

CADRE DU PROJET

Plan

1	Specification des besoins :	8
2	Conception	10
3	Description de la méthode SCRUM	11
4	Environnement de travail	12
5	Conclusion :	14

Introduction

Avant la réalisation de chaque projet, l'étude de l'existant s'avère indispensable. Dans ce chapitre, nous commençons au premier lieu de mettre le projet dans son cadre générale, on critique l'existant et on donne la solution puis nous allons préciser les études fonctionnelles de notre projet ainsi la méthodologie adaptée.

1.1 Cadre général du projet :

Le présent cahier des charges vise à définir les exigences et les spécifications pour la conception et l'implémentation d'une application Web mobile de gestion des Summer Trainings à TEK-UP. L'objectif principal de cette application est de faciliter le processus de gestion des formations d'été pour les étudiants et le personnel de TEK-UP en automatisant diverses tâches et en fournissant une plateforme conviviale pour accéder aux informations.

1.2 Etude de l'existant :

Actuellement, la gestion des Summer Trainings à TEK-UP se fait principalement de manière manuelle. Les étudiants intéressés doivent se rendre physiquement au bureau des Summer Trainings pour s'inscrire aux formations disponibles. Le personnel administratif enregistre ensuite les inscriptions dans des fichiers Excel ou des registres papier et parfois les étudiants eux memes doivent ecrire leurs noms et leurs emails manuellement chaque séance . Ces approches présente plusieurs limites :

***Inefficacité administrative :** Le processus manuel nécessite beaucoup de temps et d'efforts pour gérer les inscriptions, les annulations et les listes d'attente, ce qui entraîne souvent des erreurs et une perte de productivité pour le personnel administratif.

***Difficulté d'accès à l'information :** Les étudiants ne peuvent pas facilement accéder aux détails des formations disponibles, aux dates, aux formateurs, etc., car ces informations ne sont pas accessibles en ligne, et qui sont publié sur les groupes reseaux sociaux et il ya certains etudiants qui n'utilisent pas ces plateforme donc dans cette cas l'information est totalement absente .

***Communication inefficace :** Le manque de communication centralisée entre les étudiants, les formateurs et le personnel administratif peut entraîner des retards dans la transmission d'informations importantes.

1.2.1 Problématique

Le processus actuel de gestion des Summer Trainings à TEK-UP est limité par son caractère manuel, inefficace et peu convivial. Il est nécessaire de mettre en place une solution automatisée et conviviale pour améliorer la gestion des formations et offrir une expérience plus satisfaisante aux étudiants et aux formateurs.

1.2.2 Solution

Nous proposons la conception et l'implémentation d'une application Web mobile de gestion des Summer Trainings à TEK-UP. Cette application permettra aux étudiants de consulter facilement les formations disponibles, de s'inscrire en eux . Elle facilitera également la communication entre les étudiants, les formateurs et le personnel administratif.

1.3 Méthodologie adoptée (SCRUM) :

1.3.1 Planification de travail :

Dans notre projet, nous allons suivre la méthode SCRUM. Dans cette partie, nous identifions l'équipe SCRUM, puis nous représentons le backlog du produit ainsi que la planification des sprints tout au long de la période de réalisation.

1.3.2 Pilotage du projet avec SCRUM :

Pour pouvoir continuer les différentes phases de ce projet, nous sommes amenés, en premier lieu, à les planifier. Ayant choisi la méthodologie SCRUM, nous procédons dans ce qui suit à présenter cette planification en utilisant les principes de cette méthodologie.

Le choix de la méthodologie SCRUM qui fait partie de la méthodologie Agile a été basé sur :

- La flexibilité.

- Renforcement de la collaboration au sein des équipes.

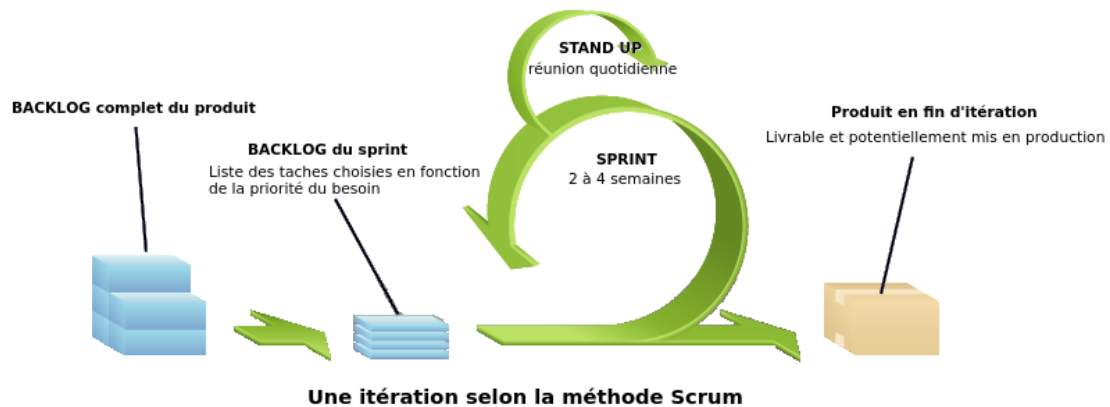


FIGURE 1.1 : Cycle de vie de la méthodologie SCRUMM

- L'adaptation.

1.4 Les rôles dans SCRUMM :

SCRUM consiste à la définition des rôles, artefacts et réunions. Il définit trois rôles qui sont :

- **Product Owner** : Le rôle du Product Owner consiste à chercher à comprendre les attentes des utilisateurs, qu'ils soient exprimés ou non par ce dernier. Il va chercher à creuser la réflexion de celui-ci pour connaître le ou les véritables problèmes auxquels il doit faire face.
- **SCRUM Master** : Le rôle du SCRUM Master ou le « servant-leader » - nomination tirée du guide SCRUM - est crucial dans une équipe SCRUM. Il est le garant de la mise en œuvre du cadre SCRUM, mais il est aussi et surtout coach/guide/formateur/facilitateur de l'équipe
- **SCRUM Team (équipe de développement)** : Le dernier des rôles SCRUM est l'équipe de développement SCRUM. L'équipe SCRUM contient entre 2 et 5 développeurs. Elle doit répondre à tous les besoins techniques nécessaires pour livrer le produit ou le service. L'équipe de développement est guidée par le SCRUM Master, mais doit être autonome. Chaque développeur doit être polyvalent et suffisamment responsable pour effectuer toutes les tâches requises.

1.5 Conclusion :

]Nous avons précédemment présenté le projet dans son cadre général. Le prochain chapitre consiste à décrire les besoins fonctionnels et non fonctionnels de ce projet ainsi que les outils utilisés pour le développement de cette application.

ANALYSE ET PLANIFICATION

Introduction

Cette partie est dédiée à la présentation de l'ensemble des fonctionnalités offertes par notre application web en précisant les acteurs de notre système, les diagrammes de classe et de cas d'utilisation et la planification du back log produit global. Nous allons aussi présenter les outils utilisés pour le déploiement de ce travail.

2.1 Specification des besoins :

2.1.1 Identification des acteurs :

2.1.1.1 Les étudiants

Utilisateurs de l'application qui s'inscrivent et participent aux formations

2.1.1.2 Les formateurs

Utilisateurs de l'application qui proposent et dispensent les formations.

2.1.1.3 L'administration de TEK-UP

L'administrateur une fois authentifié dans le système a le privilège de gestion les utilisateurs et gestion des formations

2.1.2 Les besoins fonctionnels

- Administrateur :

- * Gestion des utilisateurs (étudiants, formateurs, personnel administratif).

- * Gestion des formations (ajout, modification, suppression).

- * Suivi des statistiques (nombre d'inscriptions, taux de remplissage des formations, etc.).

- Etudiants :

- * Création d'un compte étudiant avec des informations personnelles (nom, prénom, numéro étudiant, etc.)

- * Consultation des formations d'été disponibles.

- * Inscription à une formation avec possibilité d'annulation si nécessaire.

- * Messagerie pour communiquer avec les formateurs et le personnel de TEK-UP.

- * Passer le quiz .

- Formateurs :

- * S'authentifier.

- * Création d'un compte formateur avec des informations personnelles et professionnelles.

- * Gestion des formations proposées par le formateur (ajout, modification, suppression)

- * Communication avec les étudiants inscrits.

- * Mettre les quiz

2.1.3 Les besoins non fonctionnels

Une fois les exigences fondamentaux sont bien définies, nous décrivons les besoins non fonctionnels qui doivent être pris en compte tout au long du processus de développement de l'application et qui doivent répondre aux critères suivants :

- * Sécurité des données des utilisateurs.

- * Performance de l'application pour un accès rapide aux informations.

- * Interface utilisateur conviviale et intuitive

2.2 Conception

2.2.1 Diagramme de cas d'utilisation

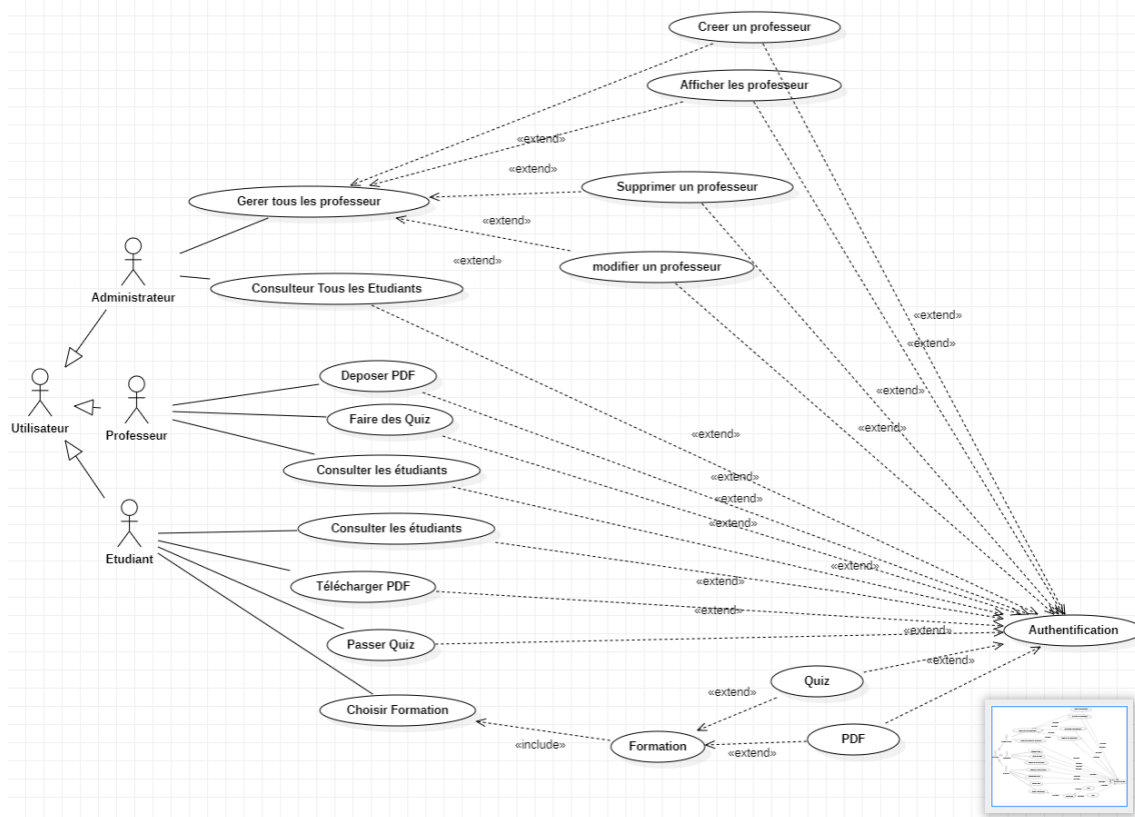


FIGURE 2.1 : Diagramme de cas d'utilisation

2.2.2 Diagramme de classe

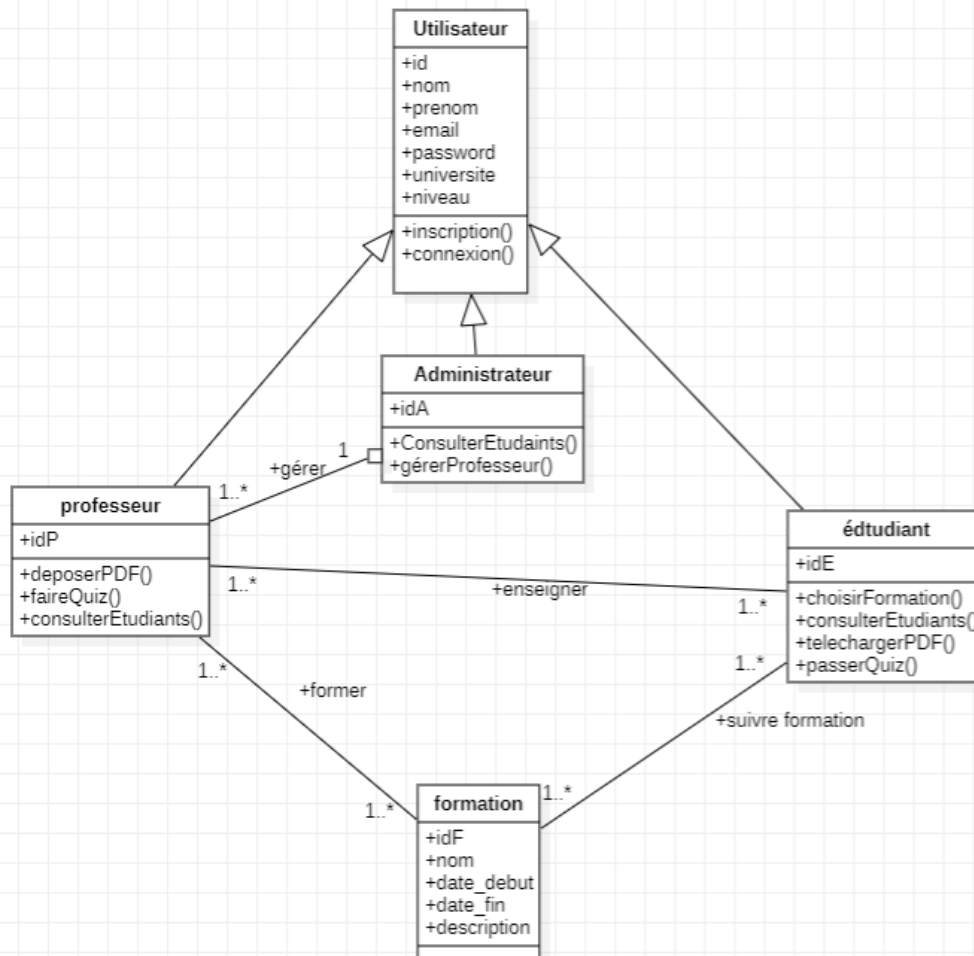


FIGURE 2.2 : Diagramme de classe

2.3 Description de la méthode SCRUM

2.3.1 Répartition des Releases :

Release ID	Nom du Sprint	Estimation
1	Sprint1 :Création et gestiondes comptes.	2-3 Semaines
	Sprint2 :Authentification.	2 Semaines
2	Sprint3 :Gestion des Formations.	2-3 Semaines
	Sprint4 :Gestion des Quiz	2-4 Semaines

TABLEAU 2.1 : Répartition des sprints

Le tableau 1 représente la répartition des différents sprints.

2.3.2 Product Backlog :

2.4 Environnement de travail

2.4.1 Outils de gestion de projet

- GitHub



FIGURE 2.3 : Logo GitHub

GitHub est un service d'hébergement de référentiel open source basé sur le cloud qui permet aux développeurs de stocker, gérer, suivre et contrôler toutes les modifications apportées à leur code.

- Trello



FIGURE 2.4 : Logo de Trello

Trello est une application de gestion de projet qui permet d'organiser un projet sous forme de tableaux, eux-mêmes composés de listes en colonnes, qui répertorient des tâches sous formes de cartes. Inspiré de la méthode agile Kanban.

2.4.2 Framework de développement

Au cours de cette section, nous allons préciser les outils utilisés pour le développement de cette application.

- Angular



FIGURE 2.5 : Logo Angular.

Angular est un framework pour clients, open source, basé sur TypeScript et codirigé par l'équipe du projet « Angular » chez Google ainsi que par une communauté de particuliers et de sociétés. Angular est une réécriture complète d'AngularJS, cadriciel construit par la même équipe.

- NodeJs



FIGURE 2.6 : Logo Node Js.

Node.js est une plateforme logicielle libre en JavaScript, orientée vers les applications réseau évènementielles hautement concurrentes qui doivent pouvoir monter en charge.

2.4.3 Système de gestion de base de données :

- MongoDB



FIGURE 2.7 : Logo MongoDB.

MongoDB est un système de gestion de base de données orienté documents, répartitionnable sur un nombre quelconque d'ordinateurs et ne nécessitant pas de schéma prédéfini des données. Il est écrit en C++.

2.5 Conclusion :

Ce chapitre a été dédié à la présentation des besoins caractérisant cette solution, la mise en place du planning à suivre et les logiciels utilisés pour la réalisation du projet. Le chapitre suivant consiste à décrire le backlog produit pour le premier et le deuxième sprint.