

MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE

UNIVERSITE DE SOUSSE

Institut Supérieur des Sciences Appliquées et de Technologie de Sousse



RAPPORT DE STAGE DE FIN D'ETUDE

**Conception et développement d'une plateforme pour les méthodes
d'analyse de réseaux sociaux**

LIEU DU STAGE : Laboratoire de recherche en informatique MARS



ELABORE PAR :

BOUSRIH Iness

ENCADRER PAR :

Mme KAROUI Wafa

Mme MAHMOUD Rihem

Année Universitaire 2021/2022

Remerciements

Je remercie ...

Je suis reconnaissant ...

J'exprime ma gratitude ...

Table des matières

1	Présentation générale du projet	2
1.1	Présentation de l'organisme d'accueil	2
1.2	Présentation du projet	2
1.2.1	Cadre général du projet	2
1.2.2	Objectifs	3
1.3	Étude de l'existant	3
1.4	Critique de l'existant	3
1.5	Solution proposée	4
1.6	Analyse et spécifications des besoins	4
1.6.1	Identification des acteurs	4
1.6.2	Identification des besoins	4
1.7	Architecture générale de l'application	6
1.7.1	Architecture physique	6
	Bibliographie	9
	Bibliographie	10

Table des figures

1.1	Modèle MVC	7
-----	----------------------	---

Liste des tableaux

Liste des acronymes

Introduction Générale

L'évolution de l'informatique et des technologies de l'internet dans le monde des universités engendre de profonds changements dans l'organisation du travail. La vitesse avec laquelle nous créons et partageons les algorithmes a considérablement augmenté.

1

Présentation générale du projet

Introduction

L'étude préliminaire est la première phase de tout projet ; Ainsi, ce chapitre va servir dans un premier temps à la présentation de l'organisme d'accueil SWMA. Nous définissons dans un deuxième temps le sujet et l'étude de l'existant ainsi que les solutions proposées. La deuxième partie sera consacrée à l'analyse et spécification des besoins. Et finalement, le choix méthodologique et architectural.

1.1 Présentation de l'organisme d'accueil

SWMA

1.2 Présentation du projet

1.2.1 Cadre général du projet

Ce projet est intitulé, «Digital Academy System » , il est réalisé dans le cadre de projet de fin d'études. Ce projet a été proposé par la société **SWMA** et élaboré au sein de

cette entreprise pour l'obtention du diplôme national d'ingénieur en génie logiciel de l'école pluridisciplinaire internationale (EPI) pour l'année universitaire 2021/2022.

1.2.2 Objectifs

Notre application va offrir plusieurs bénéfices pour ces utilisateurs qui sont les suivants :

- Éviter les impressions papier.
- Permettre les parents d'inscrire leurs enfants en ligne sans avoir se déplacer à l'école.
- Des notifications en temp réel, de suivi des demandes(acceptation ou refus), suivi des élèves(les absences, evenements

1.3 Étude de l'existant

L'étude de l'existant est une phase importante pour bien comprendre le système actuel. Il a pour objectif d'étudier le système existant et de proposer les solutions adéquates et définir les objectif à atteindre.

MyScol : MyScol (anciennement EduScol) est une seule application web de gestion des inscriptions en ligne, de la vie scolaire pour les différents établissements scolaires et bien plus encore, MyScol met en relation les différents acteurs de l'établissement :

- Administration : Système de gestion de vie scolaire qui réduit le travail administratif.
- Enseignants : MyScol fournit une interface aux enseignants pour consulter les emplois du temps, gérer les absences.
- Parents-Élèves : Assure le suivi scolaire de leurs enfants tout en veillant au respect de la confidentialité de leurs données personnelles.

1.4 Critique de l'existant

La critique du système constitue une étape utile. Elle a pour but de porter un jugement objectif afin de proposer un système plus fiable que le système ancien. L'analyse de la description nous conduits à dégager quelques anomalies :

- La même application pour tous les établissements scolaires.
- Absence d'une application mobile destinée aux parents pour le suivi de leurs enfants au cas qu'il ne possède pas un ordinateur.

1.5 Solution proposée

Notre application à proposée a plusieurs fonctionnalités communes que l'application existante.

En effet, notre projet est constitué non seulement d'une application web qui sera destinée pour toutes les tâches administratives et pédagogiques mais aussi d'une application mobile dédiée aux parents-Élève pour pouvoir poursuivre leurs enfants facilement et être toujours à la une de toutes nouveautés concerne ses enfants via des notifictaions instantanées.

1.6 Analyse et spécifications des besoins

1.6.1 Identification des acteurs

Un acteur peut être un utilisateur humain, dispositifs matériels ou autre système qui interagit directement avec le système étudié.

Pour Réaliser notre projet, on a pu identifier les acteurs suivants :

- Administrateur : c'est un acteur superviseur.
- Enseignat :Un acteur particulier qui a un rôle pédagogique.
- Parent : Dès que son inscription est validée par l'administrateur, il peut accédé à son plateforme.

1.6.2 Identification des besoins

1.6.2.1 Besoins Fonctionnels

Les besoins fonctionnels c'est ce que l'utilisateur attend en matière de fonctionnalités. Pour notre système, il existe 4 acteurs : **Administrateur**

- Gérer les rôles et les permissions pour les utilisateurs.
- Gérer les inscriptions.

Agent scolarité

- Gérer les classes.
- Gérer les niveaux.
- l'affectation des élèves au classes.
- Gérer les emplois du temps.
- Gérer les bulletins.
- Gérer le menu.
- Gérer les notes.
- Envoyer les convocations.

Enseignant

- Gérer les travaux à faire.

Parent

- S'inscrire en ligne.
- Consulter le/les bulletin(s) de son/ses enfant(s).
- Consulter les travaux à faire.
- Consulter les convocations.
- Consulter le menu de semaine.
- Recevoir des notifications en temps réel pour tous changement brusque ou événement.
- Envoyer des réclamations.

1.6.2.2 Besoins non fonctionnels

Ces sont des exigences internes primordiales pour le système. Les principaux besoins non fonctionnels de notre application se résument dans les points suivants :

- Sécurité : Nous avons garantir un niveau de sécurité optimal pour les données : Les comptes des utilisateurs sont sécurisés par mot de passe(longueur, caractères spéciaux,...), vérifier l'existence d'utilisateur dans la base de données.
- Performance : L'application répond à toutes les exigences des internautes d'une manière optimale.
- Fiabilité : Un bon fonctionnement de l'application sans détection des erreurs.
- Efficacité : L'application doit permettre l'accomplissement de la tâche avec le minimum de manipulations.
- Convivialité : Les interfaces doivent être simples, claires, concises,conviviales et facile à manipuler.
- Portabilité : L'application est multiplateforme.

1.7 Architecture générale de l'application

1.7.1 Architecture physique

Afin de séparer l'affichage des informations, les actions des utilisateurs et l'accès aux données, nous avons choisi le patron de conception MVC. En effet, c'est une solutions très répandue pour la réalisation des applications web. Ainsi, l'application se retrouve segmentée en trois composants essentiels : **Modèle** : c'est la parties qui gère les données relatives aux différentes entités de l'application et qui représente les règles métiers. **Vue** : C'est une partie consacrée à la construction de l'affichage des données venant du modèle qui va être retourné à l'utilisateur. **Contrôleur** : C'est en quelque sorte un intermédiaire entre l'utilisateur, la vue et le modèle. Son rôle est l'interception des requêtes de l'utilisateur, la demande des données du modèle et la redirection vers la vue adéquate.

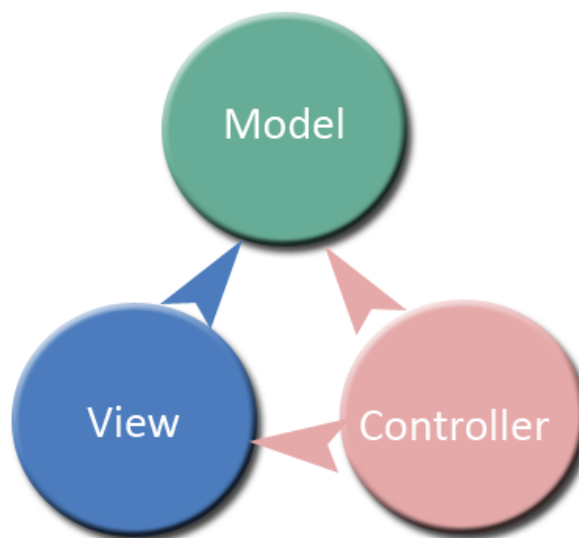


FIGURE 1.1: Modèle MVC

1.7.2 Fonctionnement de l'architecture

L'image ci-dessous résume les relations entre les composants d'une architecture MVC.

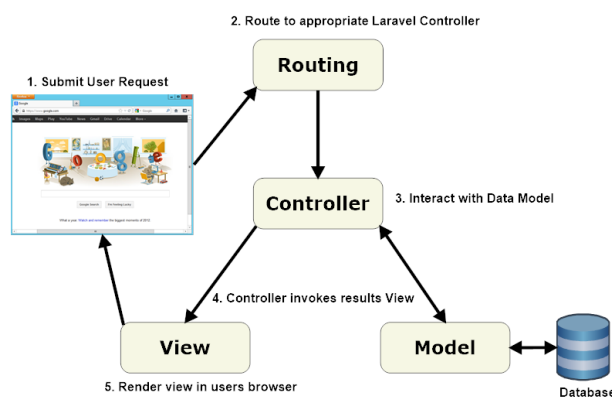


FIGURE 1.2: Architecture MVC

1.8 Processus de développement

La finalisation du projet dans les délais de livraison est le souci majeur de chaque équipe de développement d'un logiciel. L'un des problèmes les plus fréquemment affrontés lors de la

construction du logiciel est la mauvaise spécification et le changement brusque des besoins. Cela peut influencer non seulement l'équipe de développement en créant un environnement de stress, mais aussi le temps consacré pour la réalisation du projet et donc des délais de livraison dépassés. Afin d'éviter ces situations critiques, nous adoptons la méthodologie agile pour la gestion de notre projet. Les méthodes agiles [1] sont des méthodes de gestion et de développement de projets ou programmes informatiques. Elles visent à satisfaire les besoins du client au terme du contrat de développement. Elles fonctionnent sur la base de l'itératif et l'incrémental. De nos jours, il existe une variété de méthodes agiles dont les plus courantes sont :

- Scrum,
- Extrême Programming (XP),
- Rationnal Unified Process (RUP).

Dans notre cas, nous allons adopté la méthode agile **Scrum**.

1.8.1 Présentation de Scrum

SCRUM est une méthode agile dédiée à la gestion du projet. Cette méthode de gestion a pour objectif d'améliorer la productivité de son équipe. La méthode SCRUM implique que le projet progresse à travers la mise en place de séries de « sprints ». A chaque lancement d'un sprint, une réunion de planification est organisée afin que chaque membre de l'équipe puisse s'engager sur le nombre de tâches qu'il pourra exécuter, ainsi que sur la création du « sprint backlog », qui est la liste globale des tâches à réaliser lors du sprint. Chaque jour du sprint, tous les membres de l'équipe (ainsi que le responsable produit et le SCRUM Master) doivent assister à la réunion SCRUM quotidienne. Cette dernière ne doit pas durer plus de 15 minutes, et permet aux membres de l'équipe de partager aux autres ce qu'ils ont fait la veille, ce sur quoi ils travaillent le jour même, ainsi que l'identification de tout problème pouvant entraver le bon déroulement du sprint. Cette réunion permet ainsi de synchroniser tous les membres de l'équipe. La fin d'un sprint est marquée par une session de débriefing permettant de présenter le travail achevé au responsable produit, et de partager des informations pouvant influencer sur le sprint suivant.

Voilà un schéma qui représente le processus de la méthodologie SCRUM, avec un détail de chaque étape :

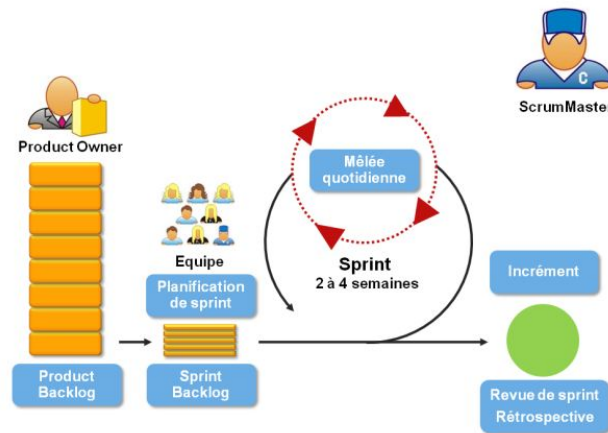


FIGURE 1.3: Cycle de vie du Scrum

a- Le Product Backlog : est le point central de tout projet Scrum. Il est sous la responsabilité unique de Product Owner, mais doit également être accessible par l'équipe. Le Product Backlog peut être défini comme étant l'ensemble des fonctionnalités du produit que l'on veut développer. Ces fonctionnalités sont connues sous la nomination "User Stories" et sont décrites en employant la terminologie utilisée par le client.

b- Sprint Backlog : c'est une partie des user stories du backlog produit que l'équipe s'engage à livrer d'ici la fin du sprint. Ainsi, le travail le plus important sera effectué en premier.

c- Mêlée quotidienne : c'est l'organisation d'une réunion d'avancement (environ 15 minutes) chaque jour avec tous les membres de l'équipe afin de s'assurer que les objectifs du sprint courant seront tenus.

d- Revue de sprint : c'est dans lequel on passe en revue ce qui a été réalisé pendant l'itération avec l'utilisateur et puis le passer à ce moment-là avec ce qui «est demandé par les utilisateurs, de ce qu'il va falloir encore ajouter au produit backlog, ou supprimer». C'est à ce moment-là qu'on fait la revue de ce que le sprint a délivré.

e- Sprint Rétrospective : c'est la dernière étape dans cette méthodologie, est là on passe en revue les éléments de la méthodologie Scrum dans l'entreprise «ce qui a fonctionné ou ce qui n'a pas fonctionné, en fait, c'est une revue de la méthodologie».

1.8.2 Les rôles du Scrum

Scrum définit seulement 3 rôles :

a- Le Product Owner : c'est un expert métier. Il définit les spécifications fonctionnelles. Il établit la priorité des fonctionnalités à développer ou corriger. Finalement il valide les fonctionnalités développées. Il joue le rôle du client.

b- Le Scrum Master : il s'assure que les principes et les valeurs de Scrum sont respectés. Il facilite la communication au sein de l'équipe et il cherche à améliorer la productivité et le savoir faire de son équipe.

c- L'équipe de Développement : pas de rôle bien déterminé : architecte, développeur, testeur. C'est celui qui réalise l'application.

1.9 Backlog

1.10 Chronologie

Diagramme de GANTT est un outil de planification des tâches nécessaires pour la réalisation d'un projet quel que soit le secteur d'activité. Il permet de visualiser l'avancement des tâches d'un projet de manière simple et concise. Le diagramme va représenter les tâches principales à réaliser dans notre projet (via le logiciel libre : Ganttproject).

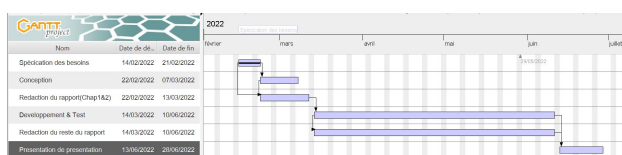


FIGURE 1.4: Diagramme de GANTT

Conclusion

Tout au long de ce chapitre, nous avons présenté l'entreprise d'accueil ainsi que notre projet. Par la suite, la construction de notre product backlog en se basant sur les besoins fonctionnels et non fonctionnels. Finalement, nous avons mentionné la chronologie qu'on

doit l'adapter tout au long la réalisation de ce projet. Dans le chapitre qui suit, Nous allons commencer par notre premier sprint « ».

Conclusion Générale

Paragraphe 1 : décrire brièvement le contexte d'application de votre logiciel dans le cadre de l'organisme d'accueil.

paragraphe 2 : résumer brièvement le travail réalisé.

paragraphe 3 : Donner quelques perspectives, qui sont réellement faisables, pour votre logiciel.

Exemples : une possible amélioration du logiciel, des fonctionnalités qui peuvent être ajoutées au logiciel, d'autres cadres possibles d'application de votre logiciel.

Bibliographie

- [1] https://ineumann.developpez.com/tutoriels/alm/agile_scrum/.
<http://www.idnext.net/la-methode-agile-quest-ce-que-cest/>.

Bibliographie

- [1] Andrea Goldsmith. *Wireless Communications*. Cambridge University Press, New York, NY, USA, 2005.
- [2] chaima chatti. *Réseau Mobile*. Hermes, Juin 2019.
- [3] UIT. Site web de l'uit. www.itu.org. (consulté le 13 novembre 2018).