

UNIVERSITÉ DU QUÉBEC À MONTRÉAL

Projet de session - TP1

Système de gestion projet et feuille de temps

Ammar Hamad

DANS LE CADRE DU COURS

INF6150

Génie logiciel: conduite de projets informatiques

Équipe : SummerCode

Groupe: 09

PAR

- Andy Del Risco Manzanares, DELA12037706
- Dennis Orozco Martinez, OROD15018506
- Filip Kulach, KULF12038704
- Vladimir Kvaratshelya, KVAV23027507

09 mai 2019

Table de matières	1
Sommaire exécutif	2
Volet administratif	2
Création, modification et suppression de projets	2
Création, modification et suppression d'employés	2
Assignment des tâches aux employés	3
Consultation des avancements des projets	3
Volet client	3
Acceptation de tâches	3
Avancement du projet	3
Consultation Client	3
Objectif	3
Terminologie	4
Description du besoin	4
Contexte	4
Problème	4
Objectif	4
Périmètre	4
Date souhaitée de mise en œuvre	4
Fonctions principales et secondaires	5
Analyse de la faisabilité et évaluation du coût du projet	5
Expérience de l'entreprise	5
Analyse de faisabilité technique	6
Analyse de faisabilité financière	6
Analyse de l'impact du projet	7
Scénarios de réalisation	7
Scénario # 1	7
1. Description	7
2. Perspective organisationnelle	8
3. Perspective technique	8
4. Perspective financière	8
5. Risques	8
Scénario # 2	8
1. Description	8
2. Perspective organisationnelle	8
3. Perspective technique	9
4. Perspective financière	9
5. Risques	9

Scénario # 3	9
1. Description	9
2. Perspective organisationnelle	10
3. Perspective technique	10
4. Perspective financière	10
5. Risques	10
Choix d'un scénario de réalisation	10

Sommaire exécutif

Le but de cette étude est d'évaluer la faisabilité du projet d'implantation d'une application web et mobile (DEV) permettant de suivre l'avancement de projets et de consulter les feuilles de temps des employés. Le système a un volet administratif et un volet client. Les actions possibles sont décrites ci-dessous.

Volet administratif

Création, modification et suppression de projets

Un projet est identifié par un code unique composé de deux lettres et trois chiffres, par exemple: AA123, il comporte un nom, un montant, une date de début et de fin, l'avancement, le nombre de personnes impliquées, une description ainsi que les tâches qui comportent le projet.

Création, modification et suppression d'employés

Un employé est identifié par son code unique, il s'agit d'une lettre et 4 chiffres, par exemple A1234, également un nom, son prénom, le taux horaire, le corps d'emploi auquel fait parti ainsi que la description de ses fonctions.

Assignation des tâches aux employés

Une tâche d'un projet est associée à un ou plusieurs employés, un employé ne peut appartenir à plus de trois projets et doit respecter la durée du projet.

Consultation des avancements des projets

Le système affiche tous les projets ainsi que leurs informations. Le système affiche aussi les feuilles de temps des travailleurs.

Volet client

Acceptation de tâches

Le travailleur accepte les tâches qui lui sont assignées, le système informe le gestionnaire lorsqu'une tâche est acceptée

Avancement du projet

L'employé enregistre le nombre d'heures travaillées par tâches dans les projets dans lesquels il est impliqué.

Consultation Client

Le système affiche les informations relatives aux tâches par projet en cours d'exécution. Le système affiche la feuille de temps des projets en cours d'exécution.

Objectif :

Ce document d'étude de faisabilité (DEF) vise à étudier la possibilité d'investir dans un nouveau système de gestion de projet et feuille de temps (GPF). Nous proposerons la possibilité de développer une solution spécifique à leurs besoins.

Puis nous ferons une recommandation de la solution qui nous semble la meilleure, en considérant les aspects technologique, économique et organisationnel.

Terminologie :

- DEV Développement de l'application web et mobile
- DEF Document d'étude de faisabilité

- GPF Gestion projet et feuille de temps
- POO Programmation orientée objet

Description du besoin :

Contexte:

L'entreprise souhaite avoir un meilleur contrôle des avancements des projets et de la productivité des employés

Problème :

Actuellement il n'y a aucun outil pour visualiser les avancements des projets

Objectif :

- Le système permettra la gestion de projets (GPF) et la saisie de données relatives aux tâches et les assignations aux employés.
- Le système permettra l'affichage de l'état d'avancements des projets.
- Le système permettra la saisie des heures travaillées par tâches à partir d'une interface mobile.

Périmètre:

L'application ne contient pas le module de paie ni le module comptable.

Date souhaitée de mise en œuvre :

30 mai 2019

Fonctions principales et secondaires :

- Gérer et visualiser les avancements des projets.
- Possibilité de rentrer les heures travailler à partir d'une interface web ou d'une l'interface mobile.

Analyse de la faisabilité et évaluation du coût du projet

Expérience de l'entreprise

L'expérience que l'on a sur ce projet provient entre autres de nos cours d'université. En effet, on a eu des cours sur divers langages de programmation comme Java, C et C++ qui nous montrent la base de la programmation sur la console(input/output, structures conditionnelles, boucles, POO) et un peu en interface graphique(Java Swing). De plus, on a eu des cours sur la manipulation de base de données(Oracle SQL) qui concerne entre autres la création de tables, l'insertion de données dans une table ainsi que la lecture et l'écriture sur les tables(bref, comment faire des requêtes SQL sur une base de données). Également, certains ont eu des cours sur la programmation web autant front-end que back-end avec des langages comme HTML, CSS, JavaScript et Python. On nous a montré d'ailleurs comment créer une page web avec des balises HTML ainsi que comment positionner les différents éléments par du code CSS ainsi que comment faire des gestions d'événements(ex: clic de bouton, apparition de message d'alerte) par du JavaScript. On a vu aussi la programmation back-end par Python(Flask) qui concerne la manipulation d'une base de données dans des applications web, comment gérer une connexion usager(ex: hacher le mot de passe pour le ne pas le stocker en clair) et aussi comment faire de l'échange de données entre applications par des API. Il faut aussi ajouter que dans le cours d'Agile on a vu comment utiliser un gestionnaire de sources. Également, certains d'entre nous ont acquis de l'expérience en informatique sur le marché du travail que ce soit par des stages ou par de l'emploi à temps partiel ainsi élargir plus nos connaissances sur ce qu'on a appris dans nos cours. Aussi, certains ont déjà été ou sont actuellement des utilisateurs de ce type d'application qui sera développé dans notre projet, car comme certains travaillent ou ont déjà travaillé pour une firme de consultants informatiques, saisir des feuilles de temps ainsi que l'avancement de projets est quelque chose de commun pour des consultants. Donc, pour certaines technologies qu'on utilisera comme les bases de données, C#, la programmation web(Front-end et back-end) et le gestionnaire de source on a déjà une certaine expérience surtout sur le marché du travail.

Analyse de faisabilité technique

Il sera possible de développer le projet selon les connaissances techniques de chacun dans l'équipe malgré que certains membres ne sont pas familiers avec certaines technologies. En effet, pour développer ce projet ça nous prend une base de données pour

contenir le temps que chaque employé passe dans tel projet et aussi pour contenir les projets et les employés eux-mêmes (supprimer, mettre à jour, insérer de nouvelles informations) et tout le monde dans l'équipe a déjà de l'expérience dans le langage de base de données SQL(Server ou Oracle) que ce soit à l'école ou sur le marché du travail donc on a pas besoin de consacrer beaucoup de temps à la formation. Cependant, pour la partie back-end en NodeJS une certaine partie de l'équipe n'a jamais travaillé avec cette technologie, ce n'est pas un gros frein quant à la réalisation du projet, mais au début, il faudrait consacrer un peu plus de temps quant à la formation via des tutoriels, crash courses ou en posant des questions aux autres membres de l'équipe qui sont plus connaisseurs sur NodeJS pour éviter un ralentissement dans le développement du projet. Également, pour la partie du codage de l'application mobile, certains ne connaissent pas non plus la technologie vu que ce n'est pas un cours qui est actuellement obligatoire dans le baccalauréat en informatique et génie logiciel alors pour ça aussi il faudrait consacrer beaucoup de temps pour les tutoriels, crash courses ou questions pour qu'on soit tous au même niveau.

Analyse de faisabilité financière

Pour bien réaliser ce projet il faut prendre aussi en compte que le coût financier n'est pas gratuit. En effet, tout d'abord c'est nos heures de travail que l'on met dans le projet qui va déterminer si le projet est rentable ou non. On peut voir ça de deux côtés, c'est sûr que si on ne met pas beaucoup d'heures dessus, le projet va être terminé à la date d'échéance et il n'y aura pas de frais supplémentaires par contre si on veut que le projet ait une bonne qualité, il faut mettre des beaucoup d'heures dessus. Donc, il faut avoir un nombre raisonnable pour les heures de travail(pas trop ni pas assez) pour que le projet soit de bonne qualité et livrer dans les temps. De plus, pour réaliser un projet informatique il faut payer l'équipement(juste un bon laptop de 500\$ par développeur c'est raisonnable) et ça prend des développeurs qui doivent être payés pour ce travail et il faut aussi payer les bureaux de travail donc il faut établir un budget pour prendre en compte tout ça. Ça peut être possible de réaliser le projet si on planifie bien le budget pour ça.

Analyse de l'impact du projet

L'impact que ce projet aura est la facilité pour gérer les projets. En effet, on peut savoir plus facilement comment avancent les projets de la firme et garder traces des assignations plus facilement lorsqu'on saisit le temps dans une application que de tout saisir sur du papier. En effet, ce n'est pas toujours facile à lire ce qui est sur du papier et on peut faire une erreur et aussi il faut fouiller manuellement dans les papiers lorsqu'on veut chercher quelque chose de spécifique alors que sur une application c'est plus lisible, structuré et même qu'on peut appliquer des filtres pour retrouver un détail précis qu'on cherche. Également, la possibilité d'avoir une application mobile rend la tâche plus simple encore, car on a plus besoin d'être devant un laptop ou un Desktop pour saisir notre temps, on peut le faire de n'importe où tant qu'on a un téléphone cellulaire dans les mains. Aussi, on peut savoir si le projet avance bien, c'est-à-dire qu'il respecte le temps et les échéances prévues quant au déploiement.

Scénarios de réalisation

Scénario # 1

1. Description

La méthode de développement itérative. Nous allons schématiser les fonctionnalités les plus importantes de notre projet. Le diagramme de composantes, le diagramme de cas d'utilisations basiques et le diagramme UML. Puis, nous allons diviser le projet en étapes. La première étape sera de créer toutes les composantes du système et les faire communiquer avec des simples appels. Après que la première étape sera fonctionnelle, nous allons ajouter les fonctionnalités aux modules.

2. Perspective organisationnelle

Nous allons faire les rencontres d'équipe hebdomadaires où chacun va présenter son travail fait pendant la semaine et parler de ce qu'il va faire la semaine prochaine. Aussi, chacun va parler de difficultés rencontrées lors du développement. Ensemble, nous allons discuter de l'avancement du projet et planifier les prochaines itérations.

3. Perspective technique

Le projet sera déposé sur Git et sera visible à toute l'équipe.

Chacun va travailler dans son domaine, par exemple quelqu'un va programmer la base de données, l'autre va programmer JavaScript, l'autre va faire l'application mobile, etc.

4. Perspective financière

Chaque membre de l'équipe devrait travailler sur le projet six heures par semaine.

5. Risques

- Compiqué de prévoir les difficultés que nous allons avoir lors de développement.
- Il nous serait difficile de faire les changements si on a commis une erreur lors de la conception.
- Risque de dépasser le deadline à cause de difficulté de faire communiquer les modules de système.

Scénario # 2

1. Description

La méthode de développement Itérative. Nous allons schématiser les fonctionnalités les plus importantes de notre projet. Le diagramme de composantes, le diagramme de cas d'utilisations basiques et le diagramme UML. Puis, nous allons diviser le projet en étapes. Le premier étape sera de créer tous les composantes du système et les faire communiquer avec des simples appels. Après que la première étape sera fonctionnelle, nous allons ajouter les fonctionnalités aux modules.

2. Perspective organisationnelle

Nous allons faire les rencontres d'équipe hebdomadaires où chacun va présenter son travail faite pendant la semaine et parler de ce qu'il va faire la semaine prochaine. Aussi, chacun va parler de difficultés rencontrés lors du développement. Ensemble, nous allons discuter de l'avancement du projet et planifier les prochaines itérations.

3. Perspective technique

Le projet sera déposé sur Git et sera visible à toute l'équipe.

Vu que notre projet sera divisé en plusieurs petites itérations, il sera plus facile aux membres de l'équipe d'être au courant de ce que les autres font. Nous allons faire la liste de tâches et les organiser selon leurs priorités. Les membres de l'équipe seront libres de choisir

les tâches qu'il aiment. Par exemple quelqu'un qui a fait SQL la semaine passée va pouvoir faire JavaScript la semaine d'après.

4. Perspective financière

Aucune surveillance sur le temps travaillé. Chacun doit être en mesure de terminer sa tâche de semaine.

5. Risques

- Il sera difficile de prévoir le temps nécessaire pour la réalisation de tâches.
- Vu que nous n'avons pas beaucoup d'expérience, parfois nos décisions vont nous amener dans les mauvaises directions.
- Il y a de risque que le produit final va être différent de ce que nous avons planifié.

Scénario # 3

1. Description

Le modèle de développement en Spirale.

Nous allons schématiser les fonctionnalités importantes de notre projet et nous allons diviser le projet en trois itérations. Chaque itération va durer deux semaines et va comprendre 4 éléments:

- a) Détermination des objectifs.
- b) Évaluation de risques.
- c) Programmation et débogage.
- d) Planification de prochaine itération.

2. Perspective organisationnelle

Toutes les deux semaines nous allons faire des rencontres d'équipe pour effectuer les quatre étapes énumérées plus haut et passer à la prochaine itération.

3. Perspective technique

Le projet sera déposé sur Git et sera visible à toute l'équipe.

Lors de rencontres d'équipe, les diagrammes de conceptions détaillées seront produits pour la prochaine itération. Chacun choisira ses tâches selon ce qu'il fait le mieux.

4. Perspective financière

Chaque membre de l'équipe devra obligatoirement terminer ses tâches de deux semaines.

5. Risques

- Risque de ne pas pouvoir terminer le projet à temps à cause que le développement risque de devenir trop lourd.
- Si une des itérations n'est pas réussie, on risque de perdre le temps.
- La gestion de projet risque de devenir difficile.
- Beaucoup de documentation.

Choix d'un scénario de réalisation

Nous allons choisir le scénario # 2, la méthode de développement itérative. Cette approche va nous donner plus de flexibilité. Grâce aux rencontres hebdomadaires, le niveau de collaboration de membres de l'équipe sera très élevé. Grâce aux simplicités des tâches et à la possibilité de choisir une tâche de différent domaine, tous les membres de l'équipe vont être au courant de tous les aspects de développement. Et, finalement, il nous sera facile d'effectuer les changements.