

Présente toutes les informations
nécessaires pour la bonne gestion
organisationnelle du projet.

Software Development Plan

GOLDEN SQUAD

2017

Fichier :		GOLDEN-SQUAD_SDP_R0.docx		
Validation				
Nom		Date	Validation (O/N)	Commentaires
GOLDEN SQUAD		02.10.2017		
Historique des modifications				
Version	Date	Etat	Description de la modification	Auteur
1.0.0	02/10/2017	Inachevé		Marc Mefoung

Table des matières

INTRODUCTION	4
VUE D'ENSEMBLE DU PROJET	4
BUT DU PROJET ET OBJECTIFS	4
EVOLUTION DU PLAN DE DÉVELOPPEMENT LOGICIEL	5
HYPOTHÈSES ET CONTRAINTES	5
ORGANISATION DU PROJET	5
ÉQUIPE DE DÉVELOPPEMENT	5
ROLES ET RESPONSABILITES	5
Tâche et responsabilité en R0	6
PRODUCT OWNER (PO)	6
GESTION DU PROJET	7
CHARGE DE TRAVAIL	7
PLAN DE PROJET	7
PLAN DE PHASE	7
DÉROULEMENT DES PHASES ET VALIDATION	7
RELEASE 0 ((Spécification, Conception))	7
RELEASE 1	8
PLANNING PREVISIONNEL	8
PLAN DES RELEASE	8
OBJECTIFS DES ITÉRATIONS	9
Itération 0 :	9
Itération 1 :	9
Itération 2 :	9
Itération 3 :	9
Itération 4 :	9
Itération 5 :	9
SUIVI DE PROJET ET CONTRÔLE	10
MOYEN DE COMMUNICATION	10
RÉUNIONS	10
CONTRÔLE DE LA QUALITÉ	11
ENVIRONNEMENT TECHNIQUE	12

1. INTRODUCTION

Ce document a pour but de présenter l'ensemble des informations nécessaires au contrôle du projet. C'est donc un plan de haut-niveau qui sera utilisé par les gestionnaires pour diriger le processus de développement.

Nous y décrirons l'approche de développement et de la mise en place du système d'information avec, d'une part l'identification des phases, leurs durées et les objectifs de leurs itérations, et d'autre part, les estimations des ressources nécessaires à son bon déroulement.

Ce document s'adresse donc à tous les intervenants du projet (c'est à dire le product owner et tous les Team members), ainsi qu'au client s'il le désire.

2. VUE D'ENSEMBLE DU PROJET

2.1. BUT DU PROJET ET OBJECTIFS

Dans l'optique d'approfondir la connaissance du business de l'entreprise, la BEM a chargé l'équipe GOLDEN-SQUAD de la mise en place d'un système décisionnel ainsi que de solutions explorant les possibilités de l'internet des objets.

Cette solution doit permettre aux **utilisateurs/clients** :

- De retirer de l'argent dans des GAB/DAB à partir de leurs téléphones portables
- Acheter/Payer avec son téléphone portable

Par ailleurs, les **Managers** de l'agence doivent pouvoir :

- Maîtriser la fréquentation réelle des agences (comptabiliser et analyser cette fréquentation)
- Définir des indicateurs de performance
- Simuler/Prédire l'évolution des données opérationnelles

Aussi, le **Système** devra être capable de :

- Gérer les biais(personnes qui passent exactement en même temps,etc...)
- Collecter puis explorer et analyser les données provenant des sources diverses (BigData)

Enfin, les **Commerçants** ayant souscrit à l'offre « **Payfree** » doivent pouvoir :

- Transmettre les demandes de paiement à la BEM
- Recevoir des éventuelles autorisations

2.2. EVOLUTION DU PLAN DE DÉVELOPPEMENT LOGICIEL

Ce plan de développement évoluera tout au long du projet. Il sera notamment mis à jour après chaque jalon de fin de phase (ou même d'itération).

2.3. HYPOTHÈSES ET CONTRAINTES

Le système complet développé devra être livré obligatoirement à la fin du mi-février. Une version stable de l'application devra être livrée impérativement à chaque COPIL. Tous les livrables doivent être remis à l'équipe pédagogique 5 jours avant le comité de pilotage de chaque Release.

L'équipe étant constituée essentiellement des étudiant en FI, nous faisons face aux empêchements de quelques membres d'équipe pour des contraintes de job, rendez-vous à la préfecture... Aussi, compte tenu du nombre de salle limitée au sein de notre composante, il nous est souvent difficile de nous réunir dans un cadre confortable. Par ailleurs, nous avons d'autres obligations par rapport aux autres unités d'enseignement (cours, révision, travaux à rendre) estimées en moyenne à 35h/semaine.

3. ORGANISATION DU PROJET

3.1. ÉQUIPE DE DÉVELOPPEMENT

L'équipe est composée de 6 étudiants de 3^{ème} année Ecole d'ingénieur en formation initiale (ING3-FI).

3.2. ROLES ET RESPONSABILITES

Personne	Rôles	Mail
DIAL Aïssata	Scrum Master	aissata-youssouf.dial@u.u-pec.fr
	<i>Analyste</i>	
	<i>Concepteur</i>	
	<i>Développeur</i>	
	<i>Spécialiste design & stratégie</i>	
PREVOST Valentin	Responsable communication	valentin.prevost@etu.u-pec.fr
	<i>Analyste</i>	
	<i>Développeur</i>	
	<i>Concepteur</i>	
	<i>Spécialiste UX</i>	
MEFOUNG EFONTSE Marc	<i>Analyste</i>	marc.mefoung-efontse@etu.u-pec.fr
	<i>Concepteur</i>	
	<i>Développeur</i>	
	<i>Spécialiste management</i>	

HAMADACHE Lynda	<i>Analyste</i>	lynda.hamadache@etu.u-pec.fr
	<i>Développeur</i>	
	<i>Testeur</i>	
	<i>Spécialiste planification</i>	
ZHANG Heng	<i>Analyste</i>	heng.zhang@etu.u-pec.fr
	<i>Développeur</i>	
	<i>Testeur</i>	
	<i>Spécialiste traduction</i>	
PONTE Sylvestre	<i>Développeur</i>	sylvestre.ponte@etu.u-pec.fr
	<i>Concepteur</i>	
	<i>Testeur</i>	
	<i>Spécialiste outils</i>	

Tâche et responsabilité en R0

N°	Livrable	Membres	Responsable
1	<u>Vision Document</u>	Valentin & Aïssata	Valentin
2	<u>UC Model</u>	Sylvestre	Sylvestre
3	<u>Software Development Plan</u>	Marc	Marc
4	<u>Roadmap du projet</u>	Lynda	Lynda
5	<u>Benchmarking technologique / Diagramme de composants de l'architecture logicielle préliminaire</u>	Tout le monde	Aïssata
6	<u>Prototype d'architecture logicielle / Déploiement et démo de ce prototype</u>	Tout le monde	Valentin
7	<u>Prototypes d'Environnements de développement et d'intégration / Démo</u>	Tout le monde	Sylvestre
8	<u>Analyse des risques du projet</u>	Heng	Heng

3.3. PRODUCT OWNER (PO)

Cet intervenant a pour responsabilité principale de donner des brèves descriptions des fonctionnalités à développer durant nos réunions. Il répondra donc à toutes nos questions

concernant le métier. Il a donc un rôle important pour les réunions de chiffrage. Par ailleurs, il est celui qui priorise les US (User Story) au sein de la Product Backlog qui est la liste ordonnée des éléments constitutifs du produit. Compte tenu de la difficulté de l'avoir en présentiel durant toutes nos séances, un membre de l'équipe fera office de représentant du PO. Il devra donc s'entretenir régulièrement avec le P.O afin de partager au mieux les retours de leurs échanges à l'équipe. Lors de la démo, il devra/pourra donc accepter ou refuser les User Story implémentées par l'équipe de réalisation. Nous aurons donc besoin de le contacter régulièrement.

Pour ce rôle nous avons choisi Mme MAYENNE Sarah-Alvine, Consultante système d'information à BNP Paribas.

4. GESTION DU PROJET

4.1. CHARGE DE TRAVAIL

La charge de travail est estimée en moyenne à **8 heures** par personne et par semaine, et **4 heures** de réunion chaque semaine.

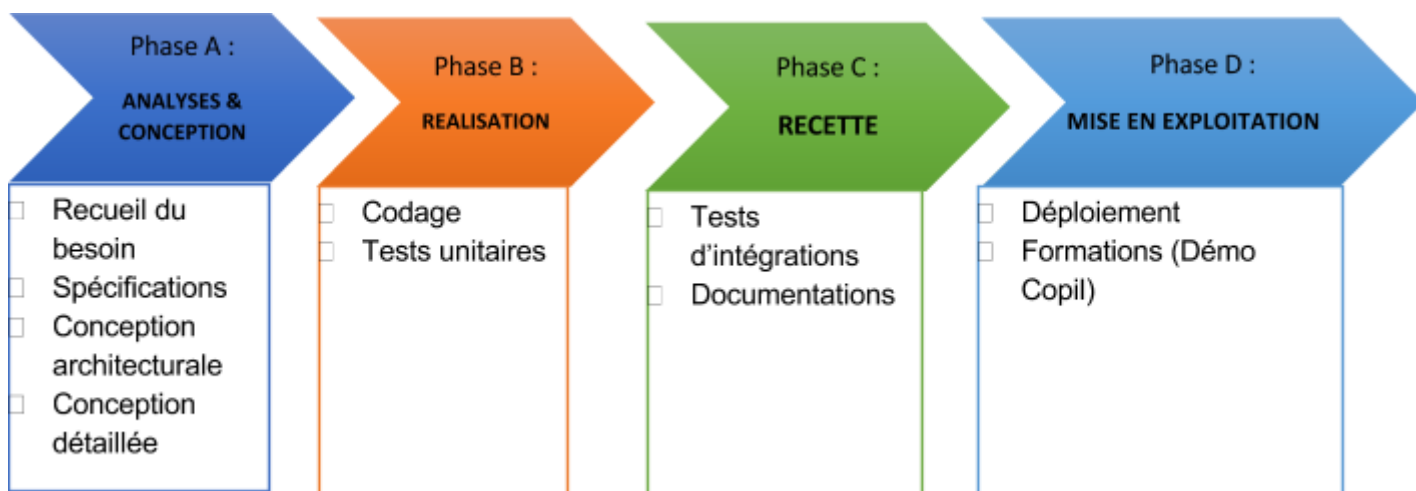
Cette charge est estimée de nouveau à chaque Release en fonction de l'importance des travaux à rendre.

4.2. PLAN DE PROJET

4.2.1. PLAN DE PHASE

Le projet est organisé en 2 Releases. Pour atteindre les objectifs fixés à chaque Release, nous suivons un développement itératif : à partir de la R1, à chaque fin d'itération sera produite une version fonctionnelle de l'application. Cette méthode a pour but **d'éliminer les risques le plus tôt possible**. A chaque Release du processus de développement pourront correspondre une ou plusieurs itérations.

Voici le déroulement chronologique de chaque itération :



4.2.2. DÉROULEMENT DES PHASES ET VALIDATION

A- RELEASE 0 ((Spécification, Conception))

Cette première release est composée de 3 phases distinctes :

- Une phase de recueil des spécifications clients (que l'on retrouve en partie dans le sujet) et l'apprentissage du métier à transposer informatiquement.
- Une phase de spécification consistant en la rédaction de documents de spécifications décrivant les modules à développer (UC Model).

Les livrables liés sont :

- Software Development Plan
- Vision document & UC Model.
- Une phase de conception dans laquelle nous allons étudier l'organisation et les moyens à mettre en place pour atteindre nos objectifs de réalisation dans les meilleurs délais.

Les documents liés sont :

- Benchmarking technologique / Diagramme de composants de l'architecture logicielle préliminaire
- Roadmap du projet
- Prototype d'Environnements de développement et d'intégration
- Analyse des risques du projet

B- RELEASE 1

Conception détaillée, chaque module va être développé testé et documenté.

Parmi les documents liés nous aurons entre autres :

- Eventuel splike requis
- Mocks
- Diagrammes(DSS, DCP, DSCP, DCCP, DSCD, DCCD) pour chaque UC
- Diagramme de Classe Métier
- Le cahier de recette
- Les Tests unitaires
- Spécification du S.u.d etc.

Ces livrables se feront au minimum une fois par itération.

- Release/Iteration plan
- Etat d'avancement
- TU/TI/TA
- 3 environnements complets
- Release de l'Application (Démo, Code source, Déploiement)
- Mocks tables

4.2.3. PLANNING PREVISIONNEL

A. PLAN DES RELEASE

Release	Nombre d'itérations	Début	Fin	Durée
R0	1	18/09/2016	23/10/2017	05 semaines
23/10/2017 - COMITÉ DE PILOTAGE				
R1	5	23/10/2017	23/02/2017	18 semaines
23/02/2017 - COMITÉ DE PILOTAGE				

La R0 fait l'objet d'une seule itération de 5 semaines. De façon prévisionnelle, la R1 fera l'objet de 5 itérations avec une durée de 4 semaines pour les 4 premières itération et 2 semaines pour la dernière itération. Il est à noter que le Mock des données se fera au fur et à mesure de l'avancement du projet.

B. OBJECTIFS DES ITÉRATIONS

Itération 0 :

- Analyse et formalisation du besoin fonctionnel
- Identification des cas d'utilisation (UC Model)
- Définition des cas d'utilisation principaux
- Benchmarking
- Software Development Plan
- Mise en place de l'environnement de développement et d'intégration continue
- Définition de l'architecture à utiliser
- Définition de tous les cas d'utilisations retenus (dans le périmètre du projet)
- Planification de la gestion du projet (Sous JIRA)
- Réalisation d'un prototype (non-métier)
- RoadMap du projet

Itération 1 :

- Composition des binômes (par l'équipe pédagogique) et attribution des UCs à implémenter
- Diagrammes (Diagramme de Classes Métier, Diagramme d'activité, Diagramme de Séquence Système, Diagramme de Classes Participantes, Diagramme de Séquence de Conception Préliminaire, Diagramme de Classes de Conception Préliminaire)
- UC Détail

Itération 2 :

- Amélioration de l'architecture
- Développement des Use case métiers
- Mocks
- Tests

Itération 3 :

- Tests Unitaire
- Correction des erreurs éventuelles
- Intégration
- Cahier de recettes
- Développement des Use case métiers
- Tests
- Déploiement

Itération 4 :

- Développement des fonctionnalités manquantes
- Mocks tables
- Tests

Itération 5 :

- Tests Unitaire

- Correction des bugs éventuels
- Intégration
- Rédaction de la documentation

5. SUIVI DE PROJET ET CONTRÔLE

5.1. MOYEN DE COMMUNICATION

Afin de faciliter le partage d'information et d'avoir une communication efficace entre les membres de l'équipe de développement, nous avons mis en place un serveur de messagerie instantanée sur la plateforme **Slack**. Ce qui nous permet d'envoyer des messages instantanés à l'ensemble des membres mais surtout faciliter les discussions en fonction des thématiques bien précises par la création de **channels** dédiés. En ce qui concerne le partage de fichiers, nous utilisons **Google Drive**, de manière à pouvoir travailler à plusieurs sur un même fichier, et pouvoir consulter le travail des autres membres.

5.2. RÉUNIONS

Pendant toute la durée du projet nous organiserons des réunions. Compte tenu des principes Agiles, nous aurons principalement 3 types de réunions:

- Du daily meeting

Etant dans l'incapacité de respecter scrupuleusement le principe de standup meeting, car n'ayant pas la possibilité de nous retrouver tous les jours, nous nous retrouvons en moyenne 2 fois par semaine pour une durée de quinze minute environ. Pendant ces réunions où chaque membre de l'équipe est généralement debout, chacun fait un point sur l'avancement de ses travaux ou éventuellement fait part des difficultés rencontrées. Chaque personne explique ce qu'elle a fait depuis la dernière réunion et ce qu'elle fera par la suite.

Nous avons également des réunions de synchronisation où nous organisons tous nos travaux pour préparer soit des séances de cadrage soit des comités de pilotage. Ces réunions sont de 3h en moyenne. En cas de force majeure nous organiserons ces réunions sous forme de Visioconférence notamment en R2 où certains membres de l'équipe seront en stage à l'étranger.

- Planning POKER

C'est pour nous une façon ludique de produire de l'estimation sur les US à développer. L'exercice consiste à ce que chaque membre de l'équipe possède une carte correspondant à une valeur (suite de Fibonacci). Lorsqu'on donne le nom de l'US à développer, chacun choisit secrètement une carte correspondant à la valeur de SP dont il estime correspondre l'US en question. Le Scrum Master se charge de collecter les résultats. S'il y a un grand écart entre plusieurs choix, nous en discutons ensuite recommençons le vote?

- Planification de Sprint

Durant cette réunion, nous essayons de prioriser les US à développer et à attribuer une dateline à chacun.

- Revue de sprint

Le but de cette réunion est de montrer ce qui a été réalisé pendant le sprint afin d'en tirer les conséquences pour la suite du projet. Les participants à cette réunion sont : toute l'équipe, avec le ScrumMaster et le ProductOwner.

- **Sprint retrospective**

Le but de la réunion est d'améliorer le processus pour le prochain sprint. Elle a lieu juste après la revue de sprint.

En plus de ces 4 principales réunions, nous aurons souvent une réunion de revue de backlog qui a pour objectif de préparer l'itération n+1 un peu avant la fin de l'itération n.

Afin d'assurer l'efficacité dans la planification de certaines dates de réunion, nous avons créé un sondage sur la plateforme **Doodle**, afin de choisir les dates qui arrangeraient le plus grand nombre des membres de l'équipe. Par ailleurs, pour que chacun puisse se rappeler des dates des réunions, nous mettons en place un **Google Agenda** afin d'inviter tous les membres de l'équipe et donc qu'ils aient un rappel automatique lorsque la date de réunion approche.

Planning des réunions :

Type de réunion	Date	Ordre du jour/Thèmes	Timing	Compte rendu
Brainstorm	22.09.2017	<i>Annexe</i>	4 H	
Avancement	29.09.2017	<i>Annexe</i>	4 H	
Mise au point	05.10.2017	<i>Annexe</i>	4 H	
Avancement	06.10.2017	<i>Annexe</i>	4 H	
Synchronisation	10.10.2017	<i>Annexe</i>	4 H	
Mise au point	16.10.2017	<i>Annexe</i>	4 H	Annulé
Mise au point	17.10.2017	<i>Annexe</i>	4 H	
Synchronisation	20.10.2017	<i>Annexe</i>	4 H	
Pré-COPIL	25.10.2017	<i>Annexe</i>	4 H	-
COPIL	27.10.2017	<i>Annexe</i>	2 H	-
Revue	13.11.2017	<i>Annexe</i>	4 H	-
Synchronisation	21.11.2017		4 H	-
Revue	05.12.2017		3 H	-
Revue	13.12.2017		3 H	-
Synchronisation	15.01.2017		3 H	-
Revue	29.01.2017		3 H	-
Revue	07.02.2017		3 H	-
Pré-COPIL	13.02.2017		3 H	-

5.3. **CONTRÔLE DE LA QUALITÉ**

Tous les livrables sont soumis à un processus de révision. Cela nous permet de s'assurer au moyen de vérification, de la qualité de chaque livrable mais aussi de la cohérence dans les

travaux réalisés parallèlement. Par ailleurs, nous mettons en place une stratégie de test, qui consiste à exécuter automatiquement les TU à chaque fois qu'un membre de l'équipe fait un nouveau push sur la branche master du dépôt git. Nous organisons également le développement dans un respect scrupuleux des branches afin que chaque binôme crée une branche pour des nouvelles features à développer et surtout pour que le build sur la branche master soit toujours fonctionnel.

6. ENVIRONNEMENT TECHNIQUE

Les résultats de notre benchmark nous ont conduit vers le choix de l'environnement technique suivant:

- Framework FrontEnd : **ReactJS**
- Framework BackEnd : **Spring**
- Langage: **Java/J2EE**
- Design pattern: **DAO, DTO, Entity, REST**
- Conteneur : **Apache Tomcat 8, Node.JS**
- B.I : **SSIS, SSRS**
- Gestion de projet : **JIRA**
- Versioning: **GitLab**
- Architecture technique : ***Document en annexe***
- Architecture cible: ***Document en annexe***