

**Rapport**

**Informatique**

WIJHATI

|  |
| --- |
| **Réalisé Par :**   * **Moncif KHOULFI** * **Imane ELHAMOUCHI** * **Soukaina BENDAIF** * **Noura EL WAFI** |

|  |
| --- |
| Encadré Par :   * **Mme. Cheikh Lehlou** |

**SOMMAIRE :**

Introduction

Liste des Figures

Liste des Abréviations

**Chapitre 1 :Presentation generale du Ptojet**

[1. NOM du projet - 3 -](#_Toc351760597)

[2. RESUME du projet - 3 -](#_Toc351760599)

[3. DESCRIPTION DE LA SOLUTION - 3 -](#_Toc351760601)

[4. EQUIPE du projet - 3 -](#_Toc351760601)

[Chapitre 2 : Gestion de projet - 4 -](#_Toc351760603)

[1. La méthode Agile : - 4 -](#_Toc351760604)

[1.1. Pourquoi Scrum ? - 4 -](#_Toc351760605)

[1.2. Cycle de vie - 5-](#_Toc351760607)

[1.3. Diagramme de GANTT - 6 -](#_Toc351760607)

**Chapitre 3: Les ressources………………………………………………………………………………………………………….. -6-**

[1. Comparatif des systèmes des gestion des bases de données…………………………………………………………….-6-](#_Toc351760609)

[1.1. Familles de critères……………………………………………………………………………………………………………………………………...-7-](#_Toc351760610)

[1.2. Solution retenues :……………………………………………………………………………………………………………………………………...-7-](#_Toc351760613)

[2. Comparatif d’un système de gestion de bases de données de NOSQL - 7 -](#_Toc351760615)

[2.1 Familles de critères - 7 -](#_Toc351760620)

[2.2 Solution retenues - 7 -](#_Toc351760620)

[3. Comparatif des Environnement de développement (IDE) - 8 -](#_Toc351760636)

[3.1 Critères de comparaison - 8 -](#_Toc351760637)

[3.2 Solution retenue - 9 -](#_Toc351760638)

[4. Comparative des Plateformes………………………………………………………………………………………………………………..-9-](#_Toc351760639)

[4.1 Description des plateformes - 10 -](#_Toc351760640)

[4.2 Solution retenue - 10 -](#_Toc351760650)

[5. Comparatif des Composants HTML : - 11 -](#_Toc351760652)

[5.1 Famille de critères - 11 -](#_Toc351760653)

[5 .2 Matrice d’évaluation - 12 -](#_Toc351760655)

[5.3 Solution retenue - 12 -](#_Toc351760656)

[6. Comparatif des ERP: - 13 -](#_Toc351760652)

[6.1 A quoi sert un ERP - 13 -](#_Toc351760653)

[6.2 Pourquoi Open ERP - 14 -](#_Toc351760654)

[6.3 Matrice d’évaluation - 15 -](#_Toc351760655)

[6.4 Solution retenue - 15 -](#_Toc351760656)

[7. RESULTAT de scoring : - 15 -](#_Toc351760657)

[Chapitre 4 : EXPRESSION DES BESOINS - 17 -](#_Toc351760658)

[1. ETUDE de cas - 17 -](#_Toc351760659)

[1.1. Choix UML - 18 -](#_Toc351760660)

[1.2. IDENTIFICATION DES BESOINS - 19 -](#_Toc351760661)

[1.3. Diagramme de cas d’utilisation - 20 -](#_Toc351760661)

[1.4. Documentation de quelques cas d’utilisation - 21 -](#_Toc351760662)

[1.5. Diagramme de class - 23 -](#_Toc351760662)

Conclusion………………………………………………………………………………………………………………………………………………..-23-

Introduction

Dans le contexte du développement de son projet, l’équipe **E-SEHATI** s’investit à donner le meilleur résultat possible. En effet, elle combine la créativité et l’innovation pour donner à son produit une utilisabilité maximale.

A la base de la réussite de tout projet, on trouve la méthodologie et l’organisation. Ces dernières permettent d’atteindre l’objectif fixé et les résultats escomptés.

Etant des futurs lauréats du Master 2 MIAGE, les membres de l’équipe d’**E-SEHATI** ont un savoir en Management des projets. D’ailleurs, ils le mettent actuellement en exercice pour la réalisation de leurs projets de stage et vont exploiter les acquis de cette expérience pour la bonne gestion de leur projet.

Pour aboutir à des livrables qui répondent aux besoins des clients, plusieurs ressources logicielles sont nécessaires. Pour cette raison, nous avons effectué une étude comparative de logiciels relative à notre besoin interne. Cette dernière nous a permis d’identifier les outils nécessaires à l’exploitation.

Théoriquement, le management des projets et les ressources logicielles sont des moyens qui permettent d’entreprendre des projets réussis. Par la pratique, nous nous sommes assurés de leur importance.

Ce document fourni l’étude effectuée et présente les outils de travail aussi qu’un aperçu des premières étapes concernant le projet **E-SEHATI**.

Le présent rapport a pour vocation, de présenter succinctement notre projet de création d’entreprise en synthétisant ses différentes phases de développement tout au long de son déroulement, tout en mettant en exergue sa consistance.

Liste des Abréviations:

|  |  |
| --- | --- |
| **Sigle** | **Signification** |
| A |  |  |
| B |  |  |
| C | **CDC** | Cahier des Charges |

Liste des Figures

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |

**Chapitre 1** : Présentation de projet

1. NOM du projet

WIJHATI

2. RESUME du projet

Notre objectif est de mettre en place une plateforme, WIJHATI **‘**GPS Social**’** est une application mobile communautaire gratuite qui permet de fournir, construire et améliorer la cartographie des routes, de notifier, d’informer sur les perturbations de circulation, les accidents les travaux, le contrôle de police/ radars et embouteillages, de réduire la consommation du carburant et gagner le temps perdus dans les embouteillages.

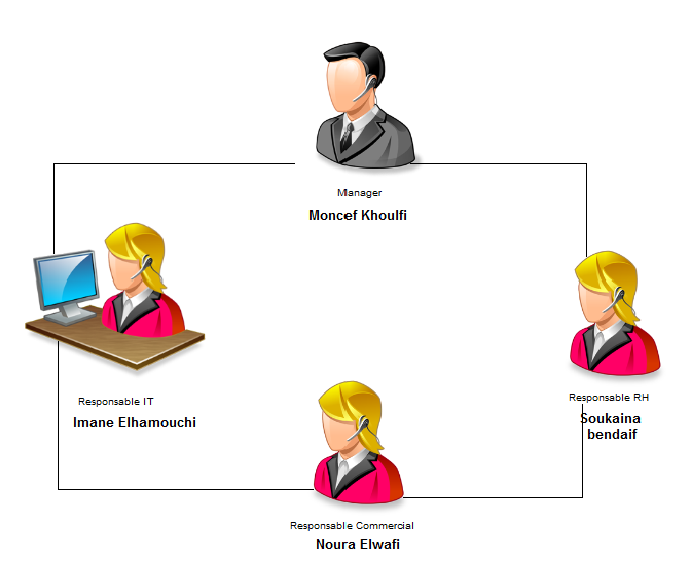
La solution proposée ne se contente pas d’indiquer les chemins les plus optimale mais

propose également des publicités relatives aux différents types d’Entreprises (restaurants, cafés, cinéma, opérateur de télécommunications,…)

4. EQUIPE du projet

Une Equipe jeune, Futurs Lauréats du MASTER 2 option NTDP, ambitieuse motivée, et ayant un esprit d’entreprenariat. Ces associés mettent leurs ressources et leur expérience en commun afin d’améliorer les problèmes infernaux de la circulation routière au niveau du grand Casablanca.

L’auteur du projet est la Société **WIJHATI**  créé par :





**Cette partie présente la méthode utilisée pour la gestion de notre projet WIJHATI**

**Chapitre 2** : Gestion de projet

|  |
| --- |
| Afin d’assurer le bon fonctionnement et organiser de bout en bout le bon déroulement du projet, une étape de gestion s’avère nécessaire |

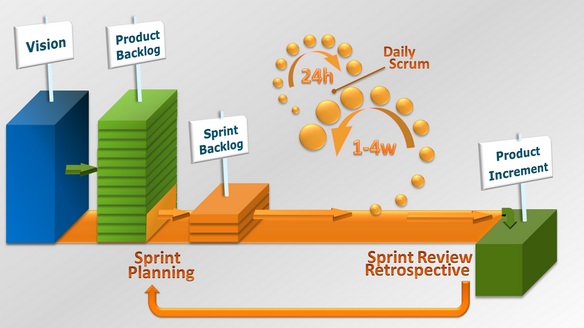
1. La Démarche Agile :
   1. Pourquoi Scrum ?

Pour pallier aux problématiques de gestion dans les projets de développement logiciel qui utilisait une approche traditionnelle, **Scrum**, une adaptation du modèle de gestion agile propre à la gestion de projet est arrivée les années 90.

Changement de culture radical qui privilégié et la collaboration aux boîtes de temps et la planification préconisée par les PMI et l’ensemble des universités à ce jour. Avec une approche Agile avec Scrum, les individus d’un projet sont invités à participer à l’évolution du logiciel.

Scrum est fondé sur la théorie de contrôle des processus empiriques. Cette théorie affirme que la connaissance s'acquiert par l'expérience et favorise la prise de décision basée sur ce qui est connu. Scrum utilise une approche itérative et incrémentale pour optimiser la prévisibilité et le contrôle des risques.

Scrum fait partie des approches itératives et incrémentales, dont le modèle de cycle de développement est basé sur une phase qui se répète plusieurs fois successivement. C’est la notion d’itération, appelée Sprint avec Scrum. Tous les Sprints se déroulent selon le même schéma et on y fait à chaque fois les mêmes types de travaux.



*Figure : Composants D’un Scrum Sprint*

la Longueur du Sprint est maintenue égale pour des raisons de jeu cadence et des métriques telles que la vélocité. La durée exacte doit être accordée à la capacité à capitaliser sur les opportunités commerciales émergentes (les transformer en incréments de logiciels de travail), mais aussi dépend de combien de temps une équipe peut travailler sans consulter les parties prenantes. Un développement de logiciels Sprint prend généralement 1-4 semaines, avec une réunion quotidienne de 15 minutes maximum (DSTUM).

A la fin d’un Sprint, le résultat attendu est un incrément du produit final, qui est Potentiellement livrable.

* 1. Phases du Projet :

|  |
| --- |
| Pour qu'il soit géré dans un contexte de qualité, un projet doit suivre différentes phases au terme desquelles des points de contrôle doivent être définis. |

L’élaboration des phases du projet décrivant son cycle de vie, constitue une étape primordiale pour bien structurer les tâches à effectuer dans le temps nécessaire, afin de bien mener le projet dans ces différentes étapes sans encombrement.

 Le cycle de vie du projet constitue l'enchaînement dans le temps des étapes et des validations entre l'émergence du besoin et la livraison du produit.

Les phases du cycle de vie du projet sont décrites dans la figure ci-dessous,

**Figure : Cycle de vie du projet**

Chaque phase du cycle de vie du projet, fait l'objet d'un livrable et d'une validation. Cela permet de maîtriser la conformité des livrables à la définition des besoins ainsi que de s'assurer de l'adéquation aux objectifs en termes de coûts et de délais.

La figure suivante, décrit les livrables et les tâches constituant chaque phase du cycle de vie du projet.

**Figure : Description des phases du Projet**

* 1. Déroulement du Projet :
     1. Planning Prévisionnel
     2. Planning Réel
     3. Analyse des Ecarts

Afin d’évaluer le rendement d’un projet, on a besoin de comparer la réalité avec le planning initial, afin d’avoir une vision claire de la réalité et de calculer les écarts en termes d’avancement, de rendement et de coûts.

* 1. Livrables

Cette partie concerne l’identification des livrables associés à chaque phase de notre projet, qui peut être un résultat ou un document présentant l’avancement de la réalisation au niveau du projet.

|  |  |
| --- | --- |
| Livrable | Description |
| Planning du Projet | Un planning comportant le déroulement du projet ainsi que les différentes étapes de ce dernier. |
| Cahier des Charges | Référentiel qui décrit la liste des besoins, des exigences et des contraintes qu'il faut respecter lors de la réalisation du projet |
| Document de Qualité : PAQ | Le gabarit du Référentiel de Qualité |
| Plan d’affaires | Référentiel qui formalise par écrit les projections d'évolution de notre entreprise. |
| Rapport Informatique | Rapport Final contenant la description Informatique du projet réalisé |

* 1. Analyse des Risques

Cette partie concerne notre démarche relative à la gestion des risques, pouvant perturber et gêner le bon déroulement de la mise en place de notre projet. Dans un premier temps, nous avons dressé un listing non exhaustif des risques pouvant survenir au cours de la mise en œuvre du projet. Pour chaque risque, nous estimerons l’impact suivi d’un ensemble d’actions préventives et correctives.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Risque | Type | Importance | Impact | Actions Correctives |
| Erreur Organisationnelle | RB |  | Echec du Projet | Prévoir des réunions et des points de validation avec l’encadrant fonctionnel au fur et à mesure de l’avancement du projet |
| Le projet est inutilisable sans un temps excessif de formation. | RB |  | Retarder la mise en production du projet | * Veiller à chaque étape de réalisation à intégrer l’ergonomie * Consolider la documentation utilisateur |
| Les compétences insuffisantes pour la mise en œuvre du projet | RNB |  | Retarder le projet | Effectuer une auto-formation et d’autres recherches afin de mettre en œuvre la solution adéquate. |
| La contrainte de temps | RNB |  | Le projet ne sera pas achevé dans la date prévue de lancement. | Ajuster le planning pour respecter la planification faite au départ. |
| Les pannes inattendues du Matériel (Bugs) | RNB |  | Ralentissement des travaux | Utiliser les autres matériaux disponibles. Recours à une réparation rapide. |
| Absence ou maladie | RNB |  | Ralentissement des travaux | Travailler en temps extra. |

|  |
| --- |
| Importance :  Forte Moyenne Faible |

**Chapitre 3**: Etude Préalable

1. Comparatif des systèmes entre les langages des bases de données

Dans cette partie nous allons comparer 2 langages de bases de Données notamment :

* SQL
* NOSQL
  1. Familles de critères

Pour avoir une étude significative, nous nous sommes basées sur différents critères d’évaluation qui seront catégorisés en 4 familles :

|  |  |
| --- | --- |
| SQL | Nosql |
| Dans le cadre de sécurité, il permet de contrôler la sécurité et les accès aux données. | **la sécurité** des bases NoSQL où, dans la plupart des cas, elle est assurée par filtrage réseau et il n’y a pas de gestion d’authentification. |
| la perte de performance lorsque on doit traiter un très gros volume de données | **La performance** des bases NoSQL sont de manière générale plus rapides que les bases SQL car elles n’implémentent que les mécanismes dont elles ont besoin et laissent de côté tout ce qui est nécessaire au fonctionnement d’un modèle SQL . |
| Il y a encore peu de retours sur la fiabilité des moteurs SQL (cela ne veut pas dire pour autant qu’elle n’est pas bonne, mais l’expérience compte dans ce domaine). | **NOSQL** garantissent la fiabilité d'une base de données.  ne remplace donc pas le modèle relationnel mais offre des solutions intéressantes. |
| les serveurs SQL ne sont pas faits pour stocker de grandes quantités de données binaires. | **facilite le stockage** efficace de quantités massives de données dans un format simple et flexible. |

*tableau 1 : familles de critères pour l'étude des SQLvs Nosql*

* 1. Matrice d’évaluation

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Etude comparative des langages de BD | | 20/03/2013 | | | |
| Poids: 1 = Faible, 3 = Moyenne, 5 = Forte Importance | | | **Rang** | | |
| Familles | Critères | ***Poids*** | | **NOSQL** | **SQL** |
| 1 | **Sécurité** | 3 | | **5** | **5** |
| 2 | **Performance** | 5 | | **5** | **3** |
| 3 | **Fiabilité** | 3 | | **5** | **3** |
| 4 | **Capacité de stockage** | 5 | | **5** | **3** |
| *Résultat/Total Score :* | | | | **80** | **54** |

1. *tableau5 : familles de critères pour l'étude des langage de BD*
   1. Méthode d’évaluation :

Afin de trouver la solution de Base de données la plus adéquate, nous avons mis en place un algorithme d’évaluation ;

|  |
| --- |
| Algorithme :  {  Résultat= Somme (Rang \* Poids) ;  } |

* 1. Solution retenue :

Il est apparut clair que la solution la plus convenable est le NOSQL, mai comme NOSQL est basé sur plusieurs SGBD, on a poursuit à une étude comparative pour choisir celui qui sera adapté à nos besoins en termes des gestions des données.

* + 1. Comparatif des SGBD de NOSQL:

L’étude comparative entre les différents SGBD du langages NOSQL est basé sur 4 critères, à savoir :

* + **Clé / valeur**
  + **Orienté colonne**
  + **Orienté document**
  + **Orienté graphe**
    1. Familles de critères :

Les solutions NoSQL existantes peuvent distingués selon 4 critères :

* **Clé / valeur**

Ce modèle peut être assimilé à une hashmap distribuée. Les données sont, donc, simplement représentées par un couple clé/valeur. La valeur peut être une simple chaîne de caractères, un objet sérialisé… Cette absence de structure ou de typage ont un impact important sur le requêtage. En effet, toute l’intelligence portée auparavant par les requêtes SQL devra être portée par l’applicatif qui interroge la BD. Néanmoins, la communication avec la BD se résumera aux opérations PUT, GET et DELETE. Les solutions les plus connues sont Redis**, Riak** et **Voldemort** créé par LinkedIn.

* **Orienté colonne**

Ce modèle ressemble à première vue à une table dans un SGBDR à la différence qu’avec une BD NoSQL orientée colonne, le nombre de colonnes est dynamique. En effet, dans une table relationnelle, le nombre de colonnes est fixé dès la création du schéma de la table et ce nombre reste le même pour tous les enregistrements dans cette table. Par contre, avec ce modèle, le nombre de colonnes peut varier d’un enregistrement à un autre ce qui évite de retrouver des colonnes ayant des valeurs NULL. Comme solutions, on retrouve principalement **HBase** (implémentation Open Source du modèle BigTable publié par Google) ainsi que Cassandra (projet Apache qui respecte l’architecture distribuée de Dynamo d’Amazon et le modèle BigTable de Google).

* **Orienté document**

Ce modèle se base sur le paradigme clé valeur. La valeur, dans ce cas, est un document de type JSON ou XML. L’avantage est de pouvoir récupérer, via une seule clé, un ensemble d’informations structurées de manière hiérarchique. La même opération dans le monde relationnel impliquerait plusieurs jointures. Pour ce modèle, les implémentations les plus populaires **sont CouchDB** d’Apache, **RavenDB** (destiné aux plateformes .NET/Windows avec la possibilité d’interrogation via LINQ) et **MongoDB**.

* **Orienté graphe**

Ce modèle de représentation des données se base sur la théorie des graphes. Il s’appuie sur la notion de nœuds, de relations et de propriétés qui leur sont rattachées. Ce modèle facilite la représentation du monde réel, ce qui le rend adapté au traitement des données des réseaux sociaux. La principale solution est **Neo4J.**

Les deux mouvements “orienté colonne” et “orienté document” découlent bien du système clé valeur et c’est la nature ou la structure de la valeur qui diffère.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | CouchDB | | Neo4j | HBase | Riak |
| Langage de programmation | | Erlang | Java | Java | Erlang, C et JavaScript |
| Protocol | | HTTP/REST | HTTP/REST | HTTP/REST | HTTP/REST |
| Projets | | http://wiki.apache.org/couchdb/CouchDB\_in\_the\_wild | Google Maps,  TomTom et plus encore. | • Facebook: infrastructures messages  • Twitter: lecture / écriture de sauvegarde des tables mysql, la recherche de personnes ...  • Yahoo!: pour détecter les doubles emplois document basé sur les empreintes digitales. | •Ask.com  •Comcast  •Mozilla  http://wiki.basho.com/Who-is-Using-Riak.html |
| Exemple d’utilisation | | CRM, systèmes CMS  l'industrie (mais pas  nécessairement pour  les transactions financières,  mais ces industries sont  beaucoup plus que cela.) | Pour le style graphique, riche et complexe, les données interconnectées.  Pour rechercher les routes dans les relations sociales, les transports publics, des cartes routières, ou topologies de réseau. | Facebook Messaging Database. | Point de vente pour collecter des données.  Systèmes de contrôle de l'usine.  Si vous avez besoin d'une très bonne site unique d'évolutivité, de disponibilité et de tolérance aux pannes, mais vous êtes prêt à payer pour la réplication multi-sites. |
| Plateforme | | Multi plate-forme | Multi plate-forme | Multi plate-forme | Linux, Mac Os |

*tableau2 : familles de critères pour les 4familles de Nosql*

***Source web :*** [**http://kkovacs.eu/cassandra-vs-mongodb-vs-couchdb-vs-redis**](http://kkovacs.eu/cassandra-vs-mongodb-vs-couchdb-vs-redis)

* + 1. Solution retenues :

Pour réaliser notre application mobile, on a besoin d’un framework qui supporte :

* Le langage java ;
* Les projets de types mobiles ;

D’après ces précisions, on a trouvé que le framework SGBD de **Neo4j** est celui qui va être utilisé pour le langage NOSQL.

1. Comparatif des Environnement de développement (IDE)

Nous avons réalisé une présélection des IDE afin de ne pas se disperser avec un comparatif trop large. Les environnements retenus sont donc NetBeans et Eclipse. Cela est principalement dû au fait que ce sont les IDE Java connus pour être les plus performants et les plus largement utilisés.

* 1. Critères de comparaison

1. Fonctionnalités

Les deux outils cités précédemment possèdent les fonctionnalités et besoins désirés. En effet, ils gèrent :

* La compilation et le déploiement d’applications web ;
* L’interfaçage avec un gestionnaire de versions ;
* Un débogueur permettant de détecter des bogues dans l’application (il peut aussi servir à tester cette dernière).

1. Connaissance

Ce critère s’avère déterminant dans le choix de l’IDE qui sera utilisé. Eclipse est l’environnement de développement java préconisé par nos enseignants et sur lequel nous avons été formé. Certaines personnes de l’équipe possèdent, une forte expérience sur Eclipse du fait de leurs travaux de stages en entreprise.

1. Ergonomie

Pour ce critère d’ergonomie, nous recherchons une application qui puisse s’installer rapidement et être facile à prendre en main. Mais il faut aussi que celui-ci dispose d’une interface agréable et permettant de pouvoir rapidement effectuer toutes les actions nécessaire rapidement. Eclipse sort ici vainqueur dans le sens où il dispose d’une interface plus intuitive et d’un réel programme d’installation. Il propose toutes les fonctionnalités en « one-clic » ce qui permet de pouvoir réaliser toutes les actions, dont l’équipe a besoin, très rapidement.

* 1. Comparaison et Résultat :

Après réalisation de ce workbench :

*Workbench des environnements de développement*

Après cette analyse décrite au niveau du diagramme en dessus, L’IDE choisi est donc **la plateforme Eclipse**.

3. Etude COMPARATIVE des plateformes.

3.1. Description des plateformes

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Plateforme | Windows Mobile | Blackberry | iOS | Android |
| Société | **Microsoft** | **RIM** | **Apple** | **Open Handset Alliance** |
| Description | Puissant  Compatibles avec tous les logiciels de Windows | Multitâches  Bien intégration avec autres plateformes  Déploiement facile pour une société  Gestion facile  La vie de la batterie plus longue | le système d’exploitation mobile développé par Apple pour l’iPhone, l’iPod touch, et l’iPad | Code Source Ouvert  API Ouvert |
| Langage de programmation | C#/VB, C++ | Java | Java | Java |
| IDE + SDK | Visual Tools v4 (gratuit) | BlackBerry JDE  (Java Développent Environnent) | Eclipse  Netbeans | Eclipse (Android SDK pour Windows, Mac OS X )  NetBeans (Android plugin) |
| Environnement | Windows/Mac OS/Linux | Windows/Mac OS | Windows/Mac OS | Windows /Linux |
| Les applications principales | calendrier  - gestion de la tâche  - Internet Explorer Mobile | Utiliser facilement quelques applications simples comme : carnet d’adresses, gestion du temps, SMS, téléphone | les fonctions de téléphonie ordinaire: contrôle d'appel, app-store, iTunes,calendrier,caméra,révéil,etc. | Les fonctions de téléphonie ordinaire: contrôle d'appel, conférences téléphoniques …  Navigateur Web (supporte le HTML et XHTML)  Réveil, calendrier, Caméra, Contacts, messagerie |

*tableau3 : familles de critères pour l'étude des OS*

* 1. Solution retenue

La plateforme choisi c’est **ANDROID et iOS.**

1. COMPARATIF des Composants HTML :

Dans cette partie nous allons comparer 2 Composants HTML notamment :

* **CANVAS:** une balise HTML5 permettant d’y dessiner ce que l’on veut.
* **SVG :** est un format d’image vectoriel.
  1. Famille de critères

|  |  |  |
| --- | --- | --- |
| **Familles** | **CANVAS** | **SVG** |
| **Niveau d’abstraction** | Basé sur les pixels (PNG dynamique) | Basé sur des formes & vecteurs |
| **Eléments** | Un unique élément HTML similaire à <img> dans son comportement | De nombreux éléments graphiques qui font alors partis du Document Object Model (DOM) |
| **Interaction** | Modifiable uniquement par Script | Modifiable par Script et CSS |
| **Modèle évènementiel** | Interaction utilisateur très granulaire (x,y) de la souris) | Interaction utilisateur abstraite et gérée par le DOM |
| **Performance** | Les performances sont meilleures avec des petites surfaces et/ou un nombre important d’objets à l’écran (>10k) | Les performances sont meilleures avec un nombre inférieur d’objets (<10k) et/ou sur de plus larges surface de dessin |

Pour avoir une étude significative, nous nous sommes basé sur différents critères d’évaluation qui seront catégorisés en **5 familles :**

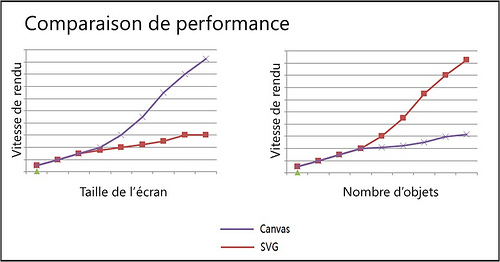
*tableau 4: familles de critères pour l'étude des 2 Composants HTML*

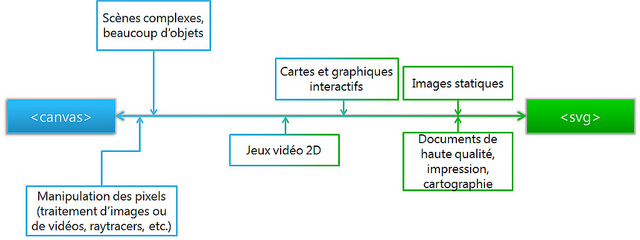
* 1. Matrice d’évaluation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Etude comparative des Composants HTML | | | 20/03/2013 | |
| Poids: 1=Faible, 3=Moyenne, 5=Forte Importance | | | **Rang** | |
| Familles | **Critères** | **Poids** | **CANVAS** | **SVG** |
| 1 | **Niveau d’abstraction** | **3** | **5** | **5** |
| 2 | **Eléments** | **5** | **3** | **5** |
| 3 | **Interaction** | **3** | **5** | **3** |
| 4 | **Modèle évènementiel** | **5** | **5** | **3** |
| 5 | **Performance** | **5** | **5** | **5** |
| Résultat/Total score : | | | **95** | **89** |

*tableau5 : familles de critères pour l'étude des composants HTML*

5.4 Solution retenue

 Après réalisation de ce workbench :



Le Composant HTML choisi est donc **la CANVAS**.

1. Comparaison des erp open source :

Dans cette partie nous allons comparer 2 de types des ERP dédié à la gestion des clients (CRM) et la gestion des paix:

* ERP Commercial
* Open ERP
  1. A quoi sert un ERP :

Un ERP (Enterprise Resource Planning) ou encore en français PGI (Progiciel de Gestion Intégré) est un référentiel central de gestion de toutes les données d'une entreprise, autour duquel gravitent un ensemble de modules fonctionnels qui effectuent les traitements métier sur ces données, par exemple les traitements liés à la vente d'un produit.Ils permettent à des petites PME de disposer d'outils de gestion complets au meilleur coût, leur apportant rapidement un vrai bénéfice en termes de compétitivité. Les seuls coûts étant alors la Formation des utilisateurs et le service éventuellement assuré par le fournisseur du logiciel.

* 1. Pourquoi OPEN ERP

L'utilisation d'une solution Open Source offre les avantages suivants :

* + **Pérennité de la solution** : la mise à disposition du code source de l'application

est le plus grand gage de pérennité de la solution.

* + **Prix** : la philosophie des logiciels Open Source est de ne payer qu'une seule fois les développements. Vous bénéficiez à ce titre (et gratuitement !) de l'ensemble des développements passés et futurs financés par les autres acteurs de la Communauté.
  + **Personnalisation de la solution** : le développement de la solution se fait en fonction de vos vrais besoins.
  + **Solution Business Intelligence :** se limite encore à des rapports paramétrables ceci dit une solution open ERP inclura également une solution de requêter cube OLAP pour des analyses plus fines et sans cout d’intégration démesuré.
  + **Module de comptabilité :** permet de l’automatisation complète (calculs de TVA, date d'échéance, équilibrage), la gestion des conditions de paiement et l’édition des documents (balance, bilan, compte de résultat, déclaration TVA, Gestion budgétaire…).
  1. Matrice d’évaluation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Etude comparative des ERP | | | | 20/03/2013 |
| Poids: 1=Faible, 3=Moyenne, 5=Forte Importance | | | **Rang** | |
| Familles | **Critères** | **Poids** | **Open ERP** | **ERP Commercial** |
| 1 | **Pérennité de la solution** | **3** | **5** | **5** |
| 2 | **Prix** | **5** | **5** | **1** |
| 4 | **Personnalisation de la solution** | **5** | **5** | **5** |
| 5 | **Solution Business Intelligence** | **5** | **5** | **5** |
| 6 | **Module de comptabilité** | **5** | **5** | **5** |
| Résultat/Total score : | | | **115** | **98** |

*tableau5 : familles de critères pour l'étude des ERP*

* 1. la solution retenue :

L’ERP choisi c’est : Open ERP.

1. RESULTAT de scoring

|  |  |
| --- | --- |
| Catégorie | Outil |
| SGBD | NOSQL/Neo4j |
| IDE | Eclipse |
| Plateforme | Androïde, Ios |
| Langage | Java, HTML5, Java Script (Ajax - JQuery) |
| ERP | Open ERP |
| Composant HTML | CANVAS |

**Cette partie présente les spécifications fonctionnelles ,l’analyse ,la Conception et la réalisation de WIJHATI**

**Chapitre 4** : EXPRESSION DES BESOINS

1. ETUDE de cas
   1. Choix UML

UML (en anglais Unified Modeling Language ou « langage de modélisation unifié ») est un langage de modélisation graphique à base de pictogrammes. Il est apparu dans le monde du génie logiciel, dans le cadre de la « conception orientée objet ». Couramment utilisé dans les projets logiciels, il peut être appliqué à toutes sortes de systèmes ne se limitant pas au domaine informatique.

UML possède plusieurs facettes. C'est à la fois une norme et un langage de modélisation objet. C’est une norme car UML offre la possibilité de s'exprimer clairement : en représentant des concepts abstraits et en limitant les ambiguïtés (parler un langage commun, au vocabulaire précis, indépendant des langages orientés objet). C’est également un langage graphique qui s’appuie sur la technologie objet et les concepts qu’elle véhicule. Il peut donc être utilisé dans différentes méthodes. Enfin, c’est un support de communication, il constitue une base de discussion entre tous les acteurs d’un projet.

* 1. Identification des besoins

On va développer trois applications, la première est un site web ,sous le nom **Live** **map** qui va nous permettre de surveiller l’état des cartes géographiques et corriger les erreurs, la deuxième sera une application mobile Androïde et IPhone, ces deux versions de l’application seront développées en parallèle, la troisième sera sous forme d’un site web officiel de l’entreprise qui va permettre aux utilisateurs de gérer leurs compte aussi à nos client de gérer leurs investissement dans la publicité géo-localisé.

Apres une réunion de notreéquipe,on a trouvé qu’il faut commencer par la première application **Live map** vue son importance et son impact sur le fonctionnement du système en sa globalité.

* **Live** **map** est un logiciel sous forme d’un site web permettant à l’utilisateur qui souhaite jouer un rôle plus actif dans la gestion, la détection et à la correction des erreurs des cartes ainsi il aura la mission de :
* Surveiller nos cartes géographiques.
* Créer, corriger et optimiser la carte géographique soient par un agent de l’entreprise ou bien par un internaute qu’on a jugé apte de le faire.
* Mettre à jour la carte géographique.
* ajouter un rond-point entre des routes.
* Donner des conseils aux autres utilisateurs qui éditent la carte.

Pour avoir ce profil il faut être, comme cité avant, un simple utilisateur ayant conduit avec l’application mobile en marche, et il faut remplir un formulaire d’engagement.

Au début, et pour éviter des problèmes supplémentaires, l’éditeur de carte a le droit de modifier dans un périmètre de 1 km des endroits dans lesquels il a conduit avec l’application en marche.

Avec le temps et les points cumulés, l’éditeur de carte aura plus de pouvoir et pourra ainsi modifier et valider des routes dans un plus grand périmètre.

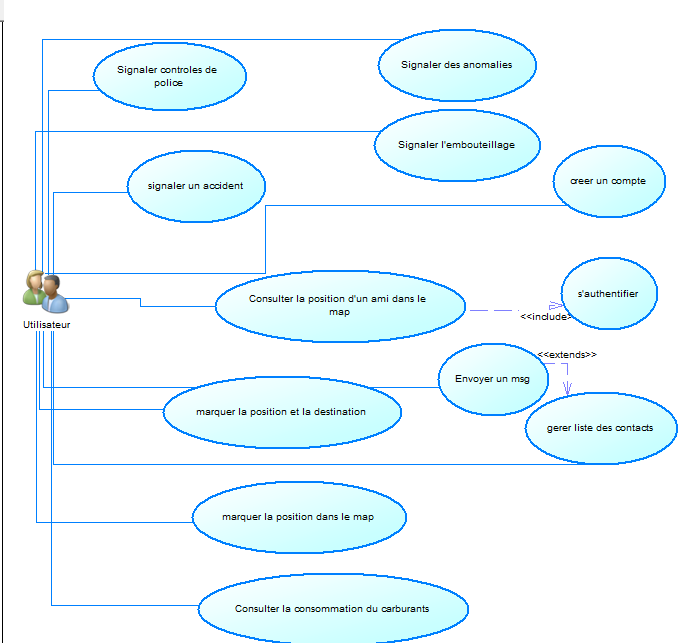
Il peut y avoir plusieurs Managers qui gèrent une même zone géographique, aussi il peut qu’un Manager gère plusieurs zones géographiquement espacées, mais une certaine connaissance de ces zones est importante.

* Les 2 versions des applications mobile devront permettre pour les utilisateurs de :
* Chercher des adresses.
* Signaler un radar fixe
* Signaler un danger ou des problèmes.
* Discussion instantanée avec les véhicules.
* Affichage de station d’essence le plus proche ainsi que le prix de carburant.
* Visualiser le chemin le plus optimale.
* Gérer le Coin de Publicité à l'unité ou en série pour les entreprises les plus proches ou occasionnelles.
* Site web officiel de l’entreprise **WIJHATI** se fera pour :
* Télécharger l’application mobile
* Télécharger le guide de l’utilisation
* Renseigner le questionnaire de feed-back.
* signaler une anomalie de l’application mobile.
* Contacter le support Client.
* Renseigner les sites de nos annonceurs.
* Suivre l’impact des publicités des entreprises

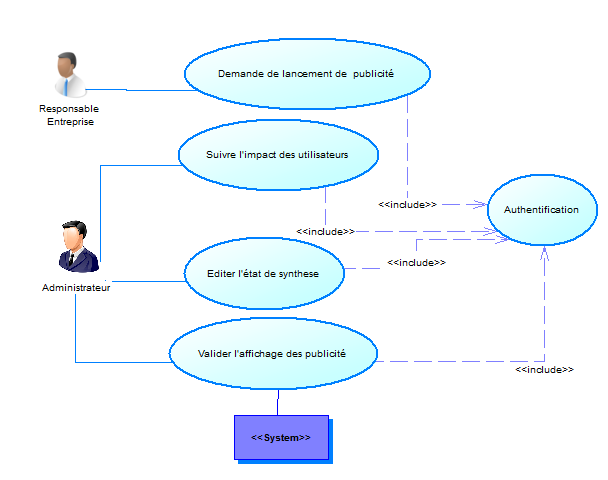
* 1. Diagramme de cas d’utilisation

Les cas d'utilisation permettent de représenter le fonctionnement du système vis-à-vis de l'utilisateur : c'est donc une vue du système dans son environnement extérieur.

Ce 1er diagramme de use case concerne l’utilisation de l’application GPS mobile **WIJHATI** par les usagers :



Le 2 éme use case concerne l’utilisation de l’application web back Office qui permet de gérer les publicités des entreprise :



* 1. Documentation de quelques cas d’utilisation

|  |  |
| --- | --- |
| Description de l’UC | Gérer la liste des contacts |

|  |  |
| --- | --- |
| Acteur principal | Utilisateur |
| Parties prenantes |  |
| Pré conditions | Authentification |
| Post conditions | Ajouter un utilisateur à la liste des contacts |
| Scénario principal | 1°) afficher le répertoire des contacts  2°) Ajouter un nouveau contact  3°) le système affiche tous les contacts Wijhati |

|  |  |
| --- | --- |
| Description de l’UC | Consulter la position d’un ami |

|  |  |
| --- | --- |
| Acteur principal | Utilisateur |
| Parties prenantes | Utilisateur |
| Pré conditions | Authentification |
| Post conditions | Consultation de la position de l’ami utilisateur de Wijhati avec succès. |
| Scénario principal | 1°) Accéder à la liste des contacts  2°) sélectionner l’ami de la liste des contacts.  3°) le système affiche la position de l’ami dans le map  4°) afficher la position de l’ami dans la carte |
| Scénarios alternatifs | 2°) sélectionner l’ami de la liste des contacts.  2.1°) l’ami n’existe pas dans la liste des contacts.  4°) afficher la position de l’ami dans la carte.  4.1°) l’ami n’est pas disponible. |

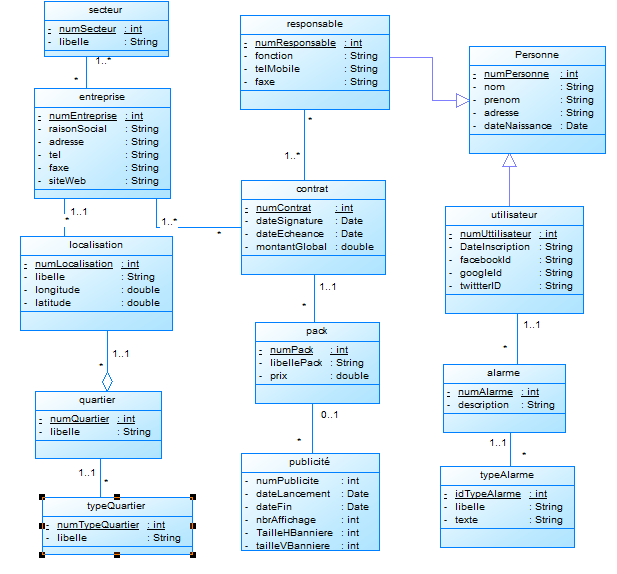
|  |  |
| --- | --- |
| Description de l’UC | Marquer la position dans le map |

|  |  |
| --- | --- |
| Acteur principal | Utilisateur |
| Parties prenantes | Aucun |
| Pré conditions | Aucun |
| Scénario principal | 1°) afficher la carte  2°) Se positionner sur l’endroit actuel.  3°) marquer les infos sur la position.  4°) La position est marqué dans la carte. |
| Scénarios alternatifs | Pas de scénarios alternatifs. |

|  |  |
| --- | --- |
| Description de l’UC | Signaler un policier |

|  |  |
| --- | --- |
| Acteur principal | Utilisateur |
| Parties prenantes | Aucun |
| Pré conditions | Aucun |
| Scénario principal | 1°) afficher la carte  2°) Se positionner sur l’endroit actuel.  3°) choisir l’icône du policier.  4°) Les utilisateur sont notifiés qu’il y’a un policier dans la position marqué. |
| Scénarios alternatifs | Pas de scénarios alternatifs. |

* 1. Diagramme de classe :



**Conclusion :**

|  |  |
| --- | --- |
|  |  |
|  |  |