Cahier des Charges

Projet WEB

Bastien

Dmitrij

RILA 15

2016

Versionning

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Date** | **Auteur** | **Validé par** | **Sujet** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Sommaire

**Tapez le titre du chapitre (niveau 1)1**

Tapez le titre du chapitre (niveau 2)2

Tapez le titre du chapitre (niveau 3)3

**Tapez le titre du chapitre (niveau 1)4**

Tapez le titre du chapitre (niveau 2)5

Tapez le titre du chapitre (niveau 3)6

# **Introduction**

Dans le cadre des technologies web, il nous a été demandé de développer une application mobile HTML 5 permettant d’afficher une carte géographique, de déterminer la position actuelle, d’enregistrer et de partager des POI (les alertes, les incidents, les évènements) sur une carte interactive. Afin de s’orienter vers une application de type Waze, la mise en place de la trajectographie ainsi que la gestion des alertes seraient un plus.

Nous avons choisi de développer une solution multiplateforme mobile en se basant sur le Framework ionic comprenant javascript, mysql et php.

Pour cela nous avons effectué des recherches sur les différentes technologies présentes sur le marché comme les développements native et multiplateforme avec différents types de framework et de langages. Nous avons ensuite fait une analyse du projet en faisant des mockup ainsi que l’architecture de la base de données pour enfin nous pencher sur le développement du projet.

Nous tâcherons ici d’effectuer une analyse fonctionnelle puis technique du projet, d’expliquer le choix des technologies utilisées, de décrire le développement du logiciel ainsi que les tests d’intégration et enfin la mise en production de l’application.

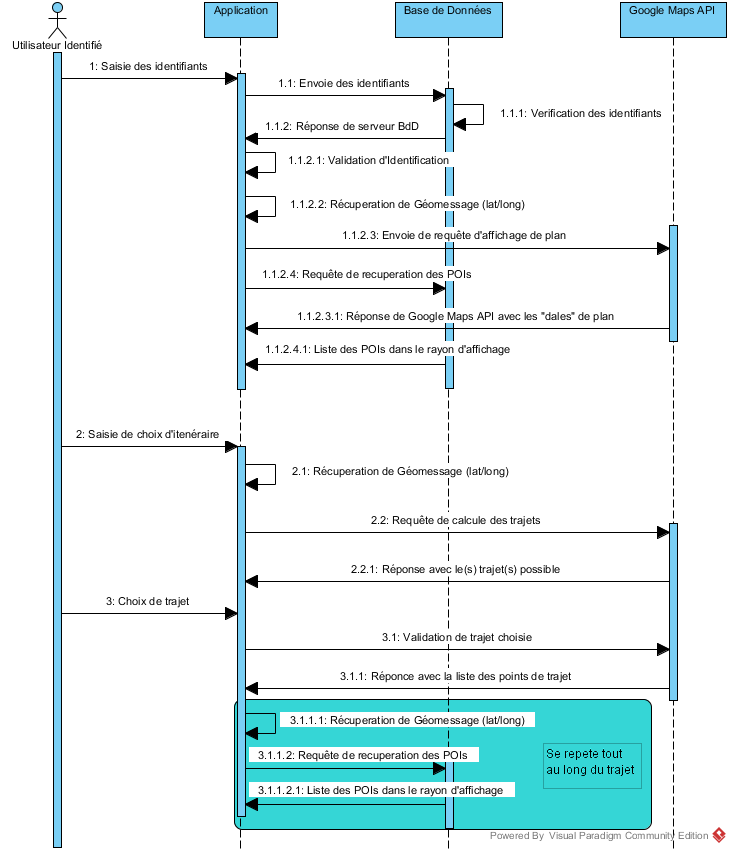
# **Les objectifs**

Les objectifs de ce projet sont les suivants :

* Définir les spécifications techniques et l’architecture logicielle de la solution (en UML)
* Développer une application HTML5 orientée MVC
* Mettre en place un comportement actif en JavaScript (JQuery, AngularJS, Ionic)
* Implémenter l’utilisation d’un WebService côté client
* Intégrer l’étude ergonomique des IHMs dans le développement

# **Analyse fonctionnelle du projet**

## Diagramme de séquence



## User-cases

Partie utilisateur

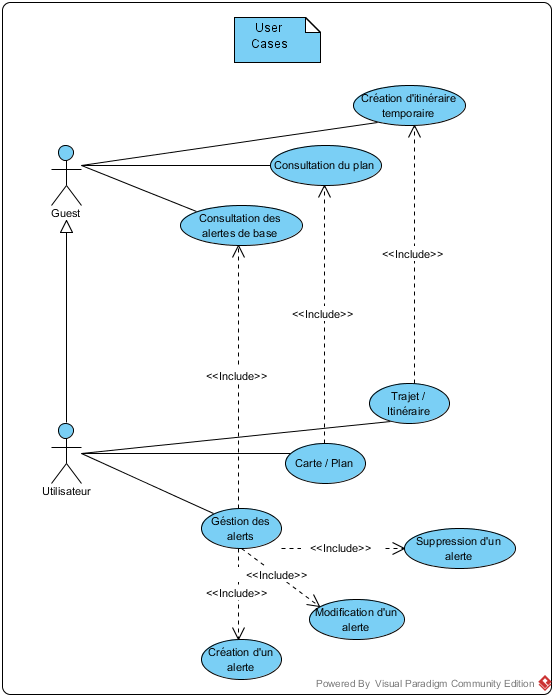
* Consultation de carte
* Possibilité de création de compte
* Gestion des points d’intérêts
* Création des trajets
* Sauvegarde des paramètres d’affichage des points d’intérêts

Utilisateur non enregistré :

* Affichage du plan
* Affichage des points d’intérêts de base
* Possibilité de création d’un compte utilisateur

Administrateur :

* Identification
* Gestion des comptes utilisateurs
* Gestion des points d’intérêts globale (ajout, suppression, modification, affichage du plan, etc.)

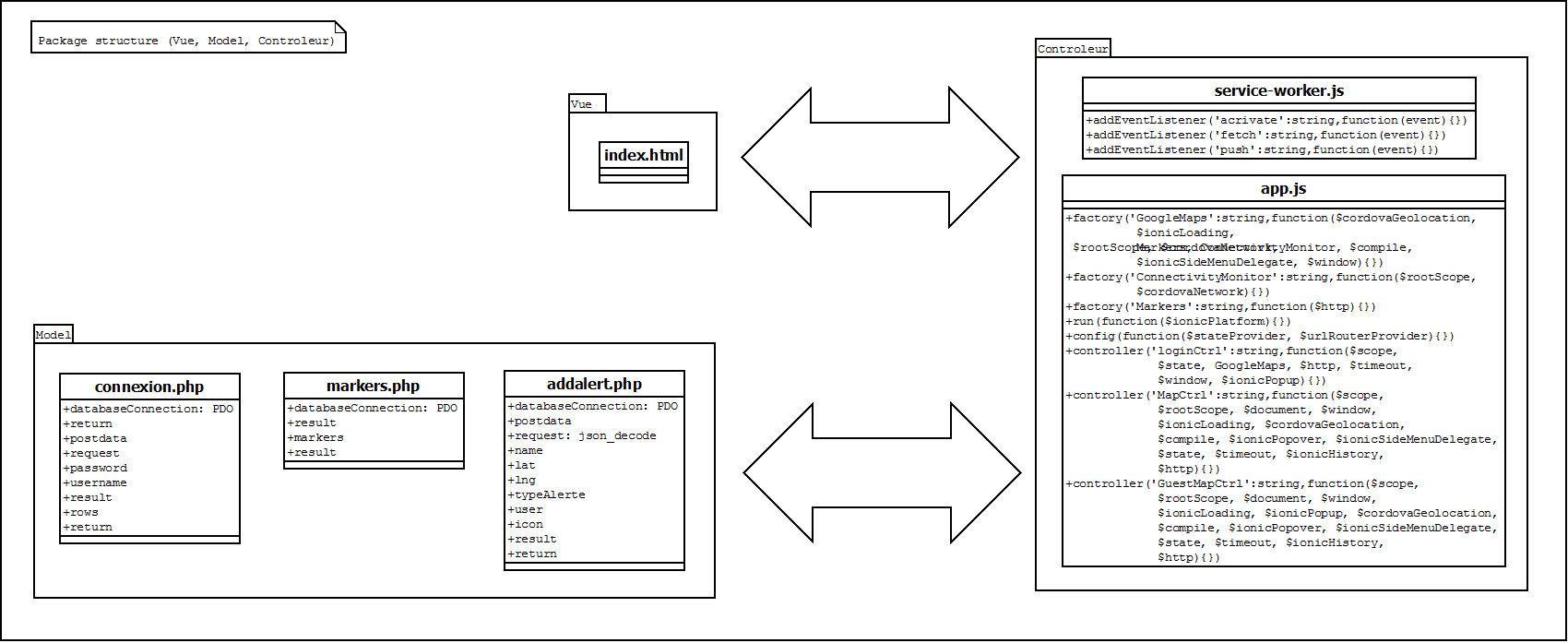


# **Analyse technique du projet**

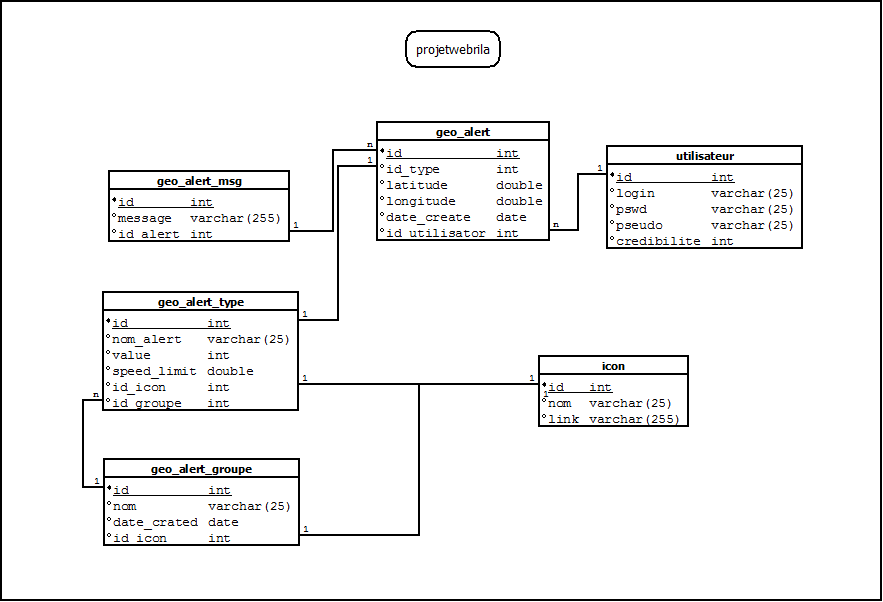
## Diagramme des classes

Diagramme des objets :

* Objets plan
* Objets point d’intérêt POI
* Objet utilisateur
* Objet parcours



## Diagramme de la Base de Données



## UML

## Recettage des tests et tests unitaire

Les tests unitaires sont basés sur les cas d’utilisation.

* Bon affichage du plan
* Ban affichage du parcours
* Affichage et mise à jour des points d’intérêts tout au long du parcours
* Refus d’ajout de nouveaux points d’intérêts en tant qu’utilisateur non identifié
* Ajout de nouveau point d’intérêt en tant qu’utilisateur identifié
* Création d’un compte utilisateur
* Modification du compte utilisateur
* Suppression d’un compte utilisateur

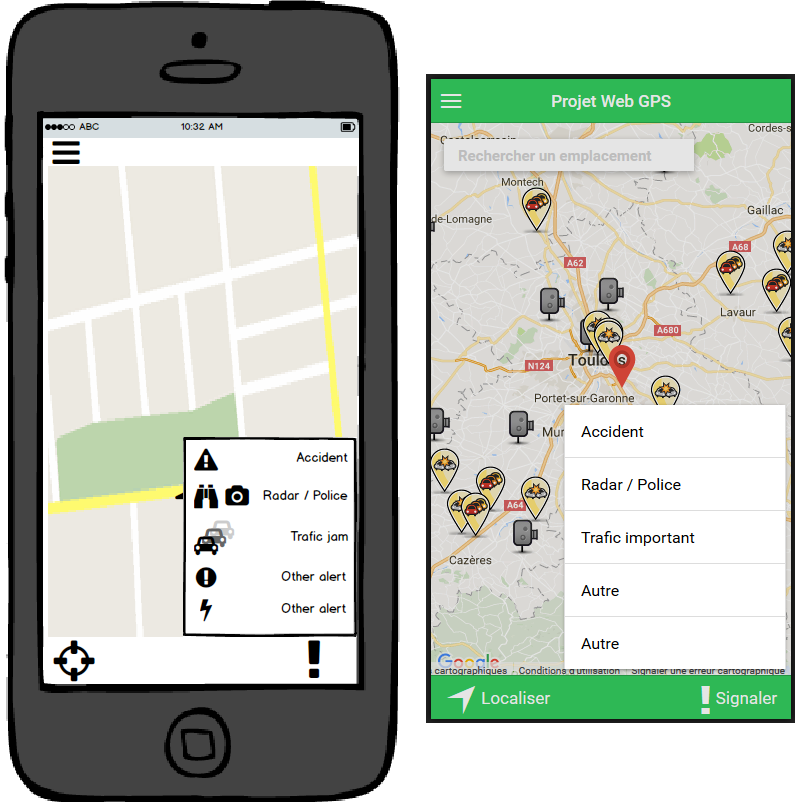
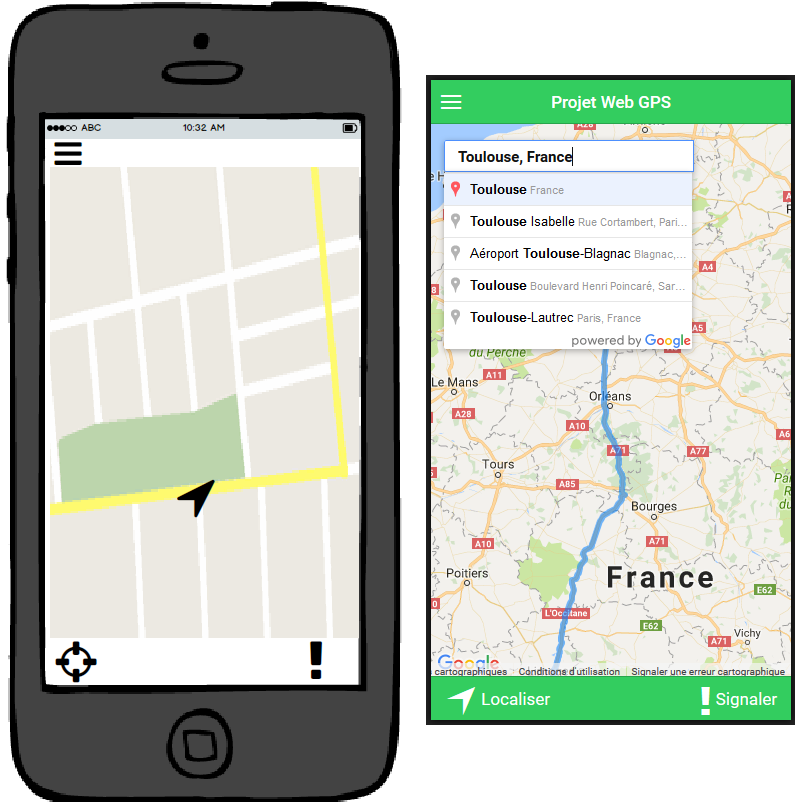
# **Choix des technologies**

Technologies préférables :

Côté serveur, il est possible de réaliser le projet en HTML, PHP ou en NODE.js. Côté client, l’utilisation de jQuery, d’AngularJS ou de toute autre librairie est possible.

**Mockups des IHMs**

Les Mockups vers IHMs de produit fini :

1. Identification : 
2. Les alertes : 
3. Choix de trajet : 

# **Développement du logiciel**

Le développement du logiciel est composé de deux parties :

1. Mise en place et configuration d’un serveur web
2. Développement d’un logiciel multiplateforme avec intégration de l’API Google maps

# **Tests d’intégration**

Se sont des tests de buggage sur du matériel physique

# **Mise en production**

Déploiement du logiciel et formation

Documentation

Critère de notation :

Ce projet vise essentiellement à mettre en œuvre les technologies modernes du Web (HTML5, CSS3, jQuery, Ajax, WebServices, etc…) d’un point de vue « expert technique » : conception, modélisation, développement, tests, documentation, etc…

* Capacité à présenter une solution, d’un point de vue fonctionnelle et technique
* Capacité d’auto-organisation, réalisation d’un PBS/WBS
* Qualité des livrables logiciels, respect des bonnes pratiques en terme d’AGL
* Pertinence des choix techniques, argumentation
* Capacité à s’exprimer en public, et à faire une présentation synthétique dans le cadre d’un projet
* Etude ergonomique, réalisation d’une maquette

Les points discutés avec Rémi Bello le 09/09/2016 :

* API client/serveur – temps réel – synchro données
* Trajectographie
* Mode Off-line – Localisation – Local storage
* IHM mobile – responsive – perspective
* Gestion des alertes :
  + Cycle de vie
    - Sondage (on/off)
    - Time-out
    - « battement du cœur »
  + Localisation et fusion des alertes
  + Pop-ups d’alertes (bonnes alertes en bon moment)
* Documentation d’architecture :
  + Diagramme des composants
  + Diagramme d’architecture
  + Choix techniques
    - POC (Proof Of Concept)
* AGL/Branches