

Projet WEB

Bastien PENETRO
Dmitrij POPOV

Promotion RILA 15
©CESI, Campus de Toulouse
2016

Versionning

Version	Date	Auteur	Validé par	Sujet
V0.1	14/10/2016	D. POPOV	B. PENETRO	Création du document ; mise en place de la structure ; MockUps
V0.2	15/11/2016	B. PENETRO	D. POPOV	Choix des technologies ; AGL
V0.3	09/12/2016	B. PENETRO	D. POPOV	Diagramme des packages/objets ; Diagramme de la base de données
V0.4	14/01/2017	D. POPOV	B. PENETRO	Mise en forme final
V1.0	18/01/2017	D. POPOV, B. PENETRO	D. POPOV, B. PENETRO	Dernières corrections

Table des matières

Versionning	2
Table des matières	3
Introduction	4
Les objectifs	4
Analyse fonctionnelle du projet	5
Diagramme de séquence	5
User-cases	6
Analyse technique du projet	7
Diagramme des packages (structure MVC)	7
Diagramme de la Base de Données	7
Recettage des tests et tests unitaire	8
Choix des technologies	8
Développement du logiciel	11
Tests d'intégration	11
Mise en production	11

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 événements) 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

Le projet a pour but le développement d'une application mobile HTML 5 permettant d'afficher une carte géographique, de déterminer la position actuelle, d'enregistrer et de partager des POI sur une carte interactive.

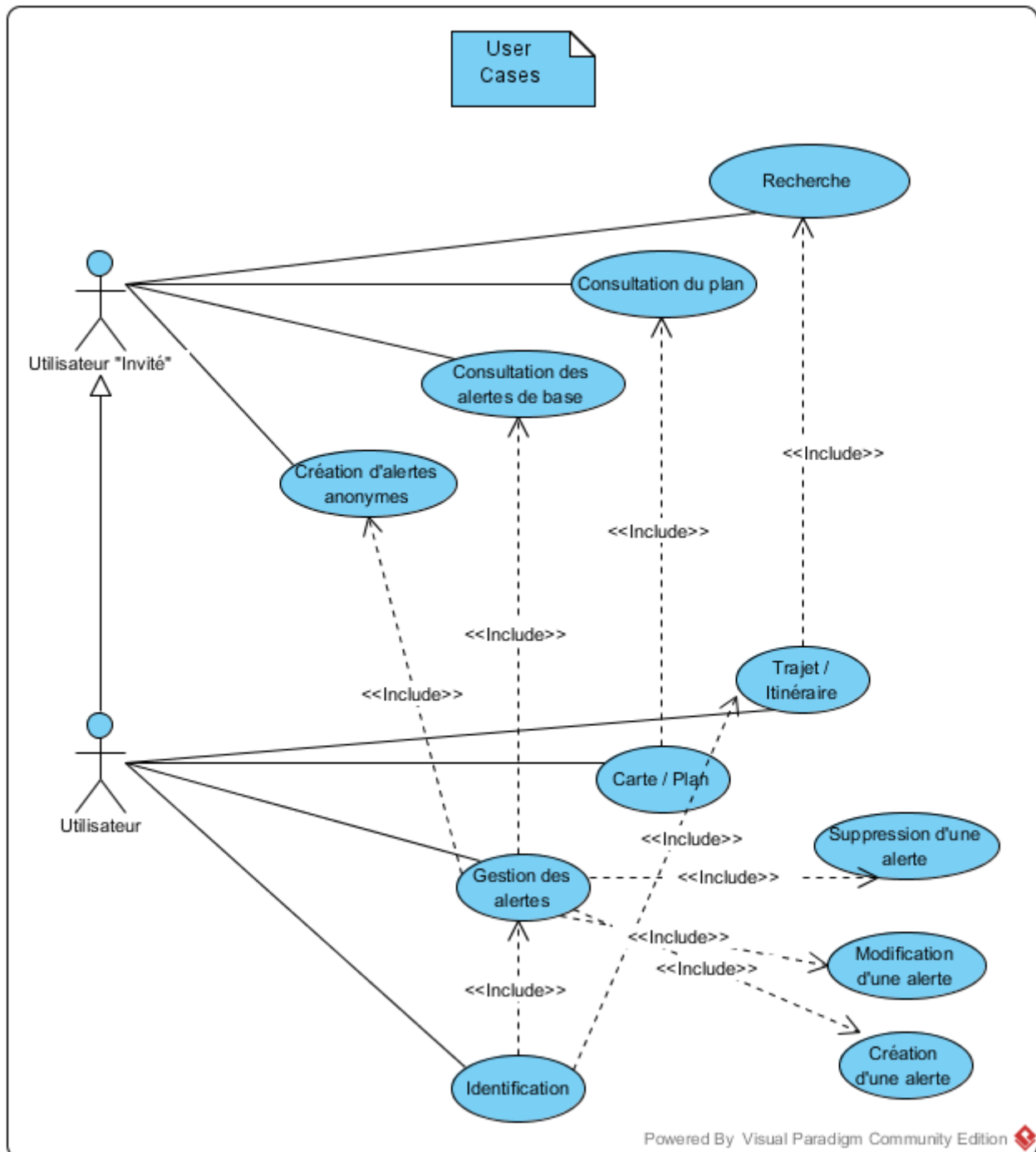
Les objectifs de ce projet sont les suivants :

- 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

User-cases

Pour ce qui est de l'utilisateur invité non enregistré, il a la possibilité de consulter de plan, de voir les alertes pré-définies, de créer les alertes en mode anonyme, de faire la recherche de destination ainsi que de créer un itinéraire.

Pour ce qui est de l'utilisateur, ce dernier hérite de toutes les fonctionnalités de l'utilisateur invité non enregistré avec l'ajout de fonctionnalités supplémentaires comme l'identification et la gestion des alertes.



Analyse technique du projet

Diagramme des packages (structure MVC)

Le logiciel va être construit en suivant les règles du métier Modèle Vue Contrôleur. Chaque partie de cette structure est constituée d'objets dont la vue se met à jour selon le choix de l'utilisateur en s'appuyant sur le contrôleur qui utilise les méthodes des objets dans le modèle.

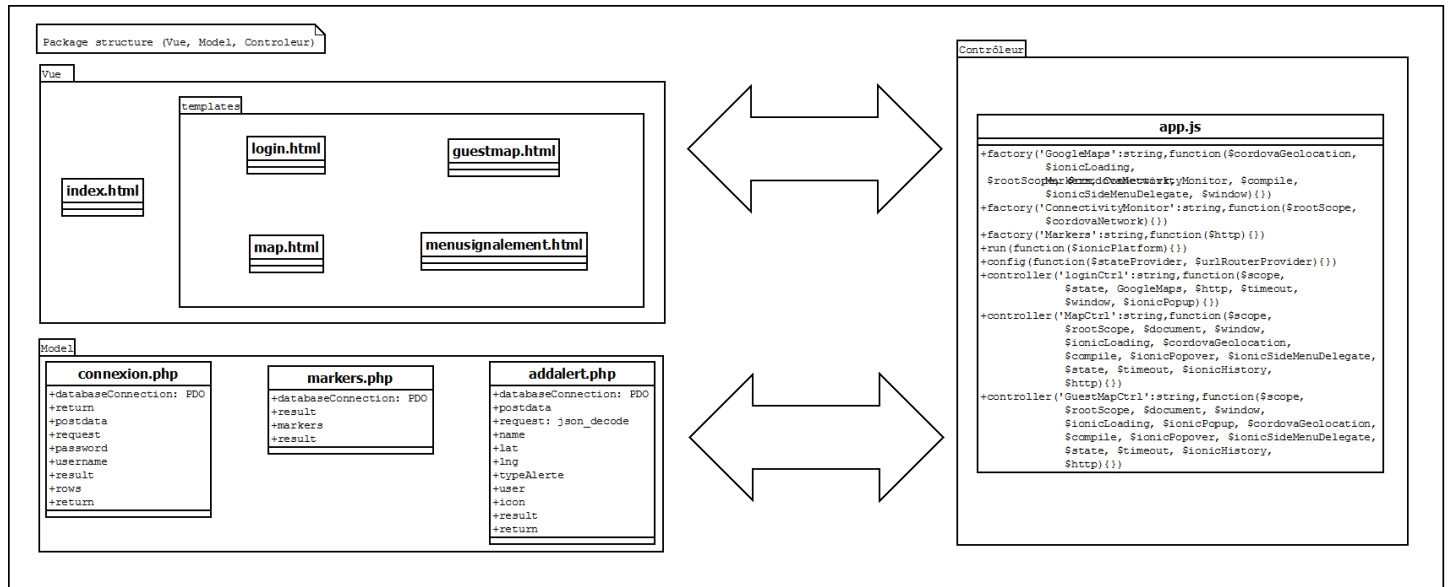
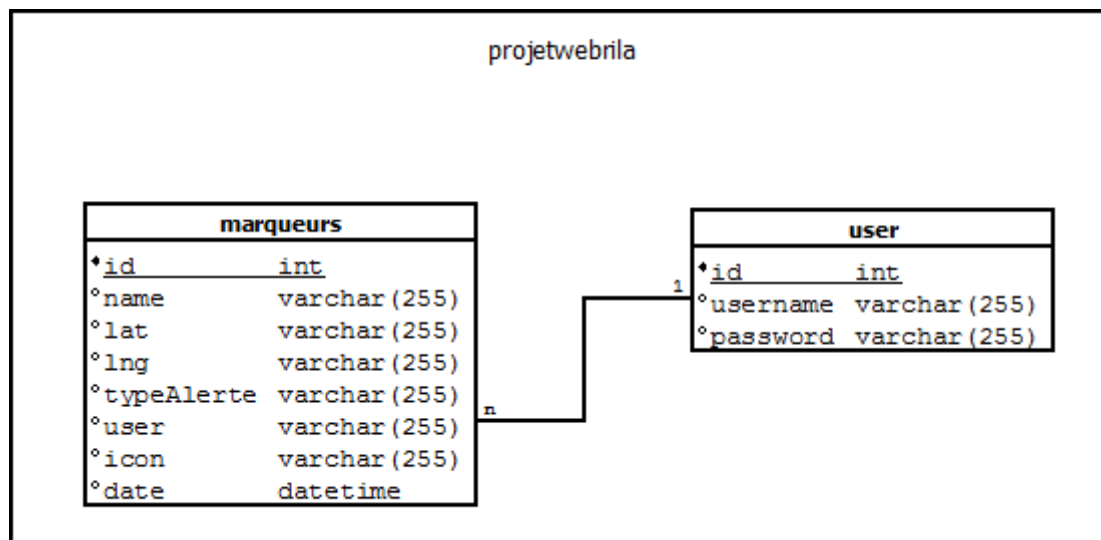


Diagramme de la Base de Données

La base de données est constituée de six tables avec des attributs. Ces tables sont : user et marqueurs. Elles sont utilisées pour la sauvegarde des données.



Recettage des tests et tests unitaire

Les tests unitaires sont basés sur les cas d'utilisation. Il existe plusieurs fonctionnalités qui doivent être testées comme :

- Le bon affichage du plan
- Le bon affichage du parcours
- L'affichage et la mise à jour des points d'intérêts tout au long du parcours
- L'ajout de nouveaux points d'intérêt en tant qu'utilisateur identifié ou anonyme

Tableau User Cases vers Tests Unitaires

User Case	Sujet	Unit Test
UC 0	Affichage de la forme d'identification	UT 0
UC 1	Affichage du plan en mode Invité	UT 1
UC 2	Affichage du menu d'ajout d'une alerte par type	UT 2
UC 3	Affichage des alertes	UT 3
UC 4	Création d'une alerte anonyme	UT 4
UC 5	Création de l'itinéraire	UT 5
UC 6	Affichage du trajet	UT 6
UC 7	Identification d'un utilisateur	UT 7
UC 8	Affichage du plan en mode Identifié	UT 8
UC 9	Création d'une alerte en mode Identifié	UT 9

Choix des technologies

En ce qui concerne la partie serveur, nous avons choisi le serveur apache avec PHP et la base de données en MySQL. Ces technologies ont été choisies pour la compatibilité entre PHP et MySQL grâce à l'outil PDO (PHP Data Object), qui sont gratuits et open source ainsi que très bien documentés avec une forte communauté active.

Pour ce qui est de la partie client, nous avons choisi trois frameworks (Ionic, AngularJS et CORDOVA), une API (Google Maps API) ainsi que deux langues de développement web : PHP et JavaScript (Angular).

Pour ce qu'il en est des outils de développement ; tout le développement se passe sur les machines avec le système d'exploitation Windows et les serveurs NODE.js et WAMP. Pour que cela fonctionne dans NODE.js il est nécessaire d'ajouter des services et des bibliothèques supplémentaires comme BOWER, CORDOVA ou encore IONIC en utilisant le NPM (Node Package Manager) qui est le gestionnaire des paquets officiel de Node.js.

Mockups des IHMs

Dans les Mockups, nous allons prototyper les IHMs de notre futur logiciel. Ici vous pouvez voir la cohérence entre les MockUps et les IHMs finaux.

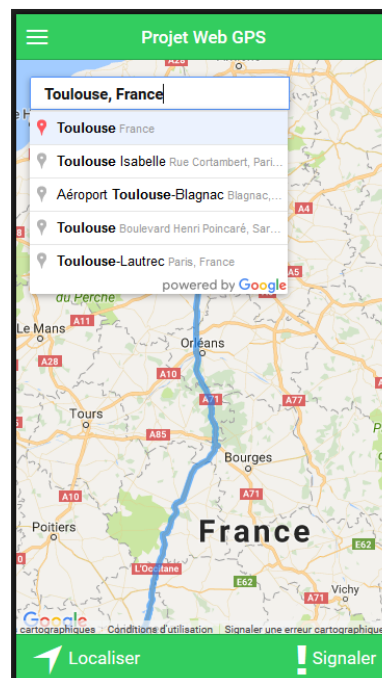
Les Mockups vers IHMs de produit fini :



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 au client et présentation du produit.