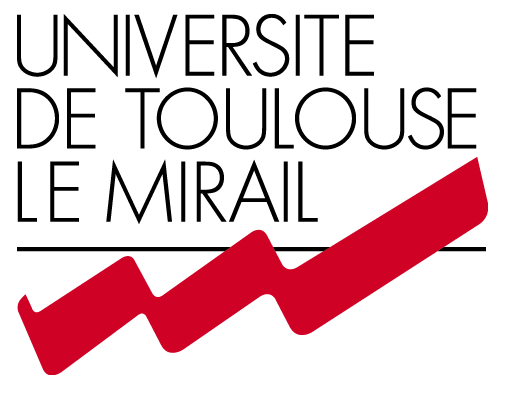
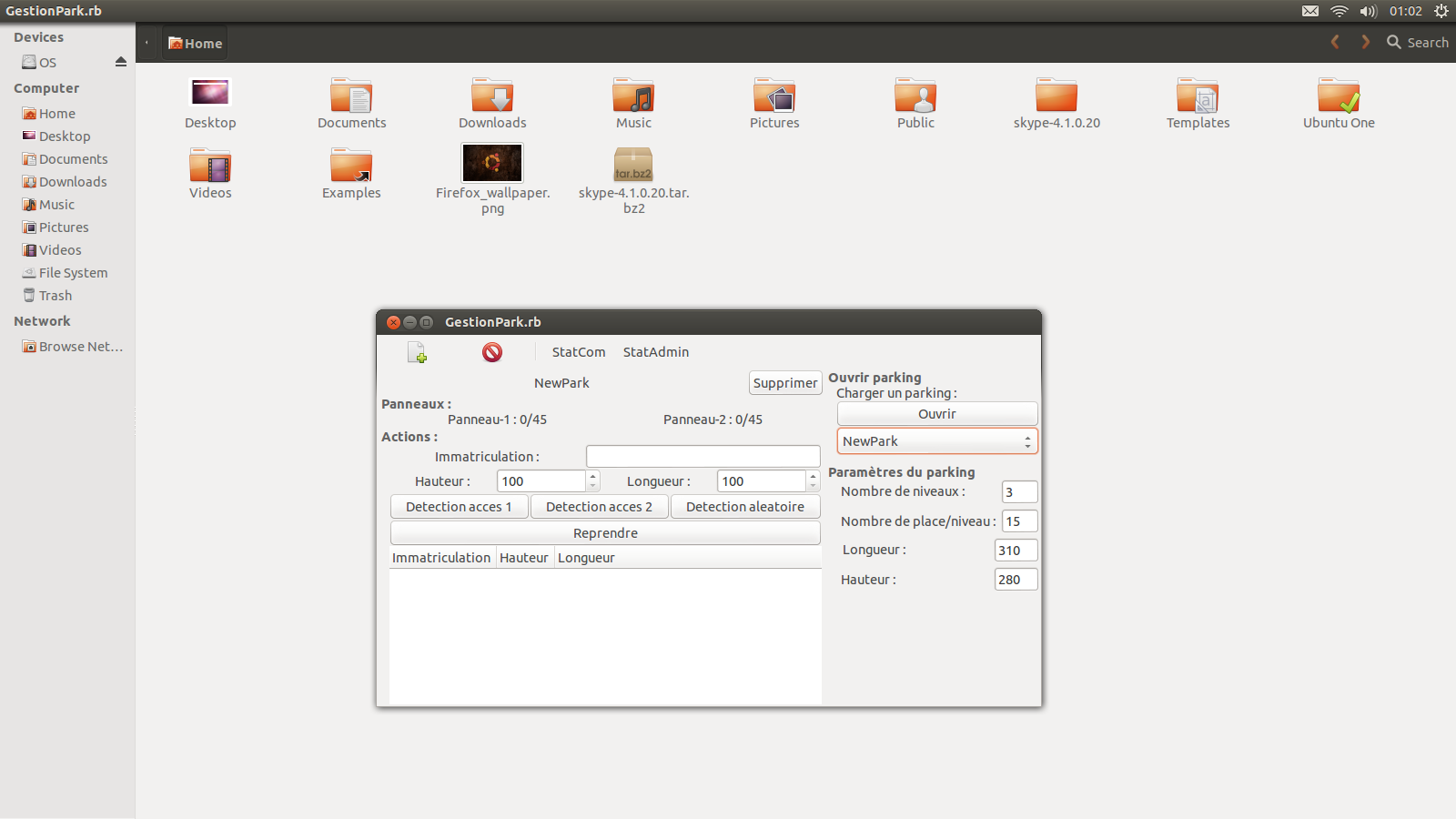
** Université TOULOUSE II – Le Mirail 2012-2013**

**Licence MIASHS**

|  |
| --- |
| Rapport de fin de projet |
| *Application DreamPark* |
| **10/01/2013** | | |



|  |
| --- |
| **David EGLEM**  **Jérémy PINTO** |

Table des matières

[Conception 3](#_Toc345584673)

[1. Diagramme des cas d’utilisation 3](#_Toc345584674)

[2. Diagramme de classe 4](#_Toc345584675)

[3. Diagramme de séquences 5](#_Toc345584676)

[1. Garer une voiture 5](#_Toc345584677)

[2. Reprendre une voiture 6](#_Toc345584678)

[Implémentation 7](#_Toc345584679)

[Graphisme 8](#_Toc345584680)

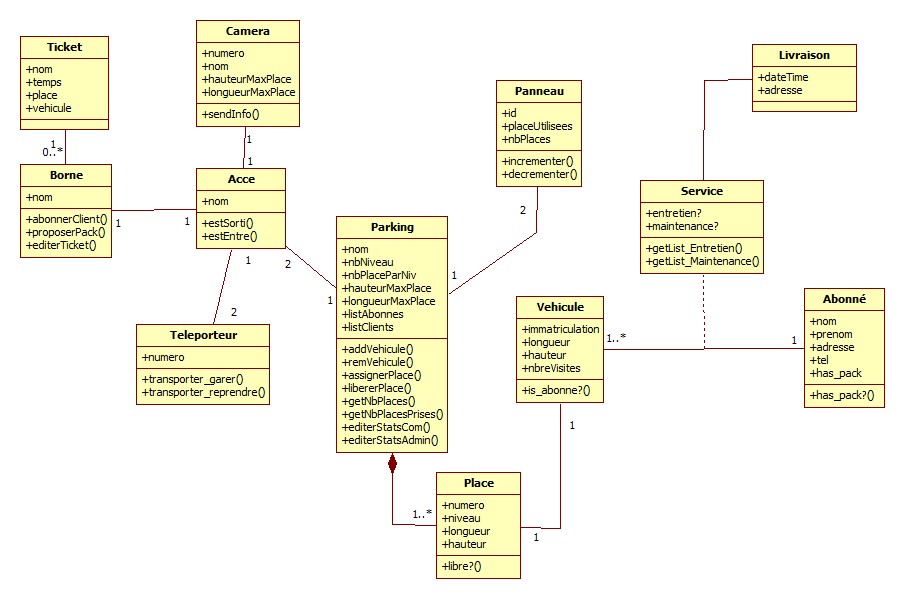
Conception

Dès l’engagement du projet nous avons toujours voulus coller à l’aspect de simulation et d’utilité que notre application devait avoir.

Ainsi en découle nos choix de conception et d’implémentation vers une simulation réelle.

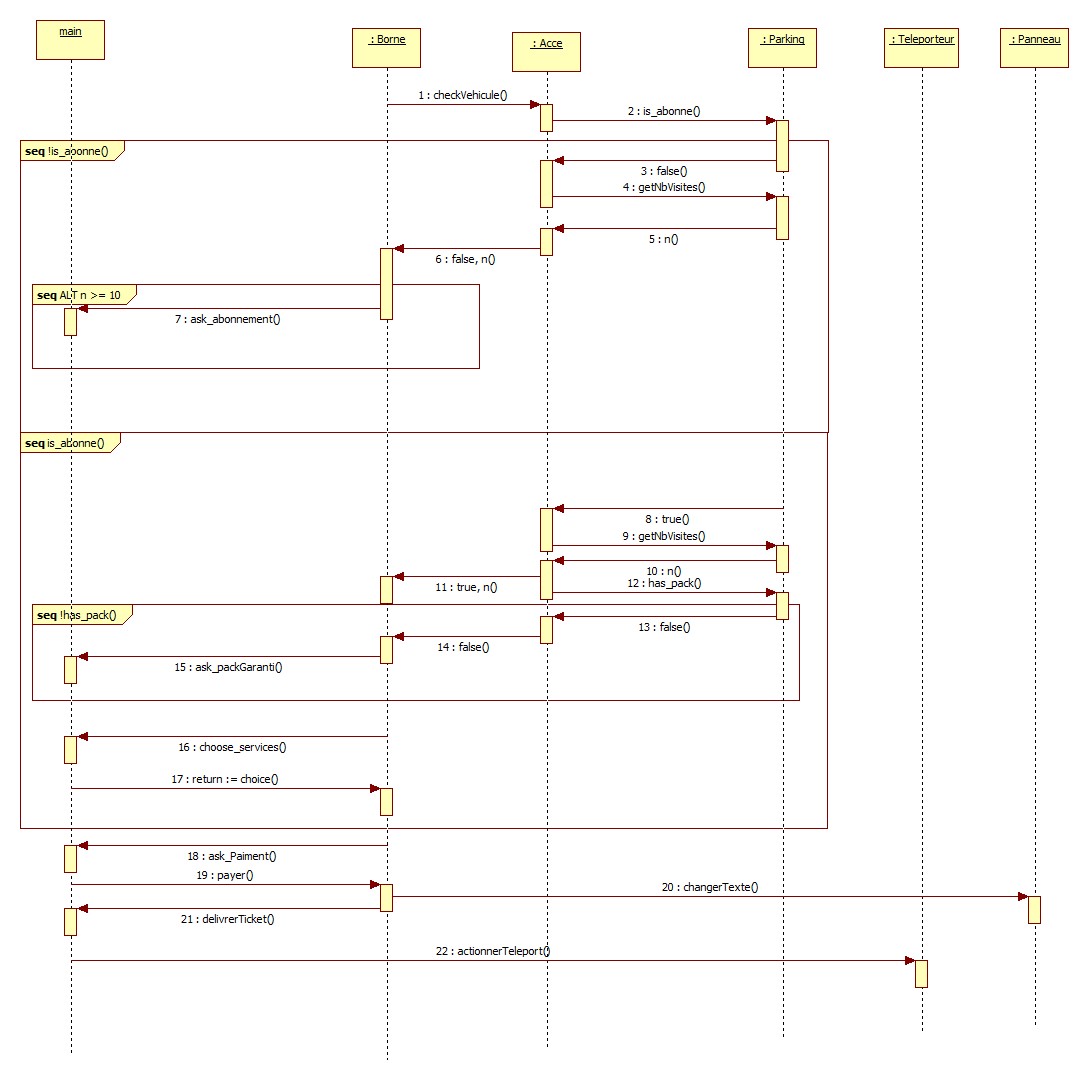
## Diagramme des cas d’utilisationC:\Users\Pinto\Dropbox\L3\Ruby\DreamPark\Conception\UseCaseDiagram.jpg

## Diagramme de classe

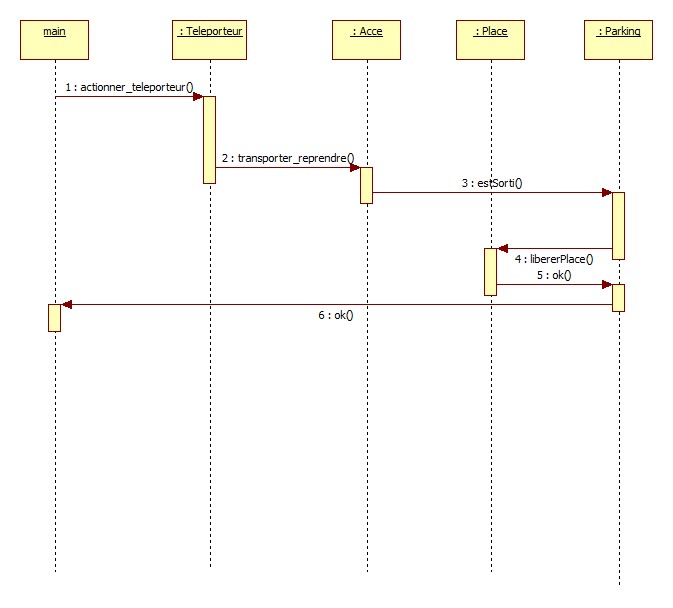


## Diagramme de séquences

### Garer une voiture



### Reprendre une voiture



Implémentation

Nos choix d’implémentation ce sont fait en respectant les règles instaurés par le projet ainsi qu’en respectant la volonté d’installation une maintenabilité et une robustesse dans l’application.

Ainsi nous avons choisis le « pattern » MVC, Modèle Vue Contrôleur afin de développer notre application ce qui permet la gestion du modèle et de la vue indépendamment et ainsi octroi une meilleur maintenabilité du code.

Notre développement a aussi profité de cycle dirigé par les tests, ainsi la rédaction des tests s’est faite avant pour mieux prévoir et comprendre ce que l’on attendait de l’application ainsi que nous permettre de corriger et tester les fonctionnalités mise en place au fur et à mesure.

Graphisme

Pour effectuer la vue de notre « pattern » nous avons utilisé la librairie ‘gtk’ adapté dans sa version 2 pour le langage ruby, cette librairie permet de faire une interface homme-machine d’un design plutôt attrayant et ergonomique, de plus elle s’adapte à la charte graphique sur laquelle elle est instanciée.

Nous avons aussi utilisé le logiciel « Glade », dans sa version 2, afin de réaliser le design de nos interfaces, ce logiciel étant pris en charge par certaines classes de la librairie il était tous naturel de l’utilisé à l’occasion de notre développement. Il faut aussi préciser que pour la barre d’outils et de menus nous avons utilisé le module « uimanager » du langage, ce module est indépendant du logiciel « Glade ».

