

Conseil 7

Joël Collinet

Félix Erb

Célia Kocher

Vincent Moitry

Adam Rimelé

Carlo Spiga

Maxime Zimmer

4.A.1 Dossier d'analyse métier

Projet Acrobatt 2014

Octavio

Tuteur : Pierre Kraemer

SOMMAIRE

Introduction

I. Modélisation Métier

a) Diagramme de Cas d'Utilisation

b) Modèle Métier

i. Diagramme de Classes d'Analyse

c) Diagramme de Séquence

II. Modèle de la Base de Données

a) Modèle Conceptuel de Données

b) Dictionnaire de Données

Conclusion

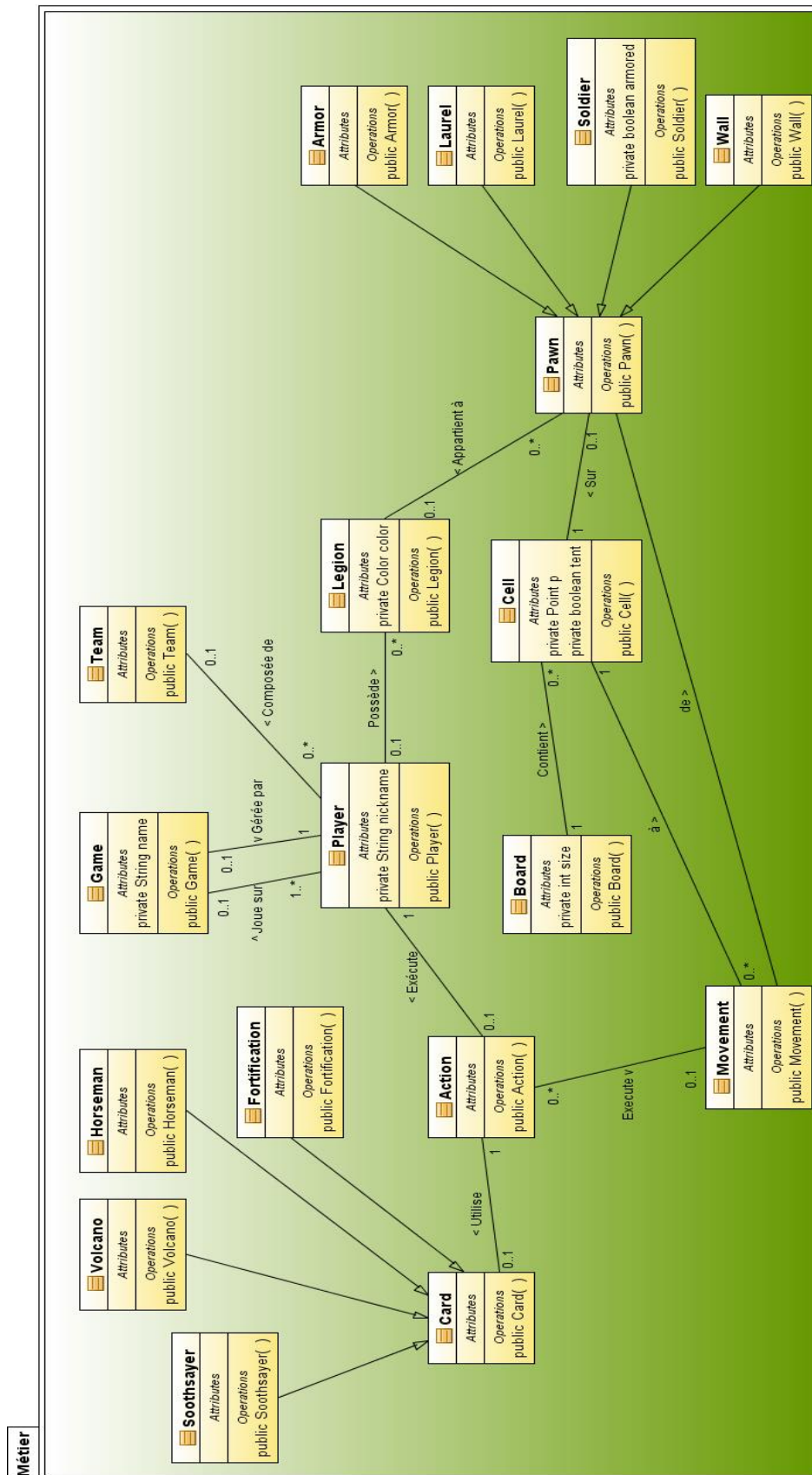
Introduction

Dans le cadre du projet, il est nécessaire d'évaluer avec précision le fonctionnement de l'ensemble de l'application. Il s'agit d'un jeu de plateau dans lequel il est susceptible de trouver bon nombre de classes (pour définir les pions, les déplacements, par exemple).

Une phase d'analyse est fortement recommandée dans laquelle divers diagrammes sont conçus. Ces diagrammes permettent de schématiser aisément les divers liens établis entre les classes définies dans le code.

b) Modèle métier

On peut ensuite définir la liste des classes qui sont utilisées dans le code, et les liens existants entre ces classes. Quel attribut peut caractériser une légion ? Combien de types de cartes existent ? Les différentes classes pouvant intervenir sont analysées, et leurs interprétations sont établies.



Conseil 7

Action : un coup (mouvement ou carte) joué.

Armor : une case contenant une armure.

Board : le plateau sur lequel est joué la partie.

Card : les cartes pouvant être jouées, chaque type de carte hérite de la classe Card.

Cell : les cases du plateau.

Fortification : la carte érigeant des murs adjacents à la case sur laquelle la carte a été activée.

Game : une partie réalisée sur un plateau avec des joueurs.

Horseman : la carte permettant à un soldat sur la case sur laquelle la carte a été activée de jouer deux coups.

Laurel : une case contenant un laurier.

Legion : les groupes de soldats contrôlés par un joueur.

Movement : le déplacement d'un élément sur le plateau.

Pawn : les pions, chaque pion hérite de la classe Pawn.

Player : un joueur.

Soldier : une case contenant un soldat.

Soothsayer : la carte permettant à un joueur de donner son coup après celui des autres.

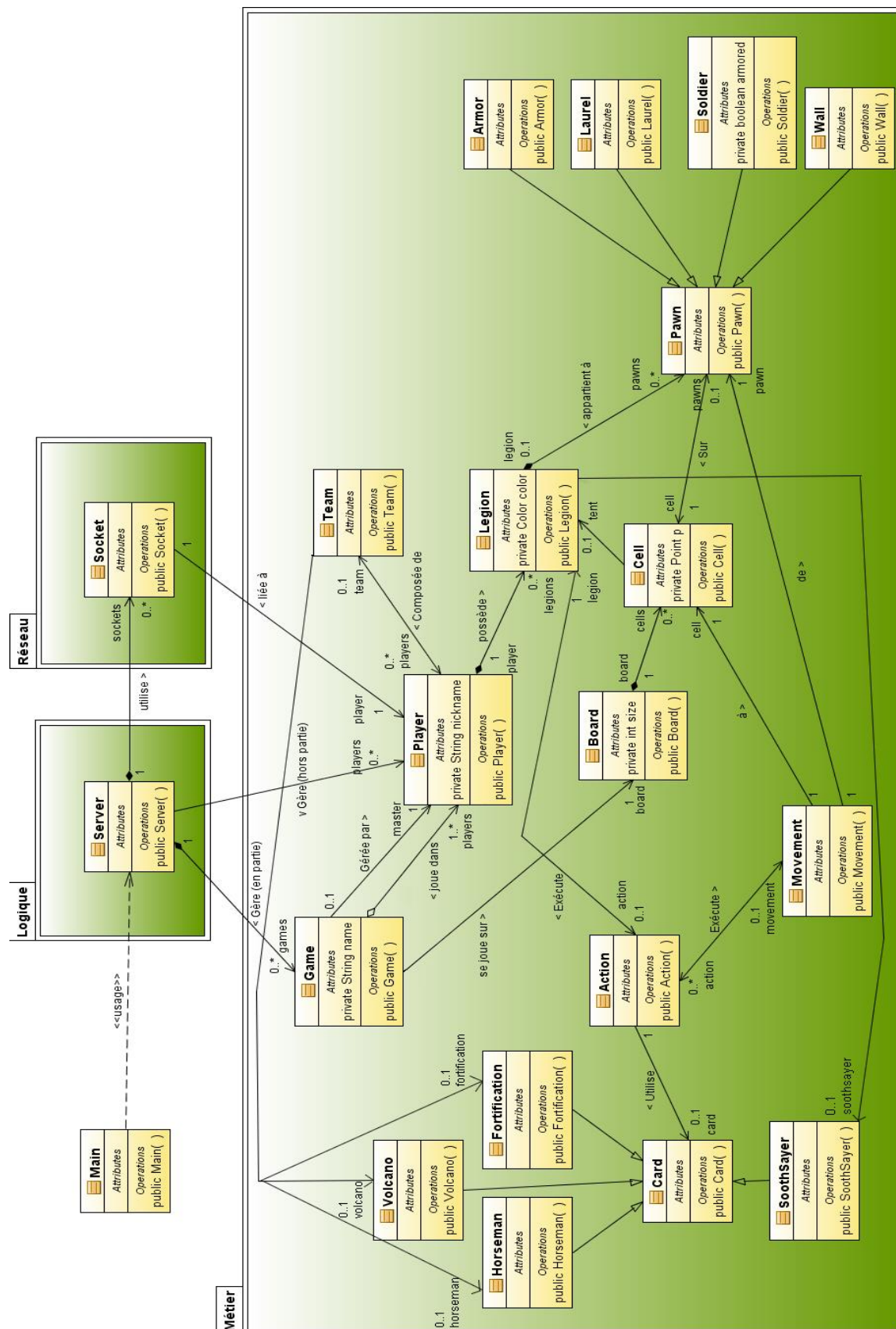
Team : l'équipe d'appartenance d'un joueur.

Volcano : la carte détruisant les soldats placés autour de la case sur laquelle la carte a été activée.

Wall : une case contenant un mur.

Conseil 7

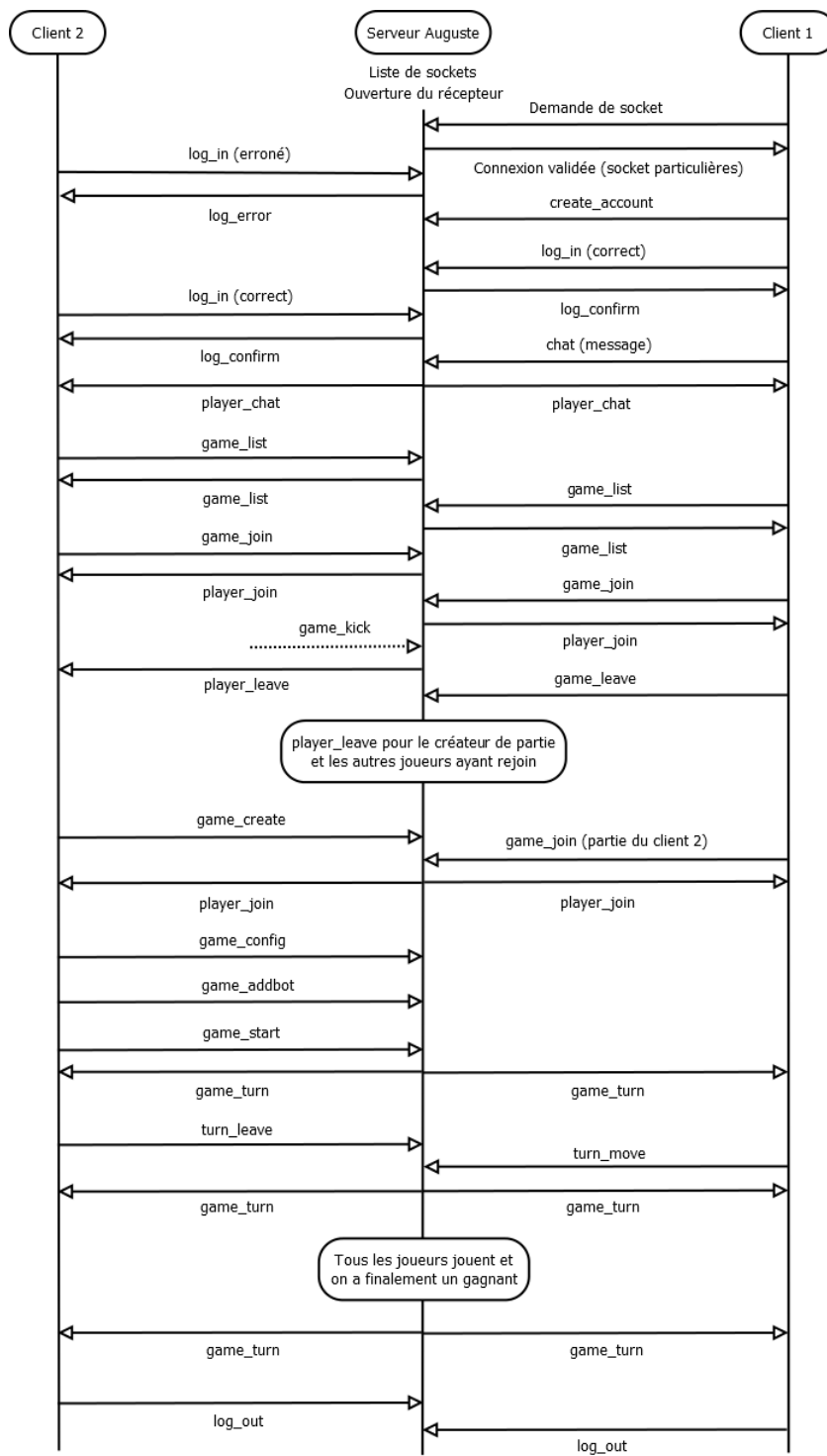
Il est ensuite nécessaire de regrouper les classes au sein de divers packages. Ici, l'interface principale n'est incluse dans aucun package. La logique serveur et les communications font chacune l'objet d'un package. Le reste des classes est regroupé dans un seul et unique package de classes métier.



ii. Diagramme de séquence

Pour pouvoir visualiser au mieux le déroulement d'une partie, un diagramme de séquence est réalisé. Les noms des commandes JSON sont utilisés pour matérialiser au mieux chaque action (portent les noms des actions).

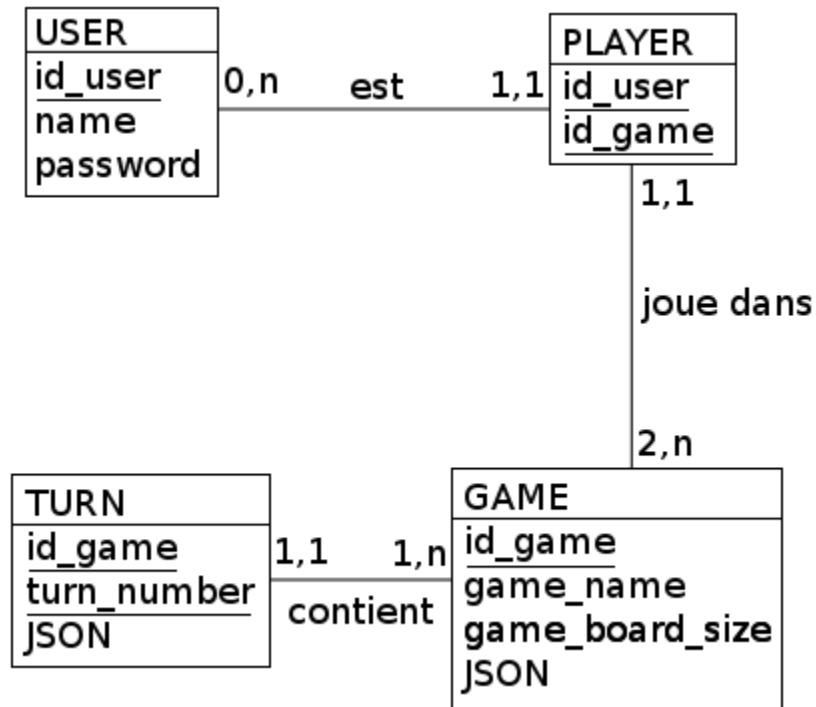
Conseil 7



II. Modèle de la base de données

a) Modèle conceptuel de données

Il faut établir la structure de la base de données. Quelles données faut-il stocker ? Quels sont les liens à établir entre ces données ?



Le modèle contient ces tables :

- User : le membre inscrit sur le site, avec son nom et son mot de passe.
- Player : le membre jouant une partie (obtenu avec l'identifiant du membre et l'identifiant de la partie).
- Game : la partie, avec son nom, la taille du plateau de jeu, et l'ensemble des JSON de la configuration de la partie.
- Turn : un tour de jeu sur une partie, avec son numéro de tour et l'ensemble des JSON reçus par le serveur au cours du tour.

Conseil 7

Un utilisateur inscrit sur le site (User) peut ne jamais jouer une seule partie, comme il peut en jouer une infinité. Un joueur d'une partie (Player) ne peut être qu'un seul joueur.

Un joueur d'une partie (Player) ne peut l'être que sur une partie. Une partie (Game) doit comporter au moins deux joueurs pour être entamée, mais théoriquement, une infinité de joueurs peut y participer (le nombre est défini à la configuration).

Une partie (Game) comporte au moins un tour, mais peut ne jamais se terminer. Le tour d'une partie (Turn) ne peut être associé qu'à une seule partie.

III. Dictionnaire de données

La base de données est nommée 'Auguste'. Le dictionnaire de données MySQL est situé dans la base de données Information_Schema. Il est possible de s'y référer pour récupérer contraintes, procédures et déclencheurs au besoin.

L'utilisation concrète du dictionnaire dans le cadre du développement n'a pas encore été clairement définie. La réflexion est encore ouverte à ce sujet.

Conclusion

Les différentes analyses à mener permettent à une logique de se construire au sein du projet. On prévoit les diverses utilisations de l'application, on établit un ensemble de classes au sein du code, des tables au sein de la base de données, on définit le bon déroulement d'une partie.

On construit ainsi en partie l'application avant de commencer à la coder. La structure pouvant s'avérer complexe, elle est envisagée avant d'être mise en application. On définit quoi coder avant de coder.