

Projet de Fin d'Etudes 2016-2017

Report week 50

# ChessProject

## Participants :

Tom Dall'Agnol  
Arnaud Garnier  
Lisa Joanno  
Simon Paris

## Encadrant :

Christophe Papazian

# 1. Résumé exécutif

Nous avons travaillé à lever les verrous techniques les plus importants et les plus risqués : le multijoueur et les interactions en réalité augmentée. Nous avons réussi à lever ces verrous et intégrer ces solutions autour d'un socle de règles de jeu basique : déplacer un pion sur les cases suivant ses mouvements possibles (ex : le fou ne peut se déplacer qu'en diagonale, le cavalier se déplace en 'L', ...) et chaque joueur joue l'un après l'autre.

Ainsi, en termes de modèle de jeu, nous avons implémenté la structure de données permettant de représenter notre plateau, qui n'est autre qu'un cube, qui amène des problématiques au niveau des déplacements permettant de passer d'une face à une autre. Nous avons implémenté les points avec des déplacements à déterminer. Ces actions de déplacement peuvent être exécutées en réalité augmentée grâce à un objet scanné au préalable (cet objet sera reconnu par la caméra et permettra de l'utiliser comme une baguette pour sélectionner le pion que l'on veut déplacer et la case où il ira) permettant ainsi d'interagir avec les objets virtuels. Enfin, nous synchronisons via un serveur les actions de chaque joueur rendant ainsi le jeu jouable à deux, sur deux ordinateurs différents. Nous avons des messages transmis volontairement courts pour réduire la latence entre les joueurs.

Nous voulions, au terme des deux semaines à plein temps, avoir un serveur pouvant gérer plusieurs parties de deux joueurs simultanément mais nous avons dû nous contenter d'un serveur n'hébergeant qu'une seule partie. Cela a permis tout de même de lever les verrous les plus importants, principalement sur la communication entre les clients Unity et le serveur codé en Go. Nous avons donc levé les principaux risques ce qui nous permet dorénavant d'avancer beaucoup plus sereinement.

Nous souhaitons aussi à ce stade avoir une version que nous aurions pu faire tester par de potentiels futurs utilisateurs. Il nous manque cependant certains éléments comme le déploiement sur smartphone avec cardboard pour rendre cela possible : actuellement nous déployons sur smartphone mais sans compatibilité avec le cardboard.

# Table des Matières

<b>Résumé exécutif</b>	<b>2</b>
<b>Description du projet</b>	<b>4</b>
Contexte	4
Objectifs	5
Risques	6
<b>Mise en œuvre</b>	<b>7</b>
Lots envisagés	7
Planning	9
<b>Retrospective</b>	<b>10</b>
Qu'avez-vous réussi ?	10
Qu'avez-vous raté ?	10
Qu'allez-vous faire autrement ?	10

## 2. Description du projet

### Contexte

Chessproject est un nouveau genre de jeu en réalité augmentée. Combinant jeu de plateau et jeu vidéo, nous proposons un jeu innovant à deux joueurs basé sur les interactions réelles et virtuelles. Pour simuler des lunettes de réalité augmentée, nous emploierons un smartphone.

Nous affrontons un autre joueur en ligne dans un jeu de stratégie au tour par tour où nous pouvons déplacer nos pions sur un plateau de jeu de forme cubique : l'adversaire peut donc arriver de tous les côtés. Divers pièges se trouvent sur le plateau, il ne faut donc pas les ignorer lorsque l'on se positionne. De plus, les différentes faces du cube sont échangeables, permettant de modifier un côté qui n'est pas à notre avantage (présence de pièges qui nous contraignent trop par rapport à l'adversaire) pour avancer vers la victoire.

La partie réelle sera alors le cube qui permet à la fois de servir de plateau, mais aussi de pouvoir changer les faces de manière réelle (les faces sont enlevables et remplaçables), mais la surcouche affichant les cases, les pièces et d'autres effets sera virtuelle, visualisable via le smartphone des joueurs.

Nous souhaitons mettre en avant dans notre projet l'interaction avec de la réalité augmentée sachant que nous avons une partie réelle (le cube physique avec les faces échangeables) et une partie virtuelle (les pions, les cases). C'est donc une partie très orientée recherche. Nous voulons aussi mettre en oeuvre un échange de données limité entre deux joueurs à distance par souci de consommation de données (étant sur smartphone, c'est une problématique importante, nous ne devons pas envoyer la totalité de l'état de jeu à chaque action). Une autre partie qui sera mise en avant dans ce projet est la montée en charge au niveau du serveur pour permettre d'avoir un grand nombre de parties en simultané sans qu'il n'y ait de soucis de latence (un temps de réponse < 100 millisecondes est souhaitable).

### Objectifs

Concernant les objectifs que nous envisageons pour ce projet, nous en avons plusieurs. On peut tout d'abord citer les objectifs que nous nous étions fixés pour le projet de PNSInnov et qui n'ont pas pu être atteints. Pour nous ces objectifs sont les plus importants, et il nous est impératif de les avoir atteints pour la fin de ce projet. Ces objectifs comprennent **l'ajout du mode multijoueur**, qui est l'intérêt principal pour les joueurs, **l'interaction en réalité augmentée** (à l'aide du Leap motion par exemple), ainsi que **l'adaptation de notre application au cardboard**. **Ces trois objectifs sont prioritaires** pour nous car ils sont nécessaires à l'obtention d'un jeu présentable au festival des jeux de Cannes.

Cependant, ce ne seront pas les seuls objectifs que nous souhaitons nous fixer. En effet, nous envisageons également d'ajouter de nouvelles fonctionnalités, afin de rendre notre jeu unique et innovant. Nous envisageons d'ajouter des comportements particuliers pour nos pièces, du type déplacement et déterminer les pièces à tuer. Mais l'idée ne s'arrête pas là : le joueur adverse ne serait pas au courant de la pièce à éliminer dans le camp adverse, ce qui fait que n'importe quelle pièce est suspecte. S'ensuit un jeu d'esprits qui vise à bluffer pour induire son adversaire en erreur. Par la suite, nous comptons ajouter une nouvelle fonctionnalité de cases piégées. Ces cases, en apparence des cases normales, auraient des comportements prédéfinis, qui auraient pour but de corser un peu la partie, en ajoutant une dimension "aléatoire" dans le jeu. Nous avons pensés à quelques pièges types :

- Une case *craquelée*, qui se détériore dès qu'une pièce s'y déplace, jusqu'à ce que la case se casse et qu'on ne puisse plus s'y déplacer ;
- Une case *incendiée*, où les pièces ne peuvent pas se déplacer (on peut même imaginer que le feu se propage, rendant les cases adjacentes risquées...);

- Une case “*convertisseuse*”, qui convertit en une pièce de la couleur du camp adverse (si on convertit la pièce à tuer, il faudra que le joueur qui perd sa pièce reconfigure une pièce à tuer) ;
- Une case *téléporteuse*, qui déplace toute pièce la franchissant sur une autre case aléatoire non occupée ;
- Une case *passeuse de tour*, qui n’a pas d’effet sur le plateau mais permet à l’adversaire de jouer deux fois de suite.

Afin de mieux exploiter la dimension de réalité augmentée, nous avons imaginé l’ajout du relief sur notre plateau. Certaines cases seraient en fait perçues comme des obstacles, qu’il faudrait escalader. Cela inclut donc l’ajout d’une capacité d’escalade pour les pions.

Concernant les objectifs que nous percevons comme des défis, on pourra citer le changement de face des modèles, qui sera sans doute la partie la plus difficile à implémenter, l’ajout du relief ainsi que les pièges.

L’implémentation des règles du jeu, du serveur pour le mode multijoueur et des interactions pour la réalité augmentée sont les trois parties principales de notre projet et il était impératif que nous travaillions en parallèle sur ces trois axes.

Cependant, l’intégration des interactions en réalité augmentée a été plus coûteuse en temps que prévu et nous avons choisi de repousser notre investigation sur les cardboards. En effet nous avons jugé ce point moins prioritaire car il nous est tout de même possible de faire une démo acceptable de notre travail, certes pas encore testable par des joueurs mais démontrant le lever d’un grand nombre de verrous techniques.

Nous avons aussi abandonné l’utilisation du Leap Motion (appareil permettant de capter les mains et de retransmettre leur mouvement dans le virtuel) car il entrerait en collision avec notre vision du produit final : un jeu de plateau transportable partout avec nous. L’ajout d’un tel appareil (certes améliorant notre précision) aurait à la fois consommé la batterie du téléphone de manière plus rapide mais aussi nous aurait fait perdre notre côté nomade et peu encombrant.

## Risques

Un de nos risques était de ne pas livrer un travail à la hauteur de ce qui est attendu pour quatre élèves de SI5. Après ces deux semaines de projet et le travail fourni sur l’implémentation de notre serveur, l’interfaçage avec les clients Unity et les interactions en réalité augmentée, nous estimons que ce risque est écarté. En effet nous avons appliqué beaucoup d’éléments que nous avons appris durant les semaines de cours avant cette période : définition d’interface, langage Go et sa gestion de la concurrence, interface homme/machine.

Concernant le déploiement sur smartphone, nous n’avons finalement pas été gêné par les problèmes de compatibilité que pouvait nous poser certains plugins. Nous avons réussi à utiliser nos smartphones pour travailler sur l’IHM en créant un exécutable compatible automatiquement.

Cependant le risque que nous avons identifié initialement concernant le matériel physique tel que les cardboards et smartphone a gagné de l’ampleur : pour pouvoir faire tester notre jeu et le présenter au festival du jeu il nous faut à minima deux smartphones avec des deux cardboards compatibles. Le fait que nous soyons amenés à solliciter énormément la batterie rend peu adaptée l’utilisation de nos téléphones, nous nous sommes donc tournés vers Polytech qui n’a pas répondu à nos attentes. Il est donc impératif pour nous de trouver une solution afin de pouvoir avancer sur nos phases de test.

Comme nous souhaitons présenter notre travail au festival du jeu de Cannes qui sera le week end entre nos deux prochaines semaines à temps plein, il nous faut atteindre la version finale de notre jeu en début de semaine 8 afin de ne pas travailler dans l’urgence. Il nous faudra donc être le plus efficace possible dans la phase qui arrive afin de continuer à être productifs malgré la reprise des cours. Nous travaillons de manière incrémentale, donc en cas de retard nous pouvons abandonner certaines fonctionnalités tout en ayant un jeu fonctionnel.

# 3. Mise en œuvre

## Lots envisagés

### Finis :

- Base du jeu
- Multijoueur (1 instance)
- Interaction avec un objet traquable

### En cours :

- Multijoueur à grande échelle (plusieurs parties en même temps)
- Interaction avec les *virtual buttons*

### Futurs :

- Cardboard
- Comportements
- Pièges
- Changement de face
- Cases en reliefs
- Interaction avec deux objets trackables

#### ● Base du jeu

Avoir une base de jeu sous Unity. Il faut avoir un système pour se déplacer de case en case, et un plateau de forme cubique d'où on peut se déplacer sur n'importe quelle case. La base sera faite en C#, langage que nous maîtrisons, aucun apprentissage n'est prévu à ce niveau là.

#### ● Cardboard : dépend de Base du jeu

La démo de notre jeu était faite grâce à une webcam. Pour qu'il soit jouable il faut que nous puissions jouer avec des masques de réalité augmentée (cardboard, ou dispositifs de meilleure qualité...).

#### ● Comportement : dépend de Base du jeu

L'idée est d'avoir un nombre de pièces prédéfinies avec pour toutes, un comportement de base (déplacement d'une case dans les 4 directions). On aurait ensuite d'autres comportements plus spécifiques, plus intéressants et en quantité limitée. Ces comportements seraient attribués comme on l'entend à nos pièces mais notre adversaire ne serait pas au courant de nos choix : ainsi s'instaurent des tactiques de bluff pour éviter que l'adversaire ne devine le comportement de chacune de nos pièces.

#### ● Changement de face : dépend de base de jeu

Notre plateau étant un cube, nous voulons ajouter la possibilité d'échanger les différentes faces. Il faut donc pouvoir sélectionner deux faces et pouvoir les échanger, que ce soit visuellement mais aussi au niveau du modèle de donnée.

#### ● Pièges : dépend de changement de face

Notre idée est de pouvoir donner la possibilité au joueur de placer des *pièges* sur le plateau. Voici des idées de pièges :

- Une case *craquelée*, qui se détériore dès qu'une pièce s'y déplace, jusqu'à ce que la case se casse et qu'on ne puisse plus s'y déplacer.

- Une case *incendiée*, où les pièces ne peuvent pas se déplacer (on peut même imaginer que le feu se propage, rendant les cases adjacentes risquées...).
- Une case *convertisseuse*, qui convertit en une pièce de la couleur du camp adverse (si on convertit la pièce à tuer, il faudra que le joueur qui perd sa pièce reconfigure une pièce à tuer).
- Une case *téléporteuse*, qui déplace toute pièce la franchissant sur une autre case aléatoire non occupée.
- Une case *passeuse de tour*, qui n'a pas d'effet sur le plateau mais permet à l'adversaire de jouer deux fois de suite.
- ... plus d'idées à venir ...

Ces pièges vont de pair avec une idée de changement de faces. En effet, si un joueur voit qu'une face du plateau est remplie de pièges, il aura la possibilité d'échanger deux faces, faisant, s'il se débrouille bien, tomber plusieurs pièces adverses dans des pièges (et peut-être même des siennes, dommages collatéraux).

- **Cases en relief**

Notre idée est d'introduire des cases en relief. Ces cases sont donc des obstacles. On associe aux pièces un niveau d'escalade, qui lui permet d'escalader les cases d'altitude inférieure ou égale à son niveau d'escalade.

Les lots suivants ont été modifiés :

- **Multijoueur : découpé en 2 parties**

- Un premier qui ne concerne qu'une seule instance (seuls 2 joueurs peuvent se connecter au serveur et jouer ensembles) et réduction au maximum des échanges de données
- Un second qui concerne la montée en charge au niveau nombre de parties en simultanées : pouvoir gérer un grand nombre de parties en simultanés sans ralentissement.

- **Interaction en RA : découpé en 3 parties**

- L'interaction en utilisant un objet trackable. Pour cela, on utilise un objet (cela peut être n'importe quel objet, dès l'instant qu'il est suffisamment complexe pour pouvoir être reconnu). On ajoute par la suite sur cet objet un objet virtuel qui servira de pointeur pour l'utilisateur, lui permettant de sélectionner un pion et la case vers laquelle il souhaite le déplacer.
- L'interaction avec deux objets traquables. Ici on souhaiterait utiliser deux cylindres colorés que nous placerions à l'extrémité de deux de nos doigts, et qui permettrait de sélectionner un pion en le "serrant" entre ses doigts.
- L'interaction en utilisant les *virtual buttons*, proposés par Vuforia. Le plugin Vuforia de Unity3D propose un outil, appelé *virtual button*. Il s'agit de boutons pouvant être associés à un plan, et qui détectent automatiquement le passage de la main de l'utilisateur. On peut par la suite associer à ces boutons des actions spécifiques. Pendant la première de ses deux semaines, nous avons levé les verrous techniques concernant l'utilisation de *virtual buttons*. La difficulté de l'implémentation de cette solution est qu'elle nécessite de travailler sur des images planes, au lieu d'un cube comme nous pouvions le faire avec les objets traquables.

## Planning

Nous avons globalement suivi le planning original. Nous avons atteint à nouveau le jeu 1 joueur que nous avions en PNSInnov. A la fin de ces deux semaines, nous avons en plus le jeu 100% jouable en multijoueur. Le bémol est l'utilisation des cardboards. Nous n'avons pas eu accès à ces cardboards donc nous n'avons pas pu tester le déploiement sur smartphone.

Le point positif est l'intégration du coeur du jeu avec les intégrations avec la réalité augmentée, sur lequel nous avons mis l'accent pour être sûrs que nous ne développions pas deux modules séparés que nous n'arriverions pas à réunir.

De plus, le coeur du jeu est beaucoup plus propre et la base de code est plus propre, ce qui sera utile pour développer plus de features dans l'avenir.

Dans les premières semaines de code, Tom et Lisa se sont concentrés sur la refonte du coeur du jeu sous Unity : recréation de la logique du jeu, en repensant l'architecture logicielle pour être plus évolutive et en étant intégrable (IHM et multijoueur).

Arnaud a travaillé sur l'IHM, en intégrant les concepts de Vuforia d'abord, puis en les appliquant pour réussir à trouver la bonne interaction : plusieurs solutions ont été essayées, pour finalement en trouver une fiable.

Simon a développé le serveur : code en GO du serveur et déploiement sur Heroku, puis communication avec deux clients Unity, eux codés en C#.

## **4. Retrospective**

### **Qu'avez-vous réussi ?**

- Nous avons implémenté une solution permettant de jouer en multijoueur et en limitant au maximum les échanges de données, ce point nous avait tenu en échec lors du PNSInnov.
- Nous proposons une première façon fonctionnelle d'interagir avec la réalité augmentée qui permet déjà de jouer basiquement.
- L'intégration entre la partie fonctionnelle, interactive et le serveur s'est faite sans encombre grâce à l'architecture des différentes parties qui a été bien pensée (car pensée pour).

### **Qu'avez-vous raté ?**

- Nous n'avons pour l'heure pas réussi à implémenter un serveur qui permettait une montée en charge, ce qui est maintenant une des priorités.
- Nous n'avons pas eu le temps de tester notre jeu avec des cardboards.
- Nous n'avons pas obtenu de résultats probants sur l'interaction avec la réalité augmentée (difficilement jouable), mais nous avons de multiples pistes pour l'atteindre.

### **Qu'allez-vous faire autrement ?**

- Continuer à utiliser trello pour le kanban mais réduire la taille des tâches pour permettre de livrer de la valeur plus régulièrement.
- Porter l'accent sur la jouabilité du jeu, notamment en testant le jeu sur cardboards. Nous allons faire les démarches pour en acheter ou nous les faire prêter.
- Nous allons également créer les cubes au FabLab, toujours pour garantir un produit final jouable.