

République Algérienne Démocratique et Populaire  
Université Abou Bekr-Belkaid - Tlemcen  
Faculté des Sciences  
Département d'Informatique

Mémoire de fin d'études  
Pour l'obtention du diplôme de Licence en Informatique

## Thème

**PORTAIL DES JEUX EDUCATIFS**

بوابة الألعاب التربوية

Réalisé par :

- MR : ABDALLAH BENSLIMANE

Présenté le 28 Mai 2015 devant la commission d'examination  
composée de MM.

- MR : M.Benazzouz (Encadreur)

- MR : M.Messabihi (Examineur)

- MR : H.Mahfoud (Examineur)

Année universitaire: 2014/2015

## *Remerciement*

Je tiens d'abord à remercier Allah pour sa grâce et sa bénédiction.

J'adresse mes remerciements et ma gratitude à mon encadreur Mr Mourtada Benazzouz qui a conjugué sa compréhension et son talent afin d'encadrer mon travail.

Mes remerciements les plus sincères à tous mes amis et spécialement Mrs Kadi Yasser, Bekaddour Abderrazak, Benkada Fethellah et tous ceux qui nous ont prêté mains fortes pour la réalisation de ce projet.

Mes chaleureux remerciements à ma famille et surtout mon Oncle Mr S.Boumediane pour sa confiance à moi et son aide.

Je remercie également les membres de jury qui m'a fait honneur en acceptant d'examiner et de juger mon travail.

Et finalement, je remercie tous qui m'ont dit NON ou TU NE PEUX PAS, car à cause de ses paroles j'ai devenu plus fort !

# Sommaire

<b>Introduction Générale</b> .....	5
<b>Chapitre I : Jeux vidéo &amp; Développement</b> .....	6
1. Les jeux vidéo .....	7
1.1. Définition .....	7
1.2. Historique .....	8
1.3. Les grandes entreprises .....	9
1.4. L'impact des jeux vidéo .....	10
1.5. Les dernières années .....	10
2. Le développement des jeux vidéo .....	11
<b>Chapitre II. Unity &amp; Développement Mobile</b> .....	14
1. Unity .....	15
1.1. Définition .....	15
1.2. Les outils intégrés dans Unity .....	15
1.3. Les avantages et les inconvénients .....	16
2. Développement Mobile .....	18
2.1. Android .....	18
2.2. iOS .....	19
2.3. Windows Phone .....	19
<b>Chapitre III. Portail Des Jeux Educatifs</b> .....	20
1. Introduction .....	21
2. Portail des enfants.....	21
2.1. Dessin .....	21
2.2. Coloriage .....	22
2.3. Lettres .....	23
2.3.1 Prononciation des lettres .....	23
2.3.2 Apprentissage des lettres .....	23
2.3.3 Quizz des lettres .....	23
2.4 Animaux avec la réalité augmentée .....	25
2.4.1 Définition de la réalité augmentée .....	25
2.4.2 RA et Unity .....	25
2.4.3 RA dans mon application .....	25
3. Portail de réflexion .....	26
3.1. Chemin Eulérien .....	26

3.2. Cerveau Vif .....	28
3.2.1 Logique .....	28
3.2.2 Visuel .....	29
3.2.3 Concentration .....	30
3.2.4 Maths .....	30
3.2.5 Mémoire .....	31
3.2.6 Statistiques de cerveau .....	32
3.2.7 Tests Spéciaux .....	32
4. Portail de Culture Générale .....	34
4.1. Mots .....	34
4.2. Chimie avec RA .....	35
4.3. Empire de Connaissance .....	36
4.3.1 Mode individuel .....	36
4.3.2 Mode Multi-joueurs .....	39
4.3.2.1 Réseau locale .....	40
4.3.2.2 Internet .....	41
4.3.2.3 Structures de communication dans « Unity » .....	43
4.3.2.4 Structure de jeu en mode Multi-joueurs .....	44
4.3.2.5 Problèmes et Solutions proposées .....	45
<b>Conclusion Générale .....</b>	<b>49</b>
Liste des Figures .....	50
Liste des Tableaux .....	50
Références Bibliographiques .....	51

## Liste Des Figures

Figure 1. Interface de « Unity » .....	17
Figure 2. Système d'animation avancé .....	17
Figure 3. Dessin .....	22
Figure 4. Coloriage .....	23
Figure 5. Lettres .....	24
Figure 6. Animaux avec la Réalité Augmenté .....	26
Figure 7. Chemin Eulérien .....	27
Figure 8. Cerveau Vif .....	33
Figure 9. Mots .....	35
Figure 10. Chimie avec la Réalité Augmenté .....	36
Figure 11. Structure de Communication avec RPC .....	44
Figure 12. Empire de Connaissance .....	48

## Liste Des Tableaux

Tableau I. Entreprises des jeux vidéo .....	9
Tableau II. Plates-formes des jeux vidéo .....	9

## Introduction Générale

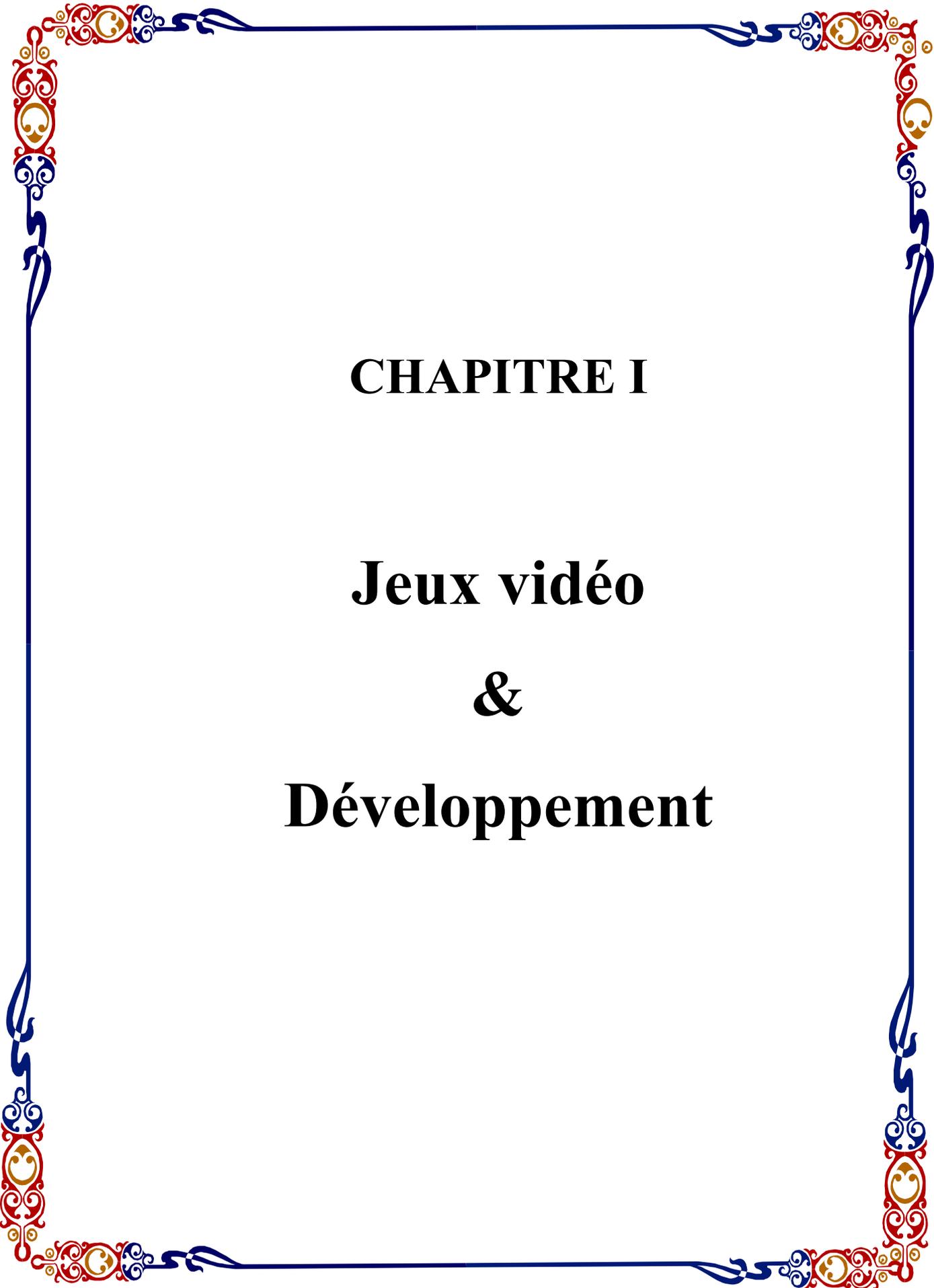
Un monde sans jeux vidéo semble impossible de nos jours. L'industrie des jeux est considérée comme l'une des plus importantes avec un impact affectant tous les domaines (économie, l'éducation, la culture, ...).

Notre objectif est de faire la conception et la réalisation d'un produit destiné particulièrement à notre société. Nous avons remarqué qu'il existe un très grand nombre de jeux, mais malheureusement très peu d'entre eux sont dotés d'une version en langue arabe. Ceci nous a poussés à faire des applications à caractère éducatif et ludique pour différents niveaux de culture ou tranche d'âge.

Le premier chapitre présente des jeux vidéo en général, de l'histoire, leurs impacts dans différents domaines, leurs types, les entreprises, les moteurs des jeux, le développement dans les dernières années, les Smartphones, le système Android, ...

Le deuxième chapitre décrit le moteur des jeux « Unity », où nous allons développer son environnement, les outils, les langages de programmation et les bibliothèques, ses avantages et inconvénients, et ce que nous pouvons faire avec ces outils.

Le troisième et dernier chapitre est destiné pour décrire notre application en précisant les dessous de la conception de chaque jeu ainsi que leurs fonctionnalités. Nous terminons notre mémoire par une conclusion des perspectives futures.



# CHAPITRE I

## Jeux vidéo & Développement

## 1. Les jeux vidéo :

### 1.1. Définition

Un jeu vidéo est un jeu électronique qui implique l'interaction humaine avec une interface d'utilisateur pour générer un feedback visuel sur un dispositif vidéo. Le mot « vidéo » dans le jeu vidéo traditionnellement référence à un dispositif d'affichage de trame, mais il implique maintenant tout type de dispositif d'affichage qui peut produire des images en deux ou trois dimensions. Les systèmes électroniques utilisés pour jouer à des jeux vidéo sont connus comme « les plates-formes ».[1]

- **Les plates-formes** : L'expression «plate-forme» fait référence à la combinaison spécifique de composants électroniques ou de matériel informatique, qui en conjonction avec le logiciel, permet un jeu vidéo de fonctionner. Le terme « système » est aussi couramment utilisé.

Quelques exemples de plates-formes: PC (sous Windows, Linux, Macintosh ...), Consoles (tels que Nintendo DS, N64, Gameboy Advance, ...), Playstation, Xbox, services de jeu en ligne, Smartphones ...

Les jeux vidéo sont spécialisés dans **des genres** qui sont [2]:

- Educationnel/Réflexion.
- Action :
  - FPS (First Person Shooter) : FarCry, Cull Of Duty, Counter Strik, ...
  - TPS (Thrid Person Shooter) : Tomb Raider, Freedom Fighters, ...
  - Stealth Games : Splinter Cell, OddWorld, ...
  - Horreur : Doom, The House, Silent Hill,...
- Aventure : Mario, Prince Of Persia, Assassin's Creed, The Witcher , ...
- Role-Playing-Games (RPG) : Deux Ex, Final Fantasy, Mass Effect, ...
- Simulation : City Builder, Transport Simulator, Sims, MineCraft, ...
- Stratégie : WarCraft, Age Of Empire, Generals, ...
- Sport :
  - Racing : Need For Speed, Blur, Asphalt, ...
  - Football : PES , FIFA , ...
  - Autres : NBA , WWE , ...

Les jeux vidéo ont continué à devenir une forme d'art et de l'industrie.

## 1.2. Historique

L'histoire de création des jeux commence avec le développement des premiers jeux vidéo, mais « quel jeu vidéo est le premier » dépend de la définition du jeu vidéo. Les premiers jeux créés avaient peu de valeur de divertissement, et leur objectif de développement était séparé de l'expérience d'utilisateur.

Le vrai développement des jeux vidéo pour le commerce est commencé au début d'année 1970 (au moment des télévisions noires et blanches). Voici une liste des premiers jeux vidéo jusqu'à 1972 :

- Cathode Ray Tube Amusement Device (1947)
- Chess (1947–1958)
- BryerBettencourt (1950)
- Nim (1951)
- Strachey's Draughts Program (1951)
- OXO / Noughts and Crosses (1952)
- Tennis for Two (1958)
- Mouse in the Maze, Tic-Tac-Toe (1959)
- Spacewar! (1962)
- Brown Box/Odyssey (1966)
- Galaxy Game (1971)
- Computer Space (1971)
- Pong (1972)

Après le succès de ces jeux, l'industrie de développement des jeux vidéo commence à se développer et à être plus populaire, et de nouvelles entreprises ont commencé à apparaître.

L'industrie a élargi de manière significative, avec le secteur « arcade » de jeux vidéo seul (représentant la plus grande part de l'industrie du jeu) **générer des revenus plus élevés que à la fois la musique pop et les films hollywoodiens combinée !**

Les grandes entreprises connues de l'industrie commence à présenter leurs premiers jeux tels que Mario -1981- (par Nintendo), Pac Man -1980- (par Namco), ... et commence à développer de nouvelles plates-formes pour leurs jeux, donc pour Nintendo, ils commencer avec le Game Boy (1989) et Super Nintendo (1990), N64, Nintendo DS, Game Cube, ... jusqu'à Nintendo Wii U, au même temps, par exemple SEGA commence à distribue leur jeu le plus populaire "Sonic", ... [3]

### 1.3. Les grandes entreprises

Différentes entreprises se spécialisent, les plus populaires dans le domaine de développement de jeux vidéo:

Atari	EA
Namco	Ubisoft
Sega	Epic
Nintendo	THQ
Sony	Konami
Bungie	Rockstar
Valve	Id Software
PopCap	BioWare

**Tableau I – Entreprises des jeux vidéo**

Voici la liste des plates-formes des jeux vidéo les plus populaires et leurs fabricants :

<b>Plate-forme</b>	<b>Fabricants</b>
Atari X	Atari
Dreamcast	Sega
Game Boy/Game Boy Color/Game Boy Advance	Nintendo
Game Cube	Nintendo
Windows	Microsoft
Linux	Linus Torvalds
iOS	Apple
Android	Google
Xbox 360/Xbox One	Microsoft
NeoGeo	SNK
NES/SNES/Nintendo 64/Nintendo DS/Nintendo Wii/Nintendo 3DS/Wii U	Nintendo
PS1/PSP/PS2/PS3/PS4/PS Vita ...	Playstation/Sony
Macintosh	Apple

**Tableau II – Plates-formes des jeux vidéo**

#### 1.4. L'impact des jeux vidéo

L'effet des jeux vidéo touche beaucoup de domaines, mais ils impactent principalement les domaines suivants :

- **Economie :**

Voici un rapport spécial pour la situation dans l'États-Unis :

Une nouvelle étude de « EconomistsIncorporated » rapporte que l'industrie du jeu vidéo a ajouté \$ 4,95 milliards au produit intérieur brut des États-Unis l'an dernier, et le côté divertissement du logiciel est développer plus rapidement que les autres segments de l'économie.

Croissance annuelle réelle parmi les fabricants de jeux dépassé les 10 pour cent entre 2005 et 2009, selon l'étude. C'est 7 fois le taux de l'économie américaine de croissance !

Le rapport souligne que la croissance annuelle réelle entre 2005 et 2008 était de 16,7 pour cent. Aujourd'hui, il y a plus de 32 000 personnes directement employées par les éditeurs et les développeurs dans 34 états.[4]

Depuis le site célèbre des statistiques *statista.com*, les recettes mondiales des jeux PC et consoles en 2014 sont **\$ 46,5 milliards**.

Et depuis des statistiques en 2014 de *newzoo.com* **l'Algérie** est classé la **82** dans la liste des pays par chiffre d'affaires de jeu tel que les recettes totales sont : **\$18,870,748**.[5]

- **Education et Culture :**

Il n'est pas inconnu que les jeux vidéo ciblés impactent positivement et négativement à l'enfant / utilisateur et sa culture, mais nous sommes ici pour parler des avantages, de sorte que les dernières études ont confirmé que les jeux éducatifs peuvent aider le cerveau à travailler et obtenir beaucoup de nouvelles compétences telles que: concentration, réflexion, visuelle, apprentissage rapide, ... et lui (cerveau) va apprendre de façon créative et essayez de penser « hors de la boîte ».

#### 1.5. Les dernières années

Au moment où les iOS systèmes (pour iPhones et iPads) était la dominante du marché des jeux mobiles, un nouveau système se présente appelé « Android » publié par Google (2005) et a pris son célèbre comme un nouveau système qui est gratuit, son matériel n'est pas cher, et est un système « open source » pour les Smartphones. Donc, le marché Android grande plus et plus chaque jour, et sa popularité devient plus grand et plus gros, et la plupart des développeurs ont tourné leur

développement au ce système étonnant. En juillet 2014, une statistique était publié montre que le marché de Google (Android) compte environ 1,3 million d'applications, de l'autre côté, le marché Apple a 1,2 million d'applications avec la connaissance que le marché Apple a été ouverte plus tôt que celui de Google.

Voici un exemple d'un développeur sur Android qu'il développe le jeu célèbre « Floppy Bird » :

Nguyen (le développeur) a plus de **50 millions** de personnes ayant téléchargé son jeu et il gagne **\$ 50 000** par jour - grâce à des milliards d'impressions publicitaires aux des millions de personnes -.

Et il ne faut pas oublier les recettes de Rovio – la créateur du « Angry Birds » - et les autres petits entreprises...

## 2. Le développement des jeux vidéo

Le développement du jeu est effectué par un développeur du jeu (Game Developer) allant d'un individu à une grande entreprise. Les développeurs indépendants comptent sur le soutien financier d'un éditeur de jeu, et Ils ont généralement de développer un jeu du concept au prototype sans financement extérieur, ce type des développeurs appellent « Indie Game Developers ».

Les membres de groupe de développement (généralement) sont :

- **Game Designer :**

Est la personne qui créer tous les objets du jeu tels que les modèles 3D (bâtiments, personnages, armes, ...), ou des graphiques 2D ... la liste des objets est générés par le concepteur de jeux elle-même (Game Designer) ou par une collection des idées de tous les membres de l'équipe, cette liste appelé (Cahier des charges) doivent être acceptés par le Directeur Créatif (Creative Director).

- Les outils qui peuvent être utilisés: 3D Studio Max, Maya, ZBrush, Cinema 4D, Blender, ...

- **Graphics Artist :**

Ce membre peut être juste le « Game Designer » dans certains projets, ou un autre membre appelé « Graphics Artist» qui est plus spécialisé dans les graphiques 2D et les interfaces graphiques (GUI) tels que le menu principal, Fonds (Backgrounds), boutons, ...

- Les outils qui peuvent être utilisés: Photoshop, Corel Draw, PaintShop Pro, ...

- **Level Designer :**

Cet membre a pour terminer le travail de « Game Designer », de sorte que le présent dernière (Game Designer) créer les objets et modèles du jeu, puis les donner à « Level Designer » pour construire avec eux les niveaux (comme placer les bâtiments, les rues, les voitures, « triggers », ...), ceci est le cas quand nous avons des jeux 3D compliqués, mais pour les jeux simples en 2D, le « Level Designer » peuvent être remplacés par le « Game Designer » (donc il est celui qui crée les objets et les placer dans le jeu) .

Les outils qui peuvent être utilisés: des logiciels de modélisation 3D, logiciels de construction de niveau.

- **Programmeur :**

Le programmeur est comme l'âme dans le corps, de sorte que tout ce qui fait par les autres membres de l'équipe sera juste des objets de pierre sans un script qui peut leur donner certaines fonctionnalités et certains IA (Intelligence Artificielle). Donc c'est lui qui contrôle la physique du jeu, « Game Play », son, animation, IA, GUI, ...

On note qu'un programmeur **débutant** peut faire, en moyenne, environ **\$70.000** par an et un programmeur **expérimenté** peut faire, en moyenne, autour de **\$125 000** par an.

- **Ingénieur du Son :**

Les ingénieurs du son sont les responsables techniques pour les effets sonores et le positionnement du son. Ils supervisent parfois l'enregistrement de la voix et d'autres la création des sonores et les sons d'effets ou les musiques.

- **Animateur :**

C'est le membre qui est responsable à tous qui bouge statiquement c.à.d. les animations et les mouvements créent avant la compilation du jeu, comme le mouvement du personnage (qui est déjà modelé par le « Game Designer »), l'animation de menu, boutons,...

- **Game Tester :**

Leur travail est juste de joué ! Mais avec des conditions qui sont de chercher les bugs et les erreurs dans le jeu, on utilisant des techniques de test (il connut exactement comment il fait pour trouver les bugs). On note que beaucoup de programmeurs et « Game Designers » commence sa carrière de ce point.

- **Creative Director :**

C'est le personne qui contrôle les taches du jeu et sont priorités, donc il décide est que cette tache peut être ici ou non, il peut changer l'objectif du jeu (ou stage), le cahier des charges est acceptable par lui,...

**Remarque :** l' « Indie Game Developer » des fois il est obligé de faire le travail de tous les membres précédents !

Et il y a d'autres membres comme le financier, le chef des programmeurs/designers...

Après que tous les membres de l'équipe font leurs travaux, tout être intégré à un **moteur de jeu** pour que le programmeur peut faire le « Game Play » et d'autres fonctionnalités.

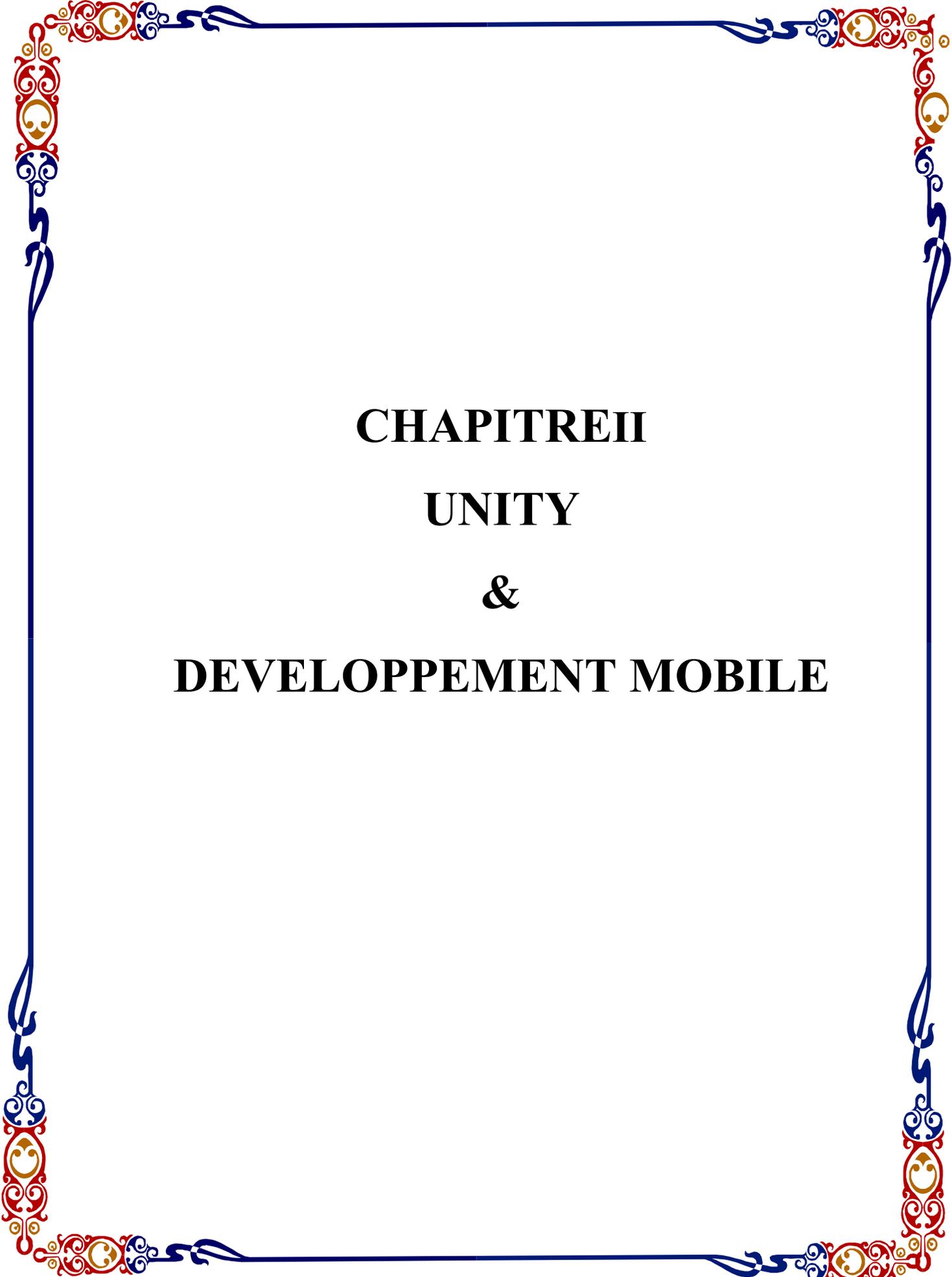
- ❖ **Le Moteur du Jeu :**

C'est un ensemble des outils de tous les domaines reliés au développement des jeux vidéo (les domaines des membres du projet) cet outils sont combiné dans un seul EDI appelé un Moteur du Jeu.

Voici une liste des moteurs du jeu les plus connus et sont gratuits :

- Unity (celui que je travaillais avec).
- UDK (Unreal Development Kit).
- Cry Engine.
- 3D Game Studio.
- Game Maker.
- Cocos2D.
- Game Salad.
- Virtools.
- Blender.

On note qu'il y a des jeux qui sont développer par des EDI qui ne sont pas des moteurs des jeux (ex : développement des jeux 2D avec Visual Studio (C++/C#/VB)).



**CHAPITRE II**  
**UNITY**  
**&**  
**DEVELOPPEMENT MOBILE**

# 1. Unity

## 1.1. Définition

« Unity » est un moteur du jeu très puissant d'un côté, et facile d'un autre côté. Il est développé par une société appelé « Unity Technologies » depuis la version Unity 1.0 au Juin 2005 jusqu'à la version actuelle Unity 5.0.1f.

Il s'agit d'une grande quantité d'outils et d'une très grande bibliothèque de programmation spécialisé pour lui.[7]

## 1.2. Les outils intégrés dans Unity

- ❖ Un Editeur de scripts appelé « MonoDevelop » qui supporte les 3 langages.
- ❖ Un Editeur de scènes 2D/3D et autre pour le test du jeu en temps réel.
- ❖ Un Moteur de physiques 2D/3D qui supporte NVIDIA® PhysX et Box2D, avec une bonne simulation de gravité, collision, ....
- ❖ Un Contrôleur du son (devient un éditeur du son depuis la version 5.0).
- ❖ Outils de contrôle et optimisation avec des statistiques de travail de RAM, Carte Graphique (VRAM) ,CPU, ...
- ❖ Un Editeur d'animation simple et une autre avancée, combinée par un contrôleur d'états d'animations.
- ❖ Un Créateur des terrains (3D) avec la texture ...
- ❖ Un ensemble de composants tel que :
  - Un rendu du Mesh (Mesh c.à.d. un model 3D) et de Texte (3D).
  - Un système de particules (pour les effets comme feu, explosion,...).
  - Un rendu de sentier et de lignes (j'ai utilisé pour le principe de dessin).
  - Des composants de collision avec la propriété « Trigger ».
  - Un simulateur de chiffon.
  - Un composant pour la simulation des physiques (2D/3D) avec les propriétés de gravité, masse, vitesse,...
  - Des composants pour la simulation de lumière (de lampe, soleil, projecteur, ...).
  - Des composants pour l'utilisation des interfaces graphiques (GUIs).
  - Un composant de transfert les données générales (comme la position, rotation,...) vers le réseau.

Et il y a beaucoup d'autres outils que je ne peux pas les citer totalement...

### 1.3. Les avantages et les inconvénients

Les avantages et les inconvénients principaux de « Unity » :

#### ❖ Les avantages :

- Il supporte trois langages de programmation (JavaScript, C#, et Boo).
- Il peut compiler le jeu en plusieurs environnements/plateformes (actuellement il supporte **21 plateformes** ! qui sont : iOS, Android, Windows Phone 8, BlackBerry 10, Tizen, Windows, Windows Store Apps, Mac, Linux/SteamOS, Web Player, WebGL, PS3, PS4 et Morpheus, PS Vita, Xbox 360, Xbox One, WiiU, OculusRift, GearVR, Android TV, et Samsung SmartTV).
- Il a un moteur de physiques 3D/2D intégré qui supporte les moteurs de physiques **NVIDIA PhysX 3** et **Box2D**.
- Il a aussi un moteur d'animation intégré qui utilise des contrôles simples (pour l'utilisation standard) ou des techniques plus avancées comme (Motion Capture).
- Il a un éditeur de scènes très rapide et compatible avec la plupart des plateformes (il est très utilisable surtout quand on développe sur Android par exemple).

Et d'autres avantages on ne peut pas citer leur totalement...

#### ❖ Les inconvénients :

- Il n'a pas des bibliothèques pour le réseau multi-joueurs locale et surtout pour **Wifi-Direct** on Android.
- Il n'a pas aussi des bibliothèques pour la **base de données** (il n'a pas aucune notion de « base de données »).
- Il ne supporte pas le **langage Arabe**.
- La taille du jeu est un peu **grande**.
- Les versions antérieures de Unity (depuis la version 4.0) ne supporte que les Smartphones de système Android qui ont un API de version 2.3.1 et antérieur.

On note que « Unity » travaille avec le principe « Making Games From Scratch » - création des jeux vidéo depuis rien - c.à.d. le développeur doit créer tous ces objets, outils, scripts,... Cet principe peut être un inconvénient pour des gens, mais il peut être aussi un très grand avantage car il vous permet de créer et réaliser n'importe quelle idée que vous avez !.

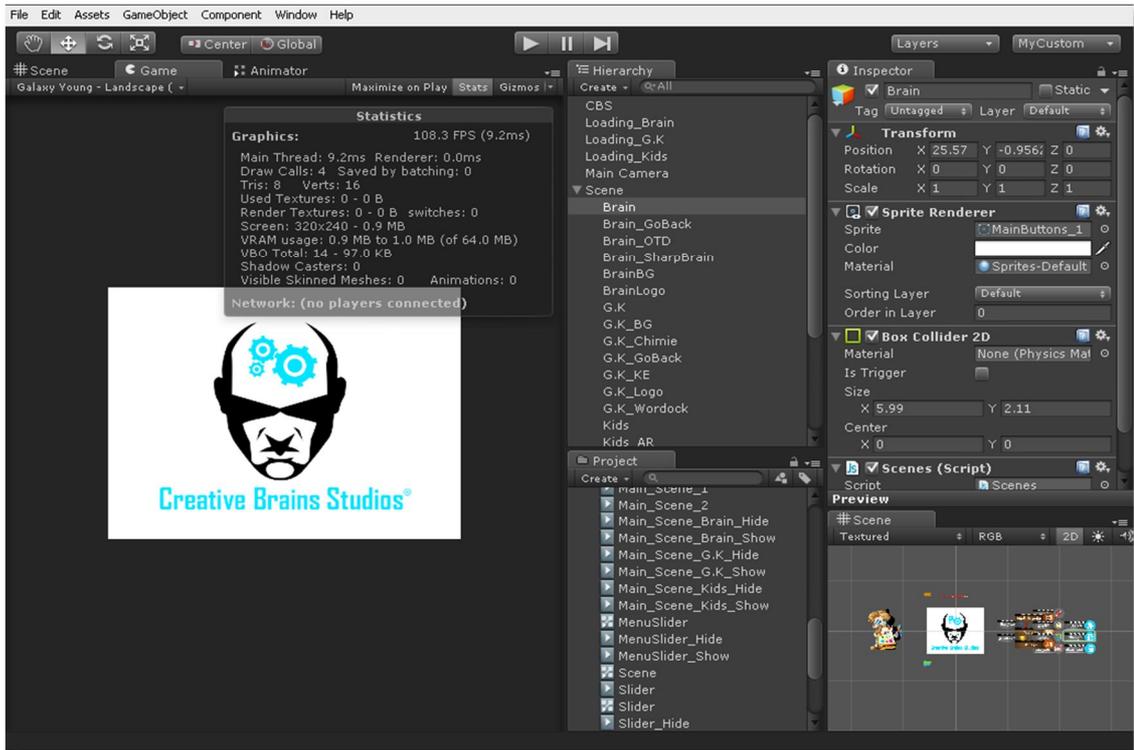


Figure 1 – Interface de « Unity »

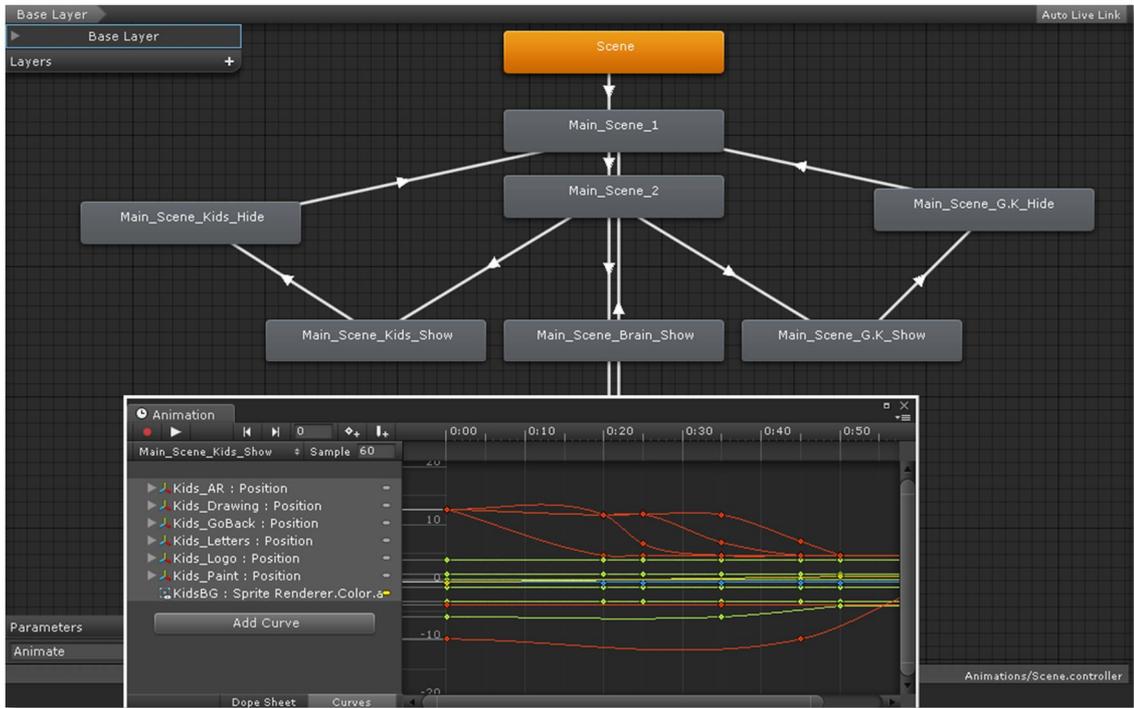


Figure 2 – Système d'animation avancé

## 2. Développement Mobile

### 2.1. Android

Android est un système d'exploitation des mobiles basé sur le noyau de Linux, et développé par Google (2005). Il est désigné pour la touche digitale d'écran de mobile. Il est utilisé dans plusieurs machines comme les Smartphones, télévisions, voitures, heures,...

Google a un très grand marché des applications d'Android (il est le 1er actuellement) connu comme « Google Play Store » qui a plus de 1.3 million applications publiés et plus de 50 millions applications téléchargés.

Les dernières statistiques confirment que plus de 80% des développeurs créant ses applications pour le système Android, et qu'il est une plate-forme prioritaire que iOS ou autres plates-formes. A l'année 2014, Google trouvé que le nombre d'utilisateurs d'Android est plus de 538 million !

Android est un « Open Source » système, et leur code source publié par Google pour chaque nouvelle version du système. Il support 68 langues différents et les architectures des processeurs (ARM) de 32-bits et 64-bits. La dernière version de cet système est 5.1.1 « Lollipop » (21 Avril 2015).[6]

Le développement pour le système Android peut être avec l'utilisation de différents outils tel que « Eclipse » pour le développement des applications (en général), les moteurs des jeux (la plus part des moteurs des jeux qui sont puissantes et connus supportent maintenant le système Android) tel que Unity, UDK, Coco2D, ...

Google développé un SDK (Software Développement Kit) – qu'il a des mis à jours pour chaque nouvelles – pour les outils de développement pour l'intégré dans ces environnements et mètre Android supporté. Ce SDK utilise le langage JAVA pour ces classes et fonctions, mais il peut support aussi d'autres langages tel que C++ (cette méthode est appelé « Native Code »). Dans « Unity », il utilise une grande partie pour ces classes et méthodes propres, mais il a la possibilité d'utilise les classes du SDK ou Native Code directement, mais avec un peu de soufrage pour le développeur.

## 2.2. iOS

C'est un système d'exploitation mobile d'Apple ciblé pour plusieurs machines. Il partage les fondations du noyau hybride XNU et les services Unix et Cocoa. Ce système a des différents plates-formes qui sont : iPhone, iPod Touch, iPad, Mac et Apple TV.

L'utilisation de ce système est très populaire dans les appareils mobiles, et Apple a un grand marché pour les applications développées pour iPhone. Ce marché a pas mal de popularité, mais l'entrée de Google et son nouveau système Android met ce dernier (marché de Apple) à une très grande compétition. La dernière version avancée du système iOS est 8.4 beta 1 (13 avril 2015).

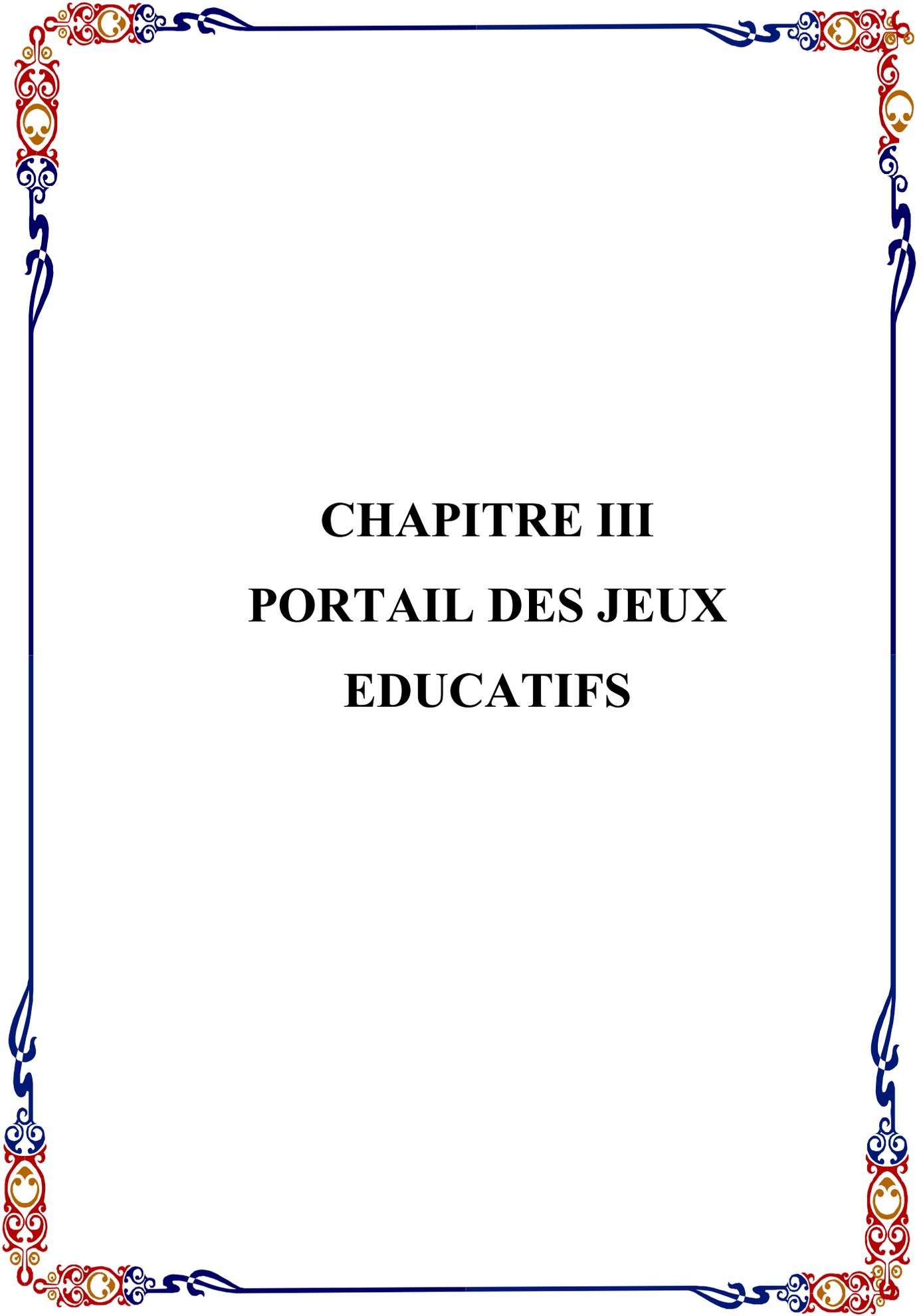
Le support d'application utilisé sur l'iPhone et l'iPod touch est basé sur une architecture ARM contrairement aux processeurs utilisés sur les anciennes versions des ordinateurs Apple (PowerPC) ou aux récents (Intel x86). De plus, iOS utilise l'API OpenGL ES tournant sur une carte graphique 3D double cœurs PowerVR. En somme, les applications développées sous Mac OS X ne peuvent pas fonctionner sur un iPhone ou un iPod Touch, toutes les applications natives sont redéveloppées spécifiquement pour l'architecture ARM et les composants logiciels d'iOS.[10]

## 2.3. Windows Phone

Windows Phone est un système d'exploitation mobile développé par Microsoft pour entrer au monde des applications mobiles et obtient sa part de marché mondiale. Ce système est destiné au grand public qu'au marché des entreprises, mais depuis Windows Phone 8, Microsoft propose des fonctions avancées pour les entreprises comme : un espace d'applications juste pour les entreprises...

Le système supporte plus de 25 langues, et il est basé sur le noyau Windows CE 7.0 et Windows NT 6.2. La dernière version stable est Windows Phone 8.10.14234.375 (Janvier 2015).[11]

Notez que le moteur des jeux utilisé pour le développement de notre application « Unity » supporte le développement dans tous les systèmes précédents (Android, iOS, Windows Phone) sans avoir changé beaucoup de choses.



**CHAPITRE III**  
**PORTAIL DES JEUX**  
**EDUCATIFS**

## 1. Introduction

Mon application est une sorte d'encyclopédie qui contient une collection de jeux et applications ciblés pour toutes les catégories de la population (petits, grands, étudiants, travailleurs, intellectuels ...), et il est divisé par trois grandes catégories: Enfants, Réflexion, et Culture Générale, chacun d'eux possède environ 2-4 jeux / applications dont ils peuvent aussi contenir plus de jeux. Toutes ces applications seront dotées d'une version en langue arabe.

L'application est développée avec le moteur des jeux « Unity » pour le système « Android » avec une exigence du système de : (**Minimum**)

- Android 2.3.1 'GingerBread'.
- API Level 9.
- CPU architecture de ARMv7.
- Supporte OpenGL ES 2.0.

Je vais passer maintenant à décrire chaque catégorie est ces applications.

## 2. Portail des enfants

Cette catégorie contient 4 types de jeux éducatifs ciblés pour les enfants, qui sont :

### 2.1. Dessin

Est une application de dessin pour les enfants, l'utilisateur peut choisir une des photos disponibles à dessiner, puis la scène de dessin va montrer avec un ensemble d'outils qui ils sont:

- Un outil de redimensionnement de la brosse, qui permet à l'utilisateur de contrôler la taille du pinceau de dessin.
- Un outil de redimensionnement de gomme, de même que le dernier mais d'effacement.
- Un système de « Annuler/Répéter » avec 10 itérations.
- Un bouton de réinitialisation, qui détruit tous les pointeurs de pinceau instanciés, et efface les tableaux de « Annuler/Répéter » et remettre leurs compteurs à zéro.
- Bouton de Guide « On / Off » pour afficher / cacher l'image sélectionnée (Guide).
- Et enfin, un bouton pour enregistrer le dessin sur le disque (snapshot), et d'autre fois, l'utilisateur peut la colorer dans l'application de coloriage ! , donc j'ai utilisé les classes de gestion des « Textures 2D » pour retirer les pixels depuis l'écran et les enregistrés dans un fichier PNG avec la conversion des couleurs qui sont proche du blanc vers le type « alpha » (transparent).

Le principe de fonctionnement de dessin est d'utiliser le composant « Trail Renderer », l'exemple d'effets de cet composant est dans les jeux de courses quand la voiture utilise le « Turbo/Nitro », mais dans mon application, j'ai modifié ses paramètres pour le voir comme des lignes de dessin.

## 2.2 Coloriage

Ce jeu est un complément du précédent (Dessin). Ici, l'utilisateur peut choisir entre les images prêtes à peindre, et ses propres images qui ont été enregistré dans l'application de Dessin.

Les outils de cette application sont presque les mêmes, sauf qu'il y a ici un tableau de couleurs, donc l'utilisateur peut choisir par glisser ces doigts dans le tableau et enfin cliquer sur le carreau de la couleur sélectionné. J'ai utilisé pour cela les classes de conversion des coordonnées d'écran vers les coordonnées du jeu (espace) et d'autre pour gérer les pixels d'écran.

Le coloriage des images a deux types :

- **Coloriage Direct** : Les parties d'image sont séparés, donc l'utilisateur a besoin juste de taper sur la partie qu'il veut peindre, et il va peindre.
- **Coloriage Libre** : Dans ce type, le coloriage est juste comme le dessin mais avec des couleurs. Généralement, ce types est utilisé pour les images enregistrés par l'application de dessin (car je ne peux pas les mètres des parties séparés).

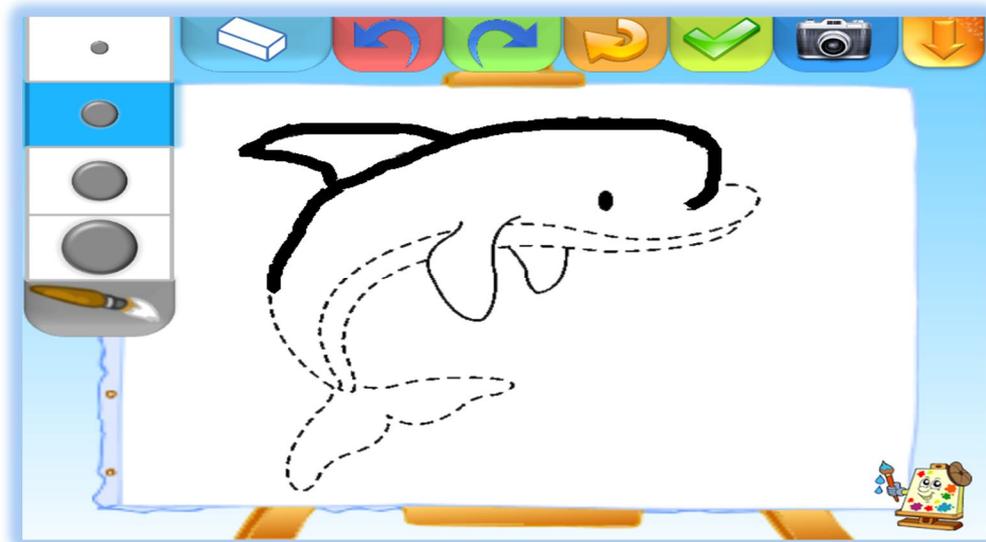


Figure 3 –Dessin



Figure 4 –Coloriage

### 2.3. Lettres

« Lettres » est une application qui permet aux enfants d'apprendre les lettres arabes par un moyen qui a un peu de créativité et d'interaction entre l'enfant et l'application.

#### 2.3.1 Prononciation des lettres

Cette application est utilisée pour enseigner aux enfants les bases des formes et les prononciations des lettres où en les parcourant automatiquement en tapant sur le bouton « Play » ou simplement par taper sur la lettre (Simple & Enough).

#### 2.3.2 Apprentissage des lettres

Ici l'enfant peut pratiquer l'écriture des lettres, donc il peut voir comment la lettre doit être écrite (avec une belle animation), et de la réécrire, il peut aussi entendre le son de la lettre, et passer d'une lettre à l'autre. Une fois, il a eu l'idée, il peut passer un test, ce dernier possède quelques règles:

- Il ne faut pas écrire en dehors de la lettre.
- Il faut respecter le nombre où on doit lever le doigt.
- Il faut suivre les points bleus pour l'écriture correcte de la lettre.

Après quand il passe successivement le test, il verra une image d'un animal avec sa prononciation et un son.

On note que les animations des lettres faire 28 fois, le même nombre est pour les scènes de chaque lettre, et pour les tests (28+28+....) qui est beaucoup de travail.

#### 2.3.3 Quizz des lettres

Pour finir cette série d'applications des lettres, je donne l'enfant quelques tests sur les formes des lettres, de prononciations et des images d'animaux et ses sons.

Il faut signaler que ces derniers jeux n'ont pas requis un grand effort technique considérable, mais présente une bonne production utile dans l'ensemble.

- **Prononciation des lettres:**

Ceci est une application simple qui montre l'enfant une lettre et lui demande de trouver la prononciation en donnant un ensemble de choix. L'enfant peut écouter les choix autant de fois qu'il le veut, et puis il confirme son choix. Si la réponse était juste, il peut passer à la lettre suivante, et il devrait recueillir 10 points pour terminer le test.

- **Sons des animaux:**

La même application que la précédente, mais cette fois il a des animaux au lieu de lettres.

- **Formes des lettres :** (le même principe).

- **Images des animaux :** (le même principe).

- **Sauvage ou Domicile :** Donc, l'enfant peut terminer son apprentissage avec cette dernière application qui demande de lui de penser si l'animal est sauvage ou domicile.



Figure 5 –Lettres

## **2.4 Animaux avec la réalité augmentée**

Cette application est un peu de spéciale ! Car elle est basée sur la nouvelle technologie AR (**Augemented Reality**) qui a le principe d'ajouter des choses irréels dans un monde réel, donc on peut voir notre monde de façon différente.

### **2.4.1 Définition de la réalité augmentée**

La réalité augmentée est une vue directe ou indirecte d'un environnement physique, le monde réel dont les éléments sont augmentée (ou complétée) avec des effets sensoriel généré par l'ordinateur telles que le son, la vidéo, des graphiques ou des données GPS, et elle est (cette technologie) applicable dans beaucoup de domaines tel que les domaines de : Archéologie, Architecture, Art, Commerce, Construction, Education, la gestion des urgences, les jeux, design industriel, Beauté , Médical, Militaire, de navigation, Bureaux, Sport, Télévision, Traduction ....[8]

### **2.4.2 RA et Unity**

L'utilisation de cette magnifique technologie est possible avec le moteur des jeux « Unity » par l'utilisation d'une extension développer depuis la société « Qualcomm » dans un produit appelé « Vuforia » (elle a aussi un SDK (Software Development Kit) spécial pour les gens qui développent sous « Eclipse » ).

Elle utilise la technologie de vision par ordinateur pour reconnaître et de suivre les images planaires « Image Targets » et des objets 3D simples, tels que des boîtes, en temps réel.

### **2.4.3 RA dans mon application**

J'ai utilisé cette technologie deux fois dans mon application (J'espérais faire plus, mais je ne l'ai pas eu le temps) dans la catégorie des enfants (les animaux) et dans la catégorie de culture général (exemple de chimie).

Dans cette catégorie, l'application doit allumer l'appareille photo et attendre jusqu'à elle reconnaître une image d'un animal (depuis les cartes que j'ai créés spécialement pour cette application). Une fois la reconnaitre est satisfait, un model 3D de la lettre pour cet animal va montrer, et avec l'utilisation des boutons de cette carte, l'utilisateur peut changer le model vers un animal avec la prononciation. J'ai créé des cartes pour les animaux et d'autre pour les contrôles, tout cela sera vu réellement ultérieurement.

On note que j'ai créé les modèles 3D des lettres et mots avec « 3D Studio Max », et les carte juste avec MS « PowerPoint » et « Word ».



**Figure 6 – Animaux avec la Réalité Augmenté**

### **3. Portail de réflexion**

Cette catégorie contient deux jeux de réflexion qui sont :

#### **3.1. Chemin Eulérien**

Le principe de ce jeu est d'appliquer les règles qui sont dans un chemin eulérien, mais avec des modifications (pour mettre le jeu plus interactif). Les règles sont :

- L'utilisateur doit passer à travers toutes les lignes (arcs) une seule fois.
- Il peut passer à travers les points d'autant qu'il veut.

Le jeu a presque **98 différents niveaux** avec un système ascendant de difficulté. Dans chaque niveau qui a quelques nouvelles conditions (règles) l'utilisateur peut obtenir une aide en temps réelle avec une belle animation et des instructions claires.

Le système de création de ligne basée sur l'utilisation du composant « Line Renderer » et le mouvement du point est avec les classes de coordonnées d'écran et la conversion.

Les règles supplémentaires (par apport les règles original du chemin eulérien) commencent à partir du niveau 35 à 58 et 80:

- Règles de niveau 1:
- Sont les mêmes que les règles de chemin Euler démontré ci-dessus.
- Règles de niveau 35:
- L'utilisateur aura un ou plusieurs « **lignes rouges** » ce qui signifie qu'il doit passer à travers cette ligne **deux fois** (donc la solution à penser sera changé).

- Règles de niveau 58:
- Ici, l'utilisateur sera plus limité par des **lignes orientées**, alors il doit passer à travers cette ligne avec une **seule direction**.
- Règles de niveau 80:  
Ceci est une sorte de magie, afin que l'utilisateur ait deux points bleus qui fonctionnent en touchant l'un d'entre eux de sorte qu'il lui apporte à l'autre (comme un portail).

Notez que ces règles peuvent être combinées ensemble.

Il ne vient pas de la sagesse que je crée 98 niveaux sans créer un **système de construction des niveaux** complet pour m'aider dans la création de ces 98 niveaux. Donc, j'ai fait ce système qui a le principe suivant : on doit jouer le niveau avant de donner à l'utilisateur pour jouer ! c.à.d. j'ai créé des fonctions spéciales pour chaque type de relation entre les points (relation normal, ligne rouge, ligne orienté, ou un portail), chaque fonction s'exécute avec une ou des clés spéciales dans le clavier, donc je dois juste imaginer le niveau et le jouer avec ces clés spéciales et enfin l'enregistré !

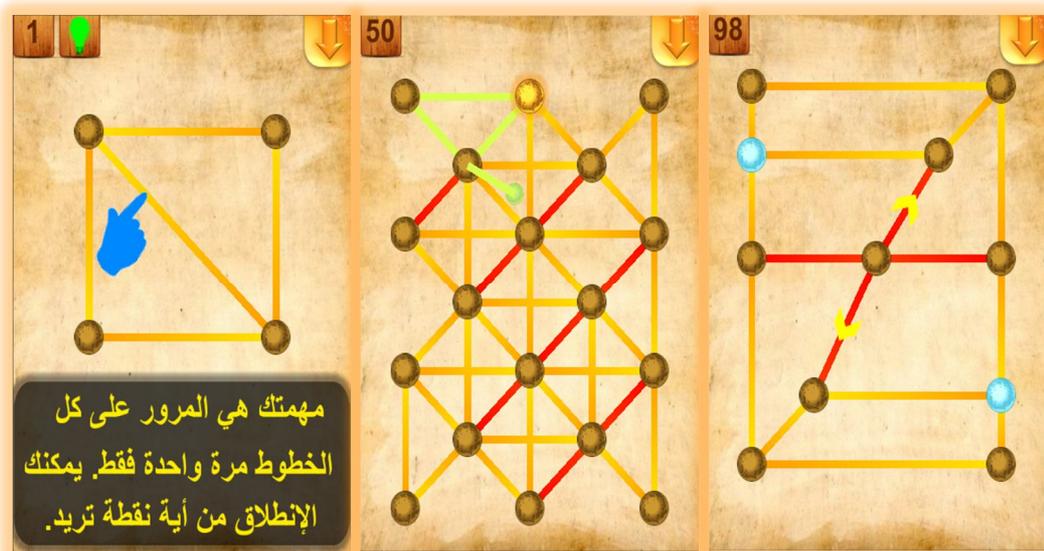


Figure 7 – Chemin Eulérien

### 3.2. Cerveau Vif

« Cerveau Vif » est une application qui vous permet de faire des tests pour votre cerveau (comme des tests de QI) avec un système complet de formation et quelques autres bons jeux de défi de cerveau.

L'application rassemble le top **5 des catégories** de travail du cerveau qui sont: Logique, Visuel, Concentration, Math et Mémoire. Chacune de ces catégories a environ **2 à 4 tests différents**, donc en total de 17 jeux différents !

#### ❖ Test Complet :

L'utilisateur peut commencer avec le mode « Test Complet », donc il aura des tests complets pour tous les 5 catégories, chaque test de ces est limité par le temps, et à quelle vitesse le joueur répondre. Une fois qu'ils sont fait, un schéma de statistiques apparaît et montre le résultat du travail du cerveau et combien des points l'utilisateur a obtenu pour chacune de ces catégories.

#### ❖ Entraînement :

Ce mode est en fait pour pratiquer les tests que l'utilisateur a déjà passés (avec de bons points, alors il les a déverrouillés). Tant que ce mode est pour l'entraînement, les tests ne sont pas limités par le temps (temps illimité).

#### ❖ Tests Spéciaux :

Le mode « Tests spéciales » est un sorte de « Bonus » pour le joueur qu'il peut apprendre quelques bonnes astuces du cerveau et sait que nos cerveaux trichent parfois à nous.

Voici une démonstration rapide de chacun de ces jeux :

#### 3.2.1 Logique

On a 4 jeux différents pour cette catégorie :

#### ❖ Balance :

Le but de ce jeu est de sélectionner l'élément le plus lourd basé sur votre logique (mais pas de votre apparence et expérience). Le joueur peut voir les instructions avec des graphiques, et a les choix apparaissent en bas.

J'ai fait un algorithme pour avoir tous les cas possibles, et j'ai finis avec 12 cas différents. Exemples: x H y (x lourd que y), x H y et y H z, ...

Ces cases sont générer aléatoirement, et les objets sont groupés par leurs catégories.

❖ Expressions :

Ceci est un jeu très simple qui génère une expression (mots) et demande à l'utilisateur de choisir l'image adéquate.

Le jeu n'a pas besoin de certains algorithmes compliqués ou de travail, il génère tout simplement une expression de façon aléatoire, puis reprend l'image juste pour lui, la mettre dans les choix et remplir le reste des choix avec d'autres icônes aléatoires différents de celle sélectionnée.

❖ Règle Magique :

Dans ce jeu, le joueur (qui est paresseux en quelque sorte) devrait pousser ses compétences logiques à leurs limites. Donc, une liste de trois éléments de couleur apparaît à gauche de l'écran, puis ils seront changés dans la couleur et / ou la forme à une nouvelle liste (à droite), mais l'un d'eux sera éliminé, ce qui signifie que la tâche du joueur sera déterminer l'élément manquant. Il va donc essayer de trouver les changements et les comparer avec la première liste (ancienne). La réponse devrait être aussi vite que possible.

❖ Reflet :

Dans ce jeu, l'utilisateur n'utilise pas seulement ses compétences logiques, mais ceux de son imagination aussi. Le jeu basé sur la génération d'une figure aléatoire avec couleur et rotation aléatoire aussi, et en appliquant ensuite une réflexion selon un axe aléatoire (peut être régulière ou non). Cela semble très simple de le faire, mais au moment de jouer, le joueur peut être confondu et surtout lors de l'obtention des figures étranges.

### 3.2.2 Visuel

Dans cette section, le joueur peut obtenir des tests pour développer ses aptitudes visuelles et de comparaison.

❖ Mosaïque:

Le joueur obtient une liste ordonnée de formes recueillies à la gauche de l'écran, et son travail est créer une nouvelle liste avec le même ordre de la première. Pour le travail technique, ce n'est pas difficile de le faire, mais au moment de jouer, cela peut être difficile à résoudre et de trouver le bon résultat parce que j'ai roulé ensemble des figures spéciales. Les figures comptent dans la liste peut être 2 ou 4, et tout cela est généré aléatoirement.

❖ Croissant :

« Croissant » est un jeu qui demande à l'utilisateur de choisir des objets de même couleur du moins au plus. Encore une fois, ce ne sont pas sur le travail technique, car il génère tous simplement des objets colorés aléatoirement (chaque couleur a sa part dans l'écran), puis attend la réponse. Mais pour le joueur, il vraiment poussé ses capacités visuelles à ses limites, parce qu'il est très difficile de le résoudre!

### 3.2.3 Concentration

L'une des compétences les plus nécessaires dans ces jours est la Concentration. Alors allons-y pour démontrer certains jeux qui aident à grandir cette partie du cerveau!

❖ La Portée Maximale :

L'objectif du jeu est de sélectionner l'élément le plus éloigné, et il commence avec une belle décoloration et animation de glissement pour afficher les éléments à choisir, et puis une autre belle animation pour le mouvement des objets avec une vitesse différente, pour mettre le joueur un peu confus. Les animations de mouvement des objets ne sont pas faites avec le système d'animation de « Unity », mais tout est développé et programmé pour contrôler la vitesse et dans quelle mesure les éléments bougent.

❖ Chiffres :

« Chiffres » est un autre test de concentration. Donc, le joueur va essayer de trouver le nombre montré en comparant ses chiffres avec les choix qu'il a obtenu. Mais ce n'est pas facile qu'il y paraît, car les chiffres ne sont pas montrés par ordre, et ils sont affichés/cachés avec une vitesse différente pour chaque nouveau test, et les choix sont tous semblables dans les chiffres. Donc, le joueur devrait travailler avec ses compétences de concentration plus que ses compétences mathématiques ou visuelles.

### 3.2.4 Maths

Voici la liste des jeux de cette catégorie :

❖ Les Suites :

Cette application est l'un des meilleurs jeux simples et assez qui forcent le joueur à calculer rapidement. Il est basé sur la « suite » connue en math, mais avec tel créativité et amusement. Le joueur aura notifié le chiffre qu'il doit utiliser et le point de départ, et il devrait ajouter / soustraire cette chiffre en suivant l'une des directions disponibles (haut / bas / gauche / droite) jusqu'à ce que la route est plus ou le nombre de répétitions est terminée. Les routes sont générées aléatoirement entre -9

et 9, et les numéros de carreaux sont de 1 à 99. J'ai créé un bon algorithme pour trouver tous les cas possibles de carreaux vides selon la position actuelle du joueur.

❖ Arithmétiques :

« Arithmétiques » est une application de calcul très simple qui donne à l'utilisateur une équation simple (addition / soustraction / multiplication / division) à résoudre, et le but sera d'écrire le nombre manquant / résultat.

Les opérateurs sont générés aléatoirement, avec le respect des cas spéciaux comme :

- Dans la soustraction, nous ne devrions pas avoir un résultat d'un nombre négatif.
- Dans la division, nous ne pouvons pas diviser par zéro.

❖ Distance :

La première fois quand le joueur voit le jeu, il pense qu'il est un jeu amusant (avec Tom & Jerry graphiques et animations), mais au moment de jouer, il sera choqué la réalité! Ceci est un jeu très dur! Le jeu est basé sur le calcul de la distance entre deux points (position de Tom et position de Jerry), mais les positions sont en unités de m/km, de sorte que certaines fois, il devrait convertir de mètre en kilomètres pour trouver le résultat, et les choix sont tous dans l'unité « mètre ».

### 3.2.5 Mémoire

En jouant les jeux de cette catégorie, l'utilisateur peut avoir des progrès sur sa capacité de mémoire et ses compétences.

❖ Flèches :

« Flèches » est un jeu très bon et facile pour l'entraînement de la mémoire. Tout d'abord le jeu sera montré à l'utilisateur un chemin à suivre, ce chemin est généré avec un algorithme similaire de « Suite » mais cette fois nous avons **25** carreaux ce qui signifie que nous avons **25 cas** à traiter, mais je l'ai fait un autre algorithme afin de simplifier et **réduire ce nombre de cas jusqu'à 9 cas**, voici un exemple du principe d'algorithme : supposant que je suis dans le carré numéro 7, pour aller à le carré supérieur, je dois soustraire 5 (car la distance entre les lignes est 5 carreaux), et pour obtenir le carré gauche / droite je dois ajouter / soustraire 1.... Les directions sont générées au aléatoirement, selon les carreaux libres. Le chemin peut être fait si le nombre de répétitions est terminé ou s'il n'y a pas de moyen d'ajouter plus de carreaux.

❖ Tuile Manquant :

Ceci est un autre jeu pour l'entraînement de la mémoire, mais il est un peu plus difficile que le précédent. Le principe du jeu est qu'il montre au joueur d'abord 2 à 6

éléments, et puis il va supprimer 1 à 4 éléments, et lui demande de les sélectionner parmi les choix disponibles. Alors ceci est un jeu très dur pour ceux qui ont une mémoire paresseux ou parfois même pour ceux qui ont une bonne mémoire!

❖ **Ordre :**

Vous pouvez voir que « Ordre » est un jeu similaire en difficulté avec « Tuile Manquant ». Alors, cette fois, l'utilisateur verra 2 à 4 objets colorées pour un moment (le temps d'attendre varie selon la difficulté qui est de façon aléatoire), puis tous ces éléments seront cachés, et le joueur devra trouver le N ième élément (N = 1..4). Donc, dans les choix, je mets la réponse correcte, et d'autres objets qui peuvent être dans la liste cachée ou non, et peut-être juste la bonne réponse mais avec une couleur différente ... (juste pour rendre le jeu un peu plus difficile, et le joueur un peu confus).

### **3.2.6 Statistiques de cerveau**

Après avoir passé tous les tests (de « Test Complet »), les cartes du cerveau apparaisse. Dans cette scène, le joueur peut obtenir une analyse statistique de ce qu'il a fait plus tôt. Cela fonctionne en stockant les points de chacun des tests passés, et les montrer ensuite avec un joli diagramme coloré et animé avec le compte des points pour chaque test, et le total des points ; donc ces sont pour les modes « Test Complet » et « Entraînement ». Allons y de sauter maintenant aux les « Tests spéciaux » :

### **3.2.7 Tests Spéciaux**

❖ **Visible mais Caché :**

Pour ce test, j'eu l'idée du canal de « Nat Geo ». Le test fonctionne essentiellement en mettant 4 points oranges et en appliquant un peu de bruit spécial (points bleus déplacer / affichage / masquage rapide dans toutes les directions), et le travail de l'utilisateur est de se concentrer sur une et un seul des points oranges. Après quelques instants, il devrait constater que l'un ou de tous les autres points oranges a disparu! S'il peut voir ça, il passera le test (ses compétences de concentration est bonne), sinon, il doit passer le test à nouveau.

Notez que les effets de bruit sont faits avec un script intégré avec « Unity ».

❖ **Lecteur d'Esprit 1 :**

« Lecteur d'Esprit 1 » est un autre test l'idée est prise à partir de « Nat Geo », mais l'algorithme a été refait dans son intégralité. En donnant au joueur 13 cases numérotées, et en lui demande de choisir l'une d'elles (dans son esprit seulement), et

chaque fois les règles seront modifiées en fonction des progrès. A la fin, le joueur trouvera à chaque le même résultat ! L'algorithme est basé sur un calcul mathématique que si un utilisateur a pris un certain nombre, il va finir avec le nombre 5 (j'ai choisi de finir avec le numéro 5). Les instructions de jeu et les progressions des boîtes (détruit) sont réalisées par programmation.

❖ Lector d'Esprit 2 :

Ceci est un jeu délicat. Il permet au joueur de choisir l'une des cartes montrées (dans son esprit seulement) et les cartes seront affichées - avec un graphisme et une animation- puis il sera demandé de brouiller les cartes (juste pour mettre le jeu un peu interactif), et finalement la carte choisie va être détruite et le résultat va être montré ... et elle a vraiment disparue ! Alors, comment j'ai fait ça ?... ok, l'astuce est très simple, je viens de changer toutes les cartes qui sont montrés pour la sélection au début du jeu, alors pour n'importe qu'elle carte l'utilisateur a choisi, sa carte sera changée!

Et ici notre tour avec « Cerveau Vif » sera terminé !



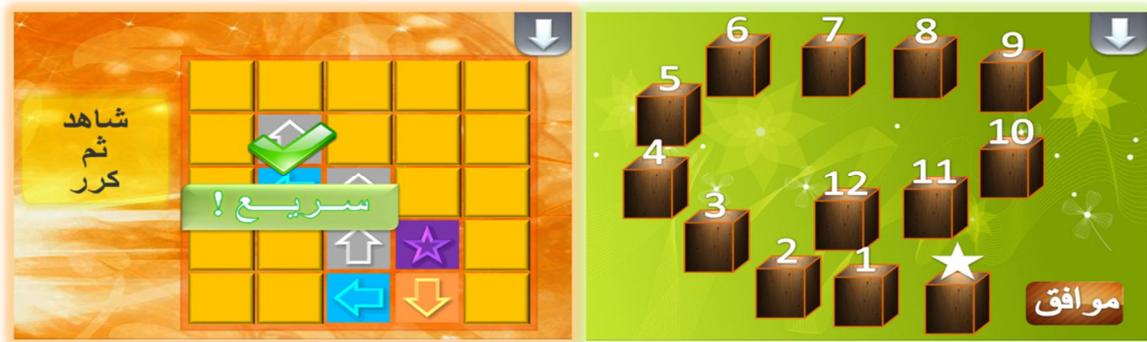


Figure 8 – Cerveau Vif

#### 4. Portail de Culture Générale

Les jeux de cette catégorie sont :

##### 4.1. Mots

« Mots » est un jeu de mots basé sur le mot le plus long que l'utilisateur peut collecter. Le jeu commence en choisissant l'une des trois langues disponibles qui sont: **anglais, arabe, ou français.**

Le processus du jeu est en associant des lettres dans toutes les directions, la ligne ne doit pas être forcément stricte !. Les mots pour les comparer avec sont chargées à partir d'une base de données locale « SQLite », que j'ai intégré dans « Unity ».

##### Remarque :

On note que « Unity » n'est pas un EDI de source ouvert -Non Open Source IDE-, donc il très difficile d'intégrer des bibliothèques externes, et ce que j'ai ajouté est une sorte d'un processus difficile.

La base de données « SQLite » est très similaire que la base de données SQL, mais elle est plus optimisée, et j'ai géré mes tableaux avec le logiciel « SQLite Browser ». Le jeu n'a pas d'autres règles que le mot doit être connu avec une minuterie. Les lettres sont générées aléatoirement, et chaque fois que le joueur trouve un mot, les lettres de ce mot seront détruites et faire instantiation d'autres nouvelles dans leurs colonnes jusqu'à ce que ces colonnes être remplis. J'ai dit que les lettres sont générés aléatoirement, mais avec un certain filtrage, de sorte que les lettres les plus communes et utilisés auront plus de chance d'être instancié que les autres (j'ai définir 3 niveaux de priorité). Donc, une fois que le joueur enlevé son doigt depuis l'écran, je lance une requête SQL pour comparer le mot collectée avec les mots de la table de langue sélectionnée dans la base de données. Il y a un outil pour détruire les lettres bannies,

cet outil peut être utilisé pour un temps limité (5 fois). Chaque fois que l'utilisateur recueille un mot identifié, le nombre d'utilisation de cet outil sera augmenté.



Figure 9 - Mots

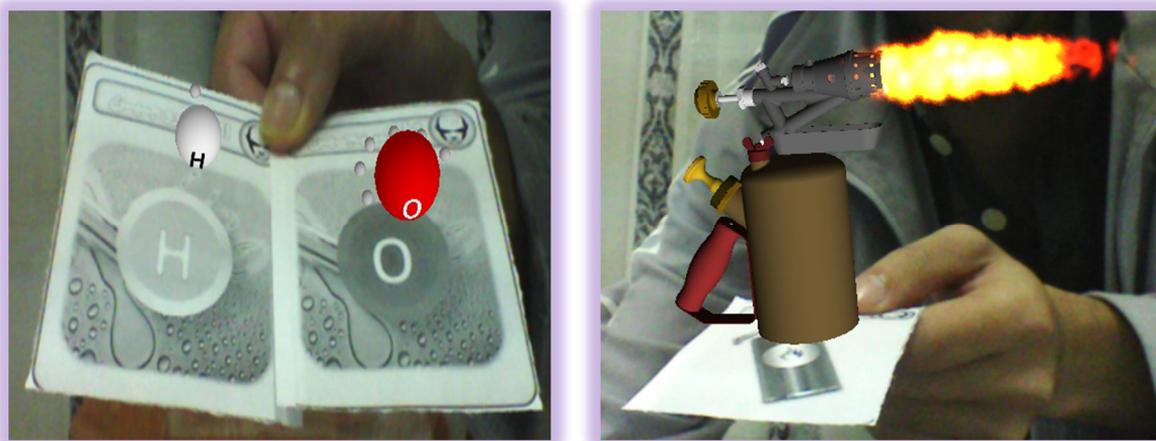
#### 4.2. Chimie avec RA

Voici une autre application réalisée avec l'utilisation de la technologie « AR ».

Cette application est de démontrer **juste un exemple** de qu'est qu'on peut faire avec cette technologie dans le domaine de Chimie.

Au début, l'utilisateur va choisir d'afficher les indications ou non, après il va suivre le progrès d'utilisation de l'application. Ce progrès est démontré comme ceux-ci :

- Il faut placer la carte d'oxygène devant l'appareil photo pour la reconnaître.
- Après, on place les deux cartes de hydrogène.
- Dans ce moment, les deux noyaux d'hydrogène seront liés avec l'oxygène, cette opération nous me donne l'EAU.
- Maintenant, vous pouvez voir l'eau avec son forme réel par approcher lui de la camera. L'eau générée a des effets presque réels comme le reflet et le mouvement.
- Vous pouvez continuer le progrès par placer la carte de « froide » pour réaliser un frigo, et mettre l'eau dans lui... dans quelques instants (5 sec) l'eau va être glacé !
- Maintenant, pour remettre la glace sous forme d'eau il faut la fuser, et vous pouvez faire ça avec la carte de « fumé ».



**Figure 10 – Chimie avec la Réalité Augmenté**

### **4.3. Empire de Connaissance**

Un des jeux les plus durement codés que j'ai réalisé est « Empire de Connaissance ». Ceci est un jeu de mono/multi-joueurs culturel fondée sur l'idée du jeu « Saïf El Maarifa » de Facebook, et le principe est de répondre aux questions et déploie l'empire le plus possible. Le jeu prend en charge 2 modes différents de réseau: **réseau local** et **Internet**, et chacun de ceux-ci possède sa propre implémentation.

#### **4.3.1 Mode individuel ou mono-utilisateur**

Dans ce mode, le joueur peut jouer avec l'ordinateur (système). Le jeu commence par sélectionner le nom et l'avatar du joueur, ainsi que la difficulté et le mode de joue. Les modes de joue sont :

- Mode 4 tours : donc le jeu sera terminé dans 4 tours pour chaque joueur (4 attaques).
- Mode de jusqu'à la mort : dans ce mode le jeu ne sera pas terminé sauf qu'un des joueurs contrôle tout le monde (toutes les zones).

Et les niveaux de difficulté sont : facile, difficile, très difficile. Ces niveaux sont pour les réponses de système, par ex : répondre juste ou faux ?, répondre rapidement ou pas ...

#### ❖ Structure générale du jeu :

La structure générale du jeu est comme suite :

- Le jeu va commencer par les sélections des zones libres dans la carte, la sélection se fait par répondre aux questions de type « vrai/faux » (5 questions).

- Après que la carte est satisfaite, la « guerre » va commencer, dans cette étape, les joueurs vont essayer d'attaquer les autres (par ordre), l'attaque peut être à un chevalier ou à un château. Donc nous avons une seule question à répondre de type « choix » dans le cas d'attaque d'un chevalier, et 3 questions pour l'attaque successive d'un château. Si les réponses sont les mêmes (toutes justes) une autre question va être ajoutée et de type « nombres » pour séparer le gagnant. Le temps de réponse est aussi compté.
- Une attaque successive sur un château mettra hors le jeu l'adversaire, et tous ses zones vont être récupérer.
- Afin que le jeu se termine (selon le mode), un classement par points va être affiché.

❖ Structure détaillée :

Voici les détails pour chaque fonctionnement de la structure générale :

Après que le jeu a commencé, l'écran va affichée les éléments suivants :

- Le nom, avatar, couleur, et score du joueur.
- Les mêmes éléments pour l'adversaire.
- Cet adversaire (le système) aura des noms aléatoires (d'hommes ou de femmes), et d'avatars aléatoires aussi.
- Des boutons pour changer la difficulté en temps réel (au moment de jouer).
- Des boutons de contrôles (son, menu principale).

➤ Distribution des zones de la carte :

Le jeu commence par des indications pour chaque étape le joueur va être dans. Premièrement, les châteaux de chaque joueur vont être sélectionnés automatiquement et aléatoirement. Par défaut, le joueur a la couleur rouge. Et après ça, les questions de type « vrai/faux » vont être générées avec une belle animation et un minuteur.

Les questions sont générées depuis une base de données SQLite (j'ai déjà parlé de la dans l'application « Mots ») que j'ai une grande difficulté de collecter (le contenu arabe est très peu).

Dans chaque tour, les joueurs qui ont réussies à répondre correctement se diviseront les chevaliers (3 par tour), et les placer dans la carte en n'importe qu'elle zone libre.

Notez que la création de la carte du monde est un processus très difficile (je devrais télécharger une carte et la diviser et de la faire pour 2 couches (carte normal des effets d'ombres pour la voire comme en 3D, et une couche pour la sélection (carte blanche),

et de générer des animations, des collisions, scripts, noms, fils (drapeaux, souligne ...), ... pour chaque partie (région) de cette carte (superficies totales = 17)).

➤ La Guerre :

Les joueurs vont attaquer l'adversaire par ordre, et l'indication est en bas de l'écran. Le résultat de l'attaque entre les joueurs peut être limité avec ces 2 \* 3 cas:

☞ Le joueur attaque un chevalier:

Le joueur est avec succès attaqué son adversaire (la réponse du joueur est juste et de l'autre est faux): la zone de l'adversaire sera ajoutée aux zones de l'attaquant, et les points seront ajoutés à l'attaquant.

- L'attaque de joueur a échouée (la réponse du joueur est fausse et de l'autre est juste): le score de l'adversaire sera augmentée par 100 points.
- Une équivalence entre le joueur et l'adversaire (joueur et l'adversaire répondent à la fois juste ou aucune soumission): une autre question sera générée, mais cette fois, la réponse est plus exacte, donc sera un certain nombre (peut-être l'année, compter, ...). Dans ce cas, nous aurons trois autres cas:
- La réponse de l'attaquant est plus près de la bonne réponse que l'adversaire: l'attaquant est attaqué avec succès (il se tournera vers le premier cas).
- La réponse de l'attaquant est loin de la bonne réponse que l'adversaire: l'attaquant a un échec (se tourner vers le cas second).
- Les réponses d'attaquant et l'adversaire sont les mêmes: ici, je vais mettre la bonne réponse au joueur qui a répondu rapidement.

☞ Le joueur attaqué un château:

- Une attaque réussie : les points de l'adversaire du château seront décrémentés avec 350 points (-350 pts), et une nouvelle question sera générée.
- Un échec d'attaque : les points de l'adversaire du château sera augmentée avec 100 points (100 pts), et la génération de questions seront arrêter (tour de joueur sera finis à ce point).
- Une équivalence entre le joueur et l'adversaire: le même que dans le cas de l'attaque chevalier.

Si les points de château de l'adversaire sont égaux à zéro, le jeu sera terminé, et la fenêtre des classements sera affichée.

### 4.3.2 Mode Multi-joueurs

J'ai utilisé deux types de réseau dans ce mode : réseau locale, et Internet. Mais avant de parler des, il faut définir quelque notions importants communes.

#### ❖ Approches de réseau :

Il y a deux approches communes et éprouvées pour la structuration d'un jeu en réseau qui sont connus comme un « **Serveur Autorité** » et un « **Serveur Non Autorité** ». Les deux approches reposent sur un serveur qui se connecte avec des clients avec l'utilisation des adresses IP et des ports.

#### ➞ Un Serveur Autorité :

Le serveur doit effectuer toutes les simulations du monde (jeu), l'application des règles du jeu et de la transformation de l'entrée des clients. Chaque client envoie leur entrée (sous la forme de frappes ou des actions demandées) au serveur et reçoit en permanence l'état actuel du jeu à partir du serveur. Le client ne fait jamais de changement. Au lieu de cela, il indique au serveur ce qu'il veut faire, et le serveur se charge à partir de la demande, et répond au client ce qui est arrivé par la suite.

#### ➞ Un Serveur Non Autorité :

Un serveur non-autorité ne contrôle pas le résultat de chaque entrée de l'utilisateur. Les clients eux-mêmes traitent l'entrée de l'utilisateur et la logique de jeu localement, puis envoient le résultat de toutes les actions déterminées au serveur. Le serveur synchronise ensuite toutes les actions avec l'état du monde.

Notez que j'ai utilisé la méthode d'un **serveur autorité** pour faire la synchronisation entre les appareils et surtout lors de la présentation des réponses, mais pour les animations et autres effets j'ai utilisé la deuxième méthode, car on n'est pas obligé de faire les depuis le serveur. Donc j'ai utilisé habituellement les deux méthodes pour obtenir des meilleurs résultats.

La connexion Client-Serveur peut être un processus complexe. Les machines peuvent avoir des adresses IP privé ou public et ils peuvent avoir des pare-feu locaux ou externes bloquent l'accès.

Revenant à notre jeu, en choisissant un nom et un avatar depuis le menu principal dans le mode de multi-joueurs, vous pouvez créer un nouveau serveur pour le jeu, où vous pouvez rejoindre un serveur qui est déjà créé depuis la liste des serveurs actuels.

#### 4.3.2.1 Réseau locale

##### ❖ Création du Serveur :

La solution de « Unity » pour le réseau est basé sur la connaissance de l'adresse IP et le port du serveur, puis l'utilisation des classes spéciales pour les communications, et la création du serveur se fait en choisissant un port, nombre de joueurs maximal, et l'état NAT (Network Address Translation qui utilise un serveur partagé pour la communication de manière à ce que l'adresse privée peut être atteinte à partir de l'adresse publique, car les machines ne peuvent pas être connectés avec les adresses IP privés). Dans ma solution de jeu, le processus de création de serveur est (d'abord) basé à remplir les informations suivantes:

- Le nom du jeu. (Pour permettre à d'autres joueurs de le voir et de le différencié dans la liste des serveurs).
- Un mot de passe (optionnel) pour protéger votre jeu d'accéder par d'autres joueurs détestés.
- Le nombre maximal des joueurs (max = 4).
- Mode de jeu: « 4 Tours » ou « Jusqu'à la mort » (j'ai déjà les expliqués).

En tapant sur « Accepter », le serveur sera créé avec la méthode habituel de « Unity », mais nous sommes dans un réseau local sous WIFI (communication avec Wifi-Direct), de sorte que les autres dispositifs devrait être notifié qu'un serveur a été créé, et recevoir ses informations (et surtout son adresse IP). Dans « Unity », cela est impossible à faire. Donc j'ai cherché et essayé de proposer ma solution pour ce problème, et je me suis retrouvé avec le suivant ...

##### ❖ Découverte du Serveurs :

Ma solution pour découvrir la liste des serveurs est basée sur l'utilisation du protocole UDP (sockets).

##### ✓ C'est quoi UDP ? :

UDP est un protocole de communication (client-serveur) entre les machines pour envoyer/recevoir des informations (paquets). Il supporte quelque services supplémentaire tel que « Multicast » (envoyé un seul paquet vers plusieurs machines), ... Mais manque d'autres services comme la sécurité, renvoyé les paquets perdus,...

✓ Pourquoi UDP ? :

J'ai choisit d'utiliser le protocole UDP parce que je ne cherche pas à envoyer des tonnes de données, je suis juste envoi un seul message (string, avoir le nom du serveur, Mot de passe, Mode de Jeu, Nombre de joueurs connectés) par second sans « Threads » ou autre méthodes de synchronisation.

L'intégration de ce type de bibliothèques (bibliothèques système) dans « Unity » est un processus facile, mais leur utilisation avec « Unity » est très difficile, car il est un IDE fermé, donc lors de l'intégration de quelque chose d'extérieur, vous devriez avoir beaucoup d'attention et vous devez savoir exactement comment « Unity » fonctionne (développement avancés).

Quand le serveur a été créé, il continuera à envoyer ses informations vers le réseau, et stopper la réception depuis les autres serveurs (dans un premier temps, la machine va réceptionner les informations d'autres serveurs avant la création d'un nouveau). Dans le côté client, il va réceptionner les informations au même temps.

L'envoi/réception des données d'UDP sera arrêtée dans ces cas:

- Lorsque le serveur annule le jeu (à la fois, le serveur et les clients arrêtera d'envoyer / recevoir des messages, et la scène sera rechargé).
- Lorsque le client se connecte à un serveur (donc il n'a pas besoin de chercher d'autres serveurs).
- Lorsque le jeu commence (tous les appareils connectés seront arrêter d'envoyer / recevoir des messages, et la connexion UDP sera fermé).
- Si nous sommes en mode LAN et nous avons changé de mode Internet.

J'ai créé en plus un petit algorithme pour avoir si le serveur est déconnecté ou pas (dans le côté client).

Une fois le joueur (client) sélectionne un serveur pour le connecte avec lui, il va être connecté par les classes et les méthodes de communications de « Unity », et je stock lui dans la liste des joueurs connectés.

Avant de présenter aux structures du jeu, laissez d'avoir l'autre mode de réseau : Internet.

#### **4.3.2.2 Internet**

Dans ce mode, et une fois que le joueur choisit un nom et un avatar, il commence à recevoir la liste des serveurs via Internet.

La méthode connue dans « Unity » pour faire la communication via Internet est d'utilisé que s'appelle « Master Server ».

❖ Master Server :

C'est un très grand serveur de « Unity » qui contrôle tous les communications des jeux créés par « Unity ». Donc tous les gens qui utilisent « Unity » peuvent utiliser ce serveur pour développer leurs applications.

Mais je n'ai pas utilisé dans mon application parce que:

- « Master Server » est créé depuis « Unity » juste pour la raison de tester.
- Il ne garantit pas l'unicité du nom de jeu (peut-il existe un ou plusieurs jeux qui ont le même nom).

Alors je suis allé vers l'idée de créer une base de données MySQL (via un serveur web) et certaines pages PHP (chacune de ces pages exécuter des requêtes SQL spécifiques), puis en créant un script (dans « Unity ») qui se connectent à ces pages PHP pour obtenir des informations depuis la base de données MySQL.

La base de données a une seule table qui a les champs:

sIP: varchar (Adresse IP du serveur).

sName: varchar (Nom du serveur).

PW: varchar (mot de passe).

pNb: int (nombre de joueurs connectés).

pMax: int (nombre maximal de joueurs).

gType: varchar (Mode de jeu).

Et il ya 3 pages PHP:

- GetServersList.php:

Il combine chaque information de serveur dans la base de données avec un caractère spécifique et puis combine tous ces informations (de tous les serveurs), et à la fin on aura une chaîne de la liste des serveurs. Donc j'utilise ce dernier pour montrer à l'utilisateur la liste des serveurs (comme dans le réseau local).

- Server\_SignIn.php:

Ce script est pour l'entrée de serveur, si le serveur existe déjà (je lance une requête pour le vérifier), il sera mis à jour (avec une requête UPDATE), sinon, il sera ajouté.

Notez que j'envoie les informations de serveur à travers une forme dans « Unity » (comme les formes utilisés dans HTML), et recevoir les données depuis la fonction \$\_POST en PHP.

- Server\_SignOut.php:

Il exécute tout simplement une requête DELETE pour supprimer le serveur spécifié (par son IP). Je l'utilise ce script lorsque le serveur a été déconnecté (manuellement ou lorsque la connexion est perdue).

#### 4.3.2.3 Structures de communication dans « Unity »

« Unity » a des classes et des méthodes spéciales pour la communication entre les joueurs dans un jeu sous réseau. Mais la méthode la plus utilisée pour la synchronisation de données usuelles (comme la position, rotation, physiques, ...) ou pour l'exécution des actions spécifiées, la méthode est RPC (Remote Procedure Calls).

##### ➤ Remote Procedure Calls :

Il est utilisé pour appeler des fonctions sur d'autres ordinateurs dans le réseau. Les clients peuvent envoyer des appels RPC au serveur, et le serveur peut envoyer à un ou plusieurs clients. Le plus souvent, ils sont utilisés pour des actions qui se produisent rarement. Par exemple, si un client retourne un interrupteur pour ouvrir une porte, il peut envoyer un RPC au serveur lui indiquant que la porte a été ouverte. Le serveur peut alors envoyer un autre RPC à tous les clients, en invoquant leurs fonctions locales pour ouvrir la même porte.[9]

- Les inconvénients de cette méthode sont:
  - Les fonctions appelées devraient être existantes déjà dans les jeux de tous les joueurs connectés (il n'y a donc rien d'extérieur).
  - Vous ne pouvez pas d'envoyer la demande pour un joueur spécifique (vous devez l'envoyer à tous les joueurs, et ensuite tester si tel est le joueur ciblé).
  - A un nombre limité de type de paramètres (qui sont: int, float, string, NetworkPlayer, NetworkViewID, Vector3, Quaternion), donc par exemple: si je veux envoyer mon avatar (qui est une image), je ne peux pas le faire, donc je besoin de stocker tous les images (d'avatars) précédemment (au moment de développement), et ensuite d'envoyer le nom d'image pour le tester avec les images disponibles, pour cela, toutes les ressources doivent être internes.

Notez que ceci est la méthode que je l'habitude de communiquer avec les joueurs dans mon jeu (**je n'avais pas le choix**).

#### 4.3.2.4 Structure de jeu en mode Multi-joueurs

Après que la création du serveur et que le client soit connecté, la liste des adversaires s'affiche dynamiquement, c.à.d. à chaque fois un joueur se connecte ou se déconnecte, la liste applique une mise à jour. Les informations des joueurs collectées sont par l'utilisation de RPC.

Le serveur est l'un qui contrôle où le jeu va commencer. Une fois c'est fait, une nouvelle scène sera chargée et elle sera presque la même dans le mode individuel, mais avec des modifications dans le menu.

L'écran de jeu est composé de:

- Avatar du joueur, couleur, nom et score (en le coin haut à gauche)
  - Bouton «Liste des Joueurs» (lorsque tapant dessus, il apparaîtra une liste de tous les joueurs connectés avec leurs noms, avatars, couleurs, scores)
- Chaque fois le joueur tape sur ce bouton, un appel RPC sera appliqué pour obtenir d'autres informations des joueurs (il obtiendra les nouveaux).

Donc, plus d'explications sur les inconvénients de la méthode RPC est dans ce cas, je veux obtenir la couleur d'autres joueurs, donc je devrais le convertir à partir de type « Color » (qui a la structure: Color (R, G, B, A) à un type de Quaternion (Quaternion (X, Y, Z, W) ou Vector3 (X, Y, Z)) et puis le reconverter à type de « Color » lorsqu'il est reçu.

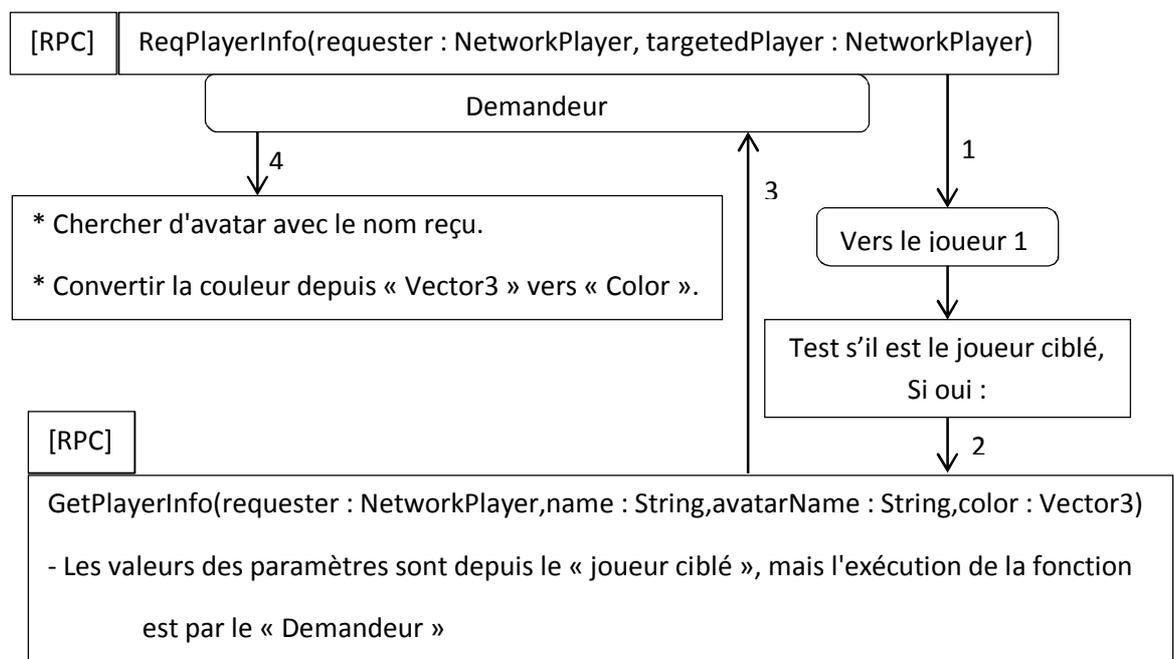


Figure 11 – Structure de Communication avec RPC

La même procédure sera appliquée pour les autres joueurs.

- Un champ de message / bouton pour écrire / envoyer des messages aux autres joueurs.
- Un champ de message reçu avec l'avatar de l'expéditeur et sa couleur, et vous pouvez faire glisser le message horizontalement si le message est long.
- Un bouton de liste des messages reçus indique les derniers 6 messages reçus avec les propriétés que j'ai mentionné précédemment (avatar, couleur, glissement).

Notez que pour une partie multi-joueurs avec seulement deux joueurs est **plus facile** qu'avec plus de 2 joueurs, car dans ce cas, je dois faire **attention à tous les joueurs et leurs cas particuliers**, et aussi les variables simples se tourneront vers des tableaux ...

A ce moment, le serveur est l'un qui contrôle le jeu, c.à.d. générer les questions, décide de quel tâche est la suivant, qui est le gagnant, ... (tous le progrès du jeu).

Donc, le fonctionnement du jeu est déjà expliquer dans la section du mode individuel, mais vous rappelez que le développement et les algorithmes **sont totalement changés !**

Un exemple très simple de la difficulté que j'obtiens au pour développer sous réseau est : **si seulement un variable X va être changé dans un coté (serveur ou un des clients), je dois créer une fonction pour l'envoyer avec RPC !**

#### 4.3.2.5 Problèmes et Solutions proposées

Voici les problèmes principaux que j'ai rencontrés lorsque le travail sur ce jeu:

❖ L'écriture en Arabe:

« Unity » **ne supporte pas le langage arabe**, donc j'ai essayé de travailler avec un plug-in appelé « UnityArabicSupport ». Ce plug-in est fondamentalement attache les lettres les uns aux autres, de sorte que vous pouvez lire le mot correctement. Mais même cela ne résout pas le problème, parce que nous lisons la phrase arabe de la **droite vers la gauche**, et vous pouvez également utiliser ce plug-in juste une **fois par mot** (si un mot est déjà fixé, et que vous appliquez à nouveau une correction, le mot sera déformé). Donc j'ai été forcé de faire **mon propre plug-in**. Mes solutions est un complément du « UnityArabicSupport », donc j'ai fait un algorithme pour

transformer les mots autour pour être lisible comme il se doit, et également un support auto habillage du texte (ce qui signifie que le texte sera retourné automatiquement à une nouvelle ligne quand arriver à une certaine longueur – j’ai eu un problème avec cela aussi, parce que quand j’ai essayé de mettre une nouvelle ligne, il a fait la nouvelle ligne en haut de l’actuel (signifie que nous écrivons de bas au haut)-. J’ai utilisé ce script pour montrer les questions et leurs réponses, mais pour les petites phrases, j’ai généralement utilisé juste le plug-in « UnityArabicSupport » avec quelques modifications (pour résoudre le problème de fixer le mot plusieurs fois).

❖ Déconnexion des joueurs :

La déconnexion du joueur est une sorte de coup de foudre pour le développeur, car il devrait revoir tout son travail pour être mis à jour avec cette modification. Donc, je souhaite que j’ai couverts tous les cas et je n’ai pas manqué aucun cas. Avant de décrire les cas, je dois mentionner que quand un joueur est connecté à un serveur, il obtiendra un ID selon l’ordre qu’il connecté avec, donc pour le serveur, son ID = 0, le premier joueur connecté ID = 1,... par exemple, le dernier lecteur connecté ID = 2, et un joueur a été déconnecté avec l’ID = 1 (ou toute autre pièce d’identité), après qu’un autre joueur (ou le même joueur) rejoint le jeu, donc il va obtenir le ID du 3 au lieu de remplacer avec l’ID du joueur déconnecté. Et la structure de jeu entier est basé sur ces ID. Donc, pour résoudre ce problème que j’ai stocké ces IDs dans un « Hashtable » (ou « Dictionary » en C #), la clé est la propriété « NetworkPlayer » comme une chaîne (fixe), et la valeur est l’ID.

Les cas que j’ai trouvés sont les suivants:

- ✓ Retirer le joueur déconnecté de la liste des adversaires (avant le début du jeu) et détruire de tout ce qui est lié avec lui (et aussi de mettre à jour le tableau des ID en envoyant un appel RPC à tous les joueurs connectés pour résoudre ce problème).
- ✓ Indiquer tous les autres joueurs que le joueur spécifique a été déconnectés. Si ce joueur est le serveur, tous les joueurs seront exclues du jeu, et le recharger.
- ✓ Mettre à jour la liste des joueurs connectés (donc passer le tour de joueur déconnecté). Si nous sommes dans le mode de partage du monde (questions de « vrai/faux »), j’enlève lui de la liste, sinon (si on est en mode de guerre), je ne

dois pas le retirer de la liste parce que je dois lui dans le classement des joueurs à la fin.

- ✓ Indiquer à la fois de joueur déconnecté et les autres de la raison de la déconnexion.
- ✓ Retirez tous les appels RPC du joueur déconnecté et ses objets instanciés dans le réseau.
- ✓ Si le joueur déconnecté est le dernier dans la liste des joueurs connectés (signifie que le serveur est seul), le serveur doit recharger le jeu (exclue).
- ✓ Si nous êtres encore dans le mode « Vrai/Faux » (signifie que les joueurs partageant toujours le monde, les zones de joueurs déconnectés seront libres, sinon, ses zones seront verrouillés (des zones grises, et aucun joueur ne peut les utilises).
- ✓ L'ordre d'attaque sera mis à jour.
- ✓ Si la déconnexion du joueur est au moment où il essayait de mettre un chevalier, la sélection de chevalier sera annulée (sinon, les joueurs vont attendre un temps infini).
- ✓ Si le joueur est déconnecté au moment de répondre à une question, il doit envoyer une réponse fausse à pour l'échapper de l'attente.

J'ai essayé d'expliquer les structures de jeux aussi simples que je peux, mais en fait, le processus réel est **plus complexe** que ce que j'ai dit, et en particulier les restrictions qui « Unity » est l'appliquons lorsque l'on travaille avec lui. Donc, je peux dire qu'il y a des dixièmes de problèmes que j'ai face, parfois ils apprendraient beaucoup de temps pour les résoudre, et parfois non.

Il suffit que le temps de développement de ce jeu est d'environ 8-9 jours de travail intensif et de se concentrer pour chaque petit détail, et le script principal de communication réseau compte environ **2250 lignes optimisés** avec des méthodes et des techniques très avancés, sans nous oublions le travail de conception, audio, ...

**معركة جماعية**

الرمز:

اسم اللاعب: **Abdullah**

معركة عبر الإنترنت

معركة محلية

تعليمات

رجوع

**معركة فردية**

الرمز:

اسم اللاعب: **Abdullah**

طبيعة المعركة: **4 جولات**

درجة الصعوبة: **حتى الموت**

**سهل** **متوسط** **صعب جدا**

ابدأ المعركة

رجوع

Abdullah **1000** الصعوبة Yacine **1000**

صح أم خطأ؟

شراب أهل الجنة هو الماء واللبن فقط.

**خطأ** **صح**



Abdullah **3250** الصعوبة Yacine **2500**

من هو الشاعر الذي ينتمي إلى أسرة من الشعراء؟

الكميت بن زيد الأسدي

أبو تمام حبيب بن أوس

زهير بن أبي سلمى

الوليد بن عبيد البخري

**المعارك المتاحة**

اسم المعركة	طبيعة المعركة	ك.م؟	عدد اللاعبين	الحالة
GameDev07	4 جولات	لا	2/3	في الإنتظار

معركة جديدة



Abdullah **2500** الحمد لله على كل حال

أول من سمي القرآن بالمصحف هو

عمر بن الخطاب

أبو بكر الصديق

علي بن أبي طالب

الزبير بن العوام

Figure 12 – Empire de Connaissance

# Conclusion Générale

Au cours de ce mémoire, on a présenté la conception ainsi que les dessous de développement des jeux vidéo et son histoire.

La réalisation des jeux vidéo se fait par des moteurs des jeux qui ont été présentés, et on a spécifié notre développement pour le moteur des jeux « Unity », avec présentation des outils, EDI, avantages et inconvénients et le développement mobile en général.

Pour ajouter une bonne production de notre société, on a réalisé une grande application « Portail des jeux éducatifs » qui est ciblé pour les différents niveaux de cultures et âges (Enfants, Culture Général, et Réflexion).

Pendant la procédure de la réalisation de cette application, on a trouvé pas mal de problèmes, et d'être forcé de créer notre solutions unique pour différents problèmes (soit depuis l'environnement de développement (Unity) ou les structures de communications réseaux, ...).

On a aussi utilisé des différents techniques et méthodes avancées pour le développement, comme :

- Graphisme 2D et Modélisation 3D.
- Communication de Réseau locale et internet.
- Base de données locale et autre depuis serveur.
- Nouvelle technologie (RA).

Les perspectives d'améliorations de notre application proposera d'ajouter une autre catégorie pour l'éducation (que des applications (ne sont pas des jeux) comme : dictionnaire, maths, ...) et autre pour notre culture (Islam). Il peut contenir aussi de continuer le développement par la technologie RA, et surtout de compléter l'application « Chimie avec RA » pour réaliser tous les atomes chimiques connus pour l'utilisation réel dans l'éducation.

## Liste Des Figures

Figure 1. Interface de « Unity » .....	
Figure 2. Système d'animation avancé .....	
Figure 3. Dessin .....	
Figure 4. Coloriage .....	
Figure 5. Lettres .....	
Figure 6. Animaux avec la Réalité Augmenté .....	
Figure 7. Chemin Eulérien .....	
Figure 8. Cerveau Vif .....	
Figure 9. Mots .....	
Figure 10. Chimie avec la Réalité Augmenté .....	
Figure 11. Structure de Communication avec RPC .....	
Figure 12. Empire de Connaissance .....	

## Liste Des Tableaux

Tableau I. Entreprises des jeux vidéo .....	
Tableau II. Plates-formes des jeux vidéo .....	

## Références Bibliographiques

[1] Wikipédia – Video Game Development

[http://en.wikipedia.org/wiki/Video\\_game\\_development#History](http://en.wikipedia.org/wiki/Video_game_development#History)

[2] Wikipédia – Video Games Platforms

[http://en.wikipedia.org/wiki/Category:Video\\_game\\_lists\\_by\\_platform](http://en.wikipedia.org/wiki/Category:Video_game_lists_by_platform)

[3] Wikipédia – First Video Game

[http://en.wikipedia.org/wiki/First\\_video\\_game](http://en.wikipedia.org/wiki/First_video_game)

[4] Video Games Impact the Economy More Than You Think

<http://www.cnn.com/id/38644484>

[5] Newzoo – Top 100 Countries By Game Revenues

<http://www.newzoo.com/free/rankings/top-100-countries-by-game-revenues/>

[6] Wikipédia – Android (Operating System)

[http://en.wikipedia.org/wiki/Android\\_%28operating\\_system%29](http://en.wikipedia.org/wiki/Android_%28operating_system%29)

[7] Moteur des jeux « Unity »

<http://www.Unity3D.com>

[8] Réalité Augmentée

<http://www.augmented-reality.fr>

[9] Unity – RPC Details

<http://docs.unity3d.com/Manual/net-RPCDetails.html>

[10] Apple – Qu'est-ce qu'iOS

<https://www.apple.com/fr/ios/what-is/>

[11] Wikipédia – Windows Phone

[http://en.wikipedia.org/wiki/Windows\\_Phone](http://en.wikipedia.org/wiki/Windows_Phone)