

Prep'ISIMA 2 :

Rapport du projet – Application Android

Sommaire

I. Présentation du projet.....	3
II. Cahier des charges.....	3
A. Liste des fonctionnalités à implémenter	3
B. Time line de la difficulté	4
III. Etapes de développement et explications techniques ..	6
<i>Projet de base – 15 février 2018.....</i>	<i>6</i>
<i>En quête d'amélioration des collisions – 27 février 2018</i>	<i>7</i>
<i>Les collisions : encore et toujours – 27 février 2018.....</i>	<i>7</i>
<i>Les collisions : dernière étape – 1^{er} mars 2018</i>	<i>8</i>
<i>L'apparition du menu – 9 mars 2018.....</i>	<i>8</i>
<i>Et la physique ? – 9 mars 2018</i>	<i>8</i>
<i>Toujours plus de réalisme physique – 14 mars 2018.....</i>	<i>9</i>
<i>Une base de données pour sauvegarder les scores – 16 mars 2018.....</i>	<i>9</i>
<i>Amélioration graphique : les bordures – 7 avril 2018</i>	<i>10</i>
<i>Un peu plus de difficulté – 13 avril 2018.....</i>	<i>10</i>
<i>Que serait un jeu sans musique ? – 14 avril 2018</i>	<i>10</i>
<i>Consulter ses scores c'est bien, être classé c'est mieux ! – 25 avril 2018..</i>	<i>11</i>
IV. Problèmes et difficultés rencontrées.....	12
V. Idées d'amélioration	13
VI. Apports personnels du projet	13

I. Présentation du projet

Notre projet consiste en le développement d'un jeu destiné aux smartphones et plus précisément à ceux qui utilisent le système d'exploitation Android de Google.

Notre jeu peut être décrit succinctement car il repose sur des mécaniques de *GamePlay* intuitives. Effectivement, le joueur est matérialisé dans le jeu par un personnage ou une balle qui le représente dans l'univers virtuel et dont le but est de survivre le plus longtemps dans un espace confiné par des bordures. Le contrôle de l'avatar est permis par l'utilisation de l'accéléromètre du téléphone. Evidemment, des ennemis surgissent des quatre côtés de l'écran pour compliquer la tâche et accentuer la difficulté. Par ailleurs, cette dernière s'accroît de manière constante en fonction de la progression du joueur notamment par l'augmentation du nombre d'ennemis ou de leur vitesse.

L'accès au jeu est permis par un menu qui offre également la possibilité de consulter l'historique des meilleurs scores du joueur, gérés grâce à une base de données.

Le développement du jeu aura été effectué grâce à l'IDE *Android Studio* en API Android KitKat 4.4 car cette dernière proposait l'ensemble des fonctionnalités nécessaires au développement tout en permettant de cibler un maximum de terminaux Android. Effectivement, le nombre de téléphones du marché utilisant cette version ou une version ultérieure de ce système d'exploitation se chiffrait approximativement à 90% en août 2017.

II. Cahier des charges

A. Liste des fonctionnalités à implémenter

La liste suivante ordonne les diverses fonctionnalités et caractéristiques que devra implémenter et respecter notre projet. A noter que certaines ne seront malheureusement qu'annexes que l'on pourra potentiellement ajouter au projet. La liste les trie des plus faciles et triviales aux plus complexes à implémenter. On veillera d'ailleurs à respecter l'ordre de développement défini dans la mesure du possible.

- Réussir à bouger la balle grâce à l'accéléromètre présent dans le smartphone.
- Ajouter des ennemis comme des fusées.
- Programmer un système de collisions le plus réaliste et optimal possible.
- Gérer les scores du joueur en local via la présence d'une base de données.
- Ajouter un menu principal basique pour permettre à l'utilisateur la consultation des scores.
- Améliorer la physique du jeu en rendant les déplacements de la balle plus réalistes (notamment en ajoutant une inertie).
- Ajouter des bordures sur les côtés de l'écran pour améliorer l'esthétique et signaler le danger au joueur.

- Augmentation de la difficulté suivant le temps (accroissement du nombre de fusées qui apparaissent et de leur vitesse, ajout d'autres ennemis comme des lasers). Se référer à la « time line de la difficulté » pour plus de détails relativement à l'augmentation de la difficulté suivant la progression du joueur.
- Inclure une musique et des bruitages notamment lors des collisions pour améliorer l'ambiance sonore.
- Améliorer le design du jeu (les sprites, le background, l'aspect graphique en général etc.)
- Rajouter un système de vies.
- Remplacer le « Timer » par un score augmentant au fil du temps.
- Connexion à une base de données distante pour le classement des meilleurs scores.
- Rajouter des bonus comme des points supplémentaires, un bouclier temporaire, un ralentisseur de déplacement des fusées, des vies additionnelles qui traversent l'interface graphique et que le joueur peut attraper, ...
- Rajouter un malus qui inverse les commandes pendant une durée déterminée.
- Rajouter des crédits que l'on obtient à chaque fin de partie, puis, ajouter une boutique qui permet d'acheter des bonus temporaires, des skins pour la bille, un nouveau thème notamment au niveau des bordures (tunnels qui permettent le passage d'un bord à l'autre ou murs qui empêchent toute traversée), etc.

B. Time line de la difficulté

Le tableau en page suivante donne une idée de la time line de la difficulté c'est-à-dire l'augmentation de cette dernière en fonction de l'avancement du joueur dans la partie. A noter qu'il peut évidemment être sujet à des modifications pour notamment des raisons d'équilibrage.

On définira plusieurs types d'ennemis :

- Ennemis de classe A (fusées par exemple) se déplaçant de manière horizontale ou verticale d'un bord vers son opposé à vitesse constante.
- Ennemis légèrement plus dangereux de classe B (couleur distincte pour marquer cette différence) se déplaçant à vitesse croissante d'un bord vers son opposé.
- Ennemis de classe C se déplaçant en courbe à vitesse constante d'un bord vers un autre.

Les types suivants sont des idées d'améliorations optionnelles :

- Ennemis à « tête chercheuse » de classe D partant d'un bord et se dirigeant vers le joueur. A noter que ces derniers disposent évidemment d'une inertie et peuvent-être esquivés par le joueur.
- Ennemis non ponctuels de classe E (comme des lasers) qui réduisent la zone de jeu.

SCORE DU JOUEUR	LOCALISATION DES APPARITIONS	FREQUENCE D'APPARITION (en Hz)		
		fA	fB	fC
0	Les ennemis A n'apparaissent que d'un bord fixé arbitrairement ou déterminé de manière aléatoire.	0.5	-	-
10	Les ennemis A apparaissent en plus depuis le bord opposé.	1	-	-
20	Les ennemis A apparaissent depuis tous les bords.	1	-	-
30	-	1.5	-	-
40	-	2	-	
50	Des ennemis B apparaissent depuis tous les bords.	2	0.5	
60	-	2	1	
70	-	2	1.5	
80	-	3	2	
90	Des ennemis C apparaissent depuis un bord fixé arbitrairement ou déterminé de manière aléatoire.	3	2	1
100	Des ennemis C apparaissent depuis tous les bords.	4	3	2
120 + (20)	Les fréquences augmentent toutes de manière proportionnelles.	+ 1	+ 0.75	+ 0.5
-	-	-	-	-

III. Etapes de développement et explications techniques

Cette partie du compte-rendu retrace l'ensemble des étapes de développement, c'est-à-dire comment le projet a effectivement été développé tout le long de sa production. Evidemment, cette ramification résume l'avancement dans l'ordre chronologique de sa réalisation.

Projet de base – 15 février 2018

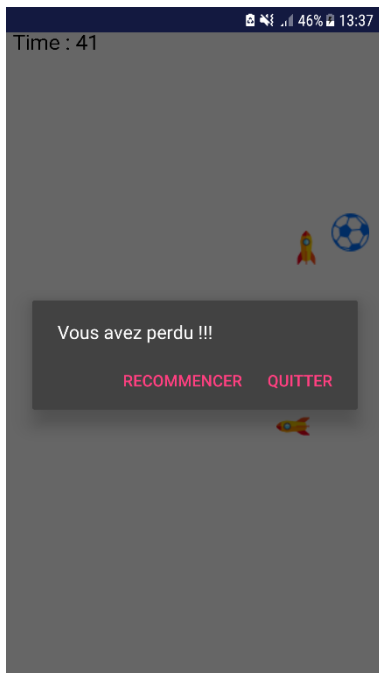


Figure 1 : Illustration du problème de collision. La fusée à gauche de la balle ne devrait pas entraîner un Game Over

Les deux premières exigences du cahier des charges sont remplies. Une balle est présente au centre de l'écran et peut se mouvoir grâce à l'accéléromètre du téléphone. En outre, des fusées apparaissent sur les côtés comme prévu. Le jeu est déjà jouable mais demeure néanmoins à l'état de prototype car la difficulté est effectivement constante.

Nous avons, par ailleurs, tenté de coder un système de collisions mais celui-ci présente des bugs et demeure dans les faits totalement inefficace. Cela est dû à un mauvais calcul au niveau des coordonnées

Par ailleurs le jeu offre la possibilité, non souhaitée, d'être joué en affichage de type « paysage ». Ce léger problème se doit d'être corrigé rapidement.

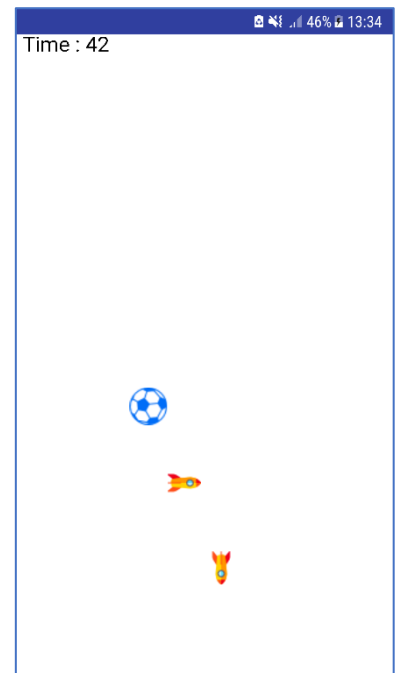


Figure 2 : Première image du jeu en plein fonctionnement

Enfin, une fenêtre telle celle présente en *figure 1* apparaît de manière aléatoire en début de partie et indique au joueur qu'il vient de perdre la partie avant même que cette dernière ne démarre. Ce bug du « recommencer » demeurera d'ailleurs récurrent car il reviendra à plusieurs reprises même après avoir été corrigé jusqu'à, heureusement, disparaître définitivement.

En quête d'amélioration des collisions – 27 février 2018

Les collisions ont été nettement améliorées même si certains problèmes subsistent notamment à cause de la rotation des fusées qui advient lorsque l'on souhaite que ces dernières soient orientées de manière horizontale. La position « de base » étant « orientée vers le haut » et, le sprite étant rectangulaire, les coordonnées ne sont pas modifiées en conséquence lors d'une rotation de 90° ou 270° ce qui provoque évidemment un problème au niveau des collisions.

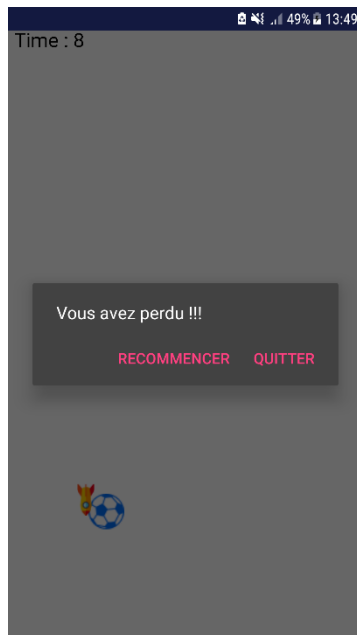


Figure 3 : Collision normale. La rotation à 180° n'affecte pas les coordonnées

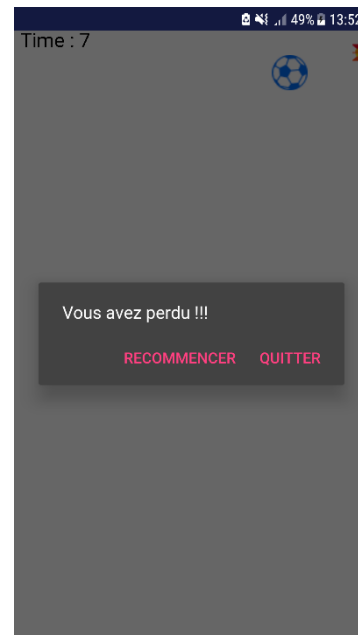


Figure 4 : Collision problématique à cause d'une rotation à 90° vers la droite

Enfin, malgré sa correction antérieure aux modifications des collisions, le « bug du recommencer » est revenu par la suite.

Les collisions : encore et toujours – 27 février 2018

Une nouvelle fois, les collisions ont été améliorées mais celles-ci présentent encore un dernier problème. En effet, les coins des fusées ne génèrent pas de collision ce qui ne provoque pas de Game Over comme escompté. « Le bug du recommencer » est par ailleurs toujours présent.

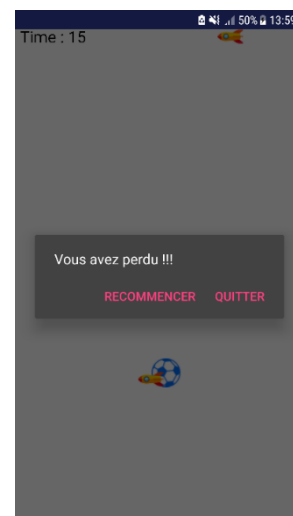


Figure 5 : Collision anormale dans le coin inférieur droit de la fusée

Les collisions : dernière étape – 1^{er} mars 2018

Tous les problèmes de collision ont été résolus grâce à l'utilisation de *bounding boxes* qui permettent de grandement simplifier le calcul et d'optimiser la tâche pour le processeur.

L'apparition du menu – 9 mars 2018

Deux nouvelles activités Android implémentées offrent l'affichage d'un menu principal basique et d'un tableau des scores. Ces deux interfaces graphiques utilisateur sont d'ailleurs disposées grâce à deux fichiers XML. Concernant les scores, le joueur est invité en fin de partie à sauvegarder son exploit ; libre ensuite à lui d'en faire le choix. En ce qui concerne le menu principal, ce dernier est constitué linéairement de haut en bas, d'un bouton permettant d'accéder à l'activité des scores, d'un champ de saisie pour le pseudonyme de l'utilisateur et enfin d'un bouton jouer pour lancer une partie. Malheureusement, la sauvegarde des scores n'est pas fonctionnelle à cause de l'absence de l'implémentation d'une base de données locale à l'application.

Le « bug du recommencer » a été corrigé et il est désormais impossible d'afficher l'application en format paysage.

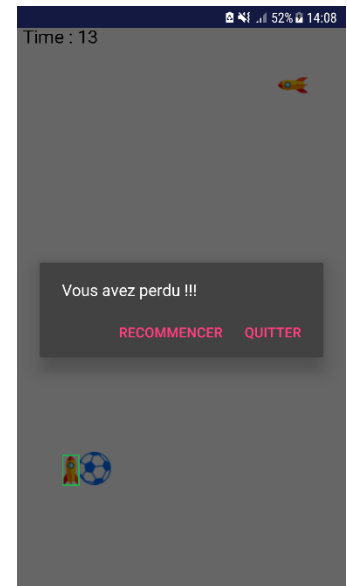


Figure 6 : La bounding box permet des résultats très satisfaisants concernant les collisions. Elle est ici mise en valeur en vert sur la fusée ci-dessus.

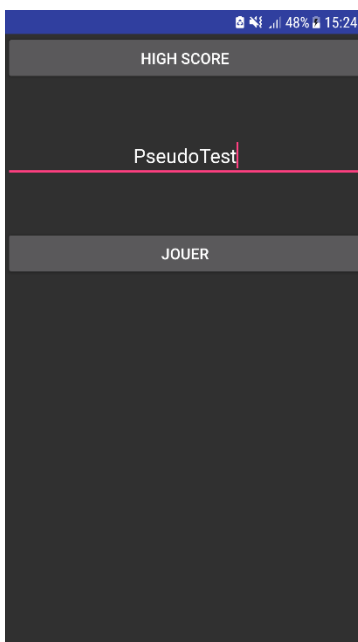


Figure 7 : Le menu principal de l'application

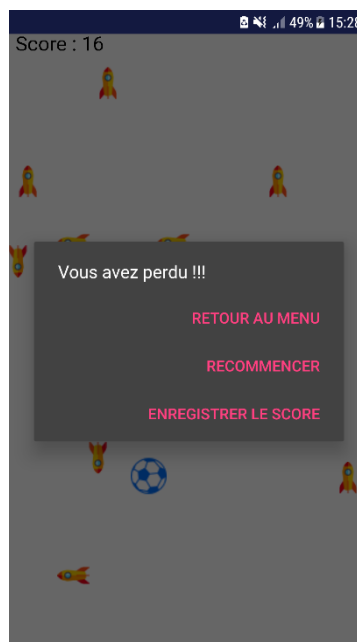


Figure 8 : Nouvelle fenêtre de fin de partie

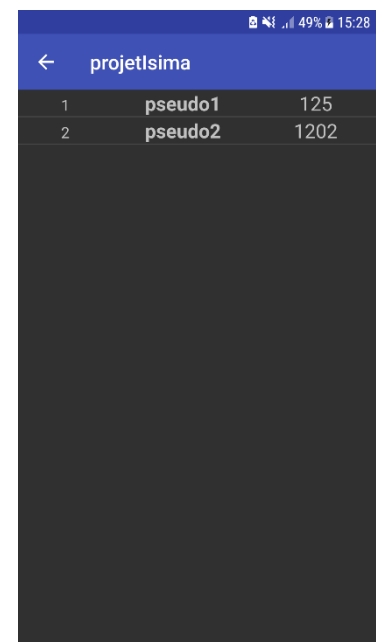


Figure 9 : Tableau des scores

Et la physique ? – 9 mars 2018

L'amélioration de la physique des déplacements est une fonctionnalité essentielle car elle offre d'une part une dimension réaliste non négligeable au jeu et permet, d'autre part une

identité certaine. D'ailleurs, cette composante mérite d'être étudiée minutieusement avant le déploiement d'un jeu car elle ne doit pas être amenée à changer régulièrement. En effet, des modifications trop fréquentes perturbent le joueur dans ses habitudes en plus de venir casser l'identité du jeu.

Après plusieurs recherches, nous avons convenu qu'il était catégoriquement plus adapté de créer nous même un système physique se rapprochant au maximum de la réalité tout en étant basé sur des observations empiriques plutôt que d'implémenter un réel moteur physique exact au sens mathématique. La complexité d'une telle solution aurait été effectivement beaucoup trop chronophage et complexe pour notre projet.

Ainsi, les déplacements de la balle sont actuellement plus réalistes d'un point de vue physique grâce à l'ajout d'une composante servant à représenter l'inertie.

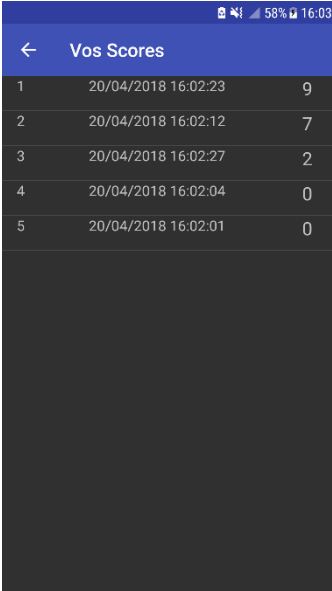
Toujours plus de réalisme physique – 14 mars 2018

L'ajout d'une composante visant à modéliser la dispersion de l'inertie, c'est-à-dire la manière que cette dernière a de s'atténuer avec le temps par la faute des frottements, nous a permis d'atteindre des résultats très satisfaisants. Les coefficients ainsi créés peuvent par la suite être modifiés, en toute légèreté néanmoins, pour parvenir à mimer la réalité de la meilleure manière possible.

Le bug du « recommencer » malgré ses corrections antérieures, s'est présenté à nouveau.

Une base de données pour sauvegarder les scores – 16 mars 2018

L'ajout de l'interface graphique, le 9 mars 2018, pour le classement des scores sous forme d'activité Android, n'aurait pas grand intérêt sans l'utilisation d'une base de données pour sauvegarder les dits scores. C'est pourquoi nous avons implémenté, via la technologie SQLite de l'application, la gestion d'une base de données locale c'est-à-dire non reliée au réseau Internet afin de garder en mémoire les dix meilleurs scores du joueur.



1	20/04/2018 16:02:23	9
2	20/04/2018 16:02:12	7
3	20/04/2018 16:02:27	2
4	20/04/2018 16:02:04	0
5	20/04/2018 16:02:01	0

Figure 10 : Aperçu du tableau des scores locaux

Amélioration graphique : les bordures – 7 avril 2018

Pour le moment, notre application empêche le joueur de traverser les bords de la zone de jeu. Cette particularité, bien que pouvant être décelée ne serait-ce que par une seule partie, se devait évidemment d'être représentée graphiquement. C'est pourquoi nous avons intégré un autre type d'ennemis, matérialisé à l'écran sous forme de lasers verticaux et horizontaux. La collision aura inévitablement été adaptée en conséquence.



Figure 11 : Les bordures améliorent l'aspect graphique

Un peu plus de difficulté – 13 avril 2018

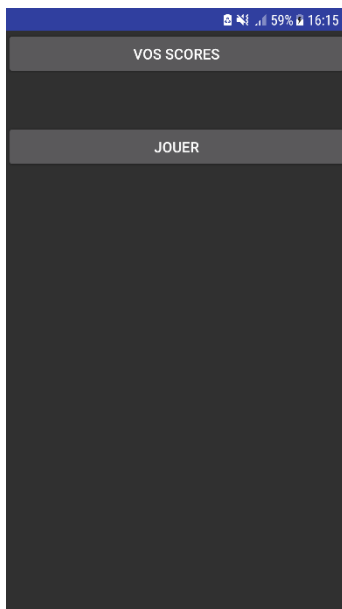


Figure 12 : Aperçu du nouveau menu principal

La « timeline de la difficulté » en page 5 permettant, comme indiqué, de baliser l'accroissement de la difficulté relativement à la progression du joueur, a été retranscrite dans le jeu à la lettre. Les nouveaux ennemis sont venus se greffer à ceux déjà implémentés. On retrouvera ainsi deux nouveaux types de fusées : les fusées de type B, de couleur bleue, se déplaçant à vitesse croissante d'un bord vers son opposé et les fusées de type C, de couleur rose, se déplaçant en courbe à vitesse constante d'un bord vers un autre.

Par ailleurs, un fond a été rajouté en jeu pour améliorer le design graphique.

Le menu principal a d'ailleurs subi quelques modifications puisqu'il n'est plus possible de renseigner un pseudonyme. Son utilité étant en effet pour le moins limitée car les scores ne sont pour le moment que gérés en local.



Figure 13 : Le joueur, submergé par les différentes fusées, tente de survivre tant bien que mal

Que serait un jeu sans musique ? – 14 avril 2018

L'intégration d'une musique ainsi que de bruitages, pour les collisions notamment, était une tâche demeurant inévitable pour obtenir un jeu plaisant à jouer qui attire au maximum le joueur. C'est pourquoi nous avons mis en place une ambiance sonore satisfaisante qui pourra tout à fait être améliorée par la suite.

Consulter ses scores c'est bien, être classé c'est mieux ! – 25 avril 2018

Le cahier des charges mentionne l'utilisation d'une base de données distante pour la gestion des scores c'est pourquoi nous avons complété notre application par l'agrégation d'un système d'inscription et de connexion et par des modifications du menu principal pour accueillir ces nouvelles fonctionnalités.

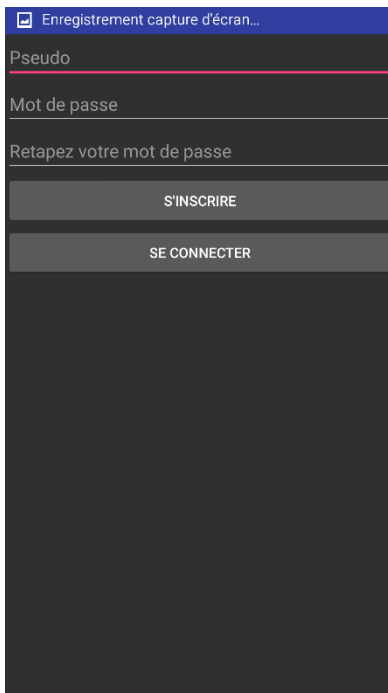


Figure 15 : La page d'inscription permet à l'utilisateur de se créer un compte

Le processus commence en premier lieu par l'inscription d'un joueur lorsque que ce dernier appuie sur « s'inscrire » dans le menu principal. L'utilisateur est alors invité à saisir un nom d'utilisateur et un mot de passe avant qu'une requête soit envoyée au serveur qui se chargera d'insérer le nouveau joueur dans la base de données.

Par la suite, à chaque lancement du jeu, si le joueur a pris soin d'activer sa connexion internet, celui-ci n'aura pas besoin de rentrer à nouveau son mot de passe.

Après chaque partie, l'application vérifie si le score que vient d'effectuer le joueur est un nouveau meilleur score en utilisant la base de données locale. Si oui, elle examine la présence d'une connexion internet et envoie une requête au serveur pour mettre à jour le classement de la base de données. Cette dernière comporte d'ailleurs deux tables : l'une regroupe les utilisateurs tandis que l'autre, les meilleurs scores.

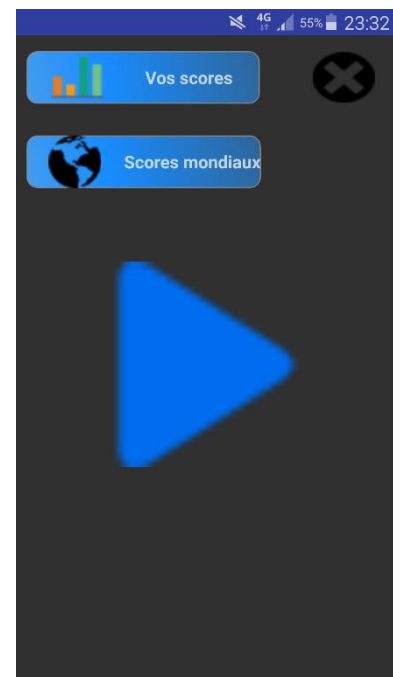


Figure 14 : Aperçu du nouveau menu principal

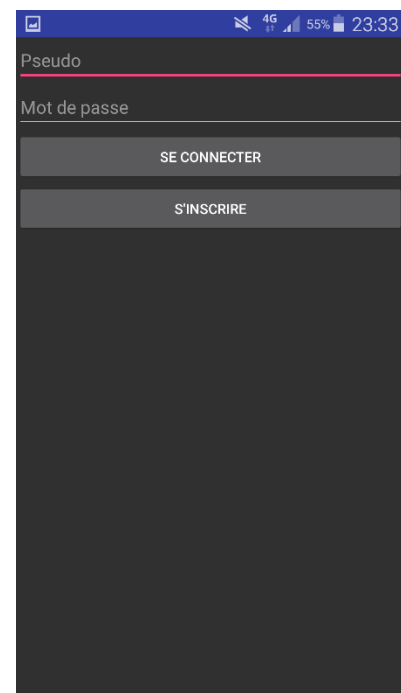


Figure 16 : Aperçu de la page de connexion

Enregistrement capture d'écran...

← Score mondiaux			
R.	Pseudo	Date	Score
1	sybessonne	2018-04-25 23:32:30	75
2	darth vader is luke father	2018-04-25 23:26:12	3

Figure 17 : Classement des scores mondiaux

Dans le menu, l'utilisateur a la possibilité de pouvoir accéder aux dix meilleurs scores mondiaux. Pour les obtenir, notre programme envoie une requête au serveur qui, par la suite, renvoie les dix meilleurs scores en format JSON de manière à ce que l'application puisse les traiter facilement.

Concernant les outils utilisés, nous faisons appel à l'hébergeur « alwaysdata » qui possède l'avantage de fournir un service gratuit pour les fonctionnalités qui nous intéressent ainsi qu'un support extrêmement sérieux. La base de donnée est, quant à elle fournie grâce au système « MySQL » à laquelle on accède via le « phpMyAdmin » du site d' « alwaysdata ».

Notre script qui récupère les requêtes de l'application est un script PHP. Celui-ci reçoit les données à partir de l'URL, en POST en utilisant le protocole https afin d'atteindre une sécurité optimale.

IV. Problèmes et difficultés rencontrées

Au cours du développement de l'application, la rencontre de problèmes et entraves a été inévitable. De manière attendue, plusieurs bugs sont apparus au cours de l'implémentation des différentes fonctionnalités prévues dans le cahier des charges qui souffraient parfois d'un problème de compatibilité avec celles développées précédemment.

Par exemple, suivant l'ajout d'un système physique pour le déplacement de la balle, afin de renforcer le réalisme du jeu, un bug apparaissait au lancement de la première partie et entraînait un crash. Il nous aura fallu plusieurs jours avant de trouver l'origine du problème qui consistait effectivement en la mauvaise initialisation d'une variable.

De même, le fait de travailler sur un projet en binôme, impliquait évidemment parfois des divergences sur nos manières d'implémenter nos fonctionnalités. Il nous aura ainsi fallu nous adapter pour les faire cohabiter dans les meilleures conditions possibles.

Par la suite, la recherche d'un système de collisions convenable n'aura pas été une mince affaire car elle aura nécessité l'apparition de plusieurs fonctions mathématiques ainsi que l'inclusion de bounding boxes pour optimiser la collision entre formes géométriques complexes. On se rappellera notamment de la collision entre la balle du joueur et les différentes fusées.

V. Idées d'amélioration

Comme inscrit dans le cahier des charges, une opportunité aurait été de rajouter des *skins* pour la balle ainsi que des fonds d'écran que le joueur pourrait choisir afin de personnaliser au maximum son expérience et pour évidemment le maintenir en attrait. Ces *skins* pourraient d'ailleurs avoir un impact sur la physique de la balle tandis que la façon dont se comportent les fusées pourrait être dépendante du choix du fond qui correspondrait à un thème prédéfini. Nonobstant, cette dernière possibilité se devrait d'être méticuleusement analysée pour éviter que les classements soit trop inégaux et, en conséquence, biaisés.

En outre, il aurait pu être également intéressant d'accoler au jeu un ensemble d'items comme des bonus ou des malus qui s'afficheraient sur l'écran à certains moments dans la partie et qui permettraient d'avoir un *GamePlay* encore plus intéressant et immersif en plus d'améliorer l'expérience du joueur ainsi que, par la même occasion, la durée de vie du jeu.

Si le temps n'avait pas été un problème, nous aurions développé un système d'achat de *skins* et de fonds d'écran et ajouté l'application sur le Play Store. Par ailleurs, un ajout de publicités aurait également pu être une opportunité pertinente.

D'autre part, comme conseillé par notre tuteur en fin de développement, l'ajout d'un protocole de délégation d'autorisation comme OAuth aurait permis à notre application (dite « consommatrice du service ») d'utiliser l'API sécurisée d'un autre site web (dit « fournisseur ») pour le compte d'un utilisateur. Cette solution aurait donné l'opportunité de déléguer le processus de connexion en offrant notamment la possibilité de s'authentifier grâce aux services de Facebook, Google ou autre.



Logo officiel du protocole OAuth

Enfin, une autre alternative pertinente aurait été d'implémenter une authentification basée sur les jetons (tokens) dont le concept général est de permettre aux utilisateurs d'obtenir, via leur nom d'utilisateur et leur mot de passe, un jeton valide pour une durée fixée arbitrairement et ainsi d'accéder aux ressources d'un site distant (ici le classement mondial). Notre application stockant actuellement le nom d'utilisateur et le mot de passe dans un cache local, cette solution aurait présenté l'avantage de ne pas nécessiter une telle sauvegarde mais seulement la conservation d'un jeton offrant, par définition, un accès à durée limitée pour renforcer la sécurité de notre application.

VI. Apports personnels du projet

Nous concernant, ce projet nous aura offert la possibilité de renforcer nos capacités en langage Java ainsi que de nous familiariser avec les API Android que nous avons eu l'occasion d'utiliser. Mener un projet en rapport avec la technologie embraquée aura été très intéressant

et enrichissant car nous sommes tous deux très intéressés par cette discipline. Par ailleurs, nous aurons pu nous rendre compte de l'utilité et de la puissance d'Android Studio qui reste une référence pour le développement mobile en offrant énormément de raccourcis et options pour faciliter les étapes d'implémentation.

M. Wagner aura été un très bon tuteur car il nous aura aiguillés dans la création de l'application en insistant sur les points importants et tout en nous laissant la liberté d'implémenter toutes les fonctionnalités selon notre envie et suivant un ordre modulable. Avant le début de la programmation au sens propre du terme, Il nous a suggéré la technologie Django côté serveur, pour faire la gestion de la base de données, que l'on aura malheureusement pas pu mettre en œuvre par faute d'un manque de temps. Par ailleurs, il a insisté sur l'importance, pour nos projets futurs, de toujours utiliser des ressources libres de droits car les bordures, bien qu'étant les seules images à ne pas suivre cette démarche, provenaient d'un jeu tiers protégé par des droits d'auteurs.