

ETML

Rapport de projet

Projet Space invaders



Evin Paramanathan
03/11/2023

Rapport de projet

A. Table des matières

UX.....	2
a. Introduction.....	2
b. Analyse	2
1. Conception centrée utilisateur.....	2
2. Choix palette graphique	3
3. Eco-conception.....	3
4. Accessibilité.....	4
c. Conception	5
d. Évaluation.....	9
DB.....	10
a. Importer les données et le schéma de la base de données.....	10
b. Gestions des utilisateurs	10
c. Requêtes de sélection	11
d. Création des index.....	15
1. Pourquoi certain Index existe déjà.....	15
2. Avantage inconvénients des Index.....	15
e. Backup / Restore	15
f. Liaison Base de données TO C#.....	16
POO.....	16
a. Introduction.....	16
b. Analyse fonctionnelle	16
c. Analyse technique	18
d. Test Unitaire	19
e. Chat GPT.....	19
f. Conclusion	19

UX

a. Introduction

Le projet Consiste à créer un jeu très célèbre, le Space Invaders. Dans ce cadre-là on va parler de la partie UX. Dans ce projet Il y a plusieurs choses à faire, il y les maquette graphique (haute et basse fidélité). Cela va servir à avoir une idée du jeu qu'on va développer. Ensuite il y a aussi les personnes qui servent à prendre du recul et se mettre à la place d'un potentiel utilisateur de notre jeu. Cela permet D'être plus objectif quand on développe notre application

b. Analyse


1. Conception centrée utilisateur

Création de personas

La création de persona va permettre de prendre du recul et d'avoir une idée des attentes des potentiels clients. La préférence, les chose qui les frustre ou des choses qui aiment bien.

Dans ce projet il y a deux personnes Martin Alex et Sarah Dupuis.

Martin Alex est un étudiant en littérature de 22 ans qui aime bien jouer aux jeux vidéo que ce soit rétro ou moderne. C'est un homme qui célibataire est jeune donc il y a des préférences et des choses qui l'agacent qui peuvent être différente d'une personne âgée mariée etc. Il est travailleur, ambitieux mais surtout très mauvais perdant.



Martin Alex

Age: 22
Genre: Homme
Location: Paris, France
Formation: Bachelors, Littérature
Famille: Célibataire
Hobby: Jeux vidéo

"Les jeux vidéo sont des fenêtres vers l'imaginaire, où chaque clic ouvre une nouvelle porte vers l'inconnu"

Goals

- Avoir des jeux fluide
- Avoir un jeu qui adapter à son daltonisme

Frustrations

- Quand les jeux ou applications charge lentement
- Avoir des jeux qui sont pas adapter à son handicap

Bio

Alex Martin, né en 2000 à Paris, est un étudiant en littérature à l'Université de la Sorbonne. Sa véritable passion, ce sont les jeux vidéo. Il explore des mondes virtuels pour s'inspirer et se divertir, que ce soit à travers des titres rétro ou modernes. En dehors des études, il joue de la guitare classique, souvent influencé par les mélodies envoûtantes des jeux vidéo. Une vie où la littérature et les jeux vidéo se mêlent pour nourrir sa créativité sans limites. Alex est Daltonien, donc il a souvent mal avec certain jeu.


Motivations

- Ses Amis
- toujours vouloir gagné
- faire le bien

Attitude

- travailleur
- Ambitieux
- Mauvais perdant

Ensuite il y a Sarah Dupuis, elle a 32 ans et elle est développeuse dans une boîte à Lausanne. Elle est mariée à Marc et elle a deux enfant Lucas et Emma.



Sarah Dupuis

Age: 32
Gender: Femme
Location: Lausanne, Suisse
Education: CFC informatique
Family: Mariée, deux enfants, une fille et un garçon
Occupation: Développeuse

Goals

- avoir des menus simple
- Avoir des jeux qui sont aussi adapter au enfant

Frustrations

- avoir des jeux avec des menus trop compliqué
- Des jeux pas adapter au enfant

Bio

Sarah Dupuis, 32 ans, est une développeuse informatique talentueuse résidant à Lausanne, en Suisse. Elle est mariée à Marc, un ingénieur, et ils ont deux enfants, Lucas et Emma. Passionnée par la technologie depuis son enfance à Lyon, elle a gravi les échelons de l'industrie tout en équilibrant sa carrière et sa vie de famille. Sarah partage sa passion pour la technologie avec ses enfants, espérant qu'ils suivront ses traces un jour.

Motivations

- Sa famille
- la réussite
- Sa passion pour l'informatique

Attitude

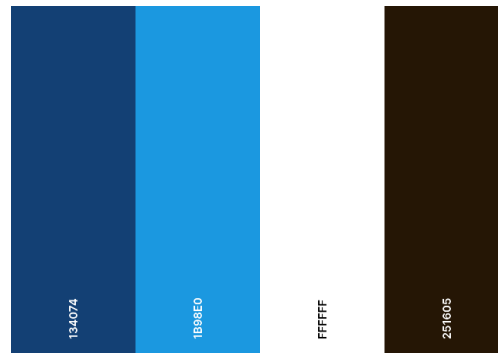
- Patiente
- Déterminé
- Active

Les deux personas sont très différents l'un est jeune célibataire et fait encore des études et à part quand il joue aux jeux vidéo il n'a pas tellement de lien avec le monde numérique alors que la deuxième et un peu plus âgée, est mariée, a des enfants et a déjà fini ses études. Et travaille dans le monde informatique donc ils ont tous les deux une vision différente des jeux vidéo.

2. Choix palette graphique

Les principales couleurs du jeu sont en bleu foncé, bleu clair, blanc et noir.

Le bleu est parfait pour ce genre d'ambiance car ça représente la technologie (quelque chose de futuriste.)



Sur la base de cette palette de couleurs, ça permet de trouver une image qui rentre dans le thème et qui ressemble le plus possible à la maquette de couleur. Sur l'image ci-dessous on peut voir que l'image correspond parfaitement au thème et qu'elle représente bien l'univers du jeu. Et c'est parfait pour le menu de la maquette haute-fidélité.



3. Éco-conception

Pour l'éco-conception, le jeu peut aussi être en noir et blanc. Dès qu'on appuie sur le mode sombre tout le jeu passe en noir et blanc. Cela permet de réduire la consommation d'énergie, car plus il y a de couleurs claires, plus l'écran a besoin d'énergie, donc plus la couleur est sombre, moins l'écran aura besoin de consommer d'énergie.



4. Accessibilité

Sur la maquette haute-fidélité on peut constater que dans paramètre on peut changer les couleurs du jeu pour les personnes qui sont daltonien. Cela va permettre aux personnes qui ne voient pas bien les couleurs d'avoir une meilleure expérience. Il y a plusieurs types de Daltonisme donc dans le jeu il y a trois types de mode (les trois types de daltonisme les plus connues)

Il faut d'abord les personnes qui ont le type Deutan ça veut dire qu'ils ont une diminution de la sensibilité principalement au rouge et vert.



Ensuite il y a les personnes qui ont le type tritan. Ce sont des personnes qui peuvent un peu ou même pas percevoir la couleur bleue.



Et pour fini il y a les personnes qui sont atteint du type Monochromacy. Ce sont de personnes qui aucune couleur don ils ne voient seulement différentes nuances de gris allant du noir au blanc,



c. Conception

1. Définition des écrans

Les maquettes sont en 1920x1080

2. Maquette Haute-fidélité

Quand on regarde en premier il y a le menu Avec le Titre en grand, ensuite il y a plusieurs choix sois jouer, le bouton classement, le bouton paramètre et pour finir le bouton quitter.



Si on appuie sur le bouton classement, on arrive sur une page ou il y a le classement du meilleur joueur solo. En bas de la fenêtre il y a aussi le bouton pour voir le classement



multijoueur, car comme expliqué dans le prochain chapitre(Choix effectués) il y a aussi un mode multijoueur.

Ensuite il y a le bouton paramètres. Si on appuie sur ce bouton on tombe sur une page où l'on peut changer la couleur du jeu pour adapter aux personnes qui sont daltonien comme expliqué dans la partie accessibilité. Il y a aussi les deux modes de couleur le mode normal ou le mode sombre.



Et le meilleur pour la fin le bouton jouer. Dès qu'on clique sur ce bouton on arrive sur une page nous laisse entre deux choix soit le choix solo pour faire une partie solo ou le mode multijoueur où l'on peut jouer à plusieurs.



Une fois le choix du mode de jeu on arrive sur la page où le joueur doit sélectionner son vaisseau entre trois types de vaisseau.



Une fois le choix du vaisseau effectué il faut sélectionner le vaisseau de l'ennemi.



Et dès que le vaisseau ennemi est choisi il suffit de lancer le jeu et de s'amuser :)

3. Maquette Console

La maquette console est beaucoup plus simple le fond est noir il y a le titre au milieu en haut de la maquette ensuite il y a les sous-titres jouer, classement et quitter



4. Choix effectués

Maquette : Pour la maquette j'ai fait exprès de mettre le titre de la page Vaisseau en bleu et la page ennemies en rouge parce que souvent les ennemies sont représentées par le rouge car ça représente le danger. Et voir ça rend le titre un peu plus agressif



Les vaisseaux du joueur ont des couleurs plus claires alors que les vaisseaux ennemis sont foncés, car souvent les méchants sont représentés avec l'obscurité.



Dans le jeu il y a deux modes de jeu le mode solo et le mode multijoueur. Dans le mode solo c'est un space invaders classique le canon est en bas de l'écran et doit avoir toutes les ennemies qui viennent d'en haut. Le mode multijoueur est totalement différent il se joue à plusieurs donc les ennemies ne viennent pas seulement d'en haut mais peut aussi les encercler et le but est d'avoir le plus de point que sont amis. Donc dans ce mode les joueurs doivent s'entraider afin de rester le plus longtemps en vie mais doivent aussi essayer de marquer plus de points que leur ami.



5. ChatGPT

Dans cette partie du projet ChatGPT à principalement été utilisé afin de créer les personnas. Pour créer leur Biographie et à partir de cette Bio développer leur motivation, leur attitude, leurs Objectifs et leurs frustrations. À part pour les personnas chat GPT n'a tellement servit car pour figma ça ne sert pas grand-chose pareil pour le rapport.

d. Évaluation

1. Test

Nom du test	Description	Etape de test	Résultat attendu	Résultat actuelle
Menu principal	Menu dès que le jeu est lancé	1. Cliquer sur le bouton jouer	Aller sur la page Mode (la page ou on choisir entre jeu solo ou multijoueur)	En appuyant ça change de page ça va bien sur la page menu
Couleur sombre	Passage du mode de couleur normal a sombre	1. Depuis le menu principalement cliquer sur le sous menu paramètre 2. Ensuite depuis la page paramètre appuyer sur le bouton ou c'est écrit normal juste à côté du titre mode	Dès que le bouton est cliqué tout le jeu est en noir est blanc.	Le jeu passe bien de colorer a noir et blanc.
Classement solo et multijoueur	Sur la page classement passer du classement solo a multijoueur	1. Depuis le menu principalement cliquer sur le sous menu classement 2. Ensuite cliquer sur le bouton multijoueur	Il faut que ça passe de la page solo a multijoueur	En cliquant ça passe bien du classement solo a multijoueur.
Choix Vaisseau	Sur la page choix du vaisseau il faut pouvoir choisir entre plusieurs styles de vaisseau	1. Depuis le menu principalement cliquer sur le sous menu paramètre 2. Ensuite depuis la page paramètre appuyer sur le bouton ou c'est écrit normal juste à côté du titre mode 3. Après avoir choisi le mode de jeu ça nous redirige sur la page choix du vaisseau Une fois sur cette page appuyer sur la flèche à côté du vaisseau.	Dès que le testeur appuie sur la flèche le model du vaisseau doit changer	En cliquant sur la flèche la model change.

DB

a. Importer les données et le schéma de la base de données

Docker : docker exec -i db mysql -h

```
6524cdd35220a9f403be8eda44b3f03922c816bb09b36bb9a0bc20768e9568ec-P 3306 -uroot  
-proot < db_space_invaders.sql
```

On sélectionne la base de données, ensuite on met l'identifiant root avec son mot de passe root et pour finir on met ce signe (<) pour dire que on importe la base de données qui dans ce cas la db_space_invaders.sql

b. Gestions des utilisateurs

Crée un Utilisateur pour chaque Partie :

Administrateur : CREATE USER 'admin'@'localhost' IDENTIFIED BY 'admin' ;

on utilise CREATE USER pour dire qu'on créer un nouveau utilisateur ensuite on lui donne un nom et attribut un serveur, l'IDENTIFIED BY permet de donner un mot de passe a cet utilisateur.

Joueur : CREATE USER 'joueur1'@'localhost' IDENTIFIED BY 'joueur1' ;

La on crée le joueur1 et on lui donne le mot de passe joueur1

Gestionnaire : CREATE USER 'gestion'@'localhost' IDENTIFIED BY 'gestion'

Et la on crée un utilisateur qui s'appelle gestion avec son mot de passe qui est gestion.

Créer des rôles :

Administrateur : CREATE ROLE 'r_administrateur';

Cette requête crée un rôle grâce au CREATE ROLE qu'on appelle r_administrateur

Joueur : CREATE ROLE 'r_joueur';

Cette requête crée un rôle qu'on appelle r_joueur

Gestionnaire de la boutique : CREATE ROLE 'r_gestionnaire';

Cette requête crée un rôle qu'on appelle r_gestionnaire

Donner les rôles à chaque utilisateur :

Admin : GRANT 'r_administrateur' TO 'admin'@'localhost' ;

Utilise GRANT afin d'accorder des autorisations dans la base de données. Dans ce cas-là sur on donne le rôle de r_administrateur a l'admin grâce au TO

Joueur : GRANT 'r_player' TO 'joueur1'@'localhost' ;

On donne le rôle de r_player à l'utilisateur joueur1

Gestionnaire de la boutique : GRANT 'r_manager' TO 'gestion'@'localhost';

Et là on donne le rôle r_manager à l'utilisateur gestion

Attribution des droits au Rôles

Admin :

Grant SELECT ,CREATE, UPDATE, DROP ON db_space_invaders.* To 'r_administrateur'

WITH GRANT OPTION ;

On Utilise GRANT afin d'accorder des autorisations dans la base de données ensuite on dit tous les autorisations qu'on donne dans ce cas on lui donne l'autorisation pour sélectionner, et de suppression d'objet dans la base de données. On utilise ON pour spécifier la base de données. Ensuite on utilise TO pour dire sur quel rôle on donne l'autorisation. Et pour fini on utilise WITH GRANT OPTION pour que le rôle qui obtient tous ces autorisations puissent lui aussi donné des autorisations a d'autre utilisateur ou d'autre rôles.

Joueur :

Grant SELECT, ON db_space_invaders.t_arme To 'r_player' ;

Ça permet d'accorder la permission de sélectionner au rôles r_player sur la table t_arme de la base données db_space_invaders

Grant SELECT, CREATE ON db_space_invaders.t_commande To 'r_player' ;

Ça permet d'accorder la permission de sélectionner et créer au rôles r_player sur la table t_commande de la base données db_space_invaders.

Gestionnaire :

Grant UPDATE, SELECT, DELETE, INSERT ON db_space_invaders.t_arme TO 'r_gestionnaire' ;

Ça permet d'accorder la permission de mettre à jour, sélectionner, supprimer et insérer au rôles r_gestionnaire sur la table t_arme de la base données db_space_invaders.

Grant SELECT ON db_space_invaders.t_commande TO 'r_gestionnaire' ;

Ça permet d'accorder la permission de sélectionner au rôles r_gestionnaire sur la table t_commande de la base données db_space_invaders.

c. Requêtes de sélection

Requêtes n°1 :

SELECT * FROM `t_joueur` ORDER BY jouNombrePoints DESC LIMIT 5 ;

idJoueur	jouPseudo	jouNombrePoints	▼ 1
5	SpaceRanger	6200	
3	PlayerThree	6000	
10	AstroAce	5750	
9	StarStriker	5600	
2	PlayerTwo	5500	

Avec cette requête on sélection toutes les colonnes de la table t_joueur et on affiche les 5 joueurs qui ont le plus de point dans l'ordres décroissant grâce au ORDER by pour ordonner, Desc pour mettre dans l'ordre décroissant et limit 5 pour juste afficher les 5 premier.

Requêtes n°2

```
SELECT Max(armPrix) AS PrixMaximum, MIN(armPrix) AS prixMinimum, AVG(armPrix) AS PrixMoyen FROM `t_arme`;
```

PrixMaximum	prixMinimum	PrixMoyen
5000	100	1773.6842

On sélection armPrix, on prend son maximum(grâce au Max) et on le renomme PrixMaximum grâce au AS, on prend aussi la moyenne(grâce au AVG) d'armPrix et on renomme par PrixMoyen et pour finir on prend le minimum (grâce au MIN)de armPrix et on le renomme Prixmoyen. Tout ça dans la table t_arme.

Requêtes n°3

```
SELECT fkJoueur AS idJoueur, COUNT(fkJoueur) AS NombreCommandes FROM t_commande GROUP BY fkJoueur ORDER BY COUNT(fkJoueur) DESC;
```

idJoueur	NombreCommandes
1	3
7	3
2	2
4	2
5	2
6	2
8	2
9	2
3	1
11	1

Ça les données de la colonne "fkJoueur" de la table "t_commande" et les renomme en "idJoueur" dans le résultat final. Ensuite Elle calcule le nombre de fois que la valeur "fkJoueur" se produit dans la table "t_commande" et la renomme en "NombreCommandes". Elle regroupe les résultats en fonction des valeurs uniques de "fkJoueur". Cela indique que les lignes associées au même "fkJoueur" sont combinées. Elle classe les résultats en ordre décroissant (DESC) en fonction du nombre de commandes pour chaque joueur (NombreCommandes).

Requêtes n°4

```
SELECT fkJoueur AS idJoueur, COUNT(fkJoueur) AS NombreCommandes FROM t_commande GROUP BY fkJoueur HAVING COUNT(fkjourer)>2 ORDER BY COUNT(fkJoueur) DESC;
```

idJoueur	NombreCommandes
1	3
7	3

Cette requête extrait des données de la table "t_commande". Elle groupe les enregistrements par la colonne "fkJoueur" (les joueurs), compte combien de commandes chaque joueur a passé, et ne renvoie que les groupes (joueurs) qui ont passé plus de 2 commandes. Les résultats sont triés par ordre décroissant.

Requêtes n°5


```
SELECT t_joueur.jouPseudo AS Pseudo, t_arme.armNom AS NomArme FROM t_commande
JOIN t_detail_commande ON t_commande.idCommande = t_detail_commande.fkCommande
JOIN t_arme ON t_detail_commande.fkArme = t_arme.idArme JOIN t_joueur on
t_commande.idCommande = t_joueur.idJoueur;
```

Pseudo	NomArme
PlayerOne	Laser basique
PlayerOne	Missile nucléaire
PlayerTwo	Canon à plasma
PlayerTwo	Bouclier magnétique
PlayerThree	Bombes à fragmentation
PlayerThree	Rayon gelant
GalacticGamer	Torpilles cosmiques
GalacticGamer	Désintégrateur
SpaceRanger	Ondes soniques
SpaceRanger	Électro-pulse
InvaderZapper	Lance-roquettes quantique
InvaderZapper	Vortex noir
LunarLander	Bouclier énergétique
LunarLander	Champ de force magnétique
CosmicCrusader	Drone protecteur
CosmicCrusader	Miroir réfléchissant
StarStriker	Générateur de brouillard
StarStriker	Capsule temporelle
AstroAce	Distorsionneur spatial
NULL	Laser basique
NULL	Missile nucléaire

Ça sélectionne la table t_joueur.jouPseudo on change de nom en Pseudo grâce au AS ensuite on sélectionne la table t_arme.armNom et on change aussi le nom par NomArme. Tout ça depuis la table t_commande. Ensuite on fait 4 jointures, entre ; t_detail_commande, t_commande, t_arme et t_joueur. Et ça nous donne le résultat comme sur l'image au-dessus.

Requêtes n°6

```
SELECT t_joueur.jouPseudo AS Pseudo, t_arme.armNom AS NomArme FROM t_commande
JOIN t_detail_commande ON t_commande.idCommande = t_detail_commande.fkCommande
JOIN t_arme ON t_detail_commande.fkArme = t_arme.idArme JOIN t_joueur on
t_commande.idCommande = t_joueur.idJoueur;
```

Pseudo	NomArme
PlayerOne	Laser basique
PlayerOne	Missile nucléaire
PlayerTwo	Canon à plasma
PlayerTwo	Bouclier magnétique
PlayerThree	Bombes à fragmentation
PlayerThree	Rayon gelant
GalacticGamer	Torpilles cosmiques
GalacticGamer	Désintégrateur
SpaceRanger	Ondes soniques
SpaceRanger	Électro-pulse
InvaderZapper	Lance-roquettes quantique
InvaderZapper	Vortex noir
LunarLander	Bouclier énergétique
LunarLander	Champ de force magnétique
CosmicCrusader	Drone protecteur
CosmicCrusader	Miroir réfléchissant
StarStriker	Générateur de brouillard
StarStriker	Capsule temporelle
AstroAce	Distorsionneur spatial
NULL	Laser basique
NULL	Missile nucléaire

Cette requête SQL récupère le pseudo des joueurs et le nom des armes associées aux commandes en reliant les tables t_commande, t_detail_commande, t_arme, et t_joueur à travers les relations existantes.

Requêtes n°7

```
SELECT t_joueur.jouPseudo AS joueur, t_commande.comNumeroCommande AS t_commande
FROM `t_joueur` LEFT JOIN t_commande ON t_commande.fkJoueur = t_joueur.idJoueur
WHERE t_joueur.jouPseudo IS NOT NULL;
```

joueur	t_commande
PlayerOne	CMD010
PlayerOne	CMD014
PlayerOne	CMD020
PlayerTwo	CMD011
PlayerTwo	CMD012
PlayerThree	CMD013
GalacticGamer	CMD001
GalacticGamer	CMD004
SpaceRanger	CMD005
SpaceRanger	CMD015
InvaderZapper	CMD006
InvaderZapper	CMD016
LunarLander	CMD003
LunarLander	CMD007
LunarLander	CMD017
CosmicCrusader	CMD008
CosmicCrusader	CMD018
StarStriker	CMD009
StarStriker	CMD019
AstroAce	NULL

Sélectionne les pseudonymes des joueurs ("joueur") et les numéros de commande correspondants ("t_commande") dans la table "t_joueur" et la table "t_commande." Elle inclut tous les joueurs ayant un pseudonyme défini, même s'ils n'ont pas passé de commande.

Requêtes n°8

```
SELECT t_joueur.jouPseudo AS JOUEUR, t_commande.comNumeroCommande AS
t_commande FROM `t_joueur` RIGHT JOIN t_commande ON t_joueur.idJoueur =
t_commande.fkJoueur;
```

JOUEUR	t_commande
GalacticGamer	CMD001
NULL	CMD002
LunarLander	CMD003
GalacticGamer	CMD004
SpaceRanger	CMD005
InvaderZapper	CMD006
LunarLander	CMD007
CosmicCrusader	CMD008
StarStriker	CMD009
PlayerOne	CMD010
PlayerTwo	CMD011
PlayerTwo	CMD012
PlayerThree	CMD013
PlayerOne	CMD014
SpaceRanger	CMD015
InvaderZapper	CMD016
LunarLander	CMD017
CosmicCrusader	CMD018
StarStriker	CMD019
PlayerOne	CMD020

Grâce cette requête sélectionne la table t_joueur. jouPseudo et on le renomme par JOUEUR grâce au AS, ensuite on sélectionne la table t_commande.comNumeroCommande et on le renomme par t_commande. Tout ça de la table t_joueur ensuite on fait une jointure droite. RIGHT JOIN est une jointure entre 2 tables qui permet de retourner tous les enregistrements de la table de droite même s'il n'y a pas de correspondance avec la table de gauche.

Requêtes n°9

```
SELECT DISTINCT j.jouPseudo AS Pseudo, SUM(dc.detQuantiteCommande) AS
NombresTotalArmes FROM t_commande AS c JOIN t_detail_commande AS dc ON
c.idCommande = dc.fkJoueur JOIN t_arme AS a ON dc.fkArme = a.idArme RIGHT JOIN
t_joueur AS j ON c.fkJoueur = j.idJoueur GROUP BY jouPseudo HAVING jouPseudo IS NOT NULL;
```

Pseudo	NombresTotalArmes
PlayerOne	7
PlayerTwo	7
PlayerThree	3
GalacticGamer	6
SpaceRanger	6
InvaderZapper	6
LunarLander	12
CosmicCrusader	7
StarStriker	7
AstroAce	NULL

Cette requête sélectionne le pseudo des joueurs et la somme des quantités d'armes associées à leurs commandes. Des tables et sont associer grâce à des jointures et des groupements. Les résultats sont filtrés pour enlever les joueurs sans pseudo.

Requêtes n°10

```
SELECT DISTINCT j.jouPseudo AS Pseudo, COUNT(DISTINCT a.armNom) AS NomArme FROM
t_commande AS c JOIN t_detail_commande AS dc ON c.idCommande = dc.fkCommande JOIN
t_arme AS a ON dc.fkArme = a.idArme JOIN t_joueur AS j ON c.fkJoueur = j.idJoueur GROUP
BY jouPseudo HAVING COUNT(DISTINCT a.armNom) > 3;
```

Pseudo	NomArme
GalacticGamer	4
LunarLander	4
PlayerTwo	4

d. Création des index

```
CREATE INDEX nom_de_l_index ON nom_de_la_table
```

On Crée un index grâce au CREATE INDEX, ensuite on lui donne un nom et grâce aux ON on sélectionne la table qu'on veut indexer

1. Pourquoi certain Index existe déjà
Mysql crée automatiquement certaine Index comme pour les clé primaire, clé étrangère et pour les champs unique.
2. Avantage inconvénients des Index
Avantage : Avoir des index permet d'effectuer des recherches plus rapides.

Inconvénients : Ils ralentissent certaines requêtes comme l'insertion, modification et la suppression.

e. Backup / Restore

Backup : Mysqldump -u root -proot --databases db_space_invaders > BackupInvaders.sql

Mysqldump est utilisée pour sauvegarder une base de données. Dans ce cas-là ça nous permet de sauvegarder la base de données db_space_invaders. Avec l'utilisateur root et le mot de passe root. Et on sauvegarde dans un fichier qui s'appelle BackupInvaders.sql.

Restore : Mysql -u root -proot db_space_invaders < backup.db_space_invaders.sql

Ça nous permet d'importer une sauvegarde fichier SQL dans une base de données MySQL nommée "db_space_invaders" en utilisant l'utilisateur "root" avec le mot de passe "root"

f. Liaison Base de données TO C#

Dans ce projet on doit relier notre base de données à notre code c# pour cela : il faut faire clique droite sur notre projet puis sélectionner Gérer les packages Nuget pour la solution. Ensuite dans la fenêtre il faut chercher l'extension MySql.Data et après ça il faut l'installer. Ensuite pour lier la base de données au Code en utilise le connexionstring. Pour cela il faut mettre les informations suivantes : l'host du serveur, le port du serveur, le nom de la base de données, le nom de ,l'utilisateur et le mot de passe.

```
public MySqlConnection connection;
private string _serverNom = "localhost"; //adresse du serveur
private string _serverId = "root"; //identifiant de connexion
private string _serverMP = "root"; //mot de passe de connexion
private string _serverPort = "6033"; //port du serveur
private string _dbName = "db_space_invaders"; //nom de la base de données
```

POO

a. Introduction

Dans le Projet SpicyInvaders nous avons aussi une partie C# qui est de donné vie à ce projet grâce à l'orienté objet.

b. Analyse fonctionnelle

Déplacement du Canon

En tant que joueur je souhaite pouvoir déplacer mon vaisseau de gauche à droite afin de pouvoir viser les ennemies.

Tests d'acceptance :

Droite	Dans le jeu Quand on appuie une fois sur la flèche de droite le canon se déplace vers la droite
Limite	Si on est à la limite de la fenêtre à droite et qu'on appuie sur la flèche droite le vaisseau ne doit pas bouger.
Gauche	Dans le jeu Quand on appuie une fois sur la flèche de gauche le canon se déplace vers la gauche
Affichage du canon	Dans le menu Je lance le jeu Le canon est en bas au milieu de la fenêtre

Invaders

En tant que joueur je veux des ennemis pour me battre contre eux

Tests d'acceptance :

Afficher	Dans le menu Quand je lance le jeu Les aliens apparaissent en formation (voir maquette)
----------	---

Déplacement des aliens	Dans le jeu les invaders se déplacent de gauche à droite
Déplacement des aliens	Dans le jeu, Les aliens se déplacent comme montré par le trait rouge dans la maquette

Canon tire

En tant que joueur je souhaite avoir un canon qui tire des projectiles afin d'abattre mes ennemis	
Tests d'acceptance :	
Tire	Dans le jeu quand j'appuie sur la barre d'espace un missile est affiché
Déplacement Missile	Dans le jeu avec un missile tiré automatiquement le missile va vers le haut

Impact

En tant que joueur j'ai besoin d'avoir un impact quand mon missile touche l'alien afin de lui enlever de la vie et l'éliminer	
Tests d'acceptance :	
Dégâts	Dans le jeu Quand un missile touche un alien L'alien perd de la vie
Disparition	Dans le jeu quand l'alien a reçu 2 impacts L'alien disparaît

Menu

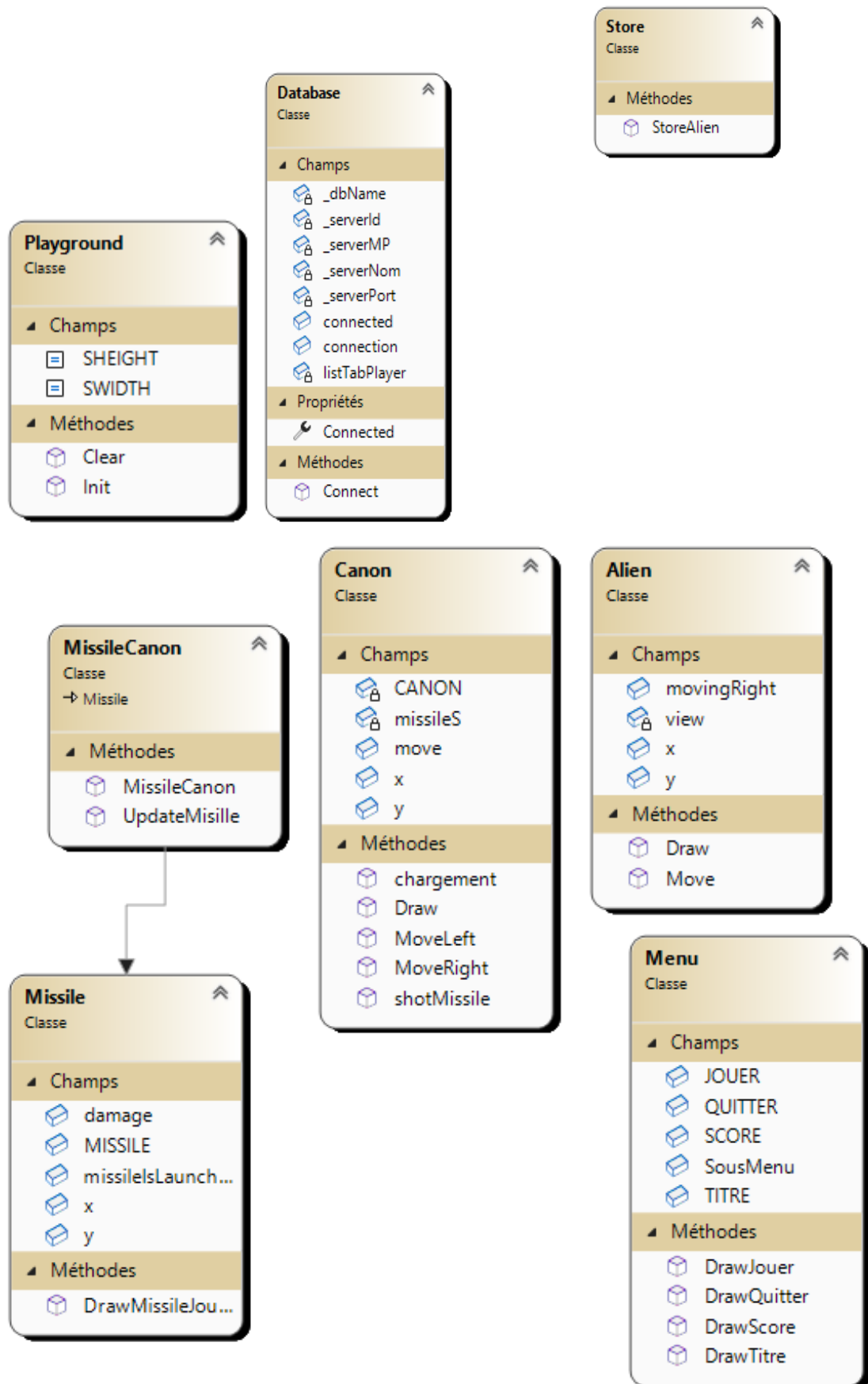
En tant que joueur je souhaite avoir un menu quand je lance le jeu afin de pouvoir choisir si je veux lancer une partie, voir le classement ou sortir du jeu	
Tests d'acceptance :	
Partie	Dans le menu quand le joueur choisi de lancer la partie une nouvelle partie est créée
Quitter	Dans le menu Quand le joueur choisi de quitter la partie la fenêtre se ferme
Classement	Dans le menu Quand le joueur choisi d'aller voir le classement La page Classement s'ouvre

Défaite

En tant que joueur quand un alien arrive en bas à gauche de l'écran le jeu doit s'arrêter et afficher Game over	
Tests d'acceptance :	
Fin	Dans le jeu quand l'alien arrive tout en bas à gauche le jeu doit s'arrêter

c. Analyse technique

1. Diagramme de classe



2. Explications

Analyse fonctionnelle est sur un fichier PDF en annexe

d. Test Unitaire

Pour les tests unitaires il y a un test pour le déplacement du canon à gauche.



```
public void MoveLeftTest()
{
    Canon canon = new Canon();
    canon.x = 3;

    // Act
    canon.MoveLeft();

    // Assert
    Assert.AreEqual(1, canon.x);
}
```

e. Chat GPT

Dans cette partie du projet l'intelligence artificiel Chat GPT à très peu servit, elle a été utiliser afin de vérifier si une partie du code était juste ou pas. Bien sur la réponse donner par chat GPT n'est pas forcément juste c'est pour cela qu'il ne faut pas entièrement se baser dessus mais ça peut servir comme un deuxième avis ou comme une prévention si ça trouve une erreur.

f. Conclusion

Ce projet ma permit de découvrir la programmation Orientée Objets et d'appliquer ce qu'on voyait dans les cours, dans un projet intéressant. Même si je rencontre des difficultés en C#, ça m'a permis de découvrir davantage et d'apprendre de nouvelle chose dans ce langage de programmation. Le jeu n'a peut-être pas parfait mais je suis tout de même satisfait de ce que j'ai produit.

Sinon une chose que j'ai appris à utiliser c'est GitHub même si c'était compliqué au début car on se dit que c'est une perte de temps, je comprends pourquoi le prof voulait qu'on travaille ça permet d'avoir un backup et c'est très pratique et quand on devra travailler en groupe ça va être très pratique.