**DOCUMENT DE CONCEPTION**

**GÉNÉRALE**

|  |  |
| --- | --- |
| **Contenu apporté** | **Version** |
|  | 1 |

**Table des Matières**

1. **Liste des exigences-------------------**

* **Menu Pong (MP)**
* **Règlement Pong (RP)**
* **Jeu Pong (JP)**
* **Mutateur Pong (MUP)**

1. **Modules---------------------------------**
2. **Fonctions/Structures----------------**

**Partie 1 : Liste des exigences :**

# MP001 : Interaction dynamique

Le jeu devra actualiser son affichage lorsqu’une action est effectuée.

# MP002 : Menu fonctionnel

Le menu principal propose les options suivantes « Jouer », « Règles », « Quitter »

# MP003 : Option quitter

L’option quitté est présente sur toutes les pages afin de permettre à l’utilisateur de quitter à tout moment.

# MP004 : Lancement de partie

Il existe un bouton « *jouer »* qui permet de lancer une préparation de partie, soit la possibilité de choisir entre le multijoueur contre une IA, contre un autre joueur sur le même clavier ou bien de jouer tout seul en entrainement. Une fois ces paramètres sélectionnés, le joueur pourra décider de son mode de jeu, à savoir contre la montre ou battle royal.

# MP005 : Pseudo

Le joueur en début de partie pourra crée/rentrer son pseudo.

# RP001 : Règlement

Le règlement est présent dans le menu fonctionnel afin de permettre au joueur de le consulter avant de lancer une partie.

# RP002 : Bouton retour

Une fois dans le menu règlement, un bouton retour est présent afin de permettre à l’utilisateur de revenir au menu fonctionnel.

# JP001 : Aire de jeux

L’utilisateur aura accès à l’aire de jeu de la taille choisi ci-dessous où apparaitrons les deux joueurs sur le terrain. Cette dernière sera fermée de façon que la balle et les raquettes ne puissent pas en sortir.

# JP002 : Taille aire de jeu

L’aire de jeu sera de 752x474.

# JP003 : Gameplay

Les deux raquettes se déplace à la verticale uniquement, la balle traverse le terrain de part et d’autre elle ne rebondit que sur les raquettes et sur les bordures de la zone, elle ne peut pas sortir de la zone de jeu.

# JBN004 : Vie (Battle Royal)

Le joueur dispose d’un nombre de vie définie qu’il perdra progressivement au cours de la partie s’il loupe la balle.

# JBN006 : Temps (contre-la-montre)

Dans le cas d’une partie au temps le joueur ayant perdu la balle le plus de fois à la fin du temps sera perdant et inversement pour le gagnant.

# JBN007 : Pause

Le joueur pourra à tout moment de la partie mettre pause.

# JBN008 : Balle

Quel que soit le mode choisi la vitesse de la balle augmentera au fil du temps.

# JBN009 : Score

A la fin de la partie les scores seront affichés.

# JBN010 : Rejouer

En fin de partie l’option *« Rejouer »* sera disponible et relancera une partie.

# JBN011 : Sauvegarde des scores

L’utilisateur pourra sauvegarder ses scores s’il le désire.

# MUP001 : Mutateur

Le mutateur permettra en début de partie de modifier la partie (taille de la balle, vitesse de la balle, taille des raquettes).

**Partie 2 : Modules :**

**IHM (affichage)**

**Moteur du jeu**

**Contrôleur**

**Sauvegarde du jeu**

**Options de jeu**

**Fonctionnalités du jeu**

**Lecture**

**Ecriture**

Interactions (clic, caractère)

Données d’options

Interactions (clic, caractère)

Données de la partie/ Résultat

Données d’options/ Interactions (clic, caractère)

Identification de l’utilisateur

Données de l’utilisateur

Données de l’utilisateur/résultats

Données de l’utilisateur/ Données de la partie

**Partie 3 : Fonctions/Structures :**

**affichage**

**Menu**

**partiedejeu**

**Score**

**Options**

**IA**

**Mode de jeu**

**optionjeu**

**taille\_raquette**

**vitesse\_balle**

**taille\_balle**

**partieJeu**

**deplacement\_raquette**

**deplacement\_balle**

**rebond\_balle**

**vitesse\_balle**

**score**