

Projet de développement multimédia : casse-briques

Auteurs : Adonis Stavridis, Mika Filleul.

Pour ouvrir le jeu :

- 1) Téléchargez le fichier "filleul_stavridis.zip".
- 2) Allez dans le dossier "html" et ouvrez index.html dans un navigateur (de préférence mozilla firefox)
 - a) Si vous utilisez firefox, il vous faudra accorder une autorisation afin d'activer les effets sonores.
 - b) Pour cela allez dans "Options>Vie privée et sécurité>Permissions".
 - c) Ensuite à la ligne "Empêcher la lecture automatique de contenus sonores", cliquez sur "Exceptions".
 - d) Collez l'adresse du jeu dans "Adresse du site web", puis cliquez "Autorisez" et "Enregistrer les modifications".

Pour jouer au jeu :

- 1) Entrez un pseudo et cliquez sur "commencer".
- 2) Choisissez un des trois niveaux :
 - a) Facile : il n'y a que des briques normales et des briques donnant des bonus.
 - b) Intermédiaire : En plus des briques normales ou donnant des bonus, il y a des briques qui se cassent en deux coups.
 - c) Difficile : En plus des briques normales ou donnant des bonus, il y a des briques incassables.
- 3) Pour déplacer la planche, utilisez les touches "q" et "d" ou les flèches "←" et "→".
- 4) Si vous cassez une brique bonus, votre bonus durera pendant 5 sec. Sauf dans deux cas :
 - a) Si vous avez la brique aimantée, vous pouvez arrêter l'aimantation avant en appuyant sur "z" ou sur "↑".
 - b) Si vous avez les deux balles supplémentaires elles resteront tant qu'elles ne touchent pas le sol.
- 5) Pour mettre le jeu en pause vous cliquez sur le bouton "⏸" et pour reprendre "□".
- 6) Pour activer/ désactiver le son ou la musique, cliquez sur les boutons "🎵" et "🔊".
- 7) A la fin de la partie, vous pouvez la rejouer en cliquant sur "rejouer" ou commencer une nouvelle partie avec "retour".

Les types de briques :



: La brique normale. Elle se casse quand une balle ou un boulet la touche.



: La brique cassable en deux fois. La première fois qu'elle est touchée elle se transforme en brique normale.



: La brique incassable.



: La brique à triples balles. Cassez la pour voir votre nombre de balles augmenter de 200%!



: La brique canon. Cette brique vous permettra d'avoir deux canons sur votre planche pour casser encore plus de briques.



: La brique aimantée. Cassez cette brique et c'est la balle viendra à vous. Vous pourrez ensuite la relancer là vous voulez.



: Cette brique permet d'agrandir votre planche si vous la cassez.

Choix techniques :

Pour l'implémentation de ce projet nous avons décidé d'utiliser des classes. Vous les trouverez dans le dossier "javascript/class".

- La classe Balle contient les méthodes :
 - renderBalle() : permet d'afficher la balle.
 - moveBalle() : permet de déplacer la balle.
 - rebondMur() : change la direction sur l'axe x de la balle quand elle touche un mur.
 - rebondBrique() : change la direction sur l'axe y de la balle quand elle touche une brique.
 - rebondPlanche() : change la direction sur l'axe y de la balle quand elle touche la planche.
 - aimantation() : colle la balle à la planche.
 - finAimant() : relance la balle vers les briques.
 - restoreSettings() : remet les paramètres de la balle comme au lancement de la partie.
- La classe Board contient la méthode :
 - clearGameBoard() : efface tout ce qui est dans le canvas.
- La classe Boulet contient les méthodes :
 - renderBoulet() : permet d'afficher un boulet.
 - moveBoulet() : permet de déplacer un boulet.
 - setDestroy() : détruit un boulet.

- La classe Briques contient les méthodes :
 - renderBriques() : permet d'afficher une brique.
 - setDestroy() : détruit une brique.
 - setCasseX2() : transforme une brique cassable en deux fois en une brique normal.
 - restoreSettings() : remet le paramètre "destroy" de la brique comme au lancement de la partie.
- La classe Game contient les méthodes :
 - runGame() : appelle les méthodes qui permettent le fonctionnement du jeu à intervalle régulier.
 - stopGame() : arrête le jeu et affiche le menu pour rejouer ou quitter.
 - restartGame() : remet les paramètres du jeu comme au lancement de la partie.
 - checkBalleCollision() : vérifie si la balle touche le sol et diminue le nombre de vie si c'est le cas.
 - updateScore() : augmente le score.
 - checkBriquesBalleCollision() : vérifie si la balle est entrée en collision avec une brique, et si c'est le cas active un bonus si c'est une brique bonus.
 - checkBouletCollision() : vérifie si un boulet touche une brique. Si c'est le cas la brique et le boulet sont détruits.
 - gameAnimation() : appelle les différentes méthodes de tests et d'affichage et de déplacements.
 - pauseGame() : met le jeu en pause.
 - reprendreGame() : arrête la pause.
- La classe Planche contient les méthodes :
 - renderPlanche() : permet d'afficher la planche.
 - renderCanon() : permet d'afficher les canons.
 - movePlanche() : permet de déplacer la planche.
 - controlPlanche() : permet de contrôler le déplacement de la planche.
 - agrandissement() : permet d'agrandir la planche.
 - rapetissement() : permet de rapetisser la planche.
 - restoreSettings(): remet les paramètres de la planche comme au lancement de la partie.

L'initialisation des différentes variables et l'appelle des méthodes se fait dans app.js.

Conclusion :

Pour réaliser ce travail nous nous sommes répartis les tâches. Mika s'est en majorité occupé de programmer les fonctionnalités de bases (faire le rendu des briques, le déplacement de la balle et les différents bonus). Quant à Adonis s'est surtout occupé d'organiser le code proprement, de créer les différents niveaux et de l'affichage et du sound design. Cependant nous nous sommes beaucoup entraider dans la réalisation de chacune de ces tâches et le travail a été équitablement partagé.

Ce fut un projet intéressant même si nous avons parfois rencontré quelques difficultés. En effet à plusieurs reprises, l'ajout d'une fonctionnalité ou d'un bonus nous obligeait à modifier le code déjà existant. De plus l'affichage des dix meilleurs scores nous a posé problème.