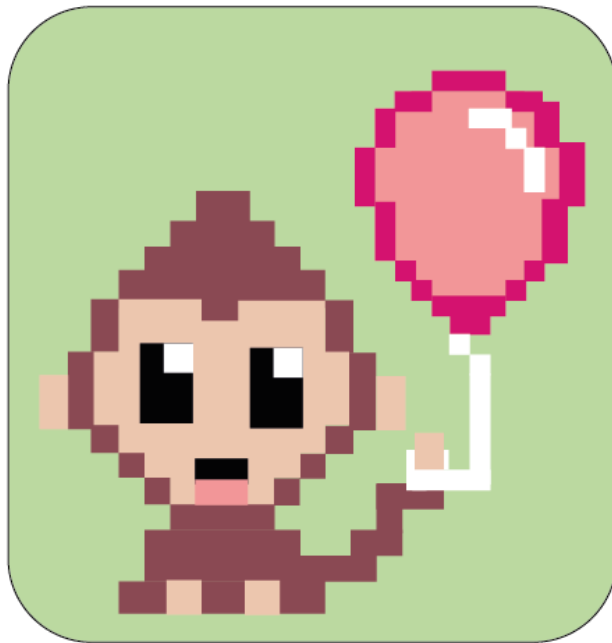




LES
TROPHÉES NSI

Édition 2023

DOSSIER DE CANDIDATURE
PRÉSENTATION DU PROJET



NOM DU PROJET : Monkey Balls

> PRÉSENTATION GÉNÉRALE :

Nous avons développé un jeu grâce au module pyxel.

Dans notre jeu, le joueur doit utiliser des singes qui jettent des darts pour faire éclater des ballons colorés sur l'écran. Le jeu est assez simple, il valorise la stratégie du joueur.

Au début de chaque jeu, le joueur a un certain nombre de vies qui varient en fonction de la difficulté du niveau. (**Facile : 200 vies; Moyen : 150 vies; Difficile : 100 vies; CHIMPS: 1 vie**) Ensuite, une certaine quantité de ballons dépendante de la manche actuelle (disponible sur niveaux.py) est affichée sur l'écran. Le joueur doit utiliser un certain nombre de singes (**qu'il peut acheter**) pour éclater tous les ballons et passer à la prochaine manche. A chaque fois qu'il éclate un ballon, il gagne de l'argent, et a chaque fois qu'un ballon traverse le chemin, il perd une vie.

Le jeu dispose de plusieurs types de ballons avec différentes couleurs et capacités. Les ballons simples éclatent lorsqu'ils sont touchés par une fléchette, tandis que les ballons plus gros nécessitent plusieurs coups avant de se dégonfler.

Le joueur peut également acheter des power-ups qui donnent des avantages tels que des darts supplémentaires.

Le jeu comporte plusieurs niveaux, chacun avec une configuration différente de ballons et un niveau de difficulté croissant.

L'objectif c'est de ne pas laisser les ballons traverser tout le chemin.

COMMENT JOUER:

Pour jouer il suffit de suivre le "tutoriel":

Il faut faire un clic gauche sur les singes et les power ups pour les acheter et faire un clic gauche pour les afficher dans le map.

Pour visualiser les améliorations des flèches, il faut cliquer sur les singes que vous souhaitez améliorer.

La touche M peut être pressée pour couper le son des éclats des ballons.

> ORGANISATION DU TRAVAIL :

Le contrôle de versions du code à été fait sur GitHub, et peut être retrouvé sur le lien suivant: [GitHub](#)

La communication a été faite grâce à Discord, une application permettant d'envoyer des messages, faire des appels et partager nos écrans entre nous.

Il a été fondamental de tester notre code, après chaque version à été déposée sur GitHub, nous nous sommes tous mis à faire plusieurs essais afin de trouver toutes les erreurs possibles.

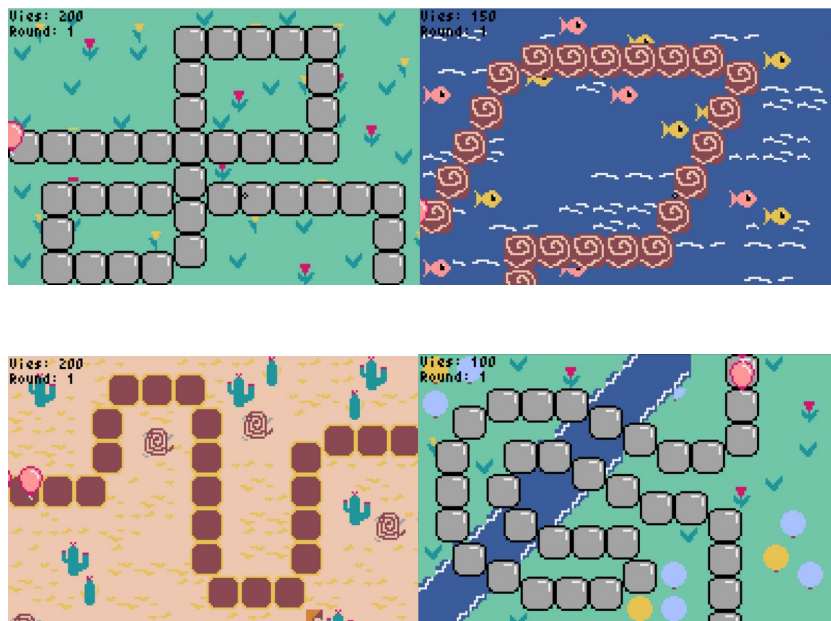
Pour le logo nous avons utilisé l'application Procreate de l'App Store

LES ÉTAPES DU PROJET :

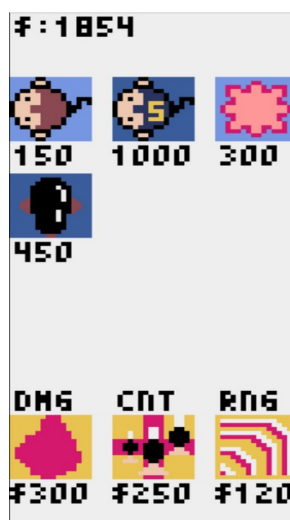
1. Créer des tests (les tests de collision, niveaux, translation)
2. Créer les singes et le ballons → implémenter la partie graphique
3. Faire le menu
4. Créer des différents monde
5. Implémenter les rounds
6. Créer les différents niveaux et difficultés
7. Jouer et tester le jeu

GRAPHIQUE / PYXEL EDIT

Le joueur peut choisir entre 4 mondes différents.



- Le "shop":



- Le menu :



Avec le pyxel edit on a été aussi capable de mettre un son quand les ballons sont touchés par les darts.

> FONCTIONNEMENT ET OPÉRATIONNALITÉ :

DIFFICULTÉS

- Le calcul pour vérifier que le ballon a été touché par la flèche.

```
#On attaque les ballons
for singe in self.liste_singes:
    tirs = 0
    for ballon in self.liste_ballons:
        if fonctions_aux.distance((singe.x, singe.y),(ballon.x, ballon.y)) \
            <= singe.range**2+singe.range \
            and pyxel.frame_count % singe.cooldown == 0:
```

- Dans notre fichier niveaux.py on utilise un tableau avec le nombre et le type du ballon. Comme c'est un grand tableau, de 100 lignes, on a dû toujours faire beaucoup d'attention pour ne pas se tromper.

```
tab = [ [(20, "Rouge"),
        [(35, "Rouge"),
        [(25, "Rouge"), (5, "Bleu")],
        [(35, "Rouge"), (18, "Bleu")],
        [(5, "Rouge"), (27, "Bleu")],
        [(15, "Rouge"), (15, "Bleu"), (5, "Vert")],
        [(20, "Rouge"), (20, "Bleu"), (5, "Vert")],
        [(10, "Rouge"), (20, "Bleu"), (14, "Vert")],
        [(30, "Vert")],
        [(102, "Bleu")],
        [(10, "Rouge"), (40, "Bleu"), (40, "Vert"), (20, "Rouge")]
```

VALIDATION DU FONCTIONNEMENT

Pour valider le fonctionnement du code, nous avons joué plusieurs parties pour vérifier que le jeu ne possède pas de bugs et qu'il suive les règles du jeu.

> OUVERTURE :

IDÉES D'AMÉLIORATION

- Comme c'est un jeu de tower défense, on a toujours la possibilité de créer plus de power-ups et différentes manières de ne pas laisser les ballons traverser toute le chemin.
- Améliorer la performance du code.
- Mettre le jeu en ligne avec un classement des meilleurs joueurs.

SOURCES

- Bloons Wiki

DOCUMENTATION

Le projet est basé sur le moteur Pyxel :

<https://github.com/kitao/pyxel/blob/main/docs/README.fr.md>

Il est donc indispensable d'installer le logiciel :

<https://github.com/kitao/pyxel/blob/main/docs/README.fr.md#comment-installer>