

# Projet Flippy

Par Fabio De Sousa  
Pierre Amorim  
Julien Chanez  
Adam Clisson

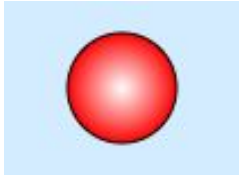
# Ce qui a été fait: -Présentation des items de base



Les flips qui frappent la balle, contrôlé par le joueur



Les cibles fixes permettant de marquer des points



Les bumpers repoussent la balle et permettent de marquer des points



Les trous téléportent la balle a un autre trou présent sur le flipper



Le launcher permet de lancer la balle en début de partie, contrôlé par le joueur



Les slingshot fonctionnent comme les bumpers, ils repoussent la balle



La balle

La zone de défaite, placé automatiquement sur les invisible, vous perdez si la balle tombe dedans. (E manuellement dans l'éditeur.)

# Ce qui a été fait: -Présentation des items particulier

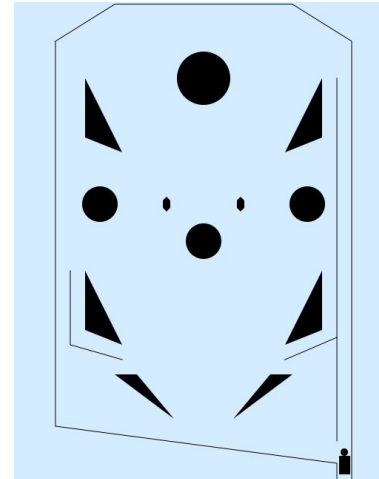
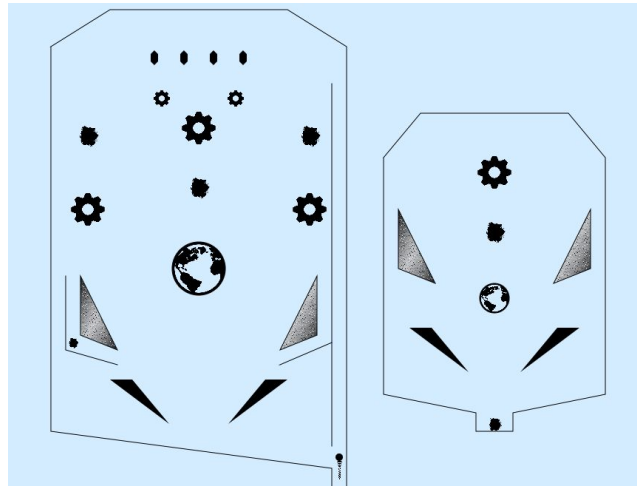
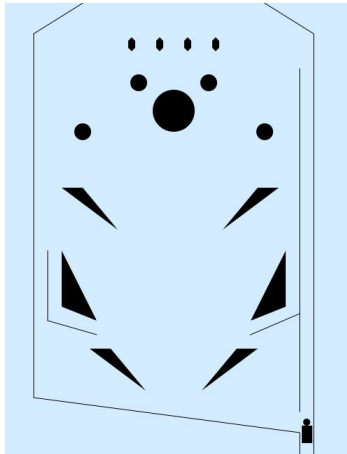
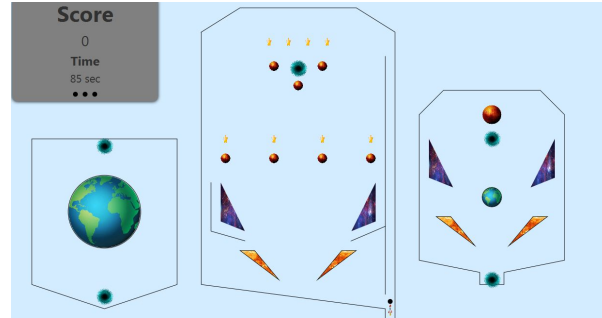
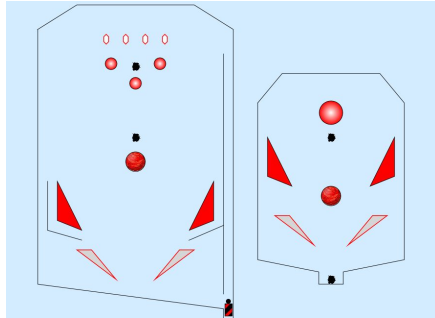


Les planètes attirent la  
balle qui “gravite”  
autour d’elles

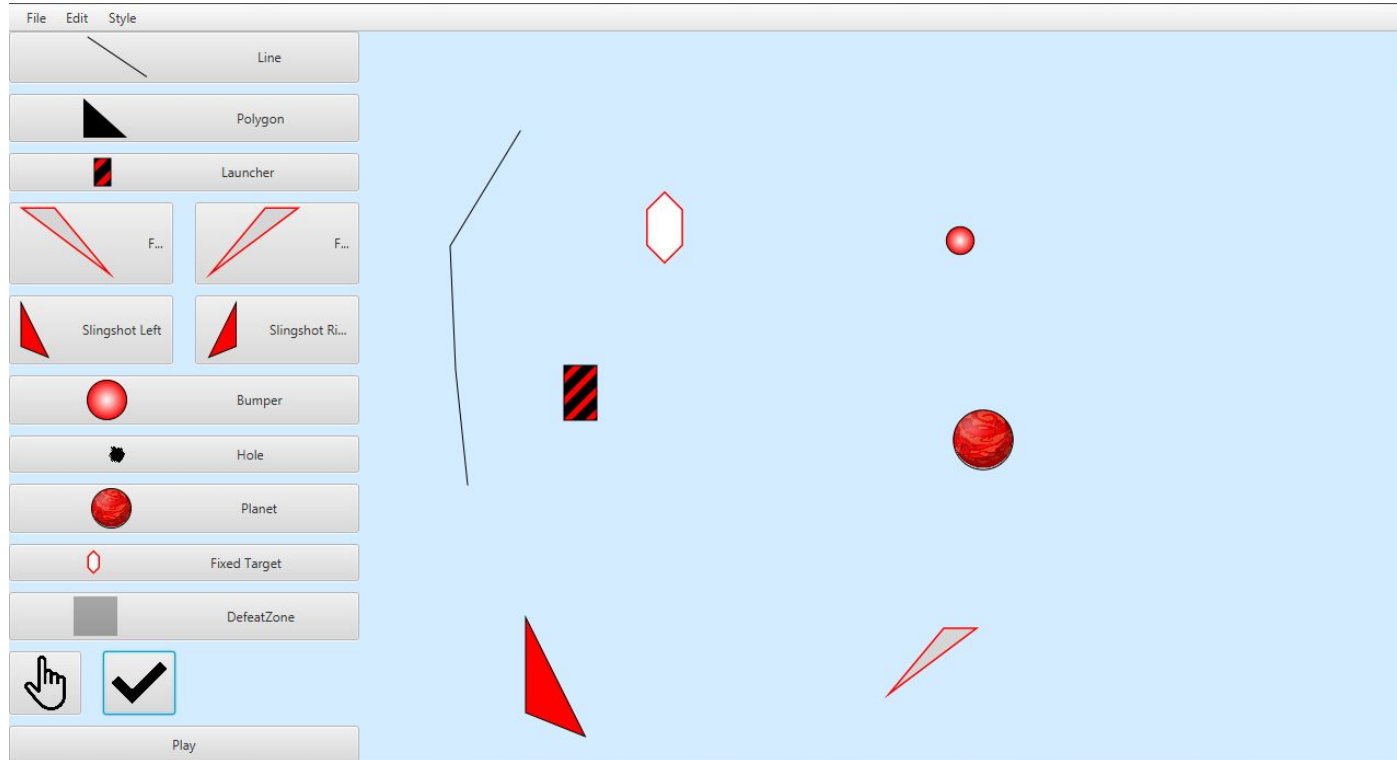


La superballe marquant  
2 fois plus de points  
qu’une balle normale

# Ce qui a été fait: -Les différents Flipper



# Ce qui a été fait: -L'éditeur



L'éditeur est le plus gros ajout par rapport à un flipper classique, il permet de créer le flipper que l'on veut, modifier la taille des objets et même de l'enregistrer

## Ce qui n'a pas été fait: -Flipper infini

Nous avons la volonté de créer un flipper infini, cependant cela aurait nécessité un algorithme de création des objets. Le but aurait été de monter le plus haut grâce aux flips. Il aurait ainsi fallu vérifier que des flips se place assez souvent pour pouvoir monter et que le nombre d'objets qui se place ne soit pas trop élever pour ne pas que le joueur soit bloqué. Le travail aurait été trop long pour le temps qui nous restait nous avons donc abandonné.

# Démo du projet: -Instructions de compilation

Le projet Maven contient un fichier Makefile, permettant de compiler. Il faut donc, dans le terminal, et dans le répertoire où se trouve le fichier, taper la commande ``make``. Une fois le projet compilé, il faut entrer la commande ``java -jar Flippy.jar``, qui s'occupera de lancer le jeu.

# Démo du projet: -Comment jouer

Une fois dans le jeu, vous pouvez choisir de soit jouer sur un flipper déjà créé en cliquant sur jouer puis choisir le flipper sur lequel vous voulez jouer avec les flèches sur les côtés de l'écran.

Une fois en jeu 4 touches pour jouer, les flèches gauche et droite du clavier pour contrôler les flips gauche et droit. la flèche du bas du clavier pour le launcher, restez appuyer sur la flèche du bas pour le tirer vers le bas puis relâchez pour l'envoyer plus vous restez appuyer plus la balle sera lancée vite. Et enfin la touche Echap pour mettre la partie en pause et retourner au menu.


Depuis le menu, vous pouvez aussi grâce à Options, modifier la résolution de l'écran, mettre en plein écran ou enlever le son.



# Démo du projet: -Faire son propre niveau

Pour créer son propre niveau commencez par cliquer sur Editor, d'ici vous pouvez cliquer sur n'importe quel items que vous désirez placer.

Pour les lignes, cliquez une fois pour placer le début de la ligne puis recliquez pour placer la fin, continuez de cliquer pour continuer le chemin avec une deuxième ligne le début de la deuxième ligne étant la fin de la première et etcetera pour la suite.

Dès que vous voulez arrêter cliquez sur le bouton valider  pour confirmer le chemin de lignes que vous avez créé.

Pour ce qui est des flips, du launcher, des slingshots, des cibles fixes, des bumpers et planètes, cliquez une première fois pour l'endroit où vous voulez placer l'item puis une deuxième fois pour la taille de l'item. Dès que l'item est placé et que vous êtes satisfait, cliquez sur le bouton valider avant d'en placer un autre.

# Démo du projet: -Faire son propre niveau


Et enfin pour les polygones et la zone de défaite, cliquez une fois pour placer un premier point et continuer de cliquer pour placer les autres sommets du polygones. Cliquez sur valider dès que vous êtes satisfait.

Quand vous êtes satisfait de votre niveau, vous pouvez y jouer en cliquant sur play, attention pour jouer il faut au moins un launcher et une zone de défaite, la balle elle, sera automatiquement créé au dessus du launcher.

En plus de ça, 3 boutons supplémentaires se trouvent en haut à gauche. -Le bouton style permet de modifier les couleurs et le design des objets.

-Le bouton edit, permettant de supprimer des objets ou de les copier-coller. Après avoir cliqué sur un objet du flipper avec sélection

# Démo du projet: -Faire son propre niveau

-Le bouton edit, permettant de supprimer des objets ou de les copier-coller. Après avoir cliqué sur un objet du flipper avec sélection  vous pouvez le supprimer avec delete, ou le copier avec copy/paste, une fois l'objet copié cliquez où vous désirez placer l'objet copié.

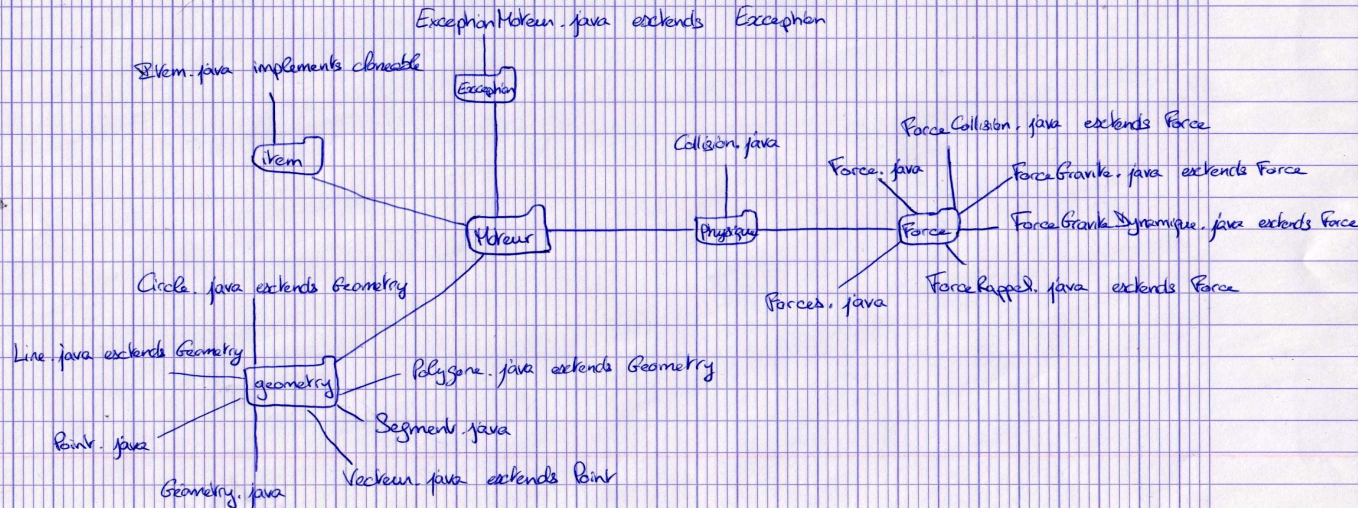
-Le bouton File permettant d'enregistrer le niveau ou de le recommencer, cliquez sur New pour supprimer le niveau et repartir sur un nouveau flipper. Cliquez sur Exit pour retourner au menu. Cliquez sur save pour sauvegarder le niveau (nous vous conseillons de le sauvegarder à Flipper/src/main/res avec les autres flippers). Et enfin cliquez sur Import pour importer un niveau créé auparavant.

# Choix techniques:

-Nous avons opté pour un projet Maven divisé en 3 sous projets, une partie Moteur pour la physique, une partie flipper pour tous les éléments propres aux flipper comme les items, et enfin une partie interface graphique pour l'affichage des pages.

La partie interface graphique a été faite sur JavaFx notamment pour la méthode `animationTimer` permettant de rafraichir la page en continu mais aussi pour avoir accès aux formes géométriques permettant de dessiner les objets.

# Architecture du projet: -Partie Moteur



La partie moteur pour les forces et la geometry pas directement relié au flipper



```
classDiagram
    class Item
    class SlingShotLeft
    class SlingShotRight
    class SlingShot
    class DefeatZone
    class FlipRight
    class FlipLeft
    class Flip
    class Ball
    class TangleFixed
    class Planet
    class Hole
    class Launcher
    class Bumper
    class Zlem
    class ReaderPLP
    class Flipper
    class Exception
    class ExceptionPLPMethod
    class ExceptionObjectPLPZincanner
    class WrittenPLP
    class ReactionPLP
    class Game
    class FlipperJava

    Item <|-- SlingShotLeft
    Item <|-- SlingShotRight
    Item <|-- SlingShot
    Item <|-- DefeatZone
    Item <|-- FlipRight
    Item <|-- FlipLeft
    Item <|-- Flip
    Item <|-- Ball
    Item <|-- TangleFixed
    Item <|-- Planet
    Item <|-- Hole
    Item <|-- Launcher
    Item <|-- Bumper
    Item <|-- Zlem
    Item --> ReaderPLP
    Item --> Flipper
    Item --> Exception
    ReaderPLP --> WrittenPLP
    ReaderPLP --> ReactionPLP
    Flipper --> Game
    Flipper --> ExceptionPLPMethod
    ExceptionPLPMethod --> Exception
    ExceptionObjectPLPZincanner --> Exception
    Flipper --> FlipperJava
```

La partie flipper pour les items propre au flipper

# Architecture du projet: -Partie Interface graphique

La partie interface graphique pour l'affichage des différentes pages

