

ACTION PLATFORMER - INTERPRETATION DES 3C

LE CONTROLLER

I. POUCE GAUCHE

1. GENERALITES

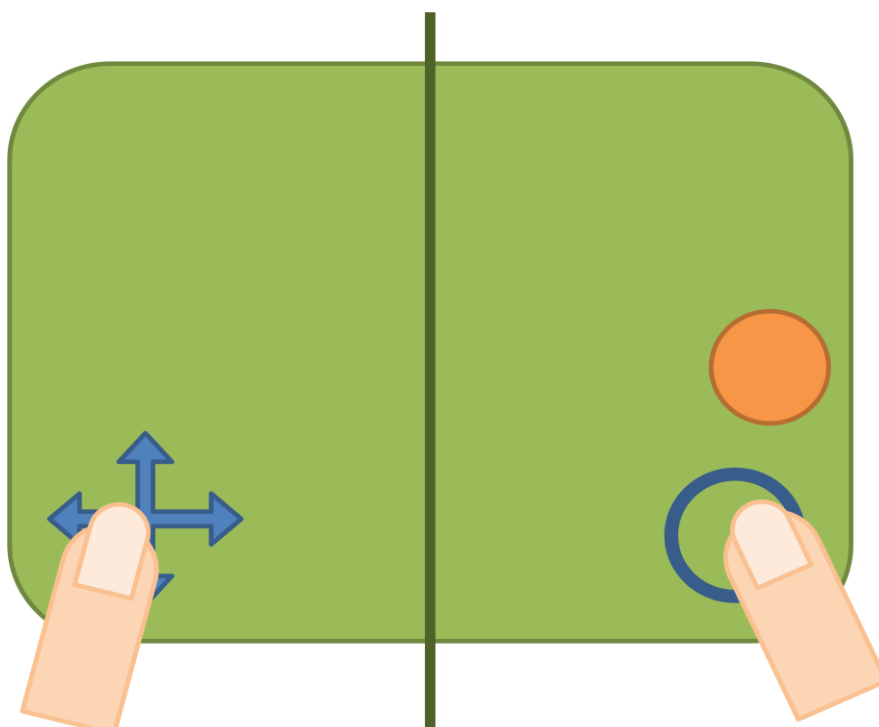
- La main gauche contrôle le déplacement du personnage vers la gauche ou la droite.
- La zone de détection de l'input est la moitié gauche de l'écran.

2. DETAILS

- Sur l'événement touchStart la position d'origine est enregistrée (point Zéro)
- Sur l'événement touchMove, la nouvelle position est enregistrée, puis l'écart sur l'axe des X est calculé. Si l'écart au point Zéro est positif (et supérieur à un seuil minimal) la variable « right » du controller passe à true. Si l'écart est négatif et supérieur au seuil, la variable « left » du controller passe à true.

***Note :** Si l'écart dépasse une certaine distance, le changement de direction peut devenir difficile car il faut parcourir une grande distance pour revenir au point Zéro et passer de l'autre côté. Pour empêcher ce problème, le point Zéro peut suivre le déplacement pour rester à une distance pas trop grande du pouce et faciliter le retour au centre et le changement de direction.*

- Sur l'événement touchEnd, les positions sont effacées.



- En bleu les inputs ne dépendent que du demi-écran (**pas de position spéciale**)

- En orange l'éventuel bouton virtuel (**taille et position à définir**)

II. POUCE DROIT

1. GÉNÉRALITÉS

- La main droite contrôle les sauts et les tirs du personnage.
- La détection de l'input se fait sur la moitié droite de l'écran.
- 2 configurations sont envisagées : sans boutons virtuels et avec 1 bouton virtuel

2. CONFIGURATION SANS BOUTONS VIRTUELS

Dans cette config, aucun bouton virtuel n'est utilisé et l'ensemble de la moitié droite de l'écran est utilisée pour détecter les inputs.

- Un **swipe vers le haut** (move vers le haut dans le quart de cercle qui regarde vers le haut par rapport à l'origine, offsetY supérieur à un seuil minimal) entraîne le passage à true de la variable controller.jump
- Un **tap** entraîne le passage à true de la variable controller.fire
- Un **hold** entraîne le passage à true de la variable controller.chargedFire

Note : *Le swipe n'est peut être pas assez dynamique pour le saut. De plus le double swipe pour un double saut peut s'avérer désagréable si le pouce accroche un peu trop la surface tactile. De plus le swipe peut rendre le contrôle de la hauteur du saut plus difficile qu'avec une durée de tap.*

3. CONFIGURATION AVEC 1 BOUTON VIRTUEL

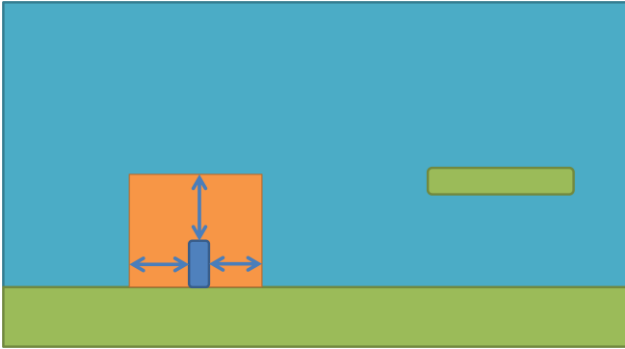
Dans cette config, on a 2 zones différentes dans la moitié droite de l'écran : un gros bouton, et le reste du demi-écran.

- Un **tap sur le demi-écran** entraîne le passage à true de la variable controller.jump
- Un **tap sur le bouton** entraîne le passage à true de la variable controller.fire
- Un **hold sur le bouton** entraîne le passage à true de la variable controller.chargedFire

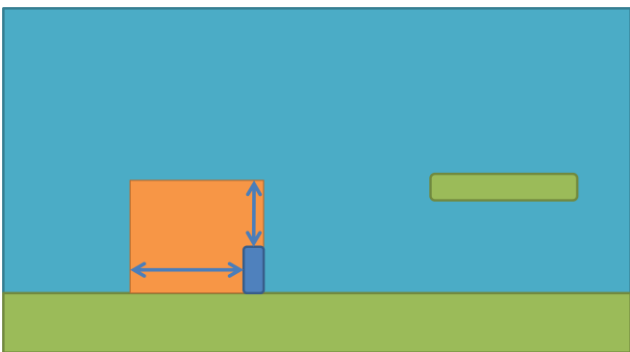
Note : *à tester l'inversion des 2 contrôles (jump sur le bouton et fire sur le reste), ainsi que la taille et la position du bouton. On préférerait éviter les boutons virtuels mais les problèmes évoqués dans la note de fin de la config 1 peuvent s'avérer trop gênants. Eventuellement proposer les 2 configurations. Avec la meilleure des 2 par défaut.*

[Tapez un texte]

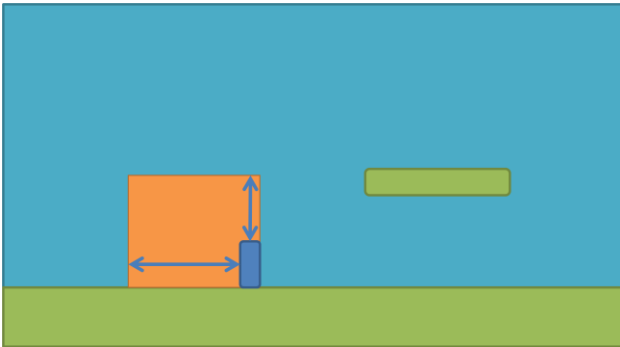
LA CAMERA



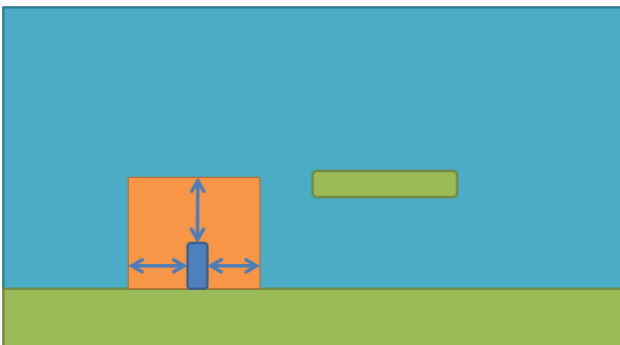
Le souhait est de faire une camera souple qui suit le personnage. Pour cela une boxCamera (en orange). ***Un offset X à 1/3 de l'écran sera testé.***



Le personnage est libre de bouger dans la boxCamera (en orange) sans faire bouger la camera



S'il va au-delà de la box, il pousse la box et la caméra suit le mouvement avec un tween.

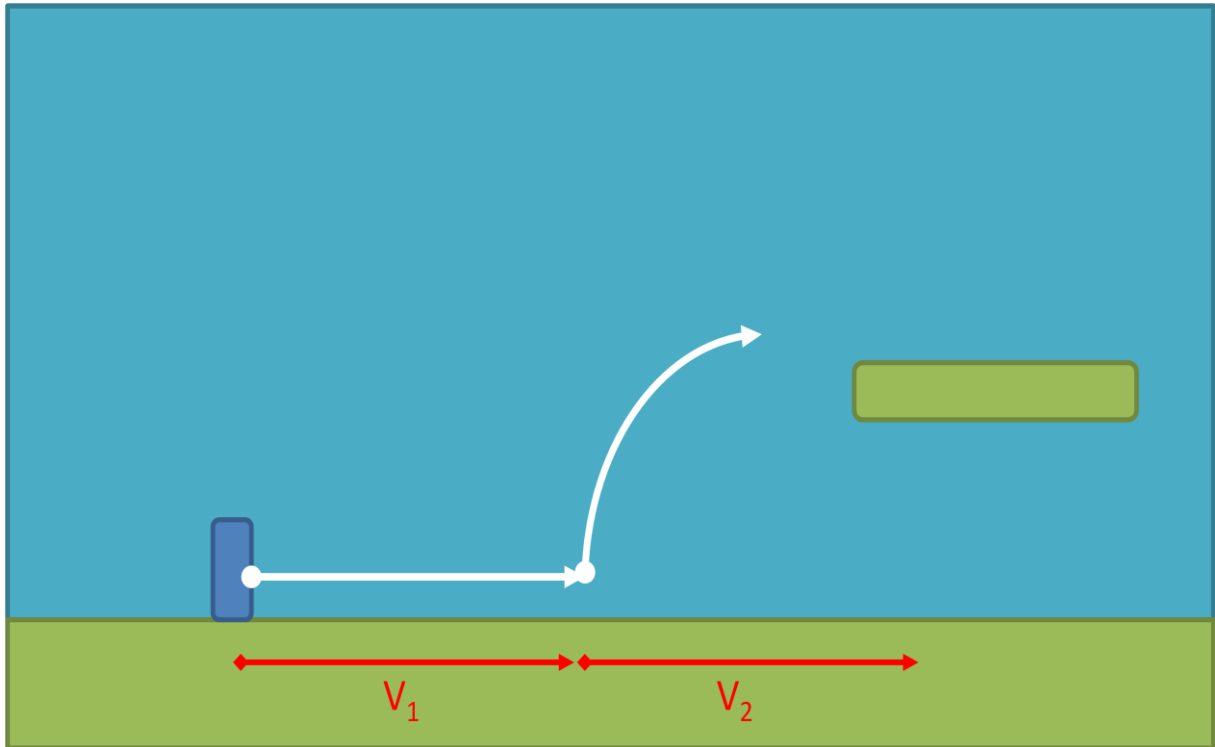


Eventuellement, à tester, la camera se repositionne par rapport au personnage

Pour le saut, on imagine une première version simple, fixée au sol, puis une version qui gère les plateformes.

LE CHARACTER

I. LE SAUT



1. QUESTION DE LA VITESSE MAX V_2 PENDANT LE SAUT

L'idée est de la laisser strictement égale à la V_{\max} au sol. Si $V_1 = V_{\max}$, $V_2 = V_{\max}$.

2. QUESTION DU CONTROLE HORIZONTAL PENDANT LE SAUT

Une fois qu'on a sauté peut-on accélérer ? (partir d'un saut arrêté et finir par avancer ?)

Une fois qu'on a sauté peut-on décélérer ? (partir d'un saut à V_{\max} et s'arrêter quand on lâche le controller ?)

L'idée serait que Oui et Oui, avec une accélération diminuée à 60-90% de sa valeur normale au sol.

3. QUESTION DE LA HAUTEUR DE SAUT VARIABLE

La hauteur de saut pourrait varier selon l'input. Si le saut est sur un tap/hold, selon la durée du hold, et si le saut est sur un swipe, selon la vitesse du swipe (pour écarter tout problème de distance lié à la taille d'écran). C'est un principe intéressant, mais il faut vérifier si c'est vraiment indispensable.

Le double saut devrait se comporter similairement au saut simple.

II. LE TIR

1. QUESTION DU TIR PENDANT LA CHARGE DU SUPER TIR

Si le tir est déclenché sur le touchEnd, c'est la durée entre le touchStart et le touchEnd qui compte. Au-delà d'un certain seuil, le super tir commence à se charger et le personnage ne tire plus ses tirs normaux. Au moment du touchEnd, le super tir est déclenché si la durée de charge a été suffisante.

- Un hold ne déclenche un tir normal que si le tir chargé n'a pas été chargé assez longtemps.

Si le tir est déclenché sur le touchStart, un tir normal est tiré dans tous les cas. Si l'input est maintenu pendant x millisecondes le tir commence à charger et y millisecondes de plus pour finaliser la charge. Au touchEnd le super tir est tiré.

- Un hold déclenche donc toujours un tir normal, et éventuellement un super tir.

Nous souhaitons tester les deux options pour déterminer la meilleure sensation de jeu.

III. LES CAPACITES SPECIALES

1. LE DOUBLE SAUT

Comme dit plus haut, le deuxième saut se comporterait comme le saut simple, si ce n'est qu'il peut être déclenché dans l'air.

2. LE SUPER TIR

Le tir chargé devrait vraisemblablement entraîner un léger recul du personnage pour augmenter la sensation de puissance du tir et ajouter du fun et un peu de challenge (tir chargé depuis un rebord de plateforme).

L'intention est un temps de charge plutôt rapide et une utilisation sans cooldown. L'idée est que le temps de charge sera suffisant pour contraindre l'utilisation du super tir à des situations qui n'exigent pas une grande cadence de tir.

3. LE BOUCLIER

On imagine un bouclier utilisable 3 à 4 fois par niveau, pour un coût de 15 à 20 collectibles (sur une centaine dans le niveau). Valeurs à affiner.

4. L'AIMANT A COLLECTIBLES

On part sur un aimant de rayon modéré (1 largeur de box dans chaque direction), mais la encore la taille sera peut-être augmentée en fonction des tests.

RESUME

PLAN DE TESTS POUR LA SEMAINE DU 20 AU 26 AVRIL

CINDY

CAMERA :

- boxCamera plus grande que la box du personnage
- suit le personnage pour qu'il reste dans la boxCamera, mais fixée au sol
- tween pour un mouvement souple de la camera
- réglage de la taille de la boxCamera et du tween
- mode plateformes

BENJAMIN

PHYSIQUE DU PERSONNAGE :

- Coder la physique du saut/double saut
- Régler les frottements de l'air (au bout de combien de temps le personnage s'arrête quand on lâche)
- Régler l'accélération en saut
- Voir comment gérer la hauteur variable

CEDRIC

CAPACITES SPECIALES

- Voir si le tir se déclenche au touchStart ou au touchEnd par un test des deux options
- Mettre en place le recul du super tir
- Test du rayon de l'Aimant à Collectibles

CYPRIEN

CONTROLLER AVEC OU SANS BOUTON VIRTUEL

- Tester les deux configurations de controller décrites plus haut.

PHYSIQUE DU PERSONNAGE :

- Régler les frottements du sol (au bout de combien de temps le personnage s'arrête quand on lâche)