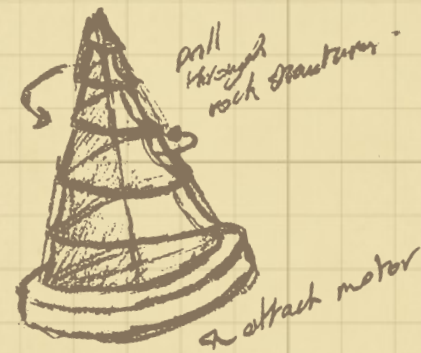
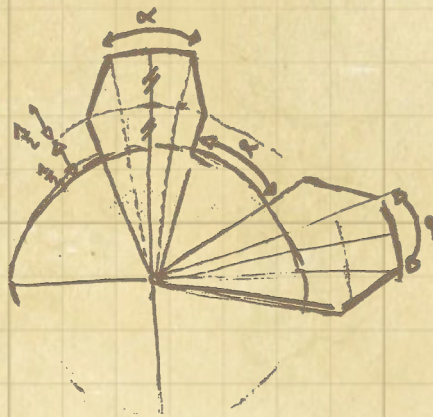
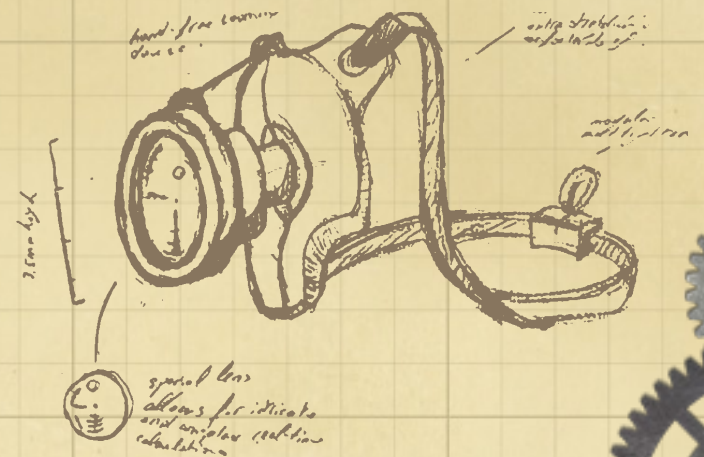
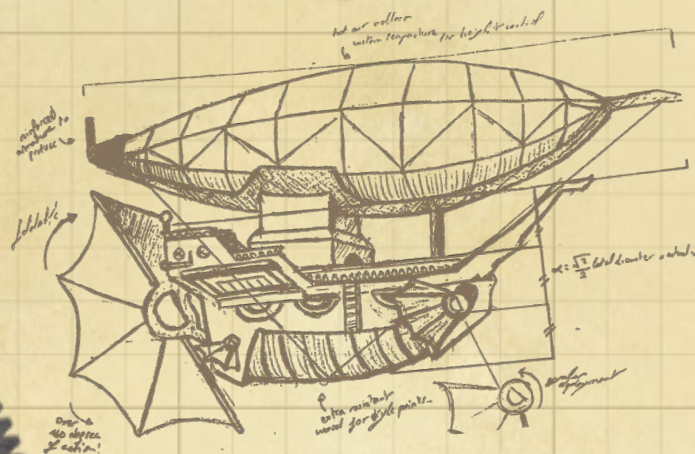




GEARS



ican
institut de création et
animation numériques

2022/2023

Sommaire

Equipe	3	Ingrédient	27
Introduction	4	I/Les Rouages	
GOD	5	II/Le Moteur	
Fiche d'identité	6	Les Connecteurs	33
Présentation & Game système	7	I/Le Pont	
3CS	8	II/Le Compteur	
Simulation d'une partie	9	Direction Artistique Visuelle	36
Tension Ludique	10	Pitch	37
Game Design Document	12	Intention Générale	38
Game Design	13	MoodBoard et Concept Art Global	39
Caméra	14	Décisions de Design	40
Mécaniques de jeu	18	I/Couleurs et matériaux	
I/Créer un objet		II/BackGround	
II/Prendre un objet		III/Interface	
III/Poser un objet		IV/Assets 3D	
IV/Déplacer un objet		V/Collection de dessins conçus pour Gears	
Les modes d'interactions	25	Direction Artistique Sonore	52
		Intention générale	51
		Son d'engrenage	52
		Son d'ambiance	53
		Event List	54

Equipe

Samuel Jeannoutot

Game Design
Sound Design
Organisation

Gabriel Bois

Game Design
Programmation
Documentation

Adrian Bednarowicz

Game Design
Direction Artistique
Production 2D et 3D

Maxime Leroi

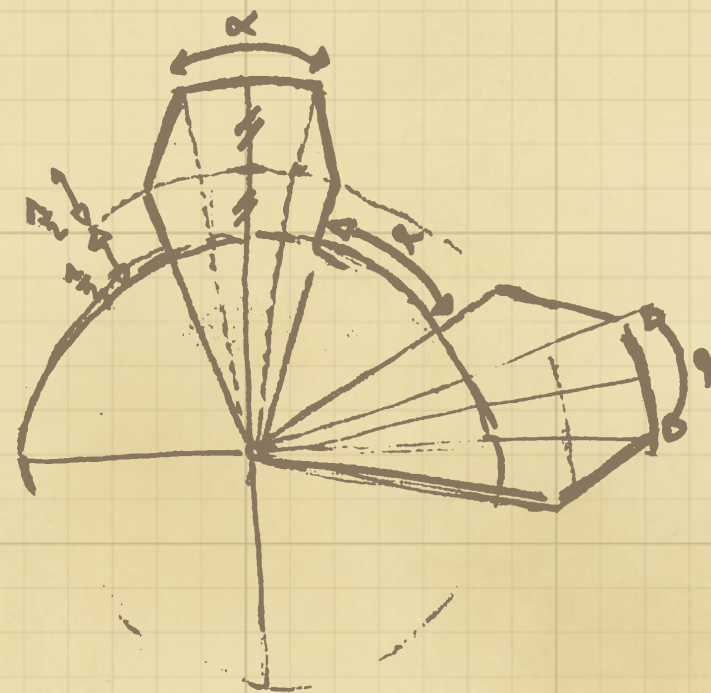
Game Design
Programmation
Production 2D

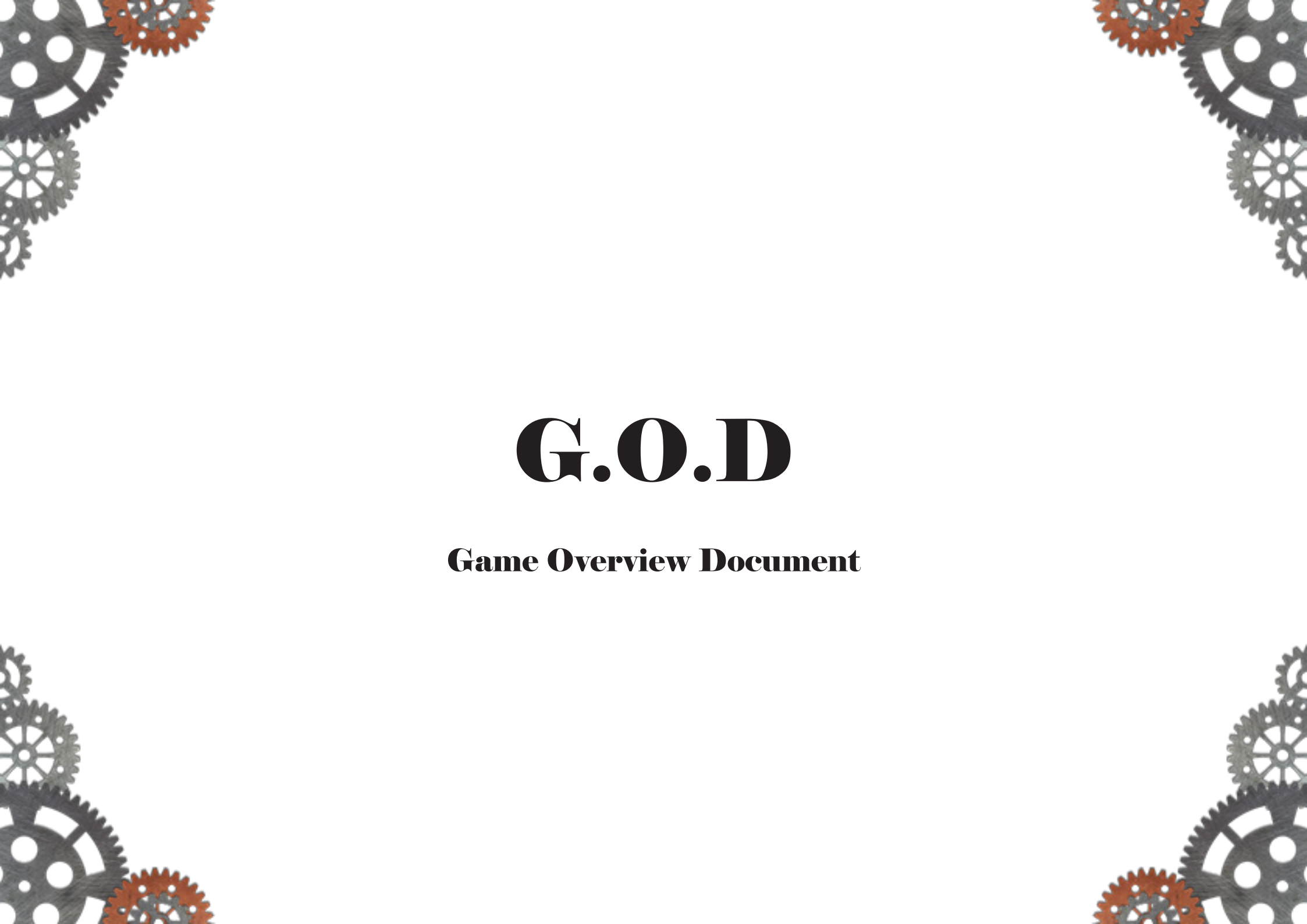
Introduction

Gears est un projet réalisé durant notre 2ème année à l'ICAN.

Ce projet est issu d'un travail étalé sur un semestre avec comme condition de faire un jouet 3D réalisé sur Unity et respectant le thème «MIN/MAX».

En présentant Gears nous voulons donner aux différents joueurs une expérience unique dans laquelle ceux-ci peuvent se plonger dans un univers «steam punk». Soit en créant des systèmes d'engrenages complexes ou bien tout simplement en observant l'esthétique du jouet, nous voulons que nos joueurs puissent se créer leurs propres objectifs ainsi que leur propre façon de jouer.



The page features decorative gear patterns in the corners. The top-left and bottom-right corners have clusters of interlocking gears in dark grey and reddish-brown. The bottom-left corner also shows a similar gear cluster. The top-right corner is partially obscured by the edge of the page.

G.O.D

Game Overview Document

Fiche d'identité

Pitch

Le joueur est un inventeur élaborant différents systèmes à partir d'engrenage dans un univers steampunk.

Support

Windows, Mac IOS et Linux

Cible

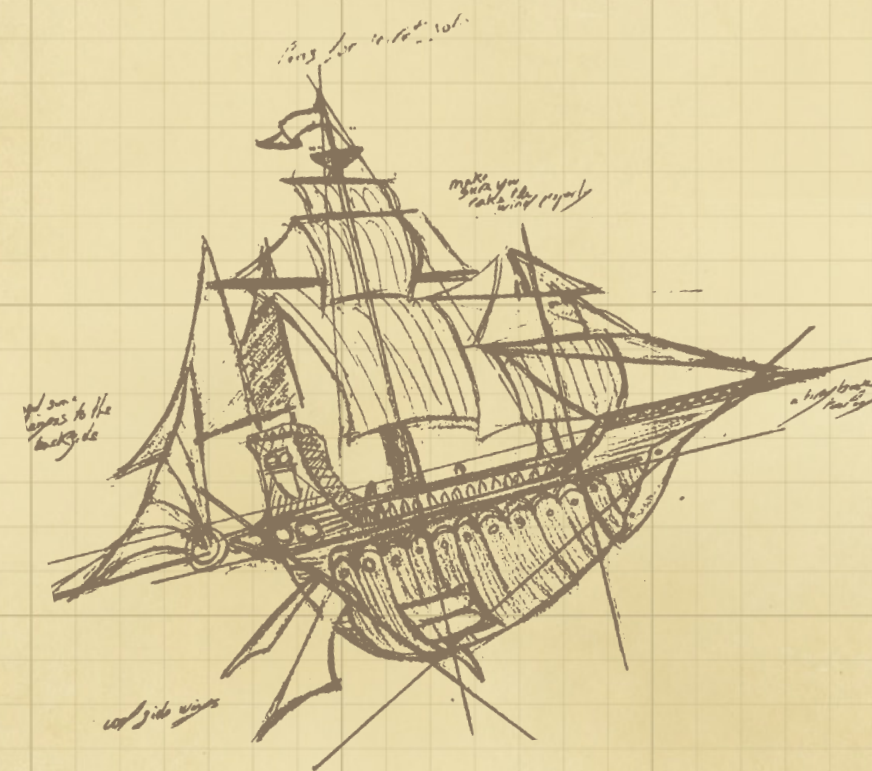
Casual gamer à la recherche d'une expérience contemplative et reposante ou hardcore gamer à voulant créer des systèmes complexes.

Type de jouet

Construction Bac à sable en 3D à la 3ème personne sur un plan 2D

Concept

Dans Gears le joueur s'amuse à construire des systèmes mécaniques simple ou plus complexe.



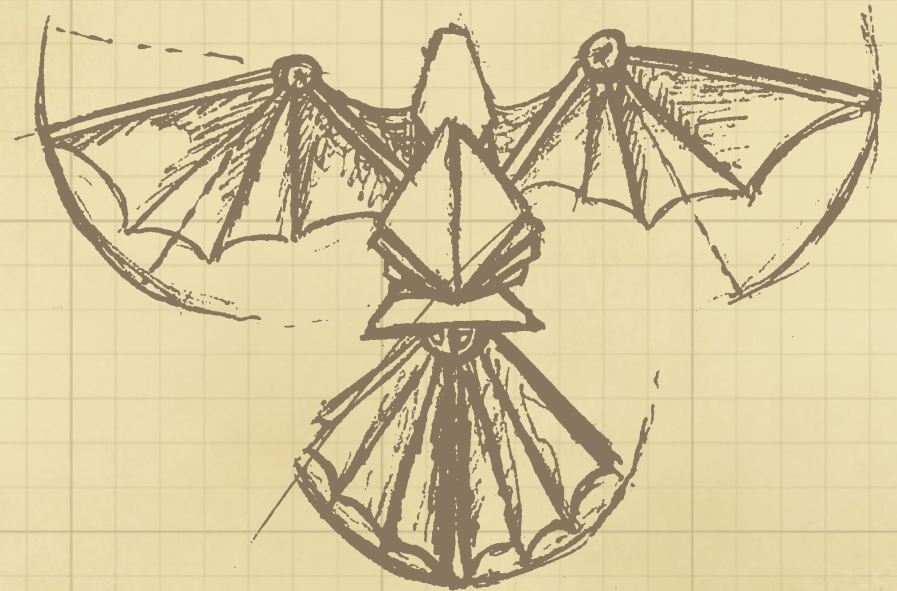
Présentation

Gears est un jouet de bac à sable interactif qui permet au joueur de placer des engrenages et divers objets sur un plan 2D. Il est doté de deux modes de jeu distincts : le mode «édition», dans lequel le joueur peut créer, modifier et conceptualiser ses systèmes, et le mode «observation», dans lequel le joueur peut visionner le fonctionnement de ses systèmes et apporter des modifications à des objets interactifs.

Game Système

Notre projet est un jouet de construction où le joueur peut placer, déplacer ou détruire des objets qui sont catégoriser en 3 catégories (Engrenages, Moteur, Connecteurs)

Ces différents objets interagissent entre eux par la transimision d'un mouvement rotatif.



3CS

Controls

Le jeu se joue uniquement à la souris grâce aux boutons de la souris et à la molette.

Character

Le character un curseur. Il peut créer, déplacer et détruire des objets. Pour créer et détruire des objets, l'avatar doit déplacer un engrenage dans la zone dédiée (Le plateau pour créer des engrenages ou dans la poubelle pour le supprimer).

Défaut



Attrapé

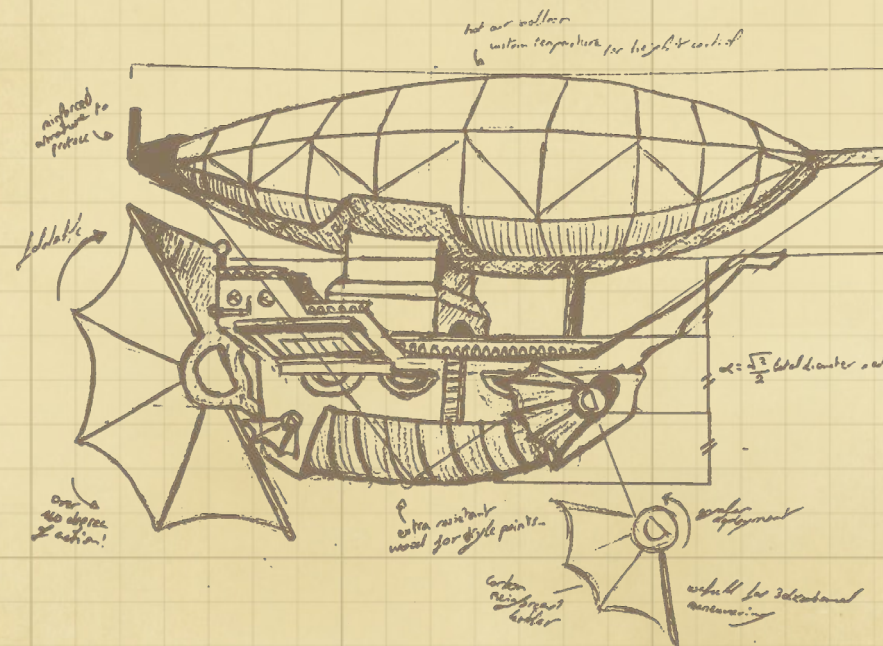


Sélectionné

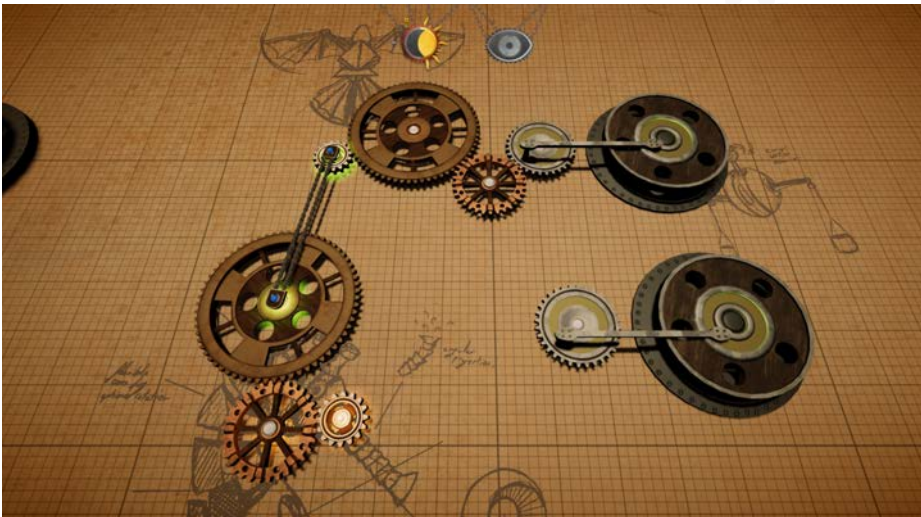


Camera

La caméra est une caméra mobile passant d'une vue Top-Down vers une vue panoramique grâce au zoom de la molette permettant au joueur de profiter de plusieurs point de vue pour voir évoluer son monde.



Simulation d'une partie



Tension Ludique

La tension ludique de Gears naît de l'opposition marquée entre les deux principales tendances vers lesquelles tend le système.

D'un côté, la création, la conception, et l'accumulation. Avec les outils dont le joueur dispose, le joueur peut créer et accroître des systèmes, tant en taille qu'en complexité. Il peut disposer librement des objets partout dans l'espace de jeu, ainsi que les agencer les uns avec les autres, en se pliant et s'adaptant aux règles du système. On se place ici dans une démarche de créativité: le joueur s'exprime et expérimente, se plaçant dans une relation active avec le jouet.

De l'autre côté, la contemplation, l'observation, la relaxation. Avec sa multitude de signaux, et le fonctionnement répétitif et immuable de ses systèmes, Gears présente un système de jeu hypnotique, face auquel le joueur va se poser dans une démarche passive: regarder plus que jouer, admirer ce qu'il a créé.

Ce qui alimente la tension ludique, c'est le clash, la tension entre ces deux tendances vers lesquelles le jouet emporte le joueur. C'est d'autant plus marqué et mis en avant avec la scission que représente le changement de mode. La création active, face à l'appréciation passive de ce qu'on a créé.

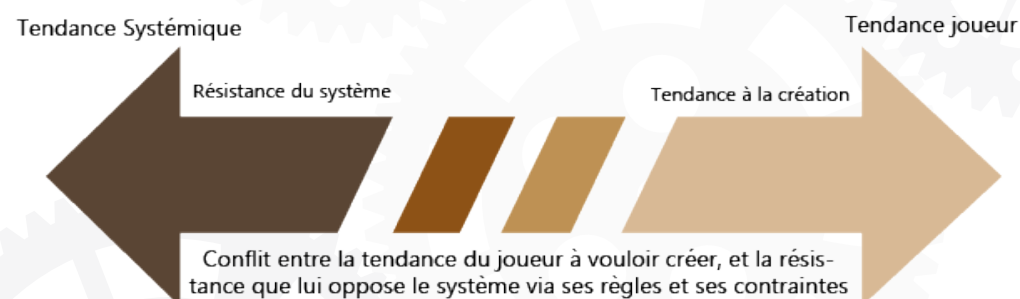
Ce n'est cependant pas du sens unique, l'utilisation du jouet est marquée d'aller-retours entre ces deux modes, et le joueur ne reste pas constamment dans un état, il alterne entre les deux.



Si le mode contemplation est passif, et met relativement peu de jeu entre le joueur et le système, le mode édition est bien plus actif. En particulier, il est marqué par l'opposition constante entre le joueur, qui veut créer, et le système, qui lui résiste. Celui-ci est en effet constitué de multiples règles, de lois, avec lesquelles le joueur va devoir expérimenter pour les comprendre, et apprendre à les maîtriser.

Cette résistance se présente principalement via les contraintes qui résident autour de l'action de Poser un objet: certaines actions ne fonctionnent pas, le système se briserait, et le joueur se verra fréquemment refuser son positionnement s'il se lance dans la construction de systèmes complexes et précis sans comprendre parfaitement comment ses composantes fonctionnent.

En outre, il n'existe pas de guide, ou de tutoriel sur ce que fait chaque objet, et de comment il interagit avec les autres. L'apprentissage, et la tension ludique est ici régie par les aller-retours constants entre le joueur, qui suggère, et le système, qui lui répond. C'est de cette interaction que naît l'apprentissage du système, et donc l'appréciation du jouet. En apprenant à comprendre son jouet, le joueur apprend à l'apprécier, et à mieux jouer avec



The image features a white background with decorative gear patterns in the four corners. The gears are in shades of grey and brown, with some having white circular cutouts. They are arranged in a way that suggests a mechanical or steampunk theme.

G.D.D

Game Design Document

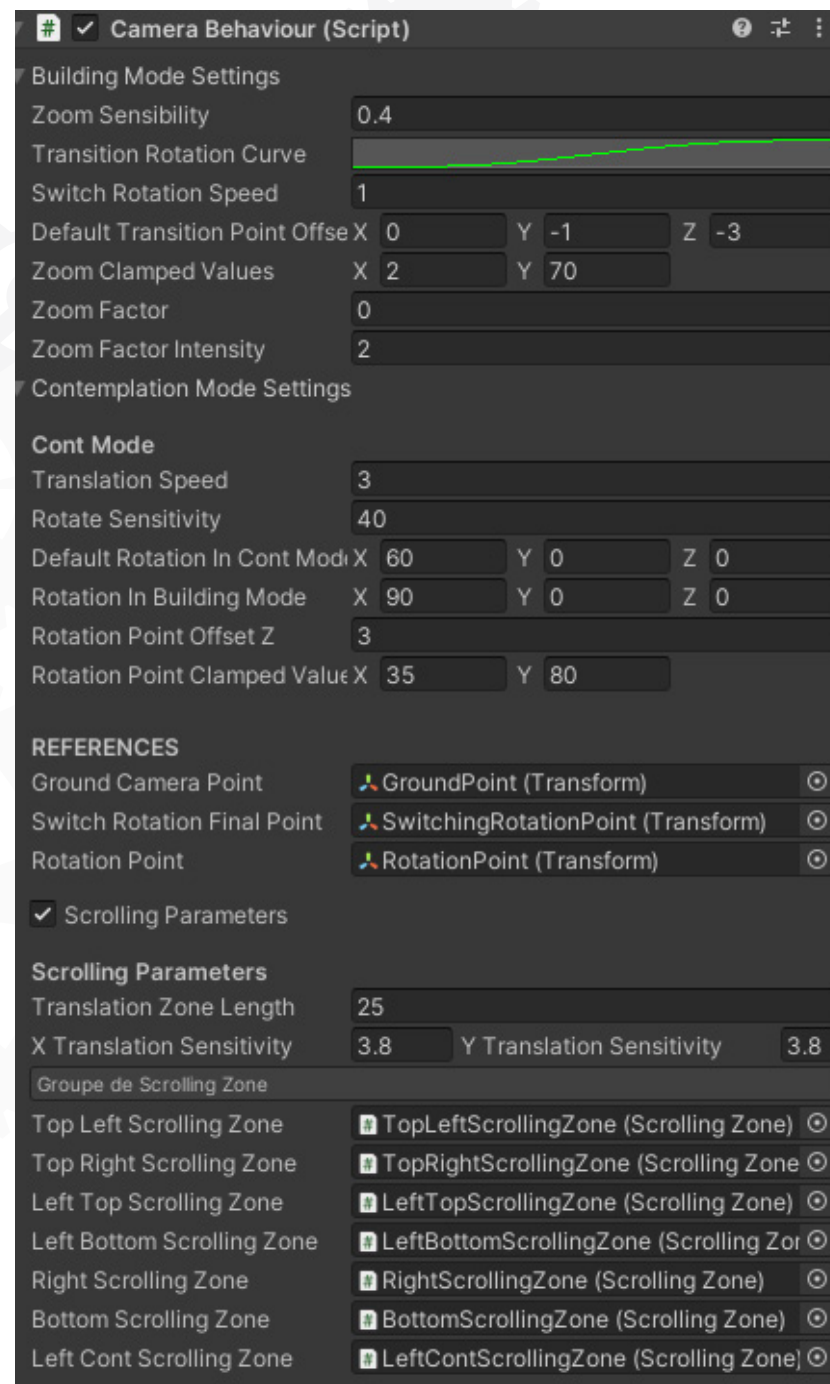
The image features a white background with decorative gear patterns in the four corners. The gears are rendered in a dark grey or black color, with some smaller gears in a reddish-brown hue. They are arranged in a way that suggests a complex mechanical system, with some gears meshing together.

G.D

Game Design

Caméra

La caméra possède 5 comportements. Chacun de ces comportements est composé de ses propres paramètres. Il existe 2 conditions dans laquelle certains comportements seront actif. Certaines fonctionnalités seront actives en mode édition puis d'autres en mode contemplation. Les paramètres sont donc séparés et disposés de sorte à être regroupés selon ses 2 conditions, et une partie est à part car un comportement qui est actif dans les deux cas mais qui subit des modifications selon le mode. Nous avons utilisé la classe GUI de Unity pour mieux organiser les variables dans l'inspector comme il est montré si contre. La dernière partie contient la référence vers toutes les zones qui seront utilisées pour le scrolling de la caméra, qui sera expliqué par la suite.

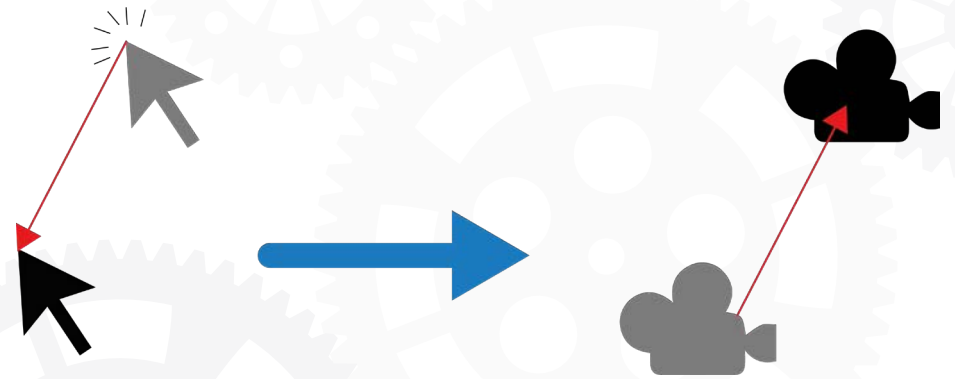


La caméra peut zoomer et dézoomer, le zoom a une limitation, contrairement au dézoom.

La caméra a plusieurs façons de se déplacer.

1er mode : Le drag

En maintenant le bouton de la molette de la souris, le curseur passe en mode « drag ». A partir de là, la caméra va se déplacer dans la direction opposée à celle de la souris, comme si on attrapait le sol pour pousser dans une direction. Il s'arrête quand le curseur atteint un bord de l'écran. (mettre le schéma du drag)



2eme mode : La rotation en mode Contemplation

En maintenant le clic droit, on peut tourner la caméra sur l'axe X en déplaçant le curseur vers le haut ou le bas. La caméra ne tourne pas directement sur son axe mais autour du point « RotationPoint ». Ce point suit la caméra en gardant un offset particulier, le « RotationPoint Offset » pour être toujours à une certaine distance de la caméra. Lorsque le bouton d'action est maintenu, la caméra devient un enfant du RotationPoint et c'est ce point ci qui va tourner sur son axe X. Etant un enfant de l'objet, la caméra va suivre sa rotation mais à un rayon différent. Une fois le bouton relâché, la caméra n'est plus un enfant du point et ce dernier retient la dernière valeur de l'angle sur son axe. (mettre le schema du rotateDrag)

Transition au changement de mode :

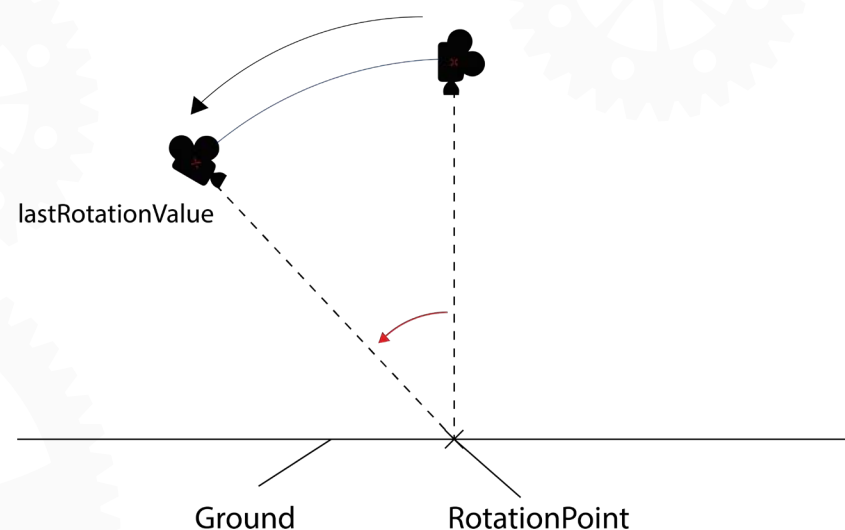
Lors du changement de mode, le RotationPoint effectue une rotation et la caméra suit sa rotation.

Edition -> Contemplation : le RotationPoint effectue une rotation entre sa valeur dans le premier mode et la dernière rotation enregistré dans le mode Contemplation, pour la première transition, une valeur de l'angle de base est définie.

Contemplation -> Edition : le RotationPoint effectue une rotation entre sa rotation actuelle et une valeur définie pour le mode édition, parfaitement perpendiculaire au sol en vue de dessus. (soit 0,-90,0)

La transition entre les deux modes est effectuée grâce à une vitesse qui sera multipliée par un paramètre de type Curve pour une meilleure manipulation de la vélocité du mouvement étape par étape.

Schéma de la rotation de la caméra pendant la transition de mode

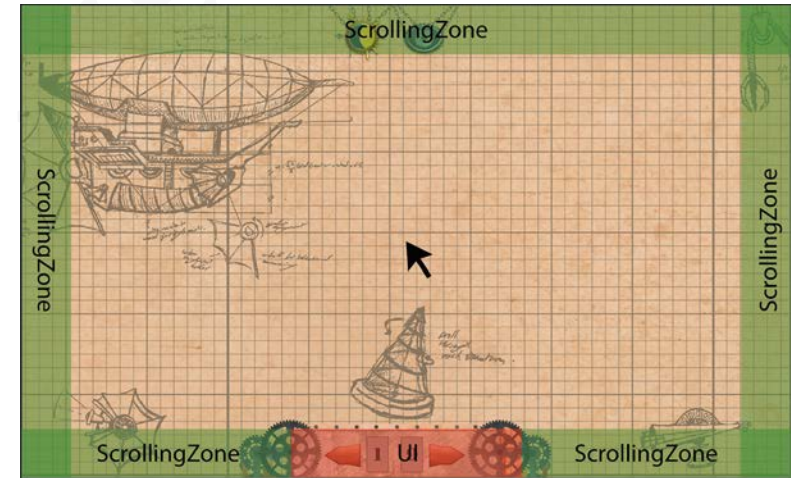


3eme mode : Le scrolling

En dirigeant le curseur sur les bords de l'écran, la caméra va effectuer une translation dans la direction vers lequel se situe le bord. Pour ce fait, il y a différentes zones pour chaque côté de l'écran. Une fois le curseur entré à l'intérieur, la caméra va se déplacer de façon continue. Plus le curseur va se situer au bout de la zone (donc au bord de l'écran), plus la caméra va se déplacer vite. Les zones sont adaptées en mode Edition pour ne pas gêner l'expérience du joueur.

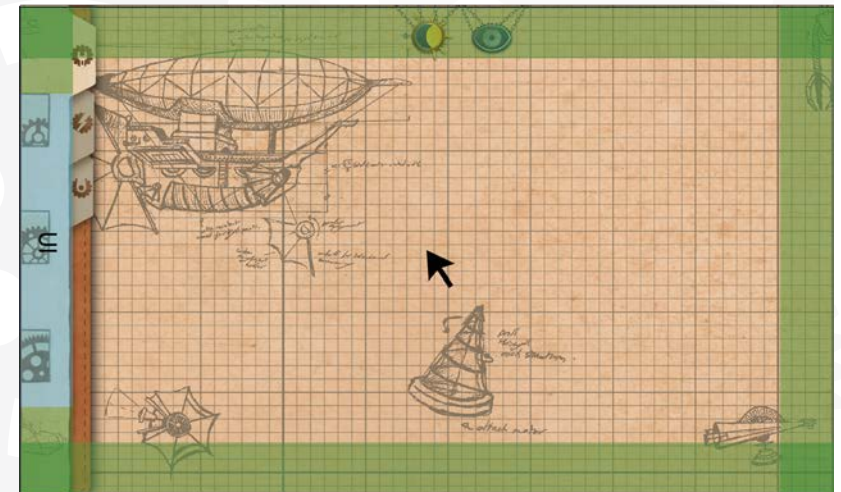
Comportement d'une scrolling zone: quand le curseur passe dans la zone, celle ci va récupérer sa position locale par rapport au point d'origine de la zone. La largeur des scrolling zone est défini par le paramètre «translationZoneLength» qui va redimensionner la largeur ou la hauteur de l'objet selon le type du scrolling zone (XScrolling, YScrolling). L'intensité de scrolling est obtenu avec l'axe x ou y (dépendant du type) de la position locale du curseur dans la zone. Cette intensité est récupéré par la caméra et est multiplié par deux multiplicateurs: XTranslationIntensity pour l'intensité récupéré par les types XScrolling et YTranslationIntensity pour les types YScrolling. On procède ensuite à la translation de la caméra en lui indiquant une vitesse de déplacement qui est le résultat des deux opérations effectuées.

Schéma Scrolling mode contemplation



Screen

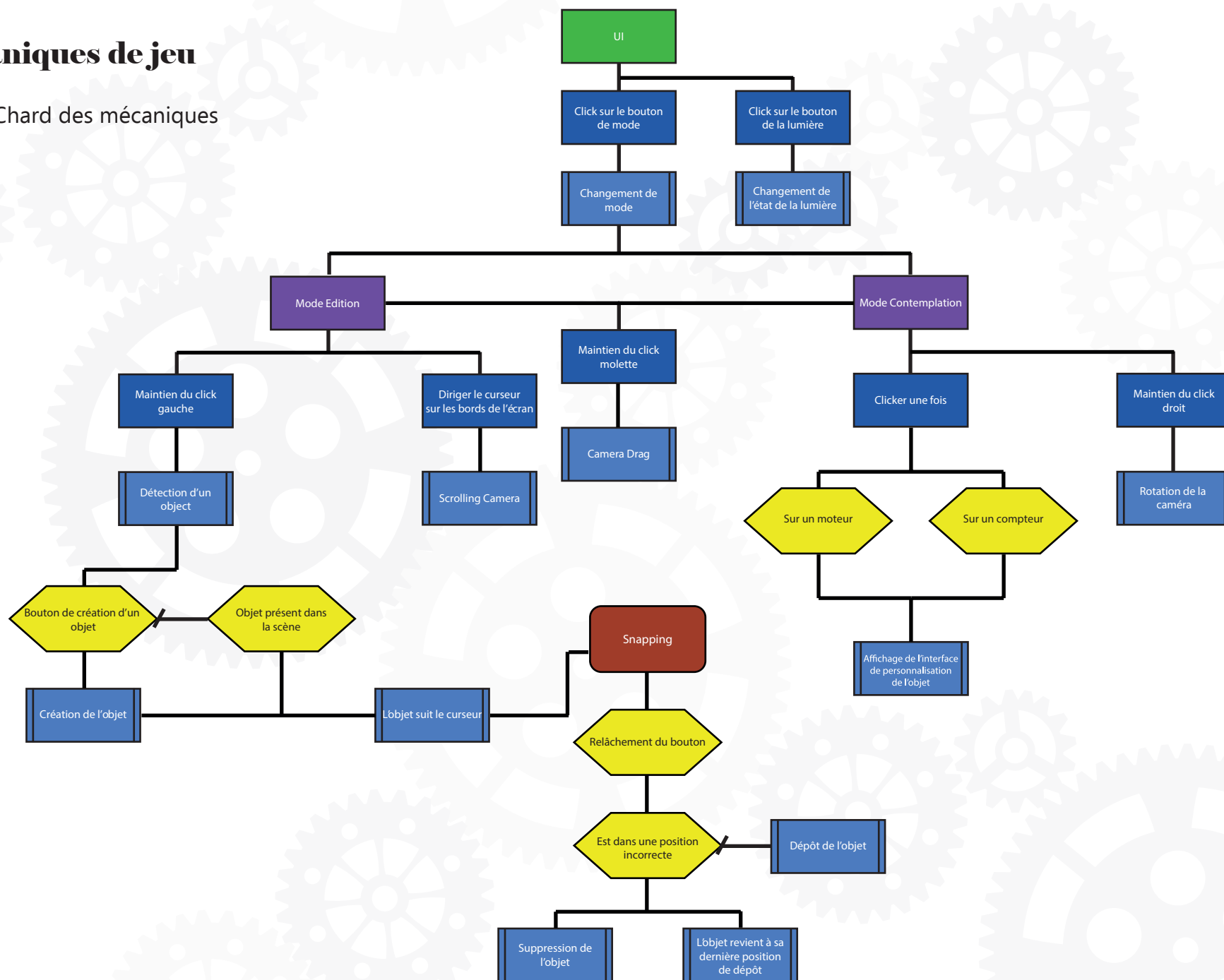
Schéma Scrolling mode édition



Screen

Mécaniques de jeu

FlowChart des mécaniques



Gears tourne autour d'un nombre relativement faible de mécaniques, qui s'emboîtent les unes dans les autres. La première grande famille de mécaniques, et la principale, régit comment l'utilisateur va interagir avec le système. On y trouve donc des mécaniques comme la création d'objets, la prise en main de ceux-ci, ou encore leur déplacement dans l'espace de jeu, qui mène à poser l'objet en question à un nouvel endroit. Ces mécaniques s'appliquent de manière générale et universelle à tous les objets présents dans l'espace de jeu, on commencera par s'y intéresser avant de s'atteler aux autres mécaniques.

La deuxième famille de mécaniques concerne le fonctionnement intrinsèque de ces objets. Rotation, transmission de la rotation, incrémentation de valeurs... Chaque objet introduit sa propre mécanique, et son propre comportement, on s'y intéressera au cas par cas.

I/Créer un objet

Il s'agit ici de la première mécanique à laquelle est confronté le joueur: face à un espace de jeu vide, la première chose à faire est de créer un objet.

Les différents objets posables et avec lesquels le joueur peut interagir sont listés sur la partie gauche de l'écran, et triés par catégorie. On nomme cet espace de jeu l'inventaire. Il a pour unique but de rendre visible les différentes options qui s'offrent au joueur, et lui donner la possibilité de les créer.

Chaque icône correspond donc à un objet, et agit comme un bouton: Enfoncer le click gauche avec le curseur sur l'un d'entre eux fait apparaître dans l'espace de jeu l'objet correspondant, qui remplace le curseur. L'objet existe désormais dans l'espace de jeu, et tant que le joueur maintient le click gauche de la souris, il peut le déplacer: on bascule ici sur la mécanique de déplacement des objets.

Cette mécanique étant la première à laquelle s'attelle le joueur, on fait attention à ce qu'elle reste la plus simple, et la plus accessible possible. On réduit donc au maximum les règles qui s'y appliquent.

Tout particulièrement, elle ne s'applique que lorsque le joueur appuie sur un des boutons dans l'inventaire. En outre, le joueur ne doit pas être en train d'interagir avec d'autres objets en même temps. Il est ainsi impossible de créer un nouvel objet pendant qu'on en déplace un, et il est impossible d'en créer plusieurs au même moment.

II/Prendre un objet

La mécanique Créer un objet issue directement à la mécanique Déplacer un objet, que l'on détaillera dans la prochaine partie. On le voit ainsi, l'objet nouvellement créé est prit par le joueur, qui peut désormais le déplacer. Ce n'est cependant pas la seule mécanique qui a cette conséquence.

Prendre un objet, permet comme son nom l'indique d'interagir directement avec un objet sur le terrain, avec pour but de le déplacer dans l'espace de jeu, ou de le supprimer. Son fonctionnement se veut le plus simple possible, et est similaire à celui de la Création d'objet.

Le joueur peut déplacer tout objet présent dans l'espace de jeu: Enfoncer le Click Gauche pendant que le curseur survole un objet a pour effet de l'aggriper. L'objet en question se détache alors immédiatement de ceux avec lesquels il était lié, et remplace le curseur. Le joueur est désormais libre de le déplacer, via la mécanique de Déplacement d'objets.

De la même façon, cette mécanique se veut le plus accessible possible, et répond à relativement peu de règles. Tout objet peut être pris, mais seul un objet l'être simultanément. Il n'est donc pas possible de Prendre un objet, tant qu'un autre objet est en train d'être créé, ou déplacé. Évidemment, cette mécanique n'est disponible qu'à partir du moment où il y a au moins un objet sur le plan.

III/Poser un objet

Déplacer un objet prend effet tant que le joueur maintient le click gauche, et s'arrête immédiatement lorsqu'il le relâche. L'objet se fige dans l'espace et le curseur réapparaît.

On distingue alors deux issues possibles, dépendant de la position dans l'espace où l'objet se situe, et du type d'objet que l'on est en train de manipuler. En effet, cette mécanique, à l'instar des précédentes, est soumise à des règles précises, qui assurent le bon fonctionnement du jouet dans son ensemble. Chaque objet a ses propres spécificités, mais il existe des règles universelles.

Au moment de le poser, tout objet se trouve dans l'un des cas suivants. Il est soit dans une position légale, soit dans une position illégale.

IV/Déplacer un objet

Cette mécanique répond directement aux deux précédentes: elle n'est disponible qu'à l'issue de l'une ou de l'autre. Ainsi, il existe deux façons de Déplacer un objet, la première est d'en Créer un, et la deuxième est d'en Prendre un dans l'espace de jeu. Ces deux actions résultent du maintien du Click Gauche sur un élément de jeu, et remplacent le curseur par l'objet concerné.

Déplacer un objet correspond à modifier ses coordonnées dans l'espace de jeu. L'objet en question agit comme le nouveau curseur de la souris: déplacer la souris horizontalement ou latéralement déplacera l'objet en conséquence. Cette mécanique s'applique tant que le joueur maintient le Click Gauche, etw prend fin immédiatement lorsque celui-ci est relâché. On bascule alors sur la mécanique Poser un objet.

Encore une fois, cette mécanique obéit à peu de règles, et se veut le plus simple possible, à l'instar de Poser un objet. Il est impossible de Déplacer un objet sans l'avoir au préalable soit Créé, soit Pris. Cette mécanique en est une conséquence directe. De la même façon, il est impossible de Déplacer plusieurs objets en même temps, ni d'en créer ou d'en Prendre un nouveau tant que cette mécanique est active.

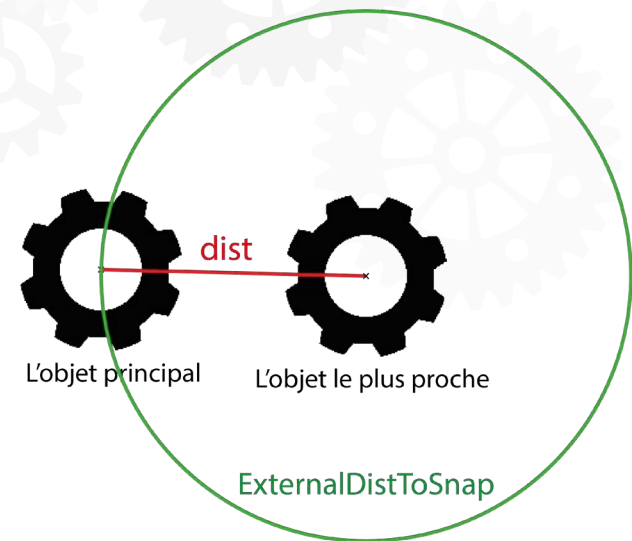
La mécanique Poser un objet est elle soumise à plus de règles et de conditions pour son bon fonctionnement. Déplacer un objet soumet déjà des feedbacks et des signes permettant au joueur d'anticiper au mieux la conséquence de Poser l'objet à l'endroit où il est actuellement en train de le placer. En particulier, chaque objet est muni d'un indicateur coloré, qui s'actualise dépendamment de son état. Déplacer un objet actualise en continu cet indicateur pour l'objet déplacé, ainsi que pour tous les objets avec lesquels il entre en contact. On entrera plus en détail sur ce système dans la partie Poser un objet, et dépend globalement de l'objet qu'on est en train de manipuler.

Snapping :

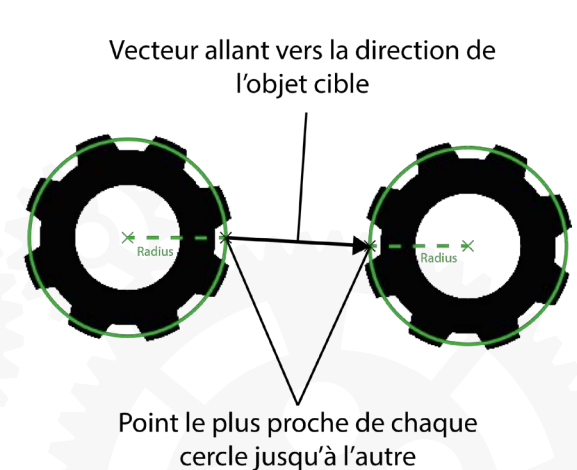
Dans notre système, il est essentiel d'assurer une bonne gestion du placement des engrenages, ils doivent être bien placés entre eux pour que la rotation se fasse. C'est pourquoi nous avons décidé d'implémenter un système d'accrochage automatique lorsque l'engrenage que nous contrôlons se rapproche d'un autre.

Condition pour que le snapping se lance : avoir une distance entre les deux objets inférieure à une valeur donnée.

Schéma du rayon de détection pour le snapping



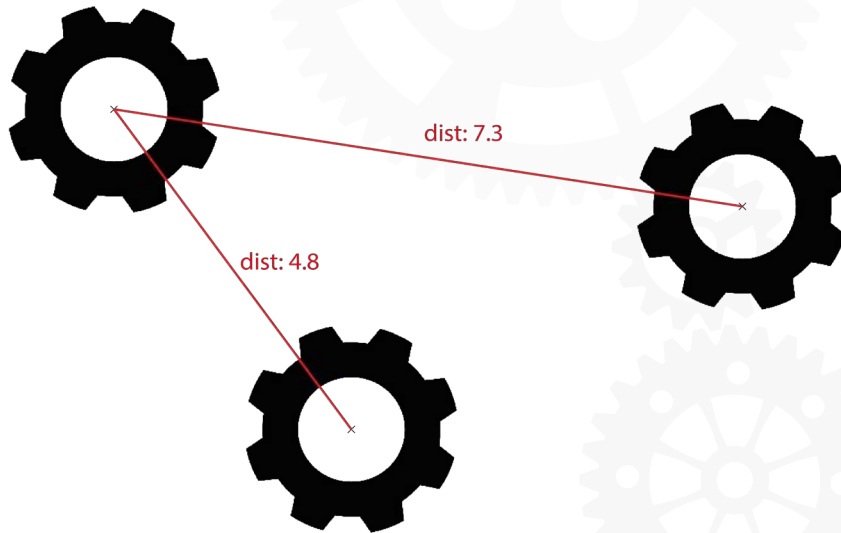
Si $dist < ExternalDistToSnap$



Cette vérification de la distance est appliqué uniquement à l'objet le plus proche de celui que nous contrôlons. Pour ce fait on mesure la distance entre l'objet tenu et tous les autres objets posés sur la scène et on ressort le plus proche.

Schéma teste de la distance entre les différents rouages

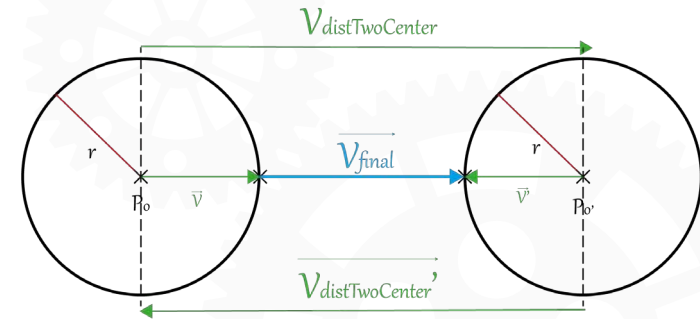
Objet que l'on déplace



Deviens la cible pour le snapping

Une fois la cible récupérée, nous pouvons procéder au système d'accrochage.

On incrémente en continue la valeur du vecteur partant du point dans le cercle le plus proche de l'autre objet jusqu'au point du l'autre cercle le plus proche de notre objet, à la position de l'objet. Ce qui fait qu'il effectue une translation vers l'engrenage égale à celle effectué grâce au curseur à l'opposer. Les deux forces égales s'oppose pour finalement le maintenir autour de l'engrenage.



$$P_0 = (x_0, y_0)$$

$$P_{0'} = (x_{0'}, y_{0'})$$

$$\overrightarrow{V_{distTwoCenter}} = P_{0'} - P_0$$

$$\overrightarrow{V_{distTwoCenter}'} = P_0 - P_{0'}$$

$$f_{distTwoCenter} = \overrightarrow{V_{distTwoCenter}}.magnitude$$

$$f_{distTwoCenter}' = \overrightarrow{V_{distTwoCenter}'} . magnitude$$

$$\overline{V} = f_{distTwoCenter}.normalized * r_0$$

$$\overline{V}' = f_{distTwoCenter}' . normalized * r_{0'}$$

$$f_{finalDirLength} = -(f_{distTwoCenter} - \overline{V}.magnitude - \overline{V}'.magnitude)$$

$$\overline{V}_{final} = -(\overline{V}_{distTwoCenter} - \overline{V} - \overline{V}').normalized * f_{finalDirLength}$$

Drag and drop :

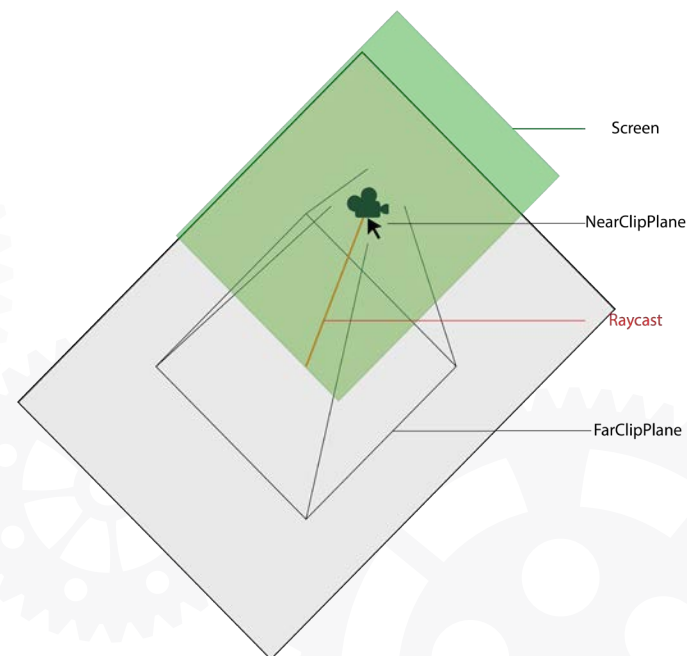
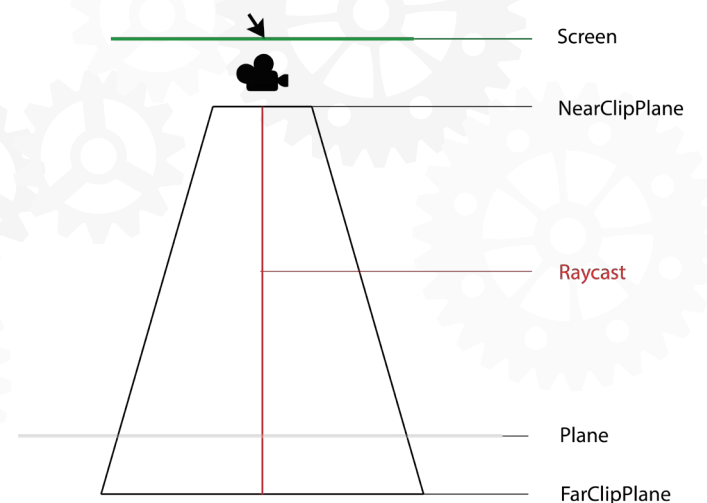
Pour le déplacement des objets, on maintient le bouton d'action de la souris sur l'objet puis on dirige le curseur pour le déplacer. Il faut d'abord pouvoir détecter le bon objet. Pour ce fait, on utilise un Raycast. Le Raycast est une ligne qui va pouvoir détecter un élément dans la scène. La vision de la caméra se fait de la manière suivante :

- NearClipPlane : il s'agit du point le plus proche de la caméra à partir duquel il peut afficher ;
- FarClipPlane : il s'agit du point le plus éloigné que la caméra peut afficher.

Création du Raycast : 2 points sont créés à partir duquel le Raycast va pouvoir se former. Le premier part de la zone du NearClipPlane pour la profondeur (axe Z) et comprend la position du curseur sur l'écran et le retranscrit en un point dans le monde 3D. Le deuxième point comprend la position du curseur dans le monde 3D et la position du FarClipPlane pour la profondeur.

Une fois l'objet détecté, celui-ci récupère la position du curseur dans le monde pour le déplacement mais sa hauteur est définie par le paramètre « upper » pour qu'il maintienne une certaine hauteur constante. Un « SmoothDamp », soit une transition entre deux positions, est appliqué au déplacement de l'objet, entre sa position actuelle et celle du curseur, pour un déplacement fluide lors du drag.

Schéma création du raycast vue de profile et isométrique



Les modes d'interactions

Finalement, et avant de s'attaquer aux mécaniques et variations propres aux objets, il reste un dernier élément auquel s'attaquer. Pour accentuer le plaisir du joueur pendant l'interaction avec le système, on décide de scinder son expérience en deux parties distinctes, qui contrastent l'une avec l'autre. On distingue dans l'utilisation du jouet deux comportements majeurs: construire le système, et l'observer dans son fonctionnement. Ces deux composantes s'alternent, et construisent à elles deux l'expérience de jeu. Pour les accentuer, on introduit donc un nouvel état, qui accentue tantôt l'un, tantôt l'autre.

Le mode contemplation correspond au désir du joueur d'observer le système dans son fonctionnement. Dans ce mode, on réduit au maximum l'interactivité du joueur avec le système: il ne peut plus ni créer d'objets, ni en prendre, ni en déplacer, ni en poser. De même, les différents feedbacks lumineux supposés aider le joueur à anticiper et placer au mieux ses objets devient obsolète. En contrepartie, on peut mettre l'emphasis sur ce qui va intéresser le joueur sur cette phase de jeu: le visuel, et le sonore, entre autres. En particulier, le fonctionnement des objets devient visible: ce qui doit tourner tourne, ce qui doit s'incrémenter s'incrémente visuellement, les signaux lumineux perdent leur sens en termes de design, et sont remplacés par des lumières plus agréables visuellement, qui réagissent avec le mouvement, et la vitesse de la rotation...



Le mode édition, à l'inverse, met l'emphasis sur le désir du joueur de construire, d'accumuler des objets, de les placer et de les déplacer dans l'espace de jeu. C'est le mode dans lequel toutes les mécaniques seront accessibles: créer, poser, prendre, déplacer... Les signaux visuels sont réduits, pour une meilleure clarté et une meilleure lisibilité. Les signaux lumineux des différents objets sont utilisés pour une autre fonction, une fonction de design: ils adaptent leur feedback en fonction des actions du joueur, pour l'aider à comprendre et anticiper les conséquences de ses actions sur le système. Les feedbacks sonores sont plus calmes, et plus propices à la concentration. Le tout s'adapte ici plus sur de la création que sur l'appréciation de ce qu'on crée.

Ici, séparer les deux tendances du joueur vont avoir un effet direct sur l'expérience de jeu du joueur: en coupant l'un, on pousse le joueur à se concentrer davantage sur l'autre, et vice-versa. Ça rend également la transition entre les deux modes importantes, et accentue la volonté du joueur: s'il change de mode, c'est parce qu'il change dans sa façon de vouloir apprécier le jouet.

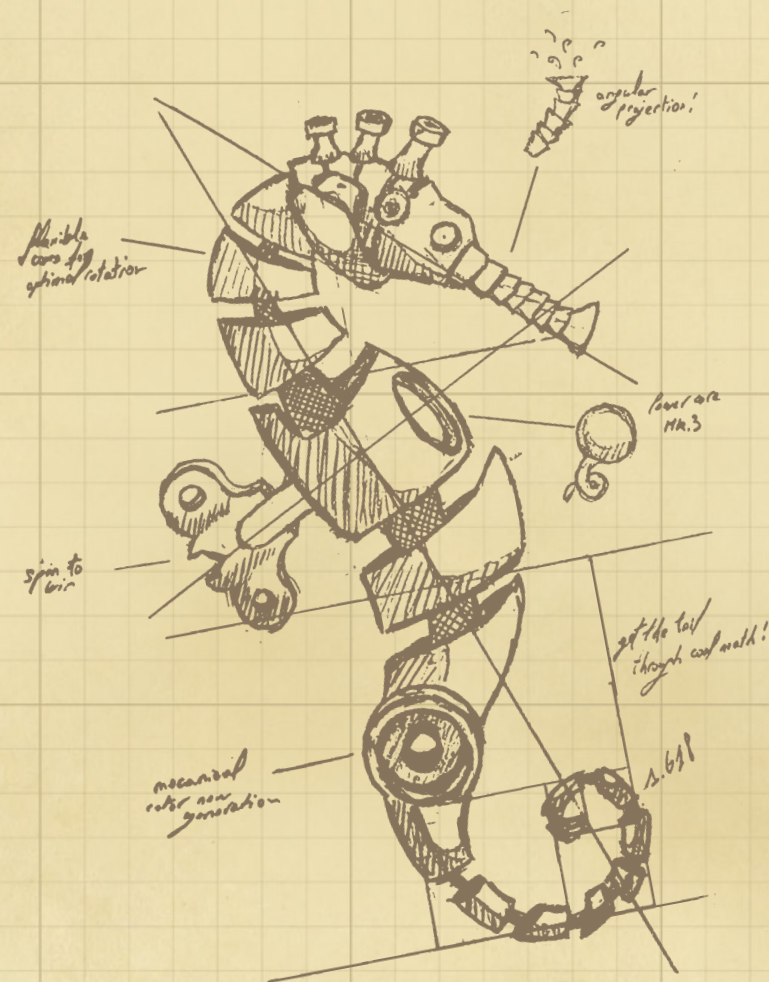
Le changement de mode est accessible à tout moment via un bouton, présent sur la partie supérieure de l'écran, et s'effectue lorsque le joueur appuie sur le click gauche avec son curseur dessus.



Ingrédient

Les principales mécaniques de jeu étant listées, notre joueur à accès à un panel d'outils lui permettant de façonner son environnement, et d'interagir avec celui-ci. Il nous reste cependant à détailler ce avec quoi il va jouer: les objets en question.

On classe les objets en 3 grandes familles: les moteurs, qui créent le mouvement et initient la chaîne; les rouages, éléments séparés de la chaîne, qui transmettent la rotation de la manière la plus essentielle. Finalement les connecteurs transmettent la rotation d'une manière plus situationnelle, contextuelle, en apportant des options pour le contrôler.



I/Les Rouages

Les rouages constituent le cœur de ce jouet, les briques de légo essentielles que le joueur pourra agencer pour créer des systèmes aussi complexes qu'il le souhaite. Leur fonctionnement se veut le plus simple possible: c'est le premier point d'accès du joueur au jeu, et ils doivent représenter les deux aspects majeurs du jeu: de l'accumulation avec une complexité croissante, autant qu'une machine à signaux visuels et sonores contemplatifs.

Créer un rouage est très simple, il suffit d'utiliser le click gauche, en ayant le curseur placé sur l'un des trois boutons correspondants: le rouage est alors généré et remplace le curseur, prêt à être déplacé.

Les rouages peuvent être placés n'importe où sur l'espace de jeu, à quelques conditions près. Si placé dans le vide, il restera simplement inactif, tant qu'on ne le connecte à aucune chaîne. Si placé contre un ou plusieurs autres engrenages en revanche, il tentera de se lier.

Comme vu précédemment, Poser un objet se résout principalement autour d'une condition: est-ce que la position de l'objet en question est légale ou non?

On considère tout engrenage lié directement ou indirectement à un moteur comme engrenage actif: au fil de la transmission, il se mettra à son tour à tourner. Ce fonctionnement met en valeur une condition illégale particulière: un engrenage ne peut pas être lié simultanément à deux engrenages actifs, que ceux-ci fassent parti d'un même système ou de systèmes différents. La raison est simple: si un engrenage venait à recevoir du mouvement de deux sources différentes, et que celles-ci n'étaient pas exactement synchronisées, le système viendrait à jammer.

Le système de Gears n'acceptant pas les situations dans lesquelles le circuit ne fonctionnerait pas, placer un engrenage dans une position illégale mène à la suppression de celui-ci. Si l'engre-

nage en question vient d'être créé, il disparaît simplement. Si en revanche, il vient d'être pris sur le terrain, on le replace simplement dans la dernière position valide connue.

Finalement, un engrenage peut se connecter à autant de rouages inactifs que voulu, il s'y liera simplement, et leur transmettra de la rotation s'il en reçoit.

La majeure fonctionnalité liée aux engrenages est la rotation. Chaque engrenage en recevant le signal tourne sur lui-même, à une vitesse constante dépendant de la nature du signal qu'il reçoit, puis signale aux engrenages voisins de faire de même.

On distingue trois tailles étalons d'engrenages. Dans un système rotatif, la taille de chaque engrenage change la vitesse à laquelle il tourne: lorsqu'il est associé à un même mouvement, un engrenage plus petit tournera plus vite, quand un plus grand tournera moins vite, le tout étant relativement à leur taille. On définit la taille de chaque engrenage pour que la variation de vitesse associée soit de nature exponentielle: le plus petit tourne 2 fois plus vite que le moyen, qui à son tour tourne 2 fois plus vite que le plus grand.

Le joueur est donc ici capable de créer un système varié, bien qu'il manque de contrôle sur la vitesse de chaque objet, il est limité entre trois vitesses fondamentales.

Principe d'héritage

Puisque l'on veut créer des chemins infinis dont leur énergie mécanique est transmise grâce à une seule source, nous ne voulons pas inclure une physique de friction. Importer une telle physique nous bloquera très vite pour la réalisation de nos intentions, la force mécanique serait affaiblie à chaque ajout de rouage à cause de cette friction.

Pour ce fait, nous avons décidé de créer notre propre physique, la force émise par les rouages serait exactement la même que celle transmise par leur parent respectif.

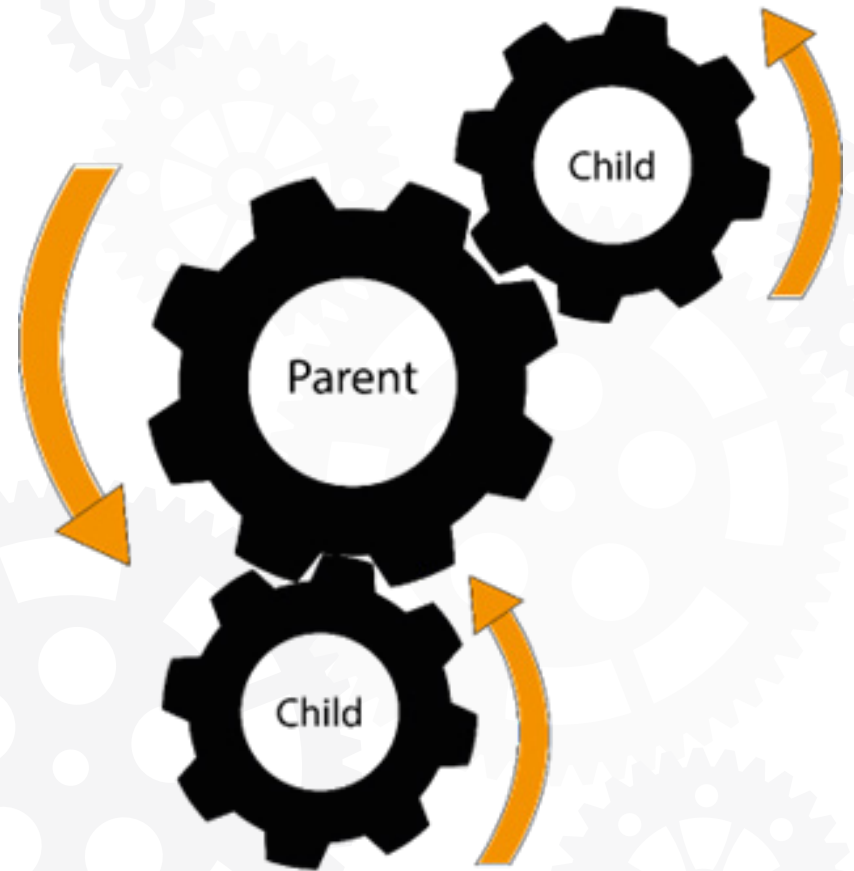
Voici un schéma pour se représenter le sens de rotation des rouages une fois attachés.

Schéma de la représentation du sens de rotation des engrenages

Le principe est que chaque engrenage hériterait du sens de rotation de son engrenage « parent » mais inversé ainsi que sa vitesse de rotation multipliée par un facteur dépendant de la taille de l'engrenage.

Nous voulons via Gears donner une expérience contemplative aux joueurs. Il sera amené à construire son propre mécanisme et le voir se mettre en marche

Gears est inspiré des univers Steam Punk et des ces mécaniques avancées



Pour l'héritage, nous voulons que toute l'énergie rotative parte d'un rouage : la racine (Ra).

Cette racine sera composée uniquement d'enfant et n'aura donc pas d'objet parent. Pour la structure, il faut s'imaginer un arbre. Elle contient la racine, des branches puis des feuilles au bout, c'est la structure que nous avons construit pour nos engrenages.

Il y a 5 états pour nos engrenages :

I. La racine (root)

La racine sera l'origine de toute la structure de notre chaîne. C'est à partir d'elle que tout va tourner.

II. La branche (branch)

La branche est constituée d'un parent et forcément d'un ou plusieurs enfants (child), tout comme une branche d'arbre. Elle est évidemment liée à la Racine.

III. La feuille (leaf)

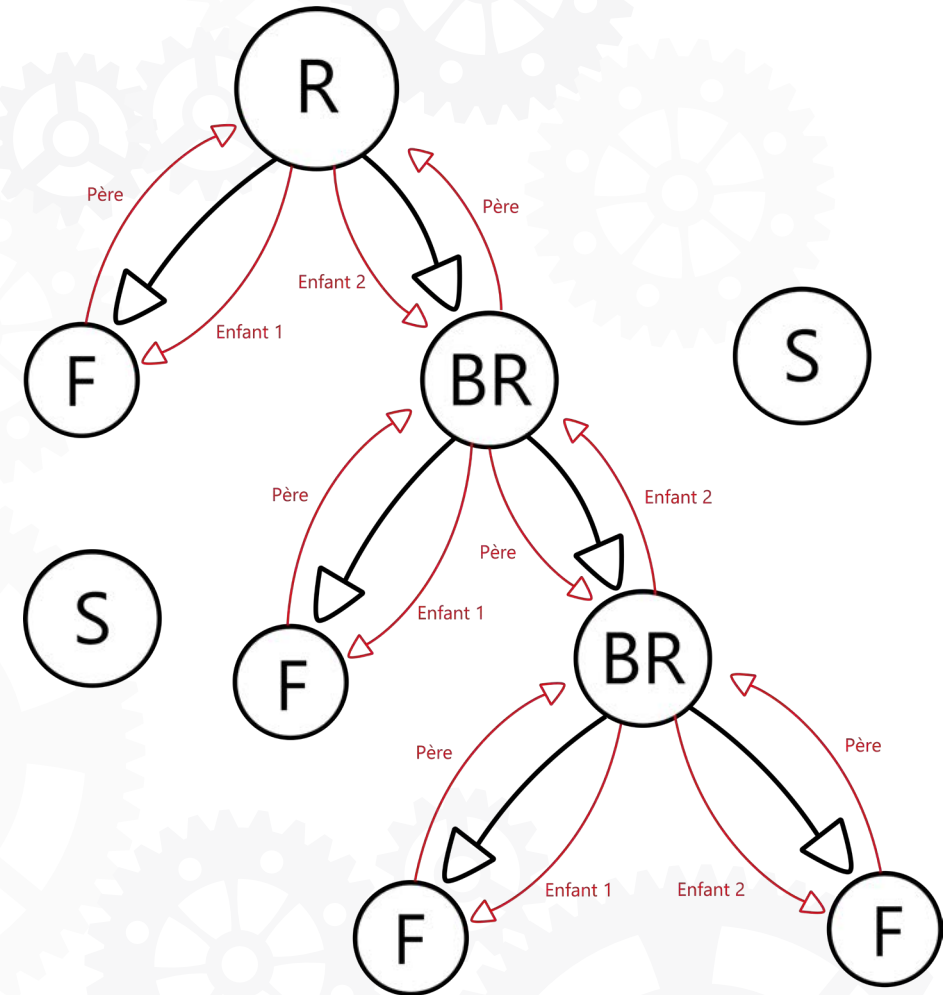
La feuille est un objet en fin de chaîne, elle a un parent mais n'a pas d'enfant.

IV. Le sauvage (Feral)

Il s'agit d'un engrenage n'étant pas lié à une racine, elle n'hérite donc pas de l'énergie rotative de cette dernière.

V. La graine (seed)

Il s'agit de l'état de l'objet avant qu'il soit posé, tout engrenage naît « graine » avant d'être posé, sauf la racine, soit le moteur.



Système de tick

Dans notre système, il est essentiel de savoir bien gérer les ticks de chaque engrenage, pour un premier temps les effets sonores des engrenages en action, mais aussi pour le compteur qui doit décrémenter sa valeur à chaque « contact » d'une dent de l'engrenage en entrée.

Le principe est de récupérer le temps T d'une dent à l'autre pendant la rotation de l'objet. Pour ce fait, on utilise la formule de la vitesse angulaire qui prend en paramètre la variation de l'angle entre 2 instants T . (image de la formule de la vitesse angulaire).

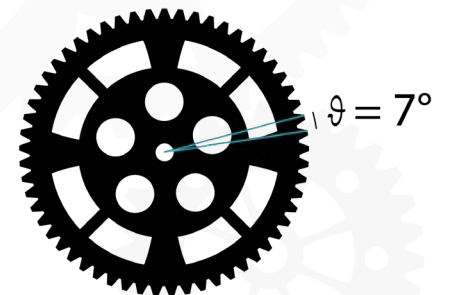
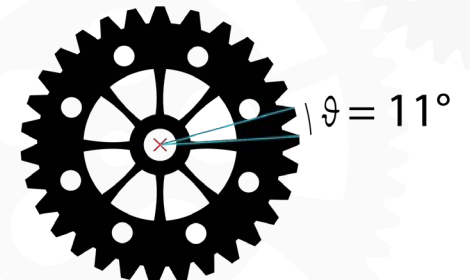
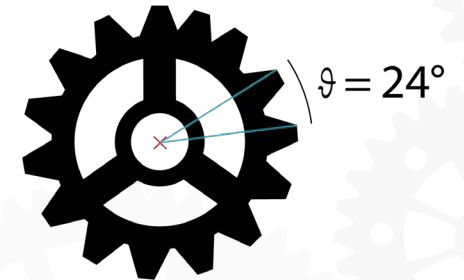
Nous voulons récupérer la valeur T et nous avons déjà la vitesse angulaire qui est défini par le paramètre « rotationSpeed » dans le GearBehaviour.

Vitesse angulaire: $\omega = \vartheta / t$

$\omega = \text{rotationSpeed}$

$\vartheta = \text{angle entre une dent}$

$$\text{tick} = \frac{\text{rotationSpeed}}{\vartheta}$$



II/Le Moteur

Le moteur constitue l'origine du mouvement, et est le point de départ de toute chaîne de rotation qui se respecte. Sans moteur, rien ne tourne!

Créer un moteur est aussi simple que n'importe quel autre objet: click gauche en plaçant le curseur sur le bouton associé fait apparaître un moteur à la place du curseur, prêt à être déplacé et posé.

Placer un moteur répond aux mêmes contraintes qu'un rouage, à la différence prêt que si un rouage peut recevoir et transmettre la rotation, le moteur ne peut que le transmettre.

Pour cette raison, le moteur ne peut être lié à aucun engrenage actif, ni à aucun autre moteur: la double source de rotation ferait jammer le système instantanément!

De la même manière, si un moteur venait à être mal placé, le système reviendrait automatiquement à son dernier état fonctionnel, soit en supprimant le moteur nouvellement créé, soit en le remplaçant à sa dernière position.

Pour ce qui est de la transmission de l'énergie, le moteur peut être placé au contact de n'importe quel nombre d'engrenage inactif: il s'y liera et les transformera en engrenages actifs. De plus, tous les engrenages connectés à ceux-ci, et les suivants à leur tour seront considérés comme actifs: une nouvelle chaîne se crée.

Le moteur génère une rotation fixe, en tournant sur lui-même, puis en la transmettant aux engrenages voisins. Sa particularité est que la vitesse est modulable: le joueur peut la modifier.

En mode contemplation, sélectionner un moteur fait apparaître un slider, qui représente visuellement la vitesse de rotation du moteur. Placer le curseur sur sa partie gauche fait tourner le mo-

teur dans le sens anti-horaire, et à l'inverse, placer le curseur sur sa partie droite fait tourner le moteur dans le sens horaire. De plus, plus le curseur est éloigné du centre, et plus la vitesse de rotation du moteur s'amplifie. Chaque rouage adaptant sa propre vitesse de rotation par rapport à celle de celui qui la transmet, toute la chaîne correspondante s'adapte de ce fait à celle du moteur en temps réel.

Il est également possible d'attribuer au moteur une rotation nulle, l'immobilisant, ainsi que toute la chaîne correspondante.

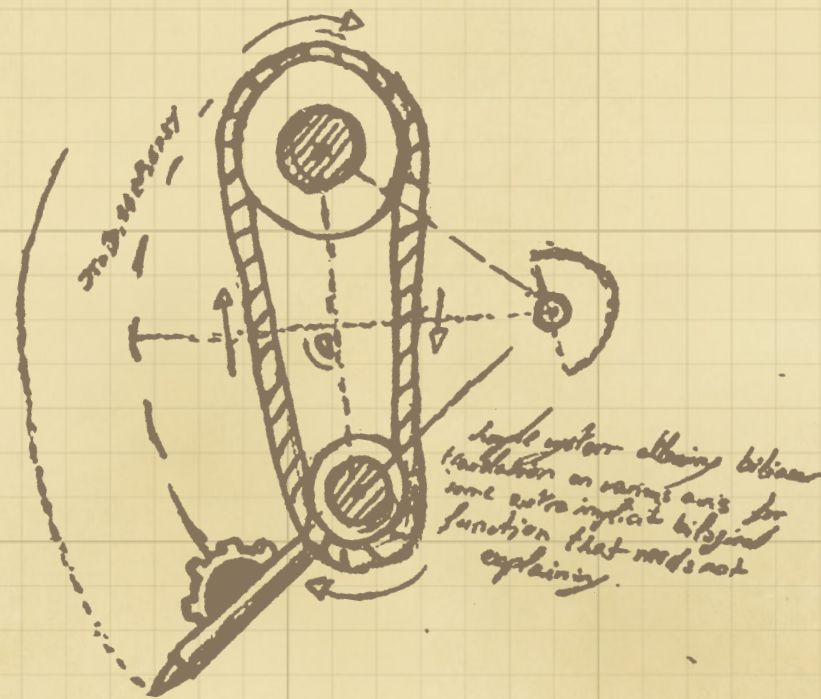
Les Connecteurs

Avec 3 calibres de rouages différents, et un moteur dont on peut varier la vitesse, le joueur à accès à quelques outils pour créer un système de taille croissante, et avec une certaine complexité. Il manque cependant pas mal de contrôle sur la vitesse de la rotation.

Le but des connecteurs est d'apporter de la variation et de la complexité au système. Chacun des objets de ce type apporte son propre twist! Il ne s'agit ici pas tant d'implémenter de nouvelles briques, que de donner les clés en main au joueur pour les faire varier. Vous pouvez voir cela comme ajouter de la couleur.

On en distingue pour l'instant deux majeurs: le premier, le Pont, permet de connecter deux rouages à distance en leur centre, permettant indirectement de contrôler et démultiplier la vitesse de rotation à travers le circuit.

Le Compteur, lui, contrôle directement la cadence du circuit! En comptant jusqu'à une valeur donnée, et en ne faisant tourner la sortie que lorsqu'elle est atteinte, il permet de saccader complètement le système, pour lui apporter variété et subtilité.



I/Le Pont

Le pont est le premier type de connecteur disponible au joueur. Il permet de connecter deux engrenages à distance et par leur centre. Ça a deux conséquences directes: d'une part, cela ajoute une nouvelle notion de spatialisation au joueur: il peut lier deux rouages peu importe leur distance l'un à l'autre, créant des systèmes dans l'espace avec une bien plus grande liberté. De plus, en faisant tourner les rouages par leur centre, la vitesse de rotation des rouages se multiplie lorsque leur taille varie, ajoutant un outil de contrôle de vitesse parmi l'éventail d'action disponible au joueur.

Comme toujours, il suffit d'un click gauche sur le bouton correspondant pour créer un pont. Son fonctionnement est cependant différent. Faisant le pont entre deux engrenages, il sera placé en deux temps. Lorsque le joueur pose le pont sur un engrenage, une nouvelle instance de pont se crée immédiatement et remplace à nouveau le curseur. Le joueur doit donc poser successivement 2 objets pour créer un lien.

De plus, le snapping s'effectue ici selon le centre de l'objet: on pose bien le pont sur un rouage, et non pas contre celui-ci.

Comme les autres objets, poser un pont répond à des règles précises. Tout d'abord, un même rouage ne peut héberger qu'un pont, afin d'éviter la démultiplication de sources de rotation, qui ferait fatalement jammer le système.

Pour les mêmes raisons, un pont ne peut pas connecter deux rouages actifs entre eux.

Enfin, si l'une des deux parties du pont est mal placée, les deux sont supprimées, remettant le système au dernier état fonctionnel: avant la création du pont.

La rotation se transmet entre les deux engrenages à travers le

pont qui les connecte: la vitesse de rotation de l'un est directement transférée à l'autre. Ici la taille des engrenages ne varie pas la vitesse de la rotation transmise: les rouages étant connectés par leur centre, la vitesse externe ne fait pas varier d'un engrenage à l'autre. Cela a une conséquence directe: si les rouages ont une taille différente, l'énergie du système se multiplie.

Ainsi, connecter via un pont un rouage sur un rouage actif plus grand AUGMENTE la vitesse globale de celui-ci, et des suivants connectés. Et à l'inverse, se connecter vers rouage plus petit la diminue. Avec plusieurs ponts bien placés, on peut donc multiplier ou diviser la vitesse de rotation comme on le souhaite sur tout le circuit, et donc créer des portions de circuits isolés les uns des autres, avec des vitesses de rotation différentes les uns des autres. On a par la même accès à une infinité de variation de vitesse, contraintes cependant par l'évolution exponentielle des vitesses.



II/Le Compteur

Le joueur a accès à l'aide du Pont à un outil de séparation de portion de circuits dans l'espace, ainsi qu'un outil de réglage de la vitesse constante des rouages.

Le Compteur a un rôle différent, bien qu'il apporte sa propre variante dans la construction du système: cet outil permet de saccader le mouvement, en définissant des timings sur lesquels le système tourne, ou ne tourne pas.

Le compteur se crée de la même façon que les rouages ou que le moteur. Suite à un click gauche sur le bouton correspondant, l'objet apparaît et remplace le curseur, prêt à être déplacé et posé. Il est constitué de deux rouages, et d'un bloc les reliant l'un à l'autre. Le rouage de gauche correspond à l'entrée, et celui de droite correspond à la sortie.

Le compteur est considéré comme bien placé si chacun de ces deux rouages est bien placé. En particulier, l'entrée ne peut être connectée qu'à un rouage actif simultanément, et la sortie ne doit être connectée à aucun rouage actif, mais peut l'être à autant de rouages inactifs qu'on le souhaite.

Sa façon de gérer la rotation est cependant assez particulière. Techniquement, les deux rouages qui le constituent sont considérés comme connectés l'un à l'autre, et donc les différents rouages qui le sont indirectement sont également considérés comme faisant partie de la même chaîne.

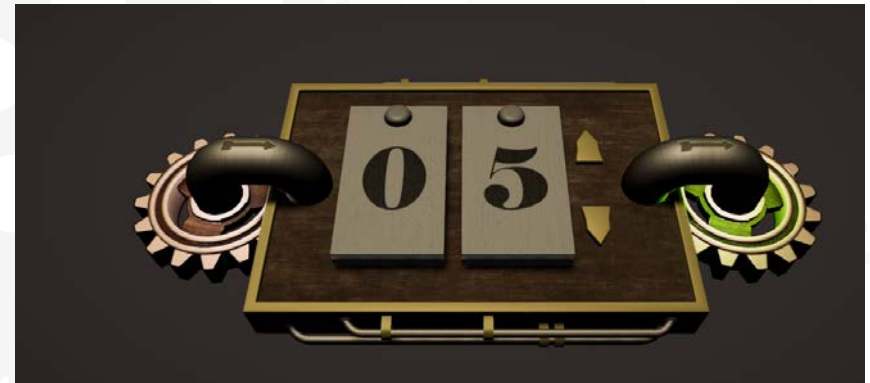
Le compteur agit ici comme une interface entre les deux rouages qui le constituent. Une valeur lui est attribuée (et modifiable par le joueur), et diminue sur la durée. Lorsqu'elle atteint zéro, et uniquement à ce moment-là, le rouage de sortie tourne, et fait tourner ses enfants récursivement pendant un court instant. Enfin, la valeur se repositionne à celle de départ, et recommence

à descendre.

On obtient ainsi un système qui alterne entre deux états: allumé ou éteint, le joueur ayant l'influence sur la durée pendant laquelle il reste éteint. Cette durée représente en fait un quotient de rotation fournie en entrée: plus le rouage d'entrée tourne vite, plus le compteur descend vite! Le joueur a donc le contrôle, via la vitesse du système, sur le rythme avec lequel le compteur alterne.

Pour modifier la valeur du compteur, le joueur peut se placer en mode contemplation, et le sélectionner via un click gauche: une interface apparaît en bas de l'écran, avec deux boutons sur lesquels le joueur peut appuyer pour augmenter ou diminuer la valeur maximale du compteur. Lorsqu'il se remet à zéro, il se positionnera à cette valeur.

Le joueur a donc maintenant accès à un outil de séquenciation et de saccadement du système, avec lequel il peut jouer pour créer des systèmes plus complexes, ou compter. Cet effet se complexifie encore davantage lorsqu'on cumule des compteurs, en les plaçant les uns après les autres, ou qu'on les croise avec une bonne utilisation des ponts!



The image features a white background with decorative gear patterns in the four corners. The gears are rendered in dark grey and a reddish-brown color, with some having white circular cutouts. They are arranged in a cluster, overlapping each other.

D.A.V

Direction Artistique Visuelle

Pitch

Dans notre projet de jouet Gears nous incarnons un inventeur dans son bureau. Via son dossier, il pourra créer des systèmes mécaniques composés de moteurs, d'engrenages et de systèmes de liaison.

Il pourra alterner entre le mode édition et contemplation via un bouton qui lui permet de faire des actions différentes.

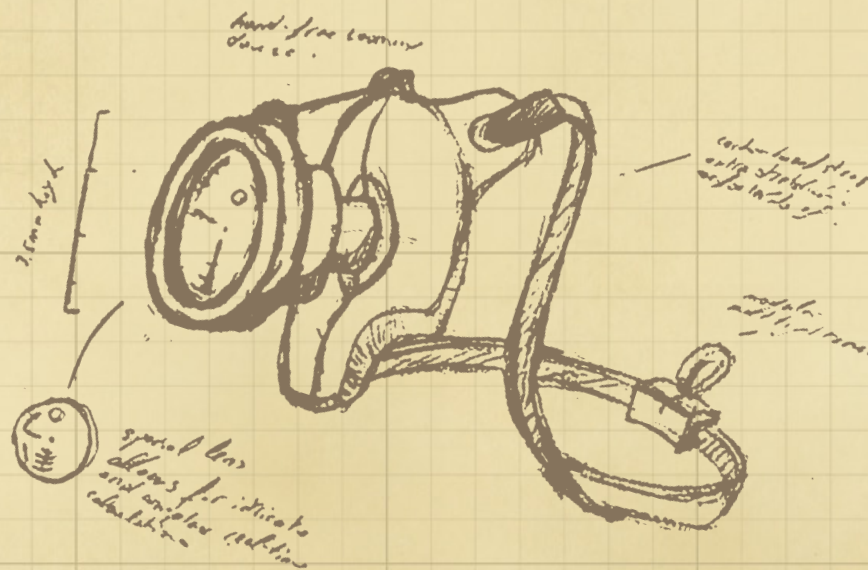
Le mode «Edition » est un mode créatif. La caméra est en vue top. Une interface à gauche de l'écran permet de faciliter la création de mécanismes pour le joueur lorsqu'il est dans ce mode.

Les LED, principaux feedbacks, sont sur les engrenages et transmettent des informations par rapport au bon placement de ceux-ci. La couleur des LED est de couleur vive ce qui rend les règles du jouet plus instinctives.

Le mode « Contemplation » est un mode vivant. Le mécanisme tourne, et les sons d'ambiance et les bruits d'engrenages se font alors entendre, renforçant cette atmosphère.

L'interface de création initialement présente disparaît. Enfin, la caméra peut s'incliner et se repositionner, afin de mieux observer notre système.

Le joueur pourra finalement jouer avec les lumières, via un bouton pour éteindre ou allumer globalement la scène.



Intention Générale

Les mécaniques de notre jouet ont permis de déterminer des intentions se dirigeant vers un univers Steam Punk. Ce courant littéraire se déroule dans un XIXe siècle dominé par la première révolution industrielle du charbon et de la vapeur. Cette révolution a permis de créer et imaginer des inventions jamais vues auparavant comme des dirigeables ou des trains. Nous nous basons donc dans cet univers, mais à une échelle micro via les engrenages qui composent ces machines.

Nous voulons ainsi une ambiance chaleureuse qui se détache des aspects froids de la mécanique pure.

Pour ce qui est de l'interface, elle sera minimaliste en mode Contemplation et plus chargée en mode Édition. Le dossier permettant de sélectionner et créer des objets sur la scène devra rester simple et réaliste afin de coller à un aspect inventeur qui dessine sur des feuilles. Les deux boutons pour le changement de lumière et de mode seront constamment visibles, il faut alors qu'il soit agréable et pas trop imposant dans l'interface.

Jules Vos, Environnement illustration, Arts station



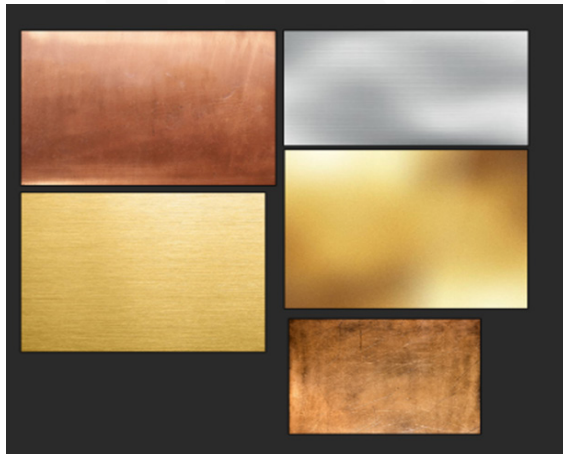
Illustration prise comme référence pour ces couleurs et ambiance de l'univers SteamPunk

MoodBoard et Concept Art Global

Dans premier temps nous avons travaillé un MoodBoard complet sur l'univers du Steam Punk en incluant des références graphiques provenant de films, de dessins ou de modélisations 3D.

Ces références nous ont permis de fixer nos objectifs en terme de visuels et d'ambiance.

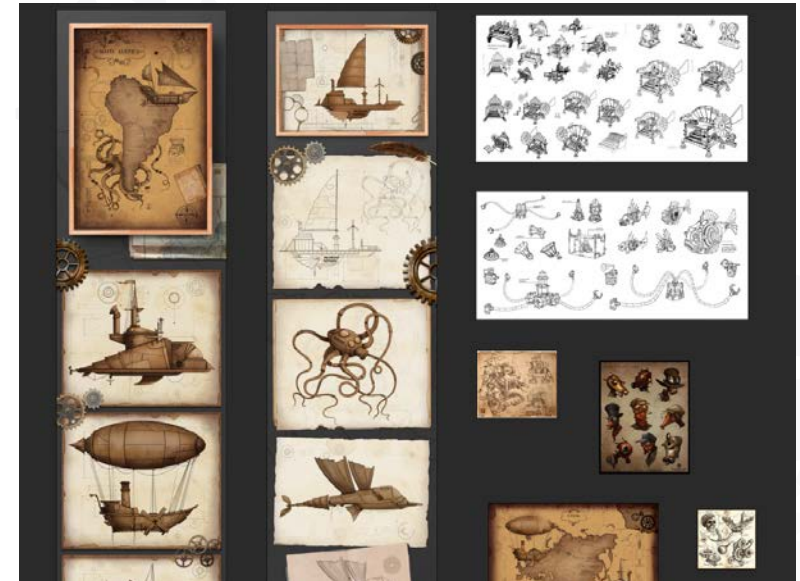
Recherche des matériaux utilisés



Recherches sur les visuels du centre et nombre de dents des engrenages



Recherche globale de l'univers steampunk



Concept Art N°1



Décisions de Design

I/Couleurs et matériaux

Nos intentions nous ont amené à faire une étude des différents matériaux et codes couleurs utilisés dans un univers Steam Punk.

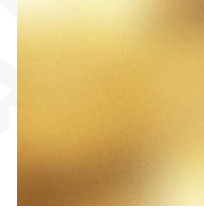
Dans cet univers on retrouve des créations mécaniques à base de Cuivre, de Bronze, d'Or, de Laiton, d'Acier et de bois. Ces matériaux étant en partie composés de couleurs chaudes, ils nous permettent de les utiliser facilement en grande quantité. Cependant l'acier n'est pas à bannir car il apporte du contraste via sa couleur froide.

Après plusieurs essais de textures sur des engrenages nous sommes partis sur un set de 4 matériaux métalliques (Acier, Cuivre, Bronze et Or) et un seul type de bois foncé.

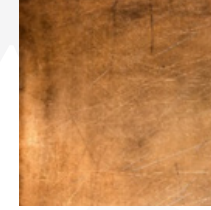
Acier



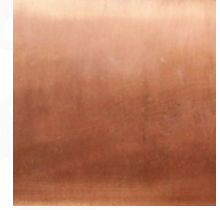
Or



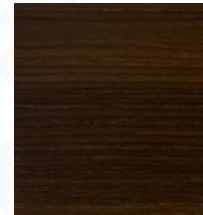
Bronze



Cuivre



Bois Foncé



Palette de couleur utilisée dans le projet



#F2E6D0

#BF9056

#D9BB96

#594432

#8B4F10



II/BackGround

Grâce au premier concept art, des questions sont venues par rapport à la 3D et les différentes hauteurs des objets dans un monde 3D. Les engrenages flottaient dans le vide, ce qui procurait un ressenti d'incompréhension des perspectives.

On s'est tournés alors vers une feuille en papier géante et infinie avec un quadrillage rappelant les tapis de découpe. La feuille garde un aspect ancien comme les feuilles de schéma positionnés dessus. Au fil des concepts, une idée d'un quadrillage plus complexe est apparue pour coller aux intentions Steam Punk et les couleurs ont été retravaillées.

Cependant les feuilles de schéma disposées sur le plan ne nous convenaient pas car elles donnent un effet de papier collé.

A terme, l'idée de créer nos propres dessins Steam Punk a émergé afin d'amplifier notre univers. Notre inventeur dessine alors ses idées sur son plan. On observe les inventions qu'il a pu imaginer ou créer .

Itération 1



Itération 2



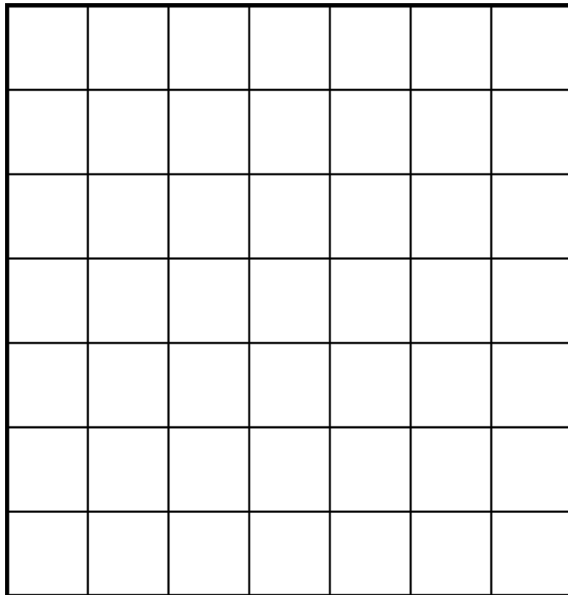
Dernière itération validée par la suite



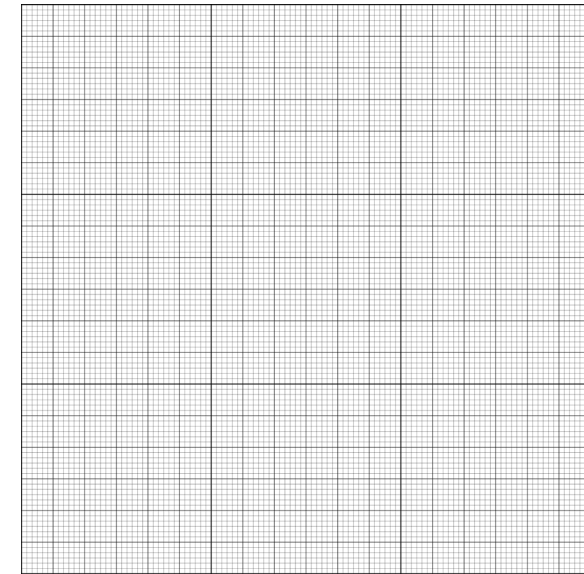
La background dans le jeu étant alors infini, sa texture se devait d'être seamless (se répéter à l'infini sans voir de coupure entre les répétitions de texture) . De plus, le joueur pouvant zoomer et dézoomer, le quadrillage sur celui-ci doit être composé de plusieurs quadrillages afin de toujours l'apercevoir malgré un dé-zoom conséquent.

Ainsi plusieurs tests ont été faits avec d'abord un quadrillage composé de 2 étages, cette itération ne permettait pas de voir parfaitement la grille. On a alors créé un quadrillage de 3 étages, permettant au joueur de toujours le voir malgré un éloignement accru.

1ère itération



Itération finale pour la grille



Background terminé avec la grille et la texture seamless



III/Interface

En suivant nos intentions générales, il existe deux modes de vision : Le mode édition et le mode contemplation.

L'interface se doit d'être minimaliste pour les boutons qui restent constamment sur l'écran (bouton pour la lumière et le bouton de changement de mode).

Bouton pour la lumière :

La représentation doit être évocatrice pour le joueur et être cohérente avec le dossier présent en mode Édition.

Pour rappeler notre univers, le bouton sera composé d'un engrenage. L'engrenage sera rempli d'un soleil et une lune.

Quand la lumière est allumée, le soleil prend plus de place que la lune. Il sera aussi plus lumineux que celle-ci, ce qui fait un retour pour le joueur dépendamment du mode dans lequel il se trouve. Une fois la lumière éteinte, le soleil se rétracte, la lune prend à son tour plus de place, sa luminosité est augmentée et celle du soleil est baissée.

Bouton pour le changement de mode :

Suivant les mêmes règles graphiques que le bouton de lumière, ce bouton représente un œil.

En mode contemplation l'œil est ouvert, il permet de façon instinctive au joueur de comprendre dans quel mode il se trouve. En suivant ce principe le mode Edition referme l'œil et un logo devient alors visible. Le logo représente un engrenage et une clé à molette qui représente parfaitement le mode Edition.

UI lumière éteinte



UI lumière allumée



UI mode édition



UI mode contemplation

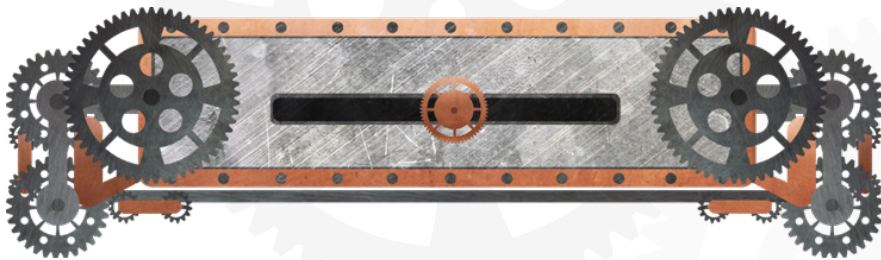


En mode contemplation, la seule interface présente permet de gérer des paramètres du moteur et du compteur. Comme c'est une interface qui modifie un composant en plein mouvement, elle se doit d'être dynamique et de nous rappeler qu'on touche à un composant interne de l'objet en mouvement.

Quand on clique sur le moteur cette interface apparaît, tout le cadre est animé et les rouages tournent ce qui donne ce dynamisme et cette compréhension du joueur par rapport à ce qu'il est en train de faire.

Elle permet de régler la vitesse et le sens de rotation via un slider.

UI gestion du moteur



Si nous cliquons sur le compteur, alors le même cadre apparaît mais seulement le contenu change :

On retrouve la même interface que sur le model 3D du compteur. Elle permet ainsi d'initialiser le nombre de dents qui vont déclencher un mouvement en sortie.

UI gestion du compteur



En mode Edition, le dossier est présent et est interactif. Il contient beaucoup de couleurs chaudes avec du papier un peu jauni et une pochette en cuir. Chaque page du dossier comporte un logo qui classe les différents objets disponibles dans le jeu.

Nous avons choisi la clé comme symbole des engrenages car il symbolise le cœur de notre jouet et c'est grâce à lui que nous bâtissons. Pour le symbole des moteurs un éclair comme symbole a été créé, l'éclair étant dans la conscience commune un synonyme de mouvement et de motricité. Enfin pour le logo de connecteur nous avons choisis une forme plus abstraite coupée en deux afin de montrer que le connecteur est un moyen d'ajouter un objet avec un ou plusieurs engrenages.

Logo page
Engrenage



Logo page
Moteur



Logo page
Connecteur



Sur chaque page on retrouve des cases comportant un objet à placer. Les objets sont représentés en vue 2D dans ces cases. Chaque dessin est fait de façon à retrouver un style proche du dessin.

Icones des objets



Une fois tous les éléments combinés nous retrouvons un dossier fonctionnel afin que le joueur se retrouve dans les 3 différentes catégories. De plus une couche de dessin est rajoutée en plus afin d'accentuer le côté inventeur et Steam punk.

UI dans le projet en mode contemplation



UI dans le projet en mode édition



Page 1



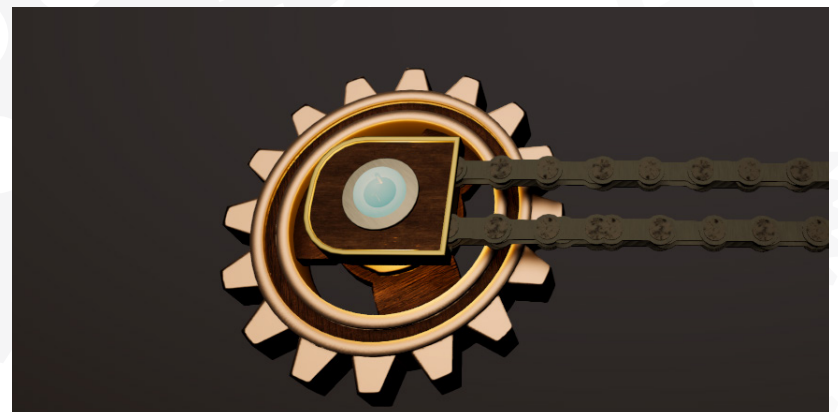
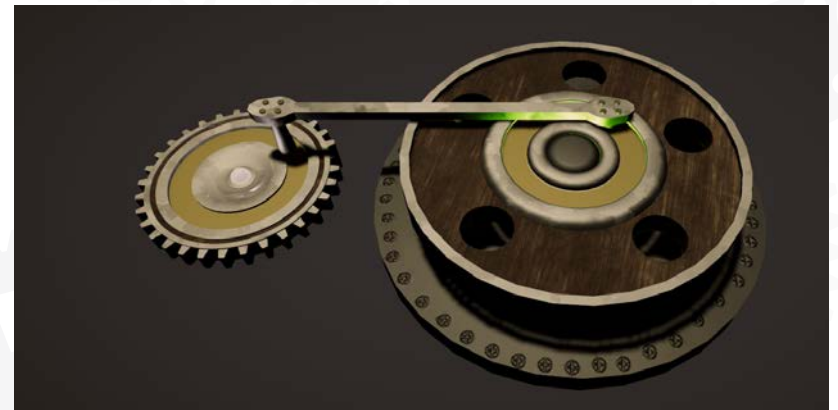
Page 2



Page 3



IV/Assets 3D



V/Illustrations

Intentions

L'ambiance et le thème général du jouet placent le joueur dans un cadre tant reposant que stimulant, celui d'un inventeur sur son bureau, dans un univers steampunk. Si le fond, et les différents objets ramènent le joueur dans cette ambiance, Gears utilise un autre outil pour l'y placer: des dessins en 2D parcourant l'espace de jeu, ainsi que le l'inventaire du joueur.

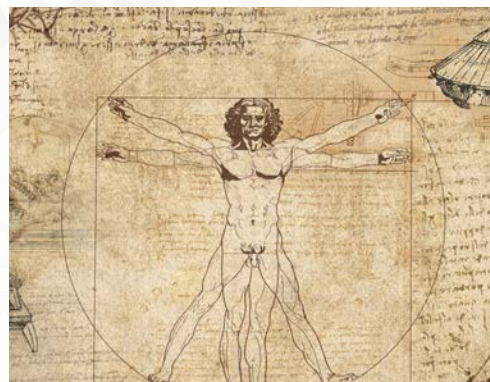
Le trait doit rappeler celui d'un dessin fait sur papier. On procède donc avec des dessins crayonnés, qu'on numérise pour les importer dans le projet.

Références et inspirations

Lorsqu'on pense à des dessins d'invention, des carnets d'inventeurs, etc... L'imaginaire commun nous mène généralement vers certaines images précises. Ces clichés sont exploitables pour plonger efficacement le joueur dans l'univers.

La référence principale, et la plus connue, est peut-être le patrimoine qui nous est descendu de Léonard de Vinci: des carnets de notes, des schémas, des figures, des formes... De dessins d'anatomie à systèmes automatisés, en passant par des machines volantes, ses inventions ont nourri notre imaginaire depuis 200 ans.

Parmi les traits marquants qu'on y retrouve, le style de dessin, apparenté à la branche académique de "dessin technique", qui s'est développé depuis dans les milieux de l'architecture et du design. Le but, en laissant les traits apparents, n'est plus d'avoir un objet beau, mais de laisser apparaître des détails, des valeurs, des formes ayant pour but de simplifier sa conception.



collection de dessins de Léonard de Vinci et Dessin technique d'un appareil photo, dans le style de Léonard de Vinci

On note deux aspects majeurs à ces styles artistiques. Tout d'abord, le matériau utilisé: pour obtenir cette couleur, sur du vieux papier, on doit s'intéresser au type de matériau utilisé à l'époque: le broux de noix, un colorant utilisé à l'époque en broyant la coque de noix, dans le but d'en faire une sorte de peinture, applicable sur du papier. Sans littéralement réutiliser ce matériau, sa couleur suffit généralement à nous ramener dans l'imaginaire collectif de l'inventeur du 18/19ème siècle.

Enfin, on retrouve et on note l'importance dans le rappel visuel de la géométrie et des mathématiques. Ce visuel, développé dans le Dessin Technique, impacte grandement le visuel des formes. On met en valeur les angles, les lignes directrices, les valeurs...

Enfin, à chaque dessin ses propres inspirations. On a pu entre autres s'inspirer du film d'animation Dragons, pour un passage par la forge de Hiccup. Du côté de l'inspiration, et de l'imaginaire steampunk, on peut s'inspirer et puiser de divers autres sources. Entre autres, le concept du bateau volant, souvent représenté avec des voiles tirées vers l'extérieur, peut-être un ballon, rappelant une montgolfière... On retrouve de telles formes dans La Planète au Trésor, ou plus récemment dans la série Arcane...

Arcane, Fortiche Studio



Dragon , DreamWorks



La planète au trésor, Disney

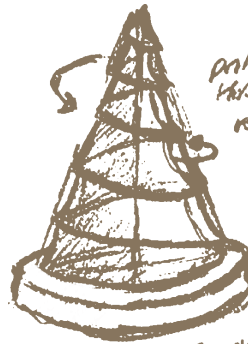
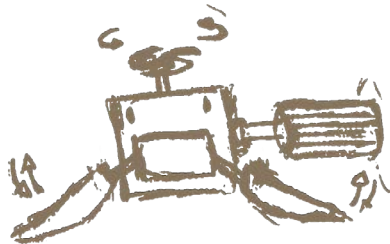
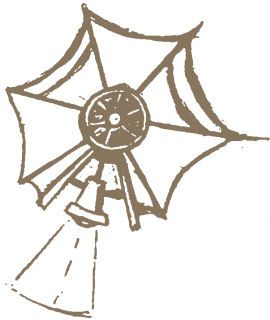
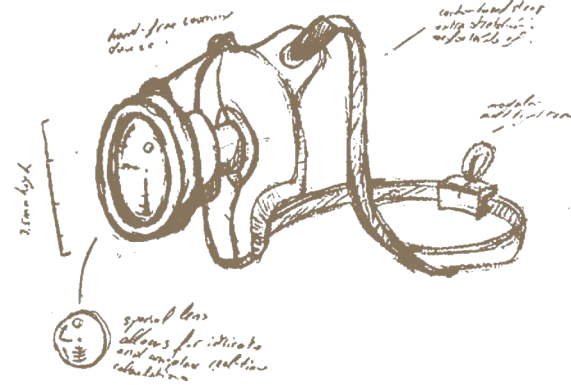


Collection de dessins conçus pour Gears



Thin Band

only one face, 1/2, 3 diameter

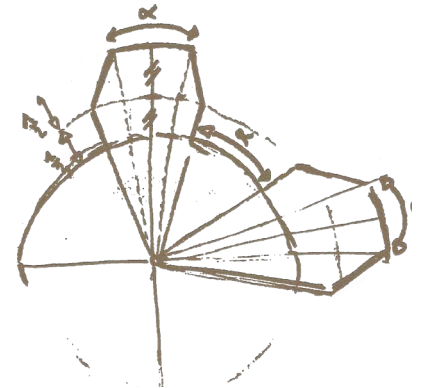
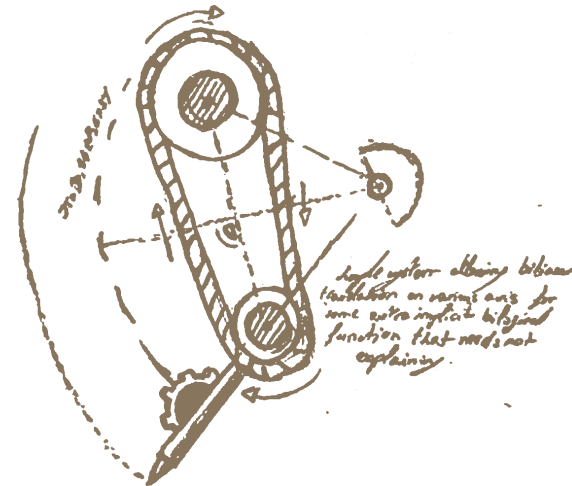
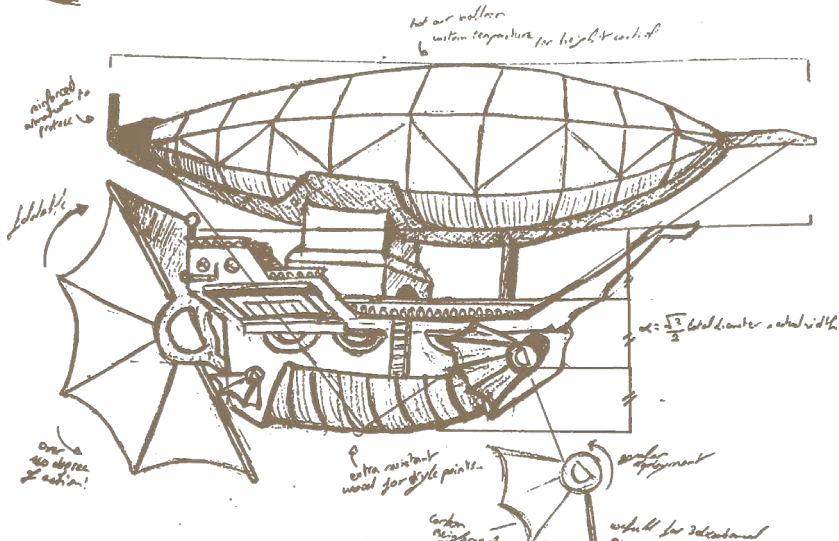
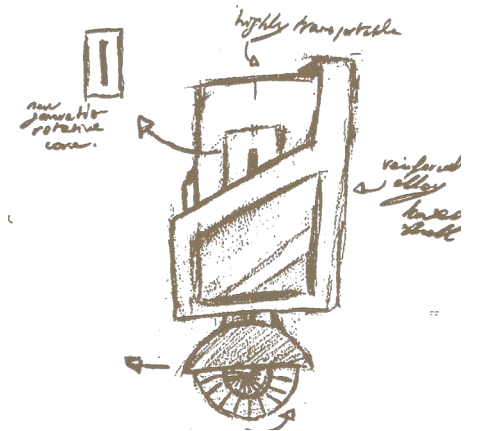


Drill through rock structures

attach motor



1/2 new design!



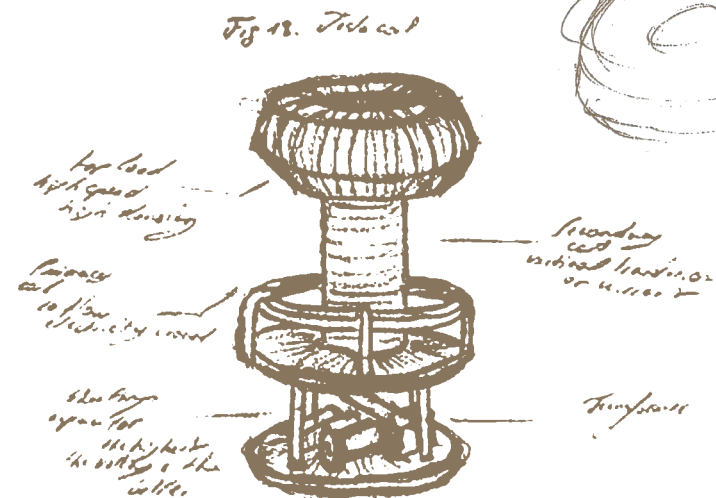
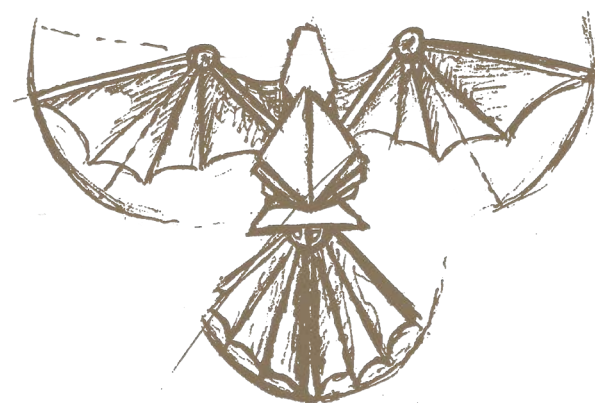
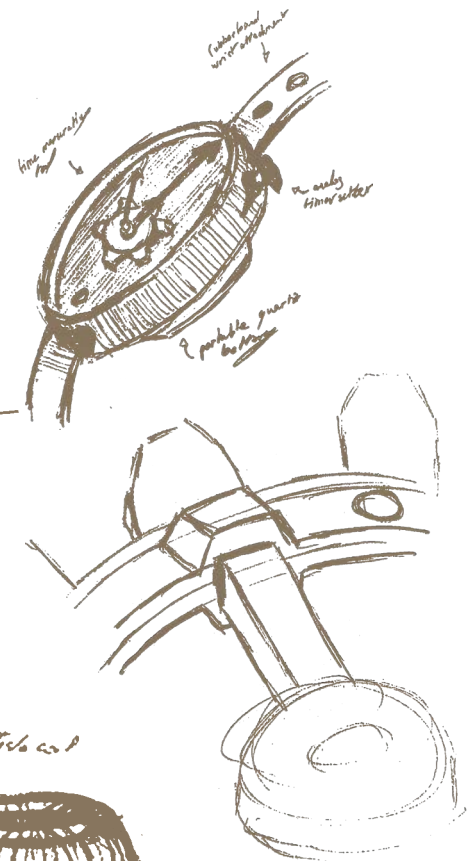
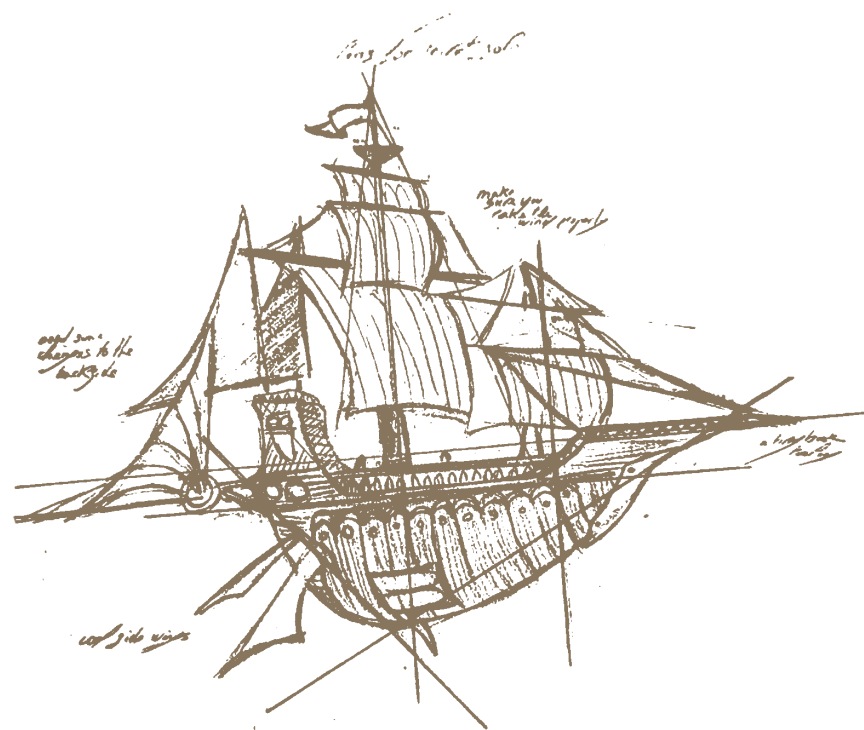
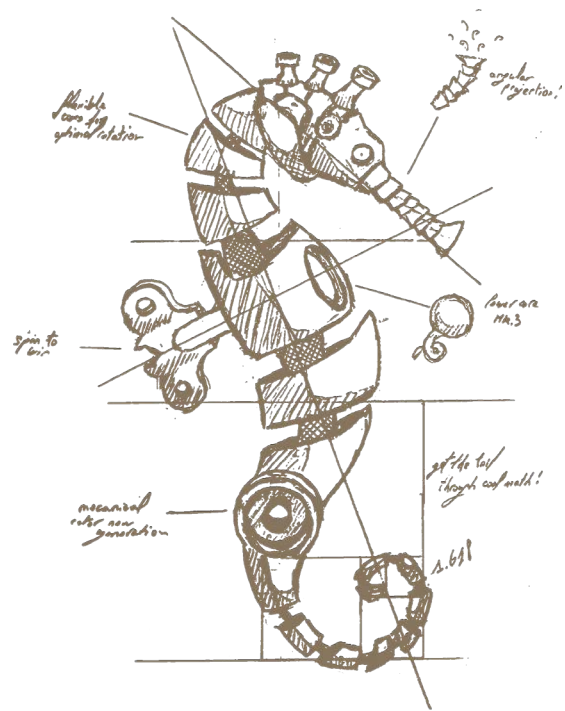


Fig 18. Tides and

The image features a white background with decorative gear patterns in the corners. The gears are in shades of grey and brown, arranged in a way that suggests a mechanical or industrial theme. They are positioned in the top-left, top-right, bottom-left, and bottom-right corners, framing the central text.

D.A.S

Direction Artistique Sonore

Notre jouet étant un bac à sable d'engrenage, les sons de celui-ci sont principalement mécaniques et plus précisément des mécanismes métalliques. Nous avons donc dirigé notre direction artistique sonore vers des sons métalliques.

Intention générale

Nos intentions générales de DA sonore suivent une ligne narrative que nous avons définie: le joueur doit pouvoir créer des mécanismes d'engrenages en ayant l'impression d'être un inventeur d'un univers steampunk se trouvant dans son bureau.

De plus, notre jouet est dissocié en 2 modes: l'un pour l'observation et l'autre pour la conception. De ce fait, nous avons décidé de montrer cet aspect narratif du sound design dans le mode d'observation afin que le joueur puisse être plongé dans l'univers steampunk du jouet en observant sa création. Et à l'inverse pour le mode conception, nous voulions que le joueur puisse conceptualiser dans le calme, la DA sonore dans ce mode se limite donc à de nombreux feedback servant à aider le joueur dans sa conception.

L'idée d'avoir ces 2 ambiances sonores différentes dans chacun des modes est de permettre une transition du mode conception où le joueur conceptualise dans le calme, au mode d'observation où le joueur va voir le système se déclencher et l'ambiance sonore narrative se lancer, cela dans le but de créer un effet "waow" en arrivant dans le mode d'observation.

Son d'engrenage

Les sons d'engrenage sont sûrement les sons les plus importants mais également les plus complexes de notre jouet. De ce fait, nous avons dû faire de longues recherches pour pouvoir les réaliser.

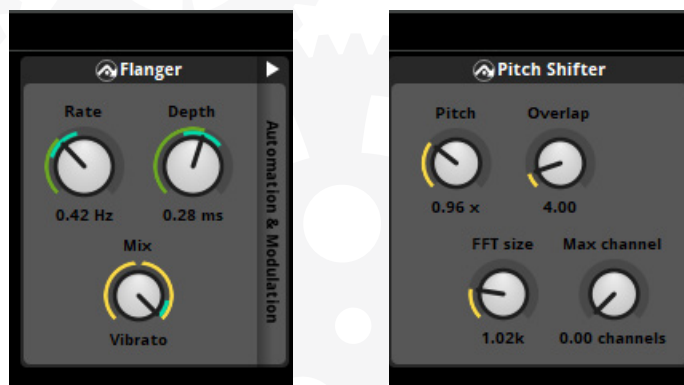
Nous avons commencé par écouter des systèmes mécaniques d'horloges en bois ainsi que des mécanismes d'engrenage en différents métaux et des différentes tailles. Nous nous sommes rendu compte que les sons d'engrenage en métal étaient certes réalistes par rapport à ce que nous voulions créer mais étaient nettement moins satisfaisant que des sons de mécanisme d'engrenage d'horloge en bois. Et certes nous voulions des effets sonores d'engrenages réalistes, mais nous ne voulions pas que le joueur ait envie de désactiver le son au bout de dix minutes de jeu. De ce fait, nous avons décidé d'utiliser des sons d'engrenage en bois puis des rendres plus métalliques grâce à des effets.



Horloge en bois de Christian Vergne

Nous avons donc commencé par enregistrer des sons d'engrenage provenant d'horloge en bois sur Reaper, nous les avons découpé, étiré puis ajusté afin d'avoir des sons satisfaisant puis nous les avons modifier sur Fmod grace aux effet "Flanger" et "Pitch Shifter" afin de les rendre plus métallique.

Fmod Flanger et Pitch Shifter

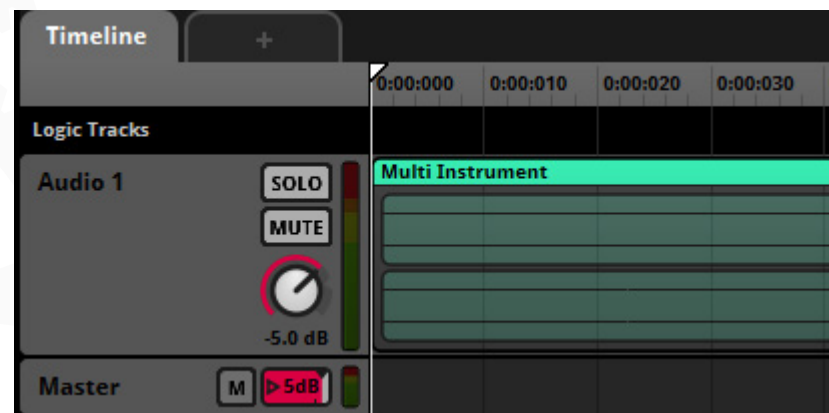


Plus en détail, lors de la réalisation nous nous sommes rendu compte qu'un son d'engrenage (comme nous voulions le réaliser) était constitué d'un premier son puis d'un deuxième et cela en se répétant, si nous devons le vulgarisé cela fait "clac clac". Afin d'accentuer l'effet mécanique nous avons décidé de réaliser deux événements sonores (chacun constitué de ces deux sons) par engrenage, ceux-ci s'alternent et se déclenchent au passage de chaque dent dans une dent d'un autre engrenage.

Notre jouet possédant trois engrenages nous avons donc réalisé six sons d'engrenage (deux pour chaque engrenage), évidemment les différents engrenages n'étant pas de la même taille nous avons réalisé des sons pour chacun d'eux qui diffèrent dans leur sonorité, le grand à un son plus grave et plus lisse contrairement

au petit qui à un son plus aigu et plus rugueux. Cette différence de sonorité entre les différents engrenages permet une ambiance mécanique presque musicale que le joueur peut concevoir à sa guise en plaçant les différents types d'engrenages. Le joueur peut ainsi jouer avec les sons des engrenages en modifiant différents facteurs: leur taille, leur vitesse, leur nombre et leur placement (les engrenages ont des sons spatialisés).

Fmod Multi Instrument pour le son des engrenages



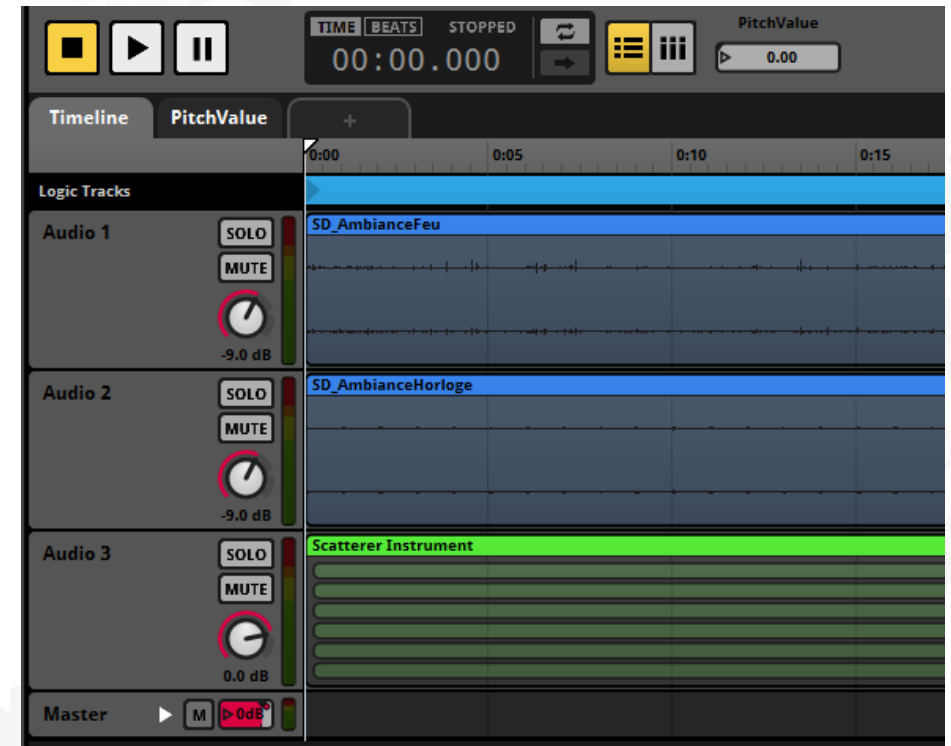
Son d'ambiance

Comme dit précédemment nous avons deux ambiances sonores, une pour chaque mode de jeu. Dans le mode édition nous voulons garder le joueur dans le calme afin qu'il puisse créer et conceptualiser en toute tranquillité. A l'inverse dans le mode observation le jouet doit permettre de faire plonger le joueur dans le monde steampunk dans lequel se trouve le jouet. De ce fait, le son d'ambiance du mode observation est composé d'un son de fond ainsi que d'événements sonores se déclenchant aléatoirement à des intervalles irréguliers, ceux-ci permettent de rendre l'ambiance sonore narrative plus vivante.

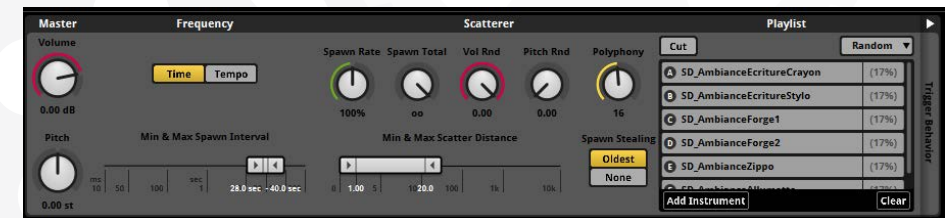
Le son d'ambiance de fond est constitué d'un son d'horloge ainsi que d'un son de feu de cheminée, ceci servent à simuler l'ambiance d'un bureau d'inventeur de sorte à ce que le joueur est l'impression d'être le protagoniste.

Afin de rendre cet espace sonore narratif plus immersif, des événements sonores se lancent toutes les 28-40 secondes. Nous avons conçu six effets sonores rappelant des événements pouvant se produire dans un bureau d'inventeur steampunk comme des sons de zippo de sorte à ce qu'on est l'impression que l'inventeur s'allume un cigare, des sons de forge qui proviendrait de l'extérieur ou encore des sons d'écritures sur un livre comme si l'inventeur notait des choses.

Ainsi le son ambiant du mode observation permet de plonger le joueur dans l'univers steampunk. Les différents sons présents dans Gear ont été fait de sorte à accentuer et compléter nos intentions de direction artistique générale.



Fmod son d'ambiance



Fmod paramètres du son d'ambiance

Event List

Voici l'évent list qui regroupe tout les sons utilisés. La liste est coupé en plusieurs point important comme la spatialisation et les différentes conditions de trigger

Objet	Mécanique	Paramètre	Description	Type	Condition de trigger	Spatialisé	Statut
Item	Snapping	Snap	Son ding satisfaisant positif	One-Shot	Drag Item On Another One	no	Completed
		Unsnap	Son ding satisfaisant positif plus grave	One-Shot	Remove From Snapping	no	Completed
	Create		Son de création métallique	One-Shot	Left Click On Item (inventory)	no	Optional
	Take		Son de drop inversé (est utilisé pour le create)	One-Shot	Left Click On Item (board)	no	Completed
	Drop		Son d'objet métallique lâché sur du papier	One-Shot	Release Right Click	no	Completed
	Delete		Son d'objet de snapping inversé	One-Shot	Right Click With Item	no	Completed
	Error	Snapping Error	Son de snapping puis dong erreur	One-Shot		no	Completed
		Main Error	Son dong erreur	One-Shot		no	Completed
Gear	Rotation	Rotation Speed	6 Sons mécaniques d'engrenages (2 par engrenage)	One-Shot		yes	Completed
Engine	Rotation	Rotation Speed	Son de moteur	One-Shot		yes	Optional
Meter	Rotation	Rotation Speed	Son mécanique d'engrenage	One-Shot		yes	Optional
Bridge	Rotation	Rotation Speed	Son mécanique de chaîne	One-Shot		yes	Optional
HUD and UI	Inventory Switch Page		Son frottement papier	One-Shot	Left Click On Inventory Page	no	Completed
Camera	Switch mode ON		Whoosh + bass (un peu comme une onde de choc)	One shot	Left Click On Mode Button	no	Completed
	Switch mode OFF		Mode ON inversé	One-Shot	Left Click On Mode Button	no	Completed
Main Lights	Switch mode ON		Son allumage lumière interrupteur	One-Shot	Left Click On Button	no	Completed
	Switch mode OFF		Son extinction lumière interrupteur	One-Shot	Left Click On Button	no	Completed
Ambiant	Ambiant Background Sound		Horloge & Cheminée	Loop		no	Completed
	Ambiant Event Sound	Random	Ecriture, Forge, Zippo	One-Shot		no	Completed



ican

institut de création et
animation numériques

