

A piece of white paper with blue handwriting and cyan lines, placed on a brown textured background. The paper is crumpled and has several sharp points. The text is written in a cursive, handwritten style. There are two horizontal cyan lines and two vertical cyan lines forming a rectangular frame around the text. The background is a brown, textured surface, possibly a piece of paper or fabric.

Heart
of paper



Heart
of
Paper



Sommaire



Equipe

Game overview

Avant-propos	6
Game Overview	6

Game design

Exemple de jeu	8
Boucle de gameplay	9
Contrôles	9
Éléments interactifs	11
Briques de gameplay	17
Caméra	22

Direction artistique

A propos	28
Univers	28
Chapitrage	28
Conception visuelle	34
Conception sonore	35

Documentation technique

Introduction	40
Problématiques	40
Architecture	44
Références	47

Ergonomie

A propos	48
Démarche	48

Production

Avant-propos	52
Cible et plateforme	52
Budget	55
Post-mortem	59

Remerciements





Equipe



Caroline Fidelaire



Gestion de projet

Marie-Cécile Defontaine



Ergonomie

Lucas Rousselot



Conception Sonore

Benjamin Michenaud



Game design



Audrey Laurent-André



Thomas Planques

Conception Visuelle

Julie Stuyck



Programmation



Florian Brulhart



Jonathan Connell

Game overview

Avant-propos

Heart of Paper est un projet réalisé par 9 étudiants dans le cadre de la deuxième année de Master de l'ENJMIN.

Le but de l'exercice est de mener à bien une pré-production d'un jeu commercial sur une durée de 6 mois, d'octobre 2010 à mars 2011, à partir de pitches proposés par les étudiants et retenus par l'équipe pédagogique.

Les projets sont présentés à un jury à trois reprises durant le développement, à intervalles réguliers. Lors de la dernière soutenance, les équipes présentent l'ensemble du projet ainsi que la maquette du jeu, comme si elles devaient le vendre à un éditeur.

Game Overview

Pitch

Entrez dans le monde changeant de **Heart of Paper**...

Dépliez les décors, donnez vie aux histoires, et plongez dans un monde où rien n'est ce qu'il semble être. Folie et démesure vous attendent au détour de chaque page... Manipulez le pop-up pour faire jaillir surprise et poésie.

Bien plus qu'un livre, une plongée dans l'illusion et un instant d'émerveillement.

Le pop-up

Il existe plusieurs types de livres pop-up :

- Les livres pop-up à proprement parler déploient des structures en volume à l'ouverture des pages.
- Les livres interactifs contiennent des mécanismes interactifs (tirettes, disques...) permettant d'effectuer des actions sur les éléments présents sur les pages (déplacement, changement d'illustration...).
- Les livres animés regroupent ces deux notions.

Le projet est basé sur le livre animé, qui nous permet d'exploiter à la fois le côté graphique intéressant du pop-up et les mécanismes du livre interactif comme base de gameplay.

Fiche d'identité

- **Plateforme** : iPad
- **Nombre de joueurs** : 1
- **Genre** : Puzzle / Énigmes
- **Cible** : 20-35 ans recherchant des expériences différentes du jeu mainstream
- **PEGI** : 7+
- **Camera** : Vue d'ensemble du livre, gestion semi-scriptée
- **Character** : Pas d'avatar, similaire aux god-games
- **Controls** : Tactiles
 - o 1 doigt : Interactions gameplay
 - o 2 doigts : Mouvements de caméra





Intentions

Lors du développement du projet, nous avons voulu mettre trois points en avant :

- **Un univers pop-up démesuré**

Utiliser le jeu vidéo pour s'affranchir des contraintes physiques du support traditionnel afin de proposer au joueur des décors improbables et démesurés.

- **Un contexte constamment renouvelé**

Utiliser la structure particulière du livre pour proposer au joueur un contexte différent à chaque niveau, et réinterpréter les mécaniques de jeu connues en fonction de chaque environnement.

- **Un environnement à façonner**

Donner au joueur une sensation de contact avec le livre, en plaçant la manipulation de l'environnement au cœur du gameplay.

Expérience de jeu

Heart of Paper place le joueur en position de lecteur-acteur, et lui propose de faire évoluer des situations narratives au travers de diverses thématiques liées au thème général de l'illusion. Pour ce faire, il devra agir sur les mécanismes du livre animé : tirettes, disques et personnages.

Ceci lui permettra de monter, démonter et modifier les décors, de jouer avec les personnages en les déplaçant, et de résoudre des énigmes en faisant appel à son imagination.

Chaque situation débloquée entraînera l'apparition de nouvelles interactions ou de nouveaux décors, et de fait, l'apparition d'une nouvelle situation, découlant de la précédente.

USP

- **Un pop-up démesuré !**

Le livre traditionnel est dépassé : toutes les folies sont possibles dans un pop-up n'ayant plus aucune limitation physique !

- **Soyez tout-puissant !**

Vous seul manipulez le décor et pouvez faire avancer l'histoire du livre !

- **Changez constamment d'univers !**

Chaque page vous emmène dans un univers différent pour vous raconter une nouvelle histoire !

- **Explorez toutes les facettes de l'illusion !**

Plongez dans un monde d'illusions à croire, à déjouer, et à créer !



Game design

Exemple de jeu

Partie type

Voici ce que pourrait être une partie typique de **Heart of Paper**, dans un niveau consacré aux dieux et aux mythes (sujet particulier exploré dans le troisième niveau du chapitre 2) :

- L'objectif général du niveau est de créer l'illusion de la présence d'un dieu. Le niveau prend place dans un décor d'extérieur défraîchi et terne, où peu de choses poussent et où les quelques édifices tombent en ruine.
- Le joueur doit premièrement faire traverser la page à un personnage du livre, représentant un fidèle, pour l'amener à l'autel du dieu.
- Il déplace les éléments du décor via les tirettes et les disques pour lui créer un chemin.
- Il peut tirer le personnage au bout de la page.
- Une fois arrivé au bout de la page, une animation se joue. Le personnage prie devant l'autel, ce qui fait apparaître un nouveau décor incomplet, représentant un temple.
- En manipulant ce nouveau décor et en actionnant les mécanismes dans l'ordre, le joueur lui permet d'obtenir sa forme finale.

- Le Temple laisse apparaître un indice textuel sur la prochaine action à effectuer : « Redonne à cette Terre son éclat passé ».

- En manipulant les nuages et le soleil, le joueur peut faire pleuvoir sur le décor. Celui-ci retrouve alors son éclat d'antan, il a réalisé une « action divine ».

- L'objectif général étant atteint, le joueur peut tourner la page, ce qui marque l'accession au niveau suivant, qui aura un thème et une narration entièrement différents.

Expérience de jeu

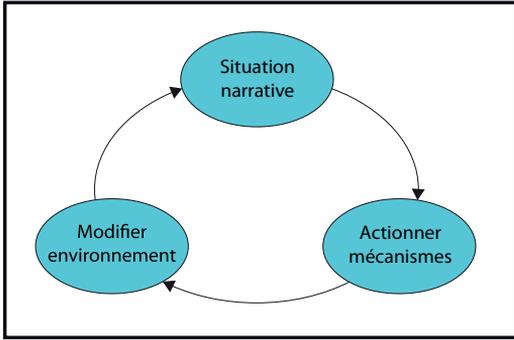
L'objectif est de reproduire l'expérience d'un véritable livre pop-up. Pour cela, on place le joueur en position de lecteur du livre.

Le joueur n'incarne donc pas un personnage précis à chaque niveau : en tant que lecteur, il est uniquement une force extérieure à la narration, qu'il fait progresser par ses actions.

Il a un point de vue extérieur sur le livre offert par une caméra scénarisée mettant toujours en valeur les éléments importants dont il a besoin pour résoudre les énigmes courantes.



Boucle de gameplay



Au début d'un niveau, le joueur est confronté à une situation narrative figée.

Des indices liés à la situation et au level design suggèrent au joueur les interactions à effectuer pour faire progresser la situation.

En agissant sur les éléments interactifs de la bonne manière, la situation évolue de manière scriptée.

L'évolution de la situation entraîne une modification de l'environnement (apparition/transformation de décor, déplacement d'éléments...). La situation est à nouveau figée et le joueur doit à nouveau la faire évoluer.

Dans la plupart des énigmes, il n'existe pas de condition de défaite :

- Le joueur ne peut pas se placer dans une situation bloquante.
- La narration reste simplement au même stade tant que l'énigme courante n'est pas résolue.

Il peut arriver exceptionnellement qu'il existe une condition de défaite :

- **Timing** : Si le joueur ne résout pas l'énigme dans un temps donné, il échoue.
- **Situations particulières** : Selon le contexte, des conditions de défaite particulières peuvent exister (disparition d'un élément du décor, perte d'un personnage...).

Quand il a perdu, il recommence immédiatement au début de l'énigme.

Contrôles

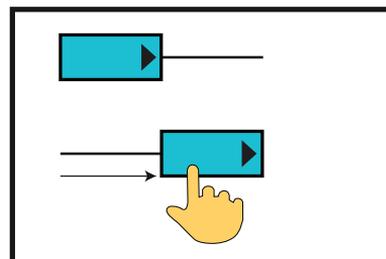
Les contrôles permettent d'une part d'agir sur l'**environnement**, et d'autre part d'agir sur la **caméra**.

Action sur l'environnement

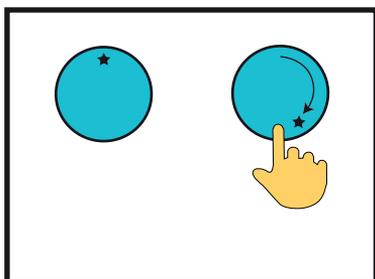
Le joueur peut interagir avec tout mécanisme dans son champ de vision, comme le pourrait le lecteur d'un vrai livre pop-up.

Il y a trois interactions directes possibles sur l'environnement, qui utilisent la même manipulation :

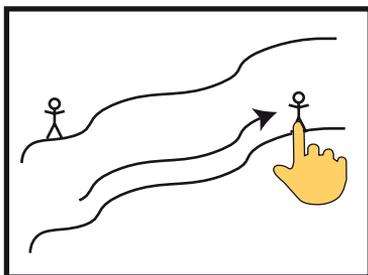
- Tirer/Pousser une tirette.



- Faire tourner un disque.



- Faire avancer un élément le long d'une rainure (équivalente à un chemin auquel l'élément est fixé).



Ces trois interactions s'effectuent en touchant l'élément avec un doigt, puis, sans relâcher, en glissant le doigt dans la direction voulue. L'élément glisse alors dans la même direction, dans la limite de son trajet défini.

Action sur la caméra

Deux actions sont possibles avec la caméra, le zoom/dézoom, et le travelling.

Zoom

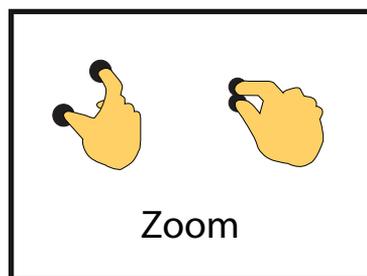
Pour zoomer, le joueur place deux doigts rapprochés sur l'écran, puis il les écarte. La puissance du zoom est proportionnelle à la distance entre les deux doigts, et la

vitesse du zoom est proportionnelle à la vitesse à laquelle le joueur les écarte.

Le zoom s'arrête dès que le joueur décolle au moins un doigt de l'écran.

Une fois le zoom terminé, la caméra reste au niveau de zoom courant, sans revenir au niveau initial.

La caméra étant toujours tournée vers un point d'attention, le joueur zoomera toujours dans cet axe, quelle que soit la position de ses doigts.



Dézoom

Pour dézoomer, le joueur effectue la manipulation inverse : il place deux doigts éloignés sur l'écran, puis il les rapproche. La puissance du dézoom est inversement proportionnelle à la distance entre les deux doigts, et la vitesse du zoom est proportionnelle à la vitesse à laquelle le joueur les rapproche.

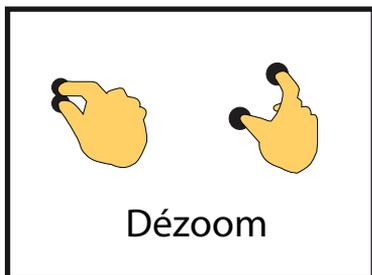
Le dézoom s'arrête dès que le joueur décolle au moins un doigt de l'écran.

Une fois le dézoom terminé, la caméra reste au niveau de zoom courant, sans revenir au niveau initial.

La caméra étant toujours tournée vers un point d'attention, le joueur dézoomera toujours dans cet axe, quelle que soit la



position de ses doigts.



Travelling

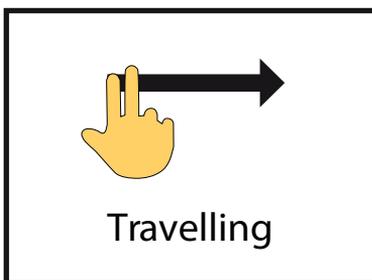
Pour effectuer un travelling, le joueur place deux doigts rapprochés sur l'écran, puis les déplace ensemble.

La caméra effectue le travelling le long d'une sphère dans la direction indiquée par les doigts.

La vitesse du travelling est proportionnelle à la vitesse de déplacement des doigts sur l'écran.

Le travelling s'arrête dès que le joueur décolle au moins un doigt de l'écran.

Une fois le travelling terminé, la caméra reste à la position courante, sans revenir à sa position initiale.



Éléments interactifs

Le jeu comporte trois types d'éléments interactifs : les **tirettes**, les **disques** et les **éléments sur rainure**.

Généralités sur les éléments interactifs

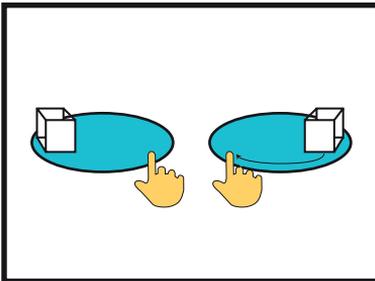
Les tirettes

- Une tirette est un rectangle clairement visible pouvant être tiré le long d'un chemin défini.
- Le joueur peut la déplacer dans les deux sens dans les limites de ce chemin.
- Une tirette est reliée à un ou plusieurs éléments de décor, et va agir sur ceux-ci selon un ou plusieurs des patterns d'action exposés ci-dessous quand le joueur la tire.
- Ce lien n'est pas représenté à l'écran, le joueur doit agir sur la tirette pour voir quels éléments de décor réagissent et comment.
- Une tirette peut toujours être tirée dans les 2 sens, et l'action effectuée est ainsi toujours réversible.



Les disques

- Un disque est une pièce de papier de forme circulaire, clairement visible, pouvant tourner sur lui-même.
- Le joueur peut le faire tourner dans les deux sens (horaire et antihoraire).
- Si des éléments de décor sont posés sur le disque, ils tournent comme celui-ci.



- Un disque peut aussi avoir une action similaire aux tirettes sur d'autres éléments de décor du livre.
- Il peut donc aussi être relié à un ou plusieurs éléments de décor et agir sur ceux-ci selon les patterns d'action expliqués ci-dessous.

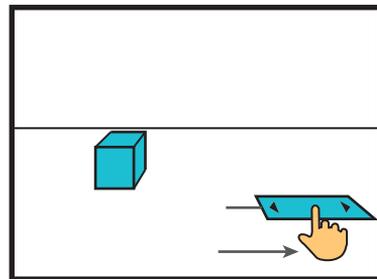
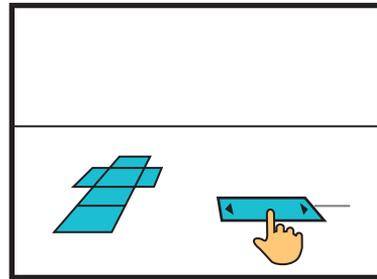
Les éléments sur rainures

Les éléments sur rainure peuvent uniquement être déplacés sur la rainure à laquelle ils sont attachés, et peuvent dans certains cas déclencher des actions ou des événements.

Actions possibles sur les éléments de décor

Faire apparaître/disparaître un élément de décor

Un élément de décor (simple pan plat ou construction plus complexe en 3D) se dresse en volume sur le livre. Il passe ainsi de l'état plat à l'état construit.

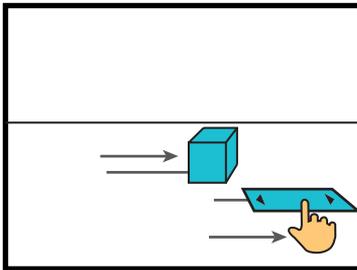
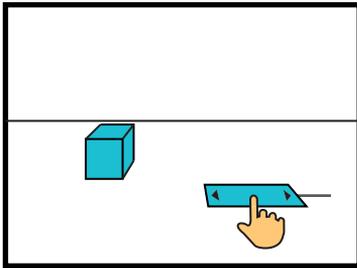


Déplacer

Un élément de décor se déplace le long d'une rainure.

Il peut ainsi par exemple bloquer ou créer un chemin, ou faire de la place pour l'apparition d'un autre élément de décor.

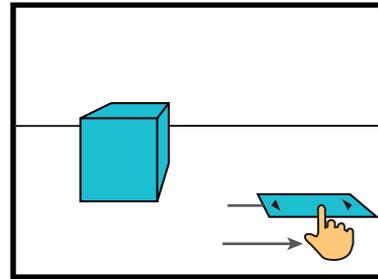
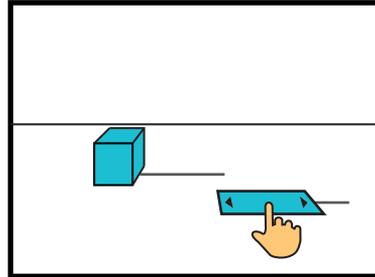




Changement de taille

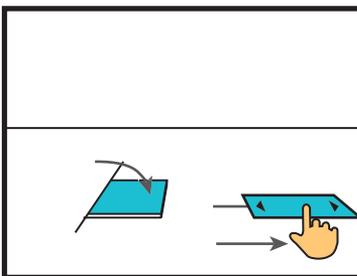
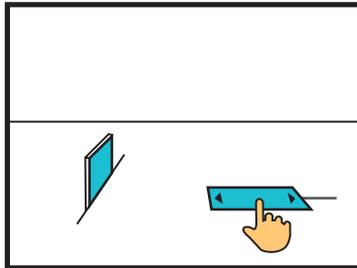
Un élément de décor change de taille, dans des limites déterminées.

Il peut grandir ou rapetisser.



Rotation

Un élément de décor pivote sur un ou plusieurs de ses axes, selon un angle déterminé.



Effets cumulatifs

Ces effets peuvent être **cumulatifs** : un mécanisme peut avoir un seul, mais aussi deux ou plus de ces effets à la fois sur un même objet.

De plus, un mécanisme peut agir sur plusieurs objets à la fois.



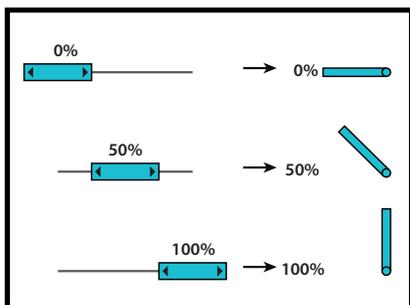
Fonctionnement des trois types d'éléments interactifs

Les tirettes

L'action effectuée sur le décor se produit proportionnellement au degré de progression de la tirette sur son chemin.

Exemple :

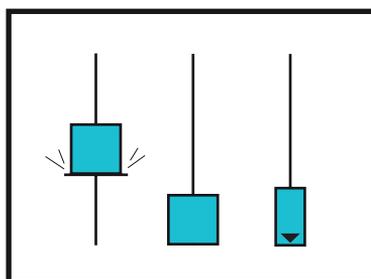
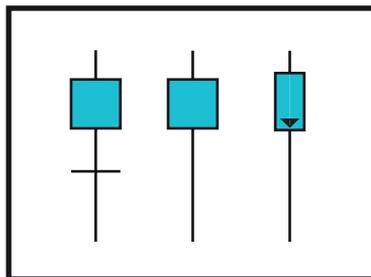
- Une tirette, tirée à 100% de son chemin, fait lever complètement (à 100%) un pan de décor.
- Si le joueur ne la tire qu'à 50%, le pan de décor est alors levé à 50%.



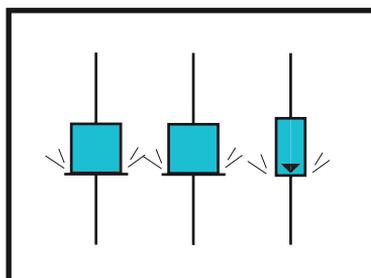
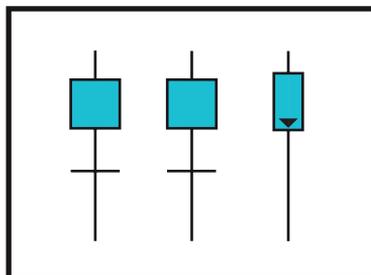
Il peut arriver que l'action effectuée sur l'élément par la tirette soit bloquée par un facteur quelconque : par exemple, la progression d'un élément le long de son trajet défini peut être bloquée par la présence d'un autre élément sur ce trajet.

Dans ce cas, il peut arriver trois choses :

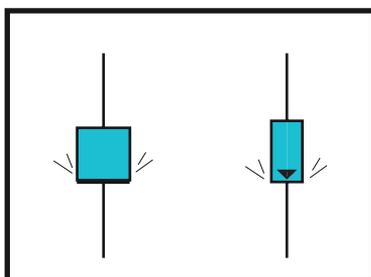
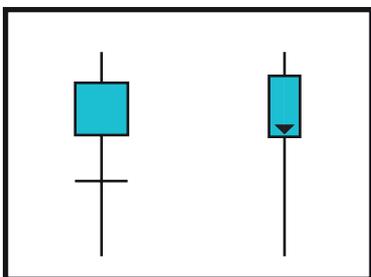
- Il est possible de continuer à tirer la tirette jusqu'au bout de son trajet, même si l'action sur l'élément ne s'accomplit pas jusqu'au bout. Comportement présent dans les cas où la tirette agit sur plusieurs éléments, et où au moins le trajet d'un élément n'est pas bloqué.



- La tirette se bloque dès le moment où tous les éléments sont bloqués. Comportement présent dans les cas où la tirette agit sur plusieurs éléments, au moment où tous les éléments sont bloqués.



- La tirette se bloque, et il est impossible de continuer à la tirer, au moment où l'action sur l'élément est bloquée. Comportement présent dans les cas où la tirette agit sur un seul élément



Les disques

Comme les tirettes, les disques fonctionnent selon un système de pourcentage.

Quand le disque est arrivé à 100% de sa progression, si le joueur continue à le tourner, l'action effectuée sur le décor s'inverse proportionnellement.

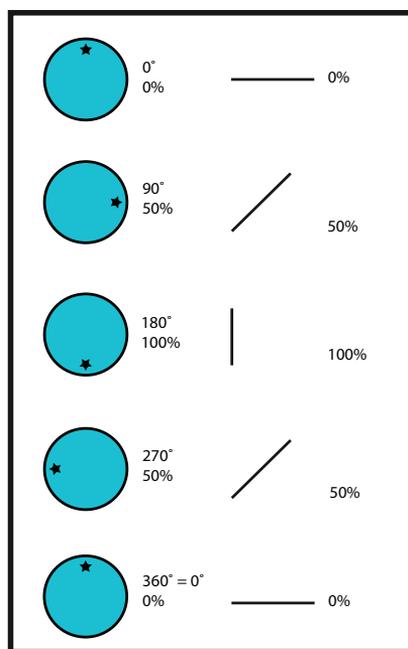
Exemple :

- Un disque fait lever un décor à 100% quand il est tourné de 180 degrés.
- S'il est tourné de 90 degrés, le décor est levé à 50%.

- Si le joueur continue à le tourner à 180 degrés, le décor est levé à 100%.

- Si le joueur continue à le tourner à 270 degrés, le décor baisse jusqu'à être levé à 50%.

- Si le joueur continue jusqu'à 360 degrés, le décor est levé à 0%, et ainsi de suite.

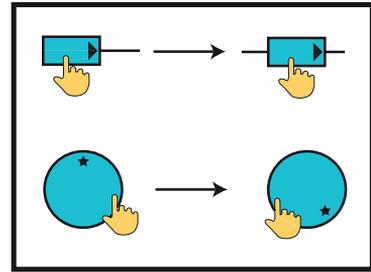
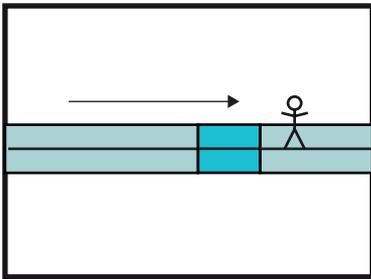
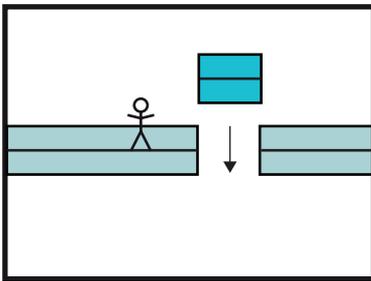
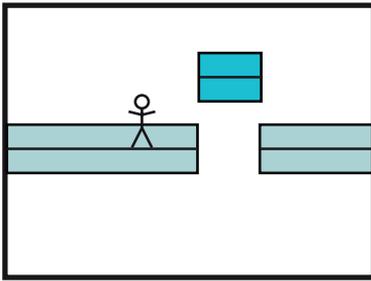


Les éléments sur rainure

Un élément sur rainure est un élément, généralement un personnage, que le joueur peut faire progresser directement sur la rainure en touchant l'élément et le faisant glisser le long de cette dernière.

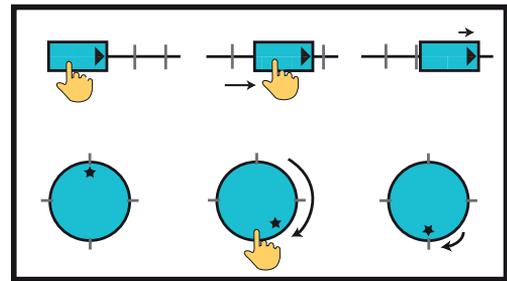
Si deux morceaux de décor comportent une rainure, et que les extrémités de celles-ci se touchent, il est possible de faire avancer le personnage d'une rainure à l'autre.





S'ils comportent des crans, on ne peut les déplacer/tourner qu'à certaines positions prédéfinies.

Si le joueur arrête la tirette ou le disque entre deux crans, il/elle se reculera automatiquement sur le cran le plus proche.



Comportement analogique ou à crans

Les tirettes et disques peuvent être analogiques, ou comporter des crans.

S'ils sont analogiques, on peut les déplacer/tourner n'importe où sur leur trajet/axe de rotation, et ils s'arrêtent précisément à l'endroit où le joueur lâche son doigt.

Textes

Les textes ne sont pas des éléments interactifs en eux-mêmes, mais sont une mécanique importante du jeu.

Ils fournissent des indices sur les énigmes, notamment pour les énigmes contextuelles (voir plus bas), en suggérant au joueur les associations d'idées à effectuer pour modifier le décor de la bonne manière.

Ils apparaissent de manière diégétique, intégrés au décor : une enseigne lumineuse dans une rue, le gros titre d'un journal que porte un personnage, etc.



Ils restent peu nombreux et courts : ils ne sont pas là pour raconter l'histoire intégralement, mais pour supporter le gameplay et supporter la narration de manière ponctuelle.

Si le joueur a plusieurs choses à faire dans un même décor, le texte placé à un endroit peut changer au fil de la progression du joueur.

Dans ce cas, une fois que la caméra revient sur l'endroit où était le texte après que le joueur ait terminé l'énigme qui y est liée, le même endroit affiche un nouveau texte utile pour la nouvelle énigme, puis la caméra se recentre sur l'objet de cette énigme.

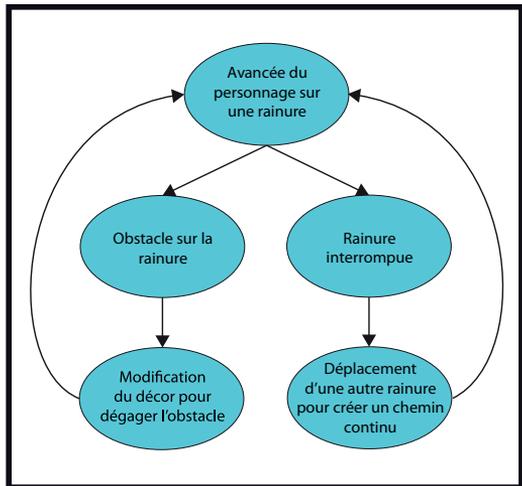
Briques de gameplay

Les mécanismes précédemment détaillés sont les éléments de base avec lesquels le joueur va interagir pour progresser dans le jeu.

Trois briques de gameplay distinctes, utilisant ces mécanismes, vont être entremêlées au fil des niveaux pour proposer trois grands types d'objectifs et de contextes d'utilisation de ces mécanismes au joueur.

Parcours de chemin

Le parcours de chemin donne pour objectif au joueur de faire aller un personnage sur rainure d'un point à un autre.



Les mécaniques principales de cette brique sont les suivantes :

- Dégager la rainure des obstacles qui empêchent le passage du personnage.
- Déplacer des pans de décor amovibles et tourner des disques, sur lesquels se trouvent des sections de rainure, pour coller celles-ci à la rainure où se trouve le personnage et lui permettre de progresser.

Le joueur alterne entre modification du décor et avancée du personnage :

- Il doit utiliser les mécanismes pour créer un chemin qui permet au personnage d'avancer un peu, puis faire avancer celui-ci.
- Il va ensuite réutiliser tout ou partie de ces mécanismes pour débloquer la suite du chemin, quitte à bloquer à nouveau la partie qu'il vient de débloquer : cela ne pose pas de problème puisqu'il a déjà fait traverser celle-ci au personnage.



- L'objectif n'est donc pas de créer un chemin complet du point de départ au point d'arrivée, mais d'alterner judicieusement entre création de chemin et avancée du personnage.

Cette brique joue principalement sur les combinaisons car :

- Plusieurs éléments de décor sont modifiés par un seul mécanisme.
- Plusieurs mécanismes agissent sur un même élément de décor.

Challenge

- Le challenge posé au joueur est donc de comprendre le bon ordre dans lequel activer les mécanismes pour libérer le chemin.

Conditions de victoire

- Parvenir au bout du chemin.

Conditions de défaite

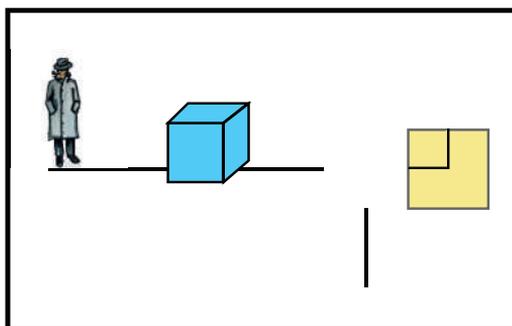
- Dans la majorité des cas, il n'y a pas de condition de défaite, le joueur reste juste bloqué jusqu'à ce qu'il trouve la bonne combinaison d'actions.

- Dans les cas où un timing est imposé au joueur, il perd s'il n'a pas atteint la fin du chemin dans le temps imparti.

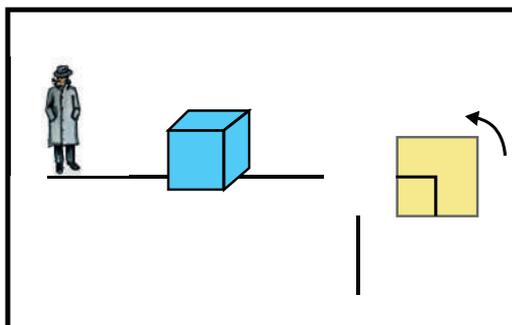
Difficulté

- La première variable de gameplay qui permet d'augmenter la difficulté est le nombre de mécanismes rentrant en jeu dans la résolution d'une énigme.

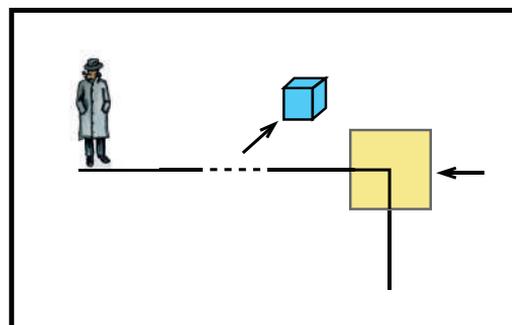
- La deuxième variable concerne le nombre d'objets affectés par un même mécanisme. Plus ce nombre est grand, plus la combinatoire est complexe à résoudre.



Le joueur doit faire avancer le personnage en bas à droite de l'écran. Un obstacle bloque le passage, et la rainure s'interrompt plus loin.



Le joueur commence par faire pivoter un élément de décor sur lequel se trouve la rainure manquante.

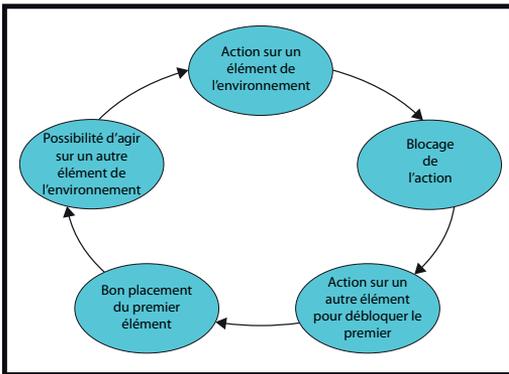


Le joueur déplace l'obstacle pour libérer la rainure, et place l'élément de décor comportant un morceau de rainure afin de compléter la rainure principale.



Construction de décor

La construction de décor donne pour objectif au joueur de reconstituer un décor entier formé de plusieurs éléments. Il fera apparaître notamment le sol, les murs, et les objets en relief posés sur le décor.



Les principaux mécanismes utilisés sont les suivants :

- Dépliage d'un objet, le faisant passer de l'état plat à l'état en relief – via tirette ou disque.
- Transformation d'un objet en un autre objet – via tirette ou disque.
- Rotation sur un de ses axes d'un pan de décor plat (mur, sol ou plafond) – via tirette ou disque.
- Déplacement d'un objet le long d'une rainure – via tirette.
- Déplacement courbe d'un objet posé sur un disque – via disque.

Cette brique joue principalement sur l'ordre dans lequel activer les mécanismes car les éléments peuvent collisionner les uns avec les autres, et donc se bloquer.

Par exemple, sur le schéma page suivante, si le joueur lève l'objet en relief avant de tirer le pan de décor de gauche, il ne peut plus faire pivoter le pan de décor complètement.

Challenge

- Le challenge posé au joueur est donc de comprendre le bon ordre dans lequel activer les mécanismes pour pouvoir monter entièrement la structure.

Conditions de victoire

- Monter la structure entièrement.

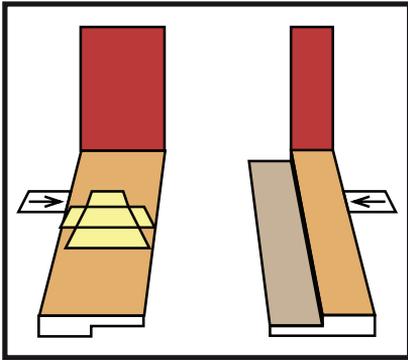
Conditions de défaite

- Dans la majorité des cas, il n'y a pas de condition de défaite, le joueur reste juste bloqué jusqu'à ce qu'il trouve la bonne combinaison d'actions.
- Dans les cas où un timing est imposé au joueur, il perd s'il n'a pas réussi à monter la structure dans le temps imparti.

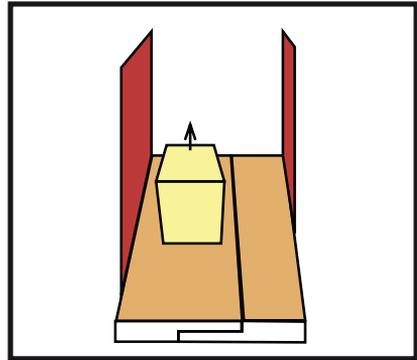
Difficulté

- La première variable de gameplay qui permet d'augmenter la difficulté est le nombre de mécanismes rentrant en jeu dans la résolution d'une énigme.
- La deuxième variable concerne le nombre d'objets affectés par un même mécanisme. Plus ce nombre est grand, plus la combinatoire est complexe à résoudre.

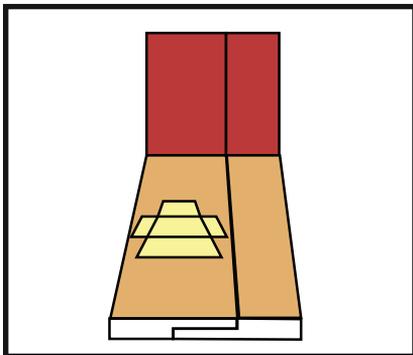




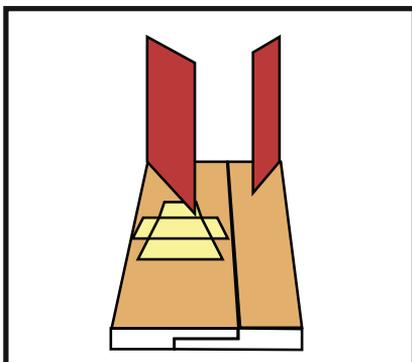
Le décor est formé de deux pièces principales.



Le joueur peut enfin faire apparaître un objet en volume. S'il l'avait fait avant, l'objet aurait bloqué la rotation du pan de décor de gauche.



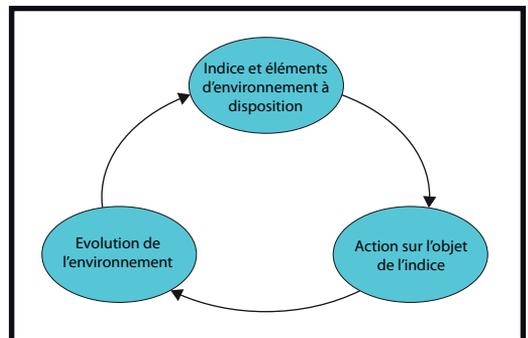
Le joueur commence par rapprocher les deux pièces de décor.



Le joueur fait pivoter les plans verticaux afin de les replacer correctement.

Énigmes contextuelles

Là où les deux briques précédentes font appel à un esprit logique, les énigmes contextuelles font appel à l'imagination du joueur et à sa capacité à penser différemment.



Le fonctionnement de ces énigmes est le suivant :

- Le joueur a un indice sur la manière de modifier le décor et sur le résultat souhaité (Cet indice peut être un texte inscrit sur le décor, un élément de décor mis en valeur par la lumière, le son, etc.)



- Il doit utiliser les éléments d'environnement à disposition pour modifier le décor de manière à parvenir à ce résultat.

- Une fois le résultat obtenu, le décor évolue et la narration progresse, permettant de passer à une nouvelle énigme.

- Plus le niveau est avancé dans le jeu, plus il sera fait appel à l'imagination et la capacité d'interprétation du joueur : le résultat demandé est de plus en plus éloigné de ce qui est possible matériellement avec les éléments à disposition, et il est de plus en plus demandé au joueur de créer une **illusion** de ce résultat plutôt qu'une réalisation concrète.

Challenge

- Le challenge repose sur la compréhension de l'indice et sur la réflexion à fournir pour comprendre les prochaines actions à effectuer.

Conditions de victoire

- Effectuer la/les bonne(s) action(s) sur l'objet de l'indice.

Conditions de défaite

- Dans la majorité des cas, il n'y a pas de condition de défaite, le joueur reste juste bloqué jusqu'à ce qu'il trouve les actions à effectuer.

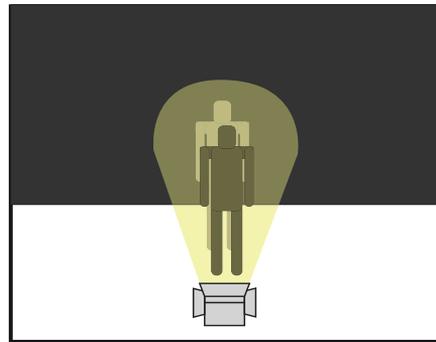
- Dans les cas où un timing est imposé au joueur, il perd s'il n'a pas réussi à effectuer les actions nécessaires dans le temps imparti.

Difficulté

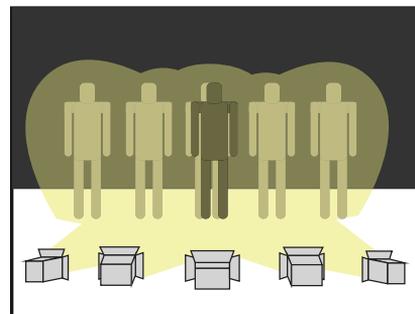
- La difficulté augmente au fur et à mesure que le lien entre l'indice et l'action à effectuer devient de moins en moins immédiat.



L'indice indique qu' « Une importante armée arriva sur les lieux. »

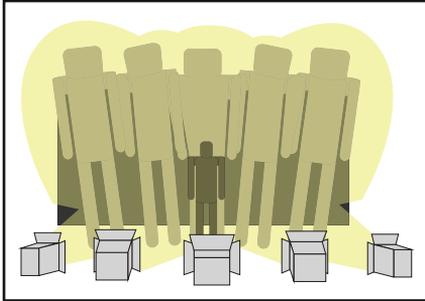


Le joueur n'a à sa disposition qu'un personnage, un plan vertical, et un ensemble de spots lumineux.



Le joueur va devoir placer astucieusement les spots pour qu'ils créent une multitude d'ombres du personnage sur le plan vertical...





...et créer l'illusion d'une armée ! Une animation se joue pour valider la résolution de l'énigme et concrétiser l'illusion, le joueur a réussi.

Éléments de variation

A partir de ces briques de base, les éléments de décor peuvent comporter divers effets particuliers.

Quelques exemples :

- Émission de lumière capable d'éclairer un chemin.
- Émission de son selon un pattern particulier donnant un indice sur la suite de l'énigme.

Caméra

Comportement semi-scripté

La caméra est **semi-scriptée**.

Elle est placée à un point prédéterminé à chaque moment du jeu, afin d'offrir un point de vue idéal sur la situation que le joueur a à résoudre.

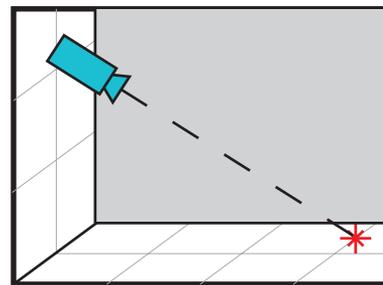
Le placement de la caméra n'est donc pas déterminé par le joueur, il change dynamiquement en fonction de son évolution dans la narration du niveau.

La caméra est toujours centrée sur un point d'attention, défini pour chaque position scriptée.

Les éléments configurables pour chaque position sont les suivants :

- Position de la caméra en x, y, z.
- Orientation selon les axes x, y, z.

Plusieurs facteurs peuvent influencer sur la position/orientation de la caméra.

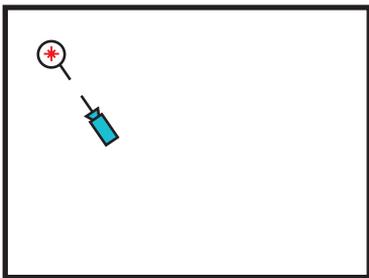


Progression dans le déroulement du niveau

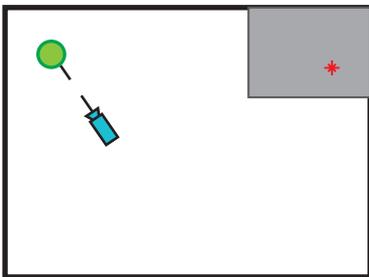
Une fois une action significative effectuée, la caméra passe à une autre position



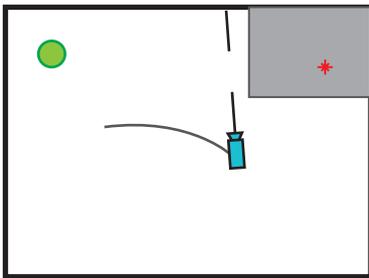
afin de mettre en valeur le lieu où le joueur devra effectuer la prochaine action pour progresser.



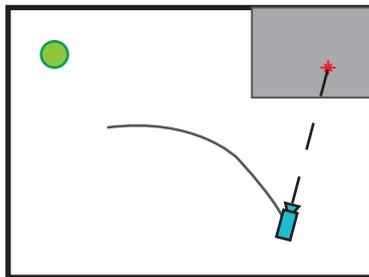
La caméra fixe le mécanisme au cœur de la première énigme.



Le mécanisme est activé, faisant apparaître un pan de décor en haut à droite de la page. Le nouveau point d'attention est situé sur ce décor.



La caméra s'y déplace.



Elle se fixe sur un point déterminé et est orientée vers le nouveau point d'attention.

Déplacement d'un personnage sur une rainure

Certains niveaux permettent au joueur de déplacer directement des objets sur des rainures, généralement un personnage.

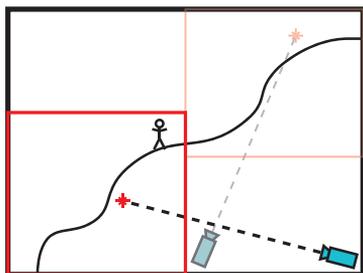
Ce personnage étant le point central d'attention du joueur et un élément clé de la résolution des énigmes, la caméra change de position pour toujours le mettre bien en vue du joueur.

Elle ne suit pas le personnage de manière continue : plusieurs points d'attention sont répartis le long de la rainure du personnage.

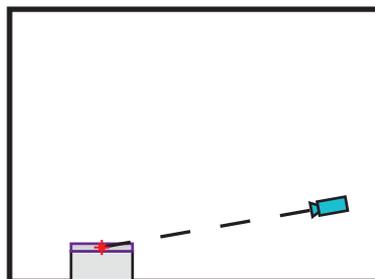
Chacun de ces points d'attention possède une aire d'effet : la caméra fixe ce point d'attention tant que le personnage est à l'intérieur de cette aire.

Quand il quitte l'aire d'effet d'un point d'attention, il rentre forcément dans l'aire d'effet d'un autre point d'attention, et la caméra passe alors dans la position liée à ce dernier.

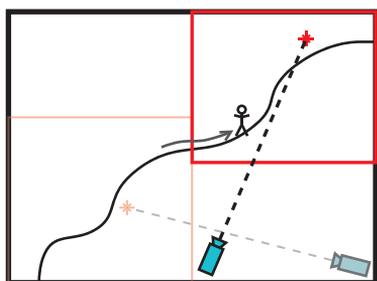




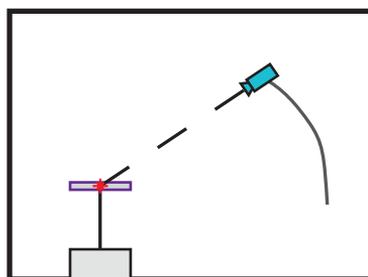
Le personnage est dans l'aire d'effet du premier couple [Point d'attention - Caméra]



La caméra fixe un mécanisme interactif.



Quand il passe dans l'aire d'effet du second, la position de la caméra et le point d'attention fixé par celle-ci changent.



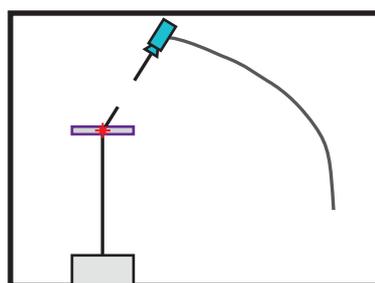
Au fur et à mesure que le joueur active celui-ci, il se déplace...

Action sur un mécanisme

La caméra peut modifier sa position et son orientation pendant que le joueur active un mécanisme donné particulièrement important, afin de mettre en valeur l'effet produit.

Cette modification est lente et progressive, et permet au joueur de continuer à activer le mécanisme en même temps.

S'il active le mécanisme dans le sens inverse avant de l'avoir activé au maximum, la position et l'orientation de la caméra reviennent à leur position initiale en fonction.



...et la caméra le suit, modifiant sa position et son orientation pour toujours observer le mécanisme.

Dans la plupart des cas, une fois le mécanisme activé au maximum, un changement de décor significatif se produit et la caméra change d'un coup son point d'attention afin de laisser voir ce changement.

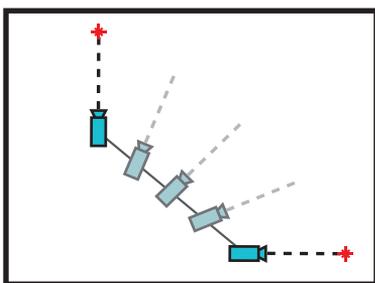


Déplacement scripté

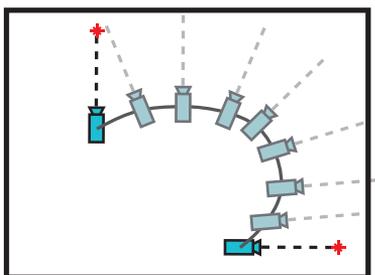
Quand la caméra se déplace d'une position scriptée à une autre, elle suit un trajet prédéfini.

Ce trajet peut suivre deux patterns différents :

- Une simple interpolation entre la position/orientation scriptée de départ et celle d'arrivée.



- Un pattern plus évolué : la position et l'orientation évoluent alors de manière plus complexe dans un but de mise en scène.



- Quand la caméra se déplace ainsi, le joueur n'a pas la main sur le jeu : il ne peut ni agir sur la caméra, ni sur aucun élément du décor.

Influence du joueur

Si la caméra possède une position et une orientation scriptées, le joueur peut avoir une influence limitée sur ceux-ci.

Ceci a pour but d'offrir un sentiment de liberté relatif au joueur, particulièrement important dans le contexte d'un livre pop-up où il doit se sentir libre de le tourner et de l'explorer à sa guise.

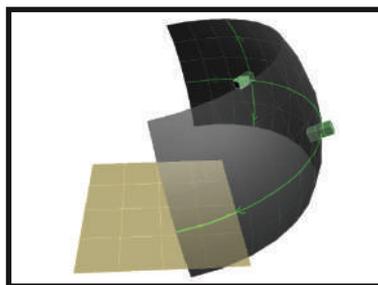
Le joueur peut donc effectuer 2 types d'action sur la caméra : le travelling et le zoom/dézoom.

Travelling

Le joueur peut déplacer la caméra le long d'une portion de sphère dont le centre est le point d'attention courant.

La caméra reste toujours orientée sur le point d'attention.

La taille et la forme de la portion de sphère sont prédéterminées et particulières à chaque position scriptée.

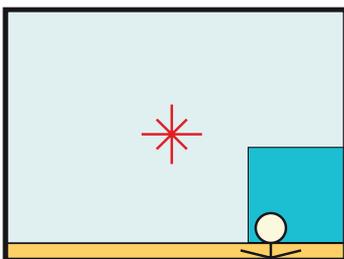
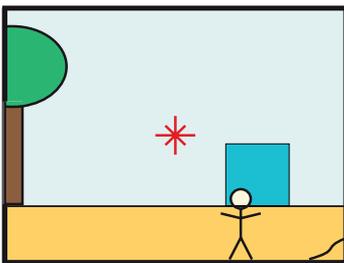
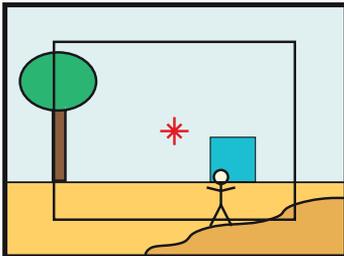


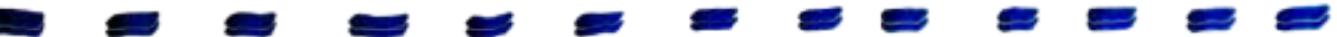
Ceci a un double but :

- Maintenir l'attention du joueur sur le point d'attention tout en lui permettant d'observer l'environnement.
- Mettre en valeur le volume et les formes du livre en permettant de tourner autour dans une certaine mesure.

Zoom/dézoom

Le joueur peut faire zoomer ou dézoomer la caméra, toujours centrée sur le point d'attention, dans des limites prédéterminées propres à chaque position scriptée.





Direction artistique

A propos

La direction artistique du projet va évoluer avec l'abstraction des thèmes abordés, tout en gardant le fil rouge du livre pop-up.

En effet, tout au long du livre, nous allons garder l'impression du papier autant au niveau des textures graphiques qu'au niveau sonore.

Tous les éléments 3D vont être traités comme des morceaux de papier découpés grossièrement et qui peuvent se plier et se déplier. Les animations soutiendront donc cet effet de pliage et de frottement entre les différents éléments découpés.

Les textures des éléments seront faites à partir de photos de papier, de carton et de matériaux qui rappellent la réalité des objets de scrapbooking de découpage-collage qui nous donnent une notion de toucher. Par-dessus ces matières premières, des effets de peinture et de dessin sont travaillés sur *Photoshop*, afin d'ajouter l'impression de fait-main.

L'ensemble des sons génériques de **Heart of Paper** se compose d'enregistrements de différentes matières (papier, carton, etc) afin de soutenir les graphismes et renforcer la sensation de réalité liée aux matériaux présents dans les différentes scènes. Leur intégration a été réalisée de façon à ce que ces sons se comportent de manière aléatoire. Au sein des différents enregistrements, on trou-

vera plusieurs «formes» : froissé, déchiré, frotté, etc... Cela dans le but de créer une diversité sonore utile aux différents feedbacks nécessaires au jeu.

Univers

La narration du jeu n'est pas constituée d'une unique histoire démarrant au début du livre et se terminant à la fin.

Elle se base sur le développement d'un thème : l'illusion.

Chaque niveau est formé par une double page du livre. Il traite un sous-thème particulier lié à l'illusion, et possède sa propre structure narrative, indépendante des autres, entièrement axée autour de ce sous-thème.

Les niveaux n'ont donc pas de continuité scénaristique entre eux autre que le fil rouge du thème général de l'illusion.

Il y a dix niveaux en tout. Neuf d'entre eux sont placés dans trois chapitres distincts qui s'enchaînent, et le dernier niveau est l'épilogue.

Chapitrage

Chaque chapitre s'axe sur une vision différente de l'illusion, partant de la plus concrète pour évoluer vers la plus abstraite.



Chapitre 1 Les illusions physiques : Donner des ailes à la matière

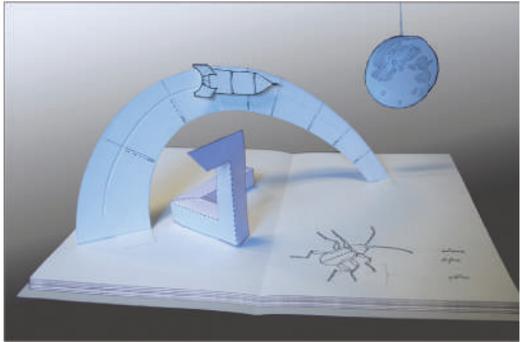
Ce chapitre traite des illusions liées purement à un support physique, comme les illusions d'optique ou les maquettes.

Direction artistique

Ce chapitre est interprété comme une définition basique de l'illusion, la première chose que l'on explique pour décrire un mot, presque comme une définition de dictionnaire. En ce sens le traitement graphique et sonore va être abordé de manière épurée.

Graphismes

Dans ce chapitre les textures seront peu colorées. On a ici une vision épurée et simplifiée de la réalité. Donc essentiellement un traité crayonné/tracé sur du papier blanc rehaussé par des touches de couleurs, porté par des illusions d'optiques simples.



Sound design

Ici, l'ambiance sonore est minimaliste : l'accent est mis sur les feedbacks donnés au joueur, car c'est bien dans ce chapitre que le joueur découvre le gameplay de **Heart of Paper**. D'un point de vue esthétique, l'ambiance sonore et musicale est constituée de sons de synthèse discrets. Le point de vue sonore est ainsi focalisé sur le gameplay, et l'ambiance épurée reste en relation avec l'esthétique visuelle.



Niveau 1

Reproduction de la réalité : Les briques de la magie

Sujet abordé

Maquettes et autres reproductions physiques littérales de la réalité. Le joueur devra y créer diverses maquettes et faire avancer des modèles réduits de personnages dans celles-ci.

Gameplay

- Introduction des mécaniques de base : tirettes, disques et rainures.
- Introduction des deux premières briques de gameplay : parcours de chemin et construction de décor.

Niveau 2

Trompe-l'œil : Tu me vois, tu ne me vois plus

Sujet abordé

Ensemble des illusions jouant sur le point de vue du spectateur pour fonctionner : anamorphoses, trompe-l'œil, etc. Le joueur devra premièrement monter le support de l'illusion, puis trouver le bon angle de vue pour les faire fonctionner et ainsi les faire vivre.

Gameplay

- Complexification des mécaniques de construction de décor.
- Introduction de la troisième brique de gameplay : énigmes contextuelles.

Niveau 3

Illusions sensorielles : Mensonge de nos sens

Sujet abordé

Illusions liées à notre perception imparfaite de la réalité, jouant sur les défauts de l'œil et de l'oreille, pour parvenir à des illusions d'optique et illusions sonores.

Gameplay

- Complexification des énigmes contextuelles, se basant sur l'exploitation des illusions sensorielles pour créer des situations trompeuses.

Chapitre 2

Les illusions qu'on se transmet : Crois en moi

Ce chapitre traite des illusions qui sont créées par une personne à destination d'une autre : art, spectacle, mythes... Se basant sur des supports physiques de natures variées (costumes ou décors pour le spectacle, écritures pour les mythes...), ceux-ci sont utilisés non pas pour créer l'illusion en elle-même, mais pour la suggérer à l'imagination du destinataire.

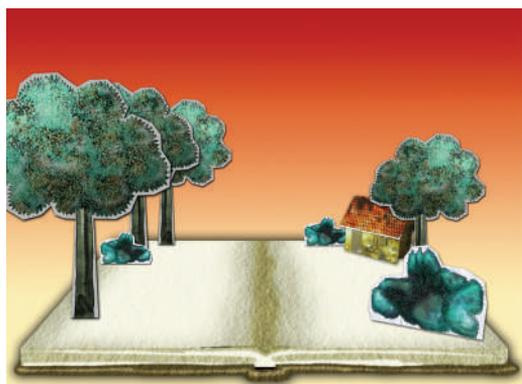
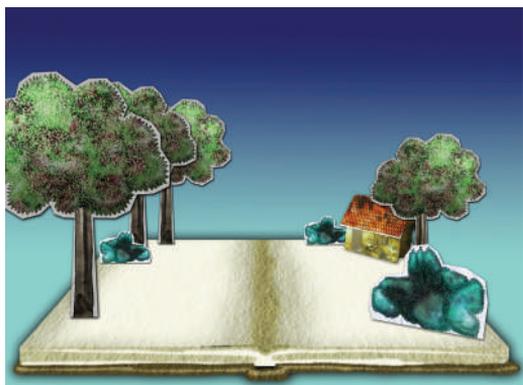
Direction artistique

Dans ce chapitre les illusions sont là pour raconter des histoires, et en ce sens la direction artistique va recréer une sorte de réalité figurative.

Graphismes

Ce second thème sera traité avec plus de couleurs et de détails que le précédent. Traité figuratif mais simplifié, rappelant des illustrations de livres.





Sound design

L'ambiance ici a pour rôle de soutenir la narration des scènes de ce chapitre : elle doit insuffler la vie dans le déroulement des événements. Les feedbacks sont ainsi en relation avec les objets représentés. Néanmoins, certains feedbacks, extérieurs à la scène, servent à personnifier les éléments narratifs, notamment les personnages. La musique quant à elle est constituée d'instruments reconnaissables et réalistes.

Niveau 4

Le spectacle : Le masque au grand jour

Sujet abordé

Le spectacle, se basant sur des décors, des costumes et des effets de scène, et invitant le spectateur à y voir plus que les artifices, mais ce qu'ils représentent : la réalité alternative qu'ils mettent en scène. Le niveau démarrera ainsi sur une scène de spectacle, que le joueur fera se dérouler, pour oublier au fur et à mesure de la progression la nature factice de ces événements, comme un spectateur se plonge dans un spectacle au point d'oublier que tout ceci n'est que trucs et illusions.

Gameplay

- L'ensemble des mécaniques et des briques de gameplay sont de plus en plus liées, pour une expérience plus rythmée et plus homogène, maintenant que le joueur a bien intégré le fonctionnement du jeu.

Niveau 5 :

L'art - Viens voir...

Sujet abordé

L'art est une réalité illusoire, un message transmis au destinataire afin de l'emmener dans un monde d'imagination.

Quelle que soit la composante physique de l'œuvre, elle ne constitue qu'une porte vers la réflexion ou les émotions où l'artiste souhaite emmener le destinataire, qui s'en créera une représentation imaginaire transcendant largement le support physique. Le joueur explorera les différents supports de l'art et les illusions les plus caractéristiques de chacune.



Gameplay

- L'accent est mis principalement sur la brique de gameplay « Énigmes contextuelles ». Le joueur va devoir créer de nombreuses illusions en rapport avec chaque forme d'art pour mettre en valeur le fonctionnement de chacune.

Niveau 6

Les mythes – Seuls les dieux sont réels

Sujet abordé

Foi en l'invisible, en les mythes et les dieux. La composante physique est de plus en plus faible et plus du tout indispensable (autels, textes, lieux de culte) et ne constitue plus qu'une porte vers des croyances incroyablement complexes. Voir l'invisible, interpréter les signes du monde naturel pour y voir l'expression de puissance supérieures, tels sont les thèmes de ce niveau. Le joueur créera ainsi l'illusion de la présence de dieux et d'esprits en manifestant ces signes à la vue de personnages.

Gameplay

- Le joueur va devoir principalement agir sur le décor et ses éléments afin de répondre aux demandes des personnages du livre, créant ainsi à leurs yeux l'illusion de l'existence de leur dieu.

Chapitre 3

Les illusions qu'on se crée : Le monde invisible

Ce chapitre traite des illusions qu'on se crée soi-même : folie, rêve... Elles n'ont plus aucune réalité physique et n'existent que dans la tête de leur créateur.

Direction artistique

Dans ce chapitre la cohérence perd le nord, et la direction artistique va soutenir l'irréel par un traitement un peu plus abstrait.

Graphismes

Les couleurs, très présentes dans ce chapitre, ne seront plus cohérentes avec les objets sur lesquels elles sont apposées. Le traité « peinture » sera plus lâché, type tâche de peinture. Et les objets seront disproportionnés les uns par rapport aux autres.



Sound design

L'ambiance sonore est ici plus expérimentale. Les sources sonores sont implicites et soutiennent l'ambiance imagination/cauchemars. Le trouble est soutenu par le fait que les sons ne sont pas explicitement diégétiques ou extra-diégétiques, perdant volontairement le joueur d'un point de vue sonore et renforçant l'impression d'un environnement irréel.

Niveau 7

L'imagination : La porte intérieure

Sujet abordé

Les choses que l'on imagine n'ont vie que dans notre tête. Processus propre à l'humain, elle nous permet de nous évader en nous créant de nouveaux univers, de nouveaux événements, tous malléables à volonté.

Gameplay

• Dans ce niveau, le joueur va faire se développer de nouveaux éléments de décor. L'idée est de partir d'un décor plutôt normal et de le faire évoluer vers des choses de plus en plus folles, comme lorsque l'on imagine quelque chose, au gré des associations d'idées. Les énigmes contextuelles serviront ici à indiquer ces associations afin d'aider le joueur à appréhender la suite des événements.

Niveau 8

Les rêves : L'incontrôlable éphémère

Sujet abordé

Quand notre inconscient prend la main pour nous faire vivre dans des mondes fantasmés. Contrairement à l'imagination, nous ne sommes pas conscients lorsque

nous rêvons, nous n'avons aucune prise sur ces univers. La seule chose à laquelle nous pouvons nous raccrocher, c'est la petite part du réel, nécessaire pour que les rêves prennent vie.

Gameplay

• Tout comme le niveau précédent, l'idée est de partir d'un décor qui ne paraîtra pas anormal au début. En plus des décors, ce niveau mettra en scène des situations décalées et improbables que le joueur devra faire évoluer. Il devra comprendre les relations entre le réel (la base du rêve) et la suite des événements (qui sera déconnectée de la réalité).

Niveau 9

La folie : La dernière frontière

Sujet abordé

Stade ultime des illusions que nous nous créons, la folie nous emmène dans une réalité parallèle à la nôtre, malgré nous, et malgré notre conscience et nos sens. Le fantasme se mêle au réel, et il n'y a alors plus aucun moyen de distinguer le vrai du faux...

Gameplay

• Ce niveau propose au joueur une atmosphère dérangement, dans laquelle il est difficile de distinguer ce qui est normal de ce qui ne l'est pas. Utilisant les différentes briques de gameplay, ce niveau va tout le temps « tromper » le joueur, sur ce qui est tangible, et sur ce qui est fantasmé.



Épilogue : Le livre animé

Cet épilogue se compose d'un seul niveau. Il constitue un feu d'artifice final liant explicitement les deux thèmes forts du jeu, le livre pop-up et l'illusion.

Direction artistique

Ce chapitre réunit des éléments des trois chapitres du livre, tout en créant une harmonie, afin de présenter une sorte d'apogée dans le livre de ce qu'il est dans son extravagance : plus de formes qui se déplient, de la démesure, tout en mettant en avant vraiment les mécanismes du livre animé.

Ce niveau de fin a pour but de mettre en avant le livre animé, comme une sorte d'hommage qu'on lui ferait. Présenter tout ce qu'il représente de beau et de magique pour nous.

Niveau 10

Le livre animé : Finalement, on en est là...

Sujet abordé

Point d'orgue du jeu, ce niveau remet le livre pop-up au centre des attentions. En effet, c'est ce support qui nous a permis de mettre en scène tout ce jeu d'illusions depuis le tout début.

Gameplay

- Mis en scène comme un feu d'artifice final, le joueur réutilisera ici toutes les briques de gameplay poussées à leur paroxysme. Plus qu'un gameplay particulier, l'idée ici est de jouer avec « l'envers du décor » et la différence entre un objet et sa représentation dans le livre.

Conception visuelle

Première approche

L'idée de ce jeu, depuis le début de sa conception et à toutes les étapes de sa réalisation, tourne autour du concept du livre animé. Ce concept issu d'un objet réel nous offre nombre de possibilités dans le monde numérique, car il peut être développé au-delà de ses capacités physiques.

Par-dessus s'est rajouté le background thématique qui nous a poussés à développer une évolution sur toute la direction artistique en tenant compte des différents chapitres et thèmes abordés.

Références

Les inspirations graphiques de ce jeu sont tirées essentiellement du livre animé (réel/physique) et de l'illustration. Les recherches ont porté sur la manière dont le mouvement était développé : depuis quel point de vue les formes sont-elles intéressantes à regarder, quelle est la place du texte dans les pages et les formes?

Tous les types d'illustrations et de traités graphiques sont amis avec le style du livre animé et du pop-up. Nous avons donc choisi d'adopter un style qui rappellerait le toucher chaleureux du papier au joueur qui est seulement face à un écran.

Personnages

La question de l'importance des personnages dans le jeu a été plusieurs fois soulevée au cours de la conception. Nous avons décidé qu'ils ne devaient pas faire de l'ombre au pop-up et donc plus s'inté-



grer au livre en tant qu'éléments de papier interactifs et non pas en tant qu'avatars.

Pour cela, les personnages ont un traité similaire aux objets du livre, et n'ont pas d'animation classique. Ce sont des bon-hommes en papier découpés qui s'animent par phases d'émotions, c'est-à-dire qu'ils ont plusieurs poses correspondant à des situations et qu'ils passent de l'une à l'autre sans images d'intervalle. Ainsi les personnages sont comme des marionnettes d'ombres chinoises qu'on remplace-rait d'une étape à une autre en fonction de la situation.

Décors

Notre livre virtuel se place dans un univers inconnu et sans importance, car ce qui est important, c'est le livre lui-même. C'est pour cela qu'il n'y a rien autour du livre, mis à part une sorte de bulle de couleur unie simplement dégradée en fonction de l'ambiance donnée dans le niveau courant. Cette bulle n'a aucune texture et ne s'apparente à rien de connu, car elle est simplement là pour soutenir les différentes phases de jeu et leur couleurs.

Pour tout ce qui est des formes dans le livre, comme décrit précédemment dans la partie sur l'univers, nous avons utilisé des textures de papiers soulignées par des crayonnés faits main et assemblés numériquement. Le traitement de la couleur, lui, varie en fonction des chapitres.

Les lumières sont essentiellement de petites sources qui soutiennent l'idée du papier comme un montage, une maquette de papier.

Conception sonore

Intentions

Le son dans **Heart of Paper** a 3 vocations:

- **Soutenir le gameplay**
- **Appuyer les sensations du joueur**
- **Rendre vivant quelque chose de figé**

En définitive, l'environnement sonore de **Heart of Paper** reste cohérent avec un univers fait en papier, proche du matériau, en le sublimant grâce à un autre niveau de lecture, appartenant à l'imaginaire et à la représentation. Cela permet ainsi de donner vie à cet univers.

Organisation des couches sonores

- Premier Plan
Sons de manipulation du pop-up et feedbacks importants.
- Second Plan
Ambiance Sonore.
- Troisième Plan
Stingers Musicaux.

Direction Artistique

La direction artistique dans **Heart of Paper** a deux niveaux :

- Niveau générique
- Niveau contextuel



Le niveau générique : Esthétique du matériau

Il est constitué principalement de sons de matière papier, carton, et de sons de mécanisme. Cela a pour effet de renforcer les sensations du joueur, et de le convaincre qu'il manipule un univers fait de papier, avec un côté « coffre à jouets » apporté par les éléments mécaniques via des sons de papier et carton sous toutes ses formes : froissé, frotté, déchiré, et des sons de mécanismes : clic d'horlogerie, boîte à musique.

Les sons de papier sont utilisés pour tout objet se déplaçant le long d'une rainure, objet se dépliant, et rotation de disque ; les sons de mécanismes sont utilisés pour les tirettes et clics de disque.

Cet ensemble de feedbacks est désigné sous le terme « générique » puisqu'il ne change pas de contenu d'un niveau à l'autre afin d'assurer une homogénéité au travers l'ensemble du livre.

Le niveau contextuel : Une direction artistique au fil du livre

Cette couche sonore est quant à elle calquée sur le niveau en cours, et s'adaptera donc en fonction du contexte dans lequel le joueur évolue. Parallèlement au traitement visuel, le traitement de l'ambiance sonore évoluera tout au long du livre. Il est constitué :

- de sons faisant référence aux représentations des objets de papier.
- de sons de synthèse.
- de stingers musicaux (ponctuation sonore).
- d'éléments personnalisant les personnages (feedbacks distinctifs).

La couche sonore contextuelle servira à rendre vivant un univers qui ne l'est pas à l'origine, celui du pop-up et de la maquette en papier.

L'exemple du chapitre 2

- Le joueur est amené à manipuler un personnage et à le tirer le long d'une rainure. C'est ici l'occasion de ne pas surenchérir sur un feedback typé « papier » et de voir l'opportunité de personnifier le personnage à travers un son spécifique.

Personnage	Son de pick-up	Son de déplacement	Son de collision
Détective	Son standard	Débit de bruits de pas anormalement élevé	Son standard
Magicien	Petit rire mesquin	Son de carillon (scintillement)	Son standard
Danseuse	Son standard	Bruits de pas (talons)	Exclamation



Technique

Qu'est-ce qu'un RTPC?

Le *Real Time Parameter Control* est un outil primordial en sound design puisqu'il permet de contrôler en temps réel un ou plusieurs effets à partir de la variation de n'importe quel paramètre du jeu. Le moteur du jeu envoie à chaque frame la valeur de cette variable au moteur son, qui appliquera en temps réel la variation sur l'effet défini au préalable.

Exemple

Dans un jeu de voiture, le son du moteur change en fonction de l'accélération. C'est le taux d'accélération (notre *RTPC*) qui détermine la hauteur du pitch du son du moteur.

Les boucles des sons de papier

A partir des prises de son de papier, le fichier a été découpé en plusieurs samples de 2 secondes. Ils sont ensuite enchaînés les uns aux autres par crossfade.



Ainsi, on obtient une boucle de son de dépliage pouvant durer indéfiniment, quelle que soit la taille de l'objet, et pouvant être interrompue à n'importe quel moment.

Le RTPC de Size

Ce paramètre, allant de 0 à 100, permet de gérer en temps réel le son d'un objet en fonction de sa taille. Ce paramètre va contrôler des effets de pitch et de filtrage.

Prenons l'exemple d'une boîte en carton et d'un rocher glissant sur une rainure. Ces deux objets font deux sons différents et pourtant la source sonore est la même.

Tous les éléments dynamiques du jeu sont gérés par ce seul et unique *RTPC* :

- Les objets le long des rainures, leurs sons de collision.
- Les disques.
- Les plaques pivotantes (*shutters*).

Cela permet de garantir une cohérence et une homogénéité des feedbacks relatifs aux objets dynamiques de la scène, et d'éviter une surabondance de fichiers sonores qui alourdirait les ressources allouées au son.

Les graphiques page suivante représentent les différentes influences du *RTPC* de size sur différents effets.

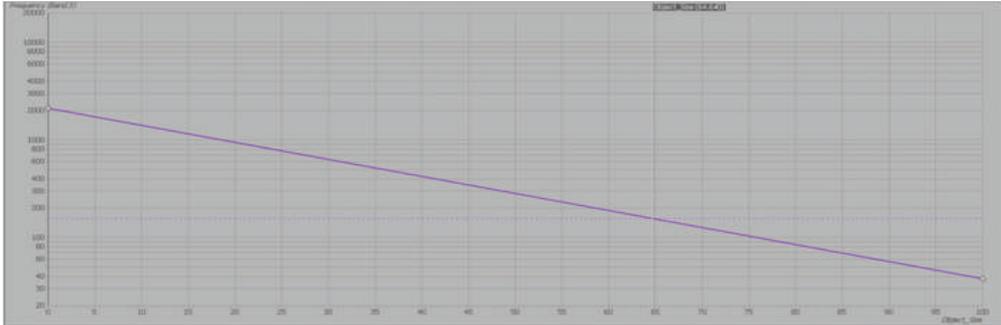




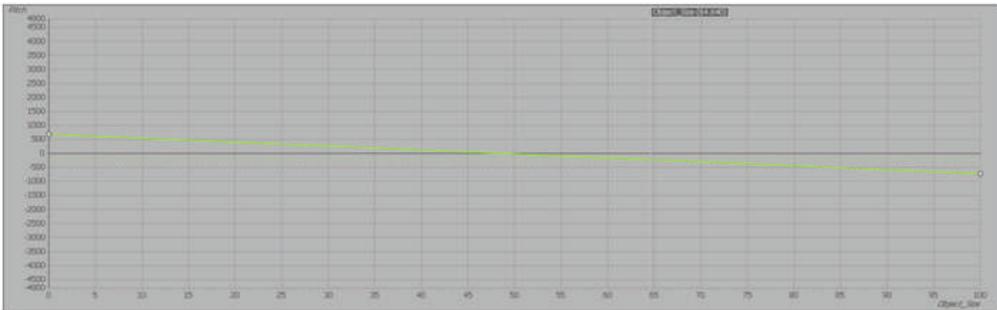
Objet léger



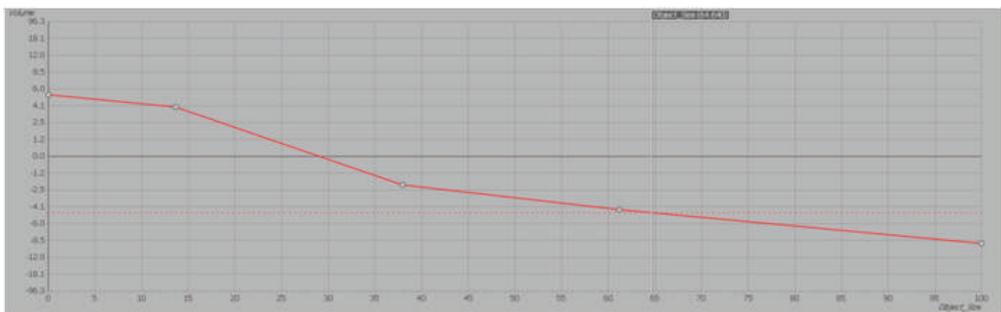
Objet lourd



Dans ces graphiques, la taille de l'objet varie entre 0 et 100. Un High Pass est utilisé de sorte qu'un petit objet n'ait pas autant de fréquences basses qu'un objet très lourd.



Ensuite, on applique un léger pitch de fréquence (en jaune), encore une fois calqué sur la taille de l'objet.



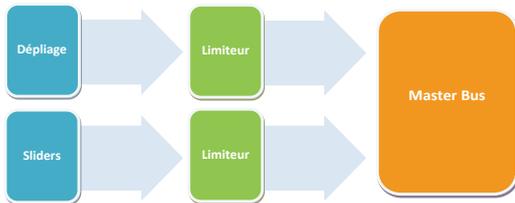
Enfin, la courbe de volume (en rouge) permet de corriger la perte de gain dû à la suppression des fréquences basses sur les objets les plus petits.



La gestion du volume

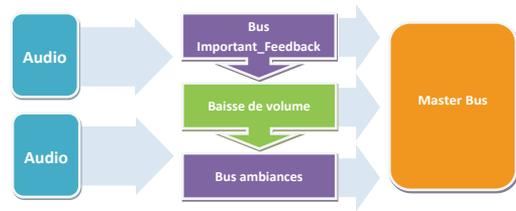
Les différents éléments constituant l'univers sonore sont répartis en plusieurs bus, permettant ainsi d'éviter les phénomènes de masquage et de saturation.

Il peut arriver dans **Heart of Paper** qu'une même tirette contrôle plusieurs objets, qu'ils poppent ou se déplacent le long de rainure (*slider*).



Ainsi les éléments « dépliage » et « sliders » sont répartis sur 2 bus différents avec un limiteur sur chacun d'entre eux (voir schéma). On évite ainsi les cas où tellement d'éléments se cumulent que le volume du signal se multiplie.

Les feedbacks importants pour le joueur sont quant à eux mis en valeur par l'application d'un *ducking* sur un bus spécifique appelé « important_feedback ». Cela se traduit par l'atténuation des autres bus lorsque le bus « important_feedbacks » émet du signal. Les feedbacks concernés par ce bus sont les sons de reward : déblocage de passage, fin du niveau, résolution d'énigme, et aussi les feedbacks d'action impossible pour le joueur (mécanismes bloqués).



Ces précautions permettent de prévenir tous les cas de figure afin d'éviter la surcharge de feedbacks et envisager toutes les situations de gameplay possibles.



Documentation technique

Introduction

Cette partie résume les diverses implications techniques de ce projet. Il met en avant les choix ainsi que les solutions qui ont jalonné la pré-production.

Problématiques

Dans un premier temps, nous allons parcourir les problématiques rencontrées ainsi que leurs solutions. Le but est de donner une vue d'ensemble du travail accompli durant ces derniers mois.

Choix de l'environnement

Avant de pouvoir attaquer le cœur du problème, nous avons dû mettre en place un environnement de travail efficace et adapté à notre équipe et notre jeu.

Le moteur

De manière réaliste, nous avons deux choix pour le poste d'arrière central du projet :

UDK	<ul style="list-style-type: none">• Moteur très complet• Editeur de shaders, d'animation, de squelettes...• Kismet• Communauté active
	<ul style="list-style-type: none">• Très lourd• Pas d'export sur plateformes mobiles• Rendu très «gloss»• Orienté FPS
Unity	<ul style="list-style-type: none">• Simple et malléable• Éditeur de shaders• Rendu neutre• Communauté active• Export multi-plateforme (Web, PC, Mac, Android, iPad/iPhone)
	<ul style="list-style-type: none">• PhysX accessible en partie seulement• Gestion de version difficile à intégrer• Pas de gestion de l'événementiel

Nous nous sommes assez naturellement orientés vers *Unity 3D* comme base. Ce choix a été confirmé quelques semaines plus tard lorsque nous avons parlé pour la première fois de plateforme mobile.



La gestion des sources

Le moteur choisi, nous nous sommes penchés sur le gestionnaire de sources que nous voulions mettre en place.

Dans un premier temps, nous avons, avec l'aide des autres groupes, testé plusieurs possibilités dont l'outil fourni par *Unity* : l'*Asset Server*.

Nous nous sommes rendu compte que les divers objets utilisés dans une scène *Unity* étaient finalement stockés en tant que fichiers binaires. Dès lors, que l'outil de gestion de sources soit interne ou externe à *Unity* importait peu puisqu'aucun ne pouvait fusionner ou manipuler ces fichiers.

Au vu de la pauvreté des fonctions de l'*Asset Server* tant aux niveaux des permissions que sur les outils de fusion, historique, etc, nous avons décidé de prendre un outil externe utilisé par l'industrie actuelle.

Après la comparaison de *Mercurial*, *Perforce*, *Git* et *SVN*, nous avons penché vers *Perforce* car il est très utilisé dans le monde du travail. Ceci nous a permis de nous familiariser avec un outil que nous allons sûrement rencontrer en dehors de l'école.

Dans le cadre d'une production où les coûts doivent être réduits, nous pensons que *SVN* ou *Mercurial* seraient des excellents choix.

L'éditeur

Dès le départ, nous avons commencé par utiliser la dernière version de *Visual Studio*.

Nous avons aussi installé *MonoDevelop* qui est un outil mieux intégré à *Unity*. Il permet entre autres de faire du debugging durant l'exécution.

Physique ou procédural ?

Dès le départ du projet, nous nous sommes attaqués à la représentation des mécanismes du pop-up. Nos premières recherches se sont orientées vers une version entièrement physique de l'environnement.

Mais nous avons été confrontés à une instabilité dans les mouvements et les stabilisations. Comme nous voulions pouvoir arrêter un objet au milieu de son dépliage, nous avons des problèmes de forces et de tremblements.

Dans un deuxième temps, nous avons tenté de stabiliser nos structures en utilisant des joints et des bones. Mais nous nous sommes rendu compte que le travail demandé par la création des structures sous *3DStudio Max* était partiellement perdu lors de l'import dans *Unity 3D*. Seul les meshes et leur hiérarchie étaient conservés.

Comme aucune de ces solutions ne nous convenait de par leur investissement en temps ou leur relative instabilité, nous avons tenté de mettre en place un système basé sur un pourcentage d'ouverture.



Cette solution se base sur l'observation des mécanismes que nous avons pu classer en deux catégories.

D'un côté nous avons les « esclaves » qui représentent tous les objets pouvant se déplier, et de l'autre les « managers » qui définissent le pourcentage de dépliement.

Nous avons ensuite décrit les diverses transformations pouvant représenter les mécanismes de bases comme les rotations, les changements d'échelle, le déroulement d'une animation, le déplacement le long d'un chemin, etc.

En partant de ces deux classes de bases, nous avons pu décrire toutes les formes de mécanismes utilisées par nos level designers.

Gameplay événementiel

A partir de Noël, nous nous sommes confronté à une deuxième grosse réflexion. Nous voulions pouvoir fournir aux level designers des outils leur permettant de mettre en place la logique de jeu par eux-mêmes.

Aux cours des discussions, il s'est avéré que les level designers voulaient pouvoir définir quand un Objet ou un Manager atteignait une certaine zone, un certain pourcentage ou encore une certaine orientation, puis assigner à ce déclencheur une série d'événements permettant de changer l'environnement.

Nous avons réfléchi à plusieurs manières d'implémenter ce type d'architecture.

Couple Trigger-Event

L'idée la plus simple est de fournir un objet permettant de représenter les déclencheurs et un autre pour les événements. Ainsi, les level designers peuvent répartir leurs événements sur les divers éléments de leur scène.

Très simple d'approche, cette solution comporte un gros problème de visualisation de ce qui est actuellement en place. Il est difficile de se souvenir où sont placés les déclencheurs et quels sont les événements qu'ils libèrent.

Manager d'événement

L'idée était d'utiliser un gestionnaire d'événements propre à la scène qui regrouperait tous les déclencheurs ainsi que leurs événements.

Ainsi il est beaucoup plus simple de retrouver un événement ou de parcourir les déclencheurs déjà mis en place puisqu'ils sont tous au même endroit.

Il reste bien évident que cette solution risque tout de même de devenir difficilement lisible lorsque le nombre d'événements ou de déclencheurs sera élevé.

Ménage à trois

Un autre problème de la solution précédente apparaît lorsque l'on veut attendre que plusieurs déclencheurs soient actifs pour lancer un événement.

Nous avons donc imaginé une solution comprenant les types communs d'opérateur (*AND*, *OR* et *XOR*).

Les opérateurs agiront sur le même



principe que les déclencheurs mais en lieu et place des objets de la scène, ils prendraient les déclencheurs en entrée.

Kismet

Dans le but de simplifier la mise en place de logique complexe ainsi que pour améliorer la lisibilité de l'état actuel de la logique, nous avons pensé à développer un outil externe de type *Kismet* que l'on trouve dans l'*UDK*.

L'idée générale était de générer des fichiers *XML* depuis *Unity* décrivant les relations « déclencheurs / opérateurs / événements » ainsi que les objets présents dans la scène.

Ensuite, dans un outil externe, le level designer pourrait modifier ou créer de nouvelles interactions entre objets, déclencheurs, opérateurs et événements.

Une fois les relations mises en place, l'outil génère un nouveau fichier *XML* décrivant la logique de jeu. Puis nous l'importerions directement dans *Unity* afin de recréer les interactions définies par les level designers.

Et alors ? La solution ?

Finalement, en discutant avec les game designers, nous avons remarqué que la relative complexité due à l'éparpillement des déclencheurs ou événements n'était pas critique. On peut, grâce à la hiérarchie très fournie des Objets *Unity*, répartir les déclencheurs, opérateurs et événements sur les objets de manière logique et plutôt claire.

Dans le même temps, nous avons ajouté une représentation graphique se positionnant sur les objets possédant un *trigger*

afin de les retrouver plus facilement. Et dans le but de faciliter la mise en place des *triggers* et des événements qui leur sont liés, nous avons aussi modifié l'éditeur de propriétés de *Unity* pour les *triggers*.

La solution du *Kismet* reste pour nous la meilleure du point de vue utilisabilité et efficacité. Elle est cependant très perméable aux erreurs, que cela soit au niveau de la génération ou de la lecture du *XML*, de l'interprétation des données ainsi que lors de diverses modifications comme la disparition d'un objet, les changements de nom ou de propriété, etc.

L'équipe pédagogique nous a clairement déconseillé d'utiliser cette voie.

Caméra

Apparue très tard dans la production, la gestion de la caméra s'est faite en utilisant les divers Managers et Objets déjà en place pour représenter les mécanismes du pop-up.

Nous n'avons qu'ajouté une description de scène permettant de définir :

- Un chemin pour atteindre la scène suivante
- Une position ou un objet à regarder

Puis nous avons mis en place un événement particulier permettant de changer de scène.



Architecture

Notre architecture se base sur 3 groupes de classes différentes :

- Les managers et les objets, représentant les mécanismes.
- Les *Triggers* et les *Events* permettant de définir la logique de jeu.
- Le *BookManager* et les *Levels* décrivant le comportement du livre et le chargement de niveau.

Dans un souci de clarté, nous avons préféré limiter au maximum les informations apparaissant dans les diagrammes de classe ci-dessous. Le but n'est pas de décrire précisément notre code, mais bien de donner une vue d'ensemble de notre architecture et de ses composants.

Managers et Objects

Cette partie est le cœur de notre architecture. Elle est la plus ancienne et celle qui s'est le plus agrandie.

Sur la gauche, nous avons les *PercentManagers* qui permettent de gérer les *PercentObjects* sur la droite.

Les *PercentManagers* sont composés de

- *PullTab* : Tirette.
- *SliderDrag* : Déplacement à la souris d'un objet sur un rail.
- *Wheel* : Roue ou disque.

Le terme *Add* désigne un *Manager* qui ne se bloque pas lorsqu'un de ses *PercentObjects* est bloqué, *Toggle* permet de passer

de 0 à 1 et inversement en une seule étape, et *OneWay* désigne une roue qui ne tourne que dans un seul sens.

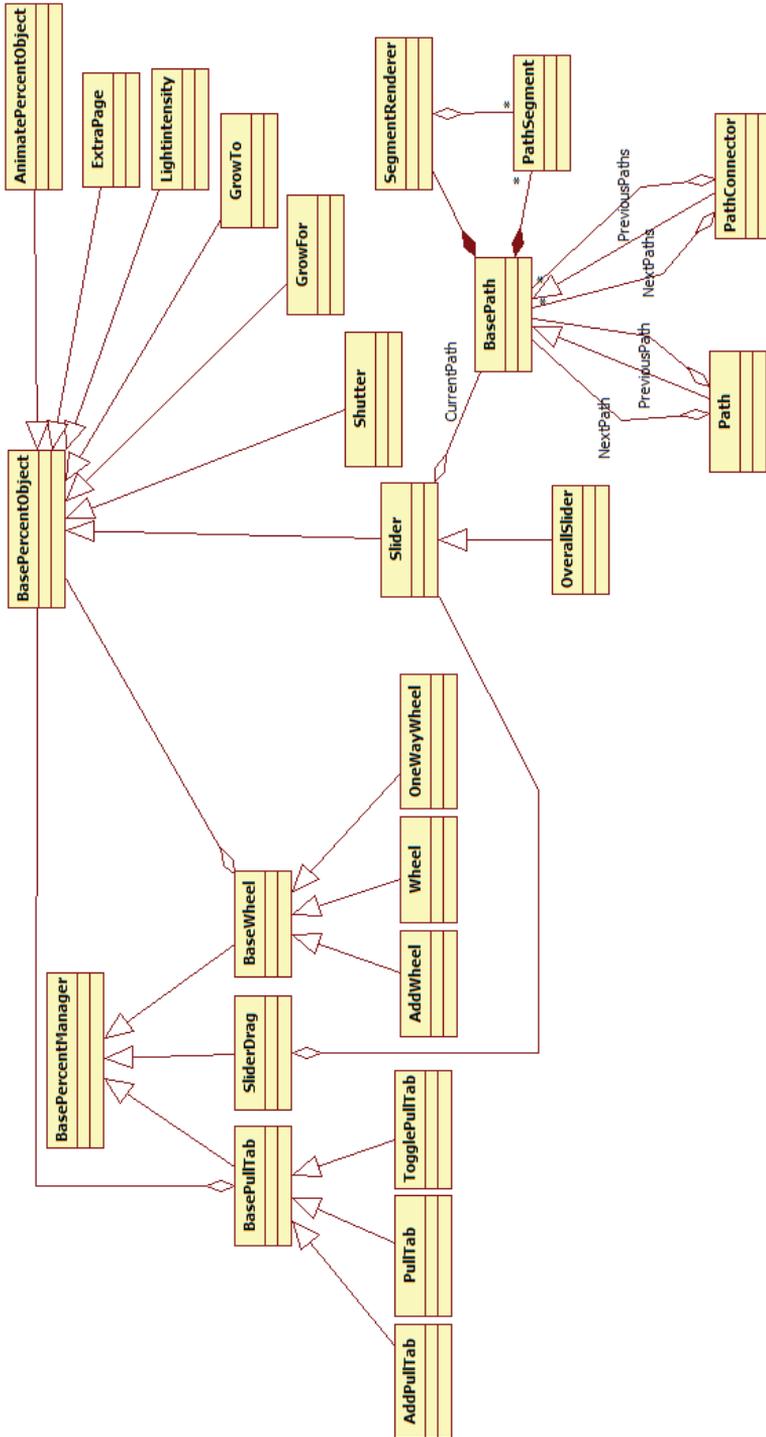
Pour les *PercentObjects*, nous avons :

- *Slider* : Représente le déplacement d'objet le long d'un *Path*. L'*OverallSlider* permet de glisser sur plusieurs *Paths* comme s'ils n'étaient qu'un seul *Path*.
- *Shutter* : Décrit le comportement d'un volet.
- *GrowFor* : Permet de modifier la position, l'orientation et/ou l'échelle d'un objet avec un delta donné.
- *GrowTo* : Permet de modifier la position, l'orientation et/ou l'échelle d'un objet jusqu'à une valeur donnée.
- *LightIntensity* : Utilisé sur une lampe, il permet de faire varier l'intensité de la lumière.
- *ExtraPage* : Permet de faire apparaître une partie de page supplémentaire. Utilisé pour agrandir la surface du niveau lorsque l'énigme avance.
- *AnimatePercentObject* : Couplé à une animation, il déroule l'animation de 0 à 100% en fonction du *Manager*.

Une sous-partie vient s'ajouter aux *PercentObjects* et permet aux *Sliders* de savoir où aller:

- *Path* : Décrit un chemin par des points. Implémente les interpolations linéaire et Bézier.
- *PathConnector* : Permet de créer un chemin avec plus d'un chemin précédent ou suivant.
- *SegmentRenderer* : Permet de créer un mesh à la volée respectant le parcours du chemin.
- *PathSegment* : Stocke les informations relatives au segment comme le point de départ et d'arrivée, les points intermédiaires, sa visibilité, etc.





PercentManagers et PercentObjects



Triggers et Events

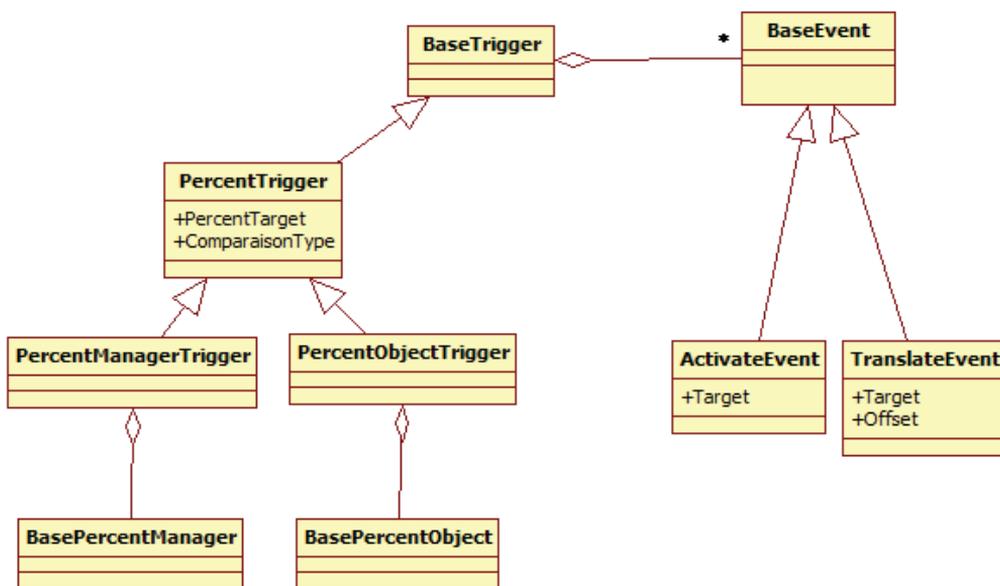
A l'heure de l'écriture de ce document, nous n'avons pas encore stabilisé l'implémentation de cette partie. Nous pensons qu'elle va encore devoir évoluer afin de répondre correctement aux requêtes des level designers.

A gauche, nous avons les *Triggers* qui permettent en fonction de l'état d'un objet de lancer les *Events* sur la droite.

Actuellement nous n'avons qu'une sorte de *Trigger* qui permet de récupérer le pourcentage d'un *PercentManager* ou d'un *PercentObject*.

Nous possédons deux *Events* différents :

- *Activate* : Permet d'activer/désactiver un *PercentObject*, *PercentManager* ou un *Path*
- *TranslateEvent* : Permet de téléporter un objet à une position locale ou globale



BookManager et Levels

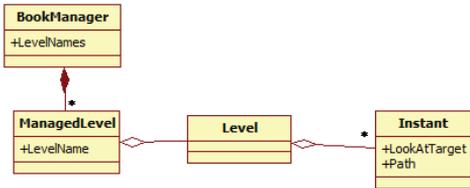
Le *BookManager* a pour rôle de choisir quels sont les niveaux à charger en mémoire et quand, puis de les libérer lorsqu'ils ne sont plus utiles. Il permet aussi de gérer l'ouverture du livre, de récupérer les objets qui doivent se déplier durant l'ouverture et de les diriger.

Lors de la création des niveaux, nous maintenons à jour une liste de chaînes de caractères représentant les noms des niveaux à charger et dans quel ordre. Le *ManagedLevel* permet, à partir d'un nom, de retrouver le niveau correspondant et de le charger dans la scène principale.

Le *Level* est utilisé dans la *Scène Unity* par les level designers pour définir le contenu du niveau courant. Il contient les *Instants* qui représentent les divers points d'intérêt pour la caméra et les chemins à utiliser pour la mise en scène.

Références

Notre principale source d'information provient du site officiel d'*Unity*. Que ce soit le forum ou la documentation technique présente en ligne, nous y avons trouvé la plupart de nos réponses.



BookManager et Levels



Ergonomie

A propos

En dépit de son développement sur PC, **Heart of Paper** est avant tout un jeu destiné à la plateforme tactile (iPad) et amène ainsi toute une problématique liée aux interfaces tactiles.

Nos intentions sur ce projet sont de proposer un sentiment d'immersion aux joueurs au cours de l'expérience de jeu, ainsi qu'une navigation simple et intuitive.

Pour ce faire, nous nous appuyons sur la littérature existante. De plus, de nombreux tests seront nécessaires pour répondre aux besoins ergonomiques de ce projet.

Démarche

Notre démarche s'est déroulée en plusieurs étapes :

- **Focus group & analyse de l'existant**
Analyser les attentes et opinions des futurs utilisateurs et positionner **Heart of Paper** par rapport au marché et aux produits similaires.
- **Design Interface**
Créer une interface en adéquation avec le concept.
- **Playtests**
Intervenir auprès du public cible pour valider les choix de conception et prévenir l'apparition de problèmes.

Les diverses étapes de la démarche ergonomique ont été réalisées en lien avec les game designers, les programmeurs et la graphiste pour plus d'efficacité.

Focus group & Analyse de l'existant

Dans un premier temps, il faut rappeler que le projet **Heart of Paper** est basé sur la rencontre entre deux concepts : celui du livre pop-up et du jeu vidéo.

Cette particularité le pose comme un projet original, n'ayant pas d'équivalent sur le marché, et nécessitant donc l'évaluation des attentes du public cible.

Le mélange du jeu vidéo et du livre pop-up laisse perplexe au premier abord, mais rapidement nous sommes arrivés à confirmer certains points de game design au cours des focus groups. Nous avons retenu trois points qui paraissaient centraux pour le joueur : l'interactivité des décors, l'aspect de livre dans le jeu importante pour le ressenti, et la notion d'échelle.

De plus, cela nous a permis d'obtenir de nouvelles pistes de réflexion, comme la présence de différents types de gameplay à l'intérieur d'un même jeu, et l'importance du ressenti tactile du pop-up.

Le focus group nous a donc permis d'établir une liste de recommandations à suivre lors de la conception du projet.

En parallèle au focus group, nous avons mené une analyse sur des jeux présentant certaines caractéristiques comparables à **Heart of Paper**. Comme il n'existe pas de



produit similaire, nous avons basé notre analyse sur deux jeux de type puzzle-game possédant des mécanismes de résolution d'énigmes : *Professeur Layton et la boîte de Pandore* et *Hotel Dusk : Room 215*.

Le problème principal rencontré durant la conception se situe autour de la compréhension des énigmes : comment faire comprendre au joueur ce qu'il doit faire sans le noyer sous un flot d'informations textuelles ? Si *Professeur Layton et la boîte de Pandore* propose des énigmes, elles sont cependant décontextualisées, contrairement à *Hotel Dusk* où les mécanismes sont intégrés au déroulement d'une histoire.

La comparaison de notre produit à ses aînés nous a aidés à faire ressortir des pistes à suivre concernant la visibilité des éléments d'interaction, l'intégration des mécanismes au décor, et la visibilité des liens mécanismes-éléments.

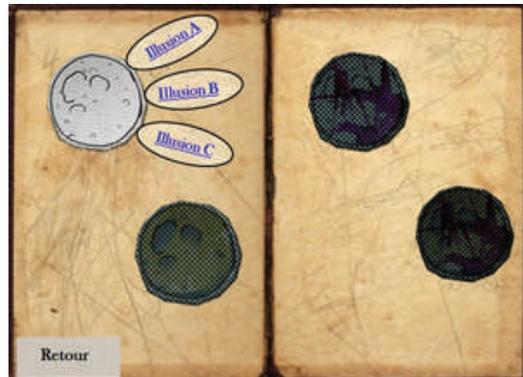
Le focus group et l'analyse de l'existant ont permis d'établir des guidelines pour orienter les game designers en phase de conception.

Interface

Avec le livre pop-up au centre du concept, nous avons choisi de poser le joueur face à un livre. De ce choix a découlé la décision de proposer une interface de menu simple et un HUD invisible afin de permettre une meilleure immersion au joueur.

Interface de menu

Le menu doit être simple et utilisable afin que le joueur ne perde pas de temps à comprendre le fonctionnement du menu avant même celui du jeu. Nous avons choisi de nous rapprocher le plus possible de la table des matières d'un livre pour rester cohérents avec le reste du jeu et donc de proposer une navigation thématique.



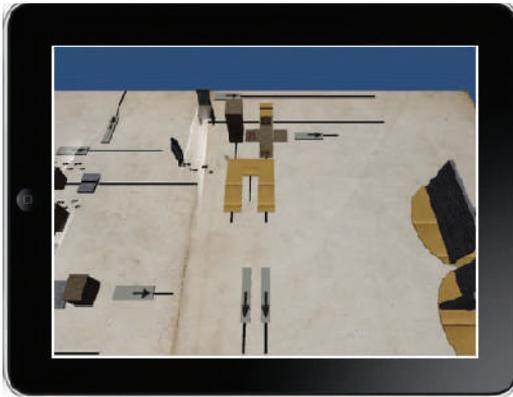
Exemples de navigation dans un menu type



HUD

Notre gameplay étant basé sur l'immersion et non sur le scoring, il paraissait évident de proposer un HUD invisible.

En effet, toute forme d'indication sera intégrée aux décors ou à la scène présentée.



Playtest

Dans un troisième temps, nous nous sommes inscrits dans une démarche de tests itératifs.

Dès les premières phases de conception amorcées, les premiers tests ont été lancés sur prototype papier (maquette de livre pop-up), puis sur le prototype alpha et enfin le prototype beta. Nous avons bénéficié d'une fidélisation des testeurs qui revenaient régulièrement participer aux playtests. Notre échantillon était composé, pour chaque test, de 10 sujets hommes et femmes âgés de 25 à 35 ans.

Chaque séance suivait un plan de playtest type. On déterminait le point de questionnement à étudier et l'on construisait le matériel en conséquence. Une fois le matériel créé, on procédait au recrutement de sujets correspondant à la cible du projet, auxquels on proposait un rendez-vous pour tester le jeu dans nos locaux.

Le sujet était amené à tester le prototype et à répondre aux questionnaires, grilles et autres formes de questions qui lui étaient posées avant d'être débriefé par l'expérimentateur.

Ce processus d'itération nous a permis de répertorier rapidement les problèmes et de les régler d'une version à l'autre.

Didacticiel

Notre prototype se déroule au niveau du chapitre 2 et ne présentera pas de didacticiel. Cependant il est évident que la version finale du jeu en comprendra un.

Le tutoriel tel que nous l'imaginons se trouvera intégré au jeu à partir du début du premier chapitre. Il sera construit sur deux niveaux. Tout d'abord, le joueur se trouvera face à des énigmes simples lui présentant les différents éléments à manipuler (tirette, disque et personnage ou objet sur rainure) afin qu'il assimile quels sont les objets avec lesquels il pourra interagir dans le jeu. Dans un second temps, il devra résoudre des énigmes simples faisant appel aux trois briques de gameplay (parcours de chemin, création/modification de décors et énigmes contextuelles) auxquelles il sera confronté dans le jeu.

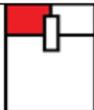
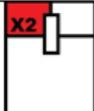
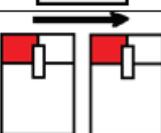
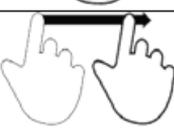
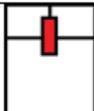
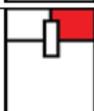
Le didacticiel sera défini étape par étape suivant une progression dans l'ajout des mécanismes et des briques.



Contrôles

Notre parti pris de développer le projet pour le marché iPad nous a obligé à penser nos contrôles en fonction de ces contraintes, et ce même pour la version PC.

Tous les contrôles ont été étudiés pour être réalisé avec une seule main : toutes les manipulations se feront uniquement avec la souris pour la version PC, et avec la main pour la version iPad.

Action	PC	IPAD
Sélectionner		
Valider		
Tirer / Pousser		
Tourner un disque		
Zoom Caméra		
Manipulation Caméra		

Equivalence des contrôles PC et iPad



Production

Avant-propos

Avant de rentrer dans les détails liés à la gestion de projet de **Heart of Paper**, il est nécessaire de situer nos postulats de départ :

- Nous avons développé la pré-production du jeu durant nos 6 mois à l'ENJMIN en tant qu'étudiants. Le planning de pré-production est donc adapté au planning qui nous a été imposé par l'école car nous devons développer un prototype de jeu en parallèle de nos cours. Nous n'avons pu obtenir des journées en équipe complète qu'à partir de la fin février.
- Pour des raisons techniques, nous avons développé le prototype du jeu sur PC via *Unity*, faute du matériel nécessaire (plusieurs Mac et iPad) au sein de l'école, mais le gameplay est pensé pour l'iPad.
- Nous proposons un planning de production fondé sur les réalités actuelles du marché du jeu vidéo qui propose en majeure partie des développements de jeux iPhone et iPad développés en petite équipe (4-5 personnes maximum, profils double spécialité) sur une durée maximum de 6 mois et des budgets de production ne dépassant pas les 100 000 euros.
- Le budget marketing n'est ici pas pris en compte car nous partons du principe que nous ne réaliserons pas de communication « physique » (affiches, goodies, communication papier...), la taille de notre équipe, le type de jeu (dématérialisé) et les coûts de réalisation étant inappropriés au regard de nos ambi-

tions de réduction des coûts. De plus, si le jeu rencontre le succès escompté, les reviews sur les sites internet de presse spécialisés, les blogs ainsi que le système de note sur l'Appstore permettront d'eux-mêmes d'accorder une bonne communication sur le jeu.

Cible et plateforme

Heart of Paper est un jeu qui s'adresse aux détenteurs d'iPad. Il s'adresse donc principalement aux hommes âgés de 30 ans et plus, ayant un emploi et jouant régulièrement sur l'iPad (2h/jour en moyenne) et utilisant la tablette pour les loisirs (*source : Kaelisoft.fr*).

Notre cible principale est fondée sur l'acheteur principal. Pour autant, ce n'est pas forcément la cible principale de joueurs. En effet, étant donné que l'iPad cible davantage les ménages qu'un profil spécifique, notre cible secondaire s'apparente à la famille.

Par exemple, **Heart of Paper** peut être joué seul par un adulte, ou acheté par ce même adulte pour permettre à son enfant ou sa compagne de jouer au jeu seul ou à deux.

Classification



Heart of Paper est accessible à un public 7+ (selon la classification PEGI), compte tenu de la simplicité de son gameplay et de la nature des thèmes abordés.



iPad

Heart of Paper a été développé sur PC pour des raisons techniques dues à un manque de matériel à l'ENJMIN. Cependant, le jeu a toujours été pensé pour un développement iPad et le planning de production prend en compte le portage sur iPad de la partie du jeu développée en pré-production à l'ENJMIN.

Étant donné l'implication de développement sur la plateforme PC et le ciblage du jeu, il serait cependant envisageable de proposer le jeu sur *Steam* afin d'augmenter les recettes de vente du jeu sur le même modèle (présenté ci-dessous) puisque les versements d'intérêts auprès de *Steam* sont semblables à ceux retenus par *Apple* (30% de parts).

Quelques jeux traitant du pop-up ont été portés sur iPad mais relèvent davantage de l'expérience interactive que du jeu vidéo (*Cardboard Castle*, *Raiponce de Grimm*) et même parfois du livre interactif (*Alice*). De plus leurs graphismes restent faits en 2D.



Raiponce de Grimm, Ideal Binary



Alice for the iPad, Atomic Antelope



Cardboard Castle, Bulkypix

Mois 1	Mois 2	Mois 3	Mois 4
<p><i>Pré-production</i></p> <ul style="list-style-type: none"> • Portage iPad chapitre 1 • Lancement du niveau 1 sur l'Appstore • Réflexion concept sur le LD des chapitres 2,3 + épilogue • Level design papier • Développement des outils gameplay • Recherche artistique pour les chapitres 2 + épilogue 	<p><i>Micro-production 1</i></p> <ul style="list-style-type: none"> • Level design chapitre 2 • Modélisation chapitre 2 • Animation chapitre 2 • Débug & polish • Lancement du niveau 2 sur l'Appstore 	<p><i>Micro-production 2</i></p> <ul style="list-style-type: none"> • Level design chapitre 3 • Modélisation chapitre 3 • Animation chapitre 3 • Débug & polish • Lancement du niveau 3 sur l'Appstore 	<p><i>Micro-production 3</i></p> <ul style="list-style-type: none"> • Level design épilogue • Modélisation épilogue • Animation épilogue • Débug & polish • Lancement du chapitre 4 (épilogue) sur l'Appstore • Lancement du pack complet (tous les chapitres) sur l'Appstore



Budget

Heart of Paper est un projet 3D pour iPad. Contrairement à *Infinity Blade* (*Epic Games*) qui est sorti récemment et qui a attiré l'attention des joueurs sur iPad, peu sont les jeux sur cette plateforme à être proposés en 3D. Les plus gros succès (*Angry Birds*, *Cut the Rope*, *Plants vs. Zombies*...) restent des jeux 2D développés par des petites équipes sur une courte durée (6 mois tout au plus).

Nous sommes partis de ce modèle pour établir le budget de production de **Heart of Paper**.



Cut the rope, Chillingo Ltd.



Plants vs Zombies, Popcap

Planning de production

Un mois de pré-production et trois mois de production permettraient de poser le level design des niveaux restant à développer pour terminer le jeu.

Au cours des trois mois de production, nous suivrions des micro-productions du jeu afin de proposer les « chapitres » du jeu par épisodes.



Infinity Blade, Epic games



Angry birds, Chillingo Ltd.



Plan de charge

Corps de métier	Tâche	Niveau nécessaire	Type de contrat	Nb employé	Coût réel chargé mensuel	Nombre de mois travaillés total	Total sur le projet
Gestion de projet	Management + game design	Expérimenté	CDD	1	4 200,0 €	4	16 800,0 €
Game design	Level design	Expérimenté	CDD	1	4 200,0 €	4	16 800,0 €
Graphismes	Graph	Expérimenté	CDD	1	4 200,0 €	4	16 800,0 €
Développement	Gameplay	Expérimenté	CDD	1	4 200,0 €	4	16 800,0 €
Sound design	Sound design	N/C	Outsourcing	N/C			5 000,0 €
TOTAUX				4	16 800,0 €	16	67 200,0 €

Investissements prévus

Désignation	Coût	Qté	Coût total
ipad	499,0 €	1	499,0 €
Licences IOS	99,0 €	3	297,0 €
Licences Unity	1 500,0 €	2	3 000,0 €
Licence Visual	122,2 €	1	122,2 €
Licence 3DS	422,2 €	1	422,2 €
Licence Photoshop	222,2 €	1	1 027,3 €
Ordinateurs	133,3 €	4	533,3 €
Totaux investissements prévus			5 901,1 €

Dépenses mensuelles

Types de frais	Coûts mensuels
Frais de structure	6 720,0 €
Frais généraux Commercial	6 720,0 €
Direction	N/C
Marketing	N/C
Total dépenses sur la production	13 440,0 €

Budget total de production

Nb de mois	4
Qté de mois/jr	16
Coût total	86 541 €

Point mort

Prix de vente HT	1,99 €
Prix de vente TTC	1,7 €
Prix de vente après royalties	1,3 €
Seuil de rentabilité sans FAUV	94 924
Seuil de rentabilité avec FAUV	47 462

Crédits d'impôts

Type	%	Récupérable
CR	Non	
JEI	Non	
CIIV	Non	
FAUV	50%	60 746,7 €



Budget total de production

Les salaires prennent en compte une marge de risque de glissement du temps de la production. Cette marge s'élève à 20%, permettant une marge de négociation correcte si nous devons discuter du coût du projet avec un éditeur (base de salaire = 3500 euros + 20% de marge). En incluant cette marge de 20%, les unités à vendre pour atteindre notre point mort s'élèvent aux environs de 90 000 unités, mais sans cette marge, ce chiffre s'élève à 80 000 unités. Nous partirons donc de cette base dans le descriptif suivant.

Le budget nécessaire à la production du reste du jeu est d'environ 80 000 euros si l'on part sur une production du jeu en 4 mois avec 4 personnes dont le profil est expérimenté dans le développement de jeu vidéo sur iPhone et iPad.

Afin d'être rentable sur les coûts de production (budget total de production), il faudra vendre environ 80 000 applications du jeu à 1,99 euros.

Étant donné que les jeux iPad connaissant un succès suffisamment rentable pour leurs investisseurs tournent autour des 20 000 unités vendues, nous sommes partis de là pour établir si notre point mort de vente calculé (cf tableau ci-contre) était réaliste par rapport au marché actuel.

En estimant que nous vendrions autour des 20 000 applications pour chaque épisode à 1,99 euros, notre total d'unités vendues avoisinerait les 80 000 applications vendues et nous permettrait d'amortir les coûts totaux de production des 4 épisodes.

Notons que notre jeu étant éligible pour le fond d'aide au jeu vidéo (FAJV) pour l'aide à la création de propriétés intellec-

tuelles. Notre point mort reviendrait dans ce cas à la vente de 40 000 unités, ce qui est largement faisable lorsque l'on s'appuie sur le postulat annoncé précédemment.

Business model

Basé sur la stratégie marketing de *Chillingo* pour les épisodes d'*Angry Birds*, notre stratégie marketing consisterait à proposer chaque niveau de notre jeu (4 en tout) en épisodes téléchargeables au prix de 1,99 euros l'unité. Ce prix est suffisamment peu cher pour inciter à l'achat impulsif et donner envie à un joueur iPad lambda d'acheter un des épisodes pour tester le concept et voir ce qu'il a à proposer. De même, il est suffisamment peu élevé pour donner envie à un joueur désireux de jouer au jeu de se procurer les autres épisodes.

Afin de donner entière satisfaction aux deux types d'acheteur de notre jeu (ceux désireux vraiment de jouer à notre jeu, et ceux désireux « tester pour voir »), notre stratégie ne concerne pas seulement le prix de vente mais également le contenu qui sera proposé.

L'iPad 2 propose de relier sa tablette à sa télévision via un câble HDMI et promet des interactions avec le gyroscope de l'appareil : en mettant en avant les graphismes HD de notre jeu et la compatibilité de certaines actions avec le gyroscope, nous pourrions justifier l'intérêt d'achat de notre jeu en mettant les innovations majeures de l'appareil au cœur de notre gameplay.

Comme expliqué plus tôt, nous proposerions plusieurs épisodes de notre jeu au fur et à mesure du développement des chapitres du jeu. Chaque épisode serait vendu à 1,99 euros excepté l'épilogue (cha-



Octobre	Novembre	Décembre	Janvier	Février	Mars
<p>Le pop up</p> <ul style="list-style-type: none"> • Réflexion sur le pop up • Focus group • Prise en main des outils • Intégration éléments de gameplay • Tests animation et modelling 	<p>Le concept</p> <ul style="list-style-type: none"> • Prototypage des mécanismes pop up • Refonte du GD • Design avatar 	<p>La refonte</p> <ul style="list-style-type: none"> • Refonte du pitch (7-3-1) • Proposition du pitch sur l'illusion • Présentation jury #1 (présentation du gameplay) 	<p>Le pitch</p> <ul style="list-style-type: none"> • Validation du pitch • Prototypage énigme 1 • Test sound design • Test ambiance graphique 	<p>Le prototype</p> <p>FPP</p> <ul style="list-style-type: none"> • Présentation jury #2 (prototype + DA) • Prototypage énigme 2 • Modélisation/anim gameplay • Prototypage du menu • Planification de la production • Rédaction gamedoc 	<p>La release</p> <p>α</p> <ul style="list-style-type: none"> • Animation de début et fin du prototype • Prototypage énigme 3 • Master candidature (build + gravure) • Présentation jury #3 (vente du projet en conditions « réelles »)



pitre 4) étant donné qu'il est plus court (un seul niveau). Il serait vendu 0,99 euro.

A l'issue de ces 4 mois, nous proposons le jeu complet sur l'Appstore dans un pack au prix de 4,99 euros. Cela permettrait d'obtenir plus de ventes que prévues à l'origine, et le cas échéant, de rattraper les ventes perdues s'il y en a pendant la production.

Post-mortem

Le projet

Au cours des 6 mois de pré-production à l'ENJMIN, la répartition des phases de développement s'est effectuée en trois blocs :

- Concept du jeu
du 4 octobre au 23 décembre 2010.
- Pré-production du jeu
du 3 janvier au 11 février 2011.
- Production du prototype de jeu
du 13 février au 25 mars 2011.

Pour un souci de clarté, voici le découpage exact de nos 6 mois de travail à l'ENJMIN (ci-contre).



Concept du jeu

Avant de rentrer dans une phase de réflexion en équipe, le cœur du projet reposait sur le thème du livre pop-up. Nous sommes partis de cette piste pour élaborer le concept et le gameplay du jeu.

La première version du jeu s'apparentait à un jeu d'action-aventure/énigmes 3D où le joueur incarnait un avatar qui devait détruire et reconstruire un décor pour avancer de tableau (des scènes représentées sur les doubles pages d'un livre) en tableau. Le joueur devait donc faire avancer son personnage pour tourner les pages du livre (et donc terminer un niveau) et développer la narration de lui-même.

Après un premier point projet avec l'équipe pédagogique à la mi-novembre, nous avons décidé de reprendre le pitch de zéro puisque nous avons perdu le thème du pop-up pour faire un jeu de plateforme. Le pop-up était devenu un simple décor et ne faisait plus partie du gameplay. A partir de ce constat, nous avons repensé le gameplay afin de le recentrer autour des interactions que l'on peut trouver dans les livres pop-up.

Notre concept final, celui qui vous a été soumis tout au long de cette documentation, fut validé par l'équipe à la rentrée des vacances de décembre, le 3 janvier 2011.

Pré-production du jeu

Une fois le concept posé, la pré-production put se lancer. Il s'agissait de réfléchir au scénario du jeu, à la charte graphique à présenter lors du point jury #2 et à l'enchaînement des briques de gameplay en posant un level design papier pour commencer à prototyper le prototype de mars.

Production du prototype de jeu

Le point jury #2 passé, nous nous sommes attelés à la production du prototype final de notre jeu. Les phases itératives se sont ensuivies afin de pouvoir livrer un contenu mettant suffisamment en avant la charte artistique et l'aspect spectaculaire des objets « popant » dans le décor, la qualité du game design n'étant plus à démontrer suite aux très bons retours du jury lors du deuxième point.

Il a été difficile pour la graphiste de venir à bout de sa liste de tâche étant donné qu'elle devait cumuler un rôle d'illustrateur 2D pour les décors, textures et dessins des personnages avec celui de modelleur/animateur.

Pour respecter la deadline, nous avons donc mis en place une solution technique avec les programmeurs pour faciliter le travail d'animation des éléments modélisés dans le jeu ainsi qu'une solution logistique puisque la majeure partie de l'équipe s'est mise à se former à *3DStudio Max* afin d'alléger la masse de travail assignée à la graphiste.

Les outils

Il était difficile de s'organiser correctement entre nous pour travailler tous ensemble sur un même espace puisque nous devions suivre des cours dont la date n'était indiquée qu'au dernier moment. Nous disposions d'une visibilité limitée à la semaine, ce qui n'était pas simple pour planifier l'ensemble des tâches sur les 6 mois à venir.

Pour pallier au problème, nous avons mis en place différents outils adaptés aux besoins des membres de l'équipe.



Des règles de vie ont été mises en place dès le début du projet afin d'aménager des plages horaires fixes (9h30-12/14h-17h30) pour travailler ensemble sur les journées où nous n'aurions pas cours.

Des feuilles d'absence ont été données à remplir et donner au chef de projet. Ce système permettait à l'ensemble de l'équipe d'anticiper quelles tâches mettre en « stand by » à cause des absences de chacun et de mieux s'organiser dans le travail au quotidien.

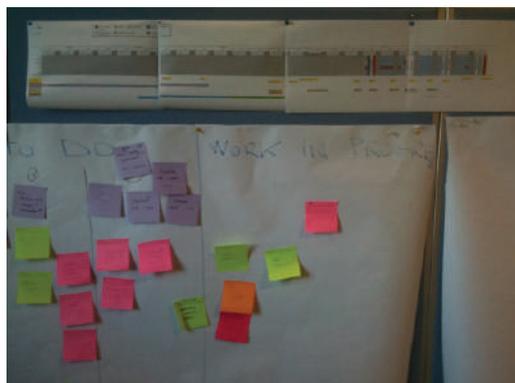
Un agenda *Google* en ligne fut fourni à l'équipe pour prévenir des événements hebdomadaires (réunions par spécialité, milestone à respecter...) et un système de reporting hebdomadaire fut mis en place afin de favoriser la transparence des tâches effectuées auprès de tous les membres de l'équipe sans exception.

Pour le versioning de fichiers, nous utilisons *Perforce* afin de simplifier l'échange de fichiers et leur mise à jour.

Communication écrite et orale : deux approches complémentaires...

Comme c'est souvent le cas en 2e année de master à l'ENJMIN, nous avons utilisé ces 6 mois de projet pour mettre en place une adaptation de la méthode SCRUM.

En matière de communication écrite, nous utilisons un rétroplanning affiché dans le box pour permettre à tout le monde de voir les milestones du projet à respecter (point jury, point avec l'équipe pédagogique, cours...).



Nous avons affiché une liste des tâches « à faire », « en cours », « fait », « validé » sur l'une des cloisons du box où chaque tâche était marquée sur un post-it (à la façon d'un backlog de tâches qu'on retrouve avec la méthode SCRUM). Chaque post-it devait indiquer la personne en charge de la tâche, combien de temps durait la tâche, quelle était la tâche, et quelle était sa priorité dans le planning (priorité 1, priorité 2).

Pour ce qui est de la communication orale, nous nous servions de stand up meetings pendant la phase de conception. Cette méthode n'ayant pas fait ses preuves, nous sommes passés à un point hebdomadaire chaque lundi matin à partir de la rentrée de janvier.

Ces « points », appelés milestone breakfast, étaient un moyen de nous retrouver autour d'un petit déjeuner convivial pour mener une discussion sur les tâches effectuées la semaine passée et déterminer les tâches faisables et à faire pour la semaine suivante (le reporting servant d'outil de suivi pour mener cette discussion).



...accompagnées d'une communication privée

Afin d'assurer le suivi personnel et le bien-être des membres de l'équipe au sein du projet, nous avons mis en place des entretiens personnalisés afin de faire des retours sur la gestion du projet et faire le point sur l'efficacité des outils de communication mis en place et des décisions prises auparavant.

Cela nous a notamment permis de faire le point sur les points positifs et négatifs du projet.

What went right

Lever les risques techniques

Les nombreuses phases de prototypes techniques sur l'ensemble du projet nous ont permis d'intégrer les éléments basiques de gameplay très tôt afin de lever les risques techniques liés aux animations pop-up, au chargement des éléments dans chaque niveau et au tournage de page du livre (pour changer les niveaux).

Le pitch final

Après quelques propositions de pitches infructueuses suite aux échanges avec l'équipe pédagogique, la graphiste et les game designers ont mené une discussion qui a abouti au pitch final qui a été présenté tout au long de la documentation. Suite à ce consensus, toute l'équipe a pu repartir sur une base solide pour développer le prototype du jeu.

Des mécanismes de gameplay simples

Les mécanismes de gameplay puisant leur origine dans les mécanismes qu'il est possible de trouver dans les livres pop-up, le game design de notre jeu se devait d'être suffisamment simple d'accès pour être compréhensible par le jury durant les trois présentations à mener sur l'année, mais suffisamment complexe pour convenir à un public adulte.

Le très bon accueil de l'ensemble du gameplay par le jury lors de la présentation de février a pu nous permettre de valider que notre jeu reposait sur une base solide...

A partir de février, notre challenge reposait donc non plus sur le game design mais sur la direction artistique afin de prouver que nous étions en mesure de proposer un univers riche et sortant de l'ordinaire grâce au travail d'une seule graphiste. L'état d'avancement de la qualité graphique du jeu n'étant pas représentative à ce jour du résultat final présenté lors du 25 mars, la réponse vous sera donnée à cette échéance.



What went wrong

Le 7-3-1

Par manque de clarté sur le pitch de départ (« Du papier, des ciseaux, de la colle... et du pop-up ») et de retours de l'équipe pédagogique sur le choix du projet, il nous a été difficile de trouver un concept de jeu correspondant au pop-up.

Nous avons donc conduits des brainstormings sur les aspects des livres pop-up qui nous semblaient intéressants à développer dans un jeu vidéo, et sommes tombés d'accord sur l'envie de découvrir ce qui se passe dans les décors des livres pop-up.

Notre pitch de base était donc fondé sur un principe de jeu d'échelles et d'interaction avec le décor : le joueur pouvait couper des éléments de décor pour s'en servir à tout autre usage (exemple : en faisant rentrer son avatar dans une maison, le joueur pouvait faire couper le lien permettant de tenir le mur d'une maison afin que le personnage se serve de ce pan de mur comme d'un pont pour traverser un gouffre).

Après avoir tenu un point avec l'équipe pédagogique, ceux-ci nous ont signifié que nous faisons fausse route car nous proposons un jeu d'action-aventure/énigme non plus basé sur le pop-up et la création spectaculaire de décor, puisque le but du joueur était de détruire le décor et voir ce qui se passait à l'intérieur, et que l'on perdait donc l'aspect magique du pop-up qui se forme au fur et à mesure que l'on tourne les pages d'un livre.

Suite à cela, nous avons décidé de suivre une phase de réflexion que l'on appelle 7-3-1 où il s'agit de proposer sept concepts

différents, puis en garder trois, et enfin en valider un seul.

Grâce à cet exercice, nous avons pu obtenir une base de mécanismes de gameplay dont les game designers ont pu se servir afin d'élaborer un nouveau pitch qui mettait le pop-up au centre du gameplay. De là est né le pitch qui vous est présenté depuis le début de cette documentation. Si c'était à refaire, nous aurions dû mener une phase de 7-3-1 dès le lancement du projet en octobre, chose qui a été faite un peu tard, en décembre seulement, afin d'avoir un filet de secours lors du point pédagogique de novembre, ce qui nous aurait évité de perdre 3 mois de travail (comme indiqué dans le planning ci-haut).

Juger les idées

Lors de la phase de conception du projet, nous avons passé énormément de temps à écouter les propositions des game designers. Notre erreur fut de juger les idées sans même essayer de les prototyper.

Si c'était à refaire, nous lancerions la phase de prototypage le plus tôt possible, que cela soit sous *Unity* ou *Powerpoint*, dont nous avons pu nous servir comme « prototype » lors du point jury #1.

Il faudrait se donner des objectifs plus clairs à réaliser en prototype sur des courtes durées afin de stabiliser le concept suffisamment tôt (début novembre) pour commencer la phase de production de prototype dès décembre.

Prototyper est la meilleure façon de savoir si une idée fonctionne ou non.



Plus de réunions

Etant donné que nous devions suivre des cours en parallèle du développement du prototype de jeu, il était difficile d'aménager des journées en équipe complète et de fixer des réunions pour rassembler tout le monde.

Cela dit, si nous devions reprendre un aspect de la gestion du projet, nous aurions dû mener plus de réunions en équipe complète sur le début de la phase de conception. Cela couplé à la phase de 7-3-1 nous aurait peut-être permis de tomber sur un pitch satisfaisant pour le projet de 2e année de master à l'ENJMIN plus tôt dans le processus de réflexion.

Direction artistique

A cause de la refonte tardive du pitch du jeu, il était difficile pour la graphiste de définir une ambiance de jeu. Il a fallu attendre que le scénario soit posé de façon définitive (fin janvier) pour que nous puissions enfin avoir une idée de la direction artistique à développer.

Ayant une seule graphiste sur le projet, cela a été d'autant plus handicapant sur l'ensemble de notre phase de production de prototype car nous n'avions qu'une seule personne en charge des textures, de la modélisation des éléments et de leur animation.

Pour pallier à ce manque, une partie de l'équipe s'est décidé à utiliser *3DStudio Max* pour aider la graphiste à produire plus rapidement les assets de base.





Remerciements



Toute l'équipe souaite remercier, pour leur soutien, leur présence et leurs conseils :

- Sophie Pierson et Thérèse Sevrin-Renier
- Les promotions d'enjminiens 2009/2010 et 2010/2011
- L'ensemble de l'équipe pédagogique
- Les membres du jury.

Et nous pensons aussi à toutes les autres personnes que nous n'avons pas pu citer.



