



AURORE



Coralie
Feniello



Sébastien
Violier



Anaïs
Riff



Dorian Souc



Théo
Verpillat



Océane
Kressmann



Jérémy
Guarober



Victor
Verpillat



Eleonore
Dambre

Nature de l'expérience



Voyage

dans un monde intérieur.

Contemplation

d'un environnement en évolution.

Redimensionner

des éléments de décor.



Objectif
du joueur

Arriver à la prochaine gare
et rattraper le train

Intentions



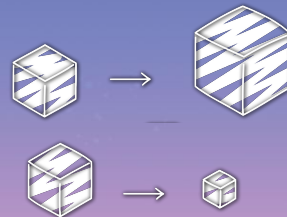
Le joueur incarne Aurore, une fillette de dix ans perdue dans ses rêveries lors d'un voyage en train...



- ★ Aventure-puzzle contemplatif
- ★ Pas de challenge d'adresse
- ★ Vue à la 3ème personne
- ★ Solo
- ★ Deux niveaux

Obstacle
Environnement

Moyen d'action
Modifier la taille des objets
qui nous entourent



3C



Aurore

Pouvoir spécial : agrandir et rétrécir les objets qui l'entourent.

Déplacement : course fixe (niveau 1), marche (niveau 2), saut.

Apparence : pas de visage, le corps noir couvert de tatouages (ils brillent quand elle utilise son pouvoir). Muette.



	Clavier / souris	Manette
Déplacements	Z Q S D	Stick gauche
Caméra	Souris	Stick droit
Saut	Espace	A
Seconde caméra	Shift	X
Agrandir	Clic gauche	RT
Rétrécir	Clic droit	LT
Pause	Echap	Start



1e Cam

- ★ Éloignée du PJ
- ★ 3D en perspective
- ★ 3e personne
- ★ Rotation
- ★ axes x et y

2e Cam

- ★ Sur rail
- ★ Plus éloignée
- ★ Fixe
- ★ 3e personne
- ★ Mise en valeur de l'environnement

Game World

Dans le train

Accessible dans le menu principal et le menu pause. Le joueur ne peut interagir qu'avec l'UI.

Niveau 1 : on retrouve des éléments du rêve dans le paysage qui défile.

Niveau 2 : on est dans un tunnel.



Dans le rêve

Niveau 1 : Lumineux, aérien, enfantin. Aurore vagabonde dans ses pensées et en contemplant l'architecture des bâtiments qui défilent, elle commence à se remémorer des bribes de souvenirs.

Niveau 2 : Sombre, mystérieux, inondé, un écho d'une visite à sa grand-mère en Inde, où elle a découvert avec crainte mais respect ce qu'était un Baoli.

Game Design

Traversée de l'environnement

Différents types d'interaction avec un obstacle ou un levier :

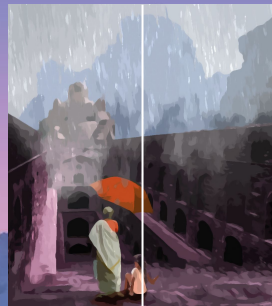
- ★ rétrécir
- ★ agrandir
- ★ maintenir pour agrandir
- ★ agrandir pour bondir
- ★ contrepoids d'objets

Du niveau 1 à la fin du niveau 2, évolution des intentions



- ★ **Niveau 1** : interactions simples, éléments déstructurés, l'environnement se désagrège après le passage du joueur.
- ★ **Niveau 2** : interactions de destruction, plus complexes.
- ★ **Arrivée à la dernière gare** : reconstruction pour avancer, retour au réel.

Les bonus Memories : l'aspect narratif valorisé, pour éviter un simple jeu de plateforme joli.



Les morceaux d'images sont récupérés dans les niveaux, dans des zones qui requièrent une observation attentive.

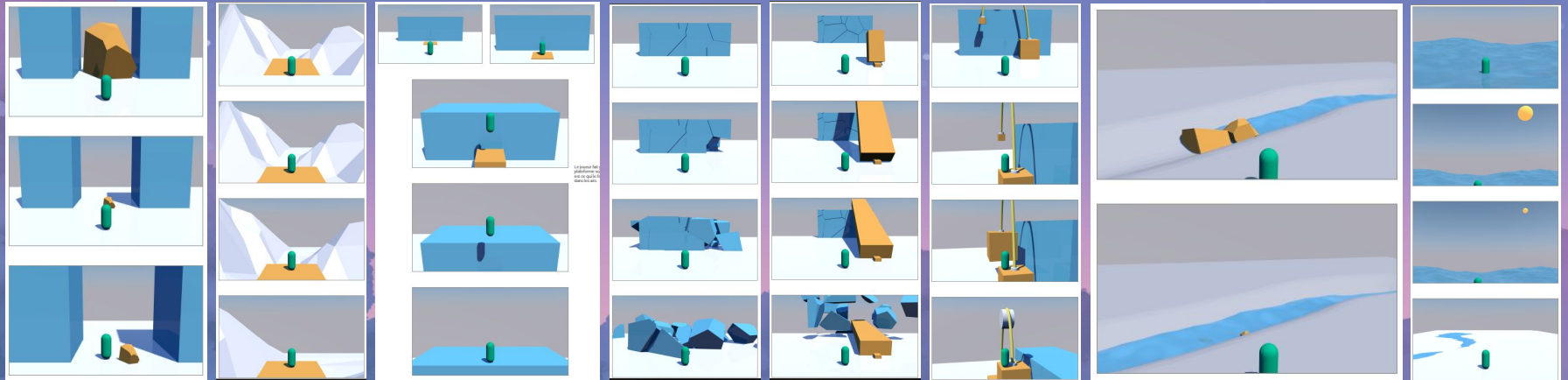
L'image 3 est en évidence, afin de signaler au joueur qu'il y avait d'autres images à récupérer au cours du jeu.

Level Design

Courbe de difficulté



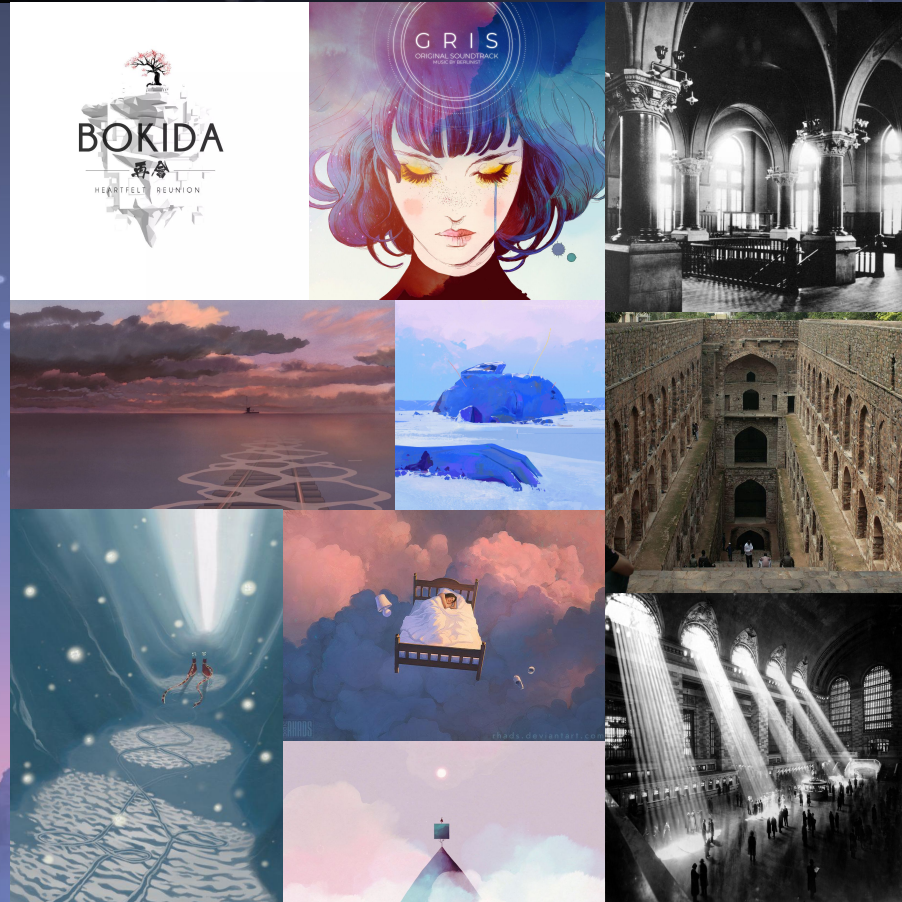
Briques de LD storyboardées



Direction Artistique

Challenges

- ★ Faire comprendre que c'est un rêve à travers les graphismes mais aussi le son.
- ★ Retranscrire graphiquement le changement d'ambiance du niveau 1 au niveau 2.
- ★ Trouver des moyens de réduire le scope conséquent et d'optimiser la production d'assets.
- ★ Accentuer le gamefeel et l'envie de contempler avec les VFX et les contrastes de taille.



Direction Technique

Principaux outils utilisés :

- ★ **Houdini** : création d'assets procéduraux (gain en temps de production !) et destructions de mesh
- ★ **Unity & Wwise**, le combo habituel des projets de 1ère année

Plugins Unity :

- ★ **Aura** : solution simple pour la gestion des lumières volumétriques et du brouillard pour le 2ème niveau
- ★ **Cinemachine** : pour l'ensemble des caméras du jeu, les nombreuses options (transitions entre caméras, rail, suivi...) nous ont permis de créer des plans efficaces
- ★ **Probuilder** : pour du prototypage et des niveaux en greyblocks rapides

Direction Technique

Développements notables :

- ★ **Character Controller** : repris à 3 reprises avec les évolutions du projet :
 - Basé sur un rigidbody avec physique d'Unity (pas assez de contrôle)
 - > Basé sur le component d'Unity CharacterController (ground detection trop instable)
 - > Basé sur un Kinematic rigidbody et une physique personnalisée (meilleure alternative !)
- ★ **Shaders** :
 - **Nuages** : Après l'essai de nuages volumétriques coûteux, la solution la plus optimisée s'est portée sur un plan utilisant du triplanar mapping et un effet de brouillard
 - **Effet "peinture"** : shader post-process basé sur le filtre de Kuwahara
 - **Eau** : shader complet avec mousse aux intersections, disparition progressive avec la profondeur, distorsion de texture selon un bruit, vagues selon le modèle de Gerstner
 - **Triplanaire personnalisé** : shader permettant de ne pas déployer les UVs des modèles, de donner un effet de matière qui glisse lors du rescaling (et d'éviter de l'étirement de texture) et personnalisé pour avoir un effet de sélection basé sur une rimlight bruitée (et d'éviter un viseur en UI)