

Rapport de Stage

Grégoire Carabeufs

Stage de Gameplay Programmer

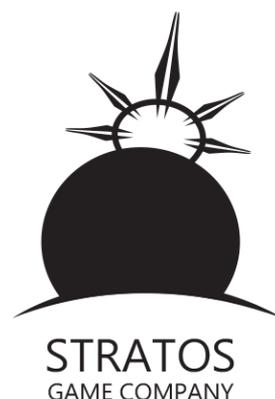
Du 11/07/2016 au 09/09/2016

Master jeux et médias
interactifs numériques cohabilité
par le Conservatoire national des
arts et métiers, l'Université de
Poitiers et l'Université de La
Rochelle

le **cnam**
enjmin



Entreprise : Stratos Game Company
Maître de stage : Paul Maurer, Game Designer



Sommaire :

Introduction :

Présentation de l'entreprise

Présentation du projet

Déroulement du Stage :

Première semaine

Développement du controller et de la caméra

Les Intelligences Artificielles et l'éditeur de FSM

Ce qu'il reste à faire d'ici la fin du stage

Conclusion :

Introduction :

Présentation de l'entreprise

Stratos Game Company est un studio de jeu vidéo créé en 2015 par Paul Maurer. L'entreprise a réalisé en interne différents prototypes de jeux mobiles auparavant, et projette de sortir bientôt un jeu multijoueur en local indépendant nommé Volley Pong.



Screenshots de Volley Pong

L'équipe était déjà composée d'un Game Designer (Paul Maurer) et d'un programmeur (Brieuc Swales), et nous étions trois étudiants de l'ENJMIN en stage (1 programmeur, 1 CDP, et 1 graphiste). Ma mission dans ce stage était d'assister le programmeur déjà présent, notamment sur la 3C du jeu et sur les intelligences artificielles des ennemis.

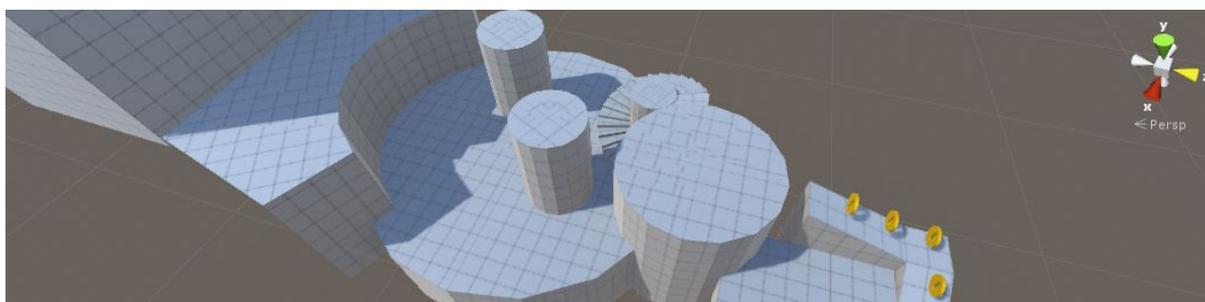
Présentation du projet

Ayant abandonné l'idée de sortir un jeu sur téléphone portable, l'entreprise souhaite se réorienter vers le jeu PC. Son projet de jeu est un plateformer 3D dans la même veine que ceux de l'âge d'or du genre (fin des années 90 - début des années 2000) avec des titres tels que Banjo-Kazooie ou Ratchet & Clank. Le but est de toucher à la fois un public nostalgique mais aussi de toucher ceux en demande de ce type de jeux de nos jours, en particulier sur PC où le genre est encore moins représenté.

Déroulement du Stage :

Première semaine

J'ai eu dès mon premier jour de travail une réunion avec l'autre programmeur de l'équipe pour décider de l'architecture du projet, et de la répartition des tâches. Le projet n'étant qu'en phase de pré production, nous avons le temps d'expérimenter ce que nous voulions, et de développer des outils pour le reste de l'équipe. Nous avons aussi décidé des différents plugins existants que nous allons utiliser pour le projet. Pro Builder s'est révélé très utile pour prototyper rapidement des niveaux, et tester le controller du jeu. Nous avons également utilisé le plugin InControl pour gérer facilement les différentes manettes existantes, et rendre le jeu compatible à la manette et au clavier souris.



Exemple de début de niveau réalisé grâce à ProBuilder

Développement du controller et de la caméra

Le controller du jeu utilise pour l'instant le component Character Controller d'Unity et prévoit déjà certaines des features essentiels au jeu. Le joueur possède un double saut, et le game designer peut paramétrer la vitesse de déplacement et la hauteur des sauts. La physique du personnage se comporte différemment dans l'eau, et le joueur peut nager. La physique permet de se déplacer sur des plateformes en



mouvement. Si les différents pouvoirs que possède le joueur ne sont pas encore développés, le joueur peut déjà naviguer entre des pouvoirs placeholder. Les paramètres du controller du personnage sont stockés dans des scriptable objects afin de pouvoir tester des paramètres différents au cours du développement, mais également de pouvoir éventuellement changer les paramètres et aptitudes du personnage facilement en cours de jeu.

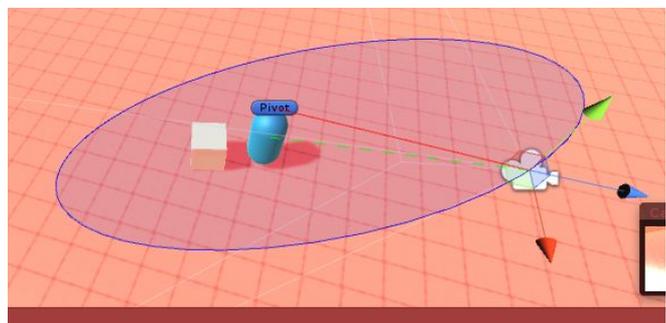


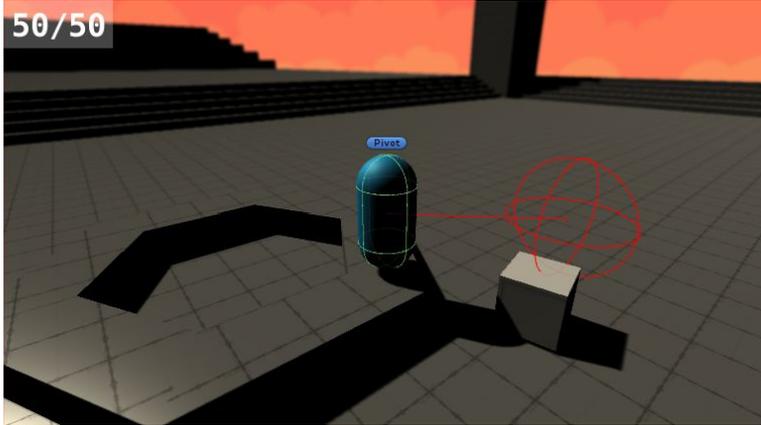
Une attention particulière a été portée également à la caméra. Elle ne peut pas passer à travers les murs, et se rapproche du joueur si elle rentre en collision avec quelque chose, ou si un objet obstrue la vue entre la caméra et le joueur, en utilisant des raycasts. Elle s'éloigne plus ou moins selon la vitesse de déplacement du joueur, et, tous ses paramètres de rotation peuvent être modifiés comme on le souhaite.

La caméra peut être focus sur le joueur, mais aussi avoir plusieurs cibles dans son champ de vision et s'adapte via un système de poids. Cela peut être utile par exemple pour une meilleure visibilité dans une pièce avec plusieurs ennemis.

L'autre programmeur de l'équipe a créé un outil de création de courbes de béziers dans Unity. Je l'ai intégré dans le système de caméra, pour que la caméra puisse suivre une courbe durant une cinématique.

Visualisation dans l'éditeur de l'axe de rotation de la caméra

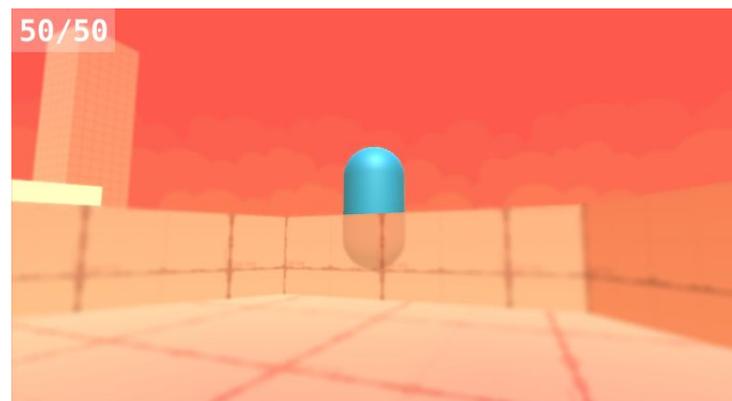




Aperçu du controller en cours de jeu, avec les gizmos de debug.

Dans l'ensemble, le controller est maintenant assez complet pour pouvoir prototyper les premiers éléments de Level Design. Le personnage se contrôle agréablement à la fois à la manette et au clavier, sa physique est proche de la physique finale voulue, il possède toute la panoplie de mouvements pour l'instant imaginée dans le game design document, et la caméra interagit comme il faut avec l'environnement.

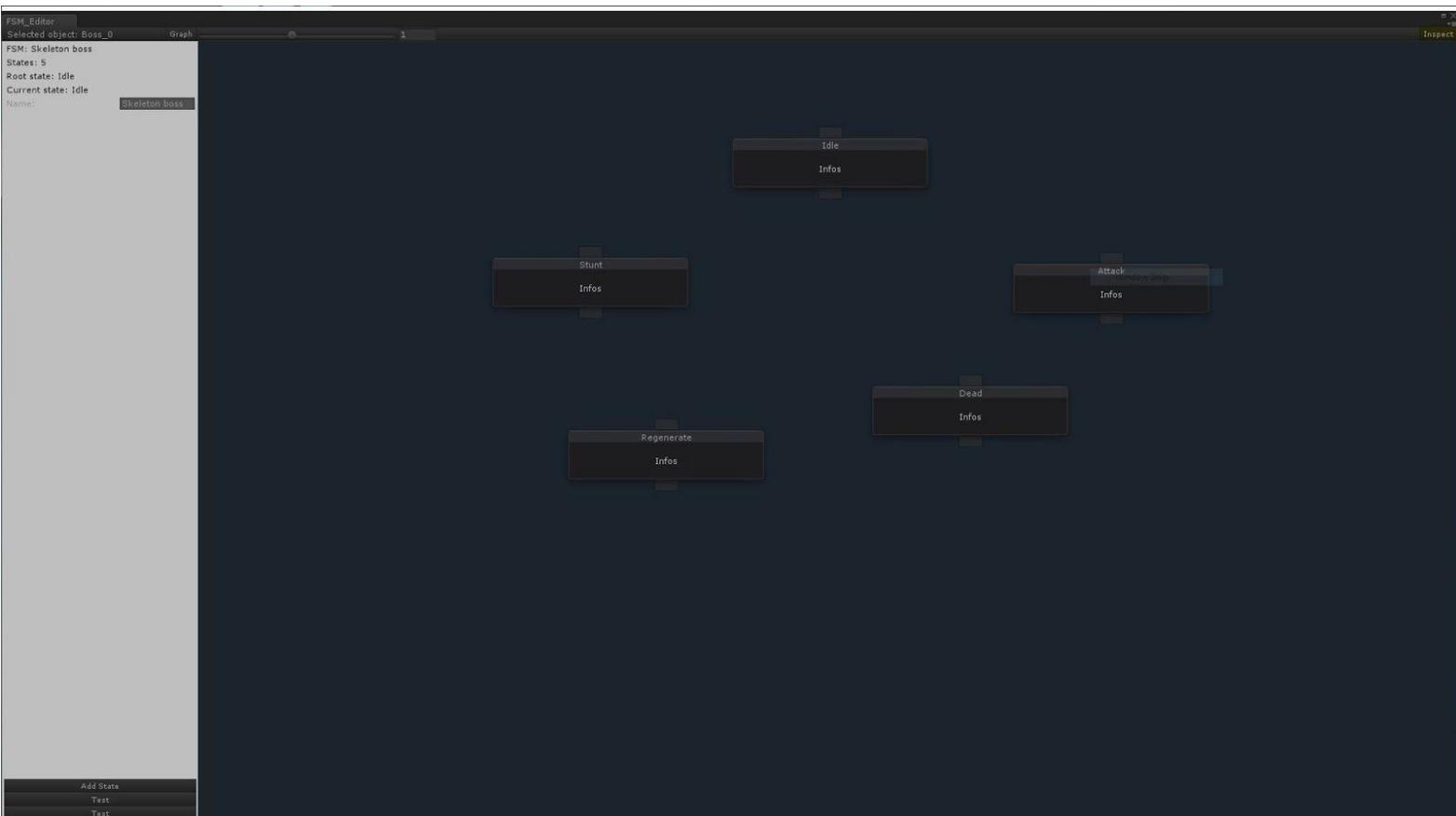
Un shader permet de voir le joueur à travers les murs



Les Intelligences Artificielles et l'éditeur de FSM

Pour les besoins du projet, nous avons décidé de créer notre propre interface dans Unity d'édition de finite state machine et de behaviour trees. Nous ne voulions pas investir dans un plugin, et nous cherchions à avoir le contrôle total sur ce que nous faisons. La classe gérant la création d'états, leurs transitions, et leurs conditions est finie et déjà intégrée dans les IA présentes dans le jeu.

L'éditeur visuel de FSM lui est encore en cours de développement. Il permettra de créer des behaviour trees sans passer par du code, et de visualiser les transitions pour un débog plus facile des IA.



L'éditeur de FSM dans son état actuel.

L'un des ennemis du jeu est terminé, et utilise notre système de FSM. Ce sont des cubes de défense se déplaçant aléatoirement dans une zone définie dans l'éditeur, ils représentent le type d'ennemis le plus basique du jeu. Si un cube se trouve seul et à proximité du joueur celui-ci va se mettre à le poursuivre lentement pour l'attaquer au corps à corps. Si jamais d'autres cubes de défense sont à proximité de lui, il préférera s'assembler avec les autres. Ils montent les uns sur les autres pour former une tour, puis attaquent le joueur. L'animation de déplacement des cubes étant simple est gérée par le code pour qu'elle soit plus réactive. L'animation d'assemblage de la tour de cubes est gérée, elle, via des courbes de béziers.

Ce qu'il reste à faire d'ici la fin du stage

L'objectif visé par l'entreprise était de réaliser une démo avec quelques-unes des mécaniques du jeu, et un début d'intégration graphique. Il reste du temps de développement avant la fin du stage, les assets 3D sont encore en cours de production par le game artist en stage, et j'aurai sûrement comme tâche de les intégrer une fois qu'ils seront finis. Certaines des mécaniques imaginées par le game designer ne sont pas encore intégrées au jeu. Le jeu prévoit que le personnage principal possède plusieurs talismans correspondant chacun à un pouvoir. Pour l'instant seul le pouvoir du laser a été commencé. Quand aux différents comportements d'ennemis seulement un est fini pour l'instant. Pour autant, il y aura sûrement assez de contenu développé afin de réaliser un niveau de démo d'ici la fin du stage avec une partie des features du jeu.



Le modèle 3D encore en cours de modélisation du personnage principal

Conclusion :

Ce stage m'a permis de travailler sur un type de jeu qui m'intéresse énormément. Les plateformes 3D demandent en effet une 3C parfaite pour être agréables à jouer. Le programmeur Gameplay doit donc à la fois coder une physique de personnage impeccable, avoir une caméra facile à contrôler qui ne gêne pas la lisibilité des phases de plateforme, et réussir à développer toutes les capacités du personnage demandées par le Game Designer. Être sur un projet de ce type représente ainsi un certain challenge, et j'ai beaucoup appris sur les meilleures pratiques à avoir pour un jeu de plateforme en travaillant sur ce projet.

J'ai pu aussi appliquer dans ce projet certains des principes de programmation vu au cours de l'année tel que l'utilisation de finite state machine et de behavior trees pour l'intelligence artificielle des ennemis.

C'était également la première fois que je travaillais en télétravail sur un projet de cet envergure.

Je tiens à remercier mon maître de stage, ainsi que l'équipe pédagogique.