

Document de design

La taverne du voyageur

TP3: Préproduction

Par:

Arroyo, Fabrice - 111 267 278

Marty, Pierre - 536 793 111

Bouchard, Alex - 111 229 958

Lacaille, Paul - 536 793 104

Équipe 04

Réalisé dans le cadre du cours :

IFT-3113 – Projet de jeu vidéo

Rapport présenté à :

Chéné, François

Remis le :

8 Mars 2021



UNIVERSITÉ
LAVAL

Table des matières

Résumé	1
Expérience de jeu prévue	1
L'intégration du thème	1
Inspirations	1
Boucle de jeu	2
Mécaniques prévues	2
Ressources de jeu	3
Plateforme technologique	3
Présentation de l'équipe	4
Annexe 1: Défis technologiques	5
Défi n°1 - Environnement Statique	5
Objectif	5
Résultats	5
Défi n°2 - Environnement Vivant	5
Objectif	5
Résultats	6
Défi n°3 - Intelligence Artificielle et Pathfinding	6
Objectif	6
Résultats	7
Défi n°4 - Modélisation 3D et Animation 3D	7
Objectif	7
Résultats	7
Annexe 2: Rapport d'avancement 1	8
Ce qui a été fait depuis la dernière étape	8
Les obstacles rencontrés	8
Les changements à apporter à la planification	8
Annexe 3: Galerie d'art de production	8

Résumé

Gestion en temps réel d'une taverne dans un univers médiéval-fantastique

Expérience de jeu prévue

Clémence s'installe devant son ordinateur. Elle a quelques heures devant elle et souhaite se détendre avec un peu de fantaisie. Sa taverne fonctionne actuellement à plein régime, malheureusement il y a trop peu de place pour les clients, qui commencent à s'en plaindre.

N'ayant pas encore les ressources pour étendre son établissement, elle décide plutôt d'engager du personnel pour proposer d'autres services. Après avoir hésité avec un barde, elle choisit d'engager un Vieux Mystérieux A Capuche (V.M.A.C) pour donner des quêtes aux aventuriers de passage. Après avoir paramétré une quête facile de recherche d'artefact, elle attend qu'un aventurier ou deux viennent s'y intéresser.

Rapidement une compagnie débutante viendra voir le V.M.A.C et quittera les lieux avec la quête. Le bâtiment se vide ainsi quelques peu et les autres autres clients sont plus heureux.

Plus tard, la compagnie revient victorieuse. Après avoir touché sa récompense, elle décide de la dépenser en partie dans la taverne en payant une tournée. Les clients sont heureux et l'argent comme la bière coule à flot.

Avec ses nouveaux bénéfices, Clémence décide de construire une extension comprenant un dortoir. Ainsi les clients restent plus longtemps et les aventuriers dorment sur place, ce qui leur permet de reprendre une quête dès le lendemain.

L'intégration du thème

Le thème du graal est essentiellement intégré par l'univers médiéval fantastique. De plus, le but est d'assurer le bonheur de clients aventuriers qui partent en quête d'artefacts, ces artefacts pouvant être littéralement ou métaphoriquement un "graal".

Enfin, quelques petits easter eggs pourront référencer le Saint Graal dans la pop-culture.

Inspirations

Le gameplay est inspiré des jeux de gestions tel que Theme Hospital où l'on gère un établissement de service. Le joueur étant confronté à des clients aux besoins et envies propres.

L'univers lui est inspiré sur les jeux de rôles sur table d'heroic fantasy tel que DnD ou Pathfinder. Nous comptons notamment jouer sur l'expression "you meet in a tavern". Le rôle étant ici inversé, le joueur incarnant non pas les aventuriers, mais le PNJ tavernier.

Boucle de jeu

Une partie de “La Taverne du voyageur” se déroule comme suit : Le joueur contrôle initialement un personnage tavernier et un bâtiment de taverne. Le système génère aléatoirement des personnages non joueurs servant de clients. Chaque client servi rapporte de l’argent au joueur, ce qui lui permet d’agrandir sa taverne ou d’embaucher du personnel.

Mécaniques prévues

- Flot dynamique
 - Actions
 - Mouvement
- Inventaire
 - Du Tavernier
 - De la taverne
- IA
 - Des employés
 - Des clients
- Progression
 - De la taverne
 - Des employés

Ressources de jeu

Des Acteurs :

- 1 modèle (Maillage riggé + texture)
- Animation (+ SFX)
 - Attente
 - Marche
 - Assis
 - Consomme
 - Fête

Environnement :

- Murs
- Portes
 - Animation d'ouverture
- Fenêtres
 - Animation d'ouverture
- Texture du sol

Menus

- Menu principale
- Menu d'options
- HUD
- GUI
- Menu de pause
- Inventaires

Plateforme technologique

Le jeu sera réalisé en 3D sous l'engin Unity3D 2019.4.20f1. Le code en C# avec l'aide de l'éditeur de solution Rider et de l'IDE Visual Studio. Les modèles seront réalisés sous Blender. Les éléments 2D (Texture, éléments de menus, mattes) seront fait sous Paint et Gimp. L"audio sera réalisé sous Audacity. Github et Trellon serviront à la gestion du projet. La suite Google Drive sera utilisée pour la documentation.

Présentation de l'équipe

Fabrice Arroyo - Artiste 3D

Pierre Marty - Développeur gameplay

Alex Bouchard - Programmeur d'intelligence artificielle

Paul Lacaille - Project manager

Annexe 1: Défis technologiques

Défi n°1 - Environnement Statique

(Paul Lacaille)

Objectif

L'environnement de jeu doit être limité à l'espace de la taverne. Cet espace doit être navigable par les personnages grâce à l'ia.

De plus, afin de permettre une évolution au joueur, l'environnement doit pouvoir s'agrandir avec le temps et selon les choix du joueur.

Résultats

Le prototype est composé de trois types d'éléments : Les portes, les murs extérieurs et les murs intérieurs. Les portes permettent la transition entre deux salles, les murs intérieurs sont les endroits pouvant être étendus et les murs intérieurs délimitent deux salles.

A l'exécution le prototype comprend une unique salle possédant une porte et neuf murs extérieurs. Quand le joueur sélectionne un mur extérieur, celui-ci est remplacé par une porte et une salle de trois murs extérieurs est générée.

Afin d'éviter la superposition d'éléments, quand deux murs se retrouvent à la même position, ceux-ci sont remplacés par un unique mur intérieur. L'existence des murs intérieurs permet d'éviter que le joueur ne puisse créer une salle à l'intérieur d'une autre.

Défi n°2 - Environnement Vivant

(Pierre Marty)

Objectif

La zone de jeu doit rester claire et cohérente afin de ne pas brouiller le champ visuel du joueur. Ce prototype utilise, le résultat des défis 1 et 3 afin de valider la disposition de la salle, le nombre d'entités se déplaçant dans la taverne, ainsi que le placement de la caméra. Ce dernier point doit prendre en compte qu'un certain nombre de boutons/menus seront possiblement affichés lors de la simulation. Il faut donc veiller à ce qu'un espace "vide" suffisant soit disponible autour de la taverne pour permettre cet affichage.

Résultats

La fenêtre de simulation fait (par défaut) 1280x720 (format HD 16:9 standard) et est modifiable. On retrouve, au centre, la salle de la taverne - où les "clients" (agents autonomes) peuvent se déplacer - et sur la gauche un menu pour augmenter ou diminuer le nombre de clients présents.

Afin d'avoir un retour visuel sur le bon fonctionnement des IAs, un compteur à été placé au-dessus de chacune d'elles. Il indique le temps restant avant que celle-ci ne change d'état. Leur état courant n'étant pas nécessaire, pour ce prototype, aucun indicateur n'a été placé pour le relater.

Le menu, à gauche de la fenêtre, permet d'ajuster la quantité de clients présents dans la taverne. Cet ajustement se fait par le biais d'un objectif (champ "target") à atteindre. Si le nombre courant de client n'est pas assez élevé alors de nouveaux agents seront ajoutés petit-à-petit, si il y en a de trop, alors certains seront supprimés.

Il est possible de régler l'objectif en fonction du nombre de sièges disponible dans la salle (*empty->vide; full->pleine; 25%; 50%; 75%*) ou bien de l'augmenter ou diminuer un par un.

A noter qu'il est possible d'ajouter plus d'agents que le nombre de sièges présents dans la salle.

L'ajout et la suppression d'un agent autonome n'est pas immédiat afin de ne pas surcharger la machine faisant rouler la simulation. Il faut observer ~1.5 secondes avant que l'action ne soit effectuée. Ainsi si l'on demande d'ajouter 60 agents il ne seront tous présents et actifs qu'après 1 minute et 30 secondes.

Défi n°3 - Intelligence Artificielle et Pathfinding

(Alex Bouchard)

Objectif

Créer un algorithme de recherche de chemins pour que les personnages non-jouable se déplacent. L'algorithme doit aussi pouvoir fonctionner avec le système de création d'environnement, Par la suite créer une intelligence artificielle de base pour que les personnages non-jouable accomplissent des actions en utilisant l'algorithme de recherche de chemins.

Résultats

Créer l'algorithme de recherche de chemin fut plutôt simple puisque Alex a pu s'inspirer des algorithmes de recherche de chemin qu'il a fait dans le passé avec A*. Au lieu que le graph de A* soit déjà fait d'avance, l'algorithme doit commencer par trouver tous les nœuds qui se trouvent dans chaque pièce de l'environnement (soit sortir tous les component "Node" qui se trouvent dans la scène) et de vérifier si un nœud peut se rendre à n'importe quel autre. Si c'est le cas, les nœuds deviennent voisins. Ce processus fait en sorte que le graph est généré automatiquement et requiert seulement que des nœuds se retrouvent dans les salles génériques que le joueur peut ajouter dans son auberge. Le seul problème avec cette méthode est que si l'environnement devient énormément grand, la génération du graph au début d'une partie pourrait ralentir le jeu, mais cela n'arrivera pas si nous limitons la grandeur maximale de l'environnement.

Pour ce qui est de l'intelligence artificielle, il s'agit simplement d'une machine à états. En effet, en changeant l'état du personnage, il agit différemment. Comme proposé à l'étape d'analyse et planification du projet, la machine à état a présentement six états:

- Un état où le client recherche une place libre où s'asseoir;
- Un état où le client attends que nous prenions sa commande;
- Un état où le client attends de recevoir sa commande;
- Un état où le client consomme sa commande;
- Un état où le client se dirige vers le comptoir pour payer;
- Un état où le client se dirige vers la sortie de l'auberge.

Il s'agirait d'états que la majorité des personnages vont partager, mais il sera facile de créer une différente liste d'états pour un personnage spécial, il sera même possible de copier le même code présent dans le code de cette machine de base pour certains états.

Défi n°4 - Modélisation 3D et Animation 3D

(Fabrice Arroyo)

Objectif

Réaliser des modèles 3D ayant un style intéressant pour le joueur sans trop être trop compliqué pour avoir le temps disponible pour créer le nombre d'objets et personnages nécessaire pour créer un espace vivant dans la scène du jeu.

Il faudra tout aussi que le style artistique du jeu permettent au joueur de voir clairement ce qu'il se passe dans la taverne.

Il est aussi très important que l'animation des modèles soit assez intéressante pour être garder le joueur intéressé au jeu. Ce qui requiert une importance à bien rigger les modèles utilisés, pour ne pas se limiter aux options pour d'animations.

Résultats

Ayant déjà une inspiration proposée par le groupe d'utiliser des modèles avec moins de polygones, non seulement permet de garder le temps de production bas, mais aussi les distingue par leurs style. La simplicité des modèles permet aussi de pourvoir les rigger beaucoup plus rapidement en utilisant des groupes de vertex reliés au squelette d'animation. Cette technique permet aussi à plus facilement retoucher certaines erreurs faite durant la distribution de ces vertex sur les os du squelettes, ce qui permet de peaufiner le modèle prototype si le style vient à changer sans avoir à retoucher le squelettes d'animation.

Cette même technique est aussi utilisée sur les tables de la taverne, permettant d'orienter le dessus quand un client vient s'appuyer dessus ou quand un objet lourd y est déposé, donnant plus de poids au personnage et rendant la scène plus vivante.

Les premiers modèles de prototype réalisés donnent une bonne impression du temps nécessaire à la création des autres. Étant 3 jours pour un modèle de personnage, y compris le rigging, et environ 10 minutes pour tout autre modèle, y compris texture ou rig si nécessaire.

Finalement, pour les textures, bien qu'initialement le but était de créer les images de texture à la main, le style ne fonctionnait pas totalement avec les modèles. Après quelque tentative, une bonne facont est de créer un modèle 3D très simple ou avec un palette de couleur, colloré celui ci. Il suffit seulement de capturer une image de ce modèle et de l'appliquer comme texture image au mur ou autre modèles. Les texture fait à base de palette de couleur sont aussi utile pour facilement changé la couleur de modèles plus simples, juste en modifiant la palette.

Annexe 2: Rapport d'avancement 1

Ce qui a été fait depuis la dernière étape

Prototype de pathfinding : 16 Février

Prototype d'IA : 18 Février

Prototype d'environnement statique : 19 février

Prototype de modèle 3D : 22 Février

Prototype d'environnement vivant : 05 Mars

Rapport d'avancement 1 : 05 Mars

Première version du document : 05 Mars

Les obstacles rencontrés

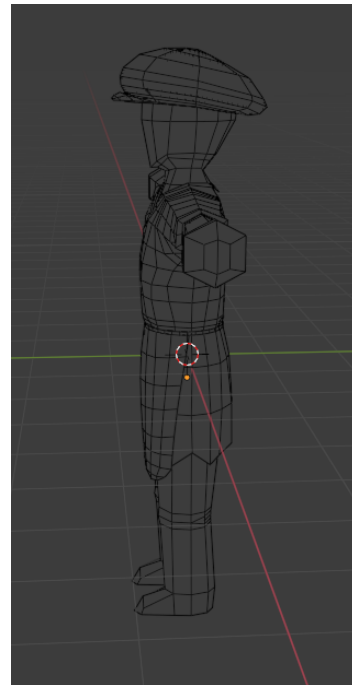
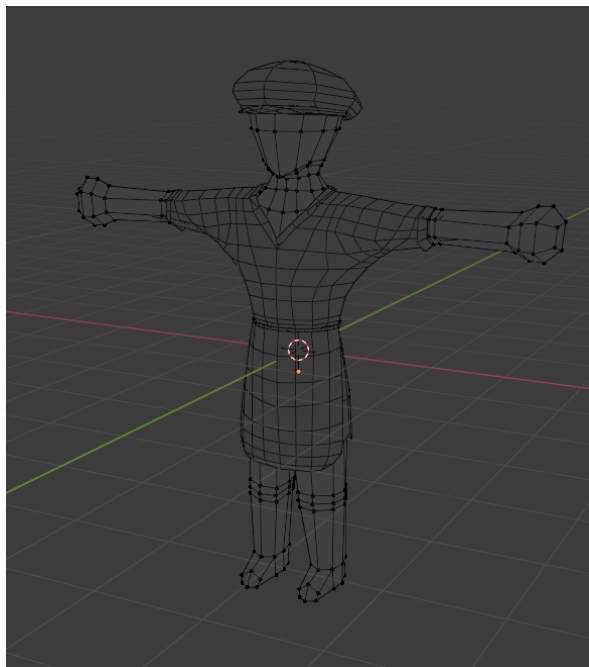
Les nombreux examens de l'équipe ont ralenti le travail pour tous les membres de l'équipe.

Les changements à apporter à la planification

Annexe 3: Galerie d'art de production

Serveur (personnage jouable)

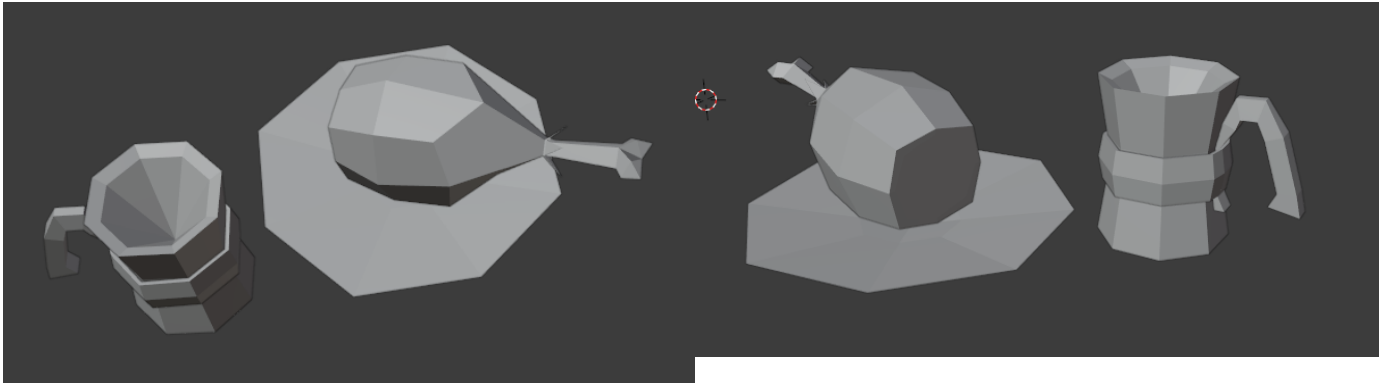
Employé à la taverne du voyageur, incarné par le joueur, il prend et sert les commandes des clients de la taverne. Étant le seul employé restant, il se doit de gérer tous les clients et tout



autre activité allant à nettoyer la taverne à cuire la nourriture, tout ceci au salaire minimum. Le tablier et le béret non seulement

Nourriture (Viande, avec assiette et chope vide)

Nourriture et breuvage qui sont servis à la taverne du voyageur. Le classic d'une grosse coupe de viande avec une chope de bière ne laisse pas grand à l'imagination pourquoi il est inclus dans le menu. Puis pour le fait d'être très facilement identifiable même par sa petite taille dans l'environnement.



Environnement de test

Composé d'une table et de murs et plancher texturé, ceci sert à visionner un approximatif de ce que la taverne finale ressemblera avec un composition de différentes textures appliquées, notamment pour faciliter la visibilité. Il sera notamment important de voir comment les textures interagissent ensemble dans le but aussi de garder le sentiment d'hospitalité

