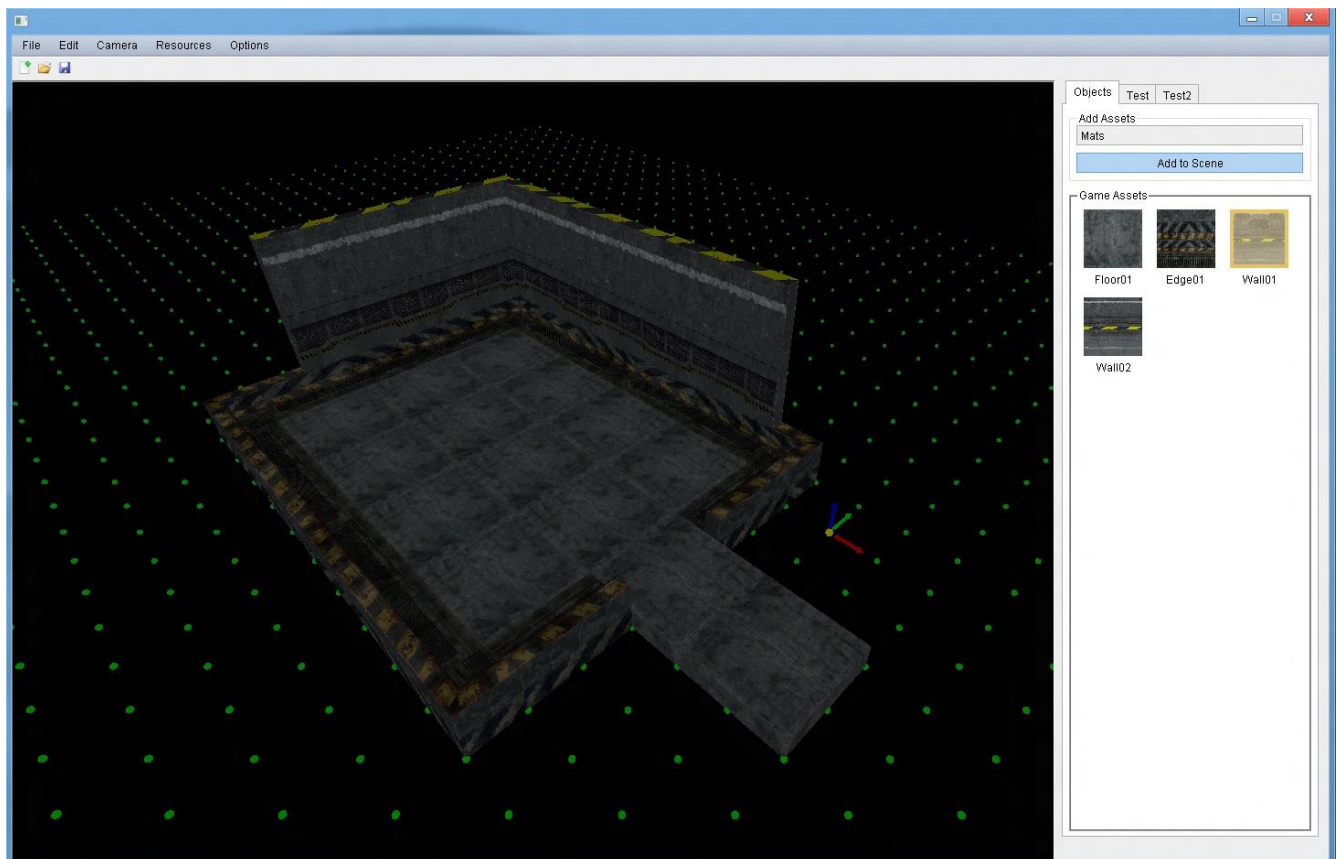


PROJET ORACLE

PHASE 2

Thème: Construction de niveaux et design de base de données



Objectifs

- Designer une base de données et des requêtes
- Programmer un programme de conception (en 2D ou 3D)
- Livrer une fonctionnalité sous forme d'itération 0

Présentation

Pour la phase 2, vous réaliserez un éditeur de niveau. Ce type de programme sert à créer des niveaux suivant notre inspiration. Vous permettrez ainsi à vos joueurs de profiter sans cesse de nouveau contenu.

Un menu permettant de choisir son niveau sera ajouté dans *Tankem*.

Il s'agira essentiellement de créer un produit fonctionnel le plus simplement possible (itération 0).

Modalités

La remise se fera à la date convenue via Léa et vaut 30% de la session. Ce travail doit être fait en équipe de 4 personnes et vous pouvez changer les équipes de la phase 1 si vous le désirez.

Soyez avisé que l'enseignant vous remettra la phase 3 avant l'évaluation de la phase 2. Ne prenez pas de retard!

Ateliers

Pour vous aider à réaliser cette tâche, l'enseignant donnera des ateliers au début des prochains cours. Voici les thématiques.

- Atelier 5: Programmation - Pointeurs de fonction + fonction lambda
- Atelier 6: Base de données – Rappel des formes normales
- Atelier 7: Programmation - Gestion de clavier et événements dans Panda3D
- Atelier 8: Programmation - UI dans Panda3D
- Atelier 9: Programmation - Singleton + Connection à la BD : paresseux versus pressé (*lazy vs eager*)

Gardez en tête les techniques apprises dans la phase 1, notamment...

- Patron de conception: *Data Transfert Object* (DTO) + *Data Access Object* (DAO)
- Utiliser GIT en équipes (*modèle central*)
- Organiser un projet d'envergure

On s'attend à ce que les équipes mettent en application ce qui est appris aux ateliers. Ces aspects seront évalués.

Livrable

Vous devrez remettre le dossier original du jeu mais il contiendra de manière ordonnée :

1. Script SQL nommé *creationTableNiveau.sql* qui contiendra les commandes de création des tables et de données.
 - 1.1. On devra être capable d'exécuter le script.
2. Le jeu modifié qui aura un nouveau menu de sélection de niveaux.
 - 2.1. Le code existant du jeu doit être modifié le minimum possible.
 - 2.2. Le nouveau code doit être organisé et isolé du code existant.
3. Le code ajouté, commenté et qui devra être organisé en module. Par exemple, mais non limité à...
 - 3.1. Module de lecture/écriture des niveaux.
4. Le programme de conception de niveaux *TankemEditor*.
 - 4.1. Il doit être accompagné d'une courte documentation expliquant comment il fonctionne.

Chacun des points est détaillé dans la section suivante. Votre organisation et la structure seront évaluées.

Division de la tâche

Voici les tâches principales. Chacun de vous devra assumer la responsabilité d'au moins une section comme dans la première phase.

1. Construction et gestion de la base de données + DAO/DTO + intégration DAO/DTO à l'éditeur de niveaux.
2. Intégration DAO/DTO à *Tankem* + création d'un menu dans *Tankem*.
3. Production de l'éditeur de niveaux (*2 personnes*)

Spécifications techniques – gestion des données

Script SQL

Ce script contiendra les instructions de création des tables de la base de données. On doit être capable d'exécuter ce script et les tables/données sont créées dans Oracle.

Pensez comment désigner vos tables qui accueilleront les données des niveaux. La forme normale 3 doit être respectée pour cette partie.

Finalement, votre script de création et la table doivent contenir 3 niveaux complets différents et intéressants.

Spécifications techniques – modification du jeu

Création d'un menu de sélection de niveau

Tankem contient un menu principal. Si on décide de jouer, au lieu d'aller directement dans le jeu, on doit aller dans un menu intermédiaire qui nous permettra de choisir le niveau dans lequel on jouera.

Voici ce qu'on doit voir dans ce menu :

- En premier choix, le joueur pourra choisir un niveau généré au hasard (même mode avec lequel le jeu a été livré).
- Tous les titres des niveaux actifs de la base de données doivent être affichés.

- L'ordre de classement des niveaux doit être en ordre alphabétique.

Un joueur, à l'aide des flèches ou de la souris, doit pouvoir choisir un niveau. Lorsqu'il confirme avec la touche entrée (*enter*), la partie débute normalement.

Si une erreur à la connection à la base de données se produit, seule l'option de création de niveaux au hasard doit être affichée. De plus, un message indiquant qu'une erreur de connection à la base de données s'est produite doit être affiché.

Pour cette partie, le style et l'élégance du menu ne sera pas particulièrement évalué, mais sa fonctionnalité le sera (lisibilité, clarté de l'information, facilité d'utilisation et bugs). Aussi, assurez-vous d'utiliser correctement les DAO/DTO.

Spécifications techniques – Programme de conception de niveaux

Avec la méthode de votre choix, vous devrez faire un éditeur de niveau. Cet éditeur permettra de concevoir des niveaux pour ensuite les insérer dans Oracle.

On ne permettra pas la modification de niveaux existants. Notez que vous avez la possibilité et les connaissances pour faire l'éditeur en 3D, mais ce n'est pas obligatoire. Par exemple, vous pourriez utiliser *tkinter* en python, une autre librairie ou même un autre langage de programmation, une page web... vous décidez.

Peu importe la solution que vous choisissiez, on se posera les questions suivantes pour l'évaluation :

1. Est-ce que l'éditeur fait son travail (possible de créer un niveau et modifier tous ses paramètres)
2. Est-ce que c'est facile à utiliser?
3. Est-ce que c'est robuste et fiable (nombres et importance des bugs)
4. Est-ce que c'est un code de qualité? (patrons de conception, lisibilité, commentaire)

Description d'un niveau

Un niveau a toujours une grosseur entre 6x6 et 12x12 cases et n'est pas obligatoirement carré.

Chaque case a un type associé. Voici les types possibles :

- Plancher (avec ou sans arbre)
- Mur fixe (avec ou sans arbre)
- Mur animé verticalement (avec ou sans arbre)
- Mur animé verticalement inversé (avec ou sans arbre)

De plus, 2 cases doivent être désignées explicitement pour l'emplacement initial des joueurs 1 et 2. Un joueur peut débiter sur une case de n'importe quel type.

Finalement, un niveau est décrit de la manière suivante :

- **Nom** : Le niveau doit être nommé (maximum 20 caractères)
- **Date** : Une date de création

- **Status** : Un niveau peut avoir 3 états : *Actif*, *Test* ou *Inactif*.
 - *Actif* : Le niveau est utilisable par tous les joueurs
 - *Test* : Le niveau est jouable par l'équipe de développement uniquement.
 - *Inactif* : Le niveau n'est pas utilisation par aucun joueur dans *Tankem*.
- **Délai minimum d'apparition des items** : Un délai minimum après lequel un item apparaît.
- **Délai maximum d'apparition des items** : Un délai maximum après lequel un item apparaît.

Étant donné qu'on ne peut pas modifier un niveau existant, pour le moment, il n'y a pas moyen de changer le statut des niveaux autre qu'à la création.

Fonctionnement de l'éditeur

Voici une maquette de l'éditeur de niveau. Le format est une suggestion mais pas obligatoire.

↓

Grosseur X: 9

↑

↓

Grosseur Y: 9

↑

↓

Temps min: 3.0 s

↑

↓

Temps max: 20.0 s

↑

Statut: Public

Contrôle
 WASD Bouger
 Espace: Cyler tuiles
 CTRL: Cyler inverse

Titre
L'enfer sur terre

Mur	Plancher	Mur	Mur	Plancher	Mur
Mur animé	Mur	Plancher	Mur animé	Mur	Plancher
Plancher	Plancher	Plancher	Plancher	Plancher	Plancher
Mur	Plancher	Mur	Mur	Plancher	Mur
Mur animé	Mur	Plancher	Mur animé	Mur	Plancher
Plancher	Plancher	Plancher	Plancher	Plancher	Plancher

Grosseur X et Y : Ces paramètres sont contrôlés par des boutons qui seront à gauche et à droite du texte indiquant la grosseur. Dès que la grosseur est changée, on doit voir le nombre de tuiles du niveau s'adapter.

Temps min et max : (Pour indiquer la fréquence d'apparition des items) Ces paramètres sont contrôlés par des boutons qui seront à gauche et à droite du texte indiquant la grosseur.

Statut : Ce texte (ou bouton) indique le statut de la carte. Seulement les statuts *Actif*, *Test* et *Inactif* sont accessibles ici car *Inactif* sera réservé pour les développeurs. On doit pouvoir le changer en cliquant sur le bouton (méthode de votre choix).

Titre : On doit pouvoir changer le titre de la carte en cliquant dessus. Le nom d'un niveau doit être unique!

Manipulation des tuiles : Le curseur peut être un modèle 3D, une image ou un autre effet de votre choix. Pensez simple. Voici la liste complète des tuiles possible :

- Tuile vide (avec ou sans arbre)
- Tuile mur (avec ou sans arbre)
- Mur mobile (avec ou sans arbre)
- Mur mobile avec rythme inversé (avec ou sans arbre)

Placement des joueurs : On doit placer les joueurs 1 et 2.

Contrôle : Les touches qu'utiliseront votre éditeur doivent être affichées afin qu'on ne devine pas.

Si vous décidez de faire l'éditeur avec Panda3D, je vous recommande fortement d'utiliser la classe *map.py*, plus particulièrement la fonction *creationNiveau*, comme référence.

Politique d'évaluation

Elles sont les même qu'à la phase 1 avec 2 modifications :

- N'oubliez pas que les absences vont compter si vous en abuser.
- Si un membre sous-performe/surperforme de manière importante (environ 20% de différence par rapport aux autres) par rapport à son groupe, un ajustement individuel sera considéré à la *discretion de l'enseignant*.