|  |  |  |  |
| --- | --- | --- | --- |
| Développement d’application AVEC BASE DE DONNÉES  420-GED-HY  **Travail personnel #2**  **Création d’un éditeur de monde**  **(15 pts)** | | | |
| **Hugo St-Louis** | | **Bureau :** B 2344 | **Poste :** **2738** **hslouis@cegepsth.qc.ca** |
|  | | | |
| **Description** | | | |
| **But :** | Implémenter un éditeur de jeu de type RPG avec l’Entité Framework. | | |
| **Objectifs** | 1. Utiliser d’un modèle relationnel de base de données dans le but de créer une application. 2. Utiliser l’entité Framework pour gérer les éléments du CRUD d’un éditeur de jeu. 3. Utiliser l’entité Framework avec Windows Form. | | |
| **Durée :** | 15 h | | |
| **Pondération :** | Travail évalué sur 100 points, mais ramené sur 15 points. | | |
| **Remise :** | Le 4 novembre 2019 à 23 h 59. | | |
| **Note** | * Ce travail se fait en équipe de trois (3). * Conserver une copie de sécurité. Il est de **votre responsabilité** de conserver une copie de sécurité dans l’éventualité où la lecture des données serait impossible. Cette copie doit être **disponible sur demande**. * Il y aura un maximum de **10% d’enlevé** pour la correction du français sur les travaux remis à raison de **0.5% par erreur**. * Le plagiat, sous toutes ses formes, est interdit. * Un maximum de 100% est donné sur le ce travail. * La PIEA et la PDEA sont respectées quant à l’évaluation | | |

# Vue d’ensemble :

Le but de ce travail est de créer les éléments de base du jeu **Hugo Land.** Ce travail sera réutilisé tout au long de la session et permettra d’implémenter les éléments vues en classe.



Figure 1: Saisie d'écran du jeu final de Hugo Land

Vous trouverez avec l’énoncé du travail une version de départ de l’application que nous modifierons pour vous donner un aperçu du genre de jeu que nous réaliserons.

Globalement, voici les étapes de développement :

* Lors du **travail pratique #1**, nous réaliserons les fonctions de bases permettant d’interagir avec la base de données en utilisant l’Entity Framework.
* Lors du **travail pratique #2**, nous réaliserons un éditeur qui permettra de faire la conception des mondes. Ce travail réutilisera les fonctions développées dans le travail pratique #1.
* Lors du **travail pratique #3**, nous réaliserons le jeu en mode client-serveur.
* Lors du **travail pratique #4**, nous compéterons le jeu mais avec la gestion de la concurrence pour le rendre multijoueur.



TP3+TP4

INTERNET

Service SOA  
(TP#3)

Serveur

Éditeur de monde

(TP#2)

BASE DE DONNÉES

Libraire pour l’accès aux données   
(TP#1)

# Explication du travail pratique :

Le but de ce travail est de créer un éditeur pour le jeu **Hugo Land.** Cet éditeur sera un outil pour les concepteurs dans le but de créer des mondes. Ainsi, cet outil sera exécuté directement sur le serveur en lien avec la partie serveur que vous avez déjà fait. Je vous **conseille de créer un nouveau projet dans la solution** de votre TP#1, car **vous devez faire communiquer le TP#2 avec le TP#1**.

Pour vous donner un coup de main, j’ai adapté un éditeur de jeu existant. **Le code contenu dans la solution de départ n’est majoritairement pas le mien, vous pouvez donc le critiquer et le modifier tant que vous le voulez.**

Par contre, je vous conseille de partir de cette solution, car cela vous conscientisera à :

1. Utiliser du code existant qui ne respecte pas les standards du cégep. Ce qui sera le cas lorsque vous serez sur le marché du travail,
2. utilisé du code qui n’est pas parfait,
3. L’importance de bien commenter son code.

# Explication des tâches à accomplir :

## Connexion au jeu en mode administrateur :

## Lors du démarrage de l’application, une fenêtre apparaît pour permettre à l’utilisateur de se connecter. Seuls les utilisateurs administrateurs pourront ouvrir l’application.

## Créer un nouvel administrateur au jeu.

Une fois l’utilisateur connecté (créé manuellement dans la base de données), celui-ci doit être capable de créer un utilisateur avec les mêmes droits que lui. Ceci dans le but de partager la tâche de conception.

## Assigner un nouvel administrateur au jeu.

Une fois l’utilisateur connecté, l’administrateur doit être capable d’assigner dans une interface *Windows Form* les droits d’administrations à un autre utilisateur. Tous les utilisateurs doivent être listés dans une grille et l’utilisateur n’aura qu’à sélectionner celui qu’il désire assigner les droits d’administrateur.

## Ouvrir un monde

Dans une nouvelle fenêtre, vous devez permettre à l’utilisateur d’ouvrir un monde existant. Vous devez permettre de visualiser dans un « Combobox » la liste des mondes.

Lorsque vous appuyez sur OK, vous devez charger le monde dans l’éditeur.

## Créer un nouveau monde

Dans une nouvelle fenêtre, vous devez permettre à l’utilisateur de créer un monde avec tous les attributs nécessaires.

Lorsque vous appuyez sur OK, un nouveau monde doit se charger dans l’éditeur.

## fonctionnalités de l’éditeur :

Le but de cet éditeur est de permettre de créer des mondes plus rapidement. Vous devez donc programmer l’éditeur pour qu’il soit possible de sauvegarder les objets du monde qui sont placés dans le monde. Dans l’éditeur, j’ai créé une structure de données qui est en lien avec les éléments que l’on peut ajouter à un monde.

private Dictionary<string, Tile> \_ObjMonde = new Dictionary<string, Tile>();

Actuellement, cette structure de données est construire, mais n’est pas en lien avec les éléments déposés dans le monde. Vous devez réaliser cette tâche. Une fois faites, vous pourrez charger ou sauvegarder un monde.

Lors de la sauvegarde, vous devrez sauvegarder :

* Ajouter un objet
* Ajouter un monstre
* Ajouter un item

Il n’est pas nécessaire de sauvegarder le gazon, car nous considérerons que c’est la texture par défaut utilisée.

Lors de la conception de votre monde, je vous recommande de prendre en compte que chaque écran de jeu sera de 8 X 8.

## Bonus( 1 pt) :

**Plusieurs éléments :**

Vous devez permettre à l’utilisateur de superposer les éléments superposables. Par exemple, certains items peuvent être par-dessus un chemin, mais pas par-dessus un mur. Pour réaliser cette tâche, vous pouvez vous fier à l’attribut IsTransparent de la classe Tile.

**Mise à jour concurrente (1 pt) :**

Vous devez permettre à plusieurs utilisateurs de mettre à jour le monde simultanément. À quelques secondes d’intervalle, l’application vérifie s’il n’y a pas eu une mise à jour sur la base de données. Si oui, seules les modifications seront transférées vers l’application et apparaitront dans le monde en cours d’édition. Dans le cas d’une collision, à vous de voir ☺

## Évaluation du travail de vos collègues :

Dans le but d’être juste avec tous vos collègues, vous devrez évaluer la performance de vos collègues dans ce travail. Vous devrez décider, par consensus, la distribution du résultat du travail sachant que vous avez 300 % à distribuer entre trois étudiants.

1. Vous allez donc assigner un pourcentage de participation au travail à chaque membre de l’équipe. Par exemple, si les trois membres de l’équipe (Alice, Bob, Claude) ont travaillé équitablement, alors Alice aura 100 %, Bob 100 % et Claude 100 %. Par contre, si les trois membres de l’équipe (Alice, Bob, Claude) n’ont pas travaillé équitablement, alors on pourrait avoir Alice 100 %, Bob 130 % et Claude 70 %. Ce pourcentage servira à calculer la note associée à chaque étudiant de telle sorte que la note sera multipliée par ce facteur (Note du travail \* facteur). Ce pourcentage devra être décidé par consensus en équipe.
2. Si vous n’arrivez pas à obtenir un consensus, j’utiliserai un ratio entre le nombre de lignes de code utile développé (pas de commentaire, de vide ou de changement de ligne) par une personne divisée par le nombre total de lignes de code utile.

Dans tous les cas, vous devez me remettre un document (avec une page de présentation et table des matières si nécessaire) qui précise la distribution et explique votre choix.

# Devis à remettre :

Ce travail peut se faire en équipe maximum de trois(3) personnes. Dans le cadre du travail pratique #2, il vous est demandé de :

1. Le code source de votre application.

Ce travail devra être remis électroniquement sur le serveur *Team Foundation* au plus tard le 4 novembre à 23 h 59.

# Voici la pondération :

1. *Liaison avec le tp#1 (1* ***points****)* ;
2. *Gestion des utilisateurs (5* ***points****)* ;
3. *Ouvrir ou créer un monde (4* ***points****)* ;
4. *Création des objets du monde (5* ***points****);*
5. *Bonus (2* ***points****);*

Les critères d’évaluation sont :

- la complétude;

- la qualité fonctionnelle (les fonctions livrées donnent les bonnes réponses et sont correctes);