

Jeu de civilisation

Terratoura

Nadia Ben Shika, Mariia Vashchenko, Pavlo Kudrenko



Introduction



Le jeu se déroule dans un monde carré constitué de différents terrains : plaines, forêts, maisons et océans. Au début, le joueur doit collecter des ressources pour construire ses premières structures et assurer sa survie.

L'objectif est de développer une civilisation à partir de rien, en gérant la population, les ressources et les relations avec d'autres peuples. Expansion, diplomatie et stratégie militaire seront essentiels pour dominer et remporter la partie.

Terrain

- Terrain : cœur du jeu, gère les interactions et ressources
- Ressources : récolte de bois/blé, régénération auto si nécessaire
- Maisons : abritent 4 perso max, ont une résistance, détruites par tsunamis
- Tsunami : arrive à intervalles, tue perso non protégés

Défis :

- Organisation des classes
- Construction sur cases occupées
- Mauvais comptage des perso protégés
- Non-destruction des maisons après tsunami



Game

Fonctionnalités importantes

- La création et mise à jour des toutes les éléments de la jeu
- Le contrôle des événements comme :
 - Le déclenchement du tsunami
 - Les attaques ennemis
- Class qui relie l'interface et le backend

Difficultés

- D'arriver à l'idée qu'on a besoin de cette classe, ce qui nous a freiné partout
- Fonction update et le passages de données dans les autres updates des éléments
- Gestion des événements
- Gestion des ennemis

Fonctions

- void update(Event &e)
- void attackEnemies()



Personnages

Fonctionnalités importantes

- Se déplacer avec un algorithme de pathfinding (findWayA)
- Entrer dans une maison pour se protéger
- Collecter du bois ou du blé
- Être tué par des ennemis ou tsunami

Fonctions

- void **updatePersonage**(Event e, const vector<Maison> *maisons, Terrain *terre)
- vector<Pair> **findWayA**(int xStart, int yStart, int xFin, int yFin, const Terrain *terrain, bool enemy)

Difficultés

- Déplacement de personnage:
 - 2 versions
 - prendre en compte environnement
 - faire personnage de prendre une pause entre chaque déplacement

Fonctionnalités en développement

- Se protéger contre les ennemis
- Participation dans la construction des maisons



•	SDL2
>	include
>	lib
↳	JeuSDL.cpp M
↳	SDLAnimation.cpp
C	SDLAnimation.h
↳	SDLEvent.cpp
C	SDLEvent.h
↳	SDLIncident.cpp
C	SDLIncident.h
↳	SDLJeu.cpp
C	SDLJeu.h
↳	SDLMenu.cpp
C	SDLMenu.h
↳	SDLMenuJeu.cpp
C	SDLMenuJeu.h
↳	SDLMessage.cpp
C	SDLMessage.h
↳	SDLSprite.cpp
C	SDLSprite.h

SDLJeu

Difficultés

- A optimiser les affichages pour avoir peu de memoir utiliser
- A trouver les bons tailles pour affichage
- A créer menu principal avec ses fonctionnalités

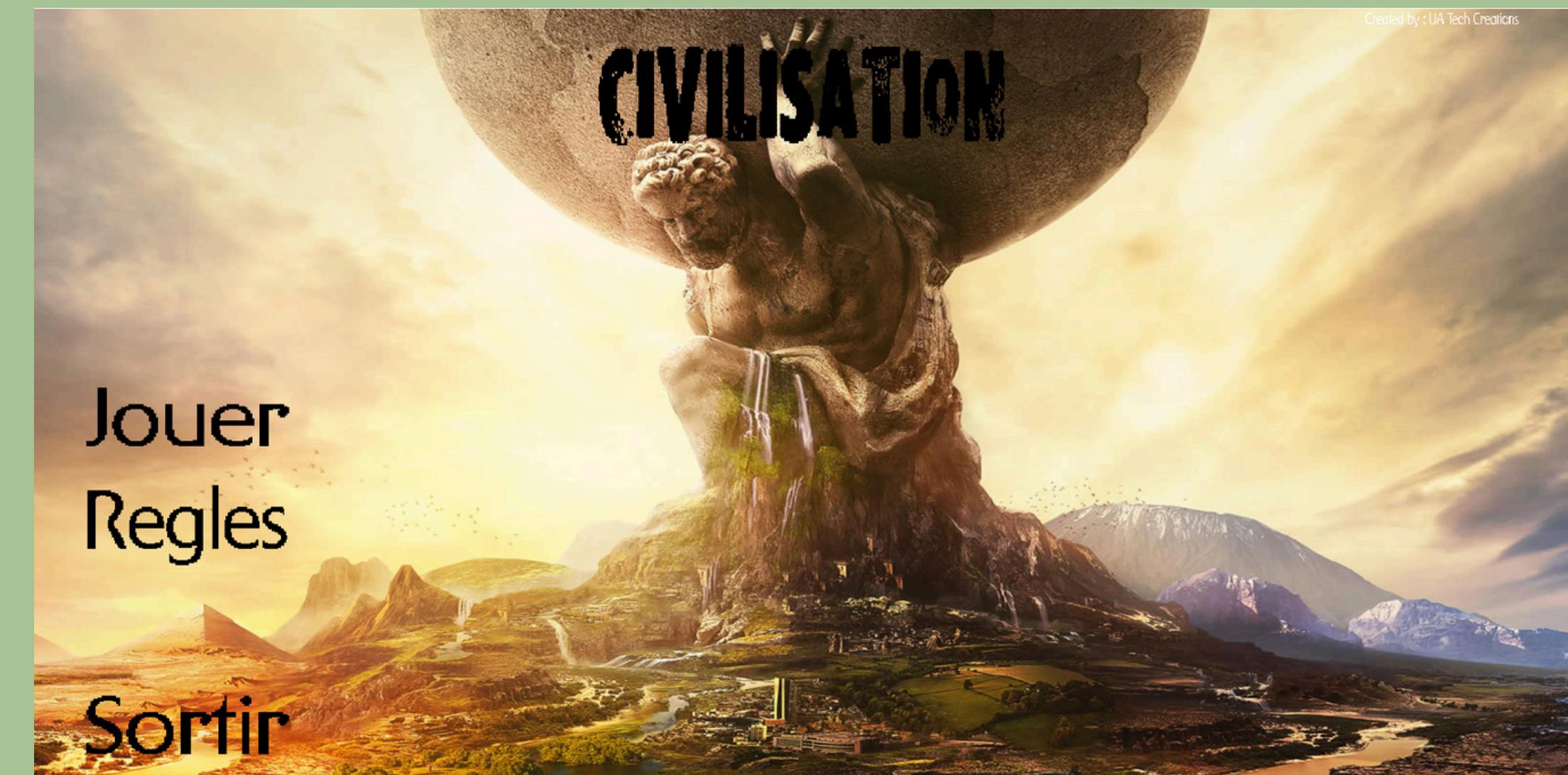
Fonctionnalités importantes

- **Tout les affichages :**
 - Affichage menu prencipal
 - Affichage de menu maison et menu dans la jeu
 - Affichage des icnidents
 - Animation des action de personages
 - Affichage des messages
- **Gestion de la musique**
- **Gestion des evenements**

Est un classe noyau de tout affichage de jeu. Il fait sur la principes de biblioteques de SDL2.

Fonctionnalités en développement

- Ameliorer affichage de tsunami pour le sortir de mode de boucle
- Ameliorer l'animation des personages
- Ameliorer des images de jeu
- Ajouter animation pour processus de construction de la maison
- Ajouter des paramètres dans menu prencipal
- Ajouter l'animation des enemis
- Ajouter des melodies et des sons



Created by : UA Tech Creations

Conclusion

- 90 % des objéctifs est atteint!
- challenges:
 - répartition de travail
 - architecture de jeu
 - les updates
 - boucle qui fonctionne bien sans prendre tout la RAM



L'avenir de votre empire est entre vos mains!

Merci

