

TunnelGL

Projet de jeu OpenGL

Plan

Démonstration

Réalisation

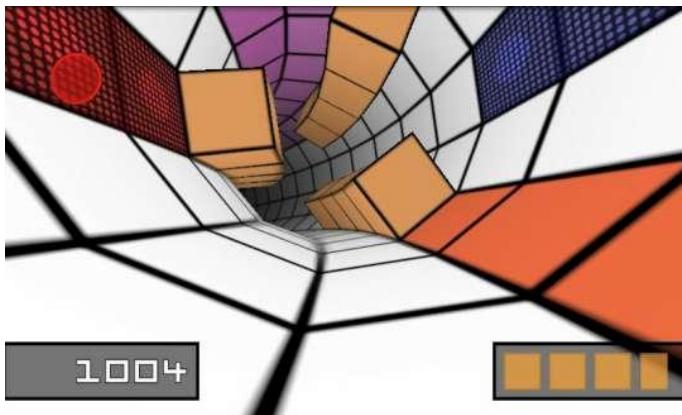
Débriefing

Rappel du jeu

But

Avancer le plus loin possible dans le tunnel en évitant les obstacles

Mock-up



Contrôles

| Contrôle | Effet |
|-------------------------|----------------|
| Touches directionnelles | Tourner |
| Espace | Turbo |
| Esc | Quitter le jeu |

Fonctionnalités

- Tunnel infini
- Décompte du score
- Augmentation de la difficulté

Démonstration

Réalisation

Tunnel

Tunnel

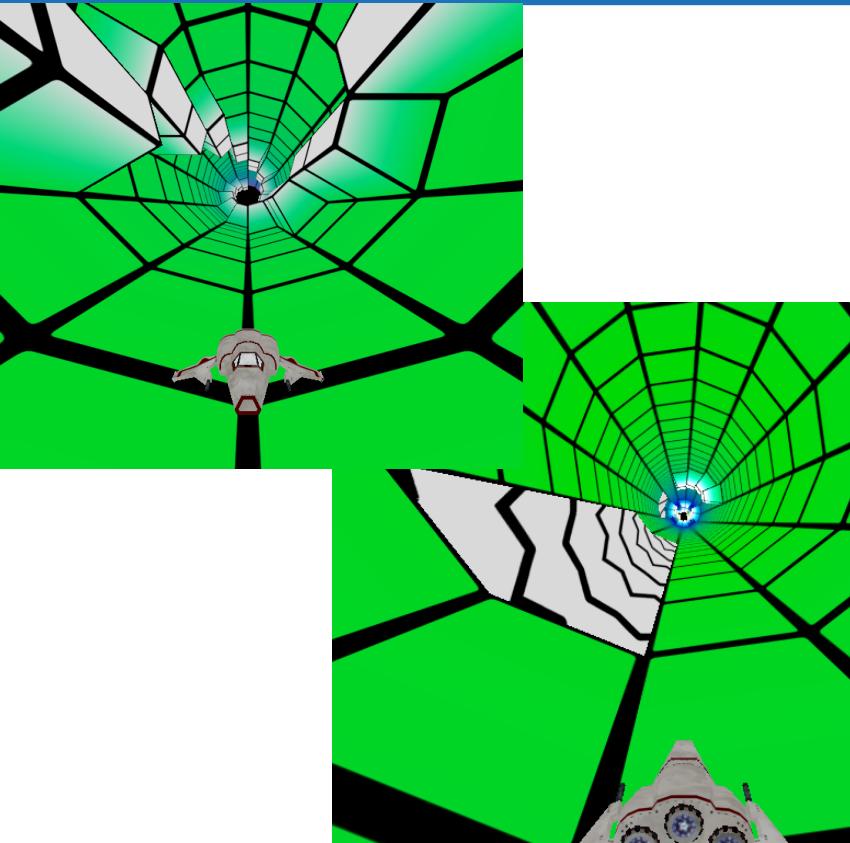
- ❑ Découpé en sections.
- ❑ Toujours 4 sections sont actives à un instant donné.
- ❑ À la fin d'une section, une autre section est générée.
- ❑ La couleur est générée en fonction de la distance en Z pour un effet lissé.

Section

- ❑ Cylindre comportant des trous et des obstacles.
- ❑ Générée par le chargement d'une matrice.



Tunnel : Obstacles et trous



Obstacle

- ❑ Pas des objets en eux-même mais générés en même temps que le mesh du Tunnel.
- ❑ Généré via l'ajout de 4 points situé à $R/2$ du centre du cylindre.

Trou

- ❑ Généré par l'omission des deux triangles formant la case.

Chargement d'une section

Fichier texte contenant toutes les sections

Une section est décrite par :

- 1ère ligne : identifiant de la section
 - 2ème ligne : difficulté (> 0)
 - 3ème ligne : identifiant section suivante
 - ◆ si -1 : pas de section particulière
 - matrice 24 x 12, codant les cases de la section :
 - ◆ 0 : terrain sûr
 - ◆ 1 : trou
 - ◆ 2 : obstacle

Déplacements

Déplacement en Z :

- ❑ Vitesse initiale de 70km/h
- ❑ La vitesse augmente au cours du temps.
- ❑ Boost : La vitesse augmente 4 fois plus vite. <Espace>

Déplacement latéral :

- ❑ Friction de 0.95 qui permet de limiter la vitesse angulaire.
- ❑ Vitesse angulaire dépend de la vitesse en Z
- ❑ Déplacement gauche et droite via <A-Q>/<D> ou <Left>/<Right>

Score et difficulté

Score :

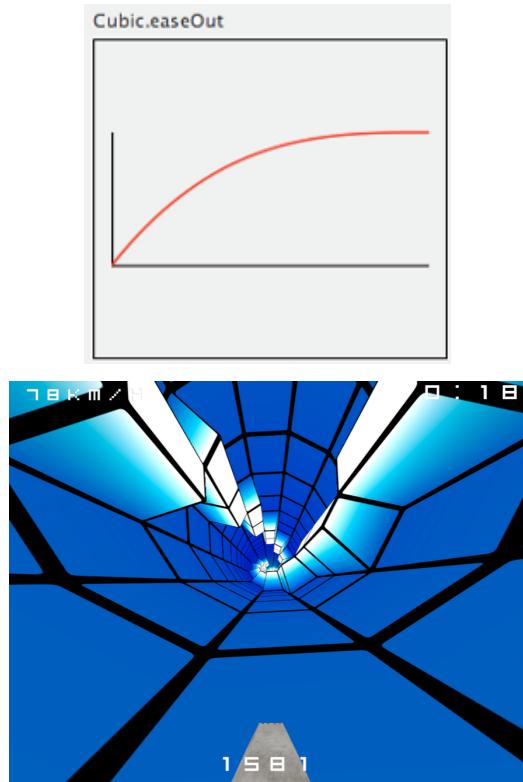
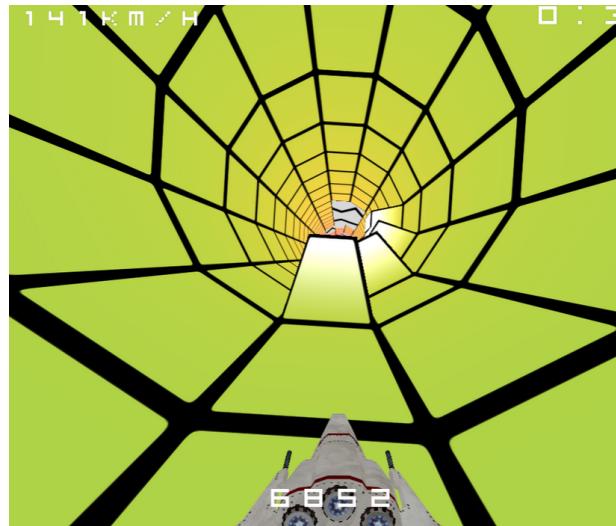
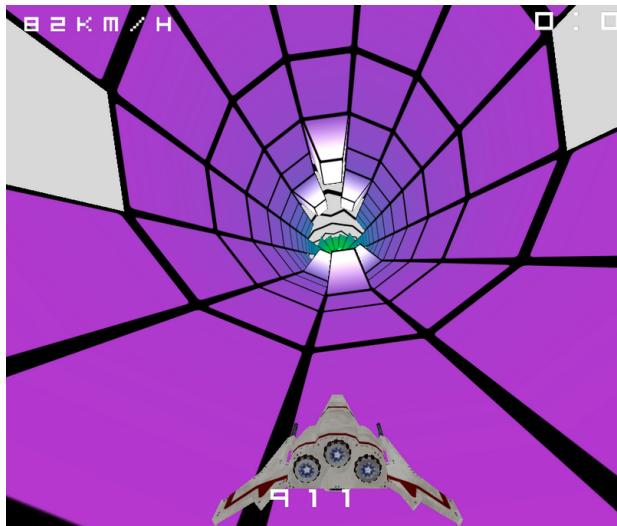
- ❑ Le score augmente avec le temps et la vitesse (`score += dt * speed.z * 1000`)
- ❑ Utiliser le boost double le score ajouté.

Difficulté progressive :

- ❑ La vitesse en Z augmente linéairement au cours du temps.
- ❑ Toutes les 10 sections terminées, la difficulté augmente de 1.
- ❑ Les sections chargées seront alors les portions de niveau \leq au niveau actuel.

Caméra

- ❑ Légèrement derrière le vaisseau pour le laisser apparaître.
- ❑ Se rapproche en vue cockpit lors d'un boost (Tweening).
- ❑ Vibre lors d'un boost.
- ❑ Se rapproche du vaisseau en fonction de la vitesse actuelle.



Écrans

Engine :

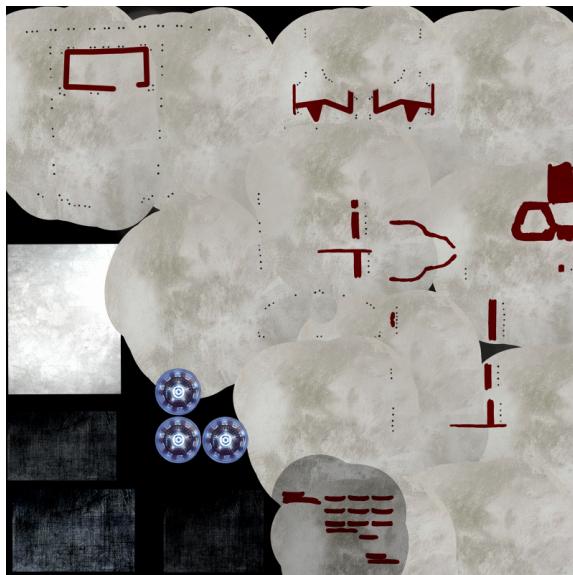
- ❑ Gère les écrans (Appel des méthodes nécessaires et passage à l'écran suivant)

Écrans :

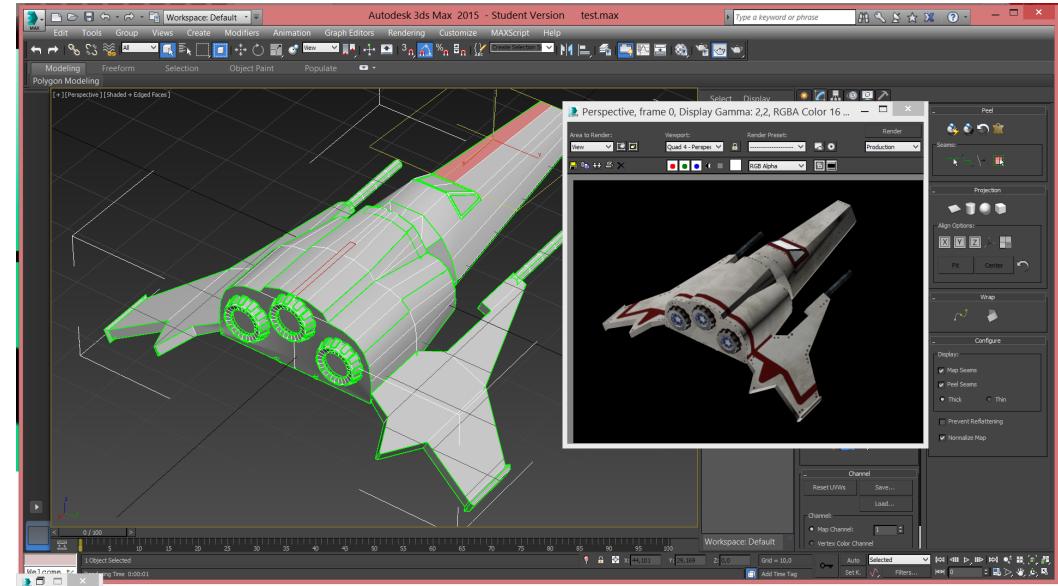
- ❑ Méthodes init, update, draw et onEnd.
- ❑ MenuScreen : écran d'accueil
- ❑ GameScreen : le jeu en lui même
- ❑ EndScreen : Affiche un résumé de la partie, permet d'en lancer une autre.

Modèle 3D

Vaisseau : modèle 3D + texturage (réalisation sous 3ds Max)

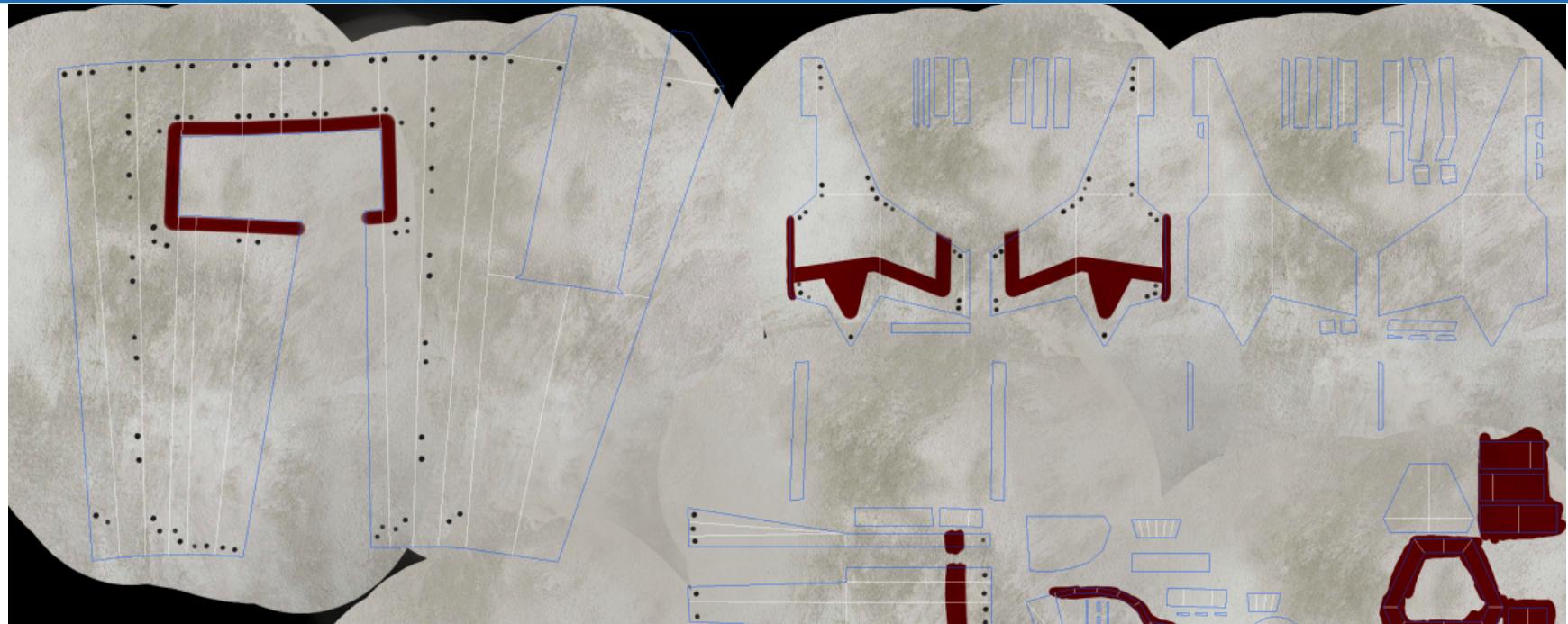


Aperçu fichier texture



Aperçu résultat final

Modèle 3D

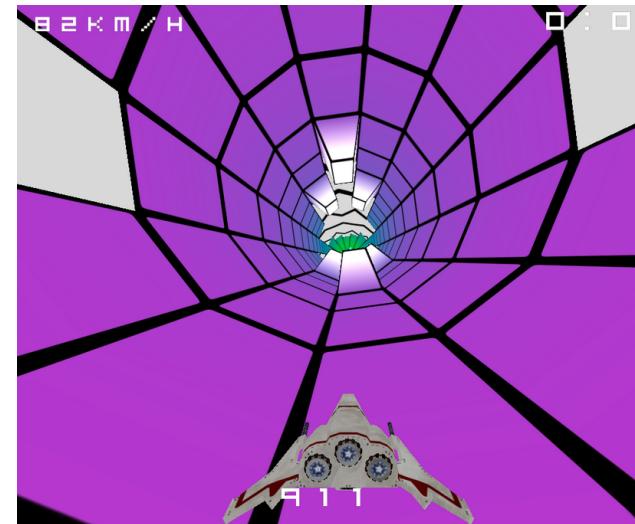


- + coordonnées uv générées automatiquement et plus facilement
- dépliage long à réaliser...

Texte

Texte :

- Chargement d'une texture DDS d'une police choisie : Visitor.ttf.
- Mappage UV de la texture pour chaque lettre.
- Transparence pour n'afficher que le texte.
- Affichage du score, de la vitesse et du temps à chaque frame.



P R E S S E N T E R T O S T A R T !

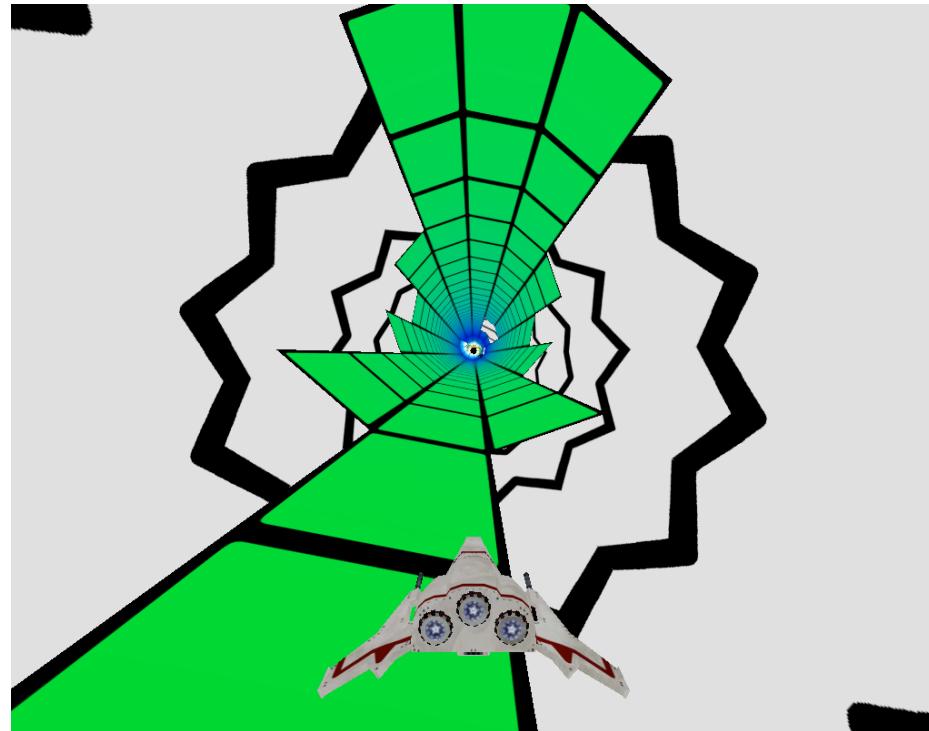
Son

Ajout basique de son grâce à irrKlang

- ❑ Musique durant une partie
- ❑ Bruitages lors d'un événement
 - Crash contre un obstacle
 - Chute dans un trou
 - Boost

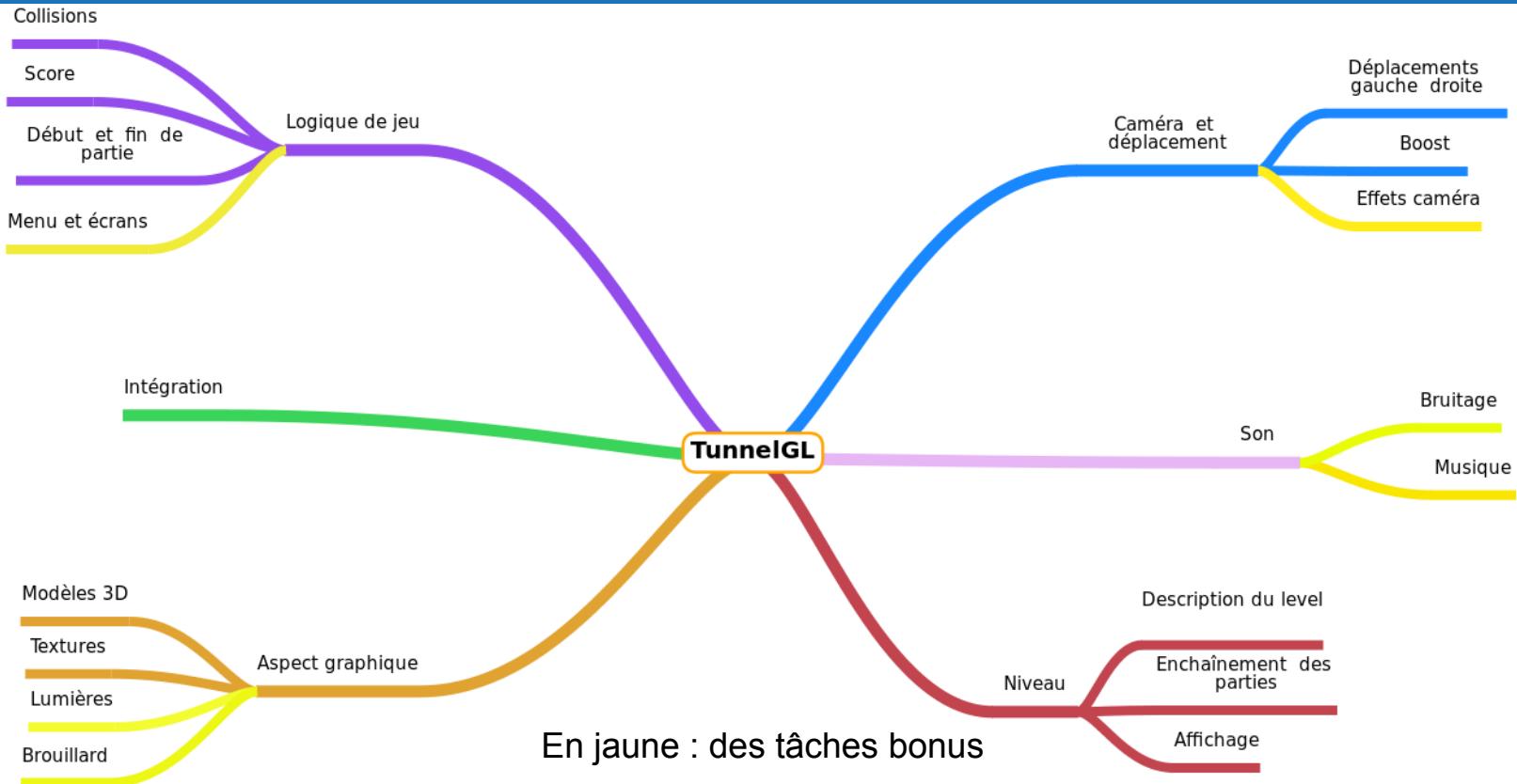
SkyTube

- Visible à travers les trous
- Tube qui suit la progression en Z du joueur
- Ne subit aucune rotation
- Améliore la visibilité des trous



Débriefing

MindMap d'avant-projet



Ce qui s'est bien passé

- ❑ Le jeu est au final fonctionnel et intéressant.
- ❑ Procéder par amélioration d'un prototype a accéléré le développement.
- ❑ La plupart des tâches bonus ont été implémentés.
- ❑ Ne pas prévoir trop de choses et classer l'importance des tâches a été pertinent.

Ce qui ne s'est pas bien passé

- ❑ L'utilisation de git a été un peu difficile.
- ❑ Le mappage UV du vaisseau a été long et contraignant.
- ❑ Le démarrage a été plutôt long (construction d'une API minimaliste).

Deux séances à une semaine d'intervalle pour un projet si intéressant, c'est dommage.

Le projet a eu lieu pendant une période difficile (rendu projet 2A, préparation examens).

Ce qui pourrait être amélioré

- Rajouter des propulseurs au vaisseau (visuel, son, ...)
- Rajouter encore plus de sections
- Avoir un mode “infini” et un mode “niveaux”
- Éditeur de sections
- Bump mapping du Tunnel

Merci de votre attention !

Annexe : Diagramme de classes

