



# LA GRANDE COURSE

*Jeu de course automobile*  
*Devis*

→ Trouvez un  
meilleur nom !

↪ Trouvez un nom à votre équipe!



LE HÉRO MASQUÉ

Préparé par  
Melino Conte  
Version 5.7 , 16 septembre 2023

## Table des matières

<b>HISTORIQUE .....</b>	<b>2</b>
<b>INTRODUCTION .....</b>	<b>3</b>
Objectifs du document.....	3
Portée du document .....	3
Définitions .....	3
Documents de référence .....	3
<b>ÉNONCÉ DU PROBLÈME.....</b>	<b>4</b>
Le mandat .....	4
Le nom du projet.....	4
<b>BALISES DE RÉALISATION.....</b>	<b>5</b>
Balises générales .....	5
INF707 – Fondements d'infographie appliquée .....	5
<b>ÉLÉMENTS ET SPÉCIFICATIONS .....</b>	<b>6</b>
Spécifications générales .....	6
Spécifications – INF707 .....	7
<b>INF707 - VOLET 1.....</b>	<b>8</b>
Objectif du volet 1.....	8
À remettre.....	8
Modalités de remise .....	8
Grille d'évaluation du volet 1.....	8
<b>INF707 - VOLET 2 (D'AUTRES SPÉCIFICATIONS SUIVRONT).....</b>	<b>9</b>
Objectif du volet 2.....	9
À remettre.....	9
Modalités de remise .....	9
Grille d'évaluation du volet 2.....	9
<b>INF707 - VOLET 3 (D'AUTRES SPÉCIFICATIONS SUIVRONT).....</b>	<b>10</b>
Objectif du volet 3.....	10
À remettre.....	10
Modalités de remise .....	10
Grille d'évaluation du volet 3.....	10
<b>INF707 - CONSIDÉRATIONS DE POO.....</b>	<b>11</b>
Les objets graphiques.....	11
Les objets chargés sur la carte graphique .....	11
Le véhicule.....	11
Le terrain ou les terrains .....	11
La caméra.....	11
Autre éléments .....	12
<b>INF707 - QUALITÉ DU CODE, STANDARDISATION, ORGANISATION, DOCUMENTATION .....</b>	<b>12</b>

## HISTORIQUE

<b>Version</b>	<b>Date</b>	<b>Auteurs</b>	<b>Description</b>
1.0	5 septembre 2006	François Jean	Énoncé et rédaction
2.0	22 août 2007	François Jean	Correctifs, ajouts
2.1	22 août 2008	François Jean	Correctifs, ajouts
2.2	27 août 2009	François Jean	Correctifs, ajouts
3.0	30 août 2011	François Jean	Correctifs, ajouts
4.0	3 septembre 2012	François Jean	Correctifs, ajouts
5.0	3 septembre 2013	François Jean	Correctifs, ajouts
5.1	3 septembre 2014	François Jean	Correctifs mineurs
5.2	1 septembre 2015	François Jean	Modifications mineures
5.3	5 septembre 2016	François Jean	Modifications mineures
5.4	6 septembre 2017	François Jean	Modifications mineures
5.5	6 septembre 2018	Pierre-Marc Bérubé	Modifications mineures
5.6	12 septembre 2021	Mykel Leclerc Brisson	Modifications mineures
5.7	16 septembre 2023	Melino Conte	Réajustement des parties

## INTRODUCTION

### *Objectifs du document*

Ce document a pour but de:

- définir le problème;
- définir le contexte d'utilisation et les objectifs du projet;
- établir les principes de fonctionnement;
- déterminer les orientations techniques et les contraintes de développement.

### *Portée du document*

Ce document s'adresse aux responsables du développement des projets, donc aux étudiants du Diplôme de Développement du Jeu Vidéo (DDJV) ainsi qu'à l'équipe professorale de première session. Suite à sa présentation, ce document servira aux développeurs comme devis de projet pour le cours **INF707-Fondements d'infographie appliquée**. Ce document fera aussi partie du dossier de développement du projet de session de ce cours.

### *Définitions*

**Plateau extérieur** ou **terrain** : noms génériques donnés à un plateau de jeu vidéo correspondant à un «paysage» (*landscape*) soit un terrain, des arbres, des immeubles, des montagnes, etc.

**Labyrinthe** : nom générique donné à un plateau de jeu vidéo possédant des murs, des planchers, possiblement un plafond et dans lequel l'action se déroule. Ce n'est pas nécessairement un «vrai» labyrinthe, il peut tout simplement s'agir d'un immeuble.

**Designer de jeux** : élabore le «scénario» d'un jeu ou une partie du scénario, ses intrigues et ses possibilités. Le **chef-designer** peut se comparer au réalisateur d'un film. Il travaille de concert avec les programmeurs pour que ceux-ci intègrent sa vision dans le jeu.

### *Documents de référence*

[ADX] Jean, François, Atelier DirectX 11 et Infographie

## ÉNONCÉ DU PROBLÈME

### *Le mandat*

Réaliser un jeu de course automobile **simple** mais complet où les aspects infographiques seront en évidence. Une piste de course ou un circuit de rallye peut utiliser une logique de jeu relativement simple et permettre de se concentrer sur les aspects graphiques. Tous les aspects infographiques de base y seront implantés ainsi qu'un certain nombre d'éléments de niveaux intermédiaires.

### *Le nom du projet*

**La grande course** (*The Great Race*) était le titre d'un film de Blake Edwards (la «Panthère rose») réalisé en 1965 dans lequel des véhicules souvent fantaisistes s'affrontaient sur une variété de routes et de terrains. Des péripéties, des aventures, de l'action, de l'humour sans oublier la petite histoire d'amour, c'était la recette de ce film.

Votre projet peut évidemment changer de nom et de recette, mais essayez de garder l'action!

## BALISES DE RÉALISATION

### *Balises générales*

1. Le jeu doit être réalisé en C++ sous forme de projet Visual Studio 2022.

### *INF707 – Fondements d'infographie appliquée*

1. Les éléments graphiques doivent être réalisés au moyen de DirectX et du SDK de DirectX (version 11), les éléments HLSL devront aussi être implantés directement ou au moyen des effets DirectX ou de leur équivalent. Notez que le SDK de DirectX fait partie du Windows SDK (distribué avec Visual Studio).
2. À l'exception de DirectX, **aucun** composant logiciel externe (contrôle, bibliothèque de fonctions, de programmes) n'est permis, qu'il soit privé ou du domaine public à l'**exception de bibliothèques de chargement de ressources** (Ex: ASSIMP) qui doivent tout de même être sujettes à approbation.

Note: des bibliothèques externes peuvent être permises pour les cours d'IA ou de Fondements scientifiques (ex: PhysX ou Havok) ou pour le cours de POO. Dans ces cas, le mentionner dans votre document de production.

3. Les éléments du projet décrits dans les pages suivantes constituent les spécifications du projet. Certains éléments des spécifications sont **volontairement** incomplets ou manquent de clarté. Quelques-uns d'entre eux seront expliqués en classe MAIS comme en entreprise, VOUS êtes responsable de vous assurer que vous avez bien compris les spécifications (le patron a toujours raison...).
4. Vous pouvez proposer des modifications à certains éléments du projet ou des livrables. Ces modifications doivent être approuvées au début des volets concernés et non en cours de développement.

## ÉLÉMENTS ET SPÉCIFICATIONS

### *Spécifications générales*

1. Le jeu devra permettre à un conducteur de déplacer son véhicule sur le circuit en respectant «les lois de la physique» (notre physique!). Des moyens doivent être choisis pour empêcher le conducteur de sortir de la scène ou de se retrouver en des endroits où la scène n'est pas définie.
2. Un certain nombre d'objets, en plus de la voiture devront être affichés. Ces éléments seront **importés** et seront des objets à votre choix (arbres, immeubles ou autres éléments de décor).
3. **Une seule** scène sera nécessaire.
4. Par contre, la scène devra utiliser un algorithme d'organisation des objets de la scène (*octree* ou autre). L'application devra pouvoir déterminer quelle est la «zone» active et n'afficher que les objets associés à celle-ci. Ce point important sera présenté en classe.
5. Les déplacements de véhicule et de caméra en fonction des collisions doivent être implantés de façon réaliste et «agréable».
6. La caméra devra pouvoir être à la 1<sup>ère</sup> personne (devant le véhicule ou dans le véhicule) **et** à la troisième personne (derrière le véhicule ou «ailleurs»).
7. Le positionnement du véhicule par rapport au terrain devrait être «correct» (pas nécessairement parfait).

## Spécifications – INF707

1. Le terrain devra être rendu d'une façon permettant de bien exploiter les différentes techniques de rendu décrites en classe.
2. Le terrain devra utiliser un minimum de 3 textures différentes.
3. Le rendu des textures doit être de qualité et éviter les pièges reliés au rendu de grandes surfaces.
4. Un «tunnel» (ou quelque chose du genre) doit être situé sur votre circuit, l'éclairage dans le tunnel n'est pas celui du reste du circuit.
5. Les limites de la scène doivent être agréablement camouflées.
6. Un petit menu permettant de «pauser» l'application et de permettre de quitter et au besoin de modifier les paramètres doit pouvoir s'afficher en réponse à une touche spécifique (par exemple la touche ÉCHAPPEMENT-*ESCAPE*). Ce menu doit s'afficher SUR le reste de l'affichage.
7. La vitesse du véhicule ainsi que le temps écoulé devront être affichés à l'écran en permanence. Vous pouvez remplacer ou compléter ces informations par d'autres.
8. Une boîte de ciel (ou l'équivalent) **doit** être utilisée et rendue de façon réaliste.
9. Au moins un élément ou un effet spécial devra être affiché au moyen de panneaux (*Billboard*).
10. La technique d'éclairage devra être au minimum du Blinn-Phong (minimum qualitatif).
11. Outre les techniques de rendu de base, des techniques «SPÉCIALES» devront être implantées. Nous utiliserons un système de points de complexité, vous devez en avoir **minimum 6, dont 1 post effect**.

Si plusieurs sont implémentés, les techniques les mieux implémentées (qui ont la meilleure note) seront choisies pour évaluer la note de cette partie.

Voici quelques exemples et leur valeur en points de complexité,

- a. 1 point : Normal/Bump Mapping, Texture Masking, Transparents
- b. 2 points : Ombres (Shadow Map), Projective Decals, VFX Ribbon, Lens Flares, Supersampling/MultiSampling Anti-aliasing, Parallax Mapping
- c. 3 points : Ombres volumétriques, Réfraction

Post effects :

- d. 1 point : Screen shake, Radial Blur, Chromatic Aberrations, Vignette, Filmic Grain, Lens Distortion, White Balance/Color-grading
- e. 2 point : Bloom, Painterly effects, Screen Space Ambient Occlusion
- f. 3 points : Depth of Field, Screen Space Reflections, FXAA
- g. 4 points : AMD FSR

Un exemple de choix pour une course sous-marine pourrait être :

Transparents(1) + Réfraction(3) + Bump Mapping(1) + Color-Grading(1) + Vignette(1) = 7 points.



## INF707 - VOLET 1

*Objectif du volet 1*

Réaliser le design de l'application soit déterminer l'architecture du moteur à développer, les formats des données, les éléments à afficher, l'organisation logique des données. Un document décrivant ces choix doit être complété pour la fin du volet.

*À remettre*

Le document de design.

*Modalités de remise*

Ce travail est à remettre d'ici au **dimanche 15 Octobre 2023 à minuit**.

*Grille d'évaluation du volet 1*

<b>Éléments "souhaités" dans le volet</b>	<b>Présent</b>	<b>Faible</b>	<b>Absent</b>
1- Une présentation générale expliquant les objectifs du jeu	10/10	5/10	0/10
2- Révision des contraintes et objectifs (rien de fixé dans le béton)	10/10	5/10	0/10
3- Organisation du programme (diagramme ou texte)	10/10	5/10	0/10
4- Description complète des modules	10/10	5/10	0/10
5- Liens entre les modules (diagramme)	10/10	5/10	0/10
6- Format du terrain (modèle de données)	10/10	5/10	0/10
7- Objet Caméra	10/10	5/10	0/10
8- Objet Véhicule	10/10	5/10	0/10
9- Gestion des zones (niveau de détail)	10/10	5/10	0/10
10- Liens avec le module de physique	10/10	5/10	0/10
11- Événements utilisateurs (user input)	10/10	5/10	0/10
12- Objets importés – Description	10/10	5/10	0/10
13- Prototypage des données (données internes)	10/10	5/10	0/10
14- Prototypage des traitements (cas d'utilisation)	10/10	5/10	0/10
15- Options du jeu	10/10	5/10	0/10
16- Logique du jeu (vitesse, collisions, accidents, dérapage...)	10/10	5/10	0/10

**Note:** Certains des éléments ci-haut sont à titre d'exemple.

## INF707 - VOLET 2

*Objectif du volet 2*

Réaliser la première phase de l'application soit développer le moteur de base, implanter une première version du terrain, lire les données. Implanter une version de base des déplacements du véhicule et de la caméra.

*À remettre*

Le volet 2 sous forme d'un projet Visual Studio bien identifié. Les documents ainsi que les directives d'installation ou d'utilisation (au besoin) devraient s'y trouver. VOUS êtes responsable de vérifier la validité des sources et des documents.

*Modalités de remise*

Ce travail est à remettre d'ici au **lundi 13 Novembre à minuit**

*Grille d'évaluation du volet 2*

	Élément évalué <i>(note: le calcul utilise le même ratio pour tous les composants)</i> Réponse aux objectifs ( 10/20 ) Algorithme ( choix, clarté ) ( 4/20 ) Qualité du code (découpage, organisation) ( 3/20 ) Documentation "de base" ( 3/20 )	Valeur
	<b>Simulation</b> (Ces composants pourront être améliorés au volet 3)	<b>45%</b>
	L1 Déplacement du véhicule (contrôles fonctionnels)	15%
	L2 Positionnement de la caméra par rapport au véhicule (une caméra finie)	15%
	L3 Collisions et/ou évitements (Les objets qui devraient ont des collision)	15%
	<b>Composants obligatoires</b>	<b>35%</b>
	O1 Niveau de détail (On cull les objets et le terrain derrière nous)	10%
	O2 Affichage de base du « circuit » et du terrain (texturé)	5%
	O3 Affichage d'objets importés (On les rends sans bug)	5%
	O4 Éclairage Blinn-Phong avec une source directionnelle	5%
	O5 Menu pause + redémarrage de « partie » (avec une touche c'est OK)	5%
	O6 Première version d'une partie (début et fin programmée)	5%
	<b>Programmation et documentation générale</b>	<b>20%</b>
	Le fichier "lisez-moi"	5%
	Identification des risques restants (Coupe / Rebudgété, par rapport au volet #1)	10%
	Qualité du code + stabilité du logiciel	5%

## INF707 - VOLET 3

*Objectif du volet 3*

Améliorer au besoin la modélisation de l'application. Améliorer au besoin les déplacements de véhicule et de caméra. Améliorer le rendu du terrain et des objets graphiques. Utiliser des effets de rendu de niveau intermédiaire.

*À remettre*

Le volet 3 sous forme d'un projet Visual Studio bien identifié. Les documents ainsi que les directives d'installation ou d'utilisation (au besoin) devraient s'y trouver. VOUS êtes responsable de vérifier la validité des sources et des documents.

*Modalités de remise*

Ce travail est à remettre d'ici au **jeudi 14 décembre à minuit**

*Grille d'évaluation du volet 3*

	Élément évalué (note: le calcul utilise le même ratio pour tous les composants, voir Volet#2)	Valeur
	<b>Simulation</b>	<b>30%</b>
	S1 Déplacement du véhicule	5%
	S2 Positionnement de la caméra par rapport au véhicule	5%
	S3 Positionnement du véhicule par rapport au terrain	10%
	S4 Collisions et/ou évitements	5%
	S5 Limites de la scène	5%
	<b>Affichage</b>	<b>30%</b>
	A1 Affichage du "circuit" et du terrain – rendu des textures	10%
	A2 Menu et affichage "2D"	5%
	A3 Boîte de ciel	5%
	A4 Panneau	10%
	<b>Effets</b>	<b>30%</b>
	Techniques - Chaque point de complexité compte pour 5%; une technique de 3 points de complexité vaut 15%.	30% (6 x 5%)
	<b>Programmation et documentation générale</b>	<b>10%</b>
	Le fichier "lisezmoi"	5%
	Qualité du code	5%

## INF707 - CONSIDÉRATIONS DE POO

### *Les objets graphiques*

Plusieurs classes d'objets graphiques sont à développer, soient

- TERRAIN ou TERRAINS
- TUNNEL et/ou PONT
- OBJETS IMPORTÉS (fixes ou mobiles)
- VÉHICULE
- "BILLBOARDS"

*Je souhaiterais:*

- Une classe de base commune, peut-être virtuelle? Toute autre implantation intelligente et innovatrice est évidemment la bienvenue.
- Une fonction d'affichage standardisée (le fameux *Draw ...*)
- Une ou des fonctions de positionnement.

Les objets graphiques devraient être regroupés dans des conteneurs (de pointeurs ou autres). Ces conteneurs pourront être, par exemple, une liste globale pour les objets qui sont toujours affichés, des listes d'affichages d'objets spéciaux (pour les effets) ou les conteneurs des zones.

### *Les objets chargés sur la carte graphique*

Certains objets chargés sur la carte graphique, principalement les maillages et les textures, sont utilisés dans plusieurs objets logiques. Il faudrait s'assurer que ces objets ne soient chargés qu'une **seule** et unique fois et non un nombre indéterminé de fois.

### *Le véhicule*

Le véhicule est probablement l'objet graphique le plus important. La plupart des éléments physiques y seront regroupés et utilisés (via une bibliothèque de «PHYSIQUE/MATH»). Porter particulièrement attention aux fonctions de positionnement, de mouvement, de vitesse, de «collision».

### *Le terrain ou les terrains*

C'est une classe «technique». **Très importante** au niveau de l'infographie. Elle n'est pas particulièrement intéressante en POO, mais...

J'aimerais qu'elle fasse un peu plus que d'afficher le terrain; par exemple, nous donner la hauteur d'une position ou la normale associée à une position. Ces éléments peuvent aussi être intégrés à d'autres classes.

### *La caméra*

Même si elle n'est pas un objet graphique affiché à l'écran et que son positionnement est relativement simple, la caméra mérite souvent sa propre classe d'objet. Le but de

la caméra est d'initialiser la vision et de gérer les déplacements de la vision (donc de la caméra). C'est une petite classe mais elle gagne à être bien conçue parce qu'elle est responsable d'une bonne partie du réalisme de l'application.

Une bonne classe de caméra devrait offrir la possibilité d'avoir plusieurs caméras, (mais une seule «active» à la fois). On devrait pouvoir changer de caméra pour changer de point de vue (1<sup>ère</sup> personne vs 3<sup>ème</sup> personne par exemple) plutôt que de modifier radicalement la position d'une caméra unique.

### *Autre éléments*

ÉCLAIRAGE, TUNNEL, etc...

## INF707 - QUALITÉ DU CODE, STANDARDISATION, ORGANISATION, DOCUMENTATION

Je ne suis pas très difficile sur ces points (question de temps) mais j'apprécie toujours un code bien conçu, des commentaires brefs mais judicieux et peut-être (on peut rêver...) des diagrammes simples illustrant l'organisation du programme, des modules, des classes...