Simulateur de débit de voitures sur un tronçon d’autoroute

Maziero Marco – CIN4A

Lausanne - ETML

Durée : 6 semaines

Chef de projet : Jonathan Melly

Table des matières

[1 Spécifications 3](#_Toc308526316)

[1.1 Titre 3](#_Toc308526317)

[1.2 Description 3](#_Toc308526318)

[1.3 Matériel et logiciels à disposition 3](#_Toc308526319)

[1.4 Prérequis 3](#_Toc308526320)

[1.5 Cahier des charges 3](#_Toc308526321)

[1.5.1 Objectifs et portée du projet (objectifs SMART) 3](#_Toc308526322)

[1.5.2 Caractéristiques des utilisateurs et impacts 3](#_Toc308526323)

[1.5.3 Fonctionnalités requises (du point de vue de l’utilisateur) 3](#_Toc308526324)

[1.5.4 Contraintes 3](#_Toc308526325)

[1.5.5 Travail à réaliser par l'apprenti 4](#_Toc308526326)

[1.5.6 Si le temps le permet … 4](#_Toc308526327)

[1.5.7 Méthodes de validation des solutions 4](#_Toc308526328)

[1.6 Les points suivants seront évalués 4](#_Toc308526329)

[1.7 Validation et conditions de réussite 4](#_Toc308526330)

[2 Planification Initiale 4](#_Toc308526331)

[3 Analyse 4](#_Toc308526332)

[3.1 Opportunités 4](#_Toc308526333)

[3.2 Document d’analyse et conception 4](#_Toc308526334)

[3.3 Conception des tests 4](#_Toc308526335)

[3.4 Planification détaillée 4](#_Toc308526336)

[4 Réalisation 4](#_Toc308526337)

[4.1 Dossier de Réalisation 4](#_Toc308526338)

[4.2 Modifications 4](#_Toc308526339)

[5 Tests 4](#_Toc308526340)

[5.1 Dossier des tests 4](#_Toc308526341)

[6 Conclusion 4](#_Toc308526342)

[6.1 Bilan des fonctionnalités demandées 4](#_Toc308526343)

[6.2 Bilan de la planification 4](#_Toc308526344)

[6.3 Bilan personnel 4](#_Toc308526345)

[7 Divers 4](#_Toc308526346)

[7.1 Journal de travail 4](#_Toc308526347)

[7.2 Bibliographie 4](#_Toc308526348)

[7.3 Webographie 4](#_Toc308526349)

[8 Annexes 4](#_Toc308526350)

# Spécifications

## Titre

Simulateur de débit de voitures sur un tronçon d’autoroute

## Description

Le projet consiste à créer un simulateur de tronçon d’autoroute en temps réel afin de pouvoir en calculer le débit de voitures en fonction de toute une liste de paramètres. Le trafic de voitures évoluera en temps réel en fonction des paramètres appliqués. L’affichage graphique du tronçon d’autoroute sera fait via le logiciel Unity 3D.

## Matériel et logiciels à disposition

* 1x PC de l’ETML
* 1x Laptop MacBook Pro personnel
* 1x documentation Unity 3D
* 1x licence gratuite Unity 3D
* Recherche web
* Microsoft Excel, Word et Powerpoint
* 1x dépôt Github pour le projet

## Prérequis

* Etre capable de tenir un journal de travail
* Avoir suivi les modules de gestion de projet
* Avoir suivi et réalisé les projets imposés à l’ETML
* Avoir suivi les cours de programmation structurée et orienté objet

## Cahier des charges

### Objectifs et portée du projet (objectifs SMART)

* + Programmer un logiciel de simulation de trafic basique les deux premières semaines en spécifiant uniquement la vitesse des voitures en km/h et la taille en km du tronçon d’autoroute.
  + Ajouter pendant les 4 semaines suivantes d’autres paramètres modifiables afin que le programme réagisse et s’adapte en temps réel.
  + Rédiger un cahier des charges et un journal de travail durant la période de projet afin d’assurer un suivi complet du travail.

### Caractéristiques des utilisateurs et impacts

* + Le programme doit être accessible à un public n’ayant pas de connaissances techniques particulières en informatique.
  + L’interface se doit d’être simple d’utilisation et le résultat doit être présenté en direct.

### 

### Fonctionnalités requises (du point de vue de l’utilisateur)

* + L’affichage de la simulation en temps réel en 3D
  + Une interface de configuration pour paramétrer le trafic
  + Les paramètres suivants pourront être modifiés en temps réel :
    - La distance de sécurité entre chaque voiture
    - Le nombre de voies de circulation
    - La limitation de vitesse sur le tronçon d’autoroute
    - Le pourcentage de chances sur un temps donné d’avoir un accident
    - La météo (ce qui influencera les chances d’accident)
    - La densité du trafic
    - La taille du tronçon d’autoroute
    - La présence de véhicules limités à une certaine vitesse
    - La fermeture d’une voie de circulation
  + La gestion automatique du changement de voie de circulation
  + La gestion automatique de la fermeture d’une voie de circulation
  + La détection des voitures envers les autres voitures
  + La gestion des accidents
  + La gestion automatique de la vitesse des voitures par rapport au trafic

### Contraintes

* + Nécessite l’installation du programme avant utilisation.

### Travail à réaliser par l'apprenti

* + Analyser et planifier les différentes tâches à faire. En outre l'analyse doit contenir une conception des fonctionnalités, un design de l'application et une liste de tests qui devront être réalisés avec les résultats attendus.
  + Mettre en œuvre l'environnement de développement.
  + Effectuer une batterie de tests afin de s’assurer de la stabilité du programme et éviter des erreurs inattendues.
  + Produire les documentations nécessaires à la mise en œuvre et à d'éventuelles modifications ultérieures par une tierce personne.
  + Tenir un journal de travail à jour afin de pouvoir lister toutes les étapes et améliorations apportées au projet.
  + Rédiger un cahier des charges qui mette en places les bases et objectifs du projet à réaliser.

### 

### Méthodes de validation des solutions

* + Vérification du comportement des véhicules lors de la simulation
  + Vérification la mise à jour dynamique de la simulation lors de la modification des paramètres de configuration.
  + Vérification du bon fonctionnement de tous les paramètres de simulation et détection des conflits potentiels entre eux.
  + Test d’installation et de fonctionnement du programme sur d’autres terminaux afin de s’assurer de la bonne exécution du code sur toutes les machines supportées par le programme.

## Les points suivants seront évalués

* Le rapport
* Les planifications (initiale et détaillée)
* Le journal de travail
* Le code et les commentaires
* Les documentations de mise en œuvre et d’utilisation

## Validation et conditions de réussite

* Compréhension du travail
* Possibilité de transmettre le travail à une personne extérieure pour le terminer, le corriger ou le compléter
* Etat de fonctionnement du produit livré

# Planification Initiale

La planification a été établie sur six semaines sans compter la semaine de vacances du mois de février. La liste des tâches a ensuite été placée dans la timeline afin d’aboutir à un diagramme de Gantt.

## Liste des tâches

Ci-dessous se trouve la liste de toutes les tâches globales du projet.

|  |
| --- |
| Rédaction du journal de travail |
| Rédaction du cahier des charges |
| Mise en place de l'environnement de travail |
| Planification et placement des tâches |
| Analyse technique du projet |
| Recherche de ressources pour le projet |
| Réalisation de la première version de base du projet |
| Réalisation / Ajout des fonctionnalités au projet de base |
| Tests du projet |
| Rédaction du manuel d'utilisation |
| Rédaction du rapport de projet |
| Divers |
| Création de la scène de base dans Unity 3D |

Ces tâches englobent les parties de planification, d’analyse de réalisation et de documentation du projet.

## Diagramme de Gantt

Le diagramme de Gantt a été créé en se basant sur les tâches à accomplir et à leur répartition dans le temps.

Le diagramme a été généré automatiquement dans le même fichier Excel que le journal de travail. Pour le voir, ouvrez le fichier de journal de travail et rendez-vous dans l’onglet « diagramme ».

# Analyse

## Opportunités

Ce paragraphe énumère la liste des difficultés potentielles de tout ordre :

Liste des compétences à acquérir ou approfondir

Liste du matériel à exploiter

Recherche d’informations particulières

Gestion du travail en équipe & collaboration

Ainsi que les solutions possibles

Si les spécifications de départ ne laissent pas de doutes sur la manière de réaliser un projet, ce chapitre ne fera que renvoyer le lecteur aux spécifications.

## Document d’analyse et conception

Ce paragraphe décrit le fonctionnement de manière détaillée.

Autant que possible de manière graphique, imagée, tableaux, etc.

Tous les cas particuliers devraient y être spécifiés…

Il s’agit d’y présenter les fonctionnalités à développer :

Découpage en étapes, en modules, en fonctionnalités, etc.

Formulaires, interfaces graphiques, pages web, etc.

Schémas de navigation, schémas événementiels, structogramme, pseudocode, etc.

Si le projet inclut une base de données :

Dictionnaire des données

Modèle conceptuel des données, modèles logique des données.

## Conception des tests

Ce paragraphe permet de spécifier la stratégie de test qui sera menée au point 5.1

Qui, quand, avec quelles données, dans quel ordre, etc.

## Planification détaillée

A ce stade, après l’analyse complète du projet, un planning détaillé et complet (avec tâches, sous-tâches, dépendances, durée, …) peut être finalisé.

Le planning détaillé doit s’inscrire dans le planning initial. Il faut que l’on puisse situer cette planification détaillée par rapport à la planification initiale.

# Réalisation

## Dossier de Réalisation

Cette partie permet de reproduire ou reprendre le projet par un tiers.

Pour chaque étape, il faut décrire sa mise en œuvre. Typiquement :

Versions des outils logiciels utilisés (OS, applications, pilotes, librairies, etc.)

Configurations spéciales des outils (Equipements, PC, machines, outillage, etc.)

Code source commenté des éléments logiciels développés.

Modèle physique d’une base de données.

Arborescences des documents produits.

Il faut décrire le parcours de réalisation et justifier les choix.

## Modifications

Historique des modifications demandées (ou nécessaires) aux spécifications détaillées.

Date, raison, description, etc.

# Tests

## Dossier des tests

On dresse le bilan des tests effectués (qui, quand, avec quelles données…) sous forme de procédure. Lorsque cela est possible, fournir un tableau des tests effectués avec les résultats obtenus et les actions à entreprendre en conséquence (et une estimation de leur durée).

Si des tests prévus dans la stratégie n'ont pas pu être effectués :

raison, décisions, etc.

Liste des bugs répertoriés avec la date de découverte et leur état:

Corrigé, date de correction, corrigé par, etc.

# Conclusion

## Bilan des fonctionnalités demandées

Il s’agit de reprendre point par point les fonctionnalités décrites dans les spécifications de départ et de définir si elles sont atteintes ou pas, et pourquoi.

Si ce n’est pas le cas, estimer en « % » ou en « temps supplémentaire » le travail qu’il reste à accomplir pour terminer le tout.

## Bilan de la planification

Distinguer et expliquer les tâches qui ont généré des retards ou de l'avance dans la gestion du projet. Indiquer les différence entre les planifications initiales et détaillées avec le journal de travail.

## Bilan personnel

Si c’était à refaire:

Qu’est-ce qu’il faudrait garder ? Les plus et les moins ?

Qu’est-ce qu’il faudrait gérer, réaliser ou traiter différemment ?

Qu’est que ce projet m’a appris ?

Suite à donner, améliorations souhaitables, …

Remerciements, signature, etc.

# Divers

## Journal de travail

Date, activité (description qui permet de reproduire le cheminement du projet), durée, liens et références sur des documents externes. Lorsqu’une activité de recherches a été entreprise, il convient d’énumérer ce qui a été trouvé, avec les références.

## Bibliographie

Références des livres, revues et publications utilisés durant le projet.

## Webographie

Références des sites Internet consultés durant le projet.

# Annexes

Listing du code source (partiel ou, plus rarement complet)

Guide(s) d’utilisation et/ou guide de l’administrateur

Etat ou « dump » de la configuration des équipements (routeur, switch, robot, etc.).

Extraits de catalogue, documentation de fabricant, etc.