

UNIVERSITÉ BLAISE PASCAL

Polytech Clermont-Ferrand



Projet de conception et programmation orientée objet
4 ème année

Département de Mathématique et Informatique
Filière : Génie Mathématique et Modélisation.

Le logiciel CAOStar
Rapport de Projet

Clavel Jean
Mahamadou Abdoul Jalil
Maisonneuve Justin
Mortabit Zouhair
Pose Céline

Année universitaire 2016-2017 .

Table des matières

I.	Introduction	2
II.	Présentation du sujet	2
III.	Planification du travail	2
IV.	Bilan	3
i.	Avancement du projet	3
ii.	Points à améliorer	3
iii.	Fonctionnalités complémentaires	3
iv.	Ressenti par rapport au projet	5
V.	Conclusion	6

I. Introduction

Dans le cadre de notre quatrième année en Génie Mathématique et Modélisation et plus précisément d'un cours d'UML (Unified Modeling Language), la société "ToutVéhicule" qui travaille dans le domaine de la conception de véhicules, nous a confié un projet à réaliser par groupe de cinq étudiants.

Cette entreprise souhaite développer son activité, ce qui ne peut se faire sans avoir recours à la CAO (Conception Assistée par Ordinateur). Après une étude de marché concernant les logiciels de CAO, la société "ToutVéhicule" préfère sous-traiter l'analyse, la conception et la réalisation du logiciel propriétaire "CAOStar".

II. Présentation du sujet

Le projet consiste à créer un logiciel de CAO en C++, et ce à l'aide du logiciel QtCreator. Le logiciel doit permettre au concepteur de véhicules de dessiner un véhicule à partir de formes élémentaires de couleur proposée dans une boîte à outils. L'utilisateur peut également déplacer le véhicule qu'il aura créé selon une direction choisie (horizontale ou verticale).

En vue d'une bonne intégration du logiciel, la société "ToutVéhicule" souhaite une interface homme-machine conviviale. Plus précisément, l'interface graphique qui s'affiche au démarrage comporte quatre options de menu à savoir "Composer", "Déplacer", "Arrêter" et "Quitter".

L'option "Composer" constitue le coeur même du logiciel. Elle permet de sélectionner à l'aide de la souris, des formes dans la boîte à outils et de les faire glisser dans la zone de composition afin de créer le véhicule.

Le bouton "Valider" permet quant à lui de valider cette composition : la fenêtre principale s'affiche alors avec le véhicule placé en son centre.

En ce qui concerne les deux boutons assurant la gestion de la suppression, il y aura un bouton "Cancel All" permettant l'annulation complète du véhicule commencé et un second bouton "Cancel" qui supprimera uniquement la dernière pièce placée.

L'option "Déplacer" quant à elle, permet d'obtenir un menu déroulant avec les options "Horizontal" et "Vertical". Une fois l'une de ces options choisies, le véhicule se déplace dans la fenêtre de dessin, dans la direction sélectionnée et rebondit contre les bords jusqu'à ce que l'utilisateur sélectionne l'option "Arrêter" du menu.

III. Planification du travail

La planification du temps de travail s'est déroulée en plusieurs étapes, que nous pouvons retrouver sur le diagramme de Gantt ci-dessous :

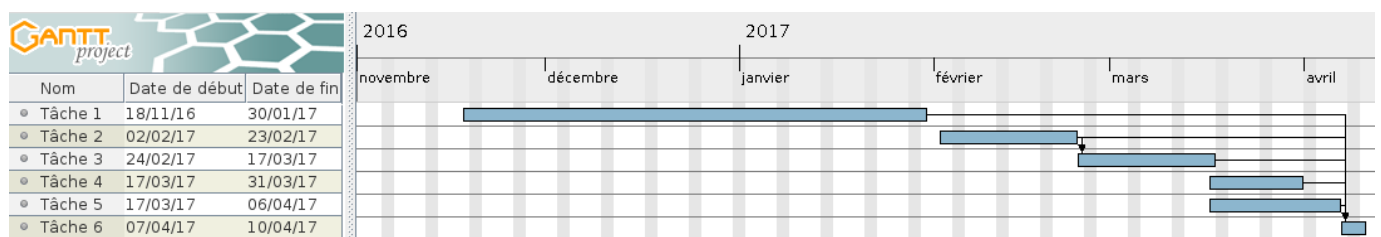


FIGURE 1 – Diagramme de Gantt.

- Tâche 1 : Partie UML
- Tâche 2 : Découverte de la bibliothèque Qt
- Tâche 3 : Codage du projet
- Tâche 4 : Modification de la partie UML
- Tâche 5 : Ecriture des rapports
- Tâche 6 : Finalisation du projet

Tout d'abord, pendant la première partie de l'année, nous avons uniquement travaillé sur la partie UML de ce projet qui consistait à modéliser notre logiciel à base de diagrammes (Diagrammes de classes, de séquence, ...). Pour cela, nous avons principalement réfléchi ensemble sur l'élaboration des diagrammes sous forme de brainstorming mais chacun avait à y réfléchir individuellement en dehors des séances afin de pouvoir proposer

ses idées la séance suivante et que celle-ci soit plus productive. Ce n'est qu'après la découverte de la bibliothèque Qt que nous avons finalement pu commencer le codage de notre logiciel. Concernant le codage, nous nous sommes répartis les différentes tâches. Chacun avait sa propre partie à coder, parties qui ont été ensuite mises en commun. Après avoir fini de coder notre logiciel, voyant que le codage ne respectait pas entièrement les diagrammes UML, nous sommes retournés modifier ces diagrammes afin qu'ils soient complètement cohérents avec le code. En parallèle, nous rédigeons les différents manuels relatifs à notre logiciel à savoir : le rapport de projet, le manuel développeur et le manuel utilisateur. Pour finir, nous nous sommes accordés quelques jours afin de finaliser notre rapport (relecture des rapports, conclusions, corrections des fautes d'orthographe). Avec une telle planification, nous avons pu mener à bien notre projet pour aboutir à un logiciel opérationnel.

IV. Bilan

i. Avancement du projet

Pour ce qui est de l'avancement de notre projet, nous avons un logiciel qui compile et qui répond à ce qui nous a été demandé de faire. En effet, au niveau de l'interface graphique, nous retrouvons les zones et boutons requis, et en ce qui concerne le fonctionnement, il permet de réaliser un véhicule, de le déplacer horizontalement ou verticalement et de quitter le logiciel. D'autre part, ayant atteint l'objectif avant la fin des délais, nous avons choisi de lui apporter quelques améliorations. Ces dernières sont la possibilité pour l'utilisateur de choisir la taille et la couleur des pièces qui composent le véhicule, la modulation de la vitesse grâce à une "réglette", et enfin le changement de direction de déplacement à tout moment.

Nous avons développé une autre fonctionnalité du logiciel à savoir la possibilité d'utiliser différents modèles de véhicules prédéfinis (voir III Fonctionnalités complémentaires). Mais le code n'étant pas "propre", nous avons préféré rendre notre logiciel sans cette fonctionnalité.

ii. Points à améliorer

Même si notre logiciel fonctionne, nous aurions souhaité apporter quelques améliorations concernant l'optimisation et la lisibilité du code. En effet, avec du recul nous aurions pu créer plus de fonctions dans le but de rendre notre code plus explicite. Par exemple, pour centrer le véhicule nous aurions pu faire une fonction `centrerVehicule()`. De plus, lors de la fermeture du logiciel, une erreur survient. Elle ne nuit pas à son bon fonctionnement mais nous aurions préféré pouvoir résoudre ce petit incident. Voici l'erreur affichée lors de cette fermeture :

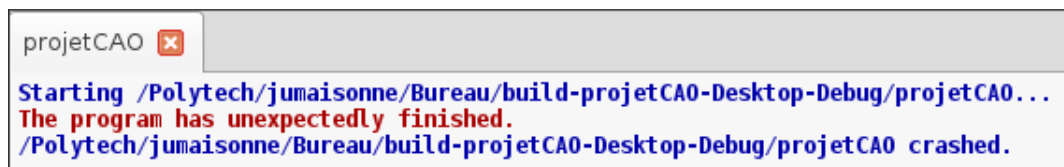


FIGURE 2 – Erreur lors de la fermeture du logiciel

Des recherches sur internet au sujet de cette erreur nous montrent que celle-ci serait due à un pointeur que nous n'avons malheureusement pas réussi à corriger.

iii. Fonctionnalités complémentaires

En ce qui concerne les fonctionnalités complémentaires, nous aurions voulu en ajouter quatre à savoir faire pivoter les formes graphiques, l'identification, l'utilisation de modèles de véhicules existants, et la possibilité d'enregistrer un véhicule créé par l'utilisateur comme modèle.

Pivoter les formes graphiques

Nous aurions souhaité, grâce à cette fonctionnalité, donner la possibilité à l'utilisateur, une fois la forme sélectionnée, de la faire pivoter. Ceci lui permettrait d'obtenir un design plus abouti du véhicule qu'il souhaite créer.

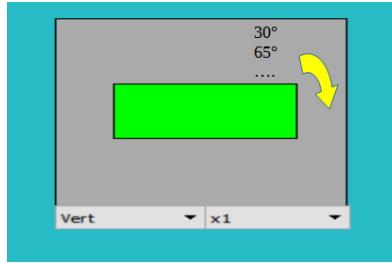


FIGURE 3 – Exemple de modèles par défaut

Identification

L'identification permettrait à l'utilisateur de se connecter avec un identifiant : Nom+Prénom et un mot de passe, à une session. L'intérêt de cette identification est de permettre à l'utilisateur de retrouver tous les modèles de véhicules qu'il aurait enregistrés.

Modèles de véhicules prédéfinis

Cette fonctionnalité offrirait à l'utilisateur la possibilité d'utiliser des modèles de véhicules prédéfinis en rajoutant une option de menu "modèles prédéfinis". En effet, lorsque l'utilisateur cliquerait dessus, une fenêtre apparaîtrait avec différents modèles déjà établis (cf figure 3). Il pourrait ainsi choisir un modèle en cliquant dessus, celui-ci se centrerait dans la zone de composition et l'utilisateur aurait la possibilité de poursuivre la composition ou alors de valider le véhicule pour ensuite le déplacer.

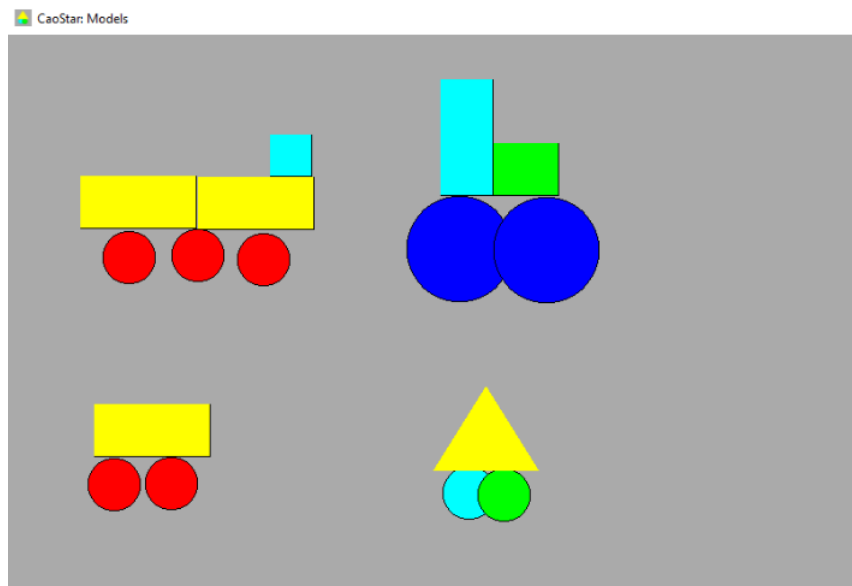


FIGURE 4 – Exemple de véhicules prédéfinis

Enregistrement de modèles de véhicules

Le but de cette fonctionnalité serait de permettre à l'utilisateur d'enregistrer des modèles de véhicules pour pouvoir les utiliser ultérieurement selon deux options : un enregistrement temporaire, c'est-à-dire que les modèles ne pourraient être utilisés que lorsque le logiciel serait actif et s'effaceraient automatiquement dès sa fermeture, et une autre option d'enregistrement permanent qui rendrait les modèles accessibles lors de l'identification de l'utilisateur.

iv. Ressenti par rapport au projet

Abdoul Jalil : Me concernant je dirai que le projet a été une expérience très enrichissante. Tout d’abord grâce au travail d’équipe, j’ai pu me rendre compte que travailler seul c’est bien mais qu’en équipe c’est mieux. Ce fut également l’occasion d’approfondir mes connaissances en C++ et de découvrir beaucoup de choses concernant la bibliothèque Qt, ce que nous n’avons pas forcément fait en cours. Ce projet a également comblé certains de mes temps libres que je consacre en général aux réseaux sociaux et à internet. J’ai donc pu me consacrer au codage et ainsi gagner en efficacité.

Céline : Pour ma part, j’ai trouvé ce projet très instructif. En effet, grâce à lui j’ai découvert le fonctionnement de l’interface de programmation QtCreator qui m’était jusque-là inconnu. De plus, la façon dont il a été présenté, c’est-à-dire comme étant une demande de la part de clients avec trois rapports distincts à rendre, a apporté une dimension plus concrète au travail que nous avions à faire. De ce fait, j’ai pu mieux cerner les attentes auxquelles nous devons répondre en tant qu’ingénieurs diplômés.

Zouhair : Pour moi, je suis très satisfait des apports dont j’ai pu bénéficier grâce à ce travail. Ce qui a été la source de ma motivation pour ce projet, c’est l’opportunité de travailler en utilisant une bibliothèque assez riche en C++ comme Qt et de livrer un produit dans un cadre scientifique. En plus de constituer pour moi une expérience en informatique, ce projet m’a aidé à développer de nombreuses compétences relationnelles. En effet, j’ai été amené à effectuer des réunions et à être en dialogue permanent avec les membres du groupe tout au long de ce travail. J’ai également amélioré mon autonomie et ma faculté d’adaptation en apprenant rapidement de nouvelles notions et en réfléchissant à des solutions d’implémentations adaptées aux besoins.

J’aurais aimé utiliser un logiciel de gestion de version (Git par exemple), qui permet de développer en parallèle de nouvelles fonctionnalités avec les différents membres du projet, afin d’éviter les problèmes liés à la gestion des versions durant la partie codage.

Enfin le projet s’est globalement révélé très instructif et très formateur pour moi.

Jean : Ce projet m’a permis d’apprendre à utiliser la bibliothèque Qt et ainsi de découvrir les nombreuses possibilités de création réalisable avec cet outil. De plus, le fait d’avoir un projet concret avec un cahier des charges précis nous a permis de mieux cerner les objectifs d’un ingénieur. Enfin, mené un projet de A à Z, de la partie UML jusqu’à la rédaction des différents rapports est un travail conséquent, c’est pourquoi le travail en équipe a été un élément très important que nous avons pu apprécier et qui nous a permis d’avancer rapidement et ainsi minimiser les pertes de temps.

Justin : Dans sa globalité, j’ai énormément apprécié ce projet. C’est quelque chose de concret qui a abouti à la création d’un vrai logiciel. J’ai pu suivre les étapes de sa création à savoir l’analyse UML de celui-ci puis son codage. J’ai développé de bonnes compétences en C++ ainsi que dans l’utilisation de l’interface de programmation QtCreator qui s’avère vraiment facile d’utilisation pour ce genre de logiciel avec interface graphique. J’ai beaucoup apprécié l’aspect modulable du projet, c’est-à-dire le fait que nous pouvions rajouter des fonctionnalités (et ça très facilement) à notre logiciel pour faire en sorte qu’il se différencie des autres. En tant que chef de projet, c’était une bonne expérience. Je pense avoir été à l’écoute de mes camarades et j’ai su déléguer et répartir le travail de sorte que le projet avance rapidement. Très bonne expérience.

V. Conclusion

Effectuer notre projet de quatrième année en travaillant sur le logiciel CAOStar a été globalement une expérience en informatique assez riche pour nous. En effet, cela nous a notamment poussé à répondre aux attentes de la société «ToutVéhicule», et nous a incité à produire un travail sérieux et des résultats performants.

Pour une première expérience dans le domaine du développement de logiciels, nous sommes fiers d'avoir abouti à la création d'un tel logiciel, d'avoir créé des fonctions de déplacement et de centrage du véhicule pour aboutir à des spécifications optimisées précises et à une implémentation satisfaisante.

Ce dernier point a été pour nous à la fois une difficulté à surmonter et un effort à réaliser qui s'est révélé très bénéfique. Nous avons en effet acquis grâce à cela une véritable capacité à nous documenter sur les outils performants de la bibliothèque Qt et des réflexes de décomposition d'un problème en sous problèmes à résoudre par les moyens du développement informatique.

Le logiciel qui a été développé a répondu aux besoins prédéfinis au début. Nous pouvons dire que notre organisation rigoureuse tout au long du projet, nous a permis d'effectuer un avancement significatif.

Par ailleurs, ce projet ne nous a pas seulement profité sur le plan technique. En effet, grâce aux différentes réunions que nous avons dû effectuer, entre nous et avec notre professeur de modélisation UML Madame BOUET, nous avons pu développer notre sens relationnel et acquérir de meilleures compétences de communication. Nous avons aussi pu développer le sens de l'analyse et de la réflexion que doit avoir un informaticien et un scientifique en général.