



Rapport de stage BUT 3

Refactoring, optimisation et modélisation de la logistique d'une chaîne de production industrielle.

Nicolas Franceschini

Tuteur pédagogique : Salima Benbernou

Tuteur entreprise : Jean-François Dion

15/06/2024

Résumé

Ce stage de fin d'année s'est déroulé au sein du Groupe Conseil Filion à Saguenay, au Canada. Le projet portait sur l'optimisation de l'IA des machines et de l'architecture orientée données, avec une mise en œuvre et expérimentation en langage Rust.

L'objectif principal était de développer une preuve de concept et d'expérimenter sur un cas réel, en vue de préparer la prochaine génération d'outils pour la simulation logistique des opérations de systèmes industriels.

Actuellement, le Groupe Conseil Filion utilise des solutions en Python pour simuler ces opérations. Le projet visait à transférer et améliorer cette technologie en utilisant Rust, un langage réputé pour sa performance et sa sécurité. Les travaux ont été réalisés principalement à distance, impliquant une collaboration étroite avec les membres de l'équipes de l'entreprise.

Le stage n'étant pas encore terminé, la solution est toujours en production et il n'y a donc pas encore de vrai résultats de performance à interpréter. Cependant, l'objectif est aussi de réaliser un outil plus moderne que le précédent, ce qui est actuellement le cas.

Abstract

This end-of-year internship took place at Groupe Conseil Filion in Saguenay, Canada. The project focused on the optimization of machine AI and data-oriented architecture, with implementation and experimentation in the Rust language. The main objective was to develop a proof of concept and experiment on a real case, with a view to preparing the next generation of tools for logistics simulation of industrial system operations.

Currently, Groupe Conseil Filion uses Python-based solutions to simulate these operations. The aim of the project was to transfer and enhance this technology using Rust, a language renowned for its performance and security. The work was carried out mainly remotely, involving close collaboration with the company's teams.

As the internship is not yet over, the solution is still in production, so there are no real performance results to interpret yet. However, the aim is also to produce a more modern tool than the previous one, which is currently the case.

Remerciements

Je tiens à exprimer ma profonde gratitude à toutes les personnes qui ont contribué à la réussite de mon stage.

Je remercie tout particulièrement M. Jean-François Dion, mon tuteur de stage au sein du Groupe Conseil Filion. Son expertise technique, sa disponibilité et ses conseils avisés ont été essentiels à la réalisation de ce projet.

Je souhaite également remercier M. Sylvain Boivin, professeur à l'UQAC et salarié du Groupe Conseil Filion, pour avoir proposé ce stage et pour nous avoir expliquer le projet et les besoin de l'entreprise. Son soutien et ses orientations ont grandement facilité le bon déroulement de mon travail.

Enfin, je remercie Mme Salima Benbernou, professeur à l'IUT de Paris Rives de Seine pour m'avoir suivis et répondu à mes questions pendant ce stage ainsi que cette année entière à l'étranger.

Merci à tous pour votre soutien et votre confiance.

Sommaire

Table des matières

I - Introduction.....	6
II – Environnement professionnel	8
1) Groupe Conseil Filion	8
2) L'équipe de développement.....	8
3) Les outils utilisés	9
III – Développement de la solution.....	10
1) Présentation du projet	10
2) ECS	11
3) Représentation en Graphes	12
4) Simulation de production	13
5) Phase de tests et validation	14
6) Problèmes rencontrés	15
IV - Conclusion.....	16
V - Lexique	17
VI - Webographie	18

I - Introduction

Mon stage de fin d'année au sein du Groupe Conseil Filion s'inscrit parfaitement dans la continuité de mes études à l'IUT Paris Rives de Seine, réalisées en échange à l'Université du Québec à Chicoutimi (UQAC). Située à Chicoutimi, au Canada, cette entreprise spécialisée dans les systèmes industriels et la simulation logistique m'a offert l'opportunité de travailler sur un projet innovant visant à optimiser l'intelligence artificielle (IA) des machines et à développer une architecture orientée données. Ce stage, réalisé principalement à distance, a permis de collaborer étroitement avec les experts de l'entreprise tout en explorant les potentialités du langage de programmation Rust.

L'un des défis majeurs dans le domaine des systèmes industriels est la simulation précise et efficace des opérations logistiques. Le Groupe Conseil Filion utilise actuellement des solutions en Python pour ce type de simulation. Cependant, avec l'évolution rapide des technologies, il est devenu crucial d'explorer des solutions plus performantes et sécurisées. Le projet confié avait pour but de répondre à cette problématique en développant une nouvelle architecture de simulation en Rust. Les principaux objectifs étaient de moderniser la solution déjà existante, d'optimiser l'IA des machines pour améliorer la précision et l'efficacité des simulations et de développer une architecture orientée données en Rust pour renforcer la performance et la sécurité des simulations.

Le travail confié s'est articulé autour de plusieurs missions clés, visant à mettre en place une infrastructure robuste et évolutive pour la simulation logistique. Il s'agissait d'abord de se documenter sur les technologies utilisées (Rust, bevy, ECS), puis de modéliser une chaîne de production sous forme de graphe orienté en Rust, à partir de fichiers déjà existants ou directement sur le logiciel. Parallèlement, un simulateur a été développé pour pouvoir simuler des changements dans une chaîne de production existante et observer les résultats, c'est l'aspect le plus important de ce stage. Enfin une phase de test et de validation par l'entreprise est prévue mais elle n'a pas encore eu lieu au moment où ces lignes sont écrites (il reste encore deux semaines de stage).

Ce stage m'a permis de mettre en pratique les connaissances acquises durant ma formation, en particulier dans le domaine de l'architecture logicielle mais aussi d'en acquérir de nouvelles comme le développement ECS ou encore le langage RUST et son moteur de jeu bevy. J'ai choisi cette mission car elle représentait une opportunité unique de travailler sur des technologies de pointe tout en contribuant à un projet concret avec un impact direct sur les opérations industrielles. Le projet était particulièrement intéressant car il abordait des problèmes complexes tels que la modélisation de graphes orientés et la simulation de systèmes industriels, tout en exigeant une rigueur technique et une approche méthodique. Les défis rencontrés,

notamment en termes de performance et de sécurité, ont enrichi mon expérience et m'ont permis de développer des compétences précieuses.

Ce rapport est structuré de manière à refléter le déroulement et les aspects clés du projet. La première partie présente l'entreprise et le contexte du stage, en décrivant le Groupe Conseil Filion et ses activités, ainsi que les enjeux du projet. L'organisation et la formation de l'équipe de stage ainsi que les outils utilisés seront également abordés. La deuxième partie se concentre sur l'aspect technique et concret du projet, notamment le développement en Rust et la mise en œuvre de l'architecture ECS. Les différentes missions qui m'ont été confiées seront abordées ainsi que les différentes solutions apportées et difficultés rencontrées.

II – Environnement professionnel

1) Groupe Conseil Filion

Le Groupe Conseil Filion, fondé en 1996, est spécialisée dans les services de consultation en ingénierie civile, répondant principalement aux besoins locaux de la région de Saguenay-Lac-Saint-Jean.

Groupe Conseil Filion propose une large gamme de services de consultation en ingénierie, notamment en ingénierie Civile (Projets d'infrastructures, développement urbain, conception et gestion de projets de construction), en ingénierie mécanique (Conception de systèmes mécaniques pour des installations industrielles) ou encore en consultation environnementale (Études d'impact environnemental, solutions de gestion des déchets, et projets de durabilité).

L'entreprise compte une vingtaine de salariés et j'ai eu la chance de travailler avec deux d'entre eux : Sylvain Boivin (Modélisation optimisation des procédés) ainsi que Jean-François Dion (Programmeur spécialisé en modélisation).

Les principaux clients du Groupe Conseil Filion incluent des administrations publiques, des entreprises industrielles majeures, et des promoteurs immobiliers. La fidélité de ces clients est le reflet de la qualité des services et de la réputation solide de l'entreprise.

2) L'équipe de développement

L'équipe de développement que j'ai rejoint lors de ce stage est composé de quatre autres stagiaires et nous sommes supervisés par Sylvain Boivin, qui en tant que product owner représente l'entreprise et ses besoins, ainsi que par Jean-François Dion qui lui est présent pour répondre à nos questions techniques et nous aider lors de la programmation de la solution logicielle.

Pour un déroulement fluide et continu du projet, nous avons adopté une organisation agile au sein de l'équipe. Les différentes réunions que nous avons eu ainsi que la communication permanente entre les membres de l'équipe nous a permis un bon suivis du projet et une adaptation rapide aux différents obstacles sans avoir à fixer des dates limites de rendu (à part bien sur la fin du stage).

3) Les outils utilisés

Pour mener à bien ce projet, nous avons eu recours à différents outils. Premièrement, nous avons choisis d'utiliser Discord pour communiquer car c'est un outil pratique et tous les membres de l'équipes y étaient familiers.

Pour stocker et suivre le projet, nous avons choisis GitHub car c'est un outil répandu, sécurisé et encore une fois l'ensemble des membres y est familier.

Ensuite, pour tous les outils de programmation, nous avons chacun utilisé notre propre machine en raison du travail en distanciel et le choix de l'environnement de développement nous as été laissé selon nos préférences, j'ai personnellement choisis d'utiliser RustRover, l'environnement de développement Rust de JetBrain.

Comme indiqué ci-dessus, il nous a été imposé d'utiliser le langage de programmation Rust pour sa sécurité et ses performance élevées ainsi que Bevy, un moteur de jeu pour la partie graphique. Il nous a également été imposer de programmer en ECS (Entité Composant Système) et non en objet comme nous avions toujours appris. Je reviendrais plus tard sur cette dernière, une partie lui est dédié plus loin (voir III-2).

III – Développement de la solution

1) Présentation du projet

Le premier jour du stage, le projet nous a été présenté par Mr Boivin lors d'une réunion en présentiel avec toute l'équipe. L'objectif est simple, reproduire en Rust une application python utilisé pour modéliser des chaînes de production puis réaliser des simulations sur cette dernière tout en la modernisant et l'améliorant. Le but n'est pas de recopier le code python en Rust mais bien de repartir de zéro et créer de toute pièce une nouvelle application ayant les mêmes fonctionnalités que la précédente, mais étant plus moderne et performante.

Il nous a été indiqué que les chaînes de production doivent être représentées sous forme de graphes orientés mathématiques (voir ci-dessous). Les nœuds des graphes représentent les machines et les arrêtes sont les différents moyens de stockage ou d'acheminement des ressources aux machines (ex : réservoir, tapis roulant, ...).

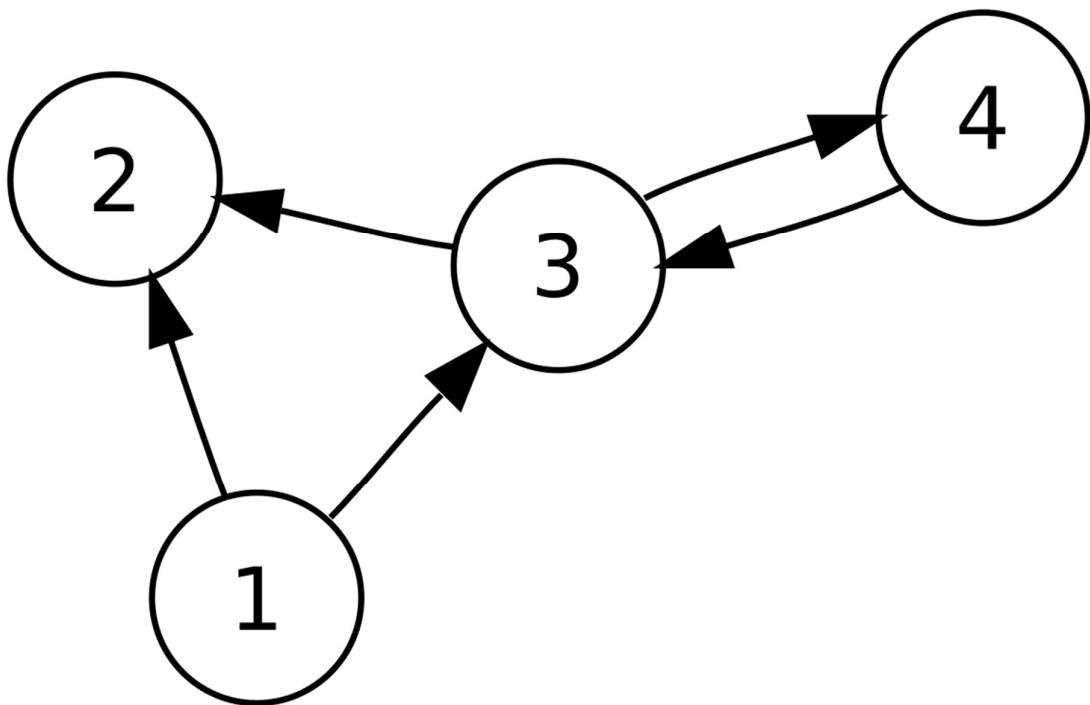


Figure 1 : Exemple de graphe orienté

Cette réunion a également été l'occasion de définir comment organiser notre travail, l'équipe et les différents outils utilisés (voir II-2 et II-3).

Comme indiqué précédemment, le projet doit être réalisé en utilisant le langage Rust, un nouveau langage pour l'ensemble des stagiaires et le moteur de jeu Bevy. De plus, il a

été décidé que le projet serait réaliser en programmation ECS (voir III-2) et non programmation orienté objet, encore une fois une nouveauté pour l'ensemble des stagiaires.

Ces trois notions étant nouvelles, une phase de documentation a été nécessaire pour l'ensemble de l'équipe pour se familiariser avec Rust, Bevy et ECS.

2) ECS

La programmation ECS est la principale notion que j'ai acquise lors de ce stage et c'est pour cette raison que je lui dédie une partie à part entière.

Le modèle Entity Component System (ECS) est une architecture de conception logicielle utilisée principalement dans le développement de jeux vidéo, mais aussi dans d'autres domaines nécessitant une gestion complexe des objets et de leur comportement.

Comme son nom l'indique, les trois principales composantes de ECS sont les entités, les composants et les systèmes.

Les entités sont des identifiants uniques qui représentent des objets dans le monde du jeu. Une entité n'a pas de comportement ou de données attachées par elle-même ; elle est simplement un conteneur pour des composants.

Les composants contiennent des données spécifiques à des fonctionnalités ou des aspects d'une entité. Dans notre cas, un composant pourrait représenter la position en x et y d'un nœud dans notre graphe, ou encore la quantité maximum d'entité qu'un arc peut contenir.

Enfin, les systèmes contiennent la logique qui opère sur les entités en utilisant leurs composants. Un système peut, par exemple, mettre à jour la quantité d'entité présente sur un arc en fonction de la quantité prise par la machine en sortie.

Cette façon de programmer combiné à un nouveau langage m'a plutôt perturbé dans les débuts. Etant habitué à la programmation orienté objet (POO), il fallait totalement changer ma façon de réfléchir aux différents problèmes.

Je ne dirais pas que je maîtrise désormais cette façon de programmer car je suis encore trop habitué à la POO mais c'est définitivement une connaissance que je souhaite approfondir dans le futur.

3) Représentation en Graphes

Après avoir bien cerné les objectifs du projet et m'être bien documenté sur l'ensemble des nouvelles notions, la première phase de réalisation de l'application a commencé.

Le principal objectif de cette première phase est de pouvoir représenter un graphe orienté (représentant la chaîne de production) sur l'écran. D'autres besoins ont également été présentés comme la création entier d'un graphe à partir d'un fichier JSON ou à l'inverse la sauvegarde d'un graphe dans un fichier JSON mais le besoin principal reste la représentation d'un graphe.

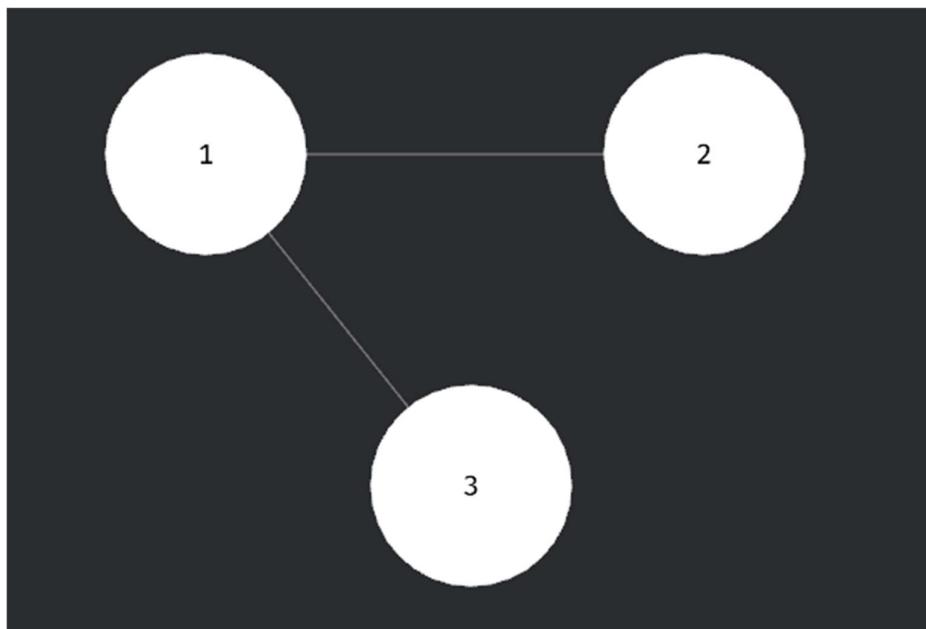


Figure 2 : Représentation d'un graphe dans notre logiciel

Pour ce faire, nous avons commencé par représenter les différents composants d'un graphe par des structures de données : les nœuds représentant les machines, les arcs représentant les moyens de stockage ou de déplacement des entités et les tampons (invisibles à l'écran) qui permettent de gérer l'interaction entre un arc et un nœud (une valve peut être un exemple concret).

L'interface utilisateur a aussi été gérée, il est possible de créer un nœud, un arc, de les supprimer ou encore de les modifier. Toutes les commandes pour réaliser ces actions sont présentes dans le menu d'aide.

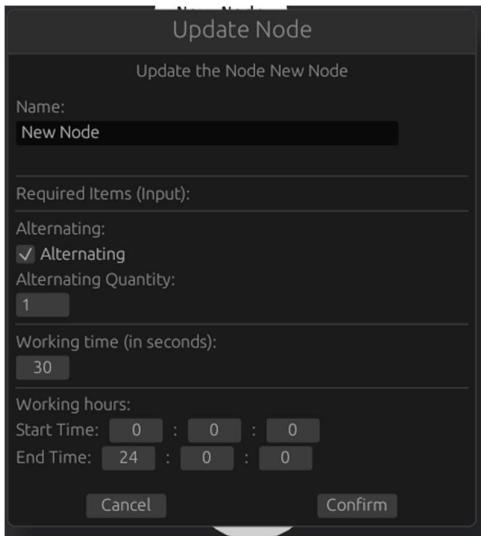


Figure 3 : Création/modification d'un nœud

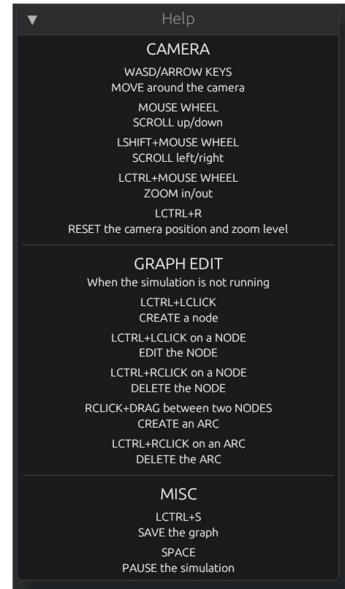


Figure 4 : Fenêtre d'aide

Bevy nous offre la possibilité avec ses caméra de permettre à l'utilisateur de zoomer et se déplacer pour pouvoir observer les différents nœuds si le graphe devient trop conséquent et ne peut pas être afficher entièrement à l'écran.

4) Simulation de production

Une fois la première phase terminée et validée par nos deux superviseurs, il était temps de commencer la phase de simulation de production. Le but est de simulé une phase de production dans le temps pour observer les résultats en fonction des modifications réalisés sur le graphe. Cette phase est toujours en cours de développement et l'ensemble des fonctionnalités ne sont pas encore prêtes.

Les paramètres de la simulation peuvent être modifiés à partir d'une interface utilisateur (ou la barre espace pour la pause/reprise de la simulation). La vitesse et la durée de la simulation peuvent être modifiées ainsi que l'intervalle d'affichage des modification à l'écran (un step correspond à une seconde). La vitesse de simulation représente combien de step de simulation sont simuler par seconde, les « Simulation steps » ou durée de simulation représente la durée totale de la simulation (en secondes) et enfin les « Resting steps » représente le temps entre chaque actualisation à l'écran.

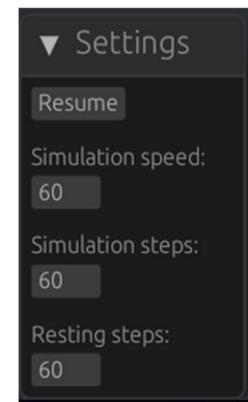


Figure 5 : Paramètres de simulation

Une date représentant la date actuelle dans la simulation est également présente pour pouvoir suivre l'avancé peu importe la vitesse de simulation (la simulation commence toujours le premier janvier 2024). Cela permet également de mieux se rendre compte à partir de quelle date les changement on les résultats souhaités ou non.

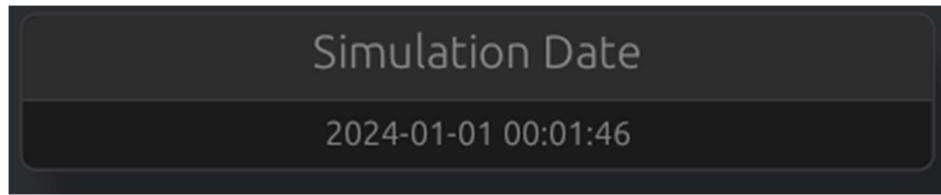


Figure 6 : Date de la simulation

Comme indiqué précédemment, cette phase n'est pas encore finie, les résultats de la simulation ne sont donc pas encore visibles à l'écran. D'autres fonctionnalité comme les changement d'état des machines (préconditions, temps d'action, etc) ont été développée mais n'étant pas très visuelles et assez techniques j'ai décidé de ne pas m'attarder dessus et de seulement présenter les principales fonctionnalités, celles visibles par l'utilisateur.

5) Phase de tests et validation

Une fois la simulation développée entièrement, une phase de tests et de validation est prévue pour vérifier que l'ensemble des besoins a été satisfait et qu'aucun bug n'est présent dans la solution proposée.

Cette phase s'accompagne également d'une phase de documentation, même si nous avons déjà commencer à la réaliser grâce à la documentation intégrée au code grâce à Rust.

A screenshot of the cargo doc generated documentation for the "stage_projet" crate. The left sidebar shows the crate's version as "0.1.0" and lists modules, functions, and crates. The main content area shows the "Modules" section with descriptions for "config", "data", "save", "schedule", "simulation", and "visuals". It also shows the "Functions" section with descriptions for "main" and "setup". The top right of the page has a search bar, a help icon, and a source link.

Figure 7 : Documentation du projet

6) Problèmes rencontrés

Le stage et donc le projet ne sont pas encore finis mais j'ai déjà pu rencontrer quelques problèmes lors de ces derniers mois de travail.

La plus grande difficulté que j'ai rencontré a été la transition POO (Programmation orienté objet) ECS car depuis que j'apprends la programmation j'ai toujours travaillé en POO. Il a donc été assez difficile de passer de l'un à l'autre et je ne pense pas y être encore totalement parvenu. Cela reste cependant une bonne expérience et un nouveau défi à relever.

Une autre difficulté, cette fois plus minime, concerne cette fois Rust et plus particulièrement le temps d'exécution de ce dernier. Je ne sais pas si c'est ma machine qui n'est pas assez puissante ou alors le projet qui est trop volumineux mais c'est la première fois que j'ai des temps d'exécution aussi longs, ce qui est plutôt pénible quand on veut relancer souvent l'application, pour la correction de bug par exemple.

A part les deux points précédents, je ne pense pas avoir rencontré d'autres difficultés lors de ces derniers mois de stage.

IV - Conclusion

Ce stage au sein du Groupe Conseil Filion a représenté une expérience enrichissante, s'inscrivant dans la continuité de mes études en informatique. Le projet principal visait à moderniser une application de simulation logistique en passant de Python à Rust, tout en adoptant une architecture orientée données. Les objectifs fixés ont été partiellement atteints : la nouvelle architecture a été développée, le simulateur fonctionnel a été mis en place et les bases de la simulation de production sont en cours de finalisation.

Le projet a permis de moderniser la solution existante en Rust, offrant ainsi des performances et une sécurité accrues. Les principales fonctionnalités de modélisation de graphes orientés et de simulation logistique ont été implémentées, bien que la phase de test et de validation finale reste à compléter. Les défis techniques, notamment la transition de la POO à l'ECS, ont été partiellement surmontés, apportant une nouvelle dimension à ma compréhension des architectures logicielles. Les résultats obtenus jusqu'à présent montrent un potentiel significatif pour l'entreprise d'améliorer ses outils de simulation, ouvrant la voie à des optimisations futures.

D'un point de vue personnel, ce stage m'a permis d'acquérir de nouvelles compétences en programmation Rust et en architecture ECS. J'ai également renforcé mes connaissances en modélisation de graphes et en développement de systèmes de simulation. Le travail en équipe, l'organisation agile et les interactions régulières avec les experts du Groupe Conseil Filion ont été des aspects particulièrement enrichissants de cette expérience. J'ai apprécié l'opportunité de travailler sur des technologies de pointe et de contribuer à un projet concret avec un impact direct sur les opérations industrielles.

Ce stage et cette année entière à l'étranger mon beaucoup appris que ce soit en informatique ou dans la vie quotidienne. En effet, cette année à l'UQAC m'as permis d'obtenir de nouvelle compétence techniques notamment en apprentissage machine ou encore en traitement de données mais cela a surtout été une expérience enrichissante au niveau personnel. Vivre un an à l'étranger m'a permis de prendre en maturité et d'avoir des expériences que je n'aurais pas eu autrement.

Ces trois mois de stage m'ont conforté dans mes choix d'avenir à court terme. En effet, cela a confirmé que je voulais bien continuer mes études puis travailler dans le domaine de l'informatique qui m'intéresse de plus en plus mais également que je ne suis pas encore totalement prêt pour le monde du travail et que j'ai encore besoin de quelques années d'études.

V - Lexique

Rust : Langage de programmation performant et sécurisé

ECS (Entité Composant Système) : Architecture de programmation orienté objet principalement utilisé pour le développement de jeux vidéo ou autres domaines complexes.

Bevy : Moteur de jeux vidéo intégrer à Rust

Graphe mathématique : Ensemble de nœuds reliés entre eux par des arêtes. Un graphe permet de représenter un ensemble d'objets reliés entre eux par n'importe quel type de lien.

Refactoring : Restructuration du code source d'un programme existant sans en modifier le comportement externe.

Step : Unité de mesure de l'avancement de la simulation, correspond dans notre cas à une seconde.

VI - Webographie

Documentation Rust :

<https://doc.rust-lang.org/stable/book/>

<https://docs.rs/>

Documentation Bevy :

<https://docs.rs/bevy/latest/bevy/>

Vidéos explication Bevy :

https://youtube.com/playlist?list=PL2wAo2qwCxGDp9fzBOTy_kpUTSwM1iWWd

Explications ECS :

<https://www.simplilearn.com/entity-component-system-introductory-guide-article>