

Ecole Publique d'Ingénieurs en 3 ans

Rapport

PROJET SYSTEME D'EXPLOITATION

Ensicaen
Année Universitaire 2024/2025
INFO

ZOUGGARI taha
TOUJANI mohamed
LY johann
BRIDA mohamed taha



www.ensicaen.fr

TABLE DES MATIERES

1. Epidémique simulation	4
1.1. About	4
1.1.1. Citizen manager	4
1.1.2. . City	5
1.1.3. Timer	5
1.1.4. Press agency	5
1.2. Difficulté rencontrer	6
1.3. Conclusion	6

Introduction

Dans le cadre du module de Systèmes d'exploitation, le projet "Épidémie 2024" a pour objectif de mettre en application les concepts étudiés en cours, notamment les mécanismes de communication interprocessus, la gestion des threads, et l'utilisation des sémaphores. Il s'agit de développer une simulation de propagation virale dans une ville, tout en intégrant des citoyens aux rôles variés, tels que médecins, pompiers, et journalistes. Ces acteurs interagissent avec leur environnement et participent activement à la gestion de la contamination.

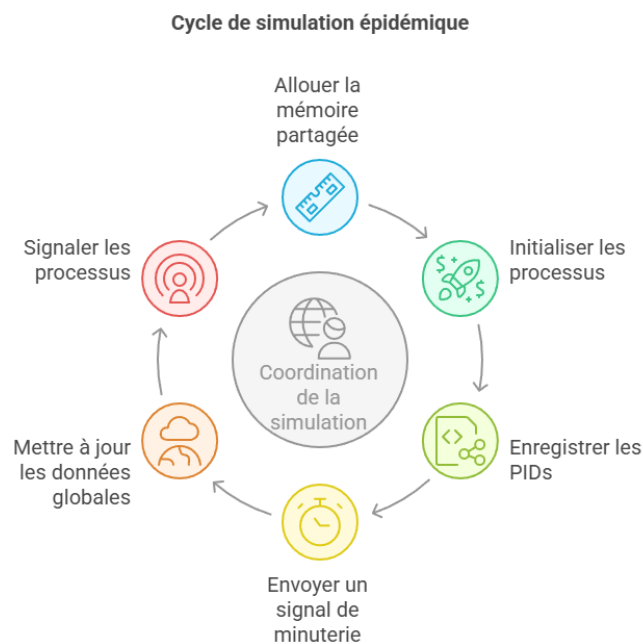
Ce projet constitue une opportunité de combiner des compétences techniques avec une réflexion algorithmique avancée, afin de modéliser un scénario réaliste et complexe. Il permet également de travailler en équipe et de développer des solutions robustes, tout en suivant une méthodologie structurée et collaborative.

Dans ce rapport, nous présenterons tout d'abord les choix de conception effectués, suivis d'une description détaillée de l'implémentation technique et des défis rencontrés. Enfin, nous proposerons une analyse des résultats obtenus et des pistes d'amélioration pour enrichir la simulation

1. Epidémique simulation

1.1. About

Le programme **epidemic_sim** débute par l'allocation d'une mémoire partagée, qui contient les données essentielles comme les informations sur la carte, les emplacements stratégiques, l'état des citoyens et l'heure actuelle. Une fois les données configurées, le programme se duplique à l'aide de **fork()** et utilise **execl()** pour lancer chaque processus complémentaire requis, tels que **citizen_manager**, **press_agency** et **timer**. Les **PIDs** des processus sont enregistrés dans la mémoire partagée afin de faciliter les échanges de signaux entre eux. À chaque cycle de simulation, un signal en provenance de **timer** déclenche la fonction, qui met à jour les données globales en fonction des événements et des évolutions, avant d'envoyer un signal aux autres programmes pour qu'ils exécutent leurs tâches respectives.



1.1.1. Citizen manager

citizen_manager gère la vie quotidienne des citoyens dans la simulation. Chaque citoyen est représenté par une structure dans la mémoire partagée et est associé à un thread distinct. Au début de la simulation, les citoyens reçoivent des attributs aléatoires, tels qu'un lieu de

résidence et un lieu de travail, pour simuler leurs déplacements quotidiens. Lors de chaque tour, déclenché par un signal, les threads des citoyens sont mis à jour pour refléter leurs actions selon l'heure (travail, déplacement, ou repos). Cette gestion centralisée permet de simuler des interactions réalistes et dynamiques entre les citoyens et leur environnement.

Chaque individu a un état de santé :

- Sain (susceptible à l'infection).
- Infecté (capable de transmettre la maladie).
- Guéri (immunisé temporairement ou définitivement).
- Décédé (fin de participation à la simulation).

1.1.2.. City

La structure de la ville est chargée dans le programme à partir de la mémoire partagée. Les threads ne modifient pas les informations de cette structure, ils se contentent de les lire.

Étant donné que le déplacement des citoyens implique une modification de la capacité actuelle des différentes cases de la ville, le mouvement d'un citoyen peut entraver celui d'un autre. Ainsi, pour suivre les capacités actuelles des cases sans modifier la structure originale, nous avons créé une copie des capacités actuelles des cases dans le programme citizen manager. Cette copie est mise à jour au début de chaque tour (pour inclure les futurs décès ou incinérations de corps). L'accès à cette copie par les threads est contrôlé à l'aide de sémaphores.

1.1.3. Timer

Le timer est un programme qui signale périodiquement la fin d'un tour à epidemic_sim après une durée spécifique.

La durée d'un tour, comprise entre 1 et 5 secondes, est transmise au programme lors de son appel. Le timer prend deux arguments : le premier est le PID du programme epidemic_sim, et le second est la durée du tour. À chaque expiration de cette durée, le timer envoie un signal à epidemic_sim à l'aide de la primitive alarm, pour l'informer de la fin du tour en cours. Ce mécanisme garantit la synchronisation régulière des étapes de la simulation.

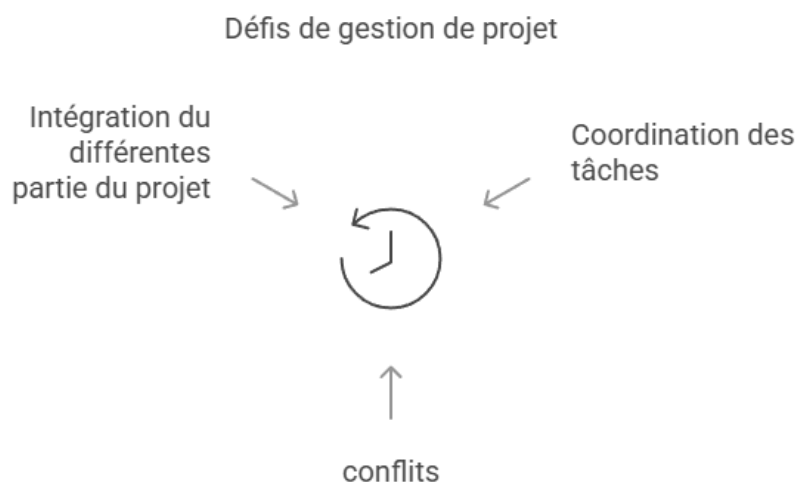
1.1.4. Press agency

Le programme press_agency permet la communication entre les journalistes et le transfert d'informations en utilisant une file de messages.

Tout d'abord, il crée une file de messages et attend que le programme Citizen Manager lui envoie des informations (le type d'information, les données et leur priorité). Ensuite, il se connecte au serveur de l'interface graphique via des sockets et transmet les informations en fonction de leur priorité, après y avoir apporté quelques modifications.

1.2. Difficulté rencontrer

Nous avons rencontré plusieurs problèmes lors de la réalisation du projet. Tout d'abord, la coordination et la répartition des tâches ont parfois été compliquées, ce qui a entraîné des chevauchements et des retards dans certaines parties du développement. Enfin, l'intégration des différentes parties du projet a révélé des incompatibilités imprévues, demandant des ajustements de dernière minute et un effort supplémentaire pour respecter les délais impartis



1.3. Conclusion

Ce projet nous a offert une expérience précieuse pour approfondir nos connaissances techniques sur la communication interprocessus. L'utilisation des files de messages, des sémaphores et des zones de mémoire partagée nous a permis de comprendre leur rôle essentiel dans la coordination et les échanges efficaces entre processus. Cette simulation nous a également confrontés à des défis complexes, tels que la synchronisation, la gestion des ressources partagées et le contrôle rigoureux de la mémoire, renforçant ainsi notre capacité à concevoir des systèmes robustes et performants



Ecole Publique d'Ingénieurs en 3 ans

6 boulevard Maréchal Juin, CS 45053
14050 CAEN cedex 04

