

Architecture logicielle

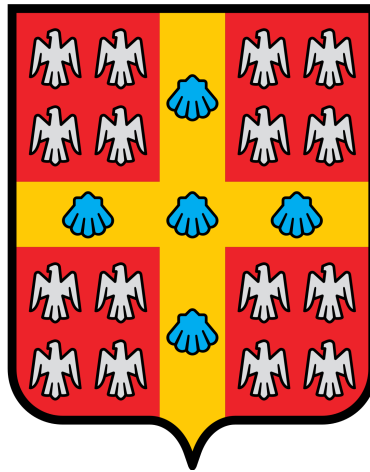
A22 - GLO-4003

Projet SULVLO

Document de remise - itération #4

Université Laval

Faculté des sciences et de génie



Équipe 5

BUQUET Antoine

CARRIER Marc-Antoine

GAUVREAU Charles

GIROUX Charles-Antoine

MARCOUX Étienne

MIVILLE DESCHÊNES Maxime

Stories

Récit #6: En tant qu'Alexandra, je peux retourner le vélo pour un voyage supérieur à 10 minutes dans un emplacement libre d'une des stations du campus afin de terminer mon voyage et d'être chargée pour l'excédent

Critères d'acceptation :

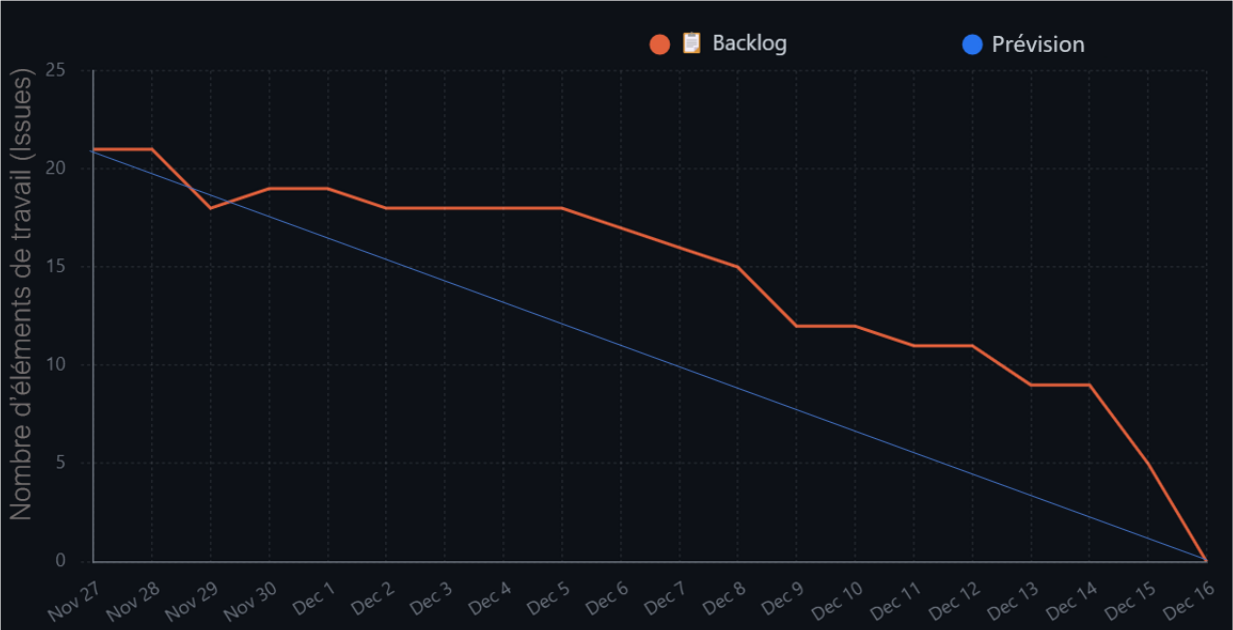
- Le vélo est maintenant stationné dans un emplacement d'une station sur le campus
- Son voyage a occasionné des frais supplémentaires
 - La formule des frais supplémentaires est la suivante :
 - Premières 10 minutes gratuites
 - 0,05 \$ par minutes supplémentaires
 - Arrondi vers le haut dès la 15e seconde
 - Exemple :
 - Un voyage de 12 minutes 14 secondes sera facturé pour 2 minutes supplémentaires
 - Un voyage de 23 minutes 15 secondes sera facturé pour 14 minutes supplémentaires
 - Un voyage de 17 minutes 39 secondes sera facturé pour 8 minutes supplémentaires
 - Les frais supplémentaires ne sont pas déduits directement sur la carte après le voyage, ils sont gardés sur un solde mensuel

Récit #7: En tant qu'Alexandra, je peux consulter mon historique de voyage sur le campus afin de constater mon utilisation du service

Critères d'acceptation

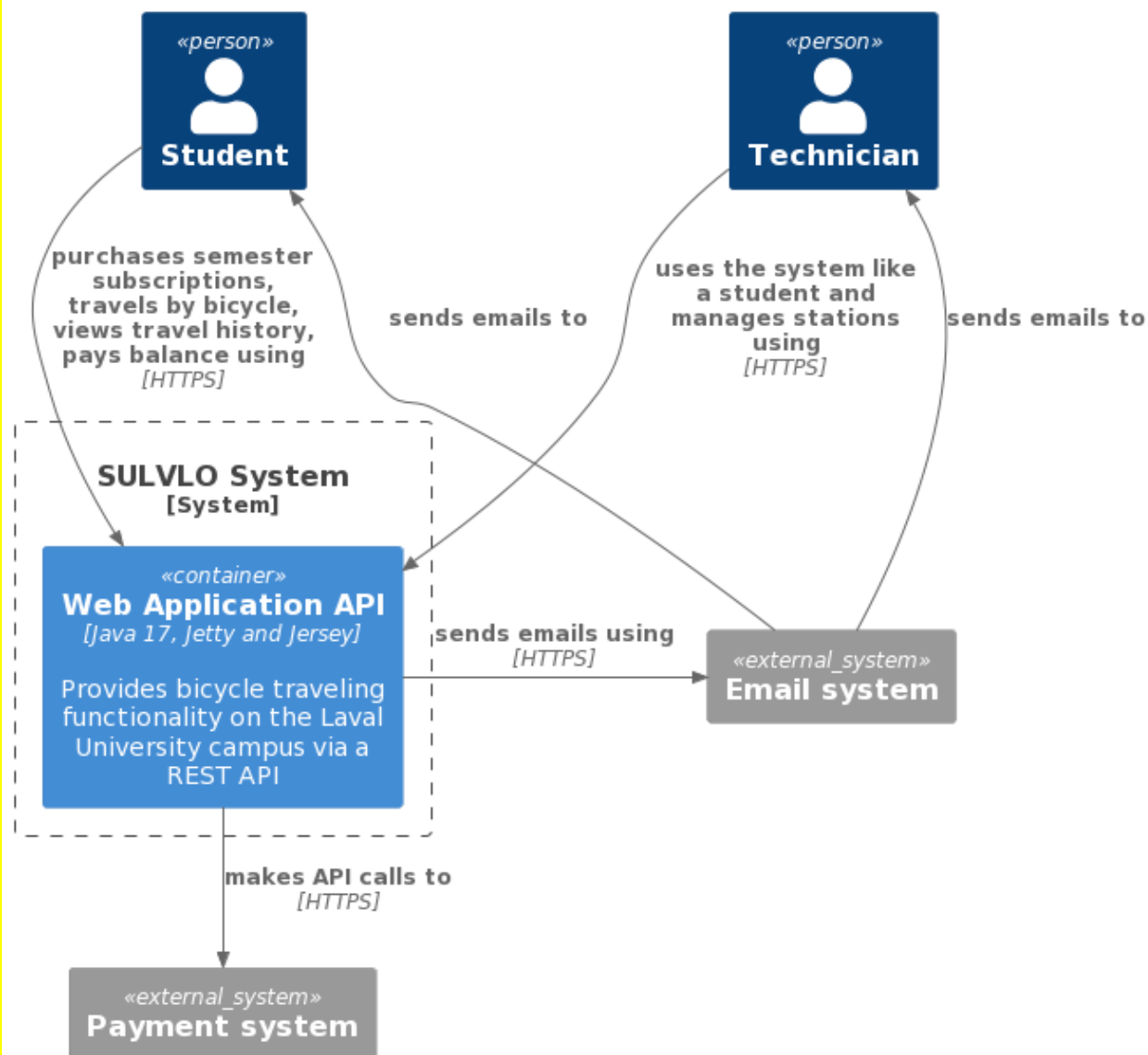
- Elle peut voir tous les voyages effectués dans le dernier mois par défaut (1er au 31e jours)
- Elle peut voir toute l'information d'un voyage dans son historique
- Elle veut avoir un sommaire d'utilisation dans son historique
 - Temps de déplacement total
 - Temps moyen d'un déplacement
 - Nombre de déplacements
 - Station favorite (le plus d'emprunts et de retours)

Burndown chart

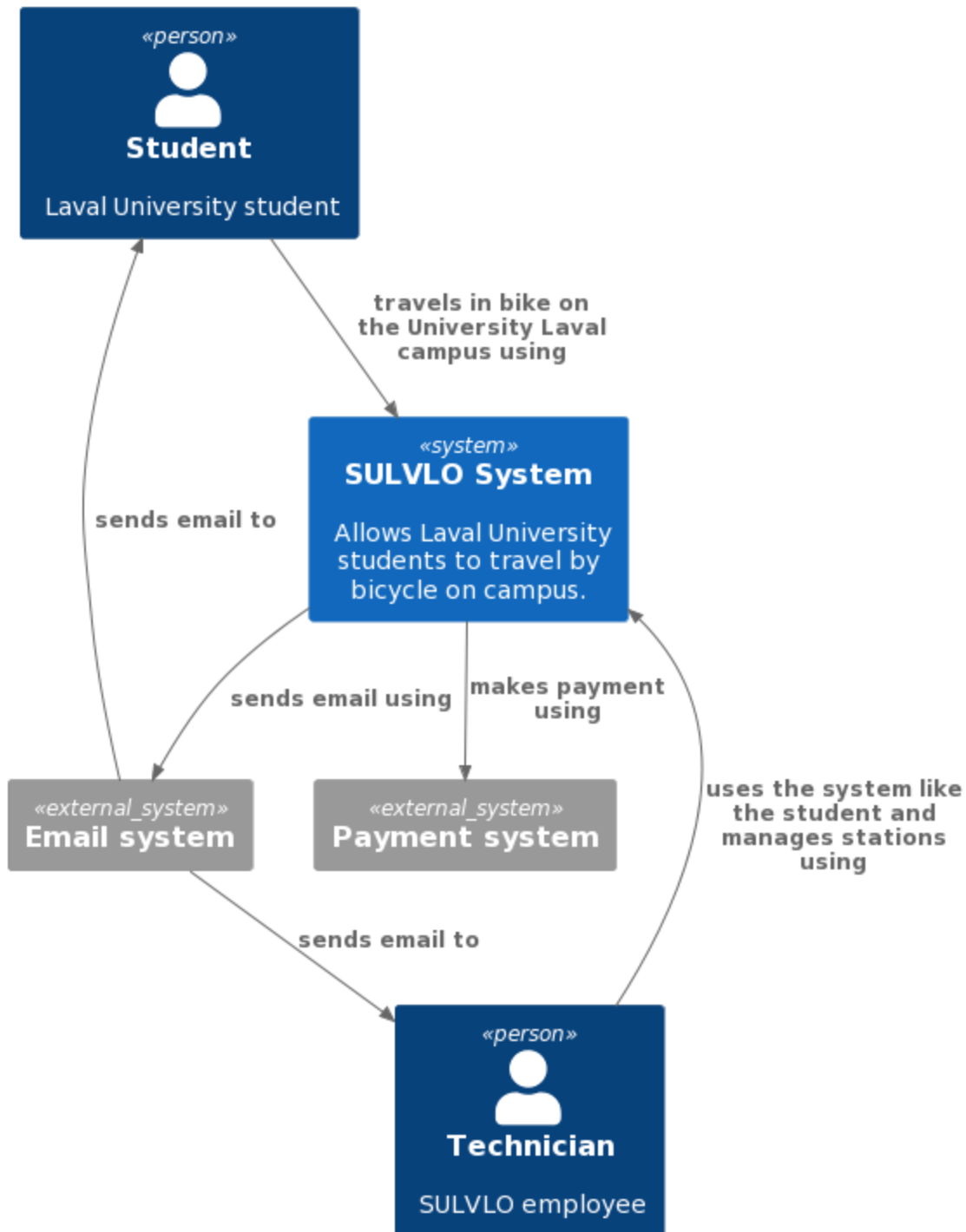


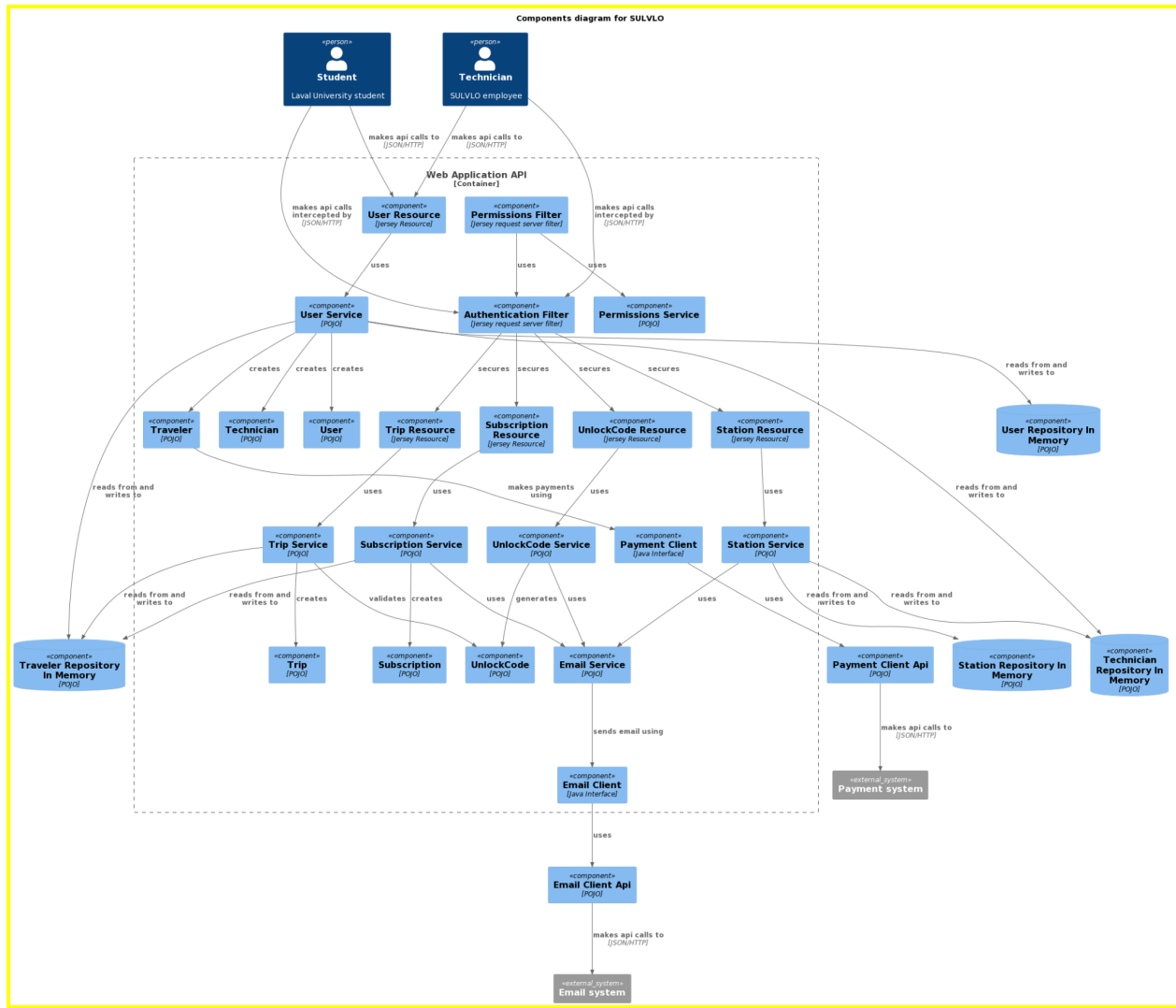
Architecture logicielle

Containers diagram for SULVLO

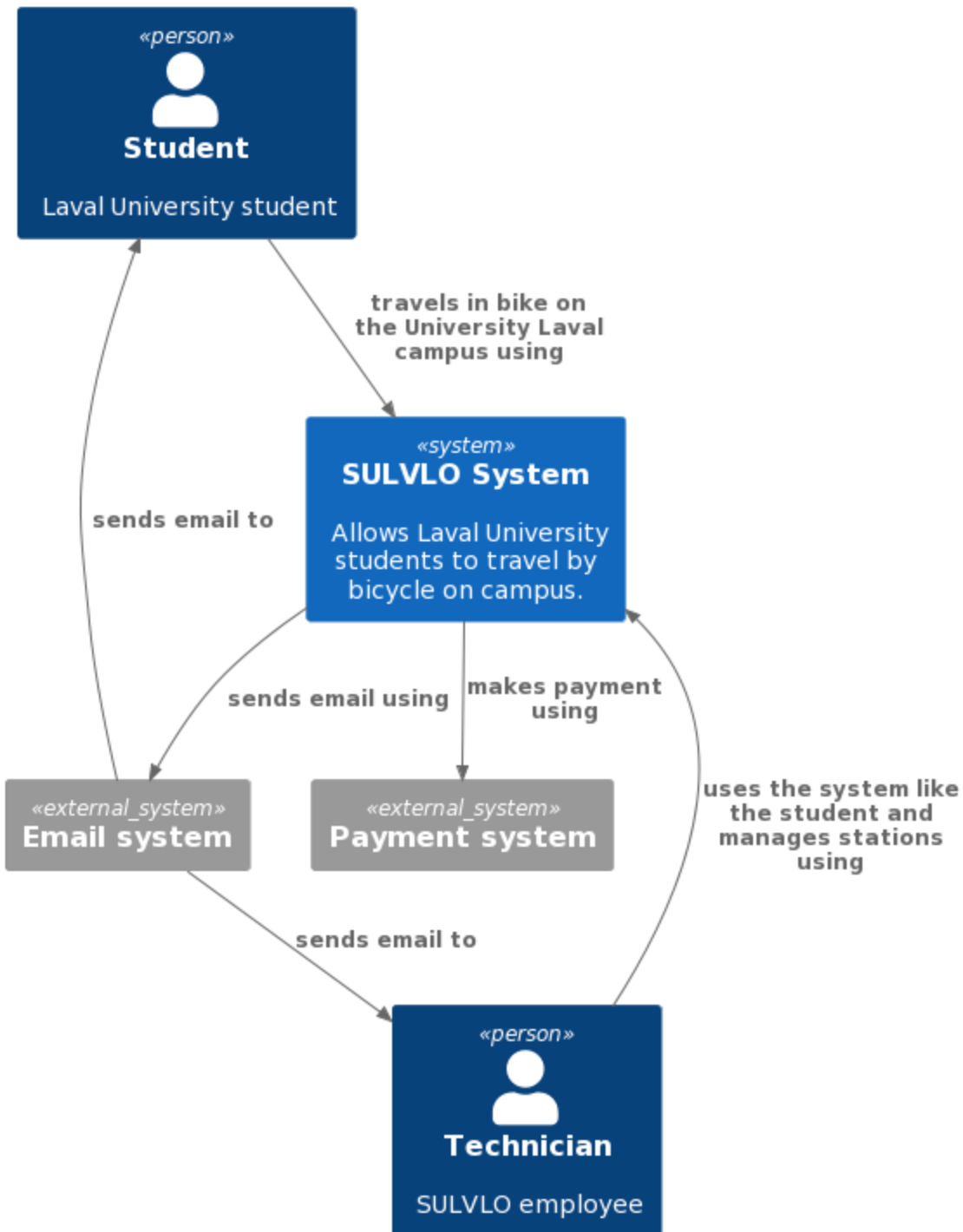


System Context diagram for SULVLO





System Context diagram for SULVLO



Patrons

- Factory:
 - Permet de respecter le SRP
 - Localiser la création des différents types d'un objet à un seul endroit. (if/switch case)
- Repository:
 - Respecter le DIP
 - Pouvoir changer de base de données
 - Permettre à la couche applicative d'avoir un accès rapproché des concepts du domaine pour la persistance
- State:
 - Changer le comportement du Bicycle selon son état (sa charge, etc.)
 - Réduire la duplication de code (DRY)
 - Respecter le SRP
 - Respecter le OCP (ajouter de nouveaux State facilement)
- Gateway:
 - Respecter le DIP
 - Parler à un système externe (envoi d'Email et système de paiement)
- DTO:
 - Communication entre la couche API et la couche applicative
 - Éviter que les couches connaissent les objets du domaine
- Builder:
 - Création d'objets valides pour les tests
 - Éviter de la duplication de code
- Décorateur (Wrapper)
 - Nous l'avons utilisé pour la classe *PermissionsFilter* qui implémente la classe *ContainerRequestFilter*
 - Permet d'envelopper la classe existante et ajoute des fonctionnalités manquantes pour notre *use case*