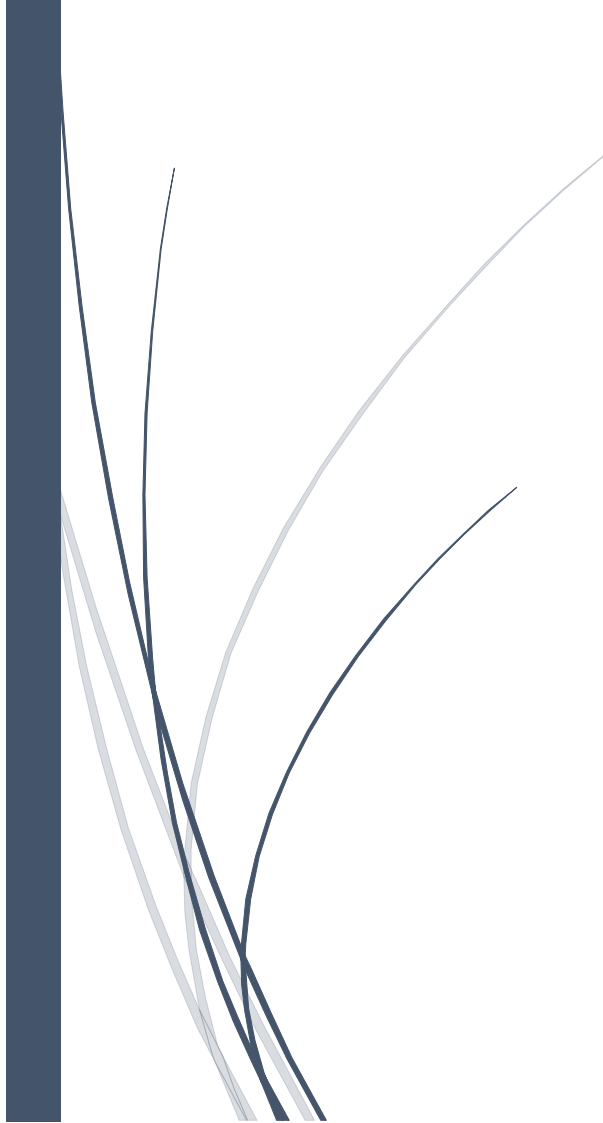




09/10/2021

Software Architecture for Cloud Computing

Rapport Intermédiaire



Anas FRANCIS
Jonas ANIGLO
Soulaïman ZABOURDINE
Patrick ANAGONOU

POLYTECH'NICE SOPHIA

Table des matières

I.	Introduction	2
II.	Personas et scénarios.....	2
A.	Personas.....	2
B.	Scénarios	3
C.	Hypothèses sur l'application	5
III.	Architecture du système	6
A.	Diagramme de classes (UML).....	6
B.	Diagramme de composant.....	7
C.	Justification du choix de l'architecture	8
IV.	Scénario de démonstration MVP	8
V.	Conclusion.....	8

I. Introduction

Dans le cadre du cours Cloud Computing, nous avons eu à faire une analyse de besoin et mis en place une architecture pour l'implémentation d'une application appelée PolyPet. PolyPet répondra aux besoins de la société PolyPet, il s'agit d'une société qui développe une large variété de produits orientés NAC (Nouveaux Animaux de Compagnie).

PolyPet aura pour but de permettre à ses clients de découvrir les différents produits présents sur le catalogue et d'en passer une commande. Il permettra aussi la mise à jour de ce catalogue par les gestionnaires du site.

II. Personas et scénarios

A. Personas

Gilbert : visiteur du site PolyPet.

Besoin 1 : **En tant que** client de PolyPet, **je veux** voir les produits triés dans des catalogues, **afin de** trouver le produit qui m'intéresse.

Besoin 2 : **En tant que** client de PolyPet ayant passé une commande sur le site internet, **je veux** que les produits de ma commande soient livrés par drone, **afin de** clôturer mon achat.

Alexandre : manager d'une industrie pharmaceutique

Besoin : **En tant que** manager d'une industrie pharmaceutique, **je voudrais** ajouter mes produits sur le site de PolyPet **afin que** les clients du site puissent voir mon produit.

Marion : cliente qui s'est connecté sur le site de PolyPet et qui veut ajouter des éléments au panier

Besoin : **En tant que** cliente qui a ajouté des produits au panier **je veux** procéder à l'achat en utilisant ma carte bancaire, **afin de** finaliser l'achat.

Frédéric : Manager du site PolyPet

Besoin 1 : **En tant que** manager **je souhaite** gérer les produits sur les sites, **afin de** mettre à jour le site avec les nouveaux produits.

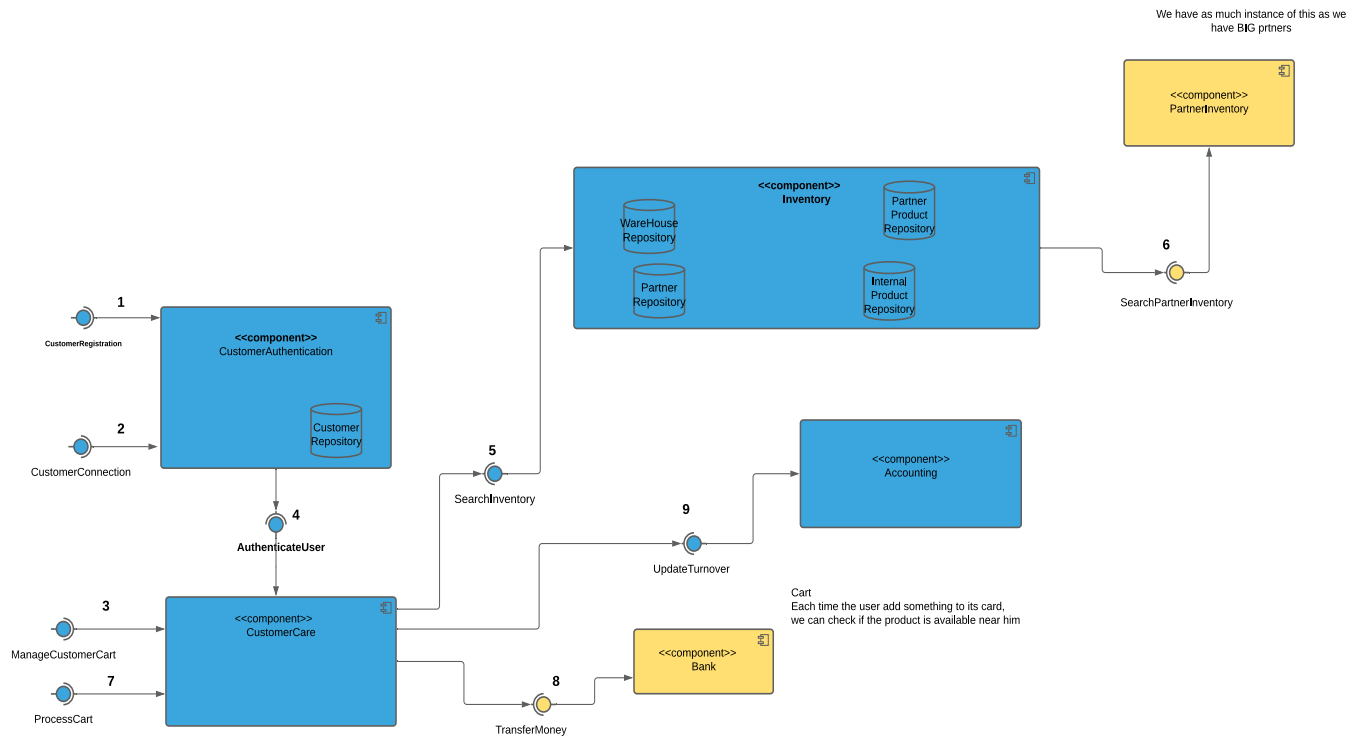
Besoin 2 : **En tant que** manager **je souhaite** mettre à jour les stocks **afin de** m'assurer qu'il n'y a pas de pénurie dans un/une entrepôt/région pendant trop longtemps.

Valérie : Membre de l'équipe Marketing

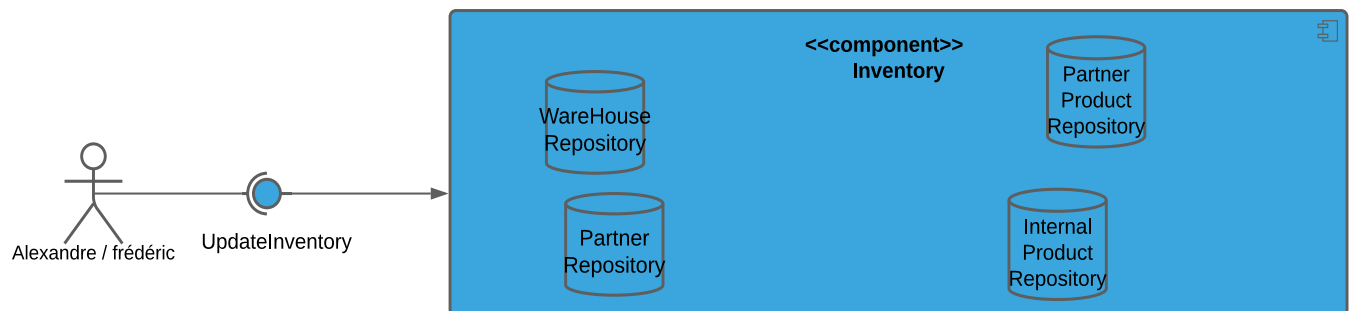
Besoin : **En tant que** membre de l'équipe Marketing de PolyPet **je souhaite** voir les statistiques des visites et les achats sur les différentes pages des produits **afin de** mettre en place une nouvelle stratégie de vente.

B. Scénarios

- Achat de produit



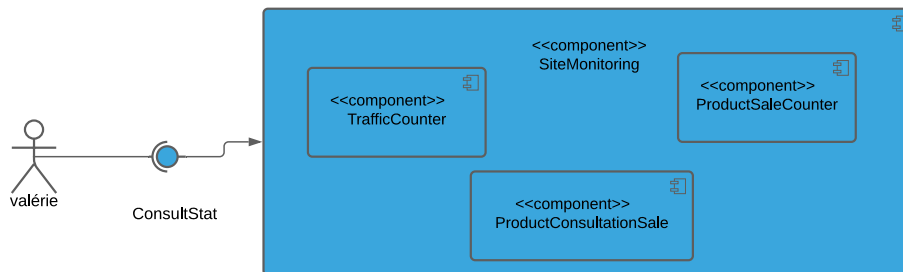
- Mettre à jour le dépôt du site web (stockage)



When we process a customer cart we update the inventory.

- Consulter les statistiques afin d'adapter une stratégie

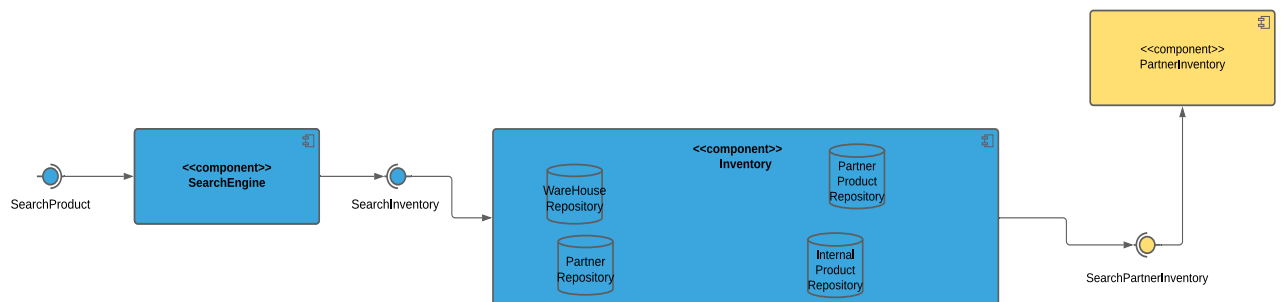
To develop as aspects (AOP) and or messaging within components



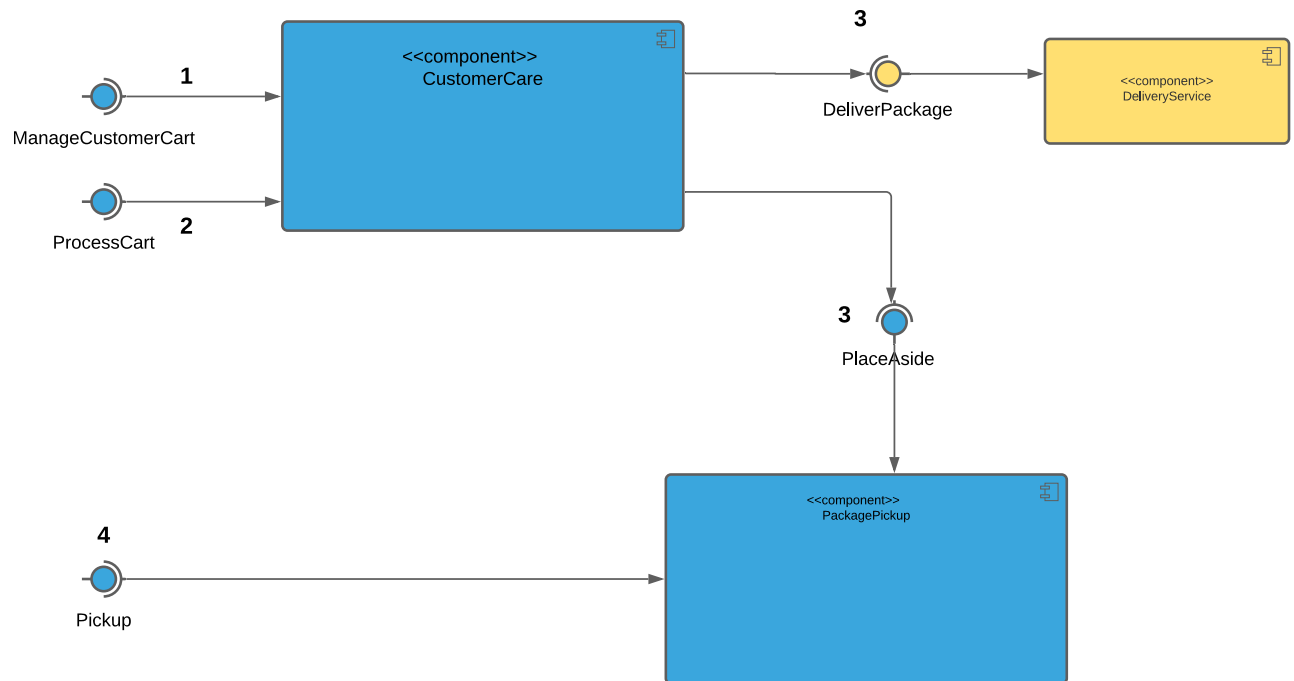
There will be an interceptor on each interface so when there is a new reaserch or a new purchase this component will update it's counter and compute the percent of reaserch or purchasing of different products

- Recherche d'un produit

We have as much instance of this as we have BIG prtners



- Demander de se faire livrer / ou récupérer au magasin



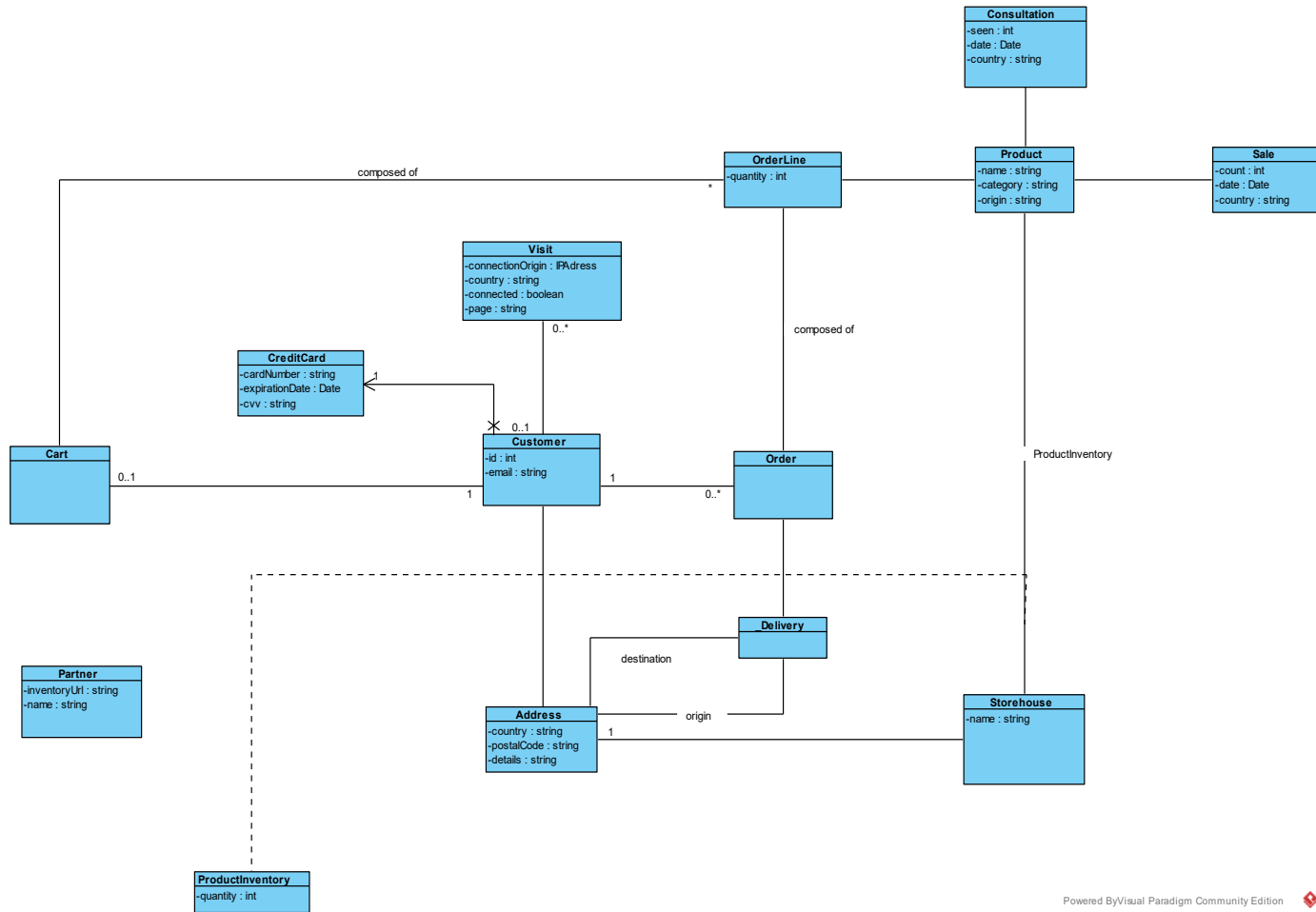
C. Hypothèses sur l'application

Afin d'implémenter notre produit minimum viable (MVP) nous supposons que :

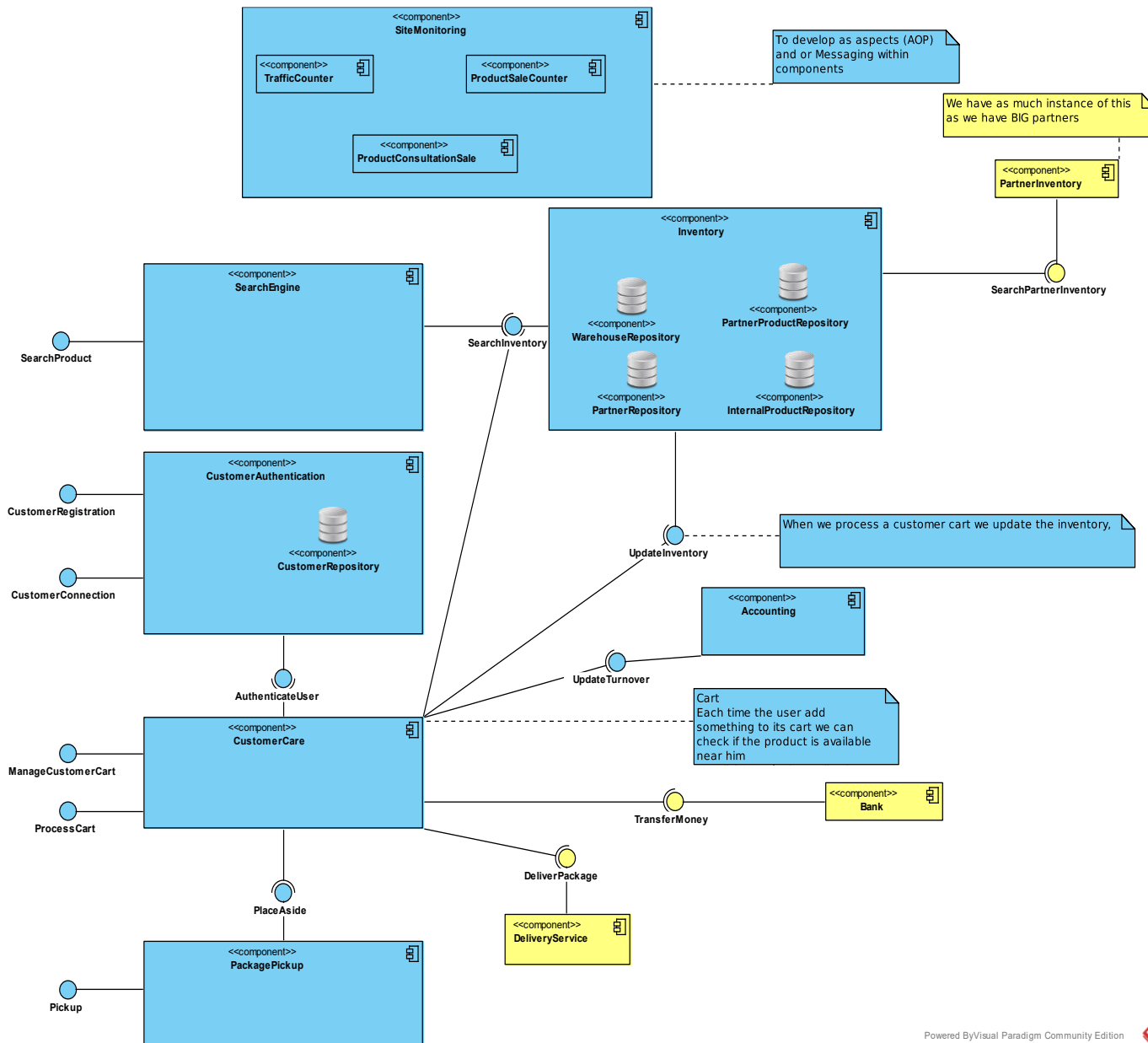
- Nous accepterons toujours de livrer au client même si le produit demandé est à une grande distance par rapport au client (il est responsable de payer la livraison).
- Joly Drone, l'entreprise de livraison par drone, sera toujours disponibles pour effectuer nos livraisons.

III. Architecture du système

A. Diagramme de classes (UML)



B. Diagramme de composant



C. Justification du choix de l'architecture

Nous comptons réaliser notre implémentation de PolyPet en Spring Boot Java. Notre découpage s'inspire des différentes étapes du métier. Nous avons notamment prévu de développer un ensemble de composants responsables du monitoring du trafic et des statistiques sous la forme d'intercepteurs (Aspect Oriented Programming). Cela va nous permettre de déplacer certaines préoccupations du métier qui interviennent à plusieurs niveaux dans des méthodes dédiées et d'éviter la duplication de code.

IV. Scénario de démonstration MVP

Gilbert souhaite acheter une laisse pour son chien. Il choisit de faire cet achat chez 'PolyPet' via leur application qui donne accès à un catalogue regroupant plusieurs produits de plusieurs vendeurs. Pendant sa recherche d'article, il décide de profiter de cette occasion pour acheter des produits supplémentaires : un sac de croquettes pour chien 'Medium Maxi - Adult' vendu par 'Ultima' (un gros partenaire de PolyPet) ainsi qu'une balle en caoutchouc vendu par 'ZooPlus' un commerçant (un petit partenaire).

Une fois le panier de Gilbert rempli, il souhaite passer à la validation de son achat en visualisant une dernière fois le contenu de son panier. L'application PolyPet fera alors une vérification de stock afin de s'assurer que tous les produits du panier sont disponibles chez les différents partenaires. Heureusement, tous les produits du panier sont disponibles en stock et Gilbert peut à présent passer au paiement de son panier.

Gilbert entre alors les informations de sa carte bancaire dans PolyPet, et l'application se charge de relayer ces informations au système de banque. Le système bancaire retourne un signal 'OK' à PolyPet et le paiement est validé. PolyPet génère alors un numéro de commande et la communique à Gilbert.

Finalement une demande est envoyée à l'entreprise Joly Drone, le partenaire de PolyPet chargé d'effectuer les livraisons à nos clients par drone.

V. Conclusion

Pour cette livraison nous avons conçu une architecture permettant de répondre aux besoins de la société PolyPet de livrer un produit minimum viable. Le besoin principal démontré est la recherche et l'achat de produit dans un catalogue alimenté par les partenaires de PolyPet.