

Architecture Logicielle

Variante 9 : Mobile App for Transporters

Groupe I

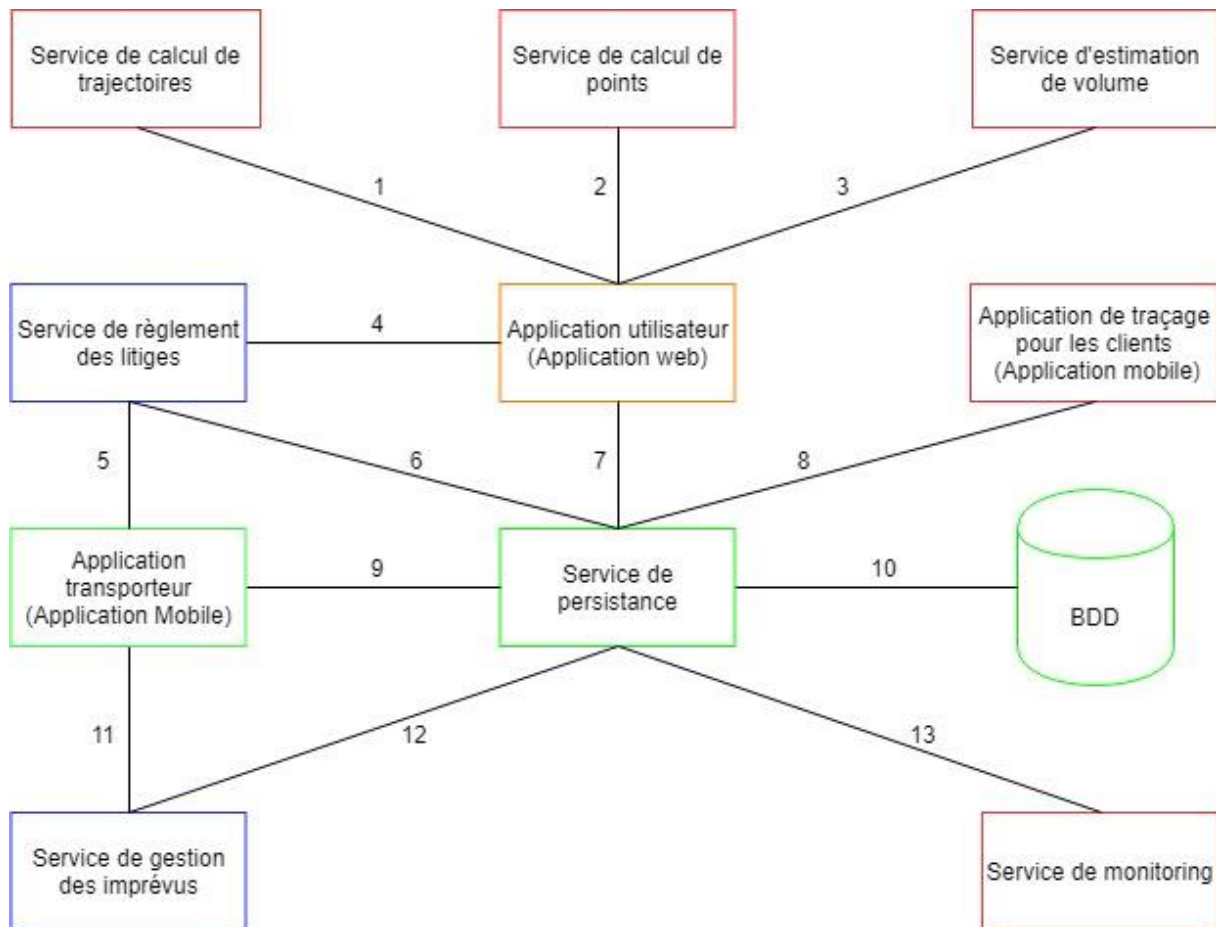
User Story	2
Diagramme de composants global	3
Choix des technologies	4
Roadmap	4
IHM	5
Diagramme de classes	5

User Story

Alice (Client)	Bob (Transporteur)
<p style="text-align: center;"><u>Phase 1a</u></p> <ul style="list-style-type: none"> -Alice s'inscrit / se connecte sur le site internet -Alice peut consulter son solde de points -Alice créer son annonce en indiquant le point de départ, le point d'arrivé, les objets à transporter et une fourchette pour la date. Le système calcule le coût en points de l'intervention 	<p style="text-align: center;"><u>Phase 1b</u></p> <ul style="list-style-type: none"> -Bob télécharge l'application Android -Bob s'inscrit / se connecte sur l'app -Bob peut consulter son solde de points -Bob lance une recherche d'annonces en indiquant, sa ville de départ, sa ville de destination et la taille maximal du bagage à transporter -Bob ajoute une annonce à sa liste de transport -L'application indique à Bob que son coffre n'est pas rempli en lui indiquant l'espace restant et qu'il peut encore ajouter des annonces à sa liste -Bob peut supprimer une ou plusieurs annonces de sa liste de transport -Bob valide son panier en indiquant ses disponibilités pour chaque annonce
<p style="text-align: center;"><u>Phase 2</u></p> <ul style="list-style-type: none"> -Alice reçoit des offres à son annonce, proposés par plusieurs transporteurs -Alice choisit l'offre de Bob -Alice reçoit deux codes par mail, un pour la preuve de dépôt et un pour la preuve de réception 	
<p style="text-align: center;"><u>Phase 3a</u></p> <ul style="list-style-type: none"> -Alice vient au RDV avec ses objets et rencontre Bob -Alice donne son code de dépôt à Bob 	<p style="text-align: center;"><u>Phase 3b</u></p> <ul style="list-style-type: none"> -Bob vient au RDV avec son véhicule et rencontre Alice -Bob entre le code de dépôt que Alice lui a donné
	<p style="text-align: center;"><u>Phase 4</u></p> <ul style="list-style-type: none"> -Bob livre, à l'adresse indiquée, les objets de Alice (cela peut-être à Alice, une personne désignée par Alice ou bien un autre livreur dans le cas d'un relais) -Bob entre le code réception que le destinataire lui a donné
<p style="text-align: center;"><u>Phase 5a</u></p> <ul style="list-style-type: none"> -Alice peut voir la preuve de reçu sur le site internet -Le compte de Alice se fait débiter du nombre de points associé à sa commande -Alice peut consulter son historique de commande 	<p style="text-align: center;"><u>Phase 5b</u></p> <ul style="list-style-type: none"> -Bob reçoit ses points sur son compte -Bob peut consulter son historique de contrats

Remarque : La user story de Alice sera mocké car elle n'est pas nécessaire à l'application en elle-même.

Diagramme de composants global



Légende :

Les différentes liaisons représentent une connexion ou des échanges de données entre les deux nœuds correspondants. Les éléments de composants encadrés en vert sont ceux à implémenter, les oranges sont ceux à mocker pour le POC du 9/11/18, les bleus sont ceux à mocker pour le POC du 15/02/18 et les rouges sont ceux qui n'entre pas dans notre scope.

- 1) Utilisé pour générer une suite de mini-trajets couvrant un départ et une arrivée.
- 2) Utilisé pour calculer le nombre de points correspondant à une annonce.
- 3) Utilisé pour l'estimation de volume lorsqu'un utilisateur prend une photo de ses affaires.
- 4) Utilisé dans le cas où il y a un problème avec le transporteur.
- 5) Utilisé dans le cas où il y a un problème avec le client.
- 6) Simple liaison pour la persistance du modèle.
- 7) Simple liaison pour la persistance du modèle.
- 8) Simple liaison pour la persistance du modèle.
- 9) Simple liaison pour la persistance du modèle.
- 10) Le service de persistance est lui-même connecté à une base de données.
- 11) Utilisé dans le cas où le transporteur à un problème avec son véhicule.
- 12) Simple liaison pour la persistance du modèle.
- 13) Liaison Service de monitoring / Service de persistance.

Choix des technologies

Nous avons choisi la technologie Android pour la partie application mobile car celle-ci représente la grande majorité des terminaux mobile à travers le monde, de plus, aucun des membres de l'équipe n'est à l'aise avec iOS qui est le second choix possible. De plus, il est beaucoup plus simple, à terme de déployer une application Android au grand public qu'une application iOS sur les stores respectifs.

Pour la partie du service de persistance, nous avons choisi de l'implémenter à l'aide de Symfony 4, un framework PHP. En effet, ce framework offre beaucoup de possibilités tant pour la création d'applications web (qui ne nous concerne pas) que la réalisation de end-point à l'aide d'un système de routes intuitif, d'un système de vérifications de droits ou encore d'un ORM intégré (Doctrine 2). Un des membres de l'équipe est familier avec cette technologie, ce qui nous fera gagner du temps.

Pour la Base de données, nous nous sommes orientés vers une base MySQL. Simple à mettre en place et à déployer, elle reste le choix idéal pour lancer son produit tant que l'on ne se heurte pas à des contraintes de persistance exotiques. De nombreux géants du numérique ont commencé avec ce SGBD (comme Facebook).

La communication entre l'application mobile et la couche de persistance s'effectuera à l'aide de routes et de verbes HTTP.

Roadmap

Semaine 42	Semaine 43	Semaine 44
Création des différentes classes à utilisées		
Création d'une IHM simple pour l'application		
Mise en place de la base de données		
Création de l'application Android sans persistance		
	Mise en place de la couche de persistance	
	Ajout des interactions entre l'application Android et la couche de persistance	

IHM

Une IHM simple de l'application a été réalisée afin de mieux concevoir les différentes étapes à implémenter lors de la création de notre application. Celle-ci a été conçue sous Marvel App et déroule entièrement la user story du transporteur.

Le lien de l'IHM : <https://marvelapp.com/f2cc2h4/screen/49045251>

Diagramme de classes

