



Projet Technique



BlablaCool

Nicolas Ducom, Hugo Drécourt,
El Mamoune El Bekri El Alaoui et Valentin Polo.
Faculté de Gestion, Economie et des Sciences.
Université Catholique de Lille.



Sommaire

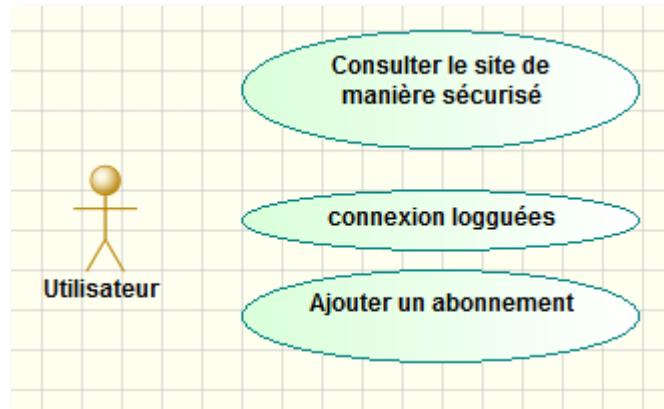
I.	Analyse des besoins	3
A.	Diagramme des cas d'utilisation et description.	3
B.	Diagramme des séquences	8
C.	Maquette et description	17
D.	Base de données.	20
E.	Périmètre	21
II.	Organisation des tâches	22
A.	L'équipe de travail	22
B.	Choix de la méthode de Gestion de Projet	22
C.	Décomposition des tâches	23
D.	Description des tâches	24
III.	Stack technique	25
IV.	Management et communication	26
A.	Communication	26
B.	Gestion des conflits	27
C.	Contrôle de l'avancement du projet	27
V.	Qualité	29
VI.	Ressources utiles	30

Rédaction	El Mamoune El Bekri El Alaoui
Correction	Nicolas Ducom
Validation	Valentin Polo
Version	1.2

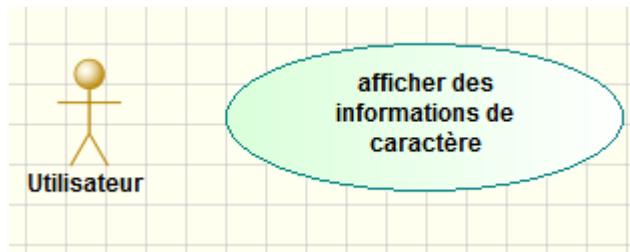
I. Analyse des besoins

A. Diagramme des cas d'utilisation et description.

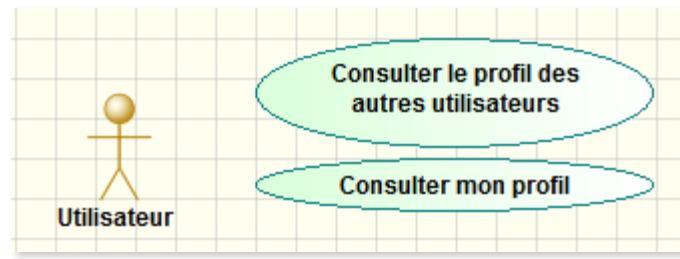
Les utilisateurs de BlablaCool peuvent être à la fois conducteur et passager. Certains peuvent avoir des droits administrateurs sans pour autant avoir un profil Conducteur et/ou passager.



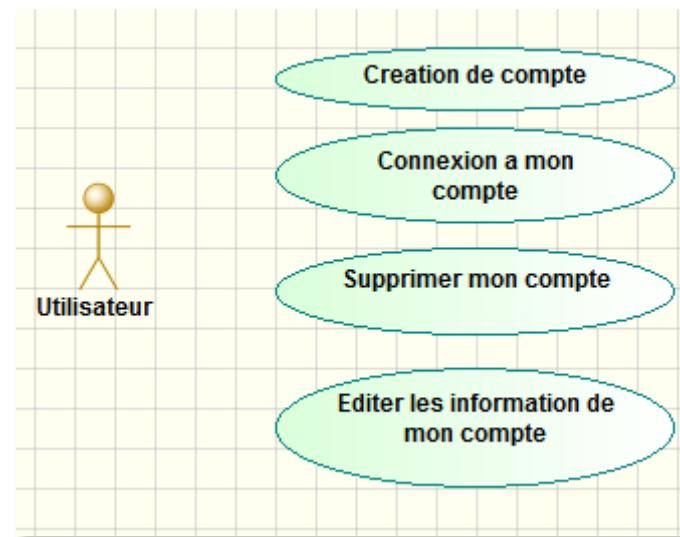
UN UTILISATEUR PEUT CONSULTER LE SITE DE MANIÈRE SÉCURISÉE.
TOUTES LES CONNEXIONS EFFECTUÉES SUR SON COMPTE SONT ÉCRITES DANS UN
FICHIER DE LOG. IL Y A ÉGALEMENT POSSIBILITÉ D'AJOUTER UN ABONNEMENT.



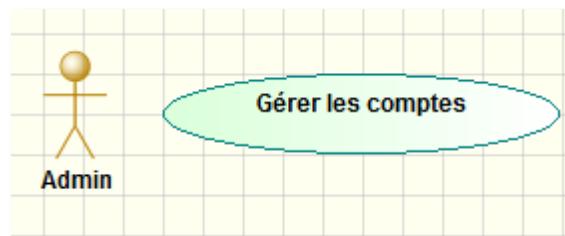
L'UTILISATEUR PEUT AFFICHER DES INFORMATIONS DE CARACTÈRE COMME LE STYLE DE CONDUITE DU CHAUFFEUR, LA POSSIBILITÉ D'AVOIR DES ANIMAUX DE COMPAGNIE, SON PENCHANT POUR LA CIGARETTE OU ENCORE SES GOÛTS MUSICAUX.



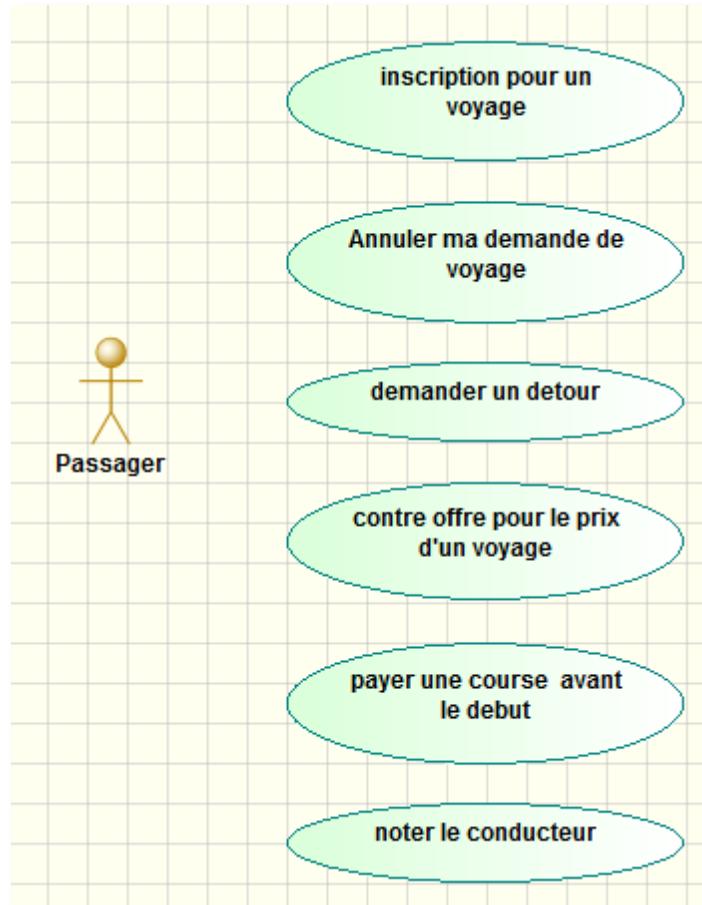
L'UTILISATEUR PEUT AUSSI CONSULTER SON PROFIL OU CELUI DES AUTRES UTILISATEURS.



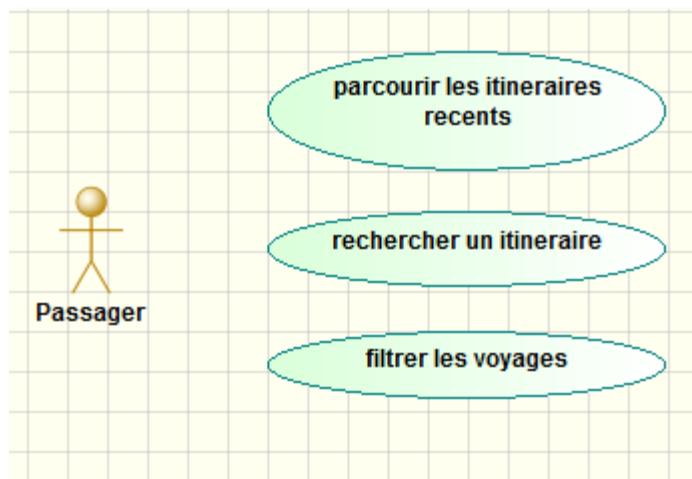
UN UTILISATEUR PEUT SE CRÉER UN COMPTE, S'Y CONNECTER, MODIFIER LES INFORMATIONS QUI Y SONT RATTACHÉES OU ENCORE LE SUPPRIMER.



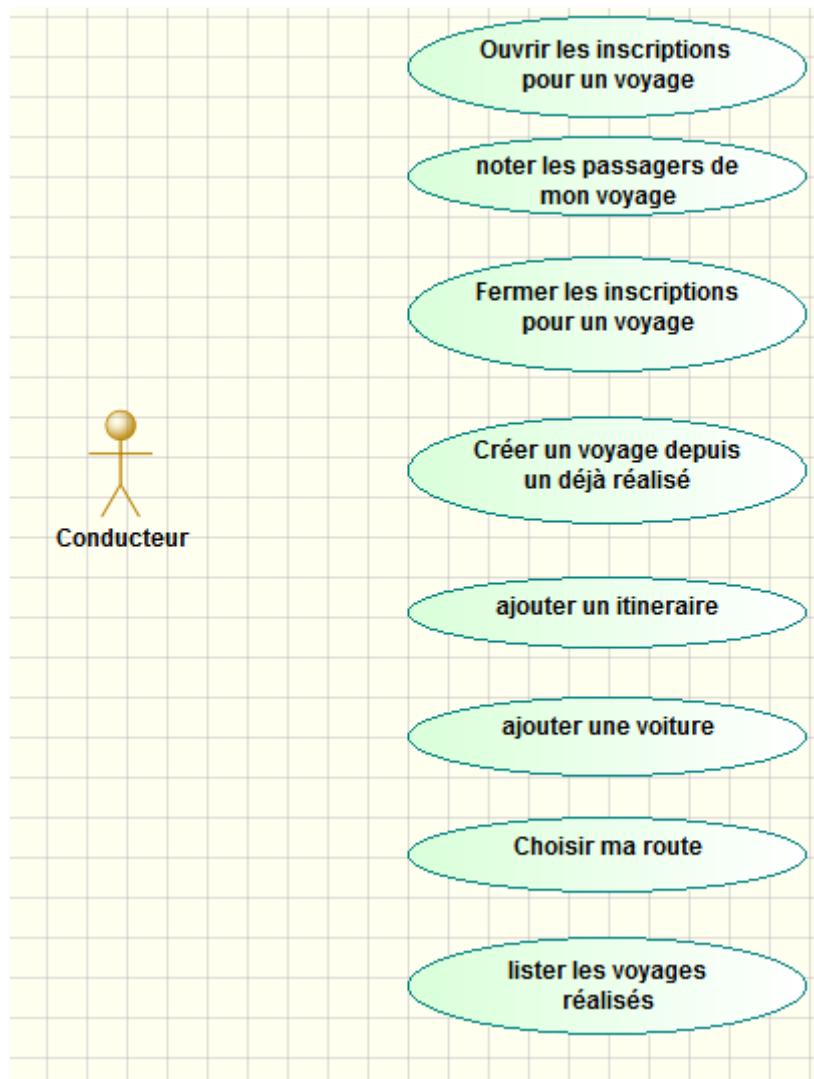
UN ADMINISTRATEUR, EN PLUS DE DISPOSER DES MÊMES DROITS QUE LES AUTRES UTILISATEURS, PEUT ADMINISTRER LE SITE (GESTION DES COMPTES, DES TRAJETS PROPOSÉS, MODÉRATION, ETC...)



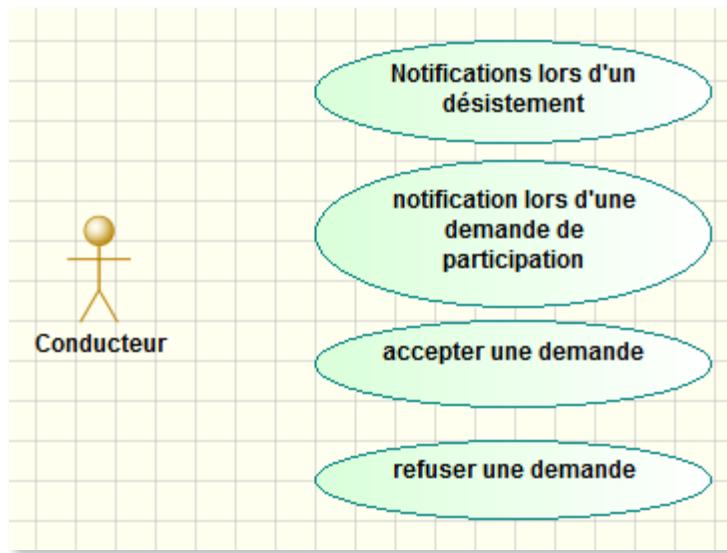
UN PASSAGER PEUT S'INSCRIRE À UN VOYAGE OU ANNULER SON INSCRIPTION.
IL PEUT AUSSI DEMANDER UN DÉTOUR AU CONDUCTEUR ET FAIRE UNE CONTRE-OFFRE
POUR LE PRIX DU VOYAGE. LE PASSAGER PEUT ENFIN CHOISIR DE RÉGLER SA COURSE
SUR LE SITE ET NOTER LE CONDUCTEUR UNE FOIS LA COURSE TERMINÉE.



UN PASSAGER PEUT PARCOURIR LES ITINÉRAIRES RÉCENTS, RECHERCHER UN
ITINÉRAIRE ET FILTRER LES ITINÉRAIRES SELON SES CRITÈRES (ANIMAUX, FUMEUR, ETC.)



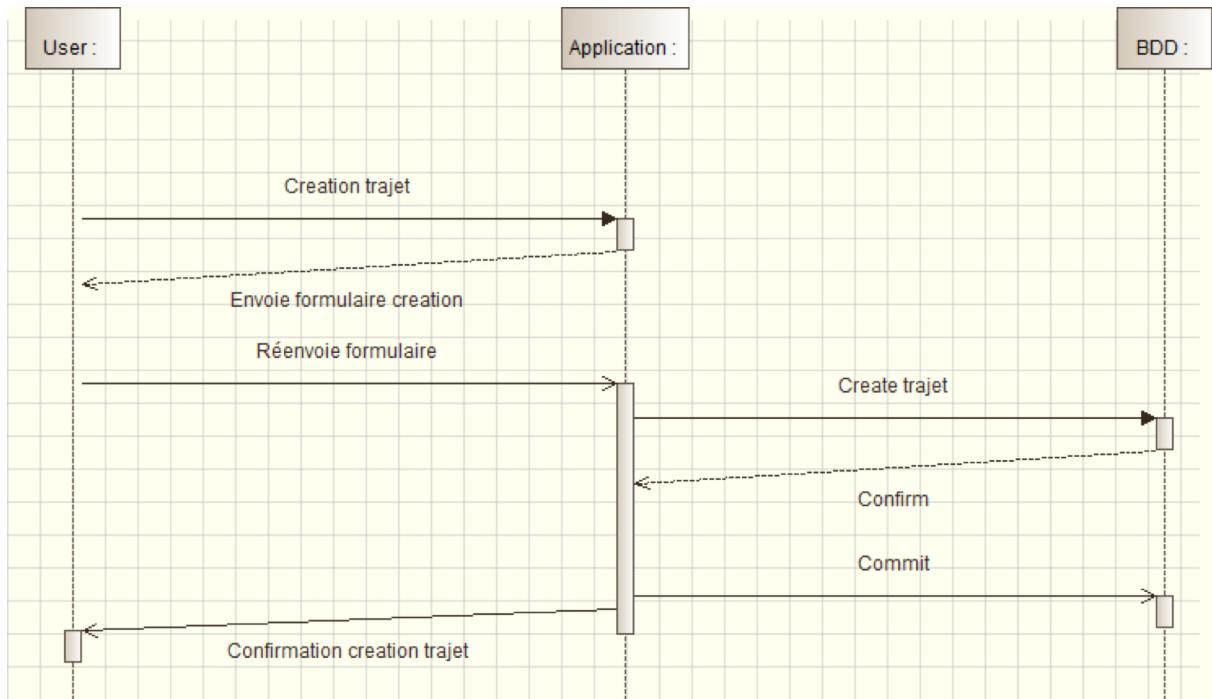
UN CONDUCTEUR PEUT OUVRIR ET FERMER LES INSCRIPTIONS À UN VOYAGE, CRÉER UN VOYAGE À PARTIR D'UN DÉJÀ RÉALISÉ, AJOUTER UN ITINÉRAIRE, UNE VOITURE, AINSI QUE LISTER LES VOYAGES QU'IL A RÉALISÉ ET NOTER LES PASSAGERS.



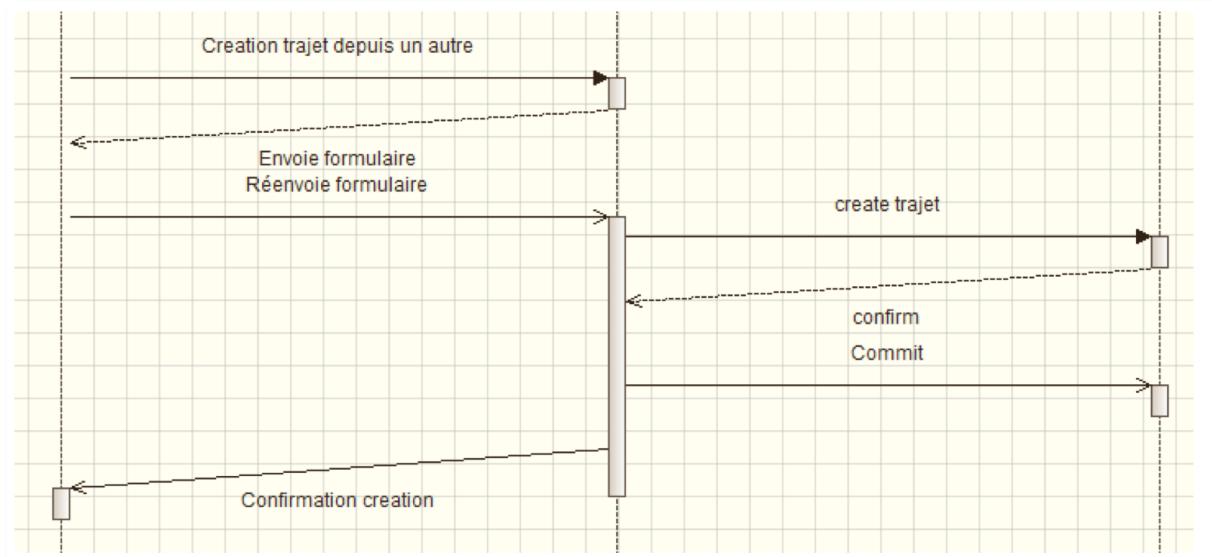
UN CONDUCTEUR PEUT ACCEPTER OU REFUSER UNE DEMANDE POUR UN VOYAGE. IL EST AUSSI NOTIFIÉ LORS D'UNE DEMANDE ET/OU D'UN DÉSISTEMENT.

B. Diagramme des séquences

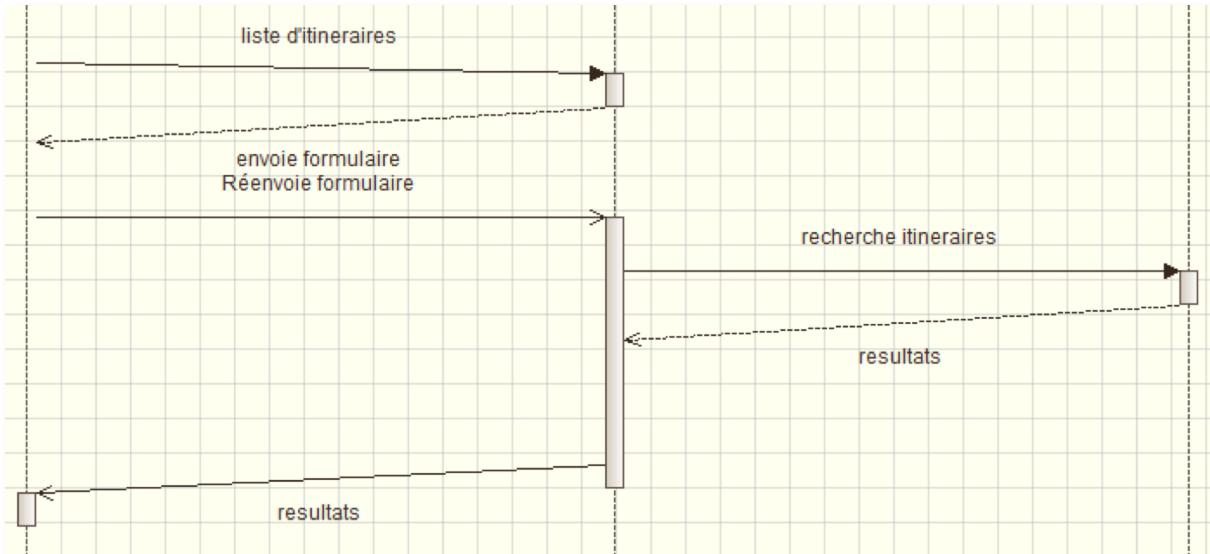
a. Trajets



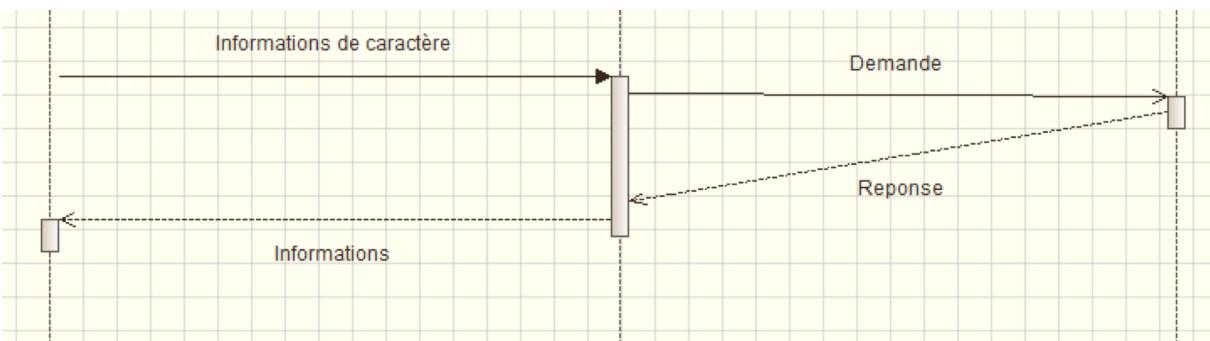
CE DIAGRAMME MONTRÉ LA CRÉATION D'UN TRAJET. LORS D'UNE DEMANDE DE CRÉATION UN FORMULAIRE EST RENVOYÉ PAR L'APPLICATION. LE CONDUCTEUR APRÈS AVOIR REMPLI LE FORMULAIRE LE REVOIE A L'APPLICATION QUI CRÉER LE TRAJET ET ENVOIE LA CONFIRMATION AU CONDUCTEUR



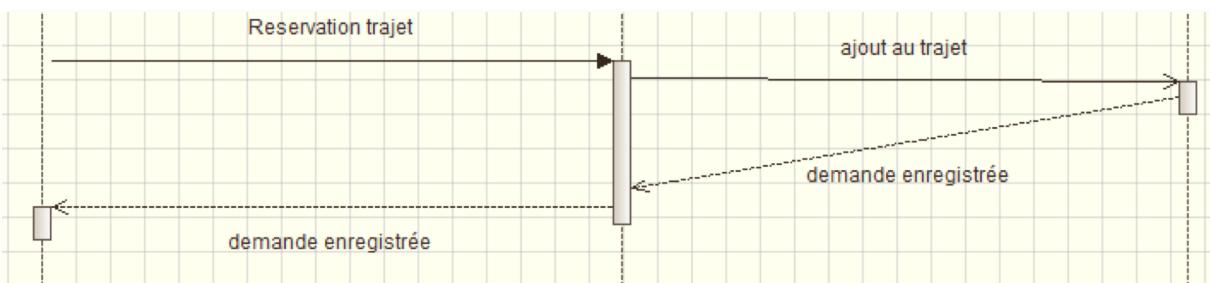
CE DIAGRAMME MONTRÉ LA MÊME CHOSE QU'AU-DESSUS SAUF QUE DANS CE CAS-CI LES ÉLÉMENTS SONT PRÉ REMPLIS PAR UN ANCIEN TRAJET CHOISI PAR LE CONDUCTEUR.



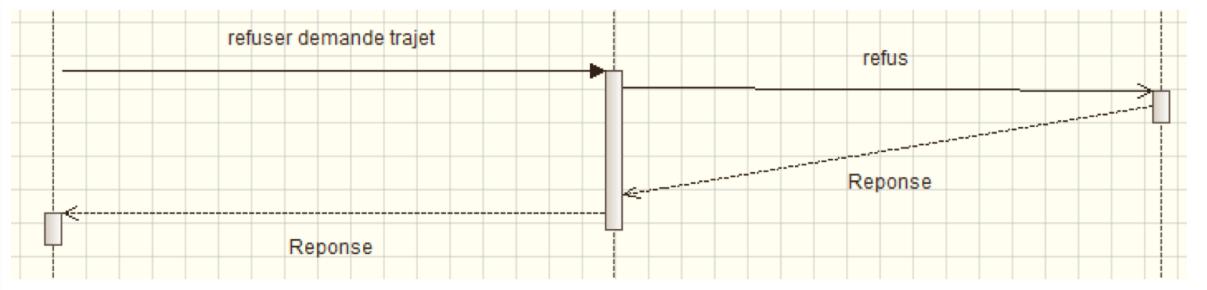
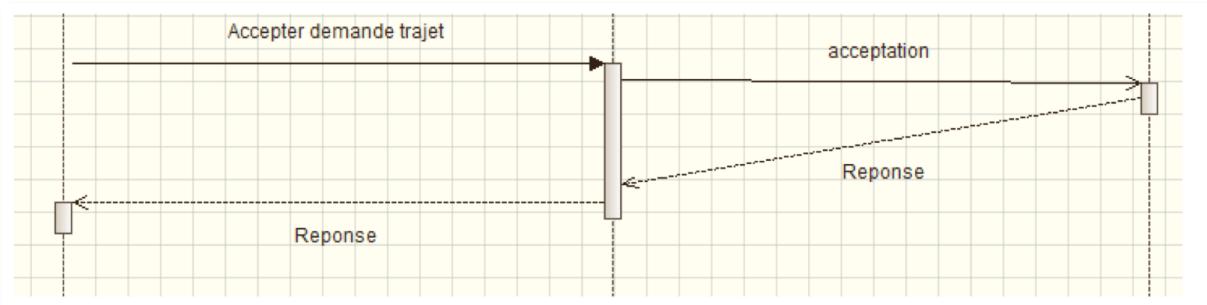
LE PASSAGER PEUT VOIR LA LISTE DES TRAJETS PROPOSÉS. UN FORMULAIRE EST PRÉVU POUR FAIRE DES RECHERCHES DANS LES TRAJETS.



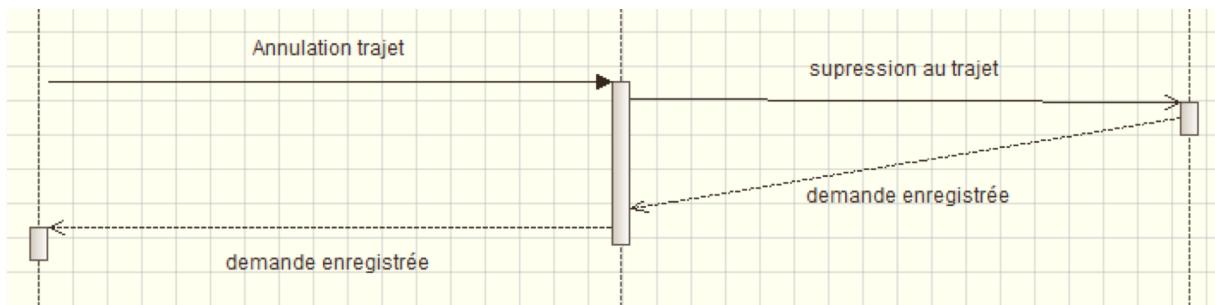
LE PASSAGER PEUT ENSUITE DEMANDER LES INFORMATIONS POUR LE TRAJET CHOISI.



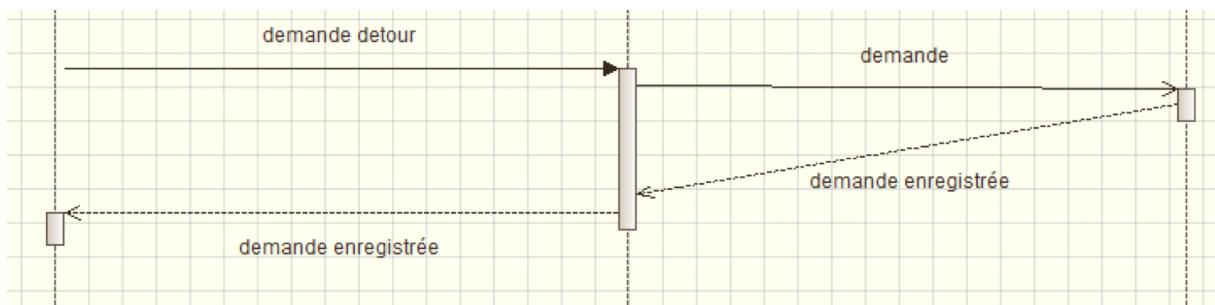
POUR RÉSERVER UN TRAJET LE PASSAGER ENVOIE UNE DEMANDE À L'APPLICATION QUI ENVOIE L'AJOUT AU TRAJET À LA BASE DE DONNÉES. UN RETOUR EST RENVOYÉ À L'APPLICATION PUIS À L'UTILISATEUR.



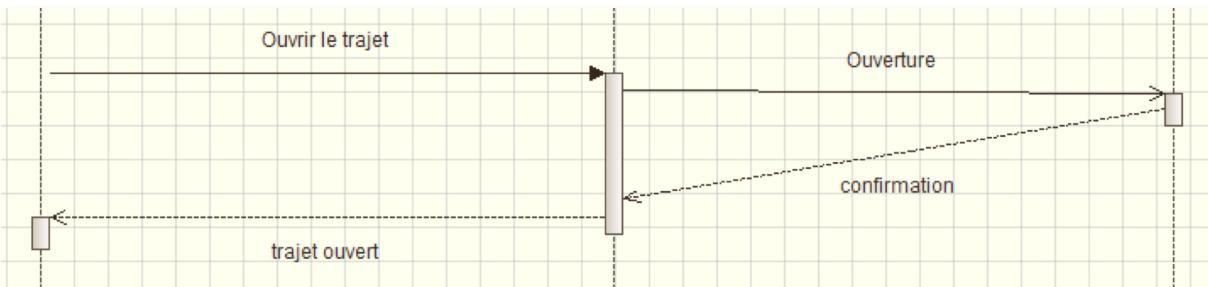
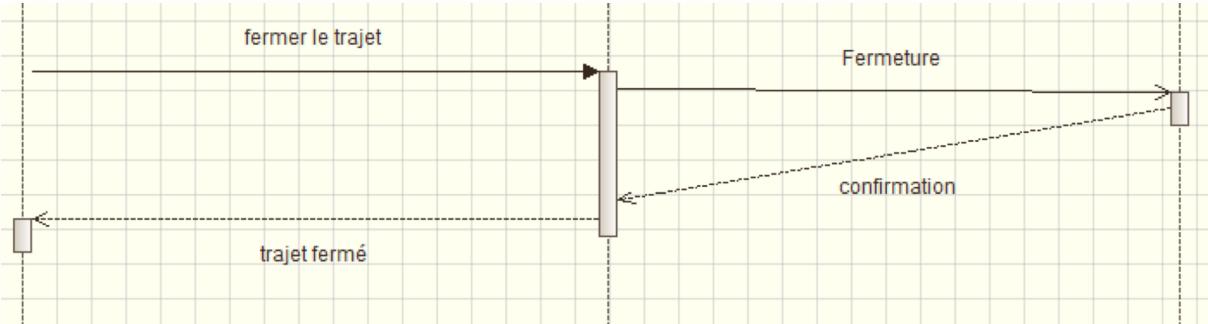
LE CONDUCTEUR PEUT ALORS ACCEPTER LA DEMANDE OU LA REFUSER.



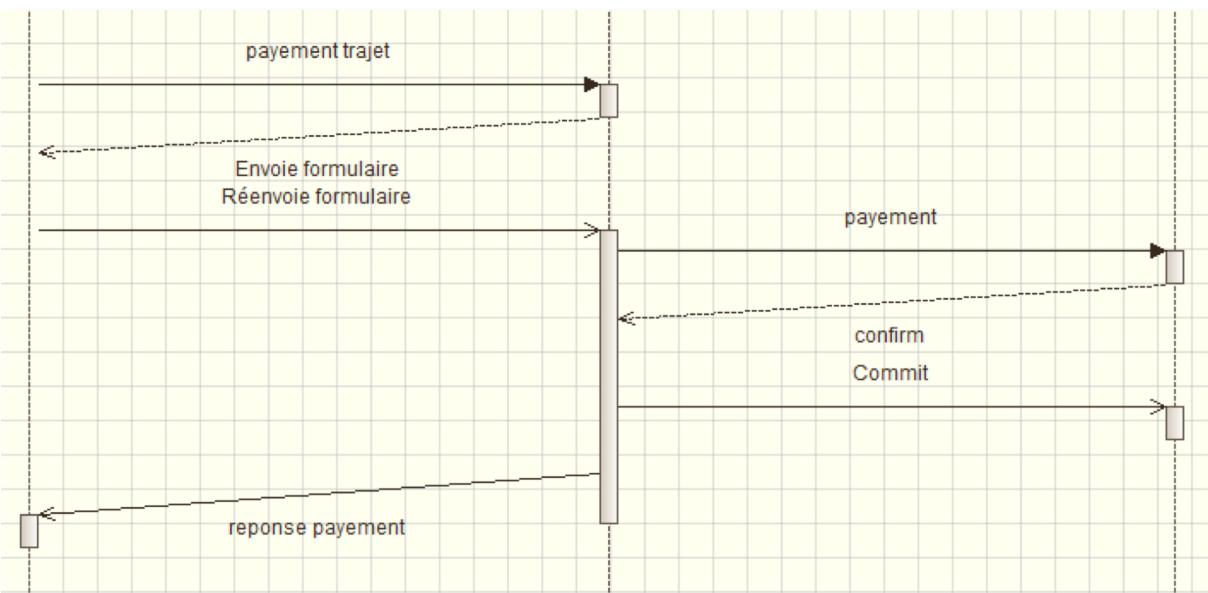
LE PASSAGER PEUT AUSSI ANNULER LE TRAJET.



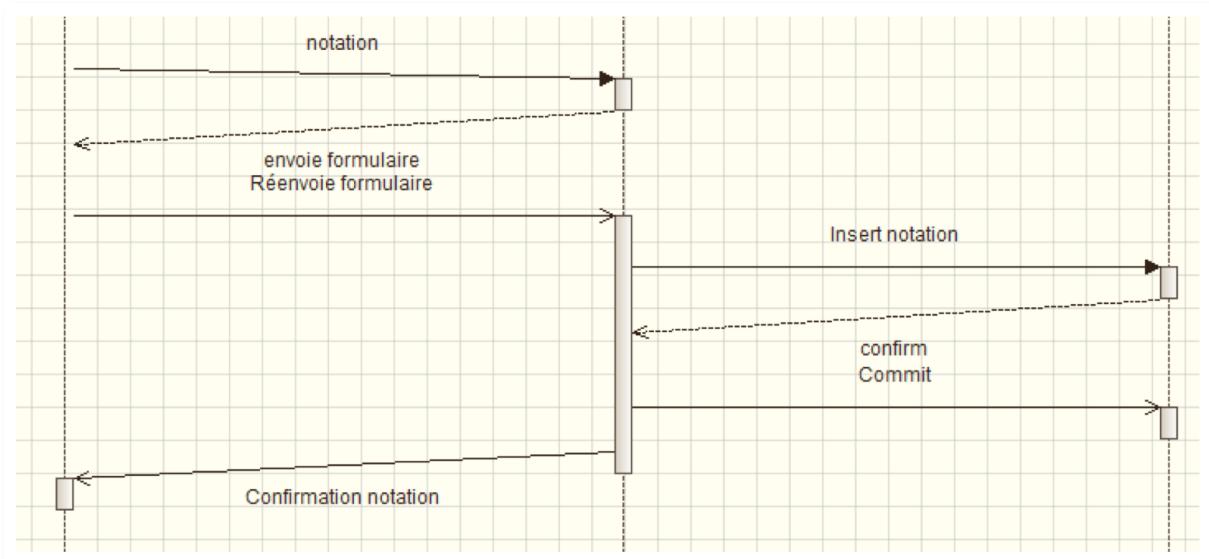
IL PEUT AUSSI DEMANDER UN DÉTOUR AU CONDUCTEUR.



LE CONDUCTEUR PEUT OUVRIR LE TRAJET OU LE FERMER AUX INSCRIPTIONS.

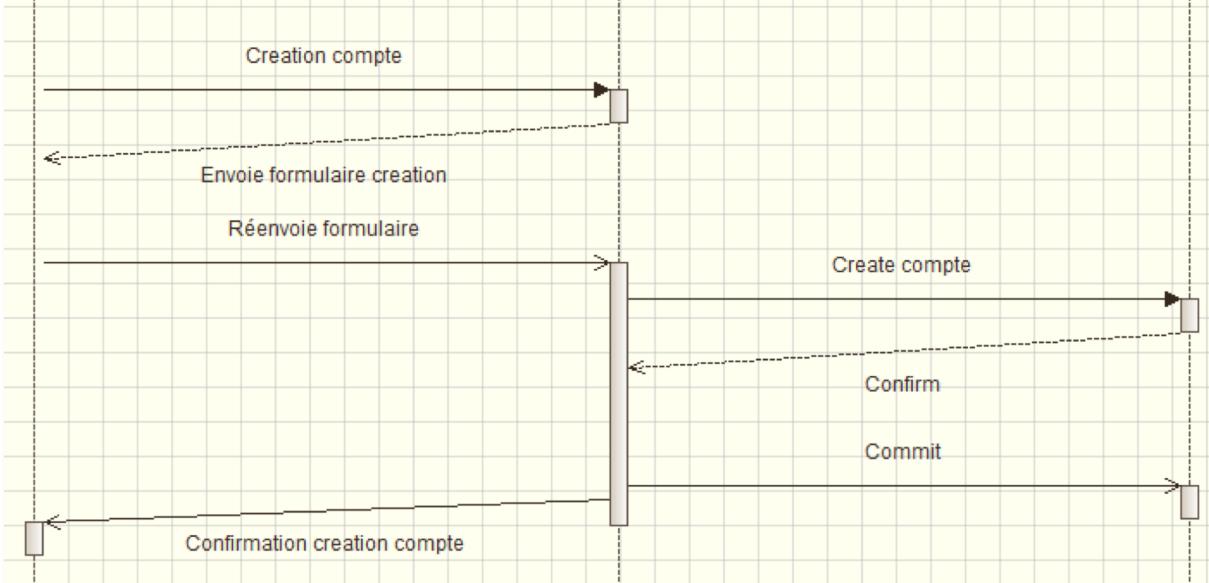


LE PASSAGER PEUT PAYER SON TRAJET AVANT LE DÉPART.
DANS CE CAS, UN FORMULAIRE DE PAIEMENT EST ENVOYÉ À L'UTILISATEUR QUI LE
RENVOI À L'APPLICATION REMPLIE. L'APPLICATION SAUVEGARDE LE PAIEMENT ET
RENVOI UNE RÉPONSE AU PASSAGER.

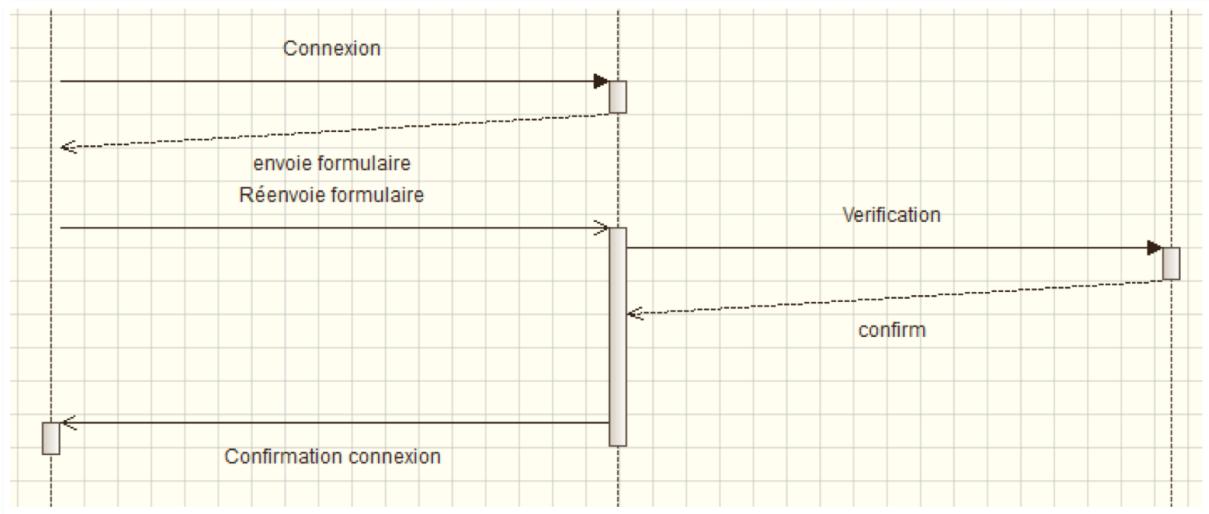


A LA FIN D'UN VOYAGE, LES UTILISATEURS PEUVENT SE NOTER ENTRE EUX.

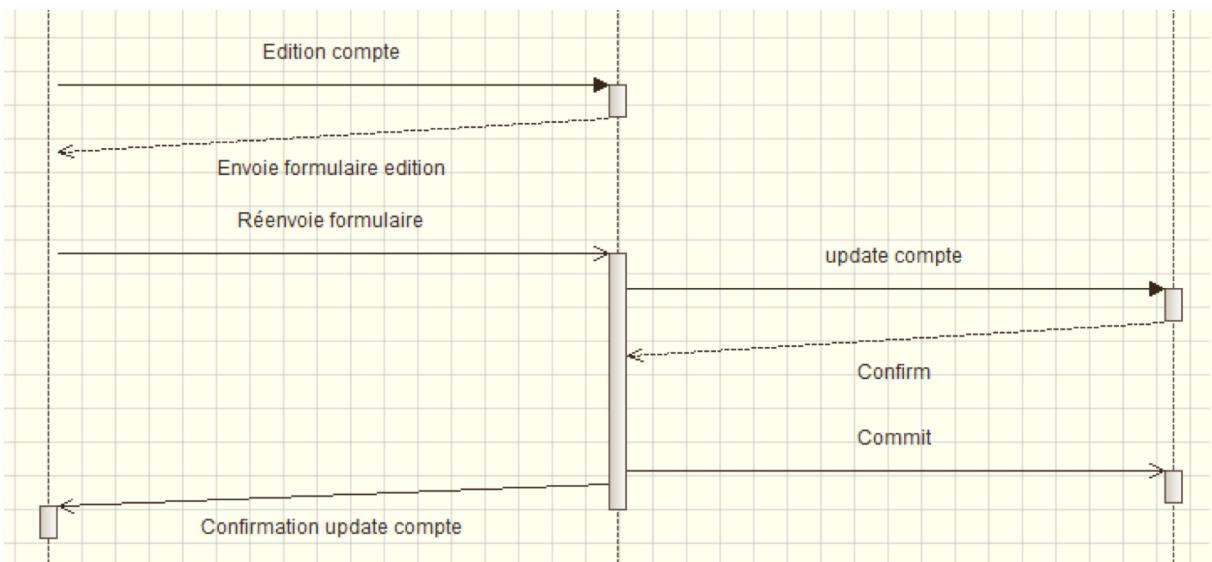
b. Compte



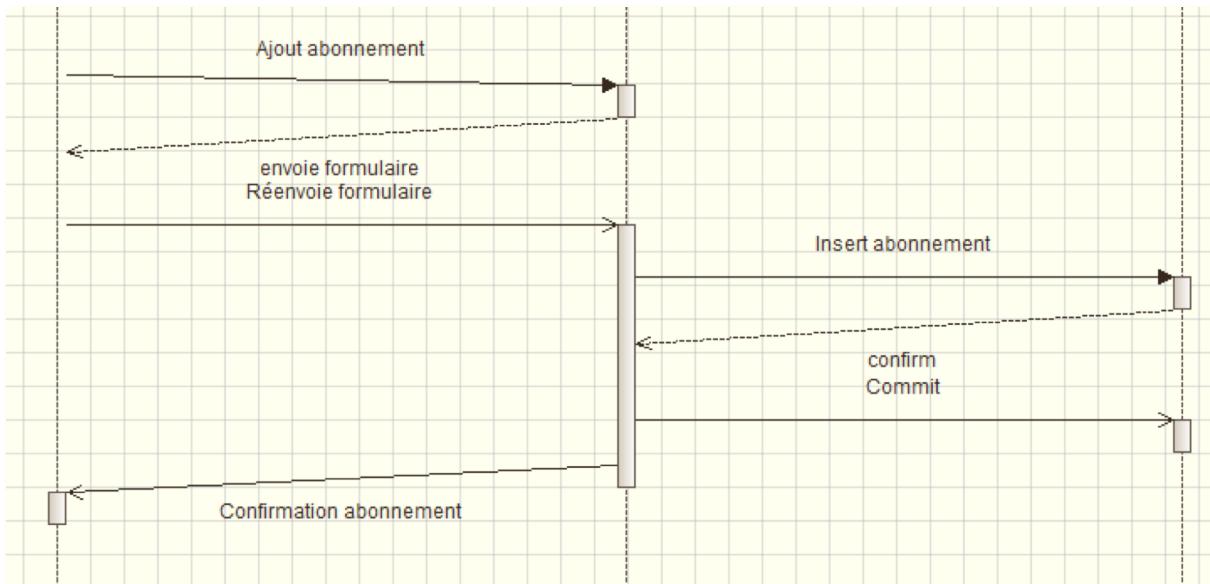
UN UTILISATEUR PEUT CRÉER SON COMPTE.



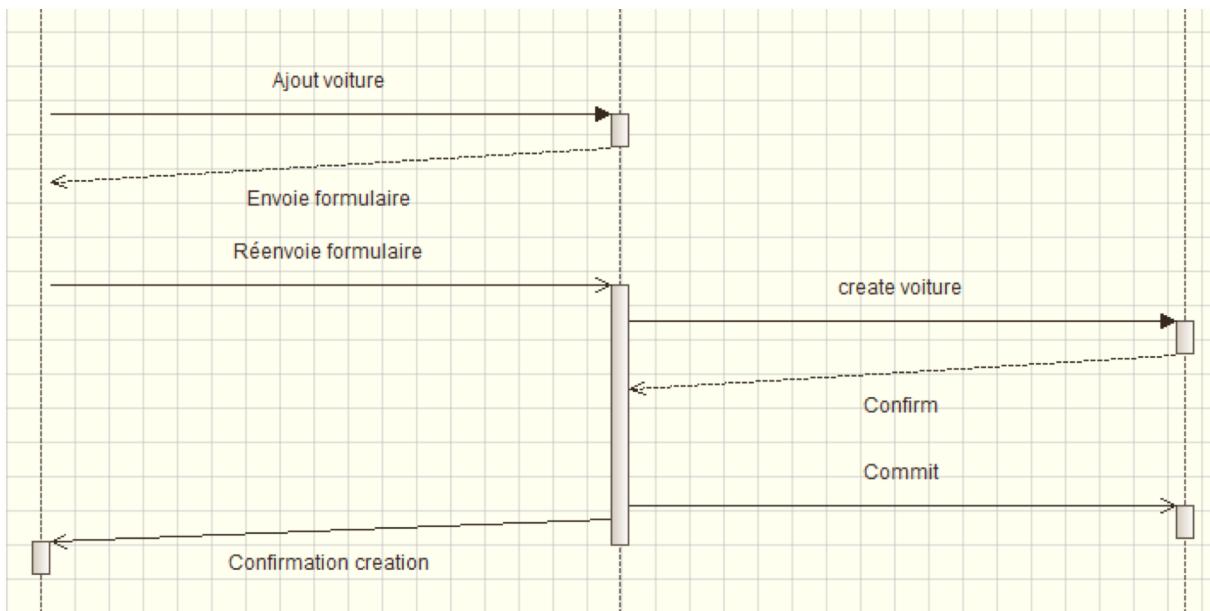
**IL PEUT AUSSI SE CONNECTER À SON COMPTE.
LA CONNEXION À SON COMPTE Étant SÉCURISÉE, CHAQUE CONNEXION RÉUSSIE EST
ENREGISTRER DANS LE JOURNAL DES CONNEXIONS.**



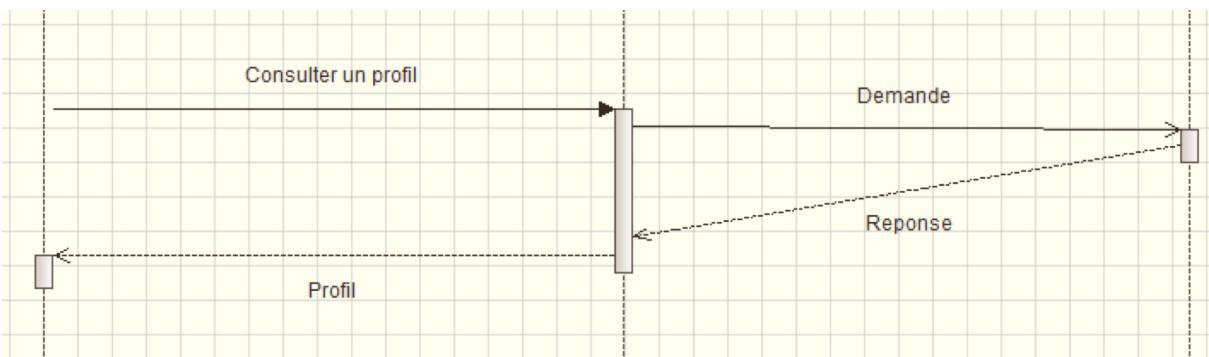
LES UTILISATEURS PEUVENT ÉDITER LEURS COMPTES.



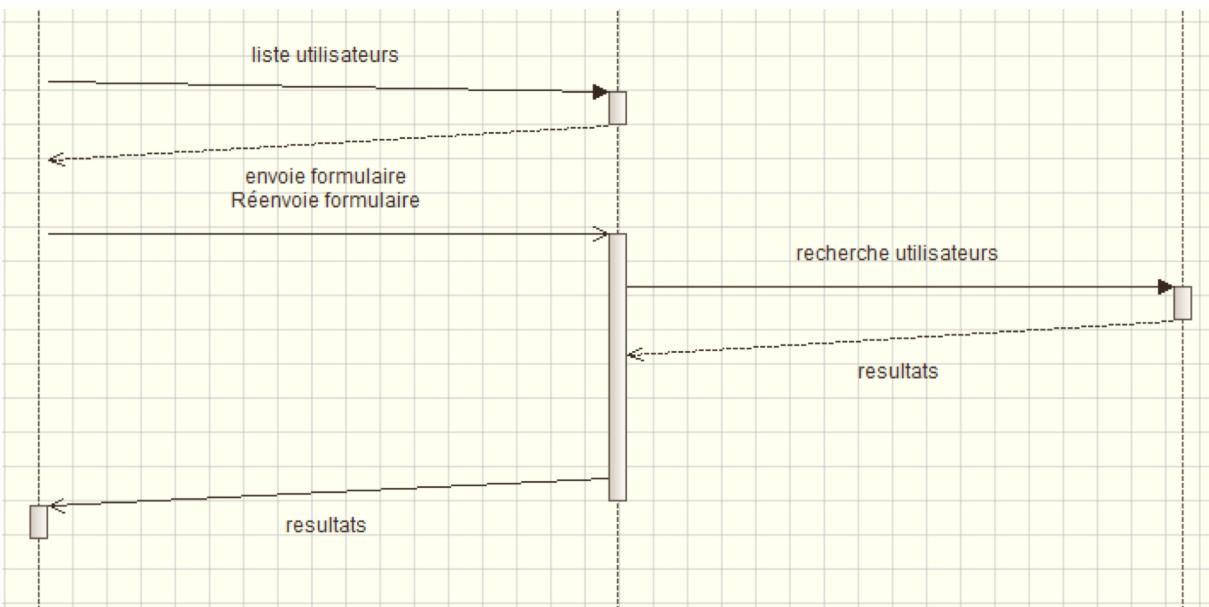
LES PASSAGERS PEUVENT S'ABONNER À L'APPLICATION.



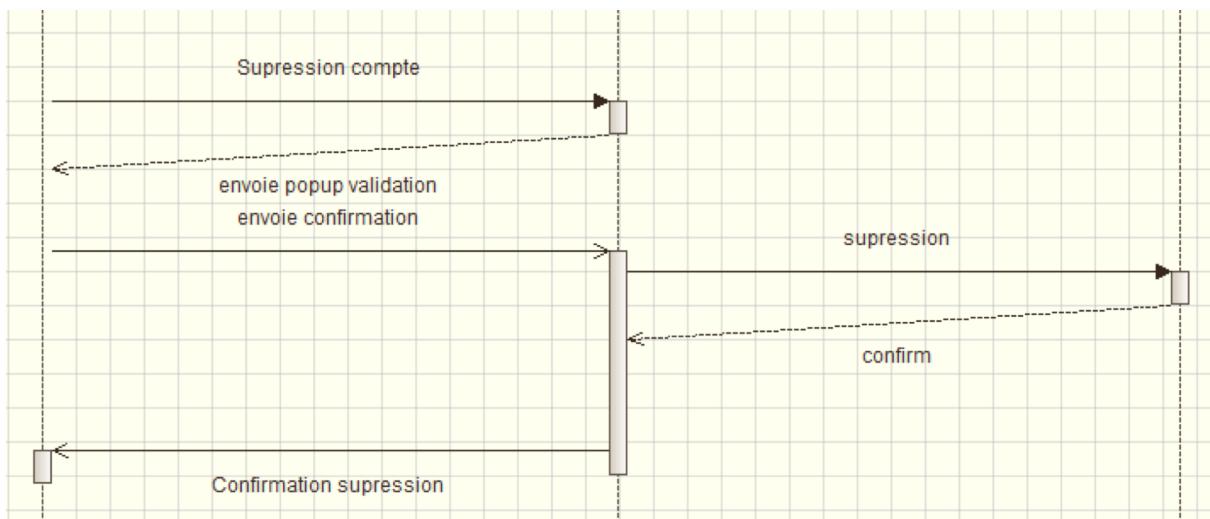
LES CONDUCTEURS PEUVENT AJOUTER UNE VOITURE À LEUR COMPTE.



TOUS LES UTILISATEURS PEUVENT CONSULTER LEURS PROPRES PROFILS ET CELUI DES AUTRES UTILISATEURS.



**L'ADMINISTRATEUR PEUT GÉRER LES COMPTES UTILISATEURS.
POUR CELA IL LISTE LES UTILISATEURS EXISTANTS ET PEUVENT ÉDITER LEURS
INFORMATIONS, VOIR SUPPRIMER LEUR COMPTE.**



LES UTILISATEURS PEUVENT SUPPRIMER LEUR COMPTE.

C. Maquette et description

BlablaCool

Bienvenue sur BlablaCool !

BlablaCool est un nouveau service de covoiturage qui privilégie...

Se connecter

Adresse email

Mot de passe

Se connecter

Créer un compte

Pseudo

Prénom

Nom

Adresse email

Mot de passe

Confirmation

Créer un compte

The form layout is as follows:

- BlablaCool** logo with a checkmark icon.
- Bienvenue sur BlablaCool !** welcome message.
- BlablaCool est un nouveau service de covoiturage qui privilégie...** service description.
- Se connecter** (Connect) button.
- Créer un compte** (Create account) button.
- Adresse email** (Email address) input field.
- Mot de passe** (Password) input field.
- Pseudo** (Nickname) input field.
- Prénom** (First name) input field.
- Nom** (Last name) input field.
- Adresse email** (Email address) input field.
- Mot de passe** (Password) input field.
- Confirmation** (Confirmation) input field.

L'ENREGISTREMENT ET LA CONNEXION ONT LIEU SUR LA MÊME PAGE, DE SORTE QUE L'UTILISATEUR PUISSE RAPIDEMENT SOIT SE CONNECTER OU S'ENREGISTRER SANS "JONGLER" ENTRE DEUX PAGES.

Lors de la création de compte, l'utilisateur est invité à saisir un pseudo, son prénom, nom, adresse e-mail et mot de passe. Nous utilisons une validation en temps réel qui indique à l'internaute les champs qui posent problème (champs requis et vide, adresse email ou pseudo déjà enregistrés, mots de passe non concordants).

Blablacool

Compte Logout

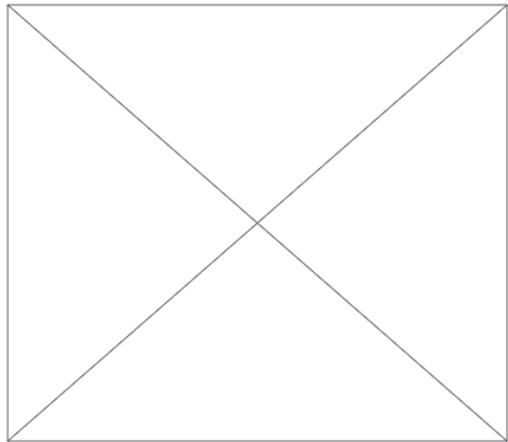
Créer un itinéraire

1. Paramètres

Voiture	Options
Voiture <input type="button" value="Choisir un véhicule"/>	<input checked="" type="checkbox"/> Accepte les fumeurs <input checked="" type="checkbox"/> Accepte les animaux Bagages <input type="button" value="Classiques"/>
Capacité <input type="text"/>	Prix <input type="text"/>

2. Etapes du parcours

Point de Départ	Point de d'Arrivée
Adresse <input type="text"/> Date <input type="text" value="30/03/2015"/> Heure <input type="text"/>	Adresse <input type="text"/> Date <input type="text" value="30/03/2015"/> Heure <input type="text"/>
<input style="width: 100px; height: 20px; margin-left: 10px;" type="button" value="+"/>	
<input style="width: 100px; height: 20px; margin-top: 10px;" type="button" value="Créer mon itinéraire"/>	



LA CRÉATION D'ITINÉRAIRES PERMET DE PRÉCISER LES INFORMATIONS GÉNÉRALES AINSI QUE LES ÉTAPES DU TRAJET.

L'utilisateur doit choisir un véhicule et le nombre de places disponibles. Il peut préciser certaines options (accepte les animaux, les fumeurs), le nombre de bagages autorisés et enfin fixer le prix du trajet. Ce dernier peut néanmoins être calculé automatiquement en fonction des étapes du trajet.

Il est possible d'ajouter autant de points d'arrêt que nécessaire. Un champs d'autocomplétion permet de préciser facilement une adresse qui sera ensuite ajouté à la carte Google Maps responsable de l'affichage de l'itinéraire.

Ajouter une Voiture

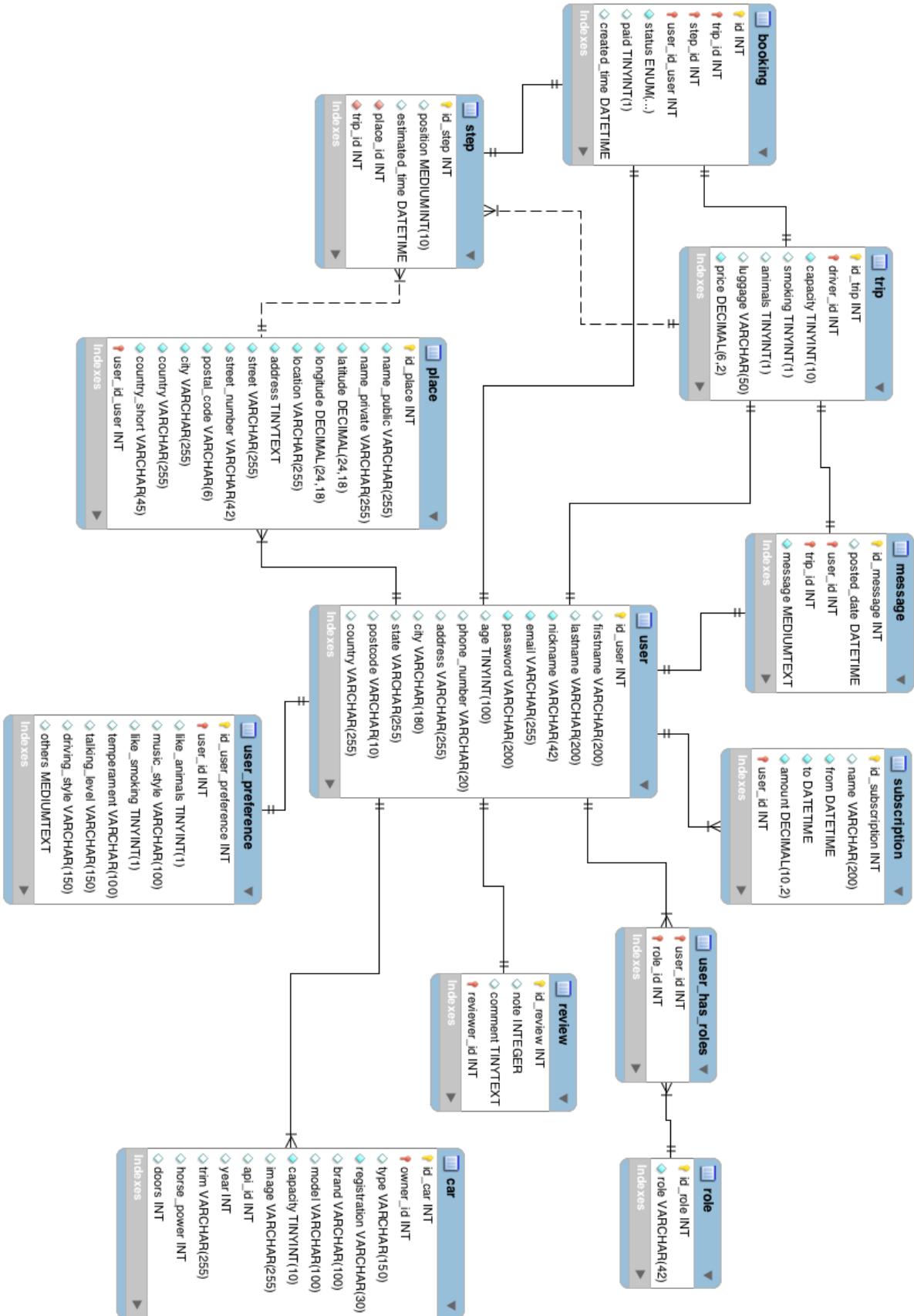
Voiture

Marque	Peugeot
Modèle	207
Année	2011
Portes	4
Places	2
Carburant	Diesel
Puissance	112
Type	
Version	CC 1.6 HDi
Immatriculation	AA-0000-AA

LA PAGE D'AJOUT DE VÉHICULE COMPORTE PLUSIEURS CHAMPS DANS LESQUELS L'UTILISATEUR PEUT PRÉCISER LES CARACTÉRISTIQUES DE SA VOITURE.

Avant de proposer des voyages, un Conducteur doit d'abord renseigner une voiture. Nous pouvons récupérer l'ensemble des données d'un véhicule grâce à une API externe (www.carqueryapi.com) qui contient une liste exhaustives des véhicules existants et leurs caractéristiques.

D. Base de données.



E. Périmètre

Nous sommes partis sur un projet initial assez simple : créer un site de covoiturage facile d'utilisation et complet. Pour cela, nous nous sommes concentré sur le point central de tout site web : l'utilisateur. Qu'il soit conducteur ou passager, les utilisateurs de notre site sont le centre de celui-ci. Chaque fonctionnalité apportée à BlablaCool n'a qu'un seul but, celui de faciliter la navigation sur le site par extension le covoiturage.

Un utilisateur pourra donc très facilement et de manière tout à fait autonome gérer son propre compte, ses trajets en tant que passager et/ou conducteur ou encore ses voitures si il dispose d'un profil Conducteur.

Les utilisateurs peuvent en outre parcourir les trajets disponibles et surtout s'y inscrire en tant que passager. Il peut très facilement consulter les trajets disponibles à partir d'une adresse de destination et d'arrivée ou peut même être géolocalisé pour lui éviter de saisir son point de départ. A charge ensuite pour le conducteur d'accepter ou non la demande de covoiturage.

II. Organisation des tâches

A. L'équipe de travail

Notre équipe est composée de 4 personnes où chacune c'est vu attribuer un rôle; toujours dans un soucis d'efficacité :

Team et Tech Lead	Valentin Polo
Tests et Développement	Nicolas Ducom
Intégration et Développement	Hugo Drécourt
Fonctionnel	El Mamoun El Bekri El Alaoui

B. Choix de la méthode de Gestion de Projet

Pour pouvoir délivrer BlablaCool dans les temps, nous avons choisi d'utiliser la méthode Agile Scrum. Celle ci nous permet d'avoir un projet fonctionnel dès le départ qui sera enrichi au fur et à mesure des itérations, autrement appelées Sprint.

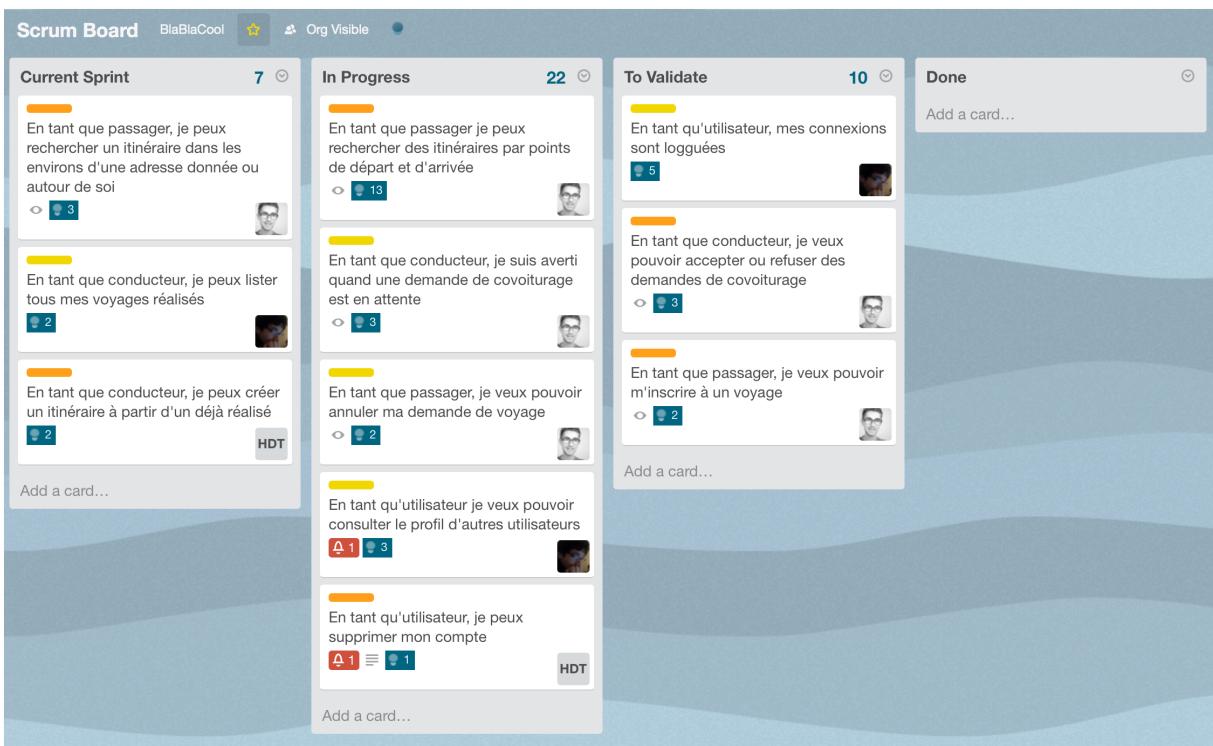
Nous avons constitués notre *Backlog Produit* au lancement du projet qui contient toutes les *User Stories* qui devaient être mises en place dans BlablaCool (telles que "En tant qu'utilisateur, je dois pouvoir me créer un compte" ou "En tant que conducteur je peux accepter ou refuser une demande de covoiturage", etc...). Chaque *User Story* s'est vu attribuer des *Story Points* qui permettent de quantifier sa complexité. Dès lors, une fois tous les *Story Points* attribués, nous sommes arrivés à un total de 137 points.

Comme il a été décidé de développer ce projet en un maximum de trois Sprints de une semaine chacun, nous devions nous assurer d'avoir un ensemble de *User Stories* qui comportaient 46 *Story Points* par Sprint.

Le premier *Sprint* contiendrait l'ensemble des *User Stories* de base nécessaires au bon fonctionnement du site et que nous estimions prioritaires (comme la mise en place du thème, de la DAO de la connexion et enregistrement des utilisateurs).

Les *Sprints* suivants bien qu'aussi soutenus en charge de travail permettraient d'apporter des fonctionnalités importantes mais non vitales (tel que le filtrage des résultats de recherche par exemple).

Afin d'organiser notre *Sprint Board* au mieux, nous avons décidé d'utiliser Trello qui permet de manipuler nos *Sprints* et *User Stories* facilement.



PRÉSENTATION DE TRELLLO

C. Décomposition des tâches

Chaque *User Story* du *Backlog* est composée de *Story Points* mais aussi d'un label caractérisant sa priorité :



Les *User Stories* prioritaires sont celles que nous avons estimées vitales et qui ont été pour la plupart réalisées lors du premier *Sprint*. Par exemple, la gestion du compte utilisateur est nécessaire pour créer un voyage ou s'y inscrire.

Les *User Stories* importantes sont celles qui sont au cœur du site et qui ont un lien direct avec le système de covoiturage : enregistrement et connexion de comptes utilisateurs, inscription à un trajet, recherche d'itinéraires, gestion des abonnements, etc...

Les *User Stories* normales quant à elles apportent des fonctionnalités qui facilitent la navigation sur BlablaCool.

Enfin les *User Stories* bonus concernent des fonctionnalités supplémentaires qui ne sont pas forcément nécessaire au projet, mais qui apportent des petits plus et qui pourraient inciter l'utilisateur à naviguer sur le site. Elles sont en quantités limitées pour ce projet.

D. Description des tâches

Une *User Story* correspond en général à une fonctionnalité, la description des tâches a été relativement simple étant donné que la description de la fonctionnalité est claire.

III. Stack technique

Sans vouloir être trop exhaustif, certains choix techniques nous ont été imposés tandis que d'autres choisis. Nous allons revenir ici sur certains d'entre eux que nous estimons importants et tenter de les justifier.

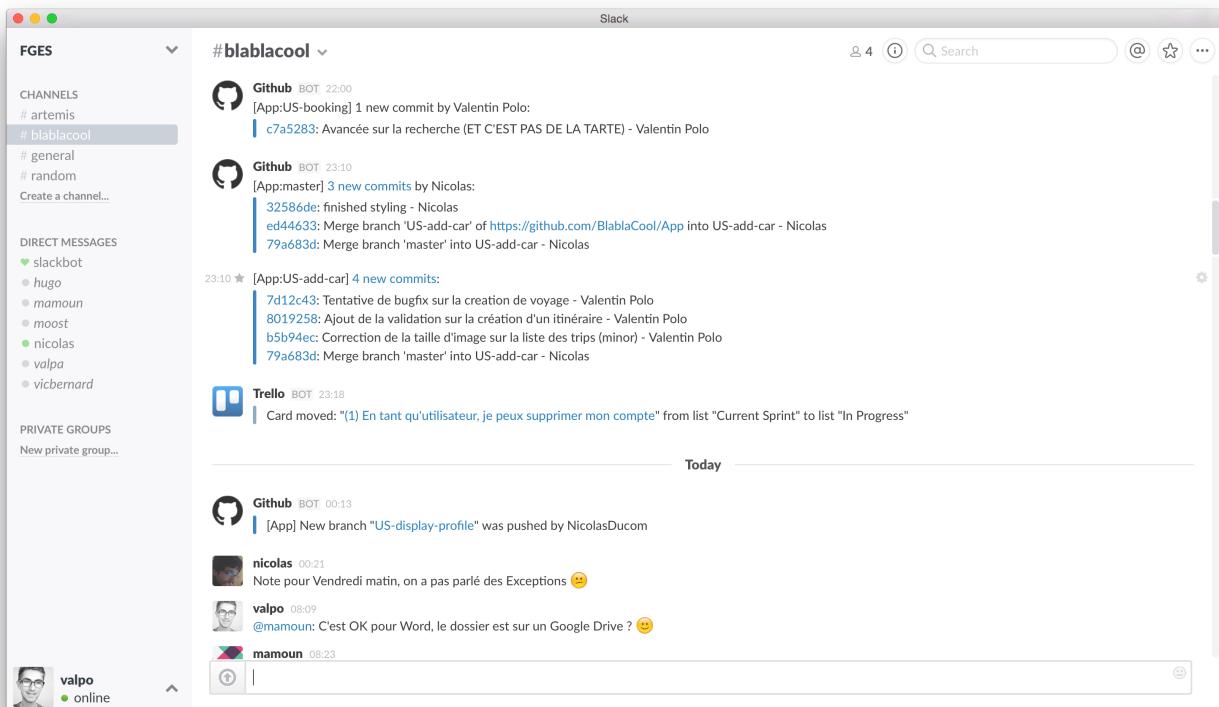
	J2EE + Spring MVC BlablaCool est une application J2EE basée sous le framework Spring MVC. Comme tout framework, celui nous a permis de démarrer rapidement notre projet en ayant les bases "architecturales" nécessaires à son bon fonctionnement.
	MySQL + Hibernate De nombreux systèmes de base de données existent, mais MySQL était une évidence pour nous de part notre connaissance plus accrue de ce système par rapport à d'autres. Un système NoSQL (de type MongoDB) n'aurait pu convenir de part les nombreuses relations à gérer au sein de l'application).
	Tiles Au moment de l'intégration du template, nous nous sommes posés la question de savoir quel était le système qui alliait simplicité d'utilisation et potentiel; Tiles a su tirer son épingle du jeu en proposant un système de templating efficace tout en restant simple à appréhender.
	Google Maps Qui dit covoiturage dit plan et itinéraire puisqu'il est tout de même plus agréable de pouvoir visionner un trajet directement sur une carte. Quelques solutions existent comme Open Street Map ou Bing Maps mais peu étaient aussi riches que celle de Google avec Google Maps. Cette dernière dispose d'une API JavaScript bien documentée qu'il aura été facile d'apprivoiser.
	Mandrill L'envoi de notifications sur BlablaCool passe par l'envoi de mails. Afin de nous assurer de leurs bonnes délivrabilité, nous avons pris la décision d'utiliser un prestataire externe pour l'envoi de mails. De nombreux services existent mais rares sont ceux qui proposent l'envoi de 12 000 mails gratuitement par mois, ce qui aura été un facteur décisif. Tout comme la simplicité d'utilisation de leur API REST.
	Stripe Nous offrons aux passagers la possibilité de payer en avance le prix de la course, de sorte qu'ils n'aient pas besoin de prévoir d'argent le jour du départ. PayPal aura pu faire l'affaire comme passerelle de paiement mais nous avons privilégié Stripe qui propose des frais d'utilisation moindres.

IV. Management et communication

A. Communication

La communication est le fondement d'un projet réussi et délivré dans les temps; elle aura donc été un point central pour notre gestion et notre organisation. Nous avions besoin d'un outil qui permettrait de communiquer efficacement et qui pourrait intégrer des services externes.

En effet, comme indiqué précédemment, nous utilisons Trello pour la gestion des *User Stories* mais également Github pour centraliser et collaborer sur notre code source. Ne serait-il pas formidable de trouver un service qui permettrait d'être informé à chaque mise à jour Trello ou *Push* sur Github ? Slack fût la réponse à cette problématique.



APERÇU DE SLACK ET DE SON SYSTÈME D'INTÉGRATION (ICI GITHUB ET TRELLO)

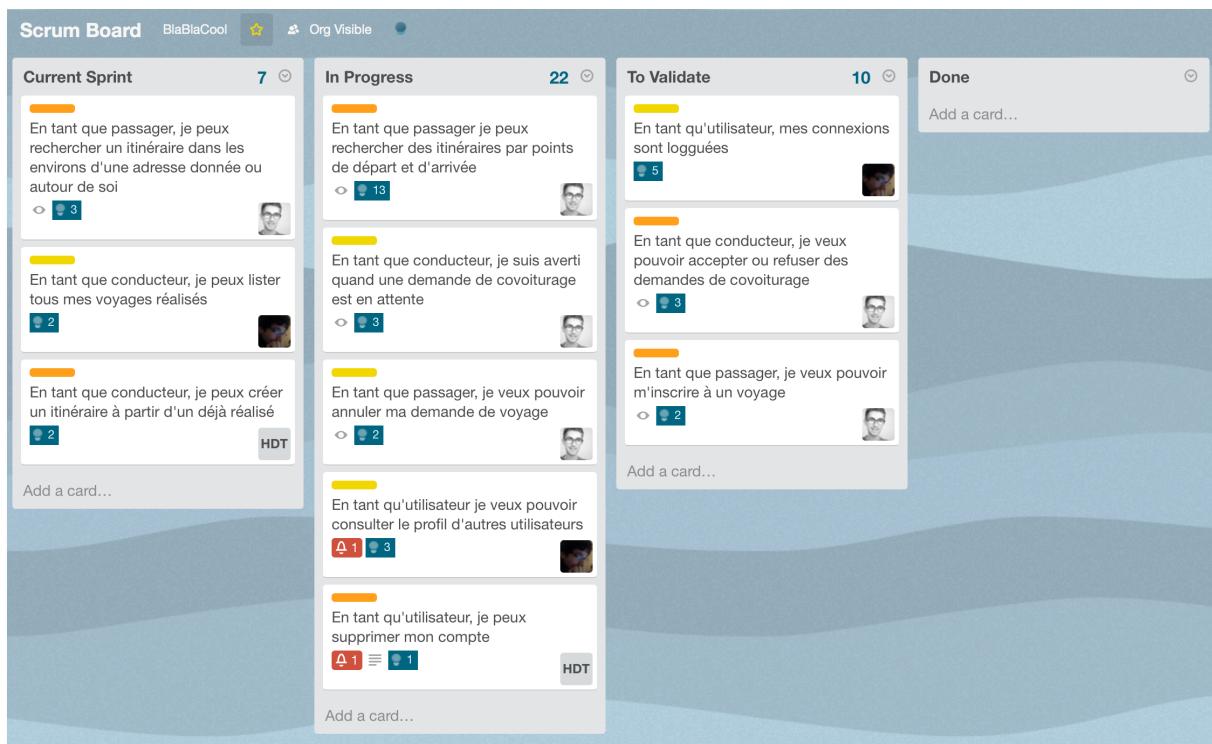
Les cours de Projet Technique nous ont également permis de nous rassembler et de pouvoir travailler ensemble sur des points qui le nécessitaient mais aussi de faire des points récapitulatifs de l'état du projet.

B. Gestion des conflits

Nous avons eu la chance d'avoir très peu voire pas de conflits au sein de notre équipe; nous nous entendons tous bien. Cependant, le fait qu'un de nos membres soit dans une classe différente a compliqué les échanges entre nous au début du projet. Les classes ne se mélangeant que pour deux cours, nous n'avons pas eu vraiment de possibilité de se voir en dehors de ces horaires au début. L'arrivée de Slack nous a permis de mieux nous organiser dans notre travail et dans nos échanges et d'avancer à un bon rythme.

C. Contrôle de l'avancement du projet

Notre Scrum Board nous permet de voir en temps réel l'avancement du projet ainsi que, pour chaque *User Story*, son avancement dans le *Sprint* en cours, ses *Story Points* et le membre responsable du développement de la fonctionnalité. et la personne.

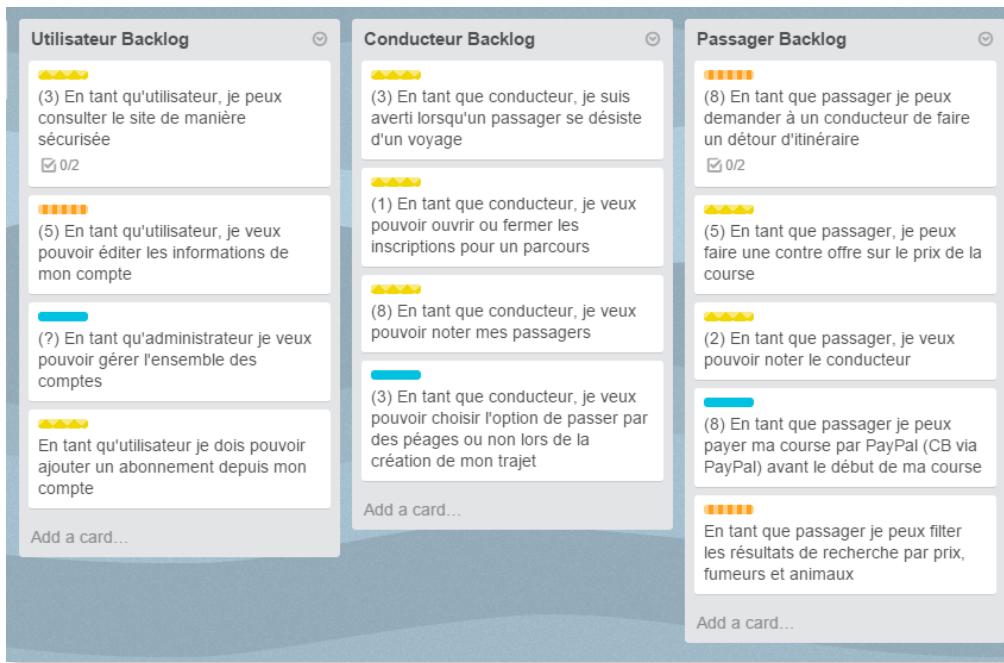


EXTRAIT DE NOTRE SCRUM BOARD PENDANT LE SPRINT 2.

Comme le montre la capture ci-dessus, chaque User Story passe normalement par quatre états représentés par des colonnes. Chaque état étant assez explicite ("Current Sprint", "In Progress", "To Validate", "Done"), nous ne les

détaillerons pas ici. La répartition des *User Stories* se faisait par attrait des membres envers les dites *User Stories* ou sinon (mais très rarement) par choix du Chef d'équipe.

D'autres listes permettent de voir l'avancement de l'ensemble du projet., notamment la liste des *User Stories* des *Sprints* à venir et qui ne font donc pas partie du *Sprint* en cours.



Cette présentation s'est révélée très pratique et nous a permis de savoir à tout moment quelles *User Stories* étaient en cours de développement et ce sur quoi chaque membre avançait.

V. Qualité

Une projet de cette envergure nécessite forcément rigueur de travail et qualité; cela a donc été un axe important au cours du développement de BlablaCool.

Pour cette raison, nous avons mis place les tests unitaires avec l'aide de JUnit. Ceci nous aura permis de tester la notre partie Backend, le coeur de BlablaCool. Ces tests nous auront aussi éviter tout soucis de régression.

JUnit est une librairie de tests unitaires qui permet de mettre en place une série de test sur chaque classe du projet. Les tests unitaires permettent par l'intermédiaire de tests préalablement écrits de s'assurer que le résultat d'une méthode est bien celui escompté et, à défaut, de trouver précisément la fonction qui pose problème à l'intérieur d'une classe donnée.

Ces tests, combinés à des essais relativement poussés nous ont permis de nous assurer de la qualité de notre projet final pendant tout son développement.

VI. Ressources utiles

Scrum Board Trello (public) : <https://trello.com/b/JrRMQsdB/scrum-board>

Repository Github (public) : <https://github.com/BlablaCool/App>