

Projet GLA : GPS-Like

Cahier des charges et Analyse

Projet GLA encadré par Burkhardt Wolff

4 Mars 2019

Groupe 7

Cerveau Eric

Malassé Juliette

Shenjin Lyu

Table des matières

| | | |
|----------|---|----------|
| 1 | Cahier des charges | 2 |
| 1.1 | Objectifs et contexte du produit | 2 |
| 1.1.1 | Cibles du produit | 2 |
| 1.1.2 | Description de la solution | 3 |
| 1.2 | Contraintes | 4 |
| 1.2.1 | Liste des fonctions | 4 |
| 1.2.2 | Liste des critères liés aux fonctions du système GPS-like | 5 |
| 2 | Analyse | 8 |
| 2.1 | Diagramme de classe | 8 |
| 2.1.1 | Utilisateur | 9 |
| 2.1.2 | Historique | 9 |
| 2.1.3 | Véhicule | 9 |
| 2.1.4 | Itinéraire | 10 |
| 2.1.5 | Options | 10 |
| 2.1.6 | Ville | 10 |
| 2.1.7 | Route | 11 |
| 2.1.8 | Tronçon | 11 |
| 2.1.9 | Énumérations : typeVille et typeRoute | 11 |
| 2.2 | Diagramme Etat-Transition | 12 |
| 2.3 | Diagrammes de séquence | 13 |
| 2.3.1 | Inscription | 13 |
| 2.3.2 | Connexion | 15 |
| 2.3.3 | Enregistrer un véhicule | 17 |
| 2.3.4 | Spécifier un trajet | 19 |
| 2.3.5 | Sélectionner un trajet depuis l'historique | 22 |
| 2.3.6 | Entrer des préférences | 25 |
| 2.3.7 | Consulter l'historique | 28 |
| 2.3.8 | Suivre un trajet | 30 |

1 Cahier des charges

1.1 Objectifs et contexte du produit

Le but de ce projet est de concevoir un prototype de GPS, ou Global Positionning System. Cet outil, sous forme d'une application Web, permettra à l'utilisateur de se diriger lors d'un trajet entre deux villes sans se perdre et selon des préférences qu'il aura préalablement choisies. Il répondra à un scénario de fonctionnement précis où l'utilisateur maîtrise chaque étape, et sera soumis à un certain nombre de contraintes. Un soin particulier sera apporté à l'ergonomie et la simplicité de cette solution afin de la rendre accessible au plus grand nombre.

Il s'agit d'un travail en équipe, nous devons donc agir en tant que tel, en se répartissant les tâches équitablement selon les forces et les faiblesses de chacun. Chaque étape du projet, de l'analyse au résultat final, sera le fruit d'une réflexion commune.

1.1.1 Cibles du produit

Notre GPS s'adresse à toute personne souhaitant effectuer un déplacement entre deux villes sans prendre de transports en commun. Grâce au choix du type de routes empruntées, le chemin peut donc se faire aussi bien en voiture qu'à vélo.

Les personnes voulant faire du tourisme sont largement concernées par notre produit, car nous répertorions tous les points d'intérêts, qu'ils soient en ville ou bien sur les routes, et permettons d'ajouter des villes-étapes, fonction idéale pour établir un itinéraire touristique. De plus, la facilité d'utilisation de notre application ne requiert pas une maîtrise parfaite de la langue pour s'en servir à des fins touristiques, nous nous adressons donc aussi bien aux touristes francophones qu'aux étrangers. Enfin, la possibilité d'éviter les péages et le signalement des radars permet de voyager en toute sérénité en limitant ses

frais.

Ce GPS peut également être utilisé par des particuliers ou des professionnels de manière quotidienne, afin d'optimiser leurs déplacements. Cette utilisation est rendue possible par les nombreux paramètres permettant de complètement personnaliser l'application, selon que l'on veuille réduire son empreinte carbone, trouver un trajet plus court, plus rapide, sans radars, sans péages... Les possibilités sont nombreuses et répondent aux besoins de tous les utilisateurs.

1.1.2 Description de la solution

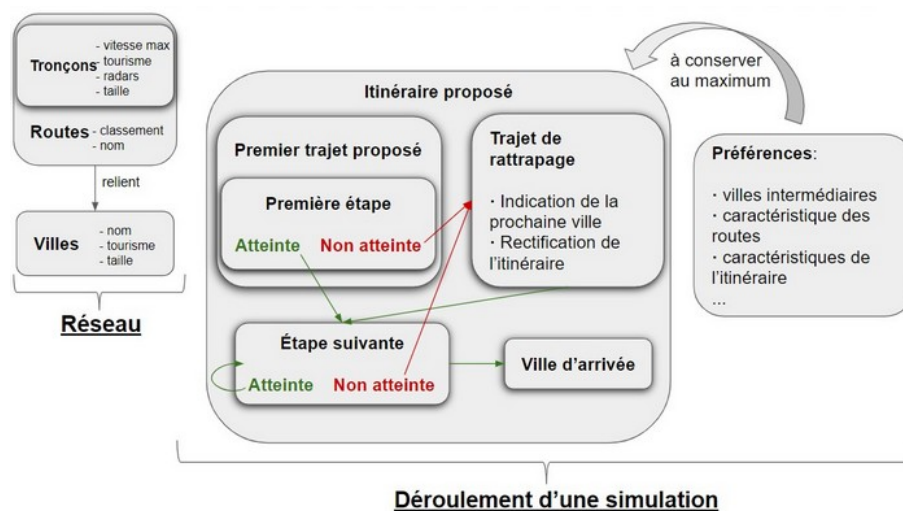
Tout d'abord, le système est basé sur un ensemble de villes reliées par des routes. Elles ont toutes deux des caractéristiques spécifiques afin de les identifier. Les routes sont décomposées en tronçons afin de pouvoir prendre en compte les spécificités de la route à chaque instant. Toutes ces données composent notre réseau et forment la base qui nous permettra de diriger l'utilisateur de façon optimale.

Le fonctionnement de l'application est le suivant :

En premier lieu, l'utilisateur indique ses villes de départ, d'arrivée, ses éventuelles villes-étapes et ses préférences pour le trajet. Le GPS lui proposera un premier trajet qui correspond au mieux à ses préférences, et lui indique si l'une de celles-ci n'a pas pu être respectée.

Ensuite, l'utilisateur indique s'il a bien atteint chaque étape. Si c'est le cas, le GPS lui indique la route pour se rendre à l'étape suivante, sinon, la solution demande quelle est la ville vers laquelle se dirige le conducteur et à partir de cette information, calcule un itinéraire pour rattraper celui du départ, ou bien un itinéraire alternatif. De nouveau, l'application s'efforce de conserver les préférences, et alerte l'utilisateur si c'est impossible.

Ce scénario se répète jusqu'à ce que l'utilisateur ait atteint la ville d'arrivée. Son trajet est ensuite archivé dans l'historique et ses préférences conservées.



1.2 Contraintes

1.2.1 Liste des fonctions

| | |
|-----|---|
| FP1 | Guider l'utilisateur lors d'un trajet d'une ville A à une ville B |
| FC1 | Permettre à l'utilisateur de sélectionner plusieurs options lui permettant de modifier l'itinéraire calculé |
| FC2 | Enregistrer des préférences utilisateur pour permettre à celui-ci de gagner du temps |
| FC3 | Fournir à l'utilisateur un court historique des trajets avec les 10 derniers trajets effectués |
| FC4 | Informar l'utilisateur sur son impact écologique |
| FC5 | avoir une interface permettant de comprendre les instructions fournies par le gps, même en étant au volant |
| FC6 | le logiciel doit être accessible sur diverses plateforme |

1.2.2 Liste des critères liés aux fonctions du système GPS-like

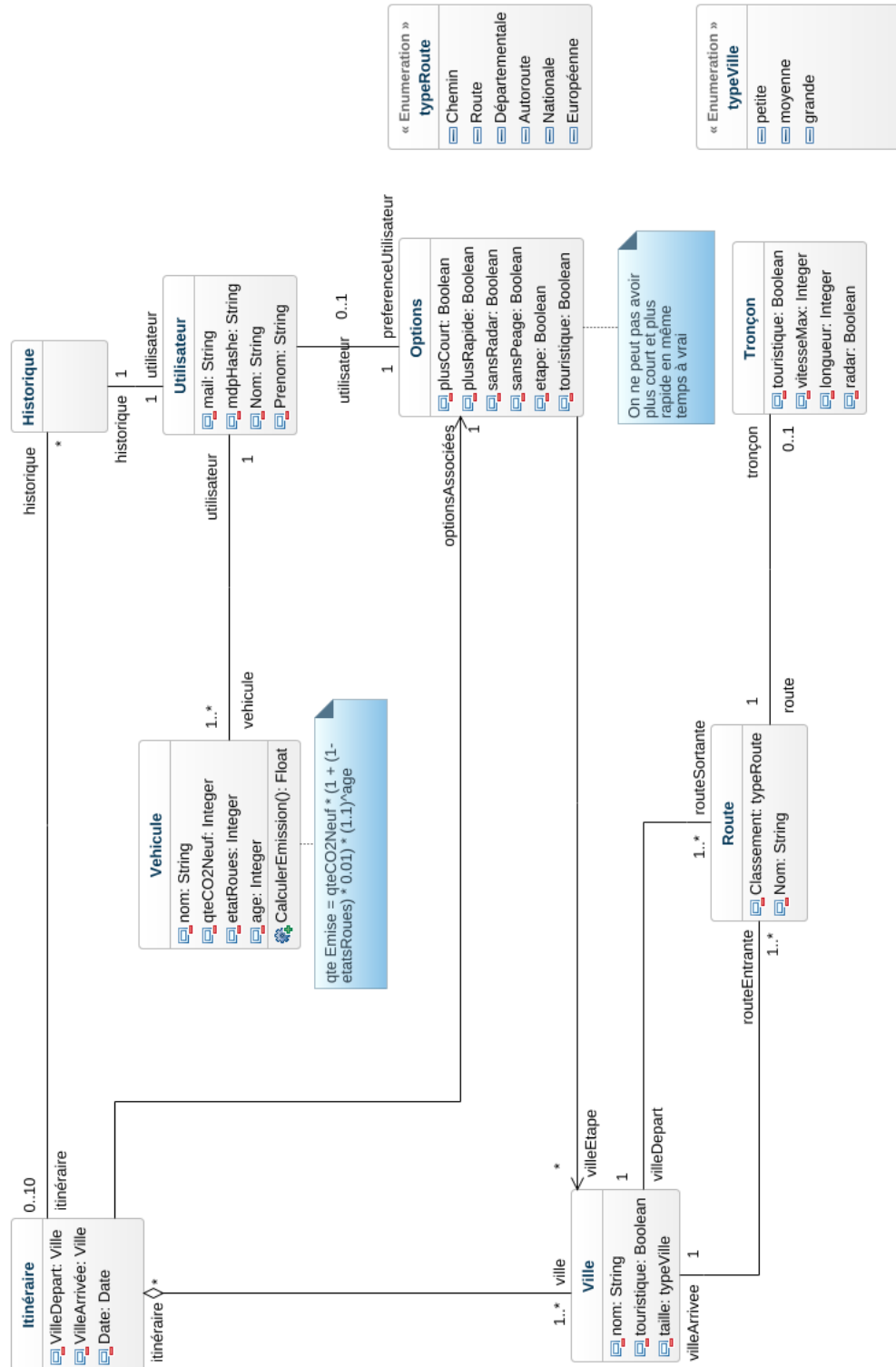
| Fonction | Critères | Niveau | Flexibilité |
|----------|---|---|-------------|
| FP1 | Choisir une ville de départ parmi celles proposées | Tous les itinéraires proposés commenceront à la ville sélectionnée | 0 |
| FP1 | Choisir une ville d'arrivée parmi celles proposées | Tous les itinéraires proposés finiront à la ville sélectionnée | 0 |
| FC1 | Sélectionner l'option "plus court itinéraire" permet d'avoir l'itinéraire le plus court possible en distance | On ne peut pas fournir un plus court chemin avec les mêmes données | 0 |
| FC1 | Sélectionner l'option "plus rapide itinéraire" permet d'avoir l'itinéraire le plus rapide possible en ne tenant pas compte des bouchons | On ne peut pas avoir un chemin plus rapide en fournissant les mêmes données | 0 |
| FC1 | Sélectionner l'option "sans péage" permet d'avoir un itinéraire sans péage | Il n'y a aucun péage sur l'itinéraire | 0 |

| Fonction | Critères | Niveau | Flexibilité |
|----------|--|--|-------------|
| FC1 | Sélectionner l'option "sans radar" permet d'avoir un itinéraire sans radar | Il n'y a aucun radar sur l'itinéraire | 0 |
| FC1 | Sélectionner l'option "étape" en précisant le nom d'une ville parmi celles disponibles permet de créer un itinéraire passant par cette ville | L'itinéraire passe par la ville sélectionnée | 0 |
| FC1 | Sélectionner l'option "tourisme" permet d'avoir l'itinéraire passant par le plus de points touristiques | Il n'y a pas de chemin avec plus de points touristiques | 0 |
| FC2 | Lorsque l'utilisateur a sélectionné ses préférences, celles-ci sont automatiquement pré-sélectionnées lorsque l'utilisateur crée un itinéraire | Les options ont la valeur des préférences enregistrées par l'utilisateur | 0 |
| FC3 | Lorsqu'on finit un itinéraire, celui-ci est enregistré dans un historique | L'itinéraire une fois fini est présent dans l'historique | 0 |
| FC3 | S'il y a plus de 10 itinéraires dans l'historique, le plus ancien est supprimé | Pas plus de 10 itinéraires enregistrés | 0 |

| Fonction | Critères | Niveau | Flexibilité |
|----------|---|--|-------------|
| FC3 | S'il y a plus de 10 itinéraires dans l'historique, le plus vieux est supprimé | Les 10 itinéraires restants sont les plus récents | 0 |
| FC3 | Seul l'utilisateur aura accès à son historique | L'unique utilisateur ayant accès à un historique sera celui l'ayant généré | 0 |
| FC4 | Calculer l'empreinte écologique du trajet en fonction du nombre de kilomètres parcourus et de la quantité de CO2 rejetée aux 100 kilomètres | La valeur calculée est égale à la distance parcourue multipliée par la quantité de CO2 rejetée aux 100km, le tout divisé par 100 | 0,01 |
| FC4 | Permettre à l'utilisateur de rentrer une valeur de CO2 rejeté aux 100 kilomètres | La valeur de CO2 rejeté aux 100km est égale à la valeur rentrée par l'utilisateur, ou égale à une valeur moyenne par défaut | 0 |
| FC5 | L'interface ne devra comprendre au maximum que 5 boutons distincts et lisibles | Le nombre de boutons à l'écran ne peut pas dépasser 5 et est supérieur ou égal à 0 | 0 |
| FC6 | L'utilisateur doit être capable d'utiliser ce GPS sur n'importe quel mobile | L'application est utilisable sur n'importe quel support ayant une connexion Internet | 0 |

2 Analyse

2.1 Diagramme de classe



2.1.1 Utilisateur

Pour s'identifier, un utilisateur utilise un couple e-mail et mot de passe. Cela permettra d'avoir directement accès à un mail pour réinitialiser le mot de passe par exemple. Afin de personnaliser un minimum le service pour un utilisateur, on lui demande aussi son nom et son prénom.

L'utilisateur possède un historique qui se résume à une collection d'itinéraires et des fonctions pour gérer cet historique. Un utilisateur n'a accès qu'à son unique historique.

Il peut aussi indiquer au moins un véhicule pour les différents trajets qu'il réalise. Par défaut, si l'utilisateur ne précise pas son véhicule on lui en fournit un possédant des propriétés par défaut.

Un utilisateur peut aussi enregistrer un ensemble de préférences qui sera sélectionné par défaut lors de la création d'un itinéraire : les cases seront pré-cochées lors de la sélection d'option. L'utilisateur ne peut enregistrer qu'un seul ensemble de préférence au maximum, mais peut s'en passer.

2.1.2 Historique

Comme dit plus haut, un historique est un ensemble d'itinéraire accessible uniquement par un seul utilisateur. On a décidé dans le cahier des charges qu'un historique comporterait au maximum 10 itinéraires, mais celui-ci doit pouvoir être vide au lancement. Dans le cas où l'on réalise un trajet Paris-Brest et qu'un itinéraire est déjà calculé pour Paris-Brest. On supprime l'ancien trajet Paris-Brest de l'historique et on le remplace par le nouveau.

2.1.3 Véhicule

Afin de calculer l'emprunte écologique d'un trajet, l'utilisateur doit rentrer quelques informations par rapport à son véhicule : de quel véhicule il s'agit, son âge, et l'état de ses pneus. La fonction `calculerEmission` permet ensuite de calculer l'émission par kilomètre du véhicule et nous permettra par la suite de calculer la quantité totale émise lors d'un trajet (FC 4 : informer l'utilisateur sur son impact écologique). Si l'utilisateur ne fournit pas d'information, on utilisera des valeurs par défaut pour créer un véhicule test. Le véhicule test

pourra ensuite être supprimé si d'autres informations ont été rentrées. On peut aussi supprimer autant de véhicules qu'on le souhaite tant que leur nombre est supérieur ou égal à 1 après suppression.

2.1.4 Itinéraire

Un itinéraire est une suite de villes étapes. Certaines peuvent être imposées par l'utilisateur depuis les options. On spécifie une ville de départ et d'arrivée, ainsi qu'une date qui sera utile pour la mise à jour de l'historique, en effet on ne conserve que les dix derniers trajets de l'historique, on doit donc pouvoir les trier en fonction de leur date respective.

On choisit de ne garder qu'une collection de ville dans l'itinéraire car on part du principe qu'une route ne relie que deux villes. Et que deux villes ne sont reliées que par une seule et unique route. On peut donc trouver la bonne route en cherchant la ville d'arrivée pour chaque depuis la ville étape

2.1.5 Options

La classe option permet d'influer sur la manière dont sera calculée l'Itinéraire, en fonction des options sélectionnées, certaines routes seront impraticables (par exemple dans le cas où l'utilisateur veut éviter les péages) tandis que d'autres seront préférées lors du calcul de l'itinéraire (dans le cas où on veut passer par des points touristiques par exemple).

On ne peut avoir `plusCourt = true` et `plusRapide = true`

Dans le cas où un objet de la classe Option est un ensemble de préférence pour l'utilisateur, étape doit être à `false` et villeEtape doit être vide : on ne prend pas en compte les étapes dans les préférences.

2.1.6 Ville

Une ville est représentée par un nom et sa taille, indiquée par son type : petite moyenne ou grande. Elle est reliée à un ensemble de routes qui lui permettent d'être connectée à d'autres villes. Elle sert d'étape dans les itinéraires.

2.1.7 Route

Les routes relient les villes entre elles : elle sont reliées à une route de départ et une route d'arrivée. On trouve donc pour chaque ville une route Ville1 - Ville2 et vice versa. Pour le moment ceci n'est pas utile, mais si on veut mettre à jour l'ensemble de la base de données concernant les routes dans le futur, il est intéressant de pouvoir modifier les vitesses des tronçons composant cette route uniquement dans un sens. Il existe par ailleurs des endroits en France où une limite de vitesse est appliquée dans un sens, et où une autre limite est appliquée dans l'autre sens.

2.1.8 Tronçon

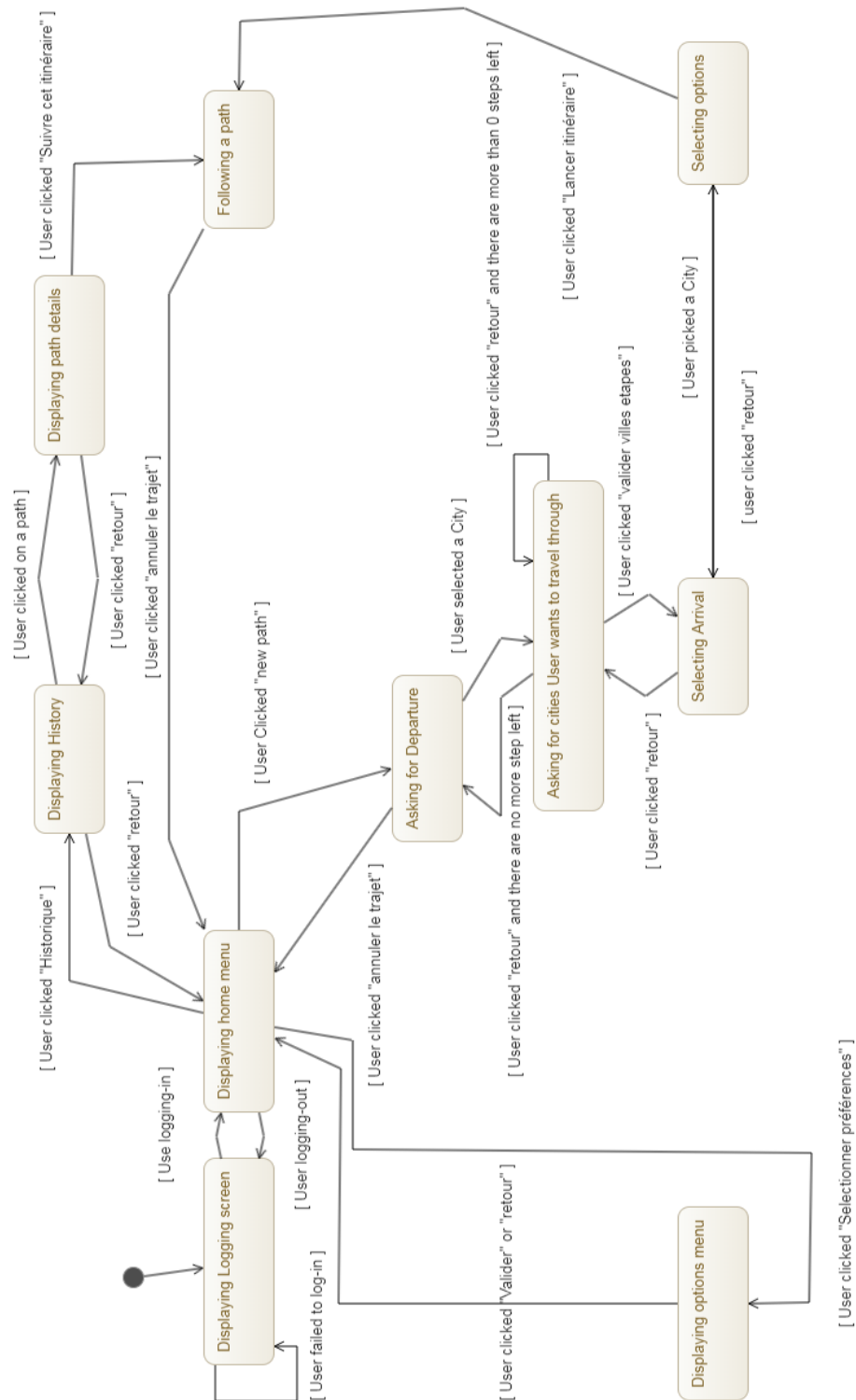
Les tronçons sont les différents éléments qui composent une route. Ils peuvent être de différents types : autoroutier ou national entre autre. Un tronçon ne compose qu'une seule route. On peut sur un tronçon rencontrer des points touristiques. De plus c'est des tronçon qu'on obtient l'information de la présence des radars ainsi que de la longueur en kilomètres et de la vitesse amximum autorisée. C'est pour cette raison que la classe route possède 3 fonctions permettant de faire remonter ces informations : `calculerLongueur()`, `calculerNbRadar()` et `calculerNombreSiteTouristiques()`.

2.1.9 Énumérations : `typeVille` et `typeRoute`

On a choisit d'utiliser des énumérations pour les types de tronçon ainsi que les types de ville.

2.2 Diagramme Etat-Transition

Pour simplifier la compréhension des lecteurs, nous avons réalisé un diagramme Etat-Transition permettant d'expliquer comment naviguer dans l'application GPS-LIKE.

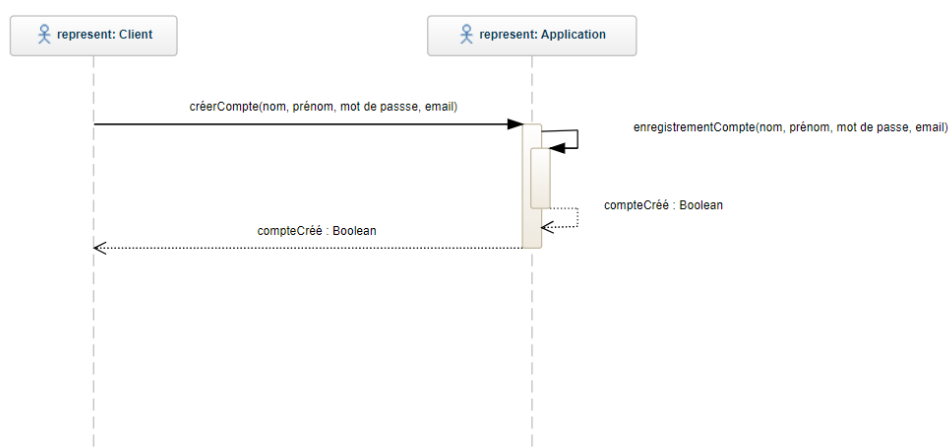


2.3 Diagrammes de séquence

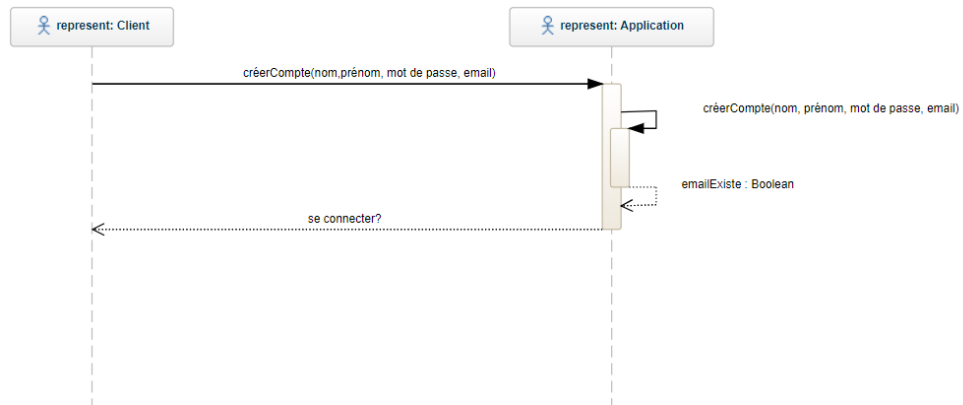
2.3.1 Inscription

| | |
|-----------------|---|
| Cas principal | L'utilisateur indique un nom, un prénom, un mot de passe et un email valide. |
| Cas alternatif | L'utilisateur indique un email déjà utilisé, il est re-dirigé sur le formulaire de connexion. L'utilisateur entre des noms, prénoms, et/ou emails invalides et/ou ne respectant pas la taille requise. |
| Cas d'exception | |
| Remarques | L'utilisateur doit utiliser un email unique, les noms, prénoms peuvent être présents en autant d'exemplaires que nécessaire. |

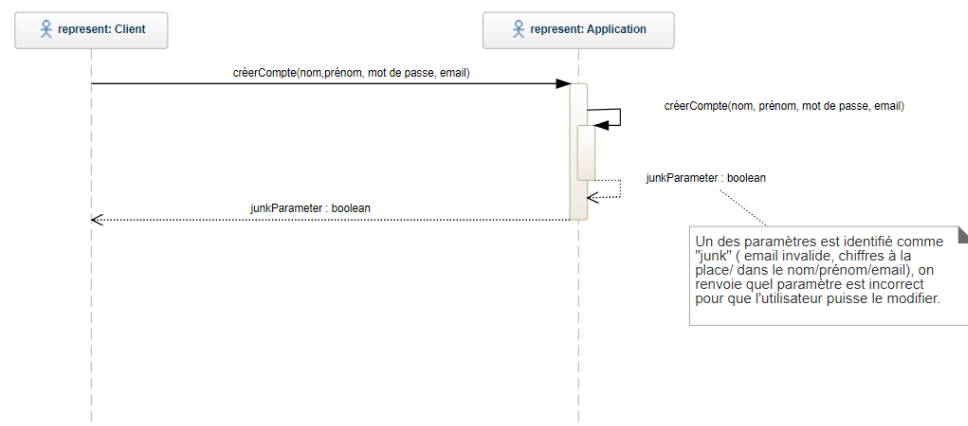
2.3.1.1 Cas principal



2.3.1.2 Cas alternatif : email déjà utilisé



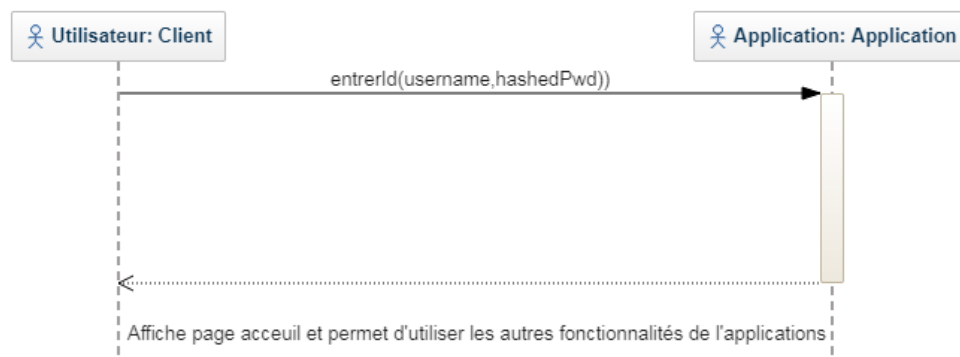
2.3.1.3 Cas alternatif : identifiants invalides



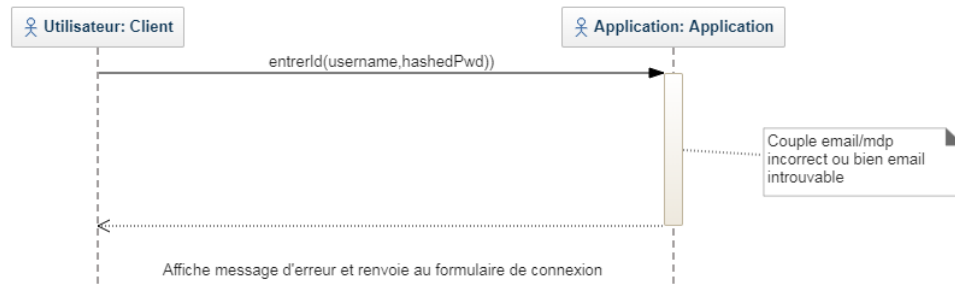
2.3.2 Connexion

| | |
|-----------------|---|
| Cas principal | L'utilisateur se connecte en entrant un mail et un mot de passe valides. |
| Cas alternatif | L'utilisateur entre un email ou un mot de passe invalide, son erreur lui est signalée et on lui propose de s'inscrire ou de recevoir un nouveau mot de passe. |
| Cas d'exception | |
| Remarques | |

2.3.2.1 Cas principal



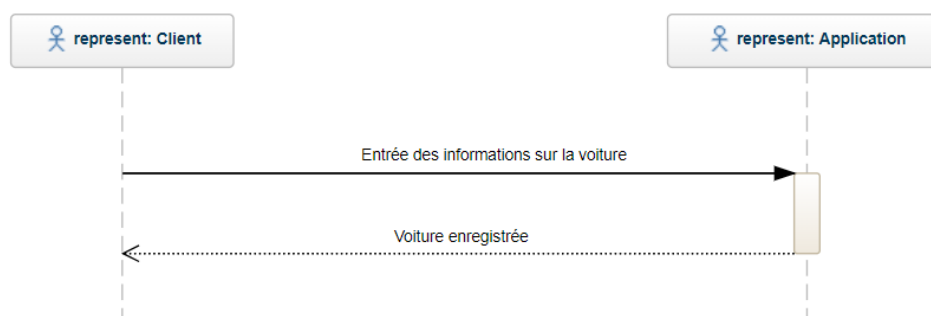
2.3.2.2 Cas alternatif



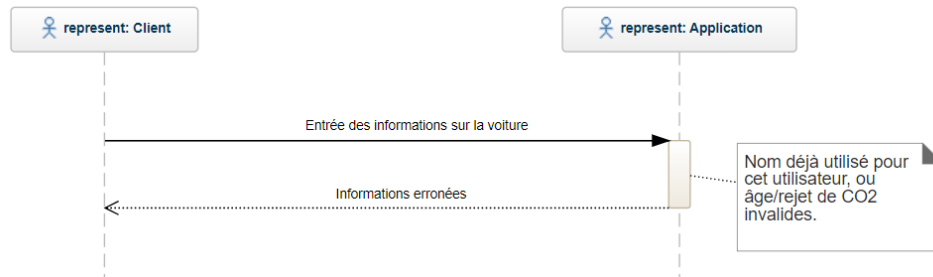
2.3.3 Enregistrer un véhicule

| | |
|-----------------|---|
| Cas principal | L'utilisateur enregistre un nouveau véhicule avec un nom, une valeur de rejet au km (trouvable dans le livret d'information du véhicule), l'état des pneus et l'âge du véhicule) |
| Cas alternatif | Enregistrement du véhicule impossible : l'utilisateur entre le même nom qu'un autre véhicule qu'il a déjà enregistré ou l'utilisateur entre des informations non valides pour la valeur de rejet ou l'âge du véhicule |
| Cas d'exception | |
| Remarques | L'âge de la voiture doit être compris entre 0 et 100 ans (éventuelles voitures de collection) et la valeur de rejet entre 0 g/km (voitures électriques) et 400 g/km (la voiture la plus polluante connue actuellement en rejette 397 g/km). |

2.3.3.1 Cas principal



2.3.3.2 Cas alternatif : entrées non valides



2.3.4 Spécifier un trajet

| | |
|-----------------|--|
| Cas principal | L'utilisateur entre une ville de départ, le système lui propose d'entrer une ou plusieurs villes étapes et sa ville d'arrivée, puis de modifier ses préférences s'il le souhaite. |
| Cas alternatif | <p>Si la ville d'arrivée est la même que la ville de départ, et qu'il n'y a aucune ville étape, on considère que l'utilisateur est arrivé.</p> <p>Si la ville de départ ou d'arrivée est la même que l'une des villes étapes, on le signale à l'utilisateur.</p> <p>Si l'utilisateur n'a jamais entré ses préférences, il doit le faire avant de valider son trajet, on le prévient qu'elles seront appliquées par défaut sur ses trajets s'il ne les modifie pas.</p> |
| Cas d'exception | |
| Remarques | Un utilisateur peut très bien décider de partir d'une ville A, faire un trajet entre plusieurs villes étapes, puis revenir à la ville A, particulièrement dans le cadre du tourisme. |

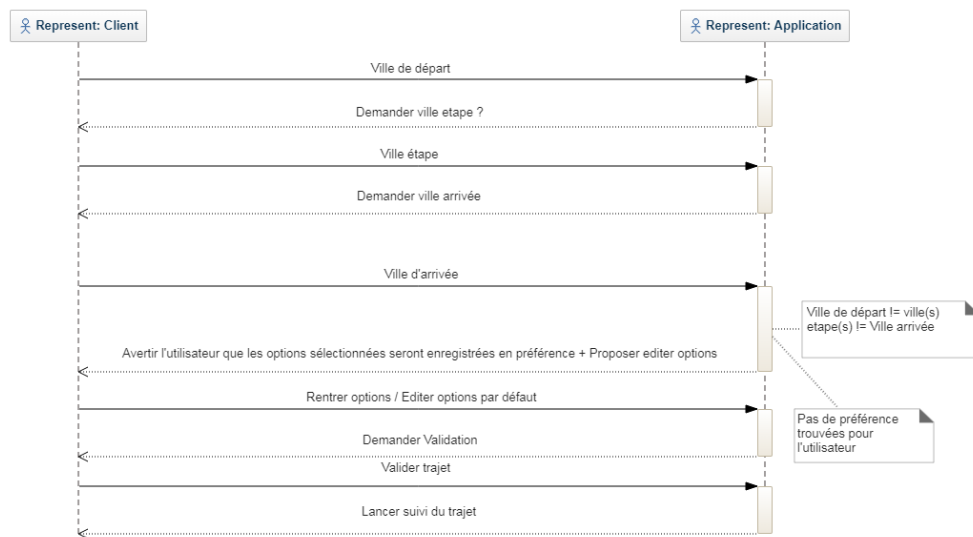
2.3.4.1 Cas principal



2.3.4.2 Cas alternatif : ville de départ = ville d'arrivée sans aucune étape



2.3.4.3 Cas alternatif : pas de préférences enregistrées pour l'utilisateur



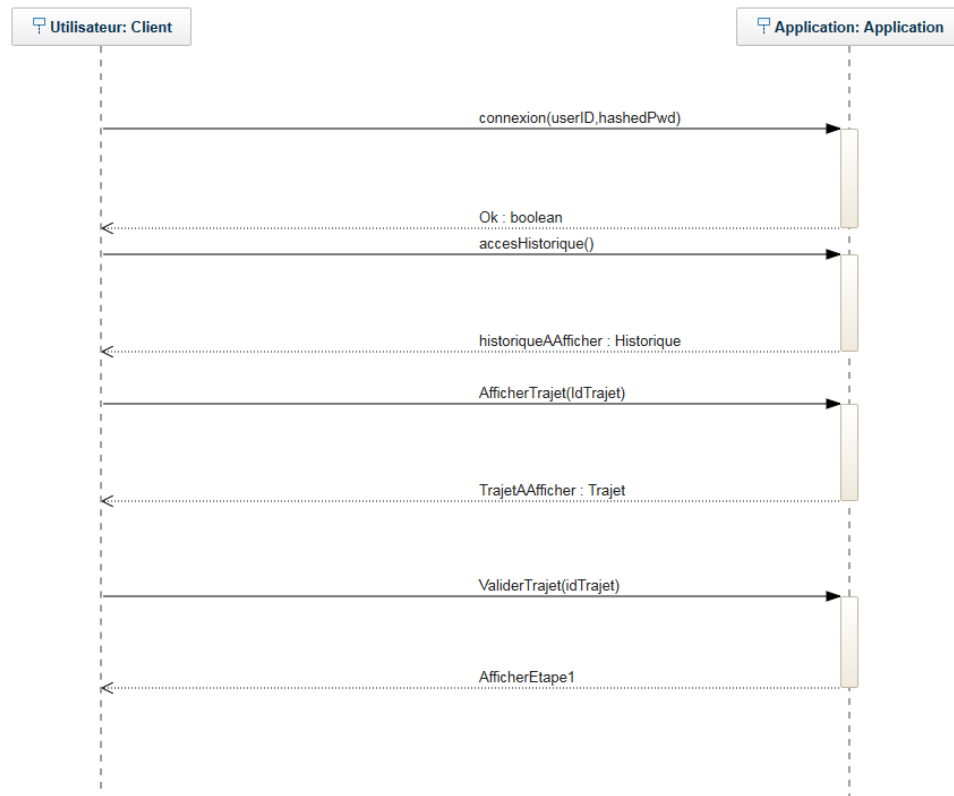
2.3.4.4 Cas alternatif : étape = ville d'arrivée ou étape = ville de départ



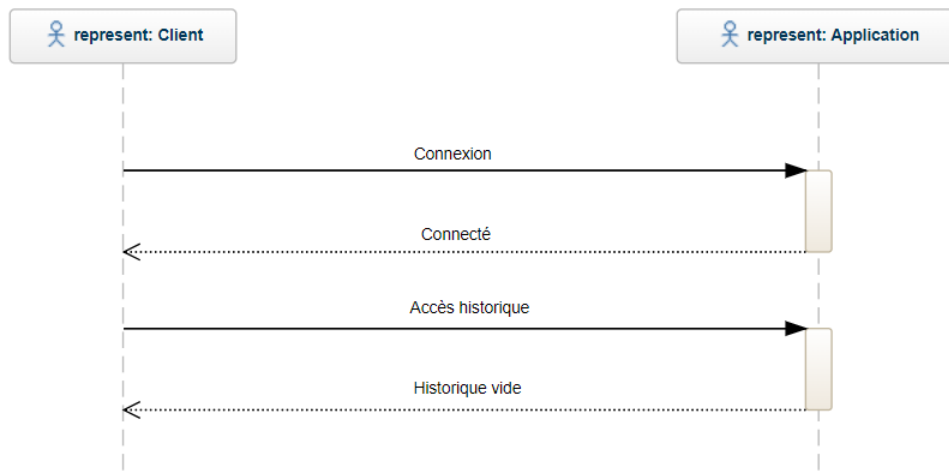
2.3.5 Sélectionner un trajet depuis l'historique

| | |
|-----------------|--|
| Cas principal | L'utilisateur sélectionne un trajet dans son historique, il est considéré comme étant le trajet suivi. |
| Cas alternatif | <p>L'historique est vide et on ne peut pas choisir de trajet.</p> <p>Si l'utilisateur a déjà rentré un trajet ou qu'il suit déjà un trajet, il ne peut pas en sélectionner un autre, à moins d'avoir confirmé d'être arrivé dans la ville de départ du trajet de l'historique.</p> |
| Cas d'exception | |
| Remarques | |

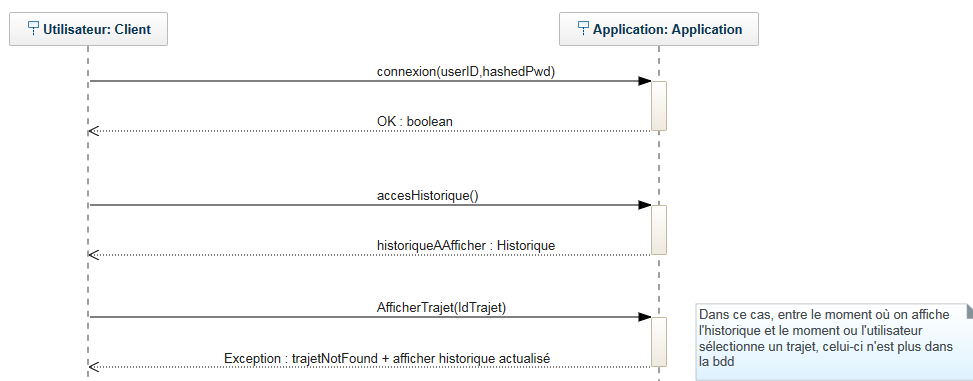
2.3.5.1 Cas principal



2.3.5.2 Cas alternatif : historique vide



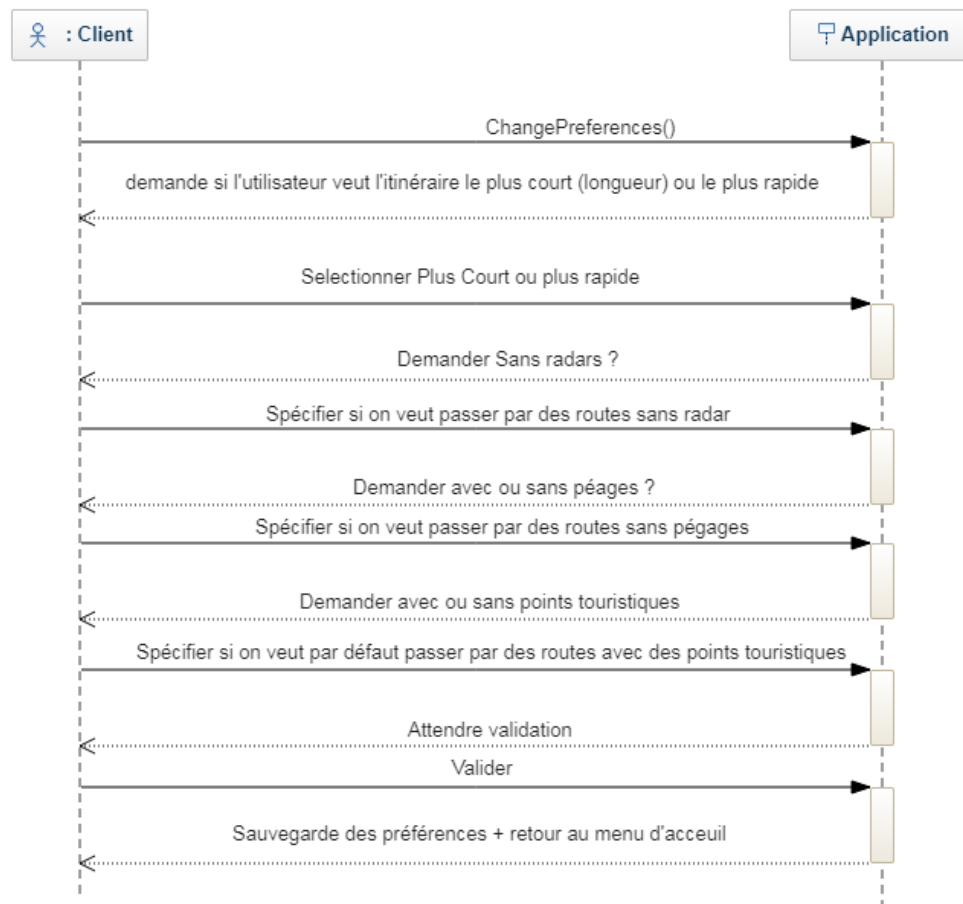
2.3.5.3 Cas alternatif : sélection alors que l'utilisateur suit déjà un trajet



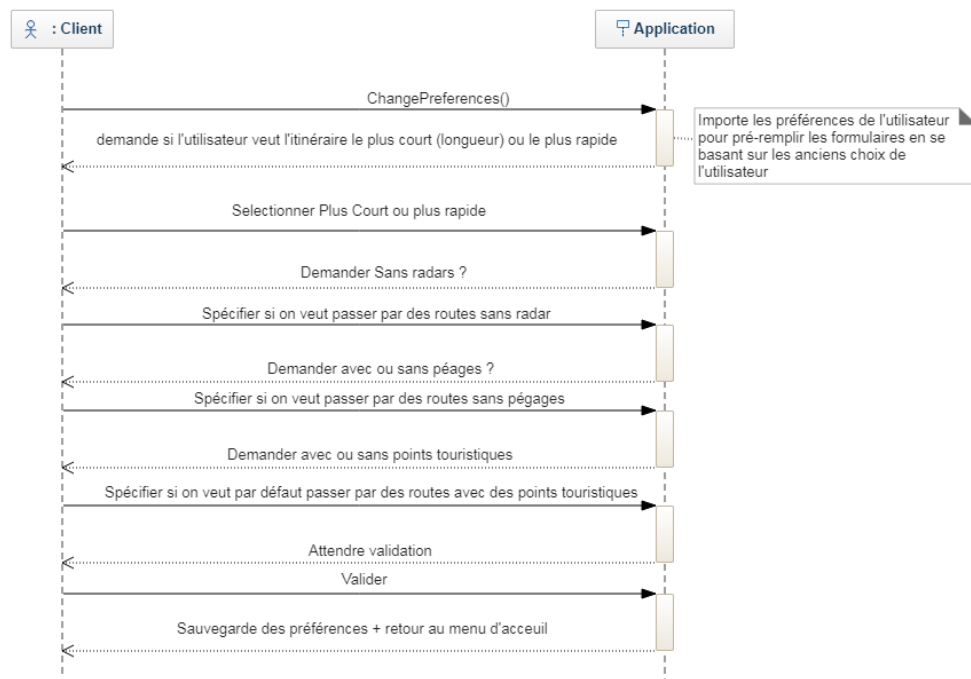
2.3.6 Entrer des préférences

| | |
|-----------------|---|
| Cas principal | L'utilisateur choisit ses préférences, elle deviennent les préférences par défaut |
| Cas alternatif | L'utilisateur a déjà entré des préférences et ne souhaite pas les modifier, elles sont donc récupérées et appliquées : rien ne diffère du cas principal, il y a juste des cases préselectionnées par défaut correspondant aux choix de l'utilisateur. |
| Cas d'exception | |
| Remarques | A chaque spécification de trajet, l'utilisateur a la possibilité de modifier les paramètres par défaut, sélectionnés lors de sa première utilisation de l'application. Ces nouveaux paramètres ne remplacent cependant pas les paramètres par défaut et sont valables uniquement pour le trajet en cours. |

2.3.6.1 Cas principal



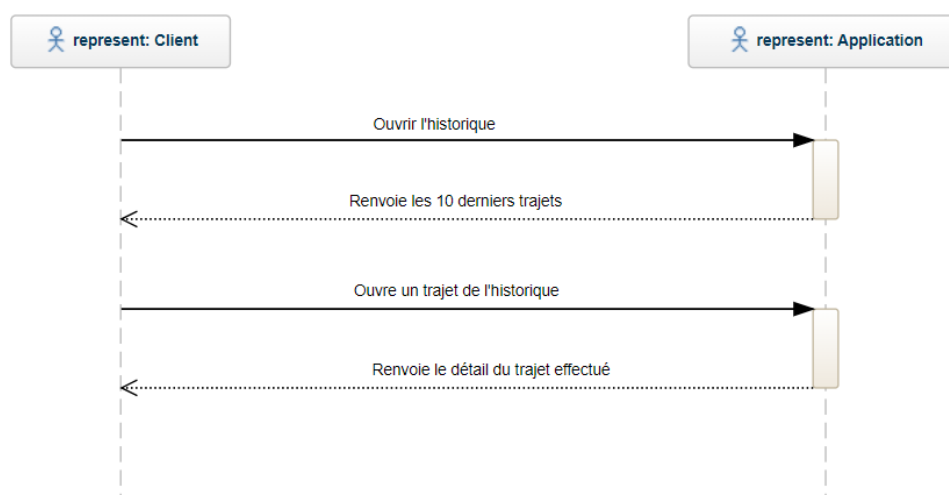
2.3.6.2 Cas alternatif : l'utilisateur ne souhaite pas modifier les préférences



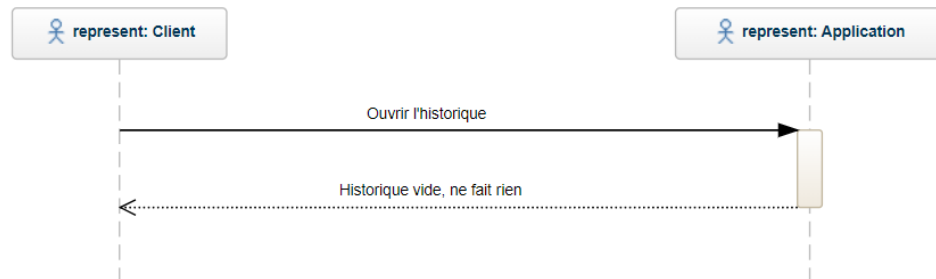
2.3.7 Consulter l'historique

| | |
|-----------------|---|
| Cas principal | L'utilisateur consulte l'historique de ses 10 derniers trajets (ou moins) et peut sélectionner un des trajets pour revoir ses informations en détail (départ, arrivée, étapes, paramètres, km parcourus, consommation). |
| Cas alternatif | Si l'utilisateur n'a pas encore effectué de trajet, l'historique est vide |
| Cas d'exception | |
| Remarques | |

2.3.7.1 Cas principal



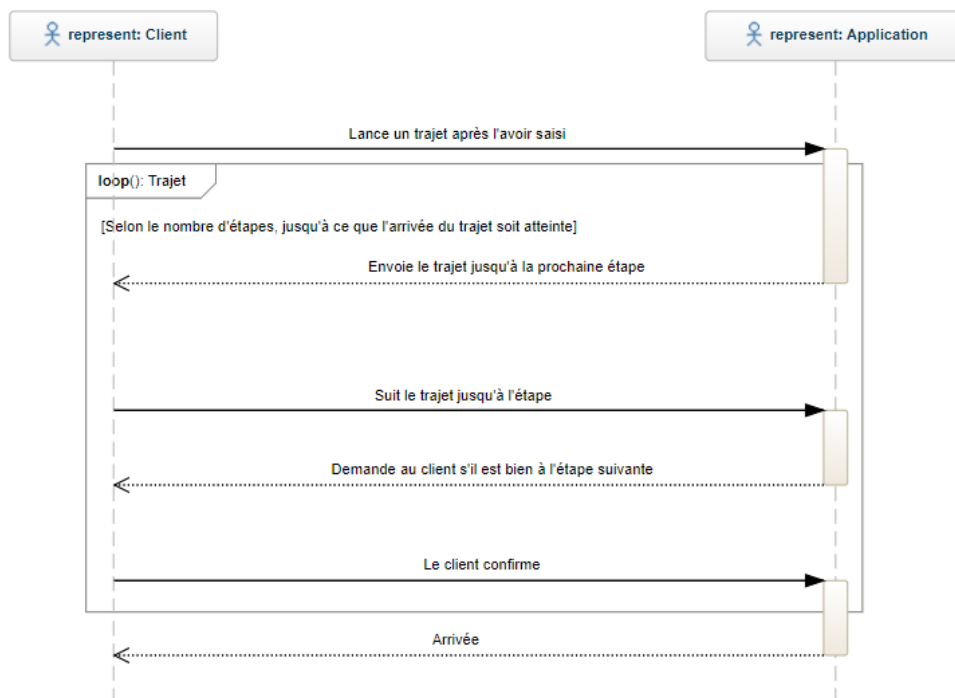
2.3.7.2 Cas alternatif : historique vide



2.3.8 Suivre un trajet

| | |
|-----------------|--|
| Cas principal | L'utilisateur lance un trajet après l'avoir spécifié, l'application lui envoie le chemin à suivre jusqu'à la première étape, l'utilisateur suit le chemin, puis l'application lui demande s'il est bien à la ville étape. L'utilisateur répond oui et l'application lui envoie la suite du trajet ou lui annonce qu'il est arrivé. |
| Cas alternatif | L'utilisateur répond non lorsque l'application lui demande s'il est bien à l'étape indiquée. Il est considéré comme perdu et l'application doit donc recalculer le trajet. |
| Cas d'exception | |
| Remarques | |

2.3.8.1 Cas principal



2.3.8.2 Cas alternatif : le client est perdu

