

PROJET GÉNIE LOGICIEL : KAYAK



Maquaire Auriane

Lejeune Alice

Pican Gaëtan

TABLE DES MATIÈRES

Introduction.....	2
État de l'art.....	3
Interface Graphique.....	4
Diagrammes UML.....	5
Diagramme de cas d'utilisation.....	5
Diagrammes de séquence général.....	6
Diagramme de séquence : relation utilisateur-application.....	7
Diagramme de séquence : relation application- serveur Google.....	9

INTRODUCTION

Plusieurs amis veulent sortir à Paris. Ils n'arrivent pas à trouver des lieux de sortie plaisant à la totalité des membres de ce groupe. Notre programme « Kayak » va choisir un bar, un restaurant ainsi qu'une boîte de nuit respectant les contraintes de tous les membres du groupe avec un temps de trajet le plus faible possible.

Notre programme aura une interface avec des entrées, qui sont une date, une liste de personnes avec leurs adresses, des préférences alimentaires et une sortie qui sera une liste chronologique de trois adresses : un bar, un restaurant et une boîte. Cette liste respectera les horaires de chaque lieu.

Notre projet va répondre aux besoins de l'utilisateur : l'interface sera claire et facile à utiliser, les réponses seront rapides et l'utilisateur devra pouvoir dire ses préférences sur le mode de transport, ses préférences alimentaires, le bon compromis entre ses préférences et le temps de trajet et pourra modifier sa demande si un des lieux ne lui convient pas.



ÉTAT DE L'ART

Il n'existe actuellement pas d'applications semblable à Kayak. Il existe de nombreuses applications pour pouvoir trouver des restaurants correspondants à ce que l'on cherche ou encore pour pouvoir trouver des soirées se déroulant sur Paris mais rien pour faire le lien entre des bars, restaurants et boîtes de nuit.

Voici certaines applications ayant des points communs avec Kayak :

Figaroscope : permet de choisir un restaurant à partir de la géolocalisation, du prix, du quartier et des envies (brunchs, terrasses, burgers,...)

ParisBouge : recense jour par jour les événements qui se passent à Paris et les bonnes adresses telles que les restaurants, les bars,...

Urban Pulse : recense tous les lieux et événements en rapport avec un mot-clé donné par l'utilisateur, puis trouve l'itinéraire en prenant en compte tous les modes de transports (Bus, Métro, RER, Vélib,...)



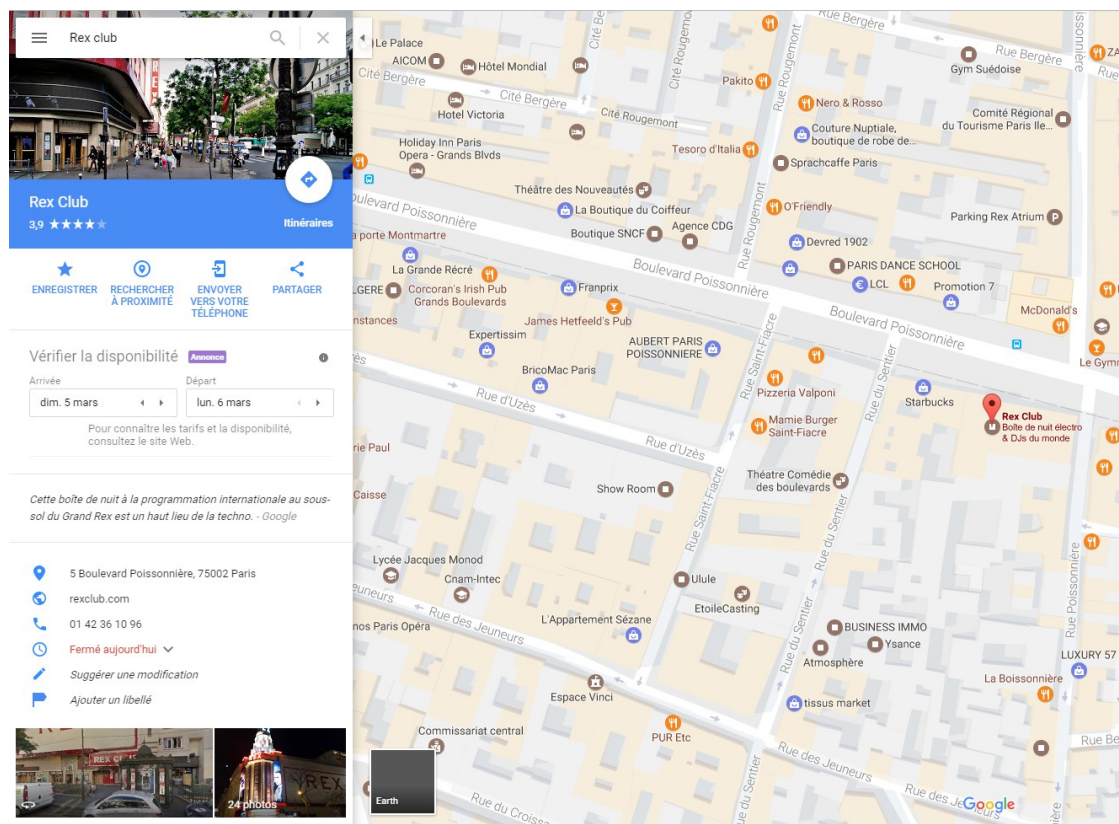
INTERFACE GRAPHIQUE

Le but de notre programme est de donner un lieu correspondant le mieux à un ou plusieurs individus.

L'interface doit être la plus simple possible, il y aura un menu pour effectuer une recherche avec différents éléments à donner : Nombres de personnes, moyens de locomotion, styles de restaurants recherchés, contraintes...

Les options seront facile à accéder : une liste déroulante permet de couvrir un grand nombre d'options, l'utilisateur peut en rajouter s'il le souhaite.

La recherche doit être modifiable si quelque chose ne convient pas. Les résultats permettent d'accéder à la page Google Maps des destinations comme pour l'exemple ci-dessous.



DIAGRAMMES UML

Diagramme de cas d'utilisation

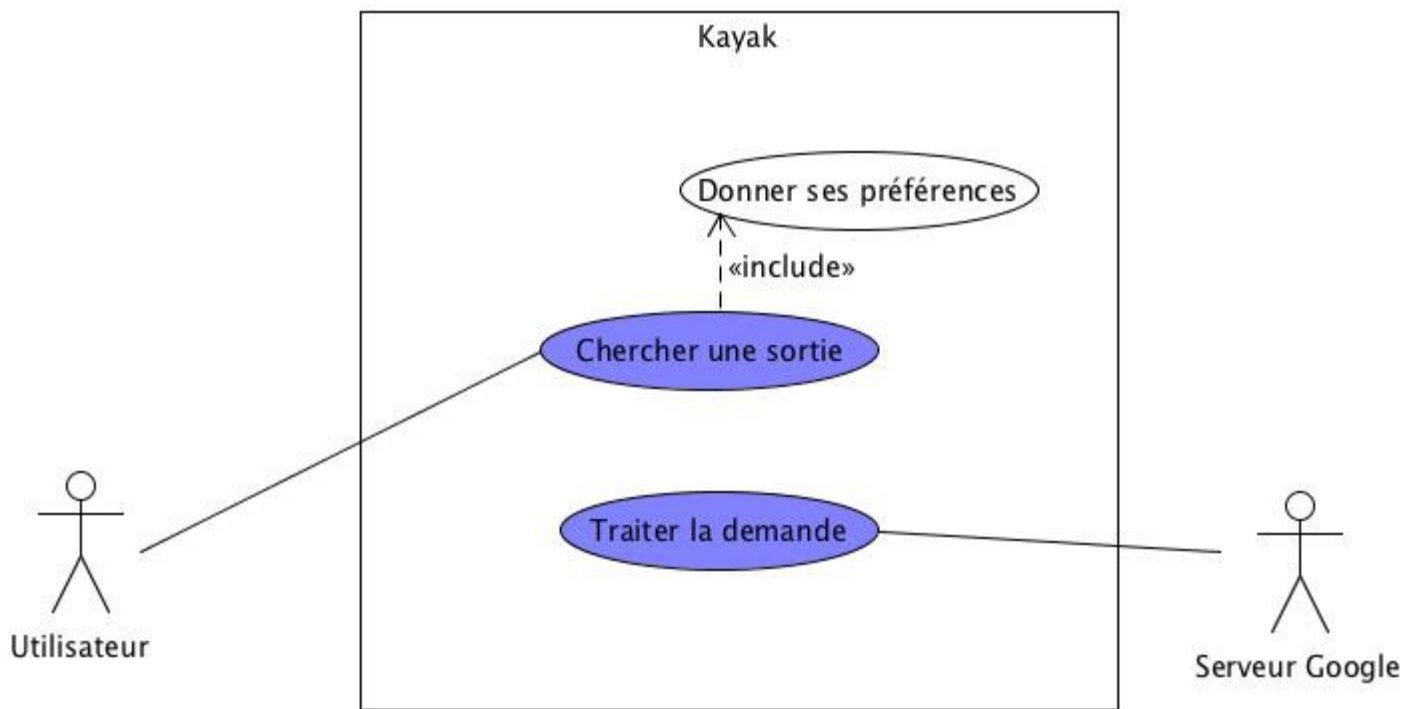


Diagramme de cas d'utilisation

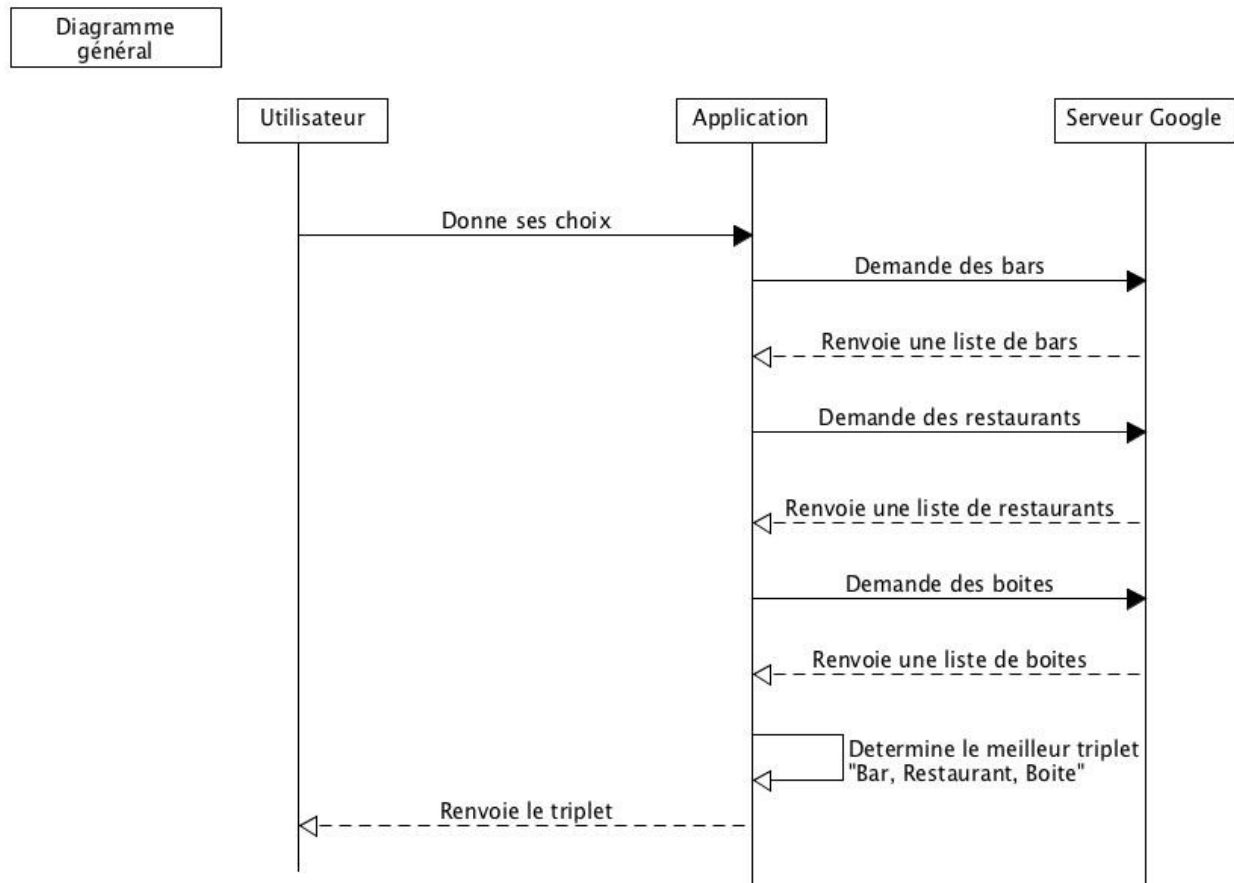
Ce diagramme de cas d'utilisation regroupe deux acteurs : un utilisateur et le serveur Google.

L'utilisateur est celui qui veut organiser sa sortie et qui va réagir avec l'interface de l'application nommée « Kayak ».

Le serveur Google traitera les informations données par l'application afin de trouver le triplet « bar/restaurant/boîte » demandé par l'utilisateur.

Les actions exercées par ces deux acteurs sont très générales et ne sont utiles que pour la compréhension de l'application que nous cherchons à créer. Elles seront explicitées dans les diagrammes de séquences suivants.

DIAGRAMMES DE SÉQUENCE GÉNÉRAL

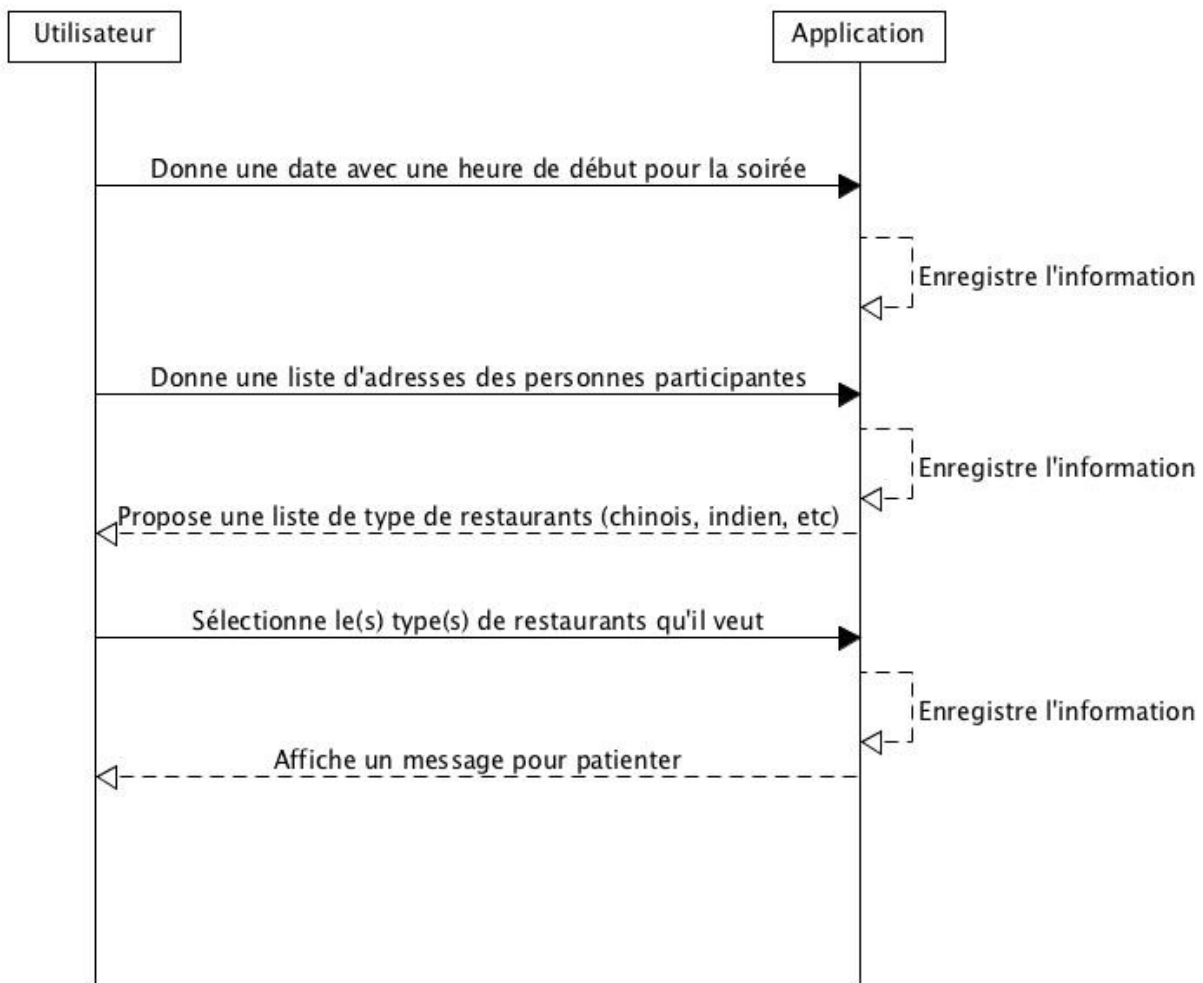


Ce diagramme de séquence représente le cas général d'un scénario entre l'utilisateur, l'application et le serveur Google.

La première flèche « donne ses choix » est détaillée dans le diagramme de séquence 2.

On a décidé de répartir les trois demandes « bar, restaurant, boite » séparément. C'est-à-dire que l'application va demander d'abord une liste de bars au serveur Google, puis de restaurants puis de boites. Elle va ensuite faire le tri et choisir le meilleur triplet pour l'envoyer à l'utilisateur.

On explicite les échanges entre l'application et le serveur Google dans le diagramme de séquence 3.

DIAGRAMME DE SÉQUENCE : RELATION UTILISATEUR-APPLICATION

Dans ce second diagramme de séquence , on représente la façon dont l'utilisateur donne ses choix.

Un utilisateur commence par donner une date et une heure à partir de laquelle il veut débiter sa soirée.

L'application enregistre ces informations.

L'utilisateur donne ensuite les adresses respectives des personnes qui vont l'accompagner pendant la soirée.

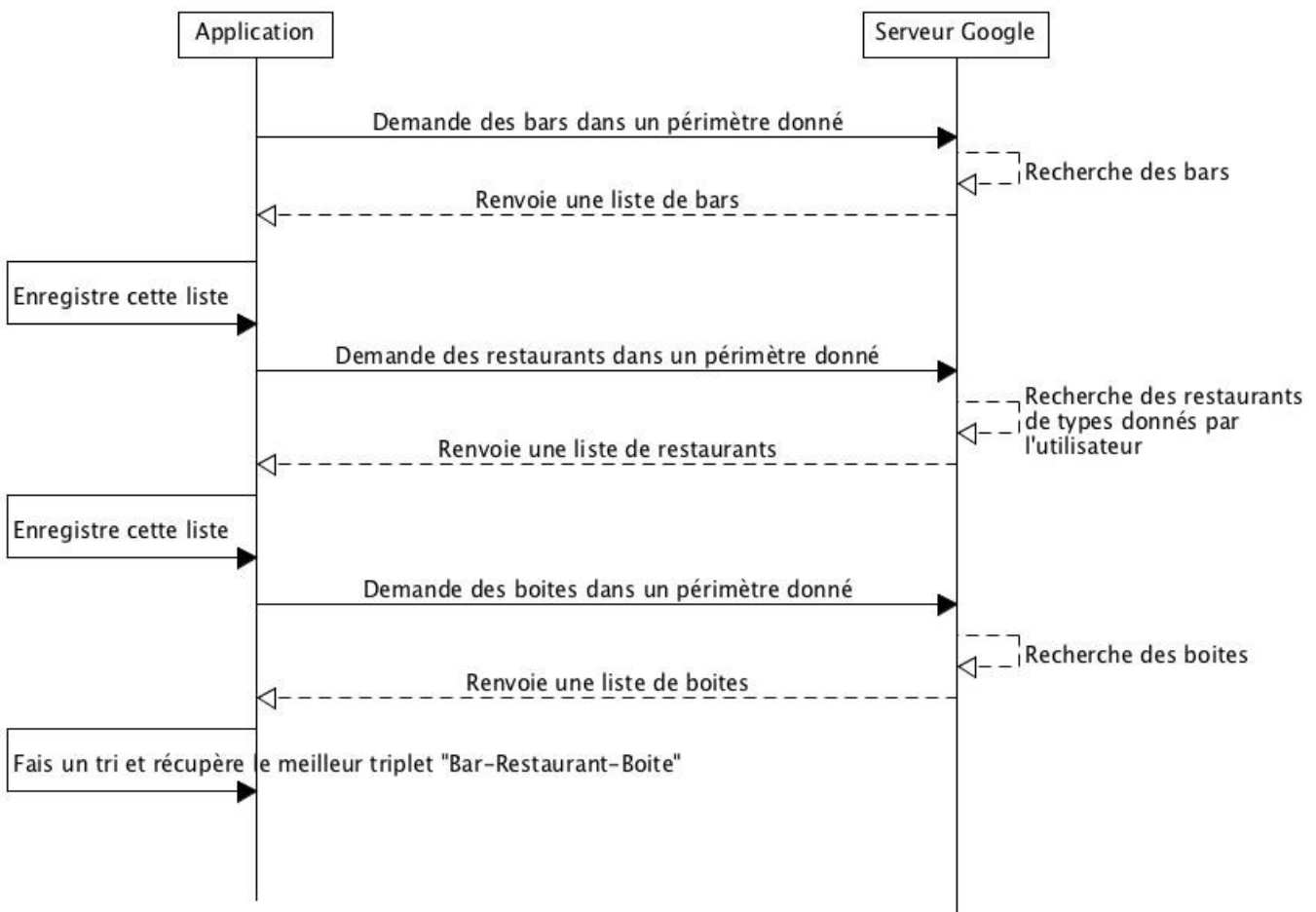
L'application enregistre toutes ces adresses et va choisir une seule localisation (telle qu'une adresse, emplacement, monument, place ou autres) qu'il donnera ensuite au serveur Google afin

que ce dernier puisse chercher le triplet bar/resto/boite dans un périmètre arbitraire donné par l'application.

L'application propose une liste (peut-être dans un menu déroulant, ou une liste de choix à cocher par exemple) de types de restaurants, tels que chinois, indien, mexicain, japonais, etc ...

L'utilisateur sélectionne un ou plusieurs types et envoie la réponse.

L'application les enregistre et envoie un message à l'utilisateur pour patienter le temps qu'il donne ces informations au serveur.

DIAGRAMME DE SÉQUENCE : RELATION APPLICATION- SERVEUR GOOGLE

Ce troisième diagramme de séquence représente la façon dont l'application va demander au serveur Google de rechercher les bars, restaurants et boites. Elle commence par demander des bars dans la localisation qu'elle avait trouvé à l'aide des adresses des personnes présentes à la soirée, et dans un périmètre arbitraire.

Le serveur renvoie une liste qui est ensuite enregistrée.

Le scénario est le même pour la recherche d'un restaurant et d'une boite. Pour les restaurants, l'application fournit les types de restaurants que l'utilisateur a référencé afin de restreindre la recherche du serveur Google.

L'application se retrouve alors avec trois listes, il va choisir un seul bar, restaurant et boite des trois listes selon la distance de chacun pour que les personnes aient le moins de trajet possible. Il renvoie ainsi le triplet Bar/Restaurant/Boite à l'utilisateur (comme représenté dans le diagramme de séquence utilisateur-application).