

Université Paris Sud

Projet Assistant événement de sorties à Paris

Projet Génie Logiciel

Alassana BALDE
Hoang-Phong NGO
Youcef LASLA

Sommaire

- I. Présentation du projet
- II. Contexte et objectifs
- III. Exigences techniques et interface
 - 1. Exigences fonctionnelles
 - 2. Exigences non fonctionnelles
 - 3. Interface
- IV. Fonctionnalités
- V. Expression des besoins et cibles
- VI. Conditions de réalisation
 - 1. Planning prévisionnel
 - 2. Contraintes
- VII. Livrable VIII.
- Conclusion IX.
- Diagrammes

I. Présentation du projet

Plusieurs amis veulent sortir à Paris afin de fêter leurs retrouvailles et ont envie de passer une bonne soirée. Pour cela, ils auraient besoin d'organiser successivement trois sorties.

1. Boire un verre dans un bar sympa
2. Manger au restaurant (attention : un des amis n'aime pas les kebabs)
3. Sortir en boîte de nuit

C'est pourquoi nous décidons de faire ce projet (Assistant évènementiel de sortie à Paris) pour permettre aux gens de bien organiser des sorties grâce à des recommandations personnalisées et qui répondent les mieux à leurs besoins.

II. Contexte et objectifs

On vit dans un monde actuellement où tout est connecté, où les jeunes tellement que leur téléphone leur est indispensable qu'ils ont un mal fou à imaginer que des gens, notamment leurs parents, aient pu s'en passer.

L'utilisation des smartphones n'est bien sûr pas sans risque sur notre éducation, vie privée, du point de vue psychique ou même notre vision du monde.

Mais notons qu'il n'y a également que des côtés négatifs. Parmi les côtés positifs on peut citer les réseaux sociaux (bien qu'ils aient aussi des côtés négatifs) donc pouvoir échanger avec ses proches, amis, faire passer le temps, pouvoir connaître le chemin d'un point A à un point B.

Ce projet consistera donc à proposer une plateforme (application ou site web) qui sera capable de faire des recommandations locales personnalisées à l'utilisateur, offrir donc aux utilisateurs mobiles une expérience de recherche efficace des lieux autour de soi.

Les utilisateurs visés sont majoritairement des jeunes, qui cherchent un restaurant, un bar et une boîte de nuit pour débiter ou finir la soirée. Outre les jeunes, les adultes peuvent également être concernés.

Il existe de nombreuses applications de sortie mais on en citera que 4, parmi elles on peut citer iNight Paris qui est plutôt complète et permet de dénicher les endroits qui bougent dans la nuit parisienne, elle recense l'ensemble des lieux ouverts sur Paris après 2h du matin. Outre la géolocalisation, elle contient les fonctions suivantes : envoyer un email avec un commentaire dans la page d'un lieu, les coordonnées du lieu sont automatiquement incluses dans le message et l'email envoyé aux amis de l'utilisateur.

Il y a aussi Dismoioù, qui est le premier site web de recommandations locales personnalisées, ce dernier connaît ses utilisateurs grâce au développement d'un algorithme de recommandation personnalisées. Une fois que le système a compris les goûts d'un utilisateur, il va recommander d'autres lieux qui lui plairont.

Urban Pulse qui pense aux provinciaux en couvrant plusieurs villes comme Lyon ou Rennes. Une application smartphone intégrant l'ensemble des informations et services utiles en temps réel tout ce qui se passe autour de vous. Et elle vous y emmène facilement par les moyens de transport les plus rapides.

Enfin Chauffeur-Privé pour aller de bar en bar ou rentrer de soirée.

III. Exigences techniques et interface

1. Exigences fonctionnelles

Les exigences fonctionnelles sont dans l'ordre :

1. Saisie de date, d'une liste de personne ayant des contraintes d'adresses et de préférences alimentaires.
2. Connaître la position actuelle de l'utilisateur.
3. Tenir compte du temps que prend une sortie, de la distance entre deux points, des horaires d'ouverture coïncidant avec les critères précédents.
4. Récapitulatif des préférences.
5. Bouton de validation.
6. Affichage d'une liste chronologique de trois adresses.

2. Exigences non fonctionnelles

Il existe également des exigences n'ayant pas trait à la fonction du produit, mais étant importantes à prendre en compte à la conception :

1. Facilité d'utilisation : pas besoin d'être programmeur pour s'en servir
2. Délai de réponse
3. L'esthétique

3. Interface

Page web par interaction avec Java :

Utilisation d'objets écrit en java qui seront attachés aux document(s) web pour former un pont entre les deux environnements et permettant ainsi le dialogue.

IV. Fonctionnalités

La fonctionnalité majeure est la recherche du lieu le plus proche.

Le système doit être capable d'exécuter les fonctionnalités suivantes :

- Une navigation par géolocalisation et un itinéraire proposé sur une carte maps (Google Maps).
- Une recherche par type de préférences de sorties, lieux et heure à laquelle la personne doit se rendre en incluant son mode de transport.
- Calcul de l'itinéraire en fonction du lieu et de l'heure ou la personne se trouve et d'où elle doit se rendre.
- Des fiches complètes avec toutes les informations sur les lieux et événements : présentation, infos pratiques, photos et avis des internautes (Flux Google).
- Possibilité de noter et d'écrire un avis détaillé sur les lieux et événements.

Pour cela, il faut :

Proposer à l'utilisateur de saisir sa position actuelle, de saisir le nom de l'endroit auquel il veut se rendre, si c'est un restaurant par exemple lui demander de saisir ses préférences alimentaires afin de lui suggérer les endroits qui lui sont les plus proches mais aussi qui correspondent à son budget. Si l'endroit est inaccessible à pied lui proposer le moyen de déplacement le plus approprié, par exemple le Velib, taxi, métro etc.

Tenir compte de l'horaire d'ouverture des lieux pour ne pas suggérer un endroit qui est déjà fermé.

Les adresses des triplets qui lui seront proposées sont accessible à pied, à vélo ou en voiture, le choix sera fait par l'utilisateur lui-même selon les conditions dans lesquelles il se trouve.

V. Expression des besoins et cibles

Un groupe d'amis souhaite passer une belle sortie où se réunir autour d'un verre, aller manger ensuite puis finir la soirée finalement dans une boîte de nuit, ce sont les critères qui construisent le plan de cette soirée. Prendre la carte de commerce du coin n'est pas une bonne idée puisqu'il est possible de mettre en place une plateforme pouvant faciliter les tâches et donc plus pratique à utiliser.

Le temps est une variable importante dans la vie des Français. Ils sont organisés et aiment arriver à l'heure à un rendez-vous ; **Qui aime attendre plusieurs heures sans rien avoir à faire ? Personne.**

1. Besoins fonctionnels

Donner :

- L'adresse d'un bar
- L'adresse d'un restaurant
- L'adresse d'une boîte de nuit

Tout cela en respectant le choix alimentaire et la date.

2. Besoins non fonctionnels

- **Rapidité** : Le système devra prendre peu de temps pour calculer le triplet en fonction des paramètres d'entrés.
- **Fiabilité** : Garantir le bon triplet dès la première recherche, tout en respectant les contraintes liées à l'adresse et à la date.
- **Economique** : En termes de consommation de batterie

3. Besoins de l'utilisateur

L'utilisateur a juste envie de passer une belle sortie avec ses amis. Il n'a donc pas une idée exacte des endroits où il veut se rendre mais ce dont il est sûr c'est ses préférences notamment alimentaires.

Il a donc besoin d'un triplet d'adresses respectives (bar, resto, boîte de nuit) qui correspond le mieux à ses attentes. Un bar sympa où il peut prendre un verre avec ses amis dans un atmosphère conviviale, un resto qui convient aux préférences de chacun et enfin une boîte de nuit où s'amuser.

4. Besoins de l'application

Nécessite de connaître la date de sortie et une heure approximative pour pouvoir déterminer les lieux en fonction des contraintes et de leurs horaires d'ouverture, connaître l'emplacement actuelle de l'utilisateur afin de sélectionner les adresses les plus proches, connaître les préférences alimentaires de chacun en cas d'abstention de consommation de certaines nourritures afin de faire le bon choix du resto qui leur convient. Ces paramètres pris en entrées déterminent le triplet le plus adéquat.

VI. Conditions de réalisation

1. Planning prévisionnel

La prochaine étape après le cahier des charges on procédera à l'élaboration d'un cahier de conception, puis de l'implémentation et enfin à la soutenance du projet.

Tâches	Dates
Cahier de charges et analyse	Avant le 19/02/2017
Cahier de conception	Avant le 26/03/2017
Implémentation	Mai 2017
Soutenance	Mai 2017

2. Contraintes

Voici les contraintes que l'on pourrait trouver dans ce sujet :

- Distance : Un utilisateur peut se situer trop loin de l'adresse la plus proche, il n'a donc pas d'autres choix que de s'y rendre, car même en cas de refus de l'adresse qui lui ai proposée rien n'est garanti qu'il aura une autre plus proche que l'ancienne.
- Problème de connexion ou de batterie : L'utilisateur peut avoir des problèmes de connexion qui peut donc impacter le délai de réponse du système, ou son appareil peut être déchargé lors de l'utilisation du système, ce qui peut se produire à tout moment et donc ne peut pas être gerer par le système.

- Horaire : Il peut arriver que le triplet qui lui ai proposé présente une adresse dont il ne peut pas se rendre soit parce que la distance qui sépare les lieux lui empêche de se rendre à cette adresse avant sa fermeture ou soit parce qu'il a passé plus de temps que prévu à aux adresses précédentes.
- Météo : Il peut arriver que les conditions météorologiques soient défavorables au déplacement.
- Budget : Certains lieux peuvent nécessiter plus de moyens financiers que prévu par le(s) utilisateur(s).
- Temps : Mettre en place toutes les fonctionnalités du système nécessitent un certain nombre de temps.
- Géographie : Certains lieux sont plus couverts que d'autres, l'emplacement de l'utilisateur joue donc une très grande responsabilité dans les propositions qui lui sont faites.

VII. Livrable

A la fin de la phase de conception du projet, il y aura :

- Un mini site web contenant les champs de saisie où l'utilisateur saisira les informations qui seront traitées pour soumettre ensuite les informations dont il a besoin.
- Réunion de suivi de projet
- Pas besoin d'un manuel d'utilisation car très pratique à manipuler.
- Compte rendu de tests

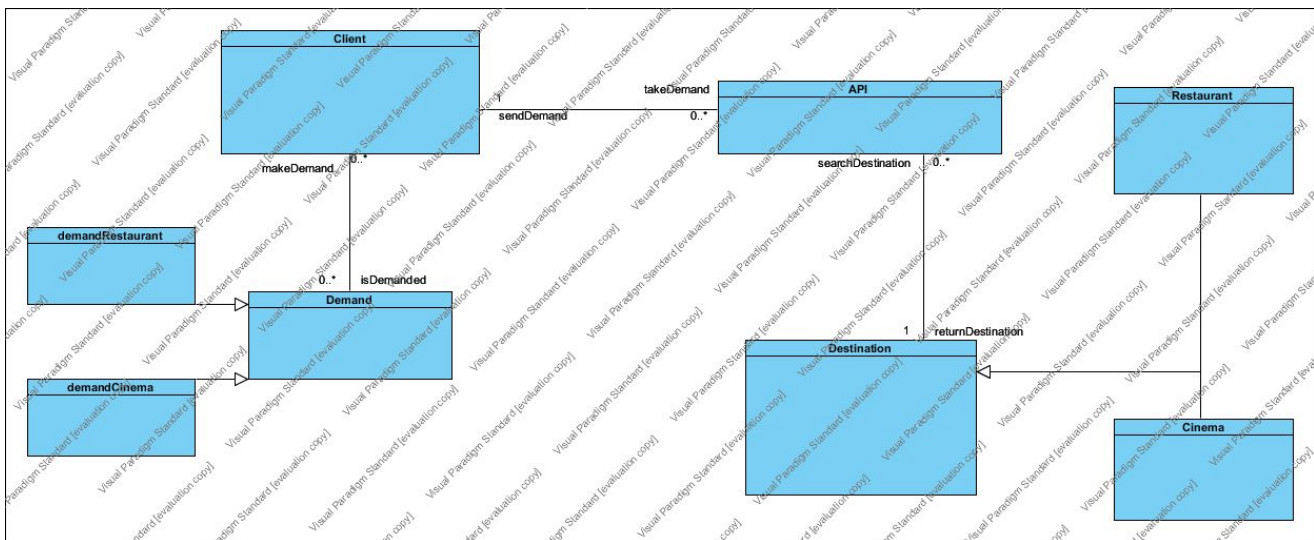
VIII. Conclusion

Mettre en place les besoins de l'utilisateur après les contraintes techniques et/ou organisationnelles, nécessite une grande suivie de démarche permettant de remonter au besoin, finalité du produit et ce dès le départ : un produit n'a de sens que s'il satisfait le besoin de l'utilisation. C'est la raison pour laquelle nous avons décidé d'avoir le minimum de fonctionnalités possibles, mais qui répondent au mieux les besoins de l'utilisateur plutôt que d'avoir plus de fonctionnalités qui lui sont pas forcément nécessaires ou indispensables, voire même dont leur mise en

œuvre n'est pas totalement garantie, soit par manque de temps ou par manque de possibilité.

LES DIAGRAMMES UML

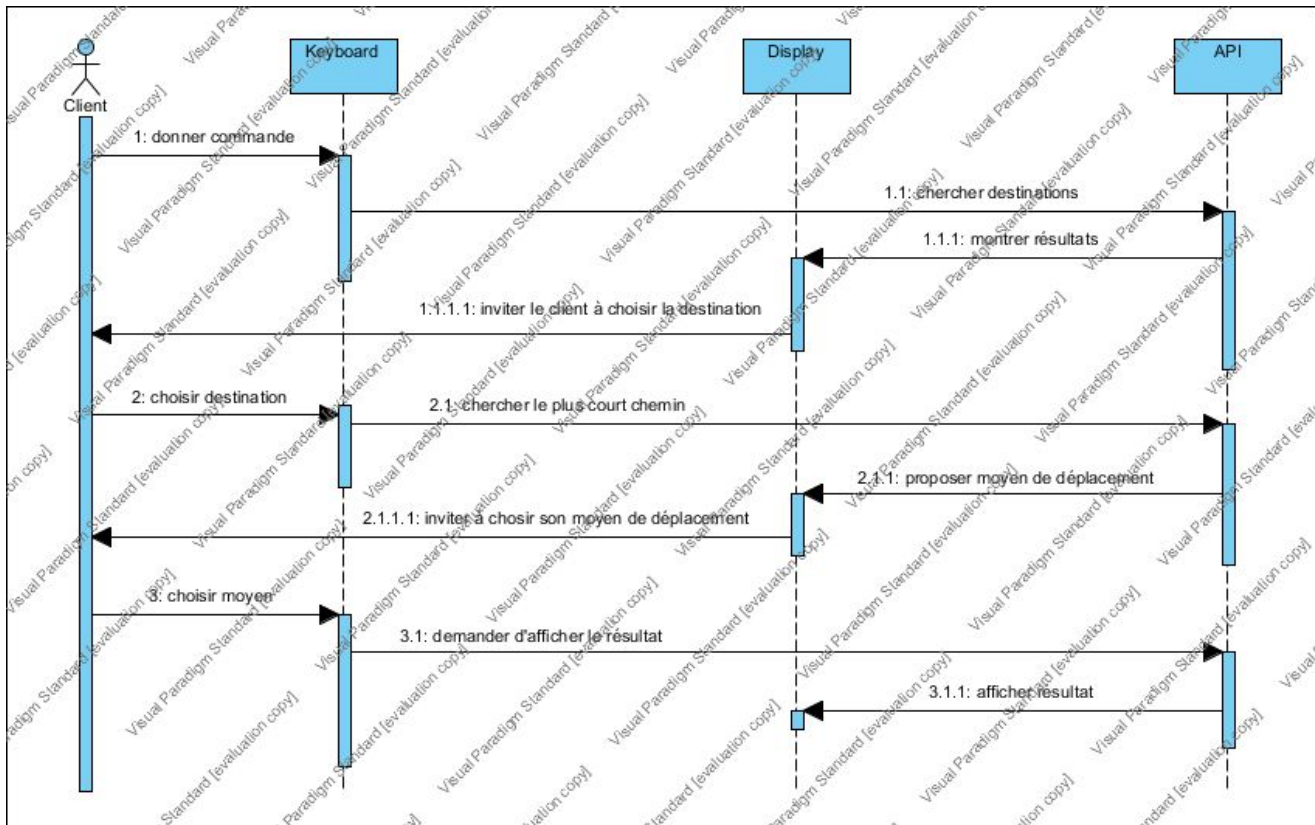
I. Le diagramme de classe



Pour ce type de diagramme, on utilise 4 classes principales:

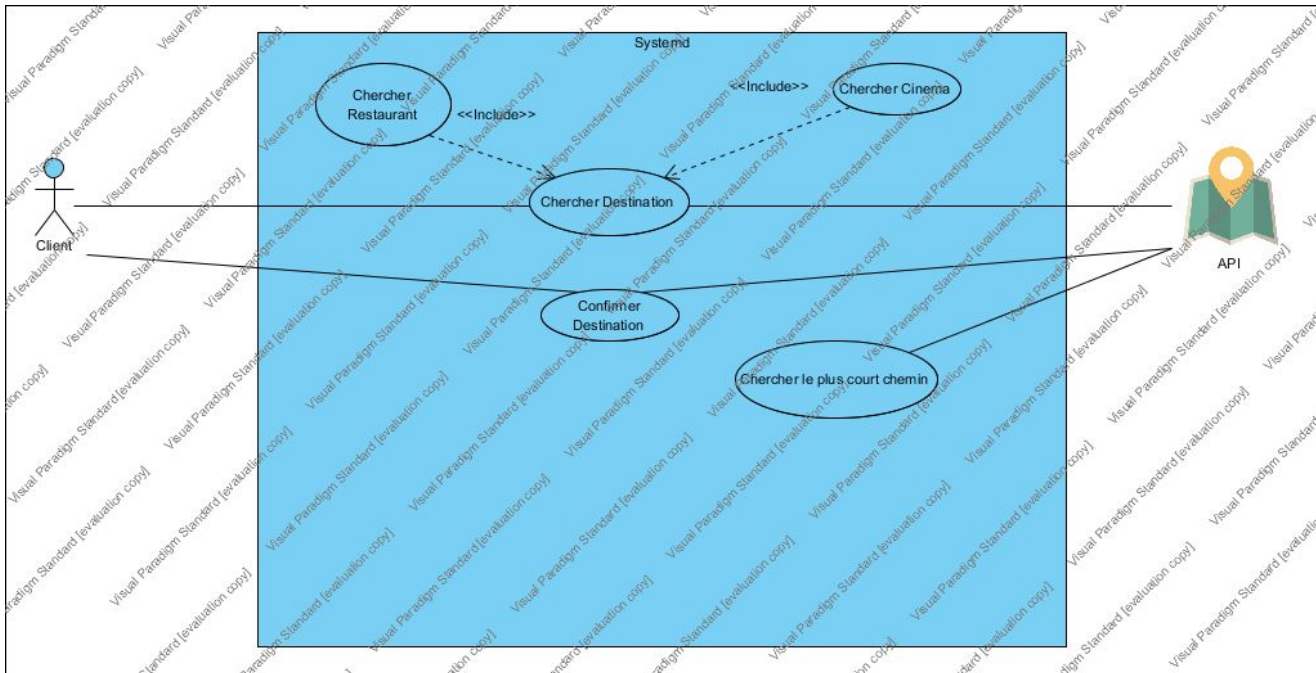
1. Client : pour les clients utilisant l'application.
2. Demand: qui représente les vœux et besoins des clients.
3. Destination: les destinations sur le plan qui corresponde aux besoins des clients.
4. API: écoute les besoins des clients et cherche les destinations qui y corresponde.
5. Pour la classe Demand, elle a des classes filles qui sont DemandRestaurant pour les recherches de restaurant, DemandCinema pour les recherches de cinéma, etc. Egalement, la classe Destination a des classes filles qui sont Restaurant, Cinéma, Bar, etc pour les lieux différentes correspondant aux besoins des clients.

II. Le diagramme de séquence



- Le client peut donner sa commande (soit de restaurant, soit de cinéma, etc) en tapant sur le clavier.
- Un message est après envoyé à l'API, lui demande de chercher les destinations possibles correspondant aux vœux et besoins des clients.
- Ensuite, l'API demande à l'interface d'afficher les résultats trouvés.
- L'interface invite le client à choisir la destination qu'il préfère.
- Le client maintenant peut choisir sa destinations et la confirmer à partir du clavier.
- L'API est ensuite demandée de calculer et trouver le plus court chemin de l'emplacement courante du client à la destination.
- Le chemin est affiché à l'interface pour que le client puisse le suivre.

III. Le diagramme de case d'utilisation



Il y a deux acteurs dans ce diagramme, Client et API. Les deux acteurs ont mêmes deux actions:

- Chercher la destination (restaurants, cinémas, etc sont inclus).
- Confirmer la destination.

Pour l'API, elle a une action supplémentaire qui est "Chercher le plus court chemin".