

# Cahier de conception

BOUYAHIAOUI Amine  
DJEDID Said  
BOUNEGTA Mehdi-Akram

## **Table des matières**

1. Contexte
2. Objectifs
3. Fonctionnement
- 3.1. Contraintes

## **Table des diagrammes UML**

1. Diagramme des cas d'utilisation
2. Diagramme de classe
3. Diagramme de création d'une sortie
4. Diagramme d'un cas particulier interdit
5. Diagramme d'ajout d'un participant
6. Diagramme d'erreur sur la localisation du bar
7. Diagramme d'erreur sur l'ajout d'un participant
8. Diagramme d'erreur sur le restaurant

## **Contexte :**

« c'est quoi ton plan ? » une question qui revient souvent lors de l'organisation d'une soirée, on perd tellement de temps à en trouver un, et souvent un plan qui n'arrange pas tout le monde, on a donc pensé à régler ce problème une bonne fois pour toutes, étant des étudiants en informatique, et des futurs développeurs, on a pensé à créer une application pouvant nous proposer des endroits selon les critères qu'on choisit, où on pourrait éventuellement passer nos soirées, des endroits qui arrangeraient tout le monde ou la plupart notamment en terme de déplacement vu qu'on habite pas tous forcément dans la même ville.

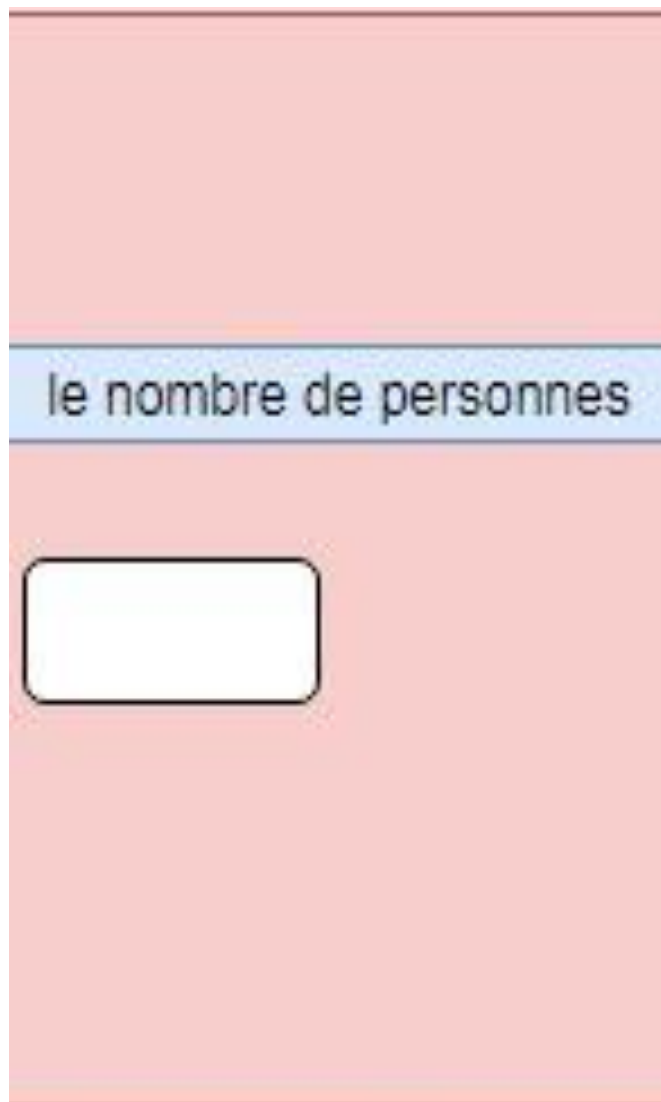
## **Objectifs :**

- L'application devra proposer trois endroits pour passer la soirée, commençant par un bar, un restaurant puis une boite, elle doit prendre en considération les adresses de tous les membres du groupe.
- L'application devra proposer des endroits accessibles à tous les membres du groupe.
- L'application proposera un restaurant et une boite en tenant compte de l'adresse du premier lieu choisi dans un premier temps comme lieu de rencontre à savoir le bar.
- Les utilisateurs de l'application doivent pouvoir choisir le genre de restaurant où ils préféreraient dîner.
- L'application proposera ensuite des restaurant correspondant aux choix des membres du groupe.
- L'utilisateur peut valider ou décliner les propositions suggérées par l'application

## Projet Gla

### **Fonctionnement**

L'application devra prendre le nombre de membres du groupe et les informations correspondants à chacun d'eux, à l'image des dessins ci-dessous :

La maquette d'interface utilisateur est présentée sur un fond rose. Elle se compose d'une barre de titre horizontale de couleur bleu clair au centre, contenant le texte "le nombre de personnes" en noir. En dessous de cette barre, à gauche, se trouve un champ de saisie rectangulaire blanc avec des coins arrondis et une bordure noire.

Maquette - le nombres de personnes.

Membre n°1

Nom

Prénom

Adresse

VALIDER

Maquette - Saisie des informations :

Elle proposera par la suite une liste de type de restaurant, l'utilisateur choisira les types qui lui conviennent et ce à l'image de ce qui est exposé ci-dessous :

A screenshot of a mobile application interface for selecting restaurant preferences. The background is a solid light pink color. At the top, the text "Membre n°1" is displayed in a dark, sans-serif font. Below this, there is a vertical list of four radio button options. Each option consists of a white circular radio button with a black outline, followed by the restaurant type name in a dark, sans-serif font. The options are "Burger", "Japonnais", "kebab", and "orientale".

Membre n°1

- ☐ Burger
- ☐ Japonnais
- ☐ kebab
- ☐ orientale

**Maquette - Sélection des préférences :**

## Projet Gla

A l'aide des informations récoltées sur les profils des utilisateurs (nom d'utilisateur, préférences) et en fonction des adresses de chacun d'eux, pour faciliter le déplacement et les transports, on fixera un lieu de rencontre. Dans un premier temps on proposera un ensemble de bars autour de ce lieu. Une fois le choix effectué par l'utilisateur principal, on passera une liste de restaurants répondant aux attentes des membres notamment en matière de goûts, et de toutes autres préférences. En dernière étape différentes boites sont mises en avant. L'utilisateur fait son dernier choix. Le bar, le restaurant et la boite de nuit choisis sont indiqués avec leurs horaires d'ouvertures et de fermetures et leurs adresses.

### **Contraintes :**

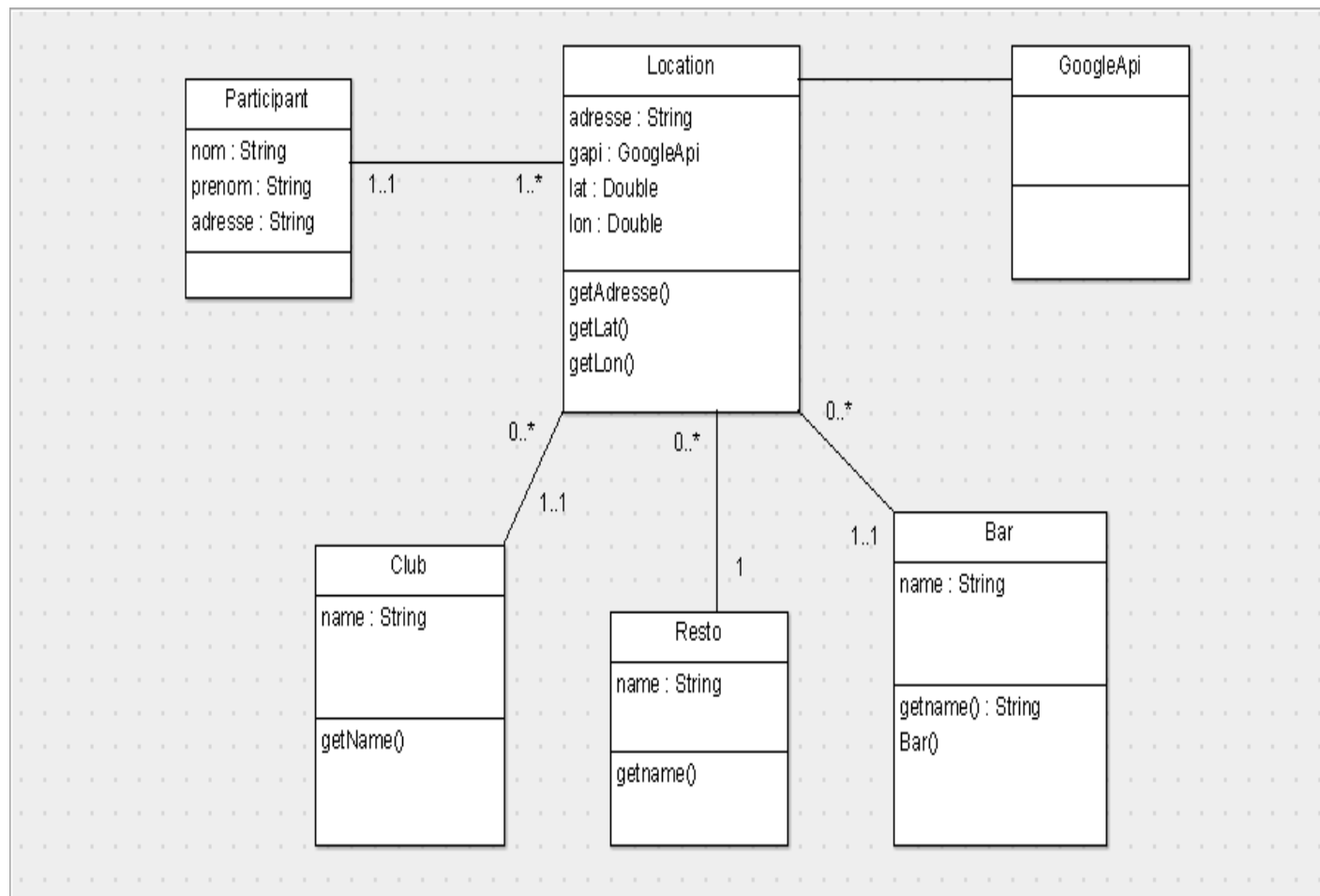
- L'application devra pouvoir fonctionner dans n'importe quelle région navigable du globe terrestre.
- l'application devra calculer des coordonnées GPS du premier lieu de rencontre qui doit être plus ou moins proche de tous les membres du groupe.
- Dans le choix du restaurant l'application doit satisfaire au maximum les choix de la majorité des membres.
- Les restaurants et les boites de nuit proposés doivent être à côté du premier lieu de rencontre (le bar).

## Diagramme de cas d'utilisation :





## Diagramme de classe :



## **Fonctionnement de l'application dans le cas général :**

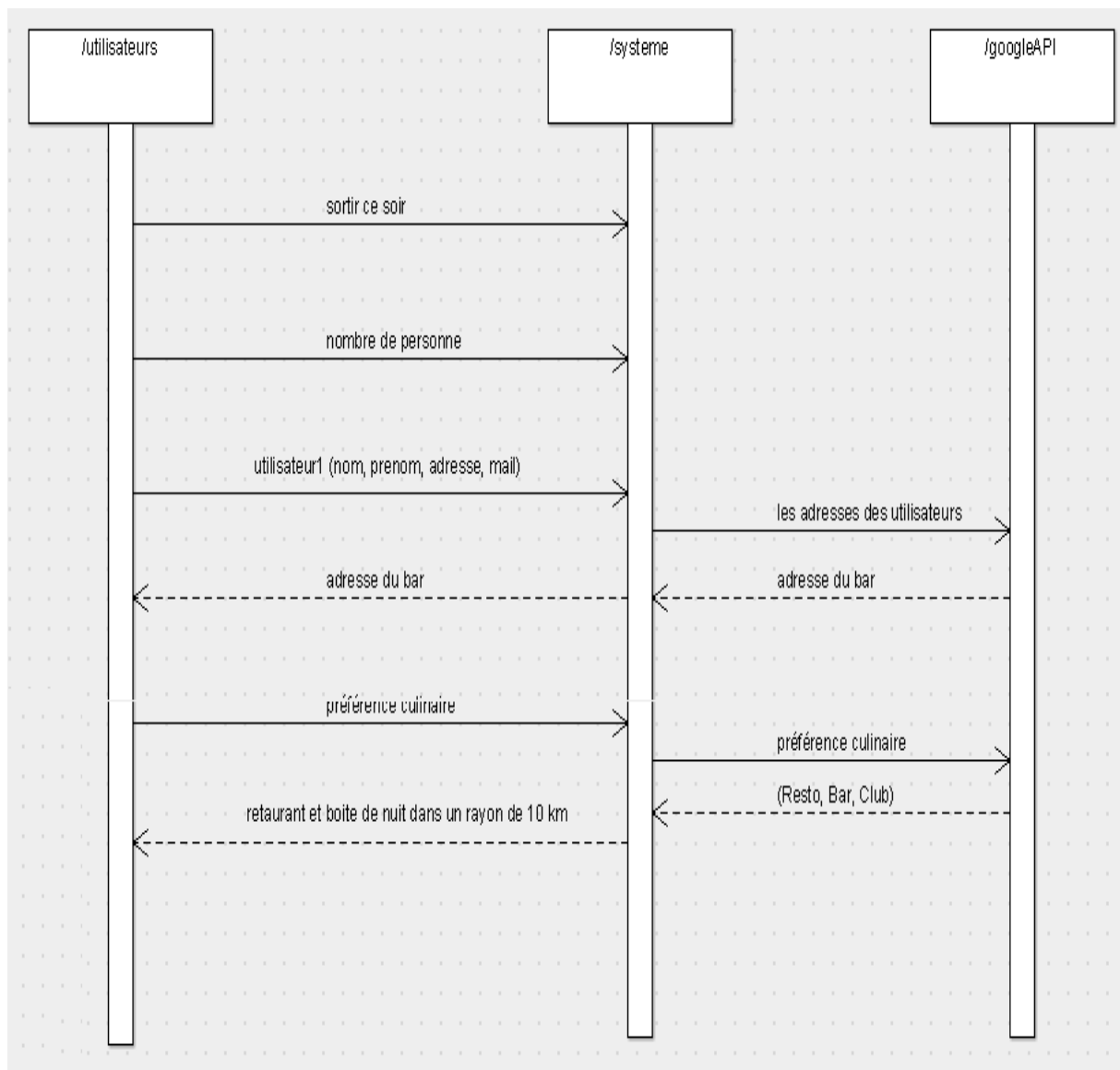
**Un utilisateur lance l'application, une première fenêtre apparaît demandant le nombre de participants, l'utilisateur donne le nombre qui est soumis au système par un bouton, puis d'autres fenêtres apparaissent pour chaque participant demandant ses renseignements (nom, prénom, adresse...).**

**Les données sont stockées dans une base de données sous forme de fichier texte pour une utilisation ultérieure, après ce stockage, les utilisateurs choisissent leurs préférences alimentaires, ces informations sont soumises au système qui propose des restaurants en prenant en considération ces préférences, l'utilisateur principal valide un des choix, le système utilise l'adresse de ce choix pour proposer un bar et une boîte de nuit à proximité et affiche le résultat final composé de trois adresses avec les noms de trois endroits différents, à savoir le bar, le restaurant et la boîte de nuit.**

## Scénario 1 : création d'une soirée sans aucune erreur

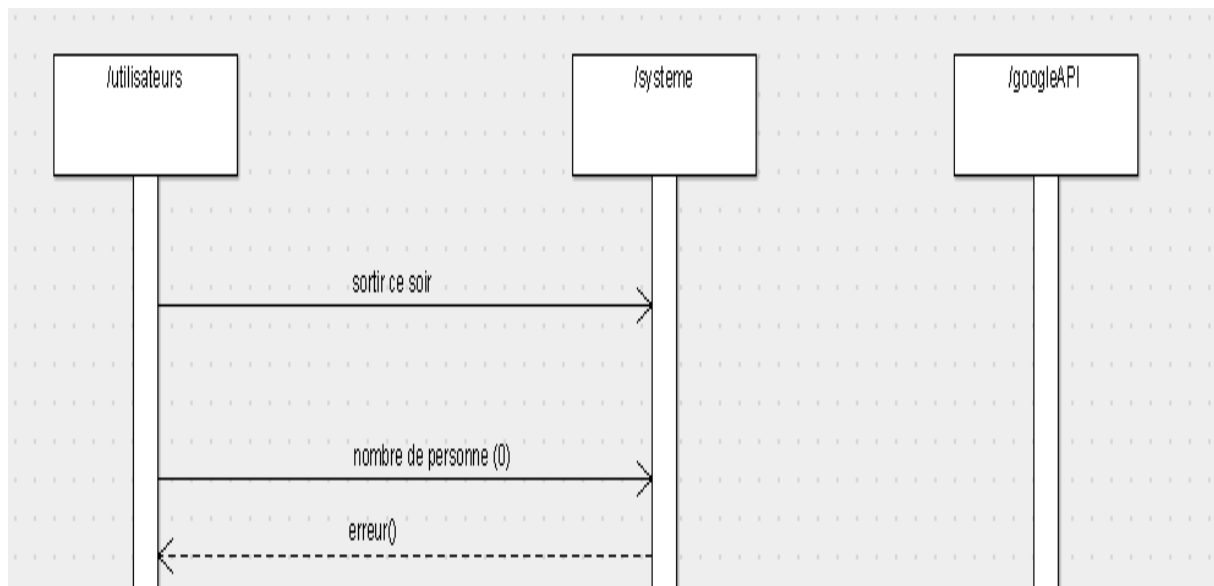
Lors du lancement de l'application, une première fenêtre apparaît, l'utilisateur entre le nombre de personnes, puis pour chaque personne une fenêtre s'ouvre demandant des renseignements, nom, prénom, adresse..., ces informations seront stockées dans une base de données sous forme de fichier texte, chaque ligne correspond à une personne, les données seront distinguées par des virgules de la forme suivante : NOM,Prénom,adresse...

La classe GoogleApi utilise ensuite l'adresse des utilisateurs pour calculer un rayon de 10km dans lequel un bar est sélectionné, le système propose aux utilisateurs des choix de préférences alimentaires, les choix sont soumis au système qui proposera un restaurant qui satisfait la plupart des membres du groupe, puis une boîte de nuit.



## Scénario 2 : création d'une soirée avec une erreur

Le cas illustré dans ce scénario est un cas particulier parmi d'autres, la démarche est la même que celle du premier scénario, sauf que au moment de donner le nombre de personnes, on rentre « 0 » qui est une valeur interdite (même chose pour les valeurs négatives) le système renvoie une erreur en affichant un message d'erreur à l'écran et proposant à l'utilisateur de créer à nouveau une soirée.

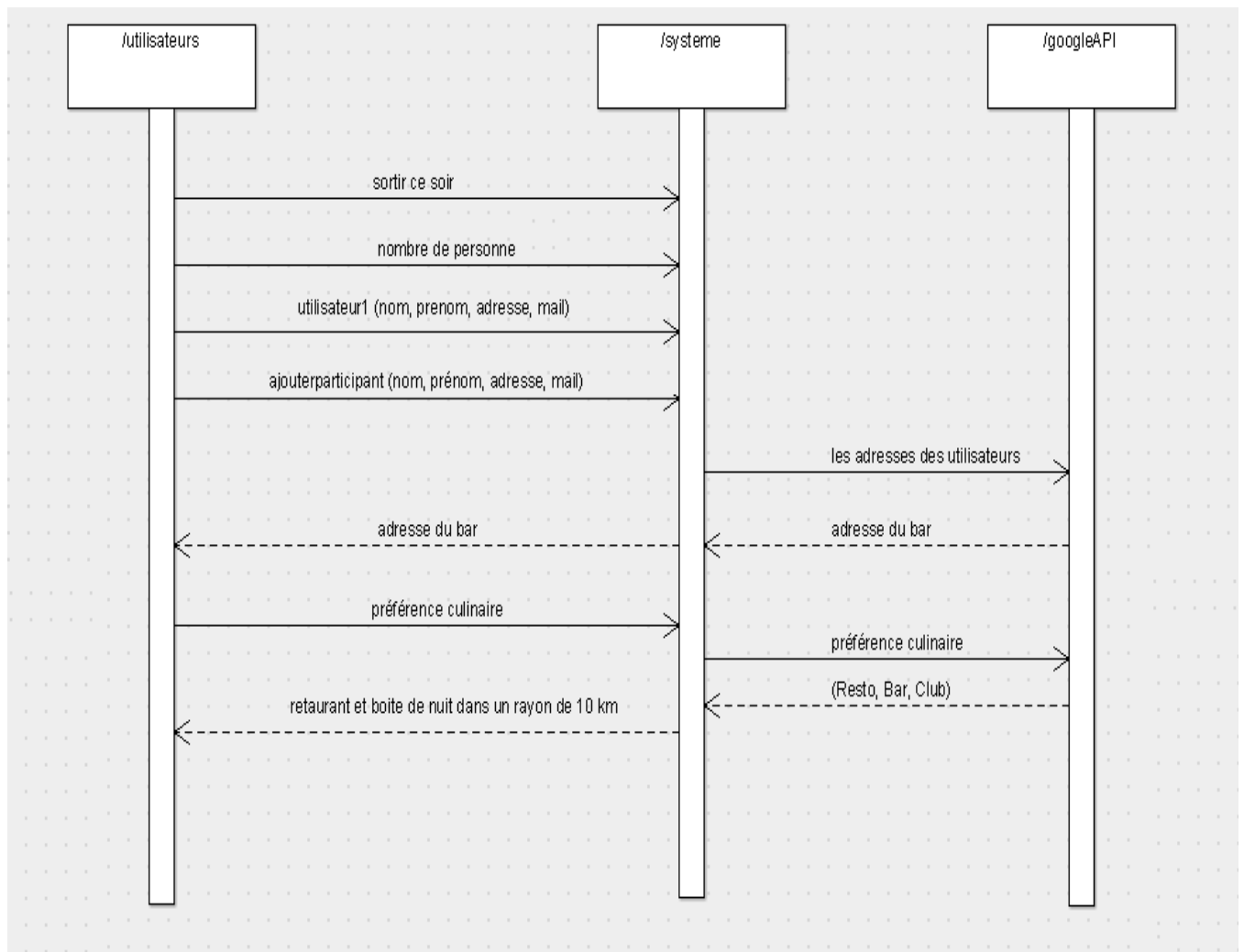


### Diagramme : Diagramme d'une erreur sur l'application :

Lorsque l'on met 0 comme nombre de participants, il y'a erreur et l'utilisateur a une nouvelle tentative.

### Scénario 3 : Ajout d'un participant après avoir validé les utilisateurs.

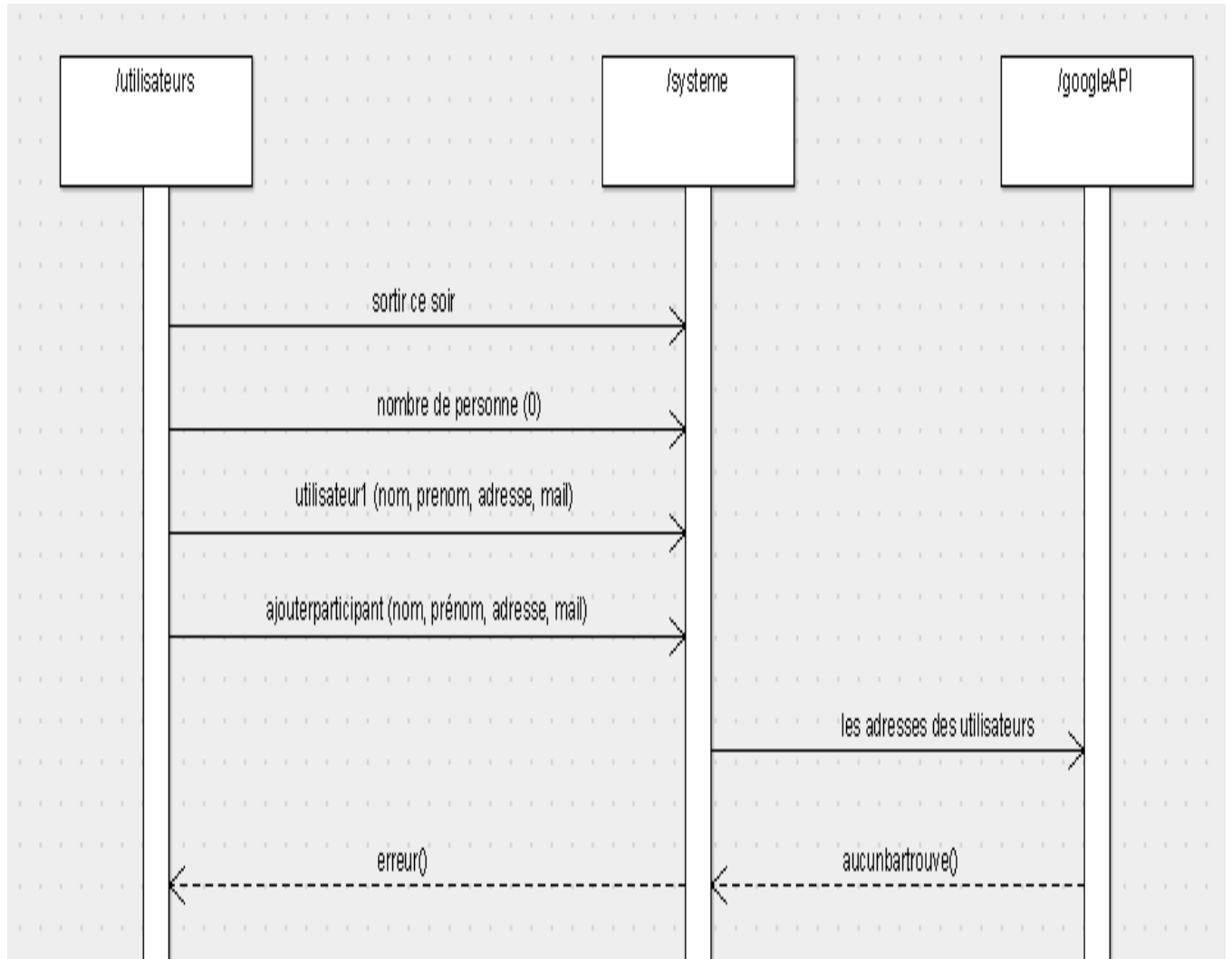
Toujours la même démarche, après l'étape du nombre de personnes et les renseignements des participants, un bouton d'ajout apparaît à la fenêtre, l'utilisateur principal peut alors ajouter des participants si il le souhaite, après l'ajout, le programme se déroule comme prévu.



**Diagramme** : Ajout d'un participant après avoir validé les utilisateurs.

### Scénario 3 : une erreur lors du calcul de l'adresse

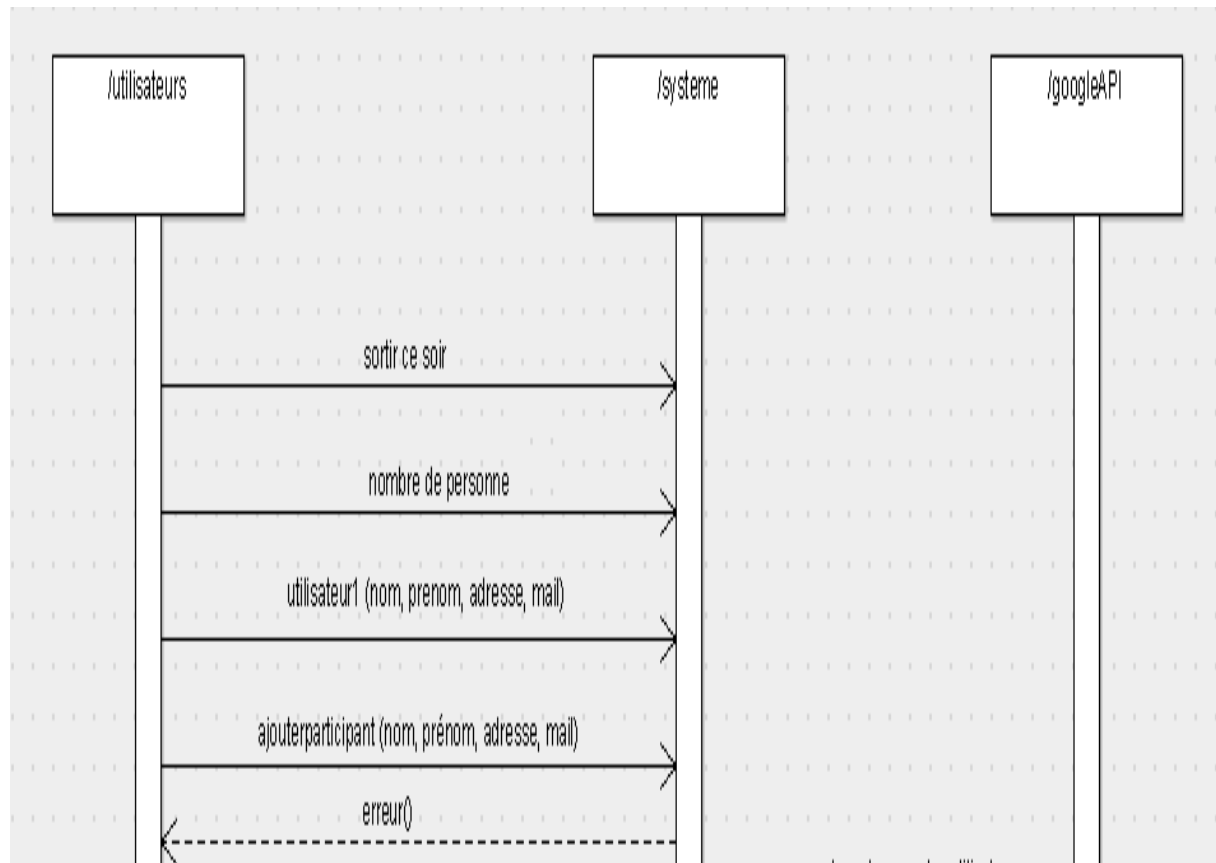
Dans ce cas le système renvoie une erreur, soit parce que y'a une fausse adresse parmi les adresses des participants, ou car y'a une distance considérable entre les adresses comme deux adresses dans deux pays différents, le système renvoie un message d'erreur dans ce cas et propose un nouvel essai.



**Diagramme :** Aucun bar trouvé ne satisfaisant les utilisateurs.

### Scénario 3 : Le participant existe déjà

Ce scénario correspond au cas d'enregistrer une personne qui existe déjà, cela est vérifié par l'adresse mail, un message d'erreur est affiché rappelant que le participant existe déjà, puis le nombre de participants est décrémenté et le programme se poursuit.



**Diagramme** : erreur dans le cas d'une adresse mail déjà utilisée.