



# SYNOPSIS

PROJET MESSENGER  
NICOLAS DUFOUR

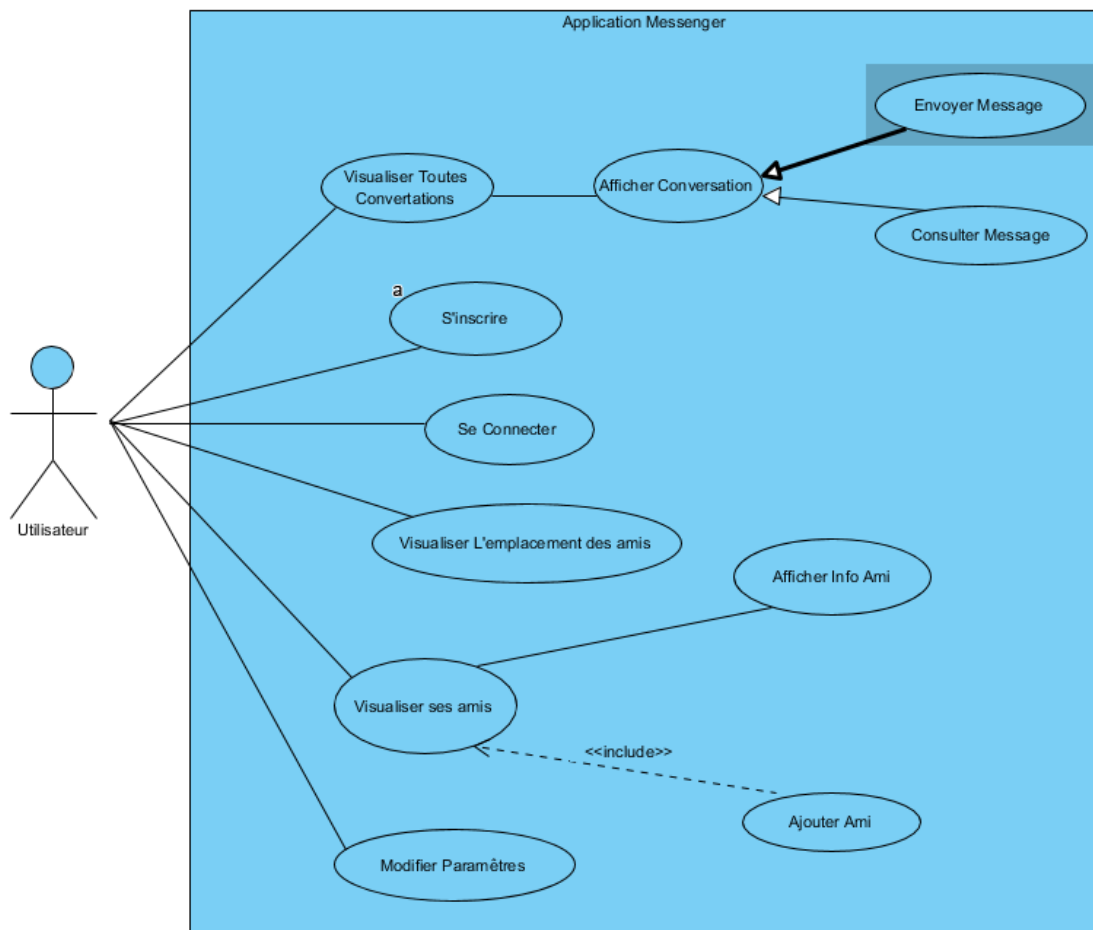
## Table des matières

Description du projet .....	2
Diagramme de cas d'utilisation.....	2
Spécifications de l'application .....	3
Prototypes D'interfaces .....	3
Règles d'utilisations.....	4
Modèle de base de donnée .....	5
Diagramme de classes .....	6
Nouveautés de l'application .....	7
1. Utilisation d'une base de données externe.....	7
2. Utilisation des API Google de géolocalisation .....	7
Référence .....	8

## Description du projet

Application Android de type réseau social permettant de communiquer avec d'autres utilisateurs via message texte. Chaque utilisateur peut créer son propre profil et le personnaliser à son gout. Il pourra ensuite chercher parmi les autres utilisateurs des amis potentiels et ainsi entamer une discussion avec lui. Il pourra lui envoyer de photos par exemple et des messages et ainsi se rapprocher de ses amis.

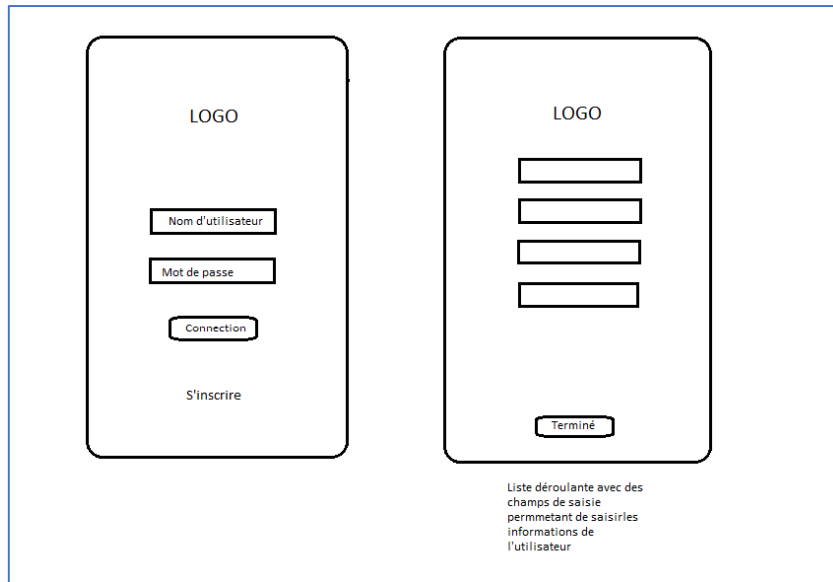
## Diagramme de cas d'utilisation



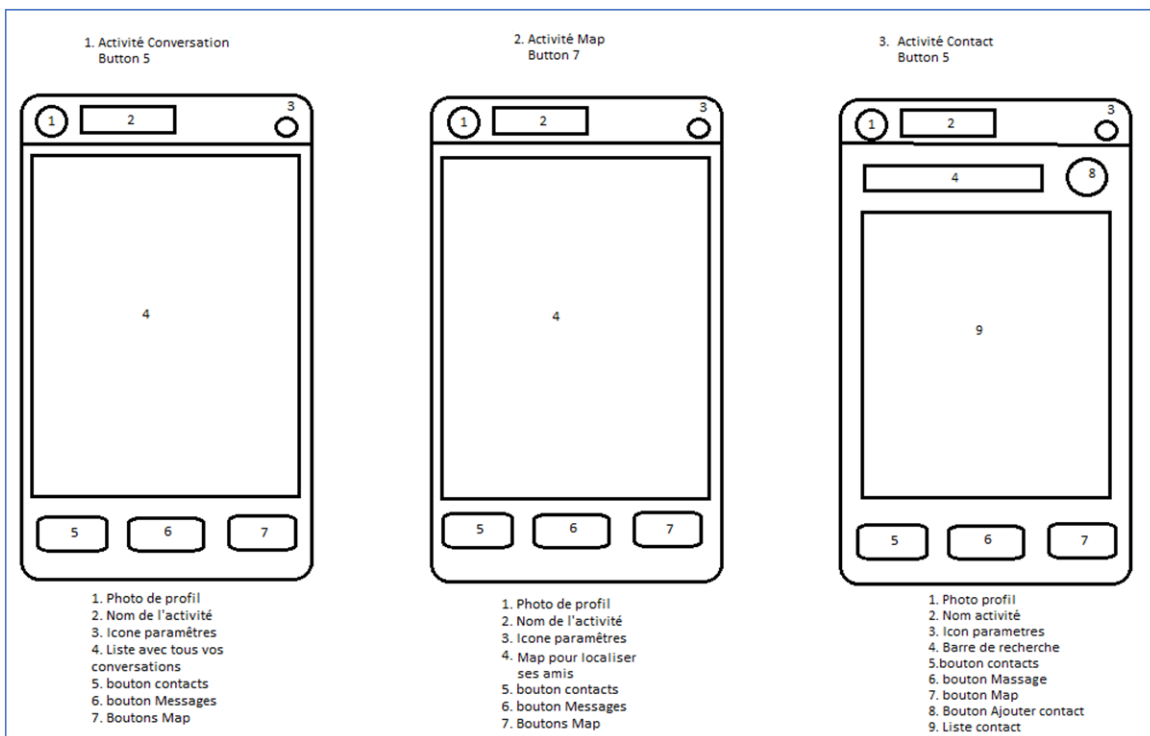
# Spécifications de l'application

## Prototypes D'interfaces

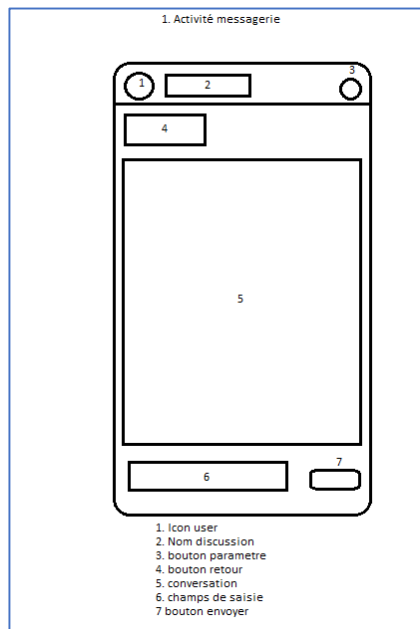
### 1. Les interface de connexion et d'inscription



### 2. Les activités principales du menu d'accueil.



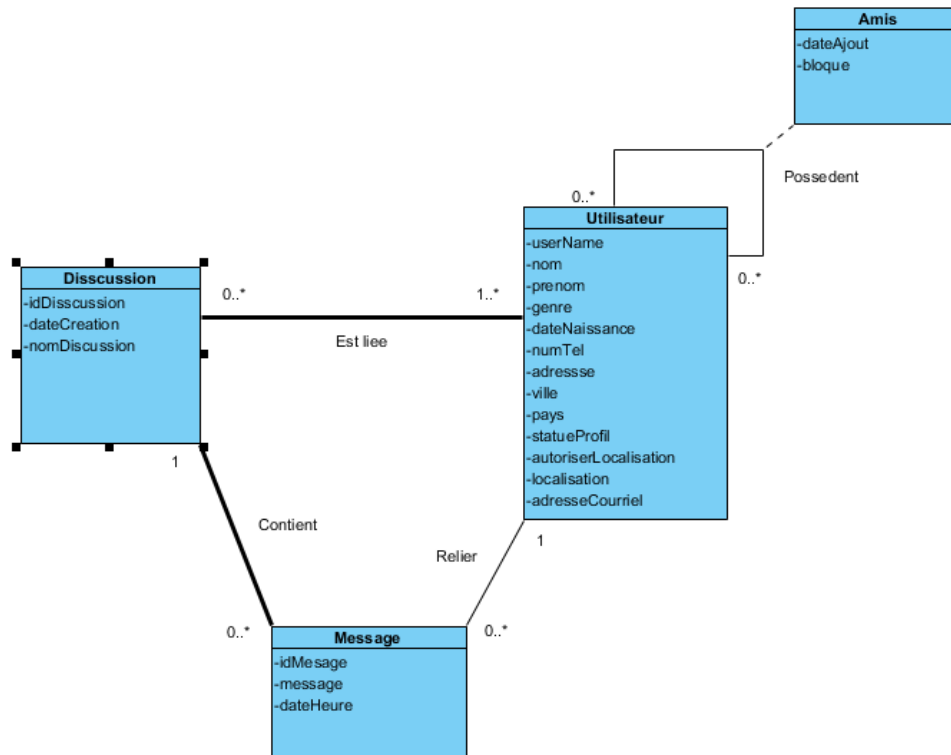
### 3. La discussion



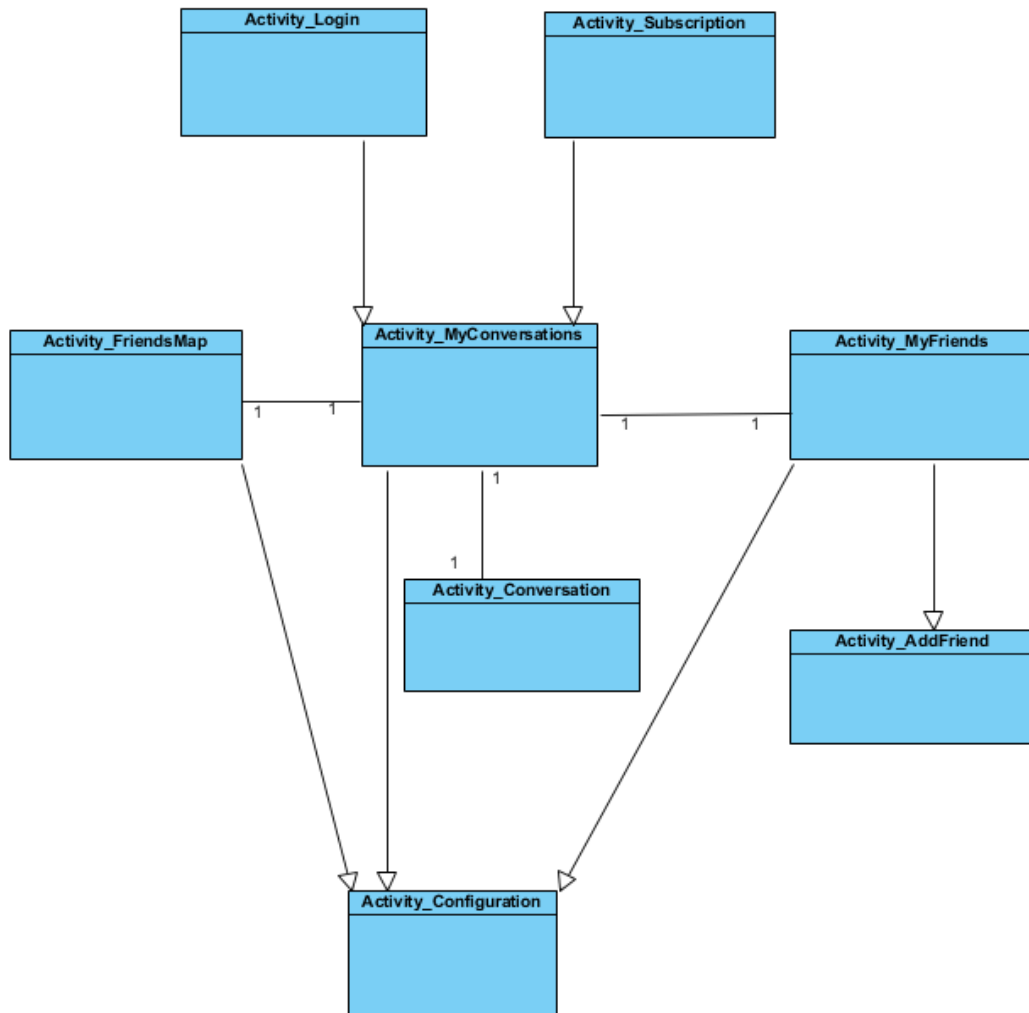
### Règles d'utilisations

- L'utilisateur doit avoir un accès à internet.
- Cette application utilise vos données de localisation.
- La reproduction de cette application est strictement interdite.
- L'utilisation à des fins inadéquates sera sanctionnée

## Modèle de base de donnée



## Diagramme de classes



## Nouveautés de l'application

### 1. Utilisation d'une base de données externe






Afin de pouvoir interagir entre plusieurs utilisateurs sur différents périphériques et cela de façon instantanée, il y aura deux parties importantes à l'application soit l'application elle-même et une base de données MySQL. L'utilisation de script PHP va permettre en envoyant une requête vers celui-ci en http de la traiter et effectuer le traitement sur la base de données et ainsi retourner le résultat à l'application en JSON.

### 2. Utilisation des API Google de géolocalisation

Chaque utilisateur aura la possibilité d'accepter ou refuser d'être localisé par ses amis. Dans le cas où il accepte, lors de chaque ouverture de l'application celle-ci sauvegardera dans la base de données la position géographique de l'utilisateur et celle-ci sera mise à jour toutes les 60 secondes jusqu'à la fermeture de l'application. Grâce à l'API de géolocalisation l'application pourra déterminer la position de l'appareil selon la position GPS de celui-ci et ainsi transmettre ces informations. Dans le cas où un utilisateur ouvre l'activité trouver mes amis, il sera en mesure de voir la position de ceux-ci. L'application va ainsi aller récupérer les informations de ses amis et leurs positions, grâce au service de Google Maps. Une API permettant de positionner des points sur une Map, il sera en mesure de les voir sur celle-ci.



## Référence

-  « Adding a Map with a Marker», dans Google map platform,  
<https://developers.google.com/maps/documentation/android-sdk/map-with-marker>, (Page consulté le 29 Janvier 2019)
  
-  « Select Current Place and Show Details on a Map», dans Google map platform,  
<https://developers.google.com/maps/documentation/android-sdk/current-place-tutorial>, (Page consulté le 29 Janvier 2019)
  
-  « Location», dans Android Developers,  
<https://developer.android.com/reference/android/location/Location>, (Page consulté le 29 Janvier 2019)
  
-  « How to Build Your Own Real-Time Chat App», dans UpWork,  
<https://www.upwork.com/hiring/for-clients/build-real-time-chat-app-like-whatsapp/>, (Page consulté le 29 Janvier 2019)
  
-  « Android Chat Tutorial: Building A Realtime Messaging App», dans ScaleDrone,  
<https://www.scaledrone.com/blog/android-chat-tutorial/>, (Page consulté le 29 Janvier 2019)
- 