**REMERCIEMENTS**

Avant d’entamer ce rapport, je profite de l’occasion pour remercier tout d’abord mes collègues dans la startup dont j’ai participé à sa création en tant que développeur web et mobile et particulièrement le chef de notre équipe et le propriétaire de ce projet « Gaith Troudi »qui n’a pas cessé de m’encourager pendant la durée du stage,. Je le remercie également pour l’aide et les conseils concernant les missions évoquées dans ce rapport, qu’il m’a apporté lors des différents suivis, et la confiance qu’il m’a témoignée.

J’exprime également ma gratitude à l’égard des membres du jury pour l’honneur qu’ils me font en acceptant d’évaluer mon travail.

Table des matières

[Introduction générale 1](#_Toc494403534)

[1. Chapitre1 : Contexte du projet 2](#_Toc494403535)

[Introduction 2](#_Toc494403536)

[1.1 Cadre générale du projet 2](#_Toc494403537)

[1.2 Organisme d’accueil 2](#_Toc494403538)

[1.3 Étude de l’existant 2](#_Toc494403539)

[1.3.1 Description de l’existant 2](#_Toc494403540)

[1.3.2 Solution proposée 3](#_Toc494403541)

[1.4 Choix méthodologique 3](#_Toc494403542)

[1.4.1 Le langage de modélisation unifié UML 3](#_Toc494403543)

[1.4.2 Modèle de développement 4](#_Toc494403544)

[Conclusion 5](#_Toc494403545)

[2. Chapitre 2 : Analyse et spécification des besoins 6](#_Toc494403546)

[Introduction 6](#_Toc494403547)

[2.1 Spécification des besoins 6](#_Toc494403548)

[2.1.1 Besoins fonctionnels : 6](#_Toc494403549)

[Service de localisation : 6](#_Toc494403550)

[Service d’appel d’urgence 7](#_Toc494403551)

[Recherche des instituts d’urgences médicales 7](#_Toc494403552)

[Recherche des stations de police 7](#_Toc494403553)

[Service d’itinéraire 7](#_Toc494403554)

[Gestion des contacts d’urgences 7](#_Toc494403555)

[2.1.2 Besoins non fonctionnels 7](#_Toc494403556)

[2.2 Acteurs de l’application 8](#_Toc494403557)

[2.2.1 Utilisateur 8](#_Toc494403558)

[2.3 Vue globale de l’application 8](#_Toc494403559)

[2.4 Raffinement des scénarios pour l’application mobile 9](#_Toc494403560)

[2.4.1 Cas d’utilisation "se localiser" 10](#_Toc494403561)

[2.4.2 Cas d’utilisation "afficher les instituts d’urgence médicale" 11](#_Toc494403562)

[2.4.3 Cas d’utilisation "afficher la liste des poste de police" 11](#_Toc494403563)

[2.4.4 Cas d’utilisation "appeler les numéros d’urgence" 12](#_Toc494403564)

[2.4.5 Cas d’utilisation "gérer les contacts" 13](#_Toc494403565)

[Conclusion 14](#_Toc494403566)

[3. Chapitre3 : Conception détaillée 15](#_Toc494403567)

[Introduction 15](#_Toc494403568)

[3.1 Architecture générale 15](#_Toc494403569)

[3.2 Conception des cas d’utilisation 15](#_Toc494403570)

[3.2.1 Diagramme de classes de l’application 15](#_Toc494403571)

[3.2.2 Cas d’utilisation "se localiser" 17](#_Toc494403572)

[3.2.3 Cas d’utilisation " appeler les numéros d’urgence " 17](#_Toc494403573)

[3.2.4 Cas d’utilisation "afficher la liste de instituts d’urgences médicales" 18](#_Toc494403574)

[3.2.5 Cas d’utilisation " afficher la liste des postes de polices" 18](#_Toc494403575)

[3.3 Cas d’utilisation " gérer les contacts" 19](#_Toc494403576)

[Conclusion 19](#_Toc494403577)

[3. Chapitre3 : Réalisation 20](#_Toc494403578)

[Introduction 20](#_Toc494403579)

[3.1 Environnement de travail 20](#_Toc494403580)

[3.1.1 Environnement matériel 20](#_Toc494403581)

[3.1.2 Environnement logiciel 20](#_Toc494403582)

[3.2 Description des interfaces 21](#_Toc494403583)

[3.2.1 Interfaces principale avant localisation 21](#_Toc494403584)

[3.2.2 Interface principale après la localisation 24](#_Toc494403585)

[3.2.3 Interfaces d’affichage de la carte routière et la vue par satellite 24](#_Toc494403586)

[3.2.4 Interface d’affichage de la liste des instituts d’urgence médicale 25](#_Toc494403587)

[3.2.5 Interface d’affichage de la liste des postes de police 27](#_Toc494403588)

[3.2.6 Interface de la liste des contacts d’urgence : 27](#_Toc494403589)

[Conclusion 29](#_Toc494403590)

[4. Conclusion et perspectives 30](#_Toc494403591)

[5. Bibliographie 31](#_Toc494403592)

Table des figures

[Figure 1:Cycle de vie d'une application](file:///C:\Users\AYMEN\Desktop\STAGE\rapport-de-stage.docx" \l "_Toc494407464) 4

[Figure 2:Diagramme de cas d'utilisation globale 9](#_Toc494407465)

[Figure 3: Diagramme de cas d'utilisation "se localiser" 10](#_Toc494407466)

[Figure 4: Diagramme de cas d'utilisation "afficher la liste des institut d’urgence médicale" 11](#_Toc494407467)

[Figure 6 : Diagramme de cas d'utilisation "afficher la liste des postes de police" 12](#_Toc494407468)

[Figure 7: Diagramme de cas d'utilisation "appeler les numéros d’urgence" 12](#_Toc494407469)

[Figure 8: de cas d'utilisation "gérer les contacts" 13](#_Toc494407470)

[Figure 9:Diagramme de classes 16](#_Toc494407471)

[Figure 10: Diagramme de séquence du cas "se localiser" 17](#_Toc494407472)

[Figure 11 : diagramme de séquence du cas d’utilisation « appeler numéro d’urgence » 17](#_Toc494407473)

[Figure 12:Diagramme de cas d'utilisation «afficher la liste des instituts d'urgence » 18](#_Toc494407474)

[Figure 13Diagramme de cas d'utilisation «afficher la liste des postes de police» 18](#_Toc494407475)

[Figure 14:Diagramme de cas d'utilisation «gérer les contacts » 19](#_Toc494407476)

[Figure 15:Interface principale avant localisation 21](#_Toc494407477)

[Figure 16:interface de l'alerte de connexion 22](#_Toc494407478)

[Figure 17:interface de l'alerte de GPS 23](#_Toc494407479)

[Figure 18:Interface principale après la localisation 24](#_Toc494407480)

[Figure 19:Interfaces d’affichage de la carte routière et la vue par satellite 25](#_Toc494407481)

[Figure 20:Interface d’affichage de la liste des instituts d’urgence médicale 26](#_Toc494407482)

[Figure 21:interface de l'itinéraire 26](#_Toc494407483)

[Figure 22:Interface d’affichage de la liste des postes de police 27](#_Toc494407484)

[Figure 23:Interface de la liste des contacts d’urgence 28](#_Toc494407485)

[Figure 24 : interface du menu de la liste de contacts 28](#_Toc494407486)

# Introduction générale

Au cours des dernières années, la technologie du développement mobile fut adoptée par un grand nombre d’entreprises et a été intégrée dans presque tous les domaines. De plus en plus de petites et grandes entreprises décident de concevoir une application mobile plutôt que de se rabattre seulement sur un site web. Ainsi les applications mobiles sont devenues une tendance et à la fois une nécessité pour réussir un certain projet.

Les applications mobiles sont de plus en plus demandées quelque soit catégorie tel que les jeux, les applications à but sociale, celles qui offre des services qui facilitent la vie quotidiennes des utilisateurs ...

Dans cette optique, le chef de la startup m’a confié la mission de développer une application mobile multiplateformes qui sert à faciliter la prise des décisions et fournit les numéros de téléphones nécessaires dans les cas d’urgence .Cette mission entre dans le cadre d’un stage d’été effectué au sein du « PynToPyn technologies ».

Ce rapport prend en charge la description de toutes les phases de la conception et de la réalisation de ce projet, il s’articule en quatre chapitres :

* Le premier intitulé « Contexte du projet »présentera l’organisme d’accueil, les objectifs à viser, l’état de l’art et la méthodologie adoptée au cours de ce projet.
* Au cours du deuxième intitulé « Analyse et spécification des besoins »nous allons présenter une analyse des besoins en définissant les besoins fonctionnels et non fonctionnels.
* Au cours du troisième intitulé « Conception détaillée »nous allons étudier les aspects conceptuels de notre projet.
* Le dernier chapitre intitulé « Réalisation » sera consacré à la réalisation et les tests de l’application développée.

Je finirais ce rapport par une conclusion et quelques perspectives proposées pour l’évolution et la continuité du notre projet.

# Chapitre1 : Contexte du projet

# Introduction

Dans ce chapitre, je vise à mettre mon travail dans son contexte général. D’abord, je vais présenter le cadre général de mon projet et l’organisme d’accueil. Ensuite, je vais poser la problématique suite à laquelle j’ai conçu à percevoir les objectifs spécifiques de cette application. Par la suite, je présenterai la méthode de développement adoptée.

## Cadre générale du projet

Le sujet introduit par le présent rapport est intitulé « Développement d’une application mobile pour les cas d’urgences »et rentre dans le cadre d’un stage d’été.

Le but de réaliser une telle application et d’un coté, l’absence de certains services d’urgences dans les applications similaires déjà proposées dans le marché, et d’un autre coté elle va nous servir comme moyen de marketing afin que la startup commence à être connue et remarquée par nos futurs clients.

## Organisme d’accueil

L’établissement qui m’a accueilli au cours de mon stage d’été est la startup « PynToPyn technologies », crée par un groupe d’étudiants dont moi même, pour le moment elle est composé seulement du propriétaire et le représentant de ce projet « Gaith Troudi » et le reste sont des développeurs web et mobile et des designers.

## Étude de l’existant

### Description de l’existant

En naviguant entre les boutiques d’applications en ligne tel que le Play Store et l’App Store les applications de SOS qui servent à aider les utilisateurs en cas d’urgence ne sont pas multiples de plus la plupart des applications similaires existantes ne sert qu’a faire des appels d’urgences alors toutes l’équipe de la startup à pris la décision de construire une telle application en intégrant d’autres fonctionnalités utiles et dans la plupart du temps nécessaires dans les cas d’urgences . Parmi les inconvénients des solutions existantes on note :

* **Accessibilité et utilisation :** certaines applications de ce type obligent l’utilisateur de s’authentifier pour y accéder, sachant que certaines fonctionnalités doivent être accessible directement tel que les appels des numéros d’urgences.
* **Utilité :** pour la majorité des applications de SOS existantes la principale fonctionnalité ne consiste à rien qu’offrir un accès rapides aux numéros d’urgences.

### Solution proposée

La solution adéquate sera de développer une application mobile multiplateforme qui ne demande aucune authentification dont les fonctionnalités principales sont :

-fournir des boutons d’appel d’urgences

-fournir un service de localisation pour l’utilisateur

-un service de recherches des établissements utiles dans les cas d’urgences tel que les hôpitaux et les stations de police

-une liste personnalisable de contacts que l’utilisateur les considère utiles dans les cas d’urgences

Le but de cette application est d’être à l’aide de l’utilisateur suite aux accidents et les situations qui nécessitent des interventions urgentes.

## Choix méthodologique

Au cours de la conception de notre projet nous avons utilisé le langage unifie pour la modélisation objet UML (Unified Modeling Language).

### Le langage de modélisation unifié UML

UML[1], se définit comme un langage de modélisation graphique et textuel destiné à comprendre et à définir des besoins, spécifier et documenter des systèmes, esquisser des architectures logicielles, concevoir des solutions et communiquer des points de vue. Il véhicule en particulier :

* Les concepts des approches par objets : classe, instance, classification, etc.
* Intégrant d'autres aspects : associations, fonctionnalités, événements, états, séquences, etc.

UML définit 9 types de diagrammes devisés en deux catégories :

* Diagrammes statiques (structurels) : diagramme de classe, d'objet, de composant, de déploiement et de diagramme de cas d'utilisation.
* Diagrammes dynamiques (comportementaux) : diagramme d'activité, de séquence, d'état-transition et de diagramme de collaboration.

Pour la modélisation des besoins, nous utilisons les diagrammes UML suivant : Diagramme de cas d'utilisation, diagramme de séquence et diagramme de classe.

### Modèle de développement

Pour cette application, j’ai choisi d'utiliser le modèle en cascade pour décrire le cycle de vie du projet.

Le modèle en cascade décrit cette succession d’étapes qui sont représentées dans la figure suivante :



Figure 1:Cycle de vie d'une application

Ce modèle comporte 7 phases : analyse des besoins, spécifications, conception de l’architecture, conception détaillée, implémentation, tests (validation) et enfin installation. (3)

• **Analyse des besoins :** c’est-à-dire exprimer les besoins de l’utilisateur sous forme précise.

• **Conception générale :** processus durant lequel on doit imaginer, proposer une architecture pour satisfaire les spécifications.

• **Conception détaillée :** consistant à définir précisément chaque sous-ensemble de l’application.

• **Implémentation** : c’est la traduction dans un langage de programmation des fonctionnalités définies lors de phases de conception.

• **Tests :** permettant de vérifier individuellement que chaque sous-ensemble de l’application est implémenté conformément aux spécifications.

## Conclusion

Dans ce chapitre introductif, j’ai présenté l’organisme d’accueil et le travail demandé ainsi que les fonctionnalités qu’il doit offrir. De plus, j’ai analysé les applications similaires existantes et situé leurs inconvénients j’ai présenté la méthodologie de développement suivie pour l’élaboration du projet. Dans le chapitre suivant, je ferais une analyse des besoins en déterminant les objectifs et les besoins fonctionnels et non fonctionnels.

# Chapitre 2 : Analyse et spécification des besoins

## Introduction

Tout au long de ce chapitre, je vais parler des différentes fonctionnalités de mon application ainsi que les besoins à satisfaire. Dans une première partie, je focaliserais sur les objectifs et les besoins fonctionnel et non fonctionnels, ensuite, je présenterais les diagrammes de cas d’utilisation globale et raffiner ensuite les différents scénarios.

## Spécification des besoins

**La spécification des besoins est l’une des principales étapes dans le développement d’une application. Ainsi, la partie suivante du chapitre est consacrée à présenter les besoins fonctionnels et non fonctionnels de mon projet.**

### Besoins fonctionnels :

Les besoins fonctionnels de cette application sont un service de localisation, service d’appel d’urgence, recherche des instituts d’urgences médicales les plus proches,, recherche des postes de police les plus proches de l’utilisateur, gestion des contacts d’urgences, service d’itinéraire pour chacun des établissements trouvés par les deux autres services cités.

### Service de localisation :

* Déterminer l’emplacement de l’utilisateur par le GPS.
* Fournir une l’emplacement sous forme d’adresse lisible.
* Fournir la possibilité de voir l’emplacement sur la carte routière de Google Maps ainsi qu’une vue par satellite.

### Service d’appel d’urgence

* Bouton pour appeler la police.
* Bouton pour appeler la protection civile.
* Bouton pour appeler le SAMU

### Recherche des instituts d’urgences médicales

* Fournir une liste contenant les instituts d’urgences médicales les plus proches de l’emplacement d’utilisateur.

### Recherche des stations de police

* Fournir une liste contenant les postes de polices les plus proches de l’emplacement d’utilisateur.

### Service d’itinéraire

* Fournir la route entre l’emplacement de l’utilisateur et l’institut via Google Maps

### Gestion des contacts d’urgences

* Ajouter des contacts.
* Effacer un contact
* Effacer toute la liste des contacts
* Appeler un contact

### Besoins non fonctionnels

Les besoins non fonctionnels de notre application sont l’ergonomie, l’efficacité, la maintenance, la rapidité de traitement et la performance.

• **Ergonomie**

L’ergonomie est un élément important dans cette application, les interfaces doivent être conviviales, simples et organisées avec cohérence en respectant les règles de l’IHM, de façon à garantir une prise en main rapide des fonctionnalités et une aisance d’utilisation.

• **Efficacité**

L’application doit être fonctionnelle indépendamment de toutes circonstances pouvant entourer l’utilisateur.

• **Maintenance**

Le code de l’application doit être lisible et compréhensible afin d’assurer son état évolutif et extensible par rapport aux besoins des utilisateurs.

• **Rapidité de traitement**

En effet, il est impérativement nécessaire que la durée d'exécution des traitements s'approche le plus possible du temps réel.

• **Performance**

Un logiciel doit être avant tout performant, c'est-à-dire à travers ses fonctionnalités, il répond à toutes les exigences des usagers d'une manière optimale.

## Acteurs de l’application

Dans cette partie, je présente l’acteur et les tâches qu’il peut entreprendre en utilisant cette application. Cet acteur est l’utilisateur.

Notre projet se compose d’une seule partie : une application mobile (Front end) fournie pour les utilisateurs afin d’accéder aux services de que je viens de citer dans la partie des besoins fonctionnels.

### Utilisateur

Les utilisateurs peuvent faire les taches suivantes :

* Faire des appels d’urgences.
* Déterminer leurs localisations.
* Consulter leurs localisations sur une carte routière
* Rechercher les instituts d’urgence médicale les plus proches.
* Rechercher les postes de police les plus proches.
* Gérer la liste des contacts d’urgence.

## Vue globale de l’application

La vue globale de mon projet présente les différentes fonctionnalités et les interactions directes avec les utilisateurs.

La figure ci-dessous illustre le diagramme de cas d’utilisation globale du mon projet.

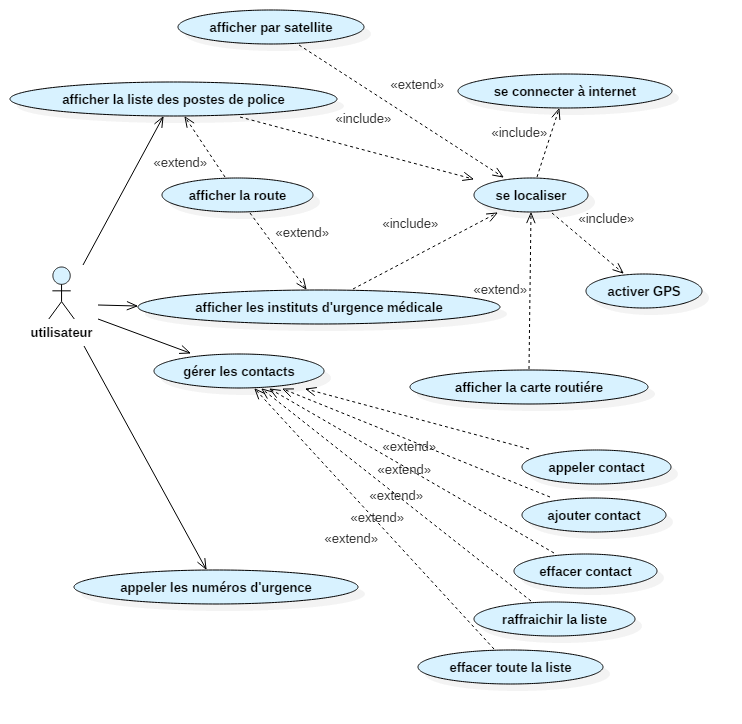


Figure 2:Diagramme de cas d'utilisation globale

## Raffinement des scénarios pour l’application mobile

Au cours de cette partie je vais décrire tous les cas d’utilisation de mon application mobile.

### Cas d’utilisation "se localiser"

Ce cas d’utilisation permet de se connecter pour utiliser les fonctionnalités de notre application. La figure et le scénario suivants permettent de décrire mieux ce cas d’utilisation.

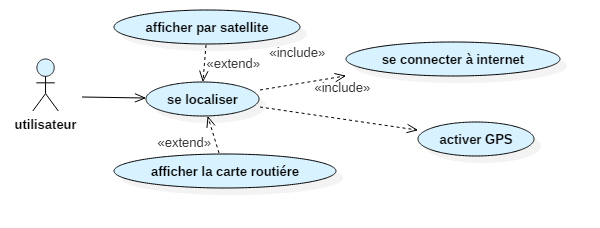


Figure 3: Diagramme de cas d'utilisation "se localiser"

* **Objectif:**

Ce cas d’utilisation permet à l’utilisateur de déterminer sa localisation en ayant recourt à l’API de Google Maps. Ce cas d’utilisation nécessite l’activation du GPS et une connexion internet. Après la localisation l’utilisateur peut consulter son emplacement sur la carte routière de Google ou par satellite.

* **Acteur principal :**

–Utilisateur.

* **Scénario:**

1 L’utilisateur accède à la page d’accueil de l’application (interface principale).

2 L’utilisateur clique sur le bouton de « get location ».

3 l’utilisateur sera informé au cas d’absence de connexion internet ou au cas ou le service de localisation sur le téléphone n’est pas activé. Et il aura la possibilité de fixer ces deux problèmes.

4 L’utilisateur reçoit son adresse courante et aura la possibilité de consulter son emplacement sur carte ou par satellite en cliquant sur le bouton « show map »qui apparaitra sous l’adresse.

### Cas d’utilisation "afficher les instituts d’urgence médicale"

Suite à la localisation, ce cas d’utilisation permettra l’affichage d’une liste contenant les instituts d’urgence médicale les plus proches de son emplacement et aura la possibilité de consulter la route vers chaque institut appartenant à la liste. Ce cas d’utilisation est illustré par la figure suivante.

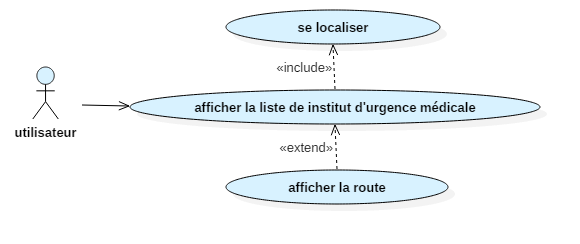


Figure 4: Diagramme de cas d'utilisation "afficher la liste des institut d’urgence médicale"

* **Scénario:**

Après la localisation l’utilisateur aura l’accès pour consulter le contenu de cette liste

1 L’utilisateur accède à la liste en cliquant sur « nearest hospitals »

2 l’utilisateur peut cliquer sur un élément de la liste et voir la route vers cet emplacement soit par l’application Google maps installé sur le téléphone ou sur le navigateur par défaut.

La liste contiendra dix élément au maximum pour garantir qu’ils sont proches de l’utilisateur et ils seront ordonnés selon leur distance par rapport à l’utilisateur.

### Cas d’utilisation "afficher la liste des poste de police"

Suite à la localisation, ce cas d’utilisation permettra l’affichage d’une liste contenant les postes de police les plus proches de son emplacement et aura la possibilité de consulter la route vers chaque institut appartenant à la liste. Ce cas d’utilisation est illustré par la figure suivante.

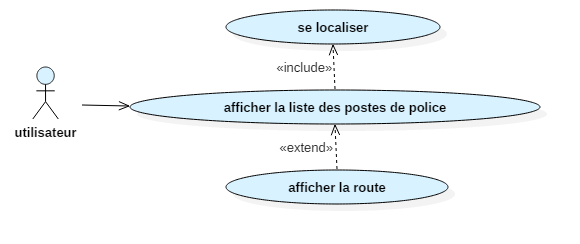


Figure 5 : Diagramme de cas d'utilisation "afficher la liste des postes de police"

* **Scénario:**

Après la localisation l’utilisateur aura l’accès pour consulter le contenu de cette liste

1 L’utilisateur accède à la liste en cliquant sur « nearest police stations »

2 l’utilisateur peut cliquer sur un élément de la liste et voir la route vers cet emplacement soit par l’application Google maps installé sur le téléphone ou sur le navigateur par défaut.

La liste contiendra dix élément au maximum pour garantir qu’ils sont proches de l’utilisateur et ils seront ordonnés selon leur distance par rapport à l’utilisateur.

### Cas d’utilisation "appeler les numéros d’urgence"

Ce cas d’utilisation permet à l’utilisateur d’avoir un accès rapide aux numéros d’urgences et les appeler directement



Figure 6: Diagramme de cas d'utilisation "appeler les numéros d’urgence"

* **Scénario:**

Dés le lancement de l’application, l’utilisateur aura la possibilité d’appeler les numéros d’urgences correspondants à la Tunisie.

1 L’utilisateur peut cliquer sur le bouton ayant une image d’ambulance pour appeler directement le SAMU.

2 l’utilisateur peut cliquer sur le bouton ayant une image d’incendie pour appeler la protection civile

3 l’utilisateur peut cliquer sur le bouton ayant une image de police pour appeler la police de secours.

### Cas d’utilisation "gérer les contacts"

Ce cas d’utilisation permet à l’utilisateur de créer une liste de contacts d’urgences personnelle et les gérer comme le montre cette illustration :

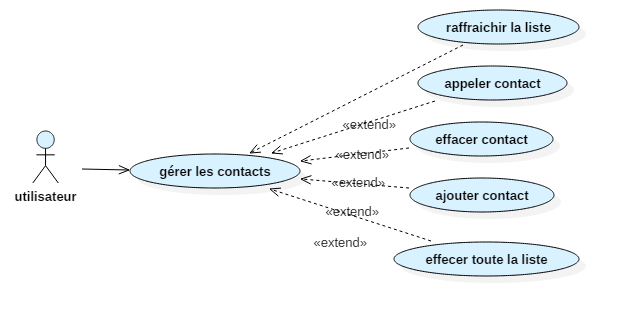


Figure 7: de cas d'utilisation "gérer les contacts"

* **Scénario:**

Dés le lancement de l’application, l’utilisateur aura la possibilité de consulter la liste de ces contacts d’urgence.

1 L’utilisateur peut cliquer sur le bouton « + » pour ajouter un contact à la liste

2 l’utilisateur peut cliquer sur le bouton « X » à coté de chaque contact pour l’effacer

3 l’utilisateur peut cliquer sur le menu dans la barre de navigation à droite pour soit rafraîchir la liste ou l’effacer entièrement.

## Conclusion

Dans ce chapitre j’ai présenté les principaux objectifs de mon projet ainsi que les besoins fonctionnels et non fonctionnels, puis j’ai fini par exposer les diagrammes des cas d’utilisation. Dans le chapitre suivant, je présenterais une conception de notre projet.

# Chapitre3 : Conception détaillée

# Introduction

Le présent chapitre sera dédié à la conception qui est une phase très importante dans le cycle de développement d’un système puisqu’elle prépare l’implémentation. Elle permet de décider comment satisfaire les besoins dégagés dans la phase de spécification. En premier lieu, il y’aura l’architecture de l’application. En second lieu, une description des aspects statiques et dynamiques à l’aide des diagrammes UML.

## Architecture générale

Certaines fonctionnalités de mon application consiste à envoyer des requêtes HTTP et récupère des réponses sous format JSON [2] à partir des APIs de Google maps[3] et Google places[4] et les convertir pour être lisible. Les autres fonctionnalités se basent sur des plugins offerts par la plateforme de Xamarin Forms[5].

## Conception des cas d’utilisation

Après la construction des diagrammes de cas d’utilisation, il faut modéliser les aspects dynamiques de ce projet et spécifier le comportement du système ainsi que l’évolution des objets dans le temps, donc dans cette partie contiendra les diagrammes de classe et de séquence pour certains cas d’utilisation.

### Diagramme de classes de l’application

Les classes principales utilisées pour la construction de cette application sont :

-**classe contact** : objet représentant les contacts gérer par l’utilisateur dont les attributs sont ; nom et numéro téléphone et ID.

-**classe places** : c’est une classe représentant les instituts d’urgences médicales, les postes de police et l’emplacement courant de l’utilisateur, ses attribut sont : nom, latitude, longitude, ID et l’adresse .cette classe et les deux autres classes qui en héritent (destination et Current location) sont nécessaires pour l’itinéraire et même pour le remplissages des listes de postes de police et des institut d’urgence médicale.

-**classe position :** cette classe est nécessaire pour pouvoir utiliser la classe Geocoder qui est implémentée par défaut dans Xamarin Forms, elle contient deux attributs : latitude et longitude.

-**classe geocoder** : est une classe appartenant à un plugin de la plateforme Xamarin, elle permet de convertir un attribut de type position en une adresse lisible.

La figure 8 suivante représente le diagramme de classe de mon application :

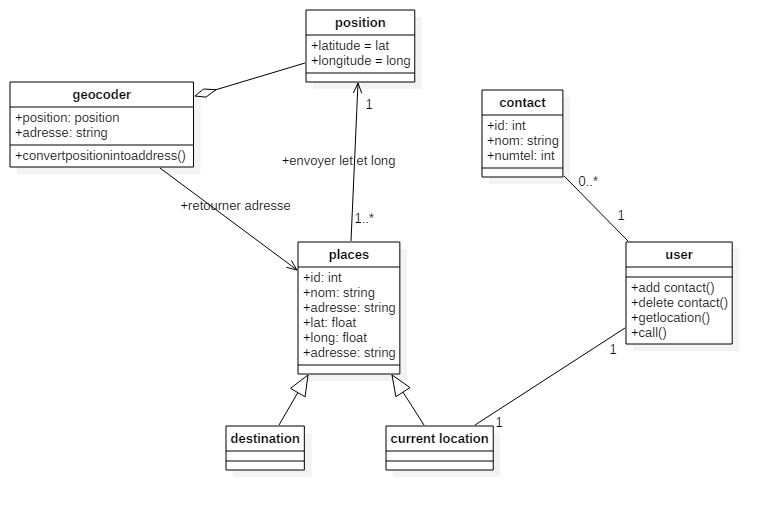


Figure 8:Diagramme de classes

### Cas d’utilisation "se localiser"

Le diagramme de séquence relatif au cas d’utilisation «se localiser" est présenté par la figure 9 suivante :

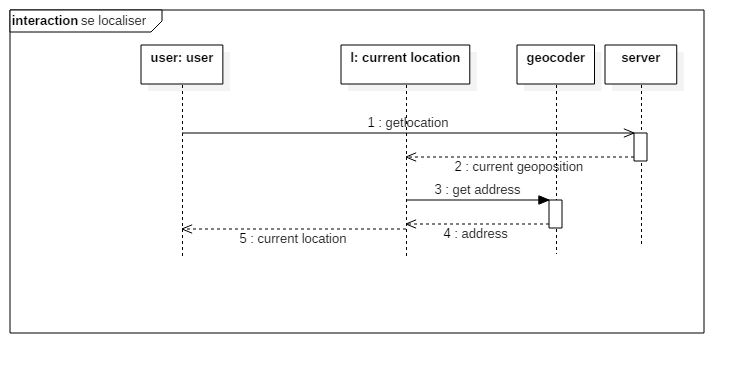


Figure 9: Diagramme de séquence du cas "se localiser"

### Cas d’utilisation " appeler les numéros d’urgence "

La figure suivante illustre le diagramme de séquence relatif au cas d’utilisation « appeler les numéros d’urgence » :

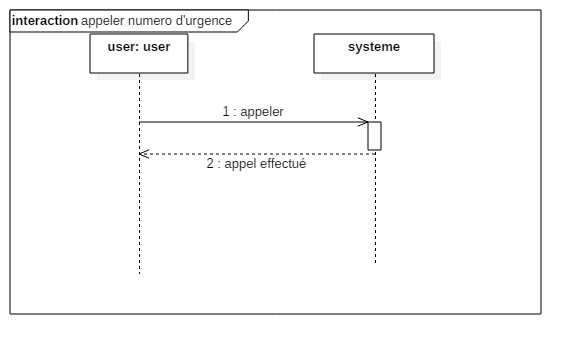


Figure 10 : diagramme de séquence du cas d’utilisation « appeler numéro d’urgence »

### Cas d’utilisation "afficher la liste de instituts d’urgences médicales"

Le diagramme de séquences de ce cas d’utilisation est illustré dans la figure ci-dessous :

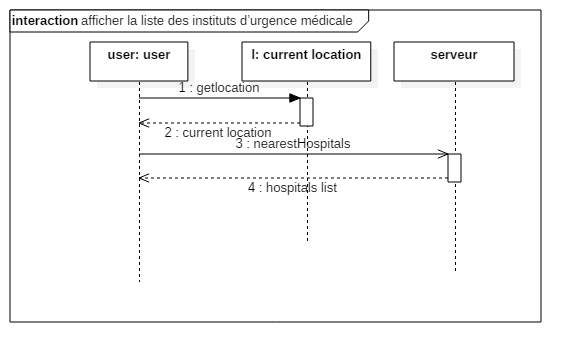


Figure 11:Diagramme de cas d'utilisation «afficher la liste des instituts d'urgence »

### Cas d’utilisation " afficher la liste des postes de police"

La figure suivante présente le diagramme de séquence qui illustre ce cas d’utilisation.

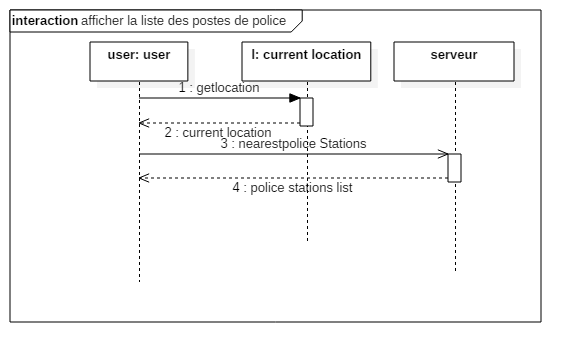


Figure 12Diagramme de cas d'utilisation «afficher la liste des postes de police»

## Cas d’utilisation " gérer les contacts"

Le diagramme complet des classes de cette application est présenté par la figure 13 suivante.

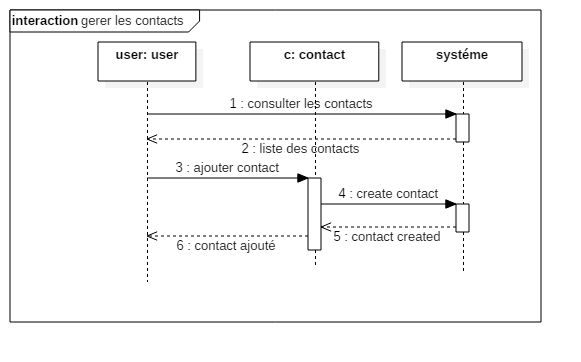


Figure 13:Diagramme de cas d'utilisation «gérer les contacts »

## Conclusion

Tout le long de ce chapitre nous avons rassemblé les différents éléments de modélisation de notre application pour réaliser la conception. La partie réalisation est décrite dans le chapitre suivant.

# Chapitre4 : Réalisation

## Introduction

Ce chapitre représente le dernier volet de ce rapport, il sera consacré à l’exposition du travail réalisé. Je commence par présenter l’environnement de travail et la justification des choix techniques concernant les plateformes de développement et de modélisation utilisées. Par la suite je vais exposer l’interface homme machine réalisée et les fonctionnalités de l’application.



## Environnement de travail

Pour réaliser les différentes étapes de ce projet j’ai utilisé l’environnement matériel et logiciel décrit dans la partie suivante.

### Environnement matériel

Je dispose d’un ordinateur " acer " avec les caractéristiques suivantes :

* Un processeur Intel i3 2.30 GHz
* Une mémoire vive 4 Go
* Un disque dur de 500 Go
* Un système d’exploitation Microsoft Windows7

### Environnement logiciel

L’environnement logiciel de développement est composé de différents outils de développement et quelques APIs fournies par Google

* **Logiciel de développement**

J’ai utilisé comme logiciel de développement pour cette application " Visual studio 2017" [6] qui est un éditeur de code très puissant.

* **Outils de développement**
* Xamarin Forms
* **Base de données**
* SQLite [7]
* **Interfaces de programmation applicative**
* Google Maps API
* Google Places API

## Description des interfaces

Afin de présenter les outils utilisés pour développer cette application j’explique son fonctionnement à l’aide des interfaces suivantes.

### Interface principale avant localisation

L’interface principale s’affiche dès que l’application est lancée. Elle permet à l’utilisateur d’accéder le service de localisation en cliquant sur le bouton «Get location » elle lui permet aussi d’accéder à la liste des contacts d’urgences en cliquant sur le bouton « emergency contacts ». De plus l’utilisateur trouve dans cette interface un lot de trois boutons sous forme d’images lui permettant d’effectuer des appels vers les numéros d’urgences :

-« 197 » appeler la police de secours

-« 198 » appeler la protection civile

-« 190 » appeler le SAMU

. Cette interface est décrite par la figure 14 suivante.

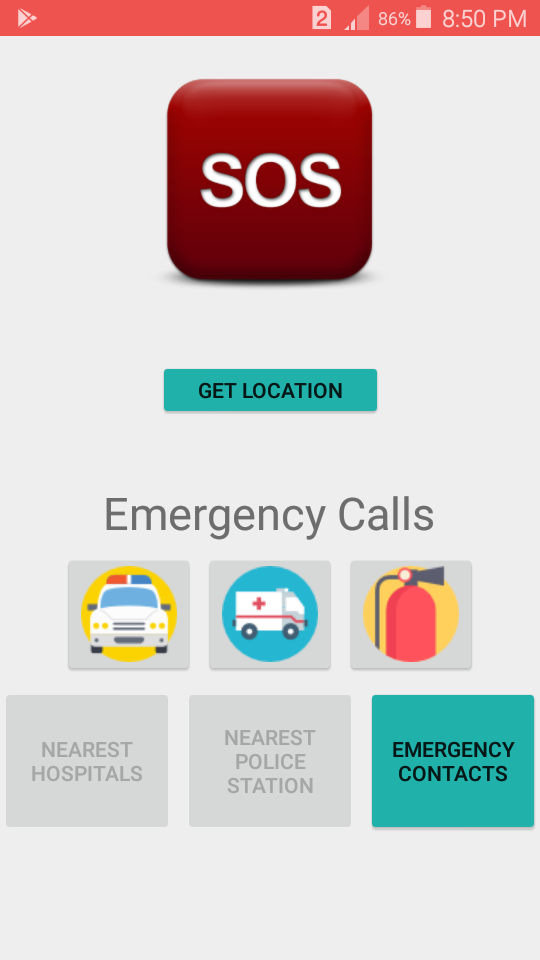


Figure 14:Interface principale avant localisation

Le service de localisation nécessite une connexion internet et l’activation du GPS sur le téléphone, ces deux conditions sont indiquées à l’utilisateur à travers deux alertes.

-**Alerte de connectivité à internet :**

C’est un pop-up déclenché si l’utilisateur essaye d’utiliser le bouton « Get location »sans être connecté à internet

En cliquant sur le bouton « connect » de l’alerte le WIFI sera activé automatiquement et en cliquant sur le bouton « cancel » l’opération sera annulée.

Cette alerte est présentée dans la figure 15 suivante :

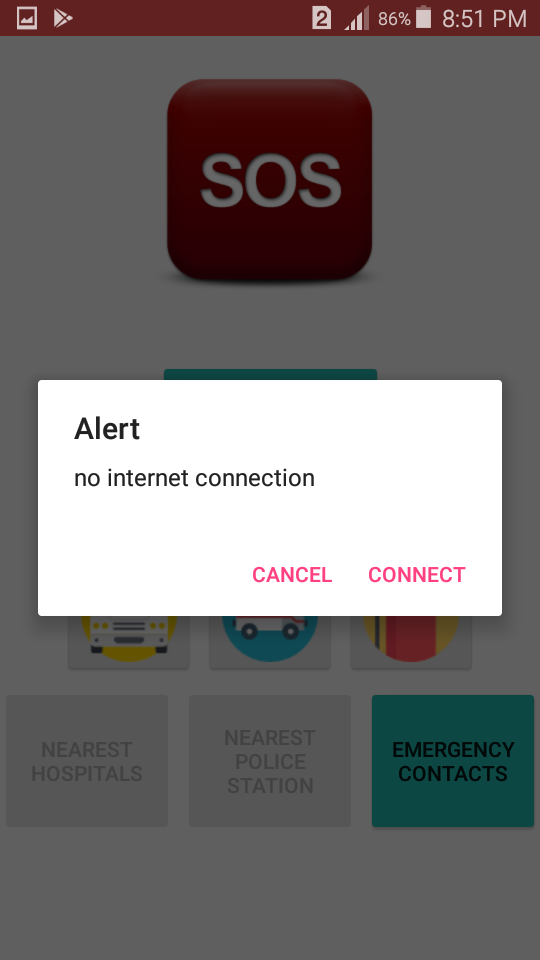


Figure 15:interface de l'alerte de connexion

-**Alerte d’activation de la localisation :**

C’est un pop-up déclenché suite à l’utilisation du bouton « Get location »sans avoir activé le GPS d’abord, et au cas d’absence de connexion internet, cette alerte sera affiché suite à celle de connectivité.

Cette alerte est présentée dans la figure16 suivante :

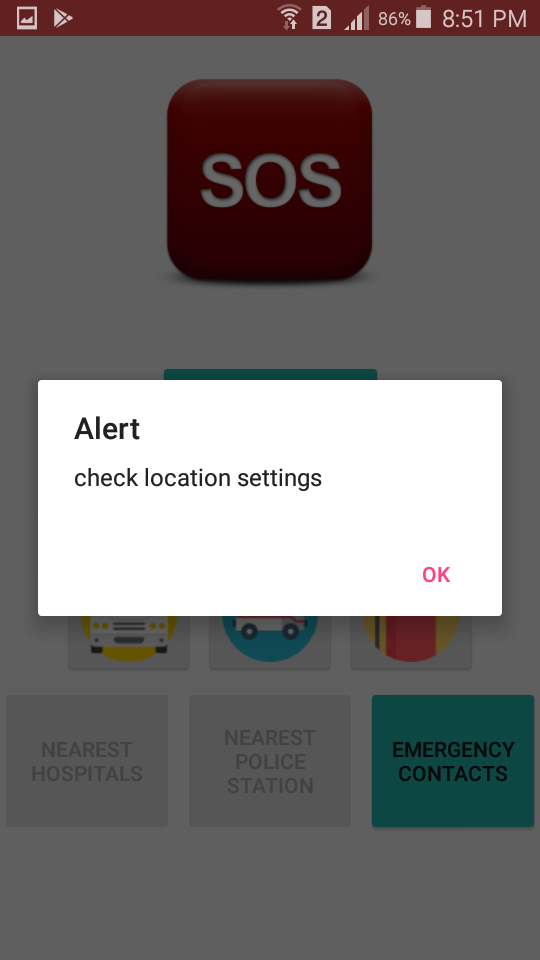


Figure 16:interface de l'alerte de GPS

### Interface principale après la localisation

Après la localisation, l’utilisateur reste sur la même interface qui précède mais, il aura accès à l’utilisation des deux boutons « nearest hospitals »et « nearest police station » et en plus un nouveau bouton « show map » apparaitra comme l’en voit dans la figure 17 ci-dessous :

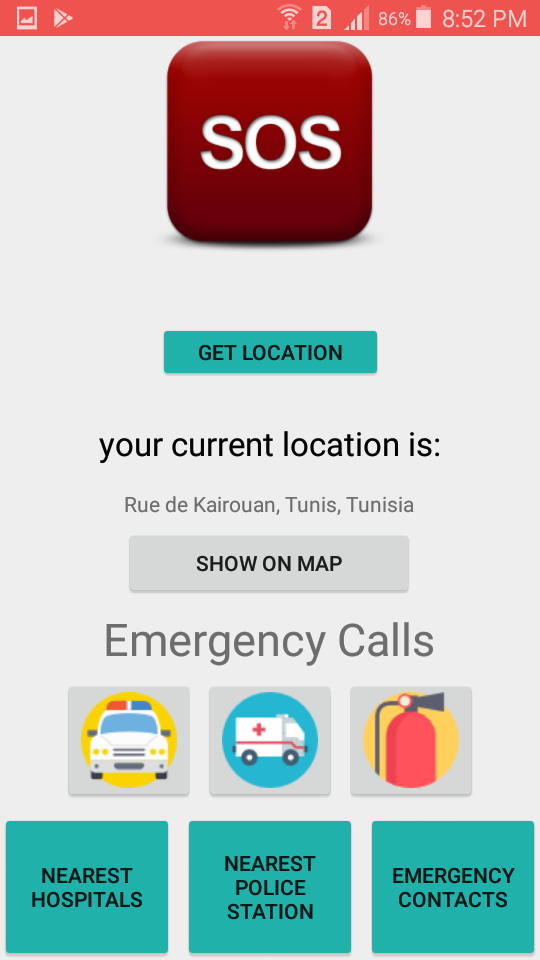


Figure 17:Interface principale après la localisation

### Interfaces d’affichage de la carte routière et la vue par satellite

En cliquant sur le bouton « show map » l’utilisateur aura accès à deux interfaces qui affichent respectivement son emplacement sur une carte routière et sur une vue par satellite

Les illustrations de la figure suivante exposent ces deux interfaces :

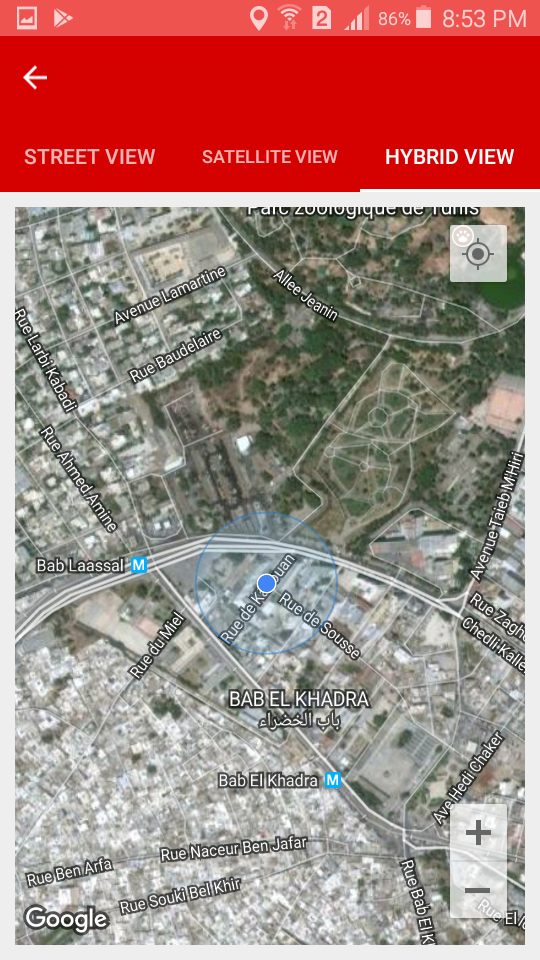
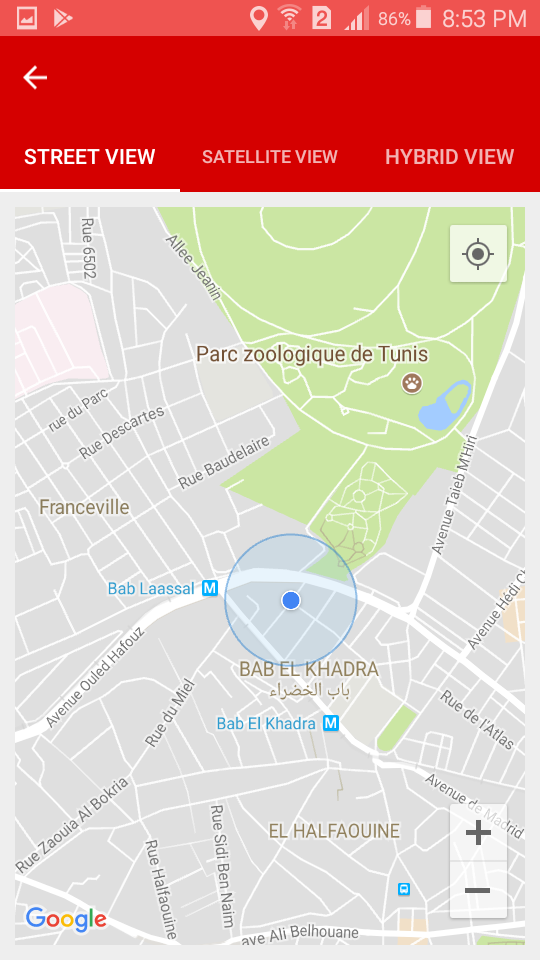


Figure 18:Interfaces d’affichage de la carte routière et la vue par satellite

### Interface d’affichage de la liste des instituts d’urgence médicale

En cliquant sur le bouton « nearest hospitals », une nouvelle interface s’affiche, contenant les noms et les adresses des instituts d’urgence médicale les plus proches de l’emplacement courant de l’utilisateur.

La liste affichera dix instituts au maximum triées à l’ordre croissant par la distance entre chaque institut et l’emplacement de l’utilisateur

Et vue que le rayon de recherche dans l’API de Google maps possède une limite implicite qui ne dépasse pas cinquante kilomètres, il se peut que la liste contienne moins que dix instituts .Cette interface est présentée dans la figure 19 :

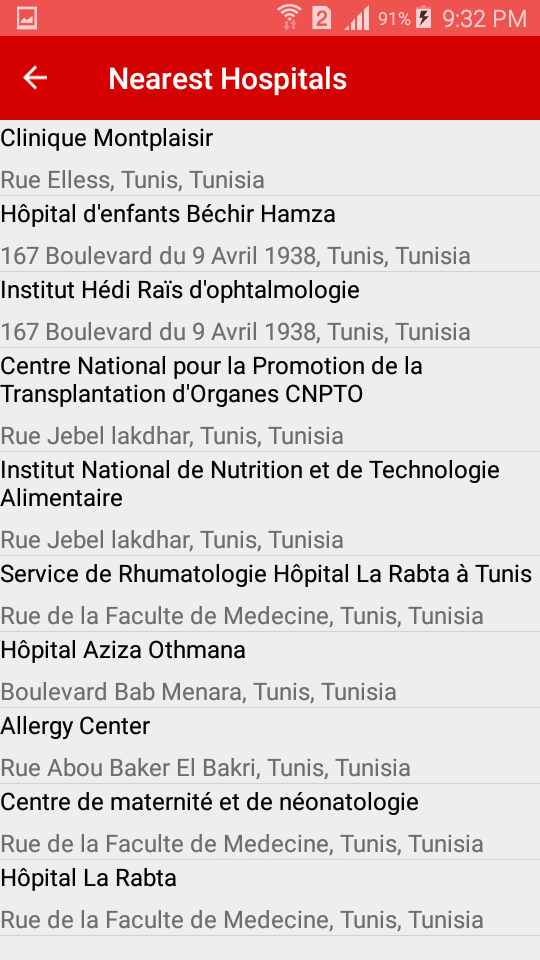


Figure 19:Interface d’affichage de la liste des instituts d’urgence médicale

En cliquant sur l’un des instituts présents dans la liste l’utilisateur aura le droit de choisir d’utiliser l’application Google Maps installée sur le Smartphone ou le navigateur par défaut afin d’avoir la route entre son emplacement et l’institut choisi.

Cette étape est indiquée dans l’illustration 20 suivante :

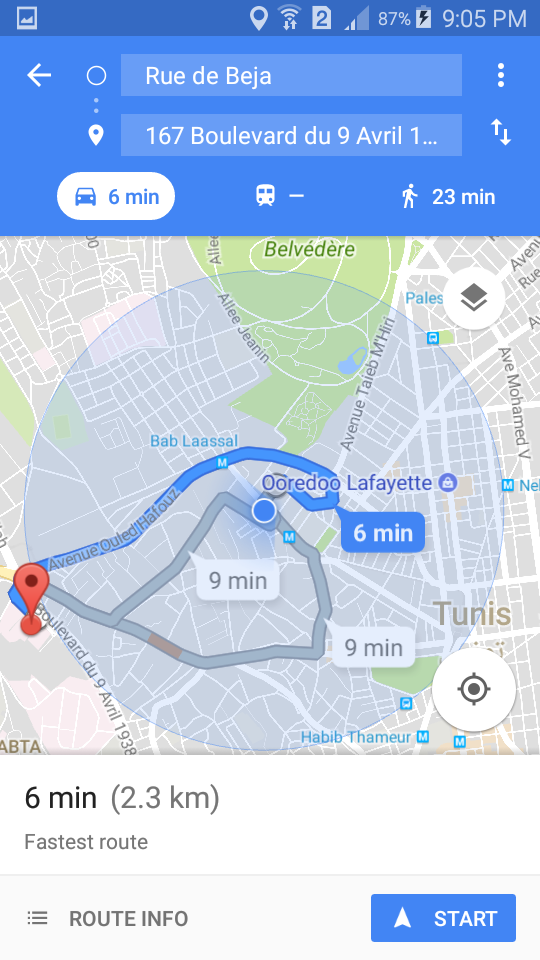
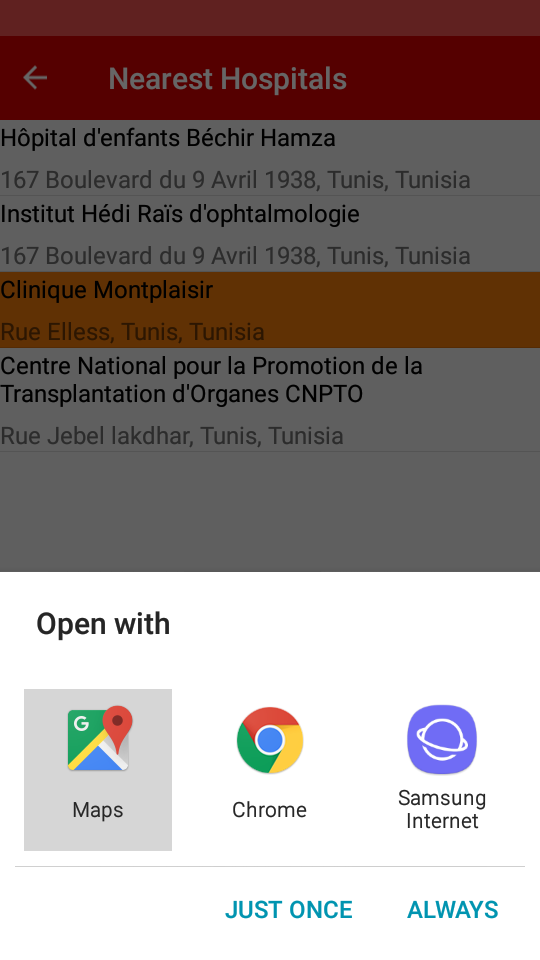


Figure 20:interface de l'itinéraire

### Interface d’affichage de la liste des postes de police

Cette interface suit le même concept que celle de la liste des instituts d’urgence médicale et offre les mêmes fonctionnalités. Voici la figure 25 qui représente cette interface :



Figure 21:Interface d’affichage de la liste des postes de police

### Interface de la liste des contacts d’urgence :

Cette interface est accédée en cliquant sur le bouton « emergency contacts »dans l’interface principale, contrairement aux interfaces précédentes cette liste affiche des données enregistrés dans une base de données locale et ne nécessite ni une connexion internet ni le GPS .Donc elle est accessible dés le lancement de l’application.

Ci-dessous la figure 26 qui décrit cette interface

L’utilisateur peut appeler chaque contact appartenant à la liste en cliquant sur l’élément correspondant.il peut aussi effacer le contact en cliquant sur le bouton « X » situé sur la ligne contenant ce contact.il peut aussi ajouter des contacts en cliquant sur le bouton flottant « + » est saisir ses données à partir du pop-up qui apparait dans la figure 27 suivante :

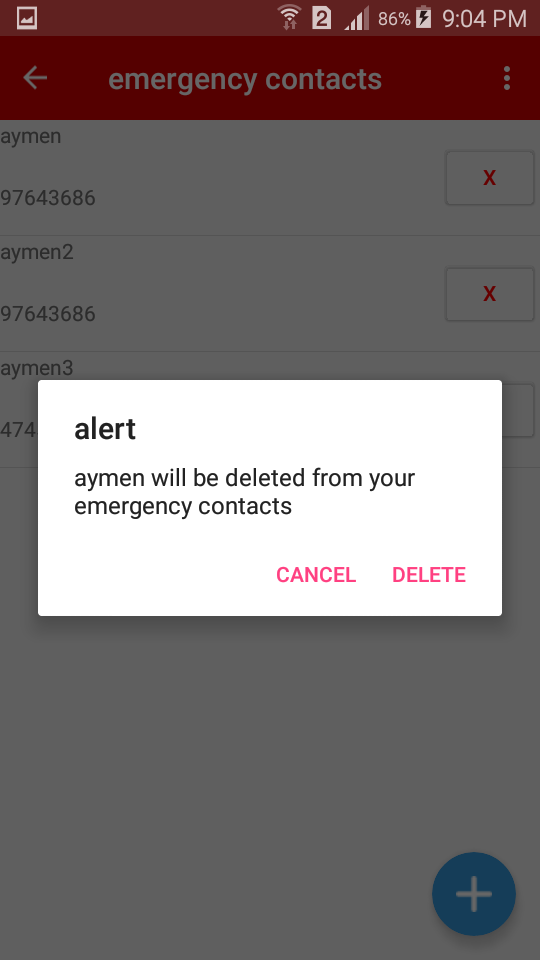
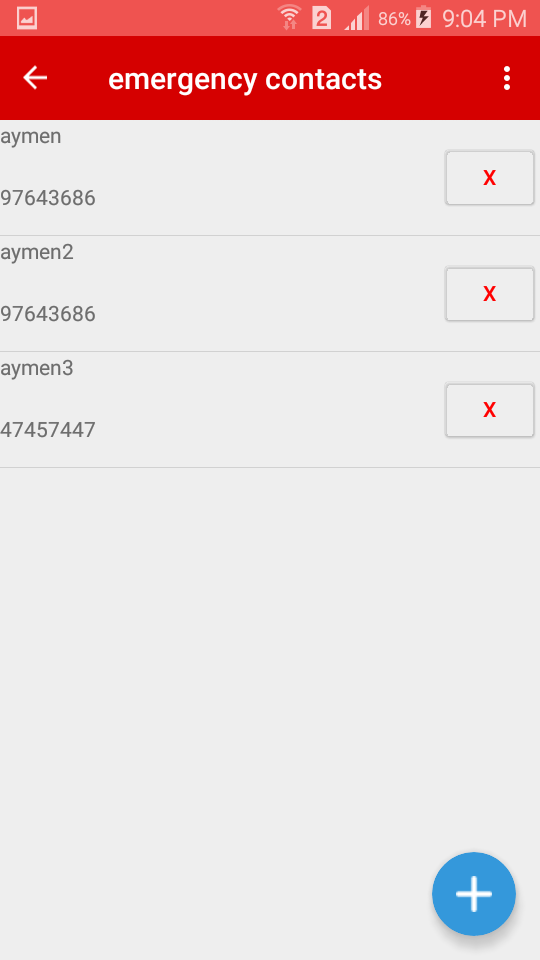


Figure 22:Interface de la liste des contacts d’urgence

Cette interface contient aussi un menu en haut permettant de rafraichir la liste ou d’effacer la liste entièrement comme le montre la figure28 :

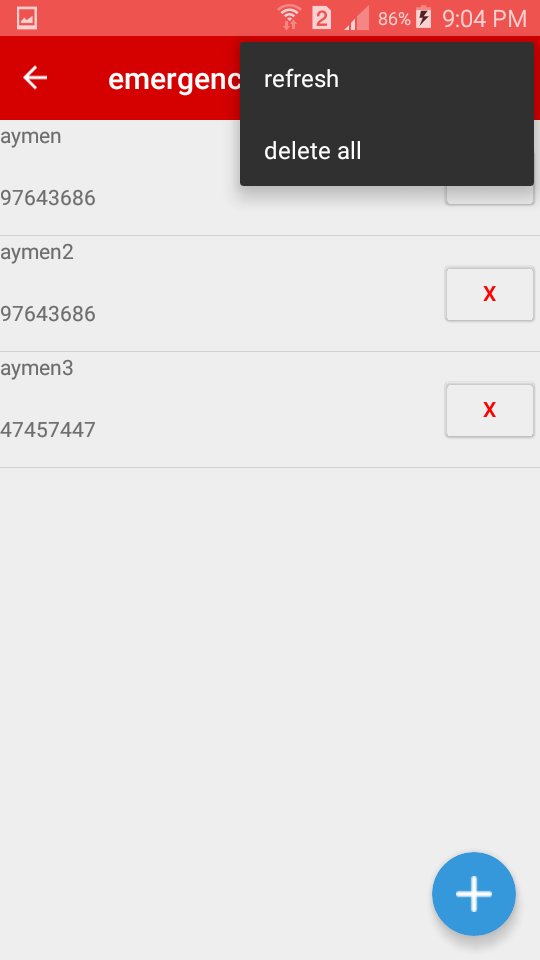


Figure 23 : interface du menu de la liste de contacts

## Conclusion

Dans ce chapitre, j’ai présenté mon environnement matériel et logiciel ainsi que les outils utilisés pour réaliser ce projet ensuite, j’ai exposé quelques interfaces de mon application.

# Conclusion et perspectives

Dans le cadre du stage d’été j’ai réalisé une application mobile de SOS emergency qui offrira l’aide à un utilisateur en danger médicale ou civile

Ce projet a été une opportunité pour apprendre à travailler sur une nouvelle plateforme de développement mobile CrossPlatform « Xamarin Forms » et appliquer mes connaissances déjà acquises dans le domaine du développement mobile. J’ai pu accomplir les objectifs fixés, mais, comme tout projet, on ne peut pas prétendre la perfection. Mon travail mérite alors quelques extensions. Ainsi pour améliorer cette application je peux appliquer un meilleur design et faire des modifications au niveau de la langue du contenu pour pouvoir viser un plus grand nombre d’utilisateurs ainsi je suis entrain de travailler pour rendre l’application utilisable à l’échelle internationale en fournissant les numéros d’urgences pour chaque pays. Je suis actuellement entrain de développer les fonctionnalités de l’application et j’essaye de l’améliorer afin de l’ajouter sur Play Store.

Au cours de ce stage j’ai développé aussi un site web vitrine pour un Youtuber que je vais vous le présenter pendant la soutenance si vous me le permettrai.

Enfin, je souhaite que ce modeste travail apporte satisfaction aux membres du Jury et aux personnes intéressées par notre application.

# Bibliographie

[1] UML

<http://www.commentcamarche.net/contents/1141-introduction-a-uml>

Consulté le 1 septembre 2017

[2] JSON

<http://www.json.org/json-fr.html>

Consulté le 1 septembre 2017

[3] Google Maps API

<https://developers.google.com/maps/?hl=fr>

Consulté le 15 septembre 2017

[4] Google places API

<https://developers.google.com/places/?hl=fr>

Consulté le 15 septembre 2017

[5] Xamarin Forms

<https://developer.xamarin.com/>

Consulté le 15 septembre 2017

[6] Visual studio 2017

<http://rdonfack.developpez.com/tutoriels/dotnet/les-nouveautes-de-visual-studio-2017/>

Consulté le 15 septembre 2017

[7] SQLite

<http://www.journaldunet.com/solutions/pratique/dictionnaire-du-webmastering/technologies-langages/19546/sqlite-definition.html>

Consulté le 15 septembre 2017

**Résumé**

Dans le cadre de ce projet j’ai développé une application mobile qui sert à aider l’utilisateur à se débrouiller aux cas d’urgences ,effectuer les appels nécessaires , découvrir facilement son entourage et son emplacement et chercher facilement les établissements d’urgence médicale et les postes de police. J’ai commencé par la détermination des besoins, ensuite l’identification des données et les traitements qui doivent constituer cette application qui est développée en utilisant Xamarin Forms , les APIs de google et SQLite .

**Mots clés**

Programmation mobile, Xamarin Forms, Google Maps API, Google places API,SQLite.

**Summary**

As part of this project, I have developed a cross platform mobile application which can be very useful in emergency cases because it provides the user with emergency numbers, a location service using GPS and afford a list of closest hospitals and police stations to the user’s location. The work began with the determination of the needs then identification of the data and the treatments that must constitute this application which is developed using Xamarin Forms, Google Maps API, Google places API, SQLite.

.

**Key words**

Xamarin Forms, Google Maps API, Google places API, SQLite, mobile development

**ملخص**

في إطار هذا المشروع قمنا بتطوير تطبيق هاتف محمول يستعمل في حالة طوارئ . بدأ عملي   بتحديد الاحتياجات ثم تحديد البيانات والعلاجات التي يجب أن تشكل هذا التطبيق الذي تم تطويره باستخدام Xamarin Forms , google Maps API Google places API, SQLite

.

**الكلمات الرئيسية:**

Xamarin Forms, Google Maps API, Google places API, SQLite, mobile development