



PROJET DE FIN D'ÉTUDES

En vue d'obtention du
Diplôme de Licence appliquée en Technologie de l'informatique

Spécialité :
Multimédia Et Développement Web

Conception et implémentation d'une Application
Android mobile de calcul de temps perdu et
gestion de temps d'attente pour le poste tunisien.

Encadré par :
Mme. Hanen Issaoui

Réalisé par :
Nayeb Barhoumi

Dédicaces

J'ai l'honneur, et le grand plaisir de dédier ce travail

*À la lumière de mon cœur qui m'a toujours tenu la main avec tellement
de tendresse et de compréhension*

*Mon très cher père **Ajimi Barhoumi***

*À la fleur de ma vie, source d'amour et d'espoir qui n'a cessé de
sacrifier pour mon bonheur*

*Ma très chère mère **Zina Hmidi***

*À mes chères sœurs **Amal Imen, Dhouha** et leurs maris*

Lotfi et Mohamed

*Et A mes chères frère **Nader, Rayen** et toute la famille*

*De **Ajimi Barhoumi***

*En les remerciant de ce qu'ils ont fait pour moi, leurs encouragements et
leurs aides, sans*

Cesse renouvelée et en leur exprimant mon profond attachement.

À tous mes amis

*Pour l'amitié, leurs aides, leurs encouragements continus, nos souvenirs
inoubliables...*

À tous mes amis et tous ceux qui j'aime et qui m'aiment.

Nayeb



Remerciements

Tout d'abord je remercie **ALLAH**, le très savant,
L'infiniment sage, à qui je dois tout.

Ensuite, je souhaite remercier vivement mon encadreur
Madame Hanen Issaoui pour sa rigueur scientifique,
ses critiques constructives, ses idées stimulantes
et ses conseils judicieux.

Je vous suis profondément reconnaissante pour votre implication
ainsi que la liberté et la confiance
qu'elle m'a accordée durant l'élaboration de mon projet.
Veuillez trouver ici l'expression de notre gratitude
et de notre profonde estime.

A tous les membres de Jury qui ont bien voulu sacrifier
un peu de leur temps pour apprécier ce travail.

Table des matières

Liste de figures	6
Listes des tableaux	7
Introduction générale	1
Chapitre 1 : Analyses et Spécifications des Besoins	2
Introduction	3
1.Présentation du projet.....	3
1.1. Cadre du projet	3
1.2. Contexte du projet	3
2.Étude de l'existant.....	4
2.1. Description de l'existant	4
2.2. Critique de l'existant.....	5
2.3. Solution proposée	5
3.Spécifications des besoins	6
3.1. Besoins fonctionnels	6
3.2. Besoins non fonctionnels	7
Conclusion.....	7
Chapitre 2 : La conception	10
Introduction	10
1.Langage de conception	10
1.1. Langage UML	10
1.2. Utilisation de l'UML	11
2. Les diagramme de cas d'utilisation	11
2.1. Les Acteurs	11
2.2. Diagramme de cas d'utilisation.....	12
2.2.1. Diagramme de cas d'utilisation global pour l'application mobile.....	12
2.2.2. Diagramme de cas d'utilisation inscription.....	13
2.2.3. Diagramme de cas d'utilisation Authentification :	14
2.2.4. Diagramme de cas d'utilisation recherché sur le maps :	14
2.2.5. Diagramme de cas d'utilisation réserver un ticket :	15
2.2.6. Cas d'utilisation « consulter les historiques » :	15
2.2.7. Description Textuelle.....	15
3. Diagrammes des séquences	16
3.1. Diagramme de séquence authentification	16
3.2. Diagramme de séquence crée un compte.....	17
3.3. Diagramme de séquence consulter l'historique.....	19

4. Diagramme de classe.....	19
Conclusion.....	20
Chapitre 3 : Réalisation	20
Introduction	21
1. Environnement de travail.....	21
1.1. Environnement matériel.....	21
1.2. Environnement logiciel.....	22
1.3. Langage de programmation et technologies utilisés	23
2. Plateforme Android.....	24
2.1. Historique	24
2.2. Architecture Android	24
3. Présentation des interfaces de l'application.....	25
3.1. Les interfaces de l'application web	25
3.1.1. Interface d'authentification	26
3.1.2. Interface création d'un compte	26
3.1.3. Interface restart mot de passe	27
3.1.4. Interface page d'accueil	27
3.1.5. Interface réservation d'un ticket	28
3.1.6. Interfaces de consulter les historiques.....	29
Conclusion.....	29
Conclusion générale.....	30
Bibliographie	31
Netographie	32

Liste de figures

Figure 1 : la poste actuellement	4
Figure 2 : présentation de la solution proposée.....	6
Figure 3 : Diagramme de cas d'utilisation global pour l'application mobile	13
Figure 4 : Diagramme de cas d'utilisation inscription	13
Figure 5 : diagramme de cas d'utilisation authentification.....	14
Figure 6 : Diagramme de cas d'utilisation chercher poste.....	14
Figure 7: Diagramme de cas d'utilisation réserver un ticket	15
Figure 8: Diagramme de cas d'utilisation consulter l'historique	15
Figure 9 : diagramme de séquence « authentification »	17
Figure 10 : diagramme de séquence crée compte	18
Figure 11 : Diagramme de séquence consulter l'historique.....	19
Figure 12 : Diagramme de classe	20
Figure 13 : Architecture générale Android	24
Figure 14 : Interface d'authentification.....	26
Figure 15 : Interface création d'un compte.....	26
Figure 16 : Interface restart mot de passe	27
Figure 17 : Interface page d'accueil	27
Figure 18 : Interfaces reservation d'un ticket	28
Figure 19 : Interfaces gérer les endroits	29

Listes des tableaux

Tableau 1 : Descriptions les acteurs	12
Tableau 2: Description textuelle de « Créer compte »	16
Tableau 3: Environnement matériel	21

Introduction générale

Dans un monde actif et continuellement évolutif, la motivation d'avoir des moyens performants et efficaces de communication et d'échange d'informations devient de plus en plus fondamentale. Cette motivation donne naissance à une révolution favorisant le travail à distance et l'accès aux besoins en temps réduit à l'aide d'internet qui a bouleversée les habitudes de travail dans de nombreux métiers.

D'après beaucoup d'analyses et statistiques effectuées, il s'avère que de plus en plus d'internautes se connectent désormais à internet via leurs téléphones portables. Nous remarquons ces dernières années un développement exponentiel des appareils mobiles qui sont répandus comme une traînée de poudre dans le monde en développement et révolutionnant le domaine des communications notamment dans les zones rurales.

L'environnement actuel évolue, l'innovation ne s'arrête Ajouter les technologies

Progressent, les moyens de production se transforment, la concurrence s'accroît, cela

Oblige toute entité économique à faire évoluer son activité à un rythme de plus en plus rapide.

Avant l'invention de l'ordinateur, nous enregistrons toutes les informations manuellement sur des supports en papier. Ce qui engendrait beaucoup de problèmes tel que la perte de temps considérable dans la recherche de ces informations ou la dégradation de ces dernières.

En outre, l'économie Tunisienne est devenue une économie ouverte sur le monde extérieur, puisque le transport suit et agit dans cet environnement, il est appelé à améliorer ses encadrements techniques, à moderniser ses services, et à être au diapason des progrès technologiques.

Dans ce contexte nous nous intéressons aux problèmes que rencontrent les pertes de temps dans la poste tunisienne et la méthode quotidiennement au niveau des services clientèles. En effet, implémentation d'une application Android de calcul de temps perdu en ligne implique de nombreux problèmes, tels que la perte du temps, le retard...etc.

Ce projet est réalisé dans le cadre de notre Projet de Fin d'Études à l'Institut Supérieur des Études Technologiques (ISET) de Gafsa et au sein de la société d'ISET Gafsa pour l'obtention de la licence appliquée en Technologie de l'informatique.

Pour une meilleure accessibilité à ces informations, nous avons décidé de développer une application mobile capable de gérer et d'exposer toutes ses informations et de les synchroniser.

En effet, les applications mobiles ne cessent d'enregistrer une croissance de plus en plus importante, qui sont facilitent la vie des internautes. Par conséquent, il doit y avoir une montée rapide en compétences afin de s'adapter aux nouveaux besoins, et développer des solutions intégrant plusieurs composantes mobiles, pour des applications mobiles sur divers plateforme comme : IOS, Windows Phone, Android etc.

Ce rapport détaillera les différentes phases dont nous sommes passées par afin d'aboutir à une application fiable et satisfaisante. Pour cela le rapport définit le travail qui nous avons effectué, il est composé par trois grands chapitre. Le premier chapitre pour présente l'analyse et spécification des besoins, le deuxième chapitre est consacré à la conception. Le dernier chapitre comporte les détails de réalisation de notre application.

Chapitre 1 : Analyses et Spécifications des Besoins

Chapitre 1 : Analyses et Spécifications des Besoins

Introduction

Dans ce chapitre, nous allons faire une étude de l'existant sur les modalités des travaux actuelles. Enfin nous allons spécifier l'ensemble des besoins fonctionnels et non fonctionnels liés à notre application.

1. Présentation du projet

Dans cette section, nous d'écrirons le contexte du projet, nous poserons la problématique et nous proposons une solution ; puis nous passerons aux méthodes de conception.

1.1.Cadre du projet

Ce stage s'inscrit dans le cadre d'un projet de fin d'études pour l'obtention d'un diplôme National licence appliqué En Technologie de l'informatique De ISET Gafsa. Notre stage a été effectué au sein ISET Gafsa Le sujet est intitulé « implémentation d'une application Android de calcule de temps perdu ».

1.2.Contexte du projet

Notre projet consiste essentiellement à développer une application mobile qui facilite et automatise la gestion de temps perdu dans la poste tunisienne.

- ❖ La Poste est l'une des plus anciennes administrations de Tunisie. Sa création remonte au XIXe

Siècle. Sa tâche principale était la transmission et la distribution du courrier, les communications

Par téléphone, télégraphe et autres services financiers. Mais le rôle n'a pas cessé d'évoluer d'un

Jour à autre en prenant en considération même l'évolution sociale, économique.

Consciente de son rôle prépondérant, la poste an essaye de développer ses moyens de production et ses produits conformément aux exigences de l'époque et pour mieux se positionner sur le marché.

2. Étude de l'existant

L'étude de l'existant est une phase importante pour bien comprendre le système actuel. Il a pour objectif d'étudier et de dégager les lacunes du système existant et de proposer les solutions adéquates et définir les objectifs à atteindre au titre de perfectionnement.

Il existe deux cas lorsque l'on procède à l'étude de l'existant :

- Soit le produit existe déjà, alors il faut l'améliorer.
- Soit le produit n'existe pas, il faudra donc le créer.

Nous allons procéder par une approche mixte associant les deux procédés. Dans la première partie, nous nous focaliserons sur l'analyse de l'existant : Critique et solutions.

2.1.Description de l'existant

Nous pouvons déclarons que l'une des majeurs problèmes le problème de temps d'attente entre services : Chaque salle postale admet un nombre limité des guichets de service. Sachant que la poste tunisienne se caractérise par l'unicité des services alors le nombre des clients ne cesse à augmenter jour par jour ce pour cela ils ont adaptés l'architecture de First In First out en utilisant le nombre des tickets : Chaque client sélectionner un ticket qui admet son ordre dans la liste d'attente or on a noté que le temps gaspillé en attente en train d'augmenté jour à jour à la fois où il peut être exploité dans des autre cadre clientèle personnel.

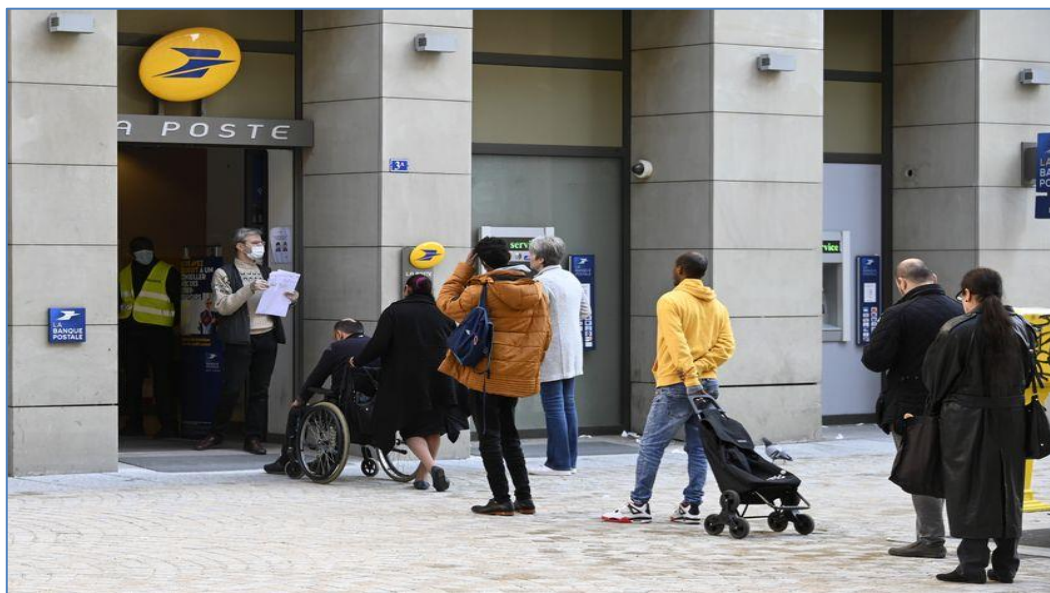


Figure 1 : la poste actuellement

2.2. Critique de l'existant

Dans le cadre de ce travail nous nous intéressons aux problèmes que rencontrent les clients de le poste tunisienne dans le processus de gestion des temps perdu et des réserver de tickets.

En effet la gestion du temps en commun « poste tunisienne » implique de nombreux problèmes comme :

- Trouver l'emplacement.
- Connaître les horaires exacts.
- Faire face à un important flux des clients et perdre son temps
- Le Problème des files d'attente au sein des bureaux de poste Perçue comme un problème ostensible.
- Mauvaise gestion d'organisation de longue : mauvaise organisation de longue permet à plusieurs problèmes de travail.
- Perte de temps : l'absence de l'un des procédures va perde beaucoup de temps pour le client.

2.3.Solution proposée

Pour résoudre les problèmes présentés dans le paragraphe précédent, nous travaillons pour le but résoudre les peines de l'attente trop longue au sein des administrations et des agences postales tunisiennes en particulier vont bientôt prendre fin.

Pour éviter ces problèmes nous proposons de concevoir et développer une application de gestion de temps en commun « poste », afin d'automatiser ses tâches et de faciliter la gestion dans ce secteur.

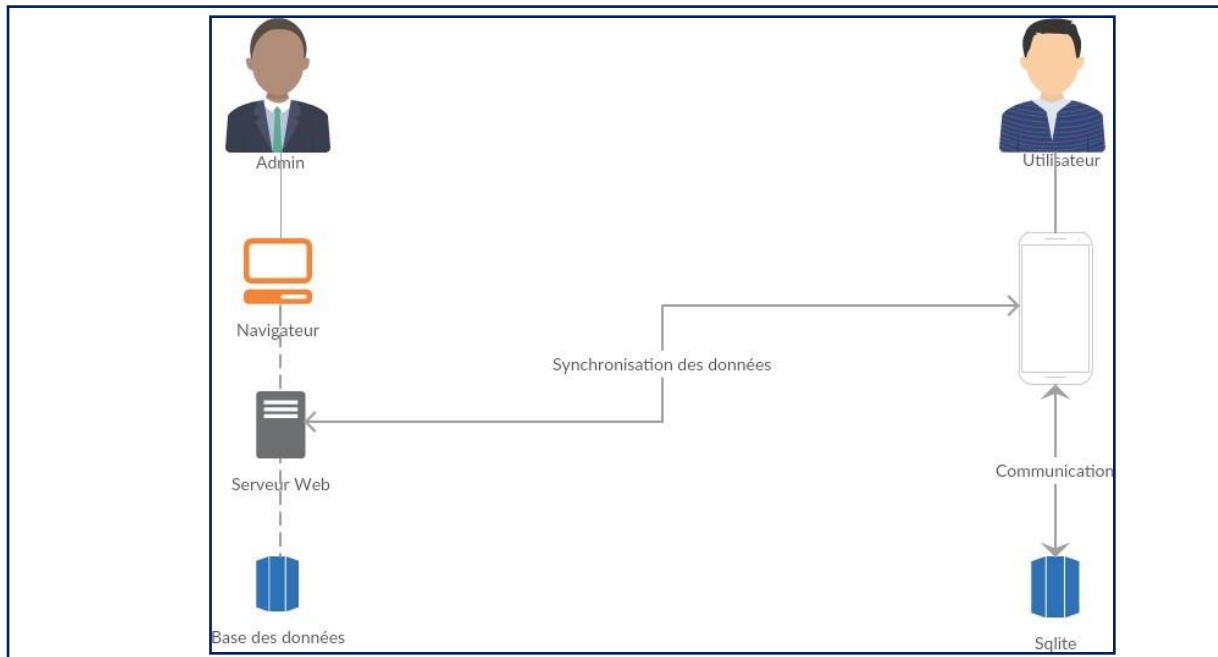


Figure 2 : présentation de la solution proposée

3. Spécifications des besoins

La spécification de besoins constitue la phase de départ de toute application à développer. Dans cette partie nous allons identifier les besoins de notre application. Nous allons mettre l'accent sur les besoins fonctionnels ainsi que les besoins non fonctionnels pour éviter le développement d'une application non satisfaisante.

3.1. Besoins fonctionnels

Un acteur joue le rôle d'une entité externe (utilisateur humain, dispositif matériel ou autre système) qui interagit directement avec le système étudié. L'acteur représenté dans notre application peut être soit un administrateur, soit un visiteur.

Les besoins fonctionnels de cette application sont les suivants :

- ✓ Réserver son ticket à distance
- ✓ Inscription
- ✓ Authentification
- ✓ Savoir si les bureaux est encombré ou pas
- ✓ Savoir la localisation des lieux des bureaux de poste les plus proches et sa

position géographique

- ✓ Connaitre la durée d'attente
- ✓ Consulter les horaires
- ✓ Consulter les services
- ✓ Consulter historique
- ✓ Gérer les réservation (consulter et accepter)

3.2.Besoins non fonctionnels

Les besoins non fonctionnels décrivent les objectifs liés aux performances du système et aux contraintes de son environnement. Ses exigences techniques sont souvent exprimées sous forme d'objectifs spécifiques que doit atteindre le système.

Les besoins non fonctionnels de cette application sont les suivants :

- ✓ Assurer une cohérence et une crédibilité aux informations stockées dans la base de données.
- ✓ L'ergonomie des interfaces de l'application mobile
- ✓ Sécurité : à développer davantage
- ✓ Qualité
- ✓ Accessibilité et performance
- ✓ Renouvellement
- ✓ Fiabilité

Conclusion

À travers ce chapitre, nous avons présenté le cadre du projet. En outre nous avons analysé et étudié les différentes applications d afin d'enrichir nos fonctionnalités et augmenter la performance de l'application pour répondre à nos besoins. Ainsi, nous nous sommes positionnés dans le contexte de notre projet, ceci va nous permettre d'entamer la prochaine étape qui consiste à présenter la phase de conception.

Chapitre 2 : Conception

Chapitre 2 : conception

Introduction

Le Modèle conceptuel de données est une représentation statique du système d'information. Il a comme objectif de constituer une représentation claire et cohérente des données manipulées dans le système d'information.

Cette section sera présentée comme suit : nous commençons par le choix de la méthodologie de conception et justification. Ensuite nous identifions les acteurs et les diagrammes des cas d'utilisation, puis nous présentons les diagrammes de séquence, enfin le diagramme de classe.

1. Langage de conception

Pour élaborer cette application on doit établir une conception modeste pour atteindre le but de notre projet pour cela on doit choisir un langage de conception adaptable avec notre besoin.

1.1.Langage UML

« UML » (en anglais Unified Modeling Language ou langage de modélisation unifié) est un langage de modélisation graphique à base de pictogrammes. Il est apparu dans le monde du génie logiciel, dans le cadre de la « conception orientée objet ».

Couramment utilisé dans les projets logiciels, il peut être appliqué à toutes sortes de systèmes ne se limitant pas au domaine informatique.

Les 14 diagrammes UML sont dépendants hiérarchiquement et se complètent, de façon à permettre la modélisation d'un projet tout au long de son cycle de vie. Ces diagrammes sont :

- **Diagrammes structurels ou statiques :**
 - Diagramme de classes
 - Diagramme d'objets
 - Diagramme de composants
 - Diagramme de déploiement
 - Diagramme des paquetages
 - Diagramme de structure composite
 - Diagramme de profils.

- **Diagrammes comportementaux :**
 - Diagramme des cas d'utilisation
 - Diagramme états-transitions
 - Diagramme d'activité.

- **Diagrammes d'interaction ou dynamiques :**
 - Diagramme de séquence
 - Diagramme de communication
 - Diagramme global d'interaction
 - Diagramme de temps.

1.2.Utilisation de l'UML

L'UML est un langage formel et normalisé en termes de modélisation objet. Son indépendance par rapport aux langages de programmation, son caractère polyvalent et sa souplesse ont fait de lui un langage universel. En plus UML est essentiellement un support de communication, qui facilite la représentation et la compréhension de solution objet. Sa notation graphique permet d'exprimer visuellement une solution objet, ce qui facilite la comparaison et l'évaluation des solutions. L'aspect de sa notation, limite l'ambiguïté et les incompréhensions.

2. Les diagramme de cas d'utilisation

2.1. Les Acteurs

Un acteur représente l'abstraction d'un rôle joué par des entités externes (utilisateur, dispositif matériel ou autre système) qui interagissent directement avec le système (réception d'information, etc.). Pour notre application, il y aura qu'un seul acteur. Nous allons nommer cet acteur tout simplement : « Utilisateur ».

Tableau 1 : Descriptions les acteurs

Acteur	Type Acteur	Descriptions
Utilisateur	Acteur principale	<ul style="list-style-type: none"> • Suivi les parcours
Administrateur		<ul style="list-style-type: none"> • Gérer les parcours
		<ul style="list-style-type: none"> • Consulte la carte géographique
Système	Acteur secondaire	<ul style="list-style-type: none"> • Garde les actions de l'utilisateur
		<ul style="list-style-type: none"> • Calcul la distance
		<ul style="list-style-type: none"> • Envoyer les notifications
Serveur google		<ul style="list-style-type: none"> • Définir la position de l'utilisateur
Map		<ul style="list-style-type: none"> • Traçage
		<ul style="list-style-type: none"> • Définir la position

2.2. Diagramme de cas d'utilisation

Chaque usage que les acteurs font du système est représenté par un cas d'utilisation. Chaque cas d'utilisation représente une fonctionnalité qui leur est offerte afin de produire le résultat attendu. Ainsi, le diagramme de cas d'utilisation décrit l'interaction entre le système et l'acteur en déterminant les besoins de l'utilisateur et tout ce que doit faire le système pour l'acteur.

2.2.1. Diagramme de cas d'utilisation global pour l'application mobile

Le diagramme illustré par la figure suivant décrit les différents besoins fonctionnels de notre application mobile chaque cas d'utilisation représente une fonctionnalité offerte par le système à ses utilisateurs.

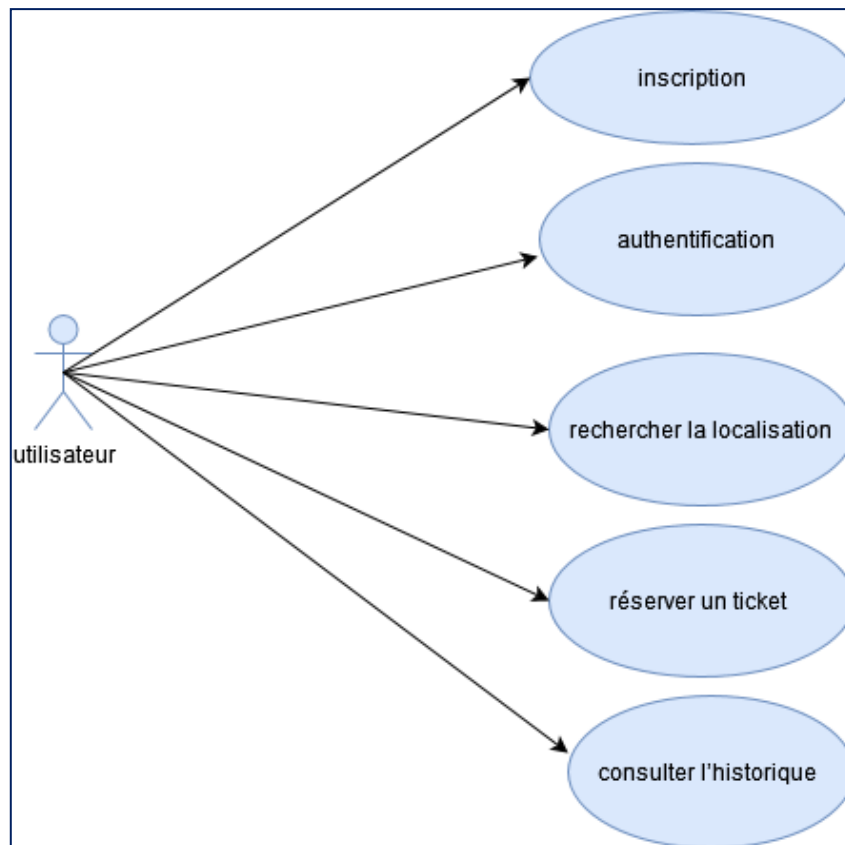


Figure 3 : Diagramme de cas d'utilisation global pour l'application mobile

2.2.2. Diagramme de cas d'utilisation inscription

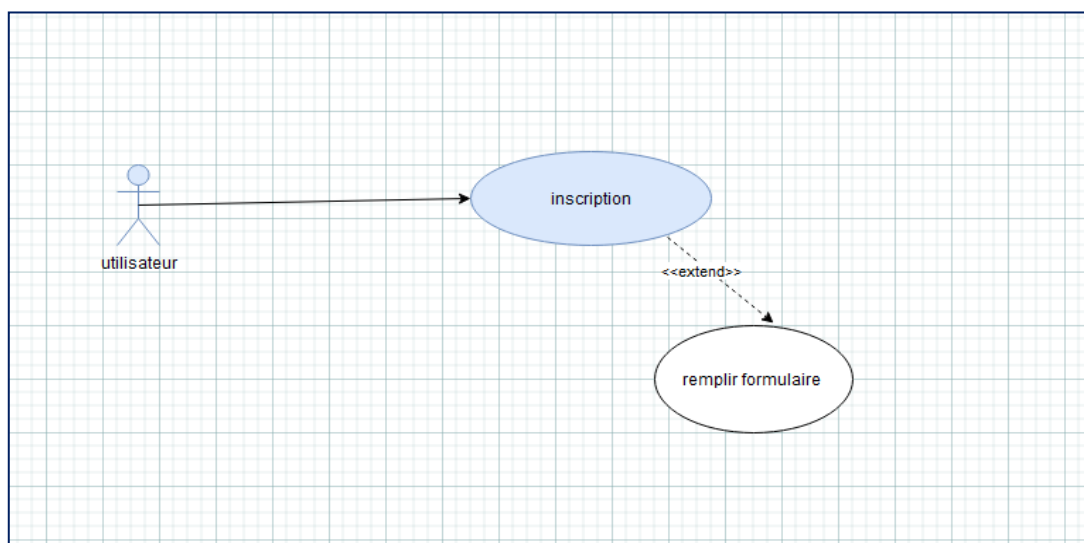


Figure 4 : Diagramme de cas d'utilisation inscription

2.2.3. Diagramme de cas d'utilisation Authentification :

La figure représente le diagramme de cas d'utilisation de l'authentification

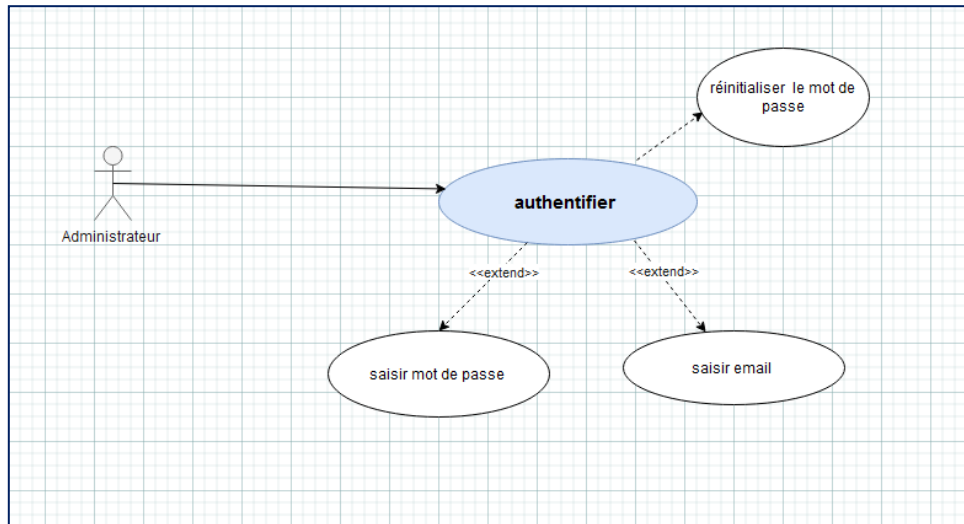


Figure 5 : diagramme de cas d'utilisation authentification

2.2.4. Diagramme de cas d'utilisation recherché sur le maps :

La figure ci-dessous représente le diagramme de cas d'utilisation qui décrit L'utilisateur doit trouver une poste

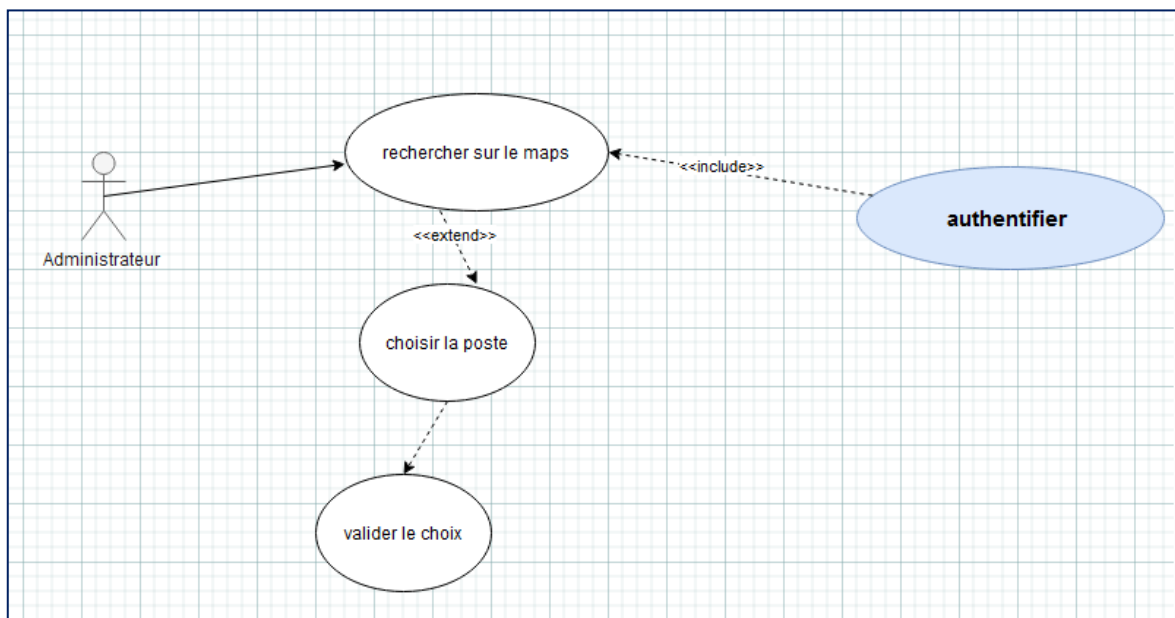


Figure 6 : Diagramme de cas d'utilisation chercher poste

2.2.5. Diagramme de cas d'utilisation réserver un ticket :

La figure représente le diagramme de cas d'utilisation de réserver un ticket

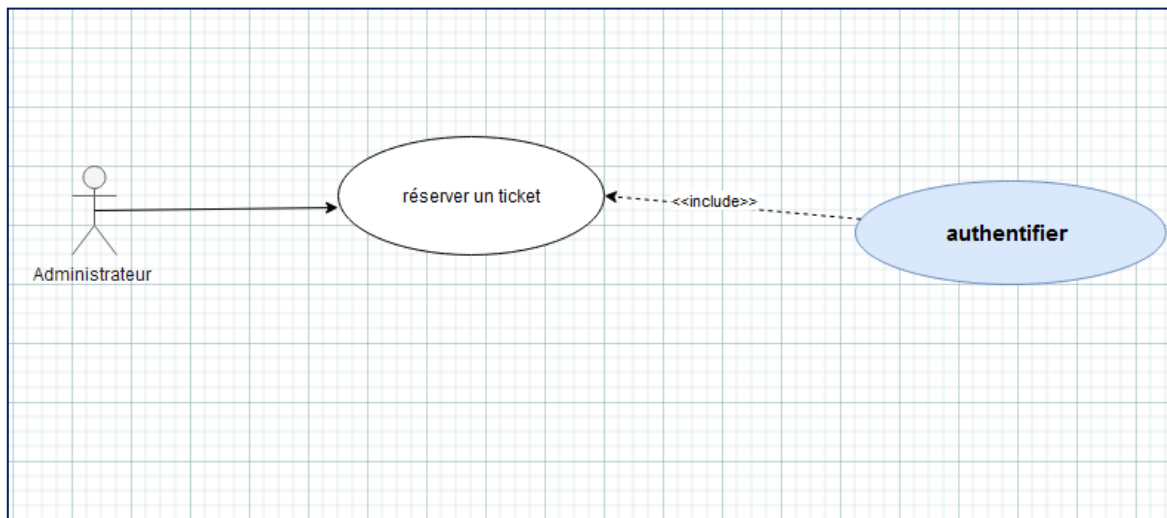


Figure 7: Diagramme de cas d'utilisation réserver un ticket

2.2.6. Cas d'utilisation « consulter les historiques » :

La figure ci-dessous représente le diagramme de cas d'utilisation qui décrit L'utilisateur doit consulter les historiques.

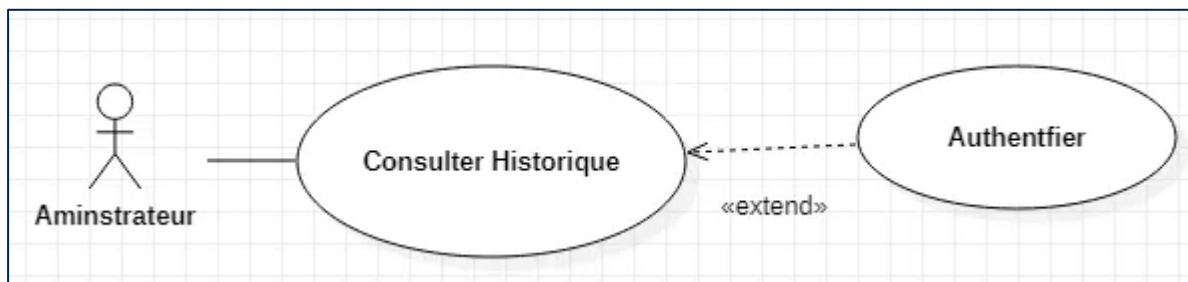


Figure 8: Diagramme de cas d'utilisation consulter l'historique

2.2.7. Description Textuelle

L'objectif de cette activité est de décrire textuellement, pour chaque Sprint, les scénarios Des cas d'utilisation. Il faut indiquer comment ce scénario démarre, comment il se termine

Et les interactions de l'utilisateur avec l'application.

Tableau 2: Description textuelle de « Créer compte »

Titre	Créer compte
Acteur	Utilisateur
Précondition	
<ul style="list-style-type: none"> ▪ Utilisateur consulte la page de création de compte. 	
Scénario nominal	
<ul style="list-style-type: none"> ▪ Utilisateur remplit le formulaire de création de compte. ▪ Utilisateur clique sur le bouton « Signe Up ». ▪ Le système vérifie la validité des données saisies. ▪ Le système enregistre le nouvel Utilisateur. ▪ Le système affiche un message de succès 	
Post-condition	Compte créé.

3. Diagrammes des séquences

Permet de décrire les interactions entre un groupe d'objets en montrant, de façon séquentielle, les envois de message qui interviennent entre les objets. Le diagramme peut également montrer les flux de données échangées lors des envois de message.

3.1. Diagramme de séquence authentification

La figure ci-dessous représente le diagramme de séquence qui décrit le processus « authentification »

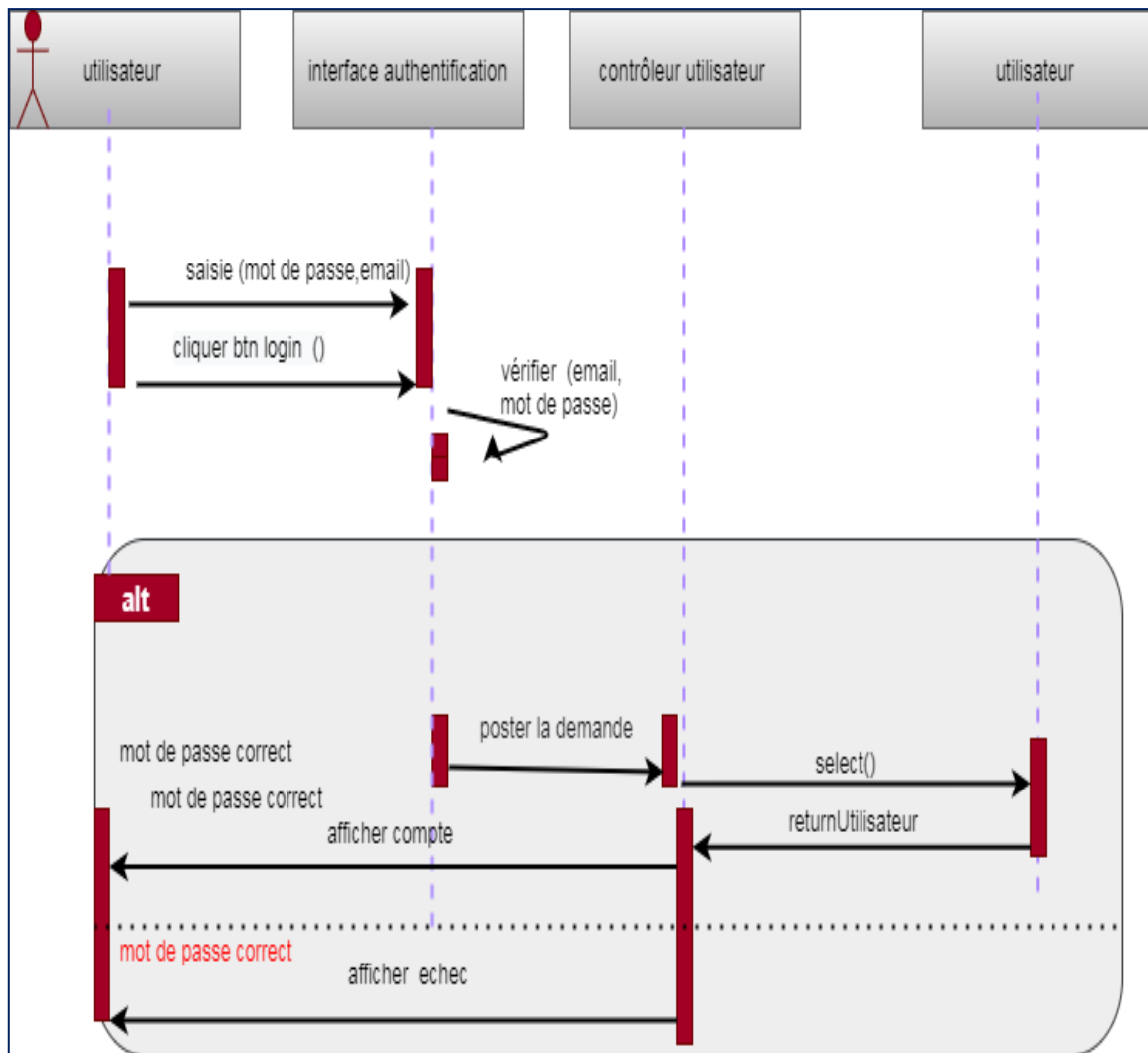


Figure 9 : diagramme de séquence « authentication »

3.2. Diagramme de séquence crée un compte

La figure ci-dessous représente le diagramme de séquence qui décrit le processus « crée compte »

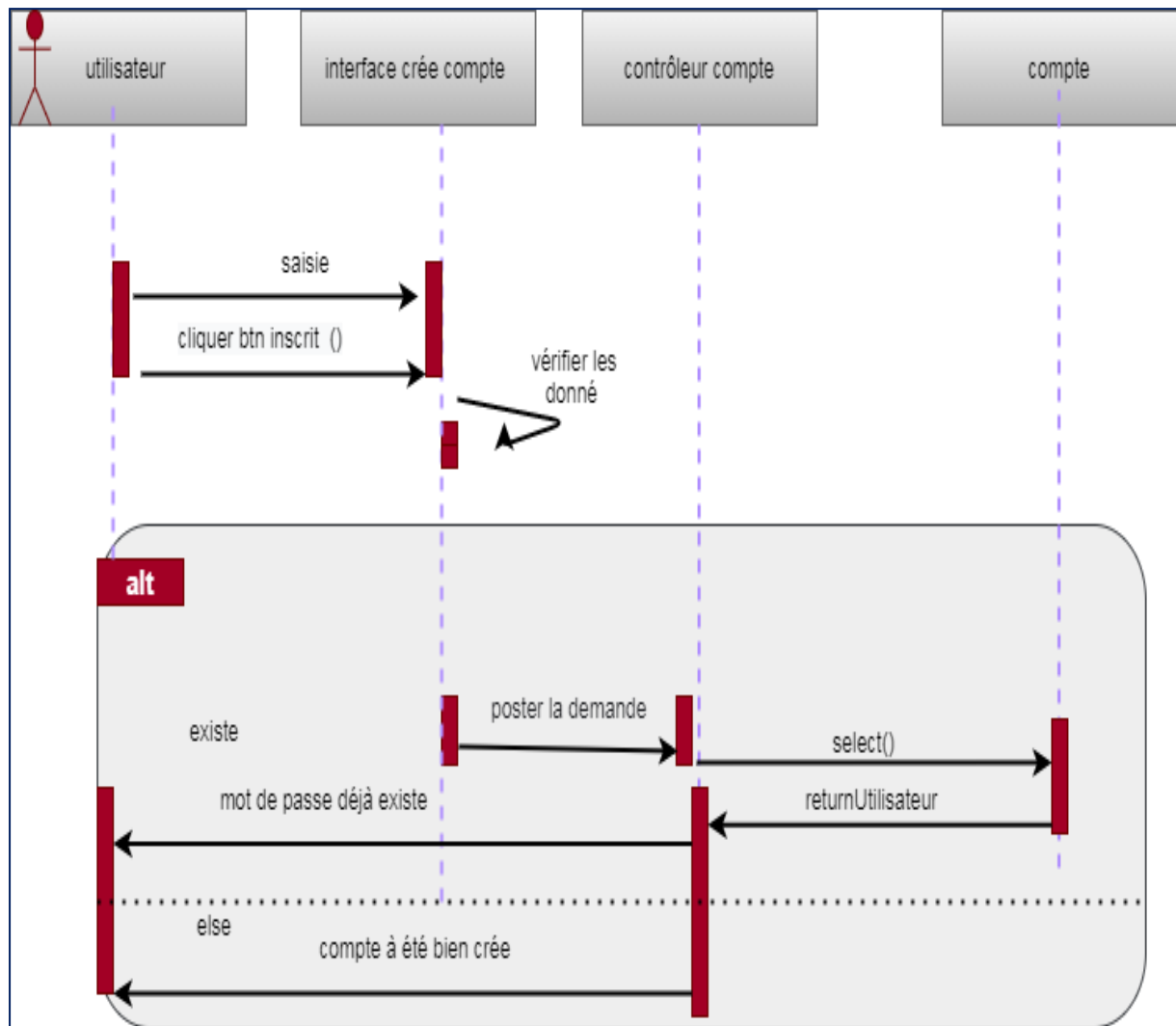


Figure 10 : diagramme de séquence crée compte

3.3. Diagramme de séquence consulter l'historique

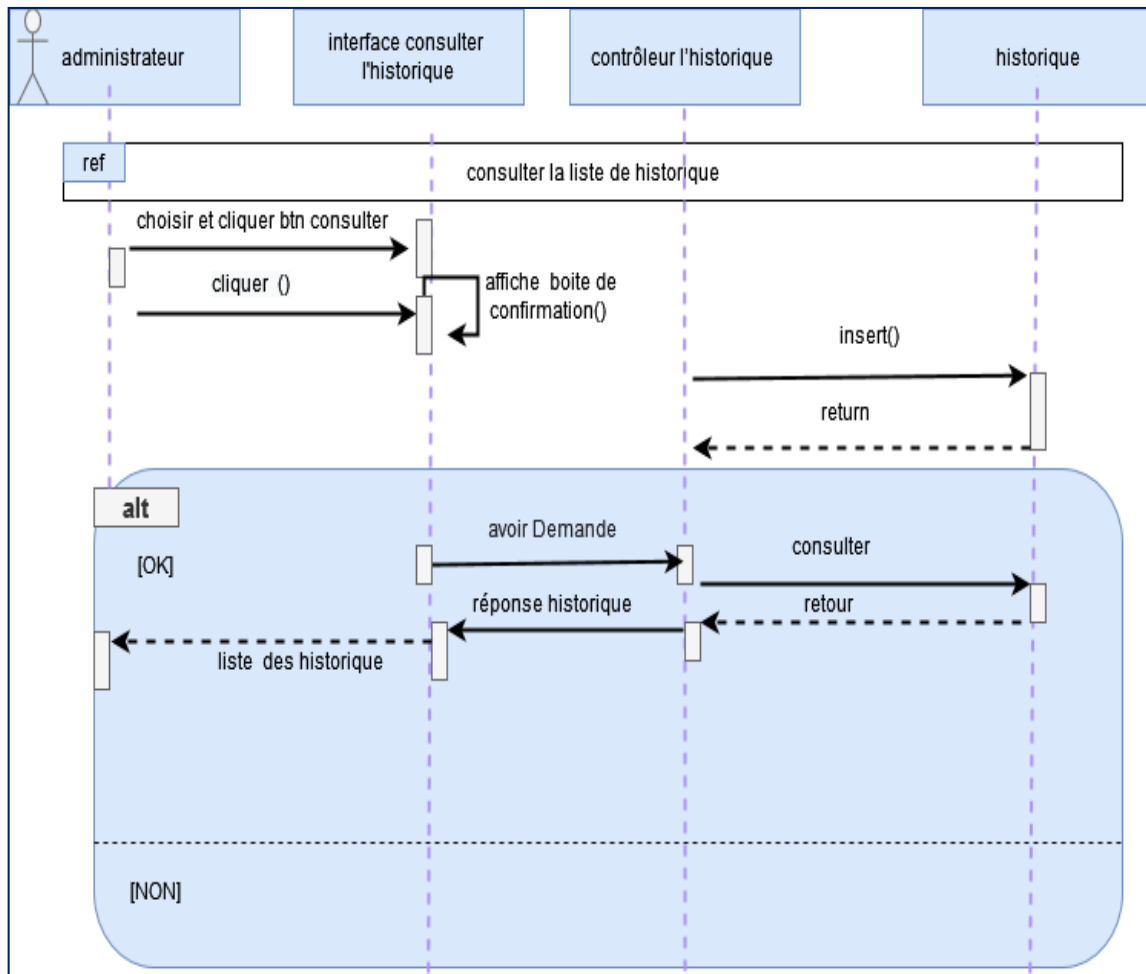


Figure 11 : Diagramme de séquence consulter l'historique

4. Diagramme de classe

Le diagramme de classes est généralement considéré comme le plus important dans un développement orienté objet. Il représente l'architecture conceptuelle du système : il décrit les classes que le système utilise.

La figure suivante décrit le diagramme de classe.

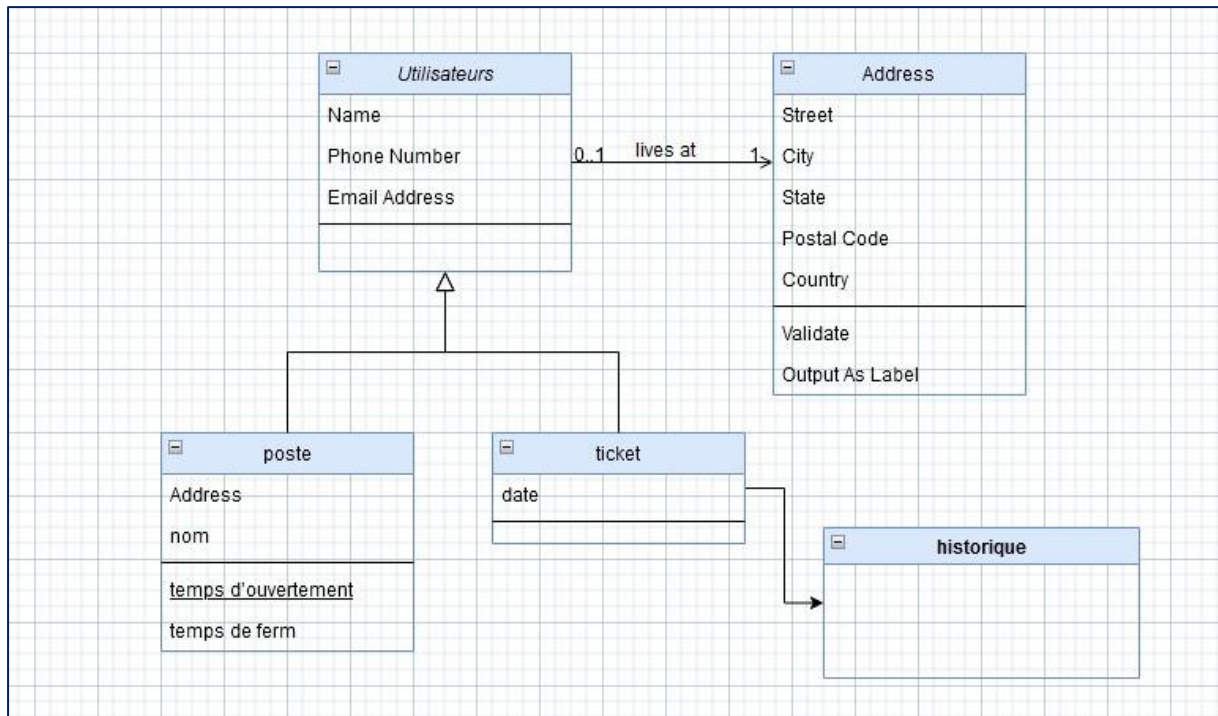


Figure 12 : Diagramme de classe

Conclusion

Ce chapitre a été consacré à différentes étapes de la conception détaillée. Du raffinement du diagramme de cas d'utilisation, vers du diagramme de séquence de plusieurs cas d'utilisations pour montrer les interactions entre les acteurs et le système. Finalement nous avons terminé avec le diagramme de classe avec lequel nous avons pour présenter la structure de notre application.

Le prochain chapitre contiendra des explications et des clarifications de plusieurs concepts en rapport avec le contexte de notre travail.

Chapitre 3 : Réalisation

Chapitre 3 : Réalisation

Introduction

Dans ce chapitre, nous sommes déjà passé par les étapes nécessaires à fin d'entraîner la phase de la réalisation. Le problème a été profondément analysé, nous avons défini une conception complète à notre jugement. Une conception qui comporte et décrit tous les besoins de l'application.

Dans ce chapitre nous commencerons par la description de l'environnement de développement (matériel et logiciel) ainsi que les différents outils utilisés. Ensuite, nous présenterons les différents aspects de fonctionnement de notre application.

1. Environnement de travail

Dans ce paragraphe, nous allons présenter l'environnement matériel et logiciel de développement de l'application que nous avons utilisée.

1.1. Environnement matériel

Pour développer l'application, nous avons utilisé comme environnement matériel un ordinateur portable et un téléphone portable qui possèdent les caractéristiques suivantes :

Tableau 3: Environnement matériel

Caractéristique	ASUS	INFINIX HOT 8
Marque	ASUS	INFINIX HOT 8
Processeur	Intel® Core (TM) i7-4500U CPU @ 1.80GHz 2.40 GHz	4x 2.0 GHz Cortex-A53, 4x 1.5 GHz ARM Cortex-A53, Nombre de noyaux : 8
RAM	8.00 G	4.00 G
Système d'exploitation	Microsoft Windows 10 Professionnel	XOS 5.0 Cheetah (Android 9.0Pie)

1.2. Environnement logiciel

❖ Android studio



Android studio est un est un environnement de développement pour développer des Applications Android. Il est basé sur IntelliJIDEA, mis gratuitement à la disposition des développeurs Android accompagné de sa documentation complète. Il est créé par Google pour remplacer l'ancien IDE Eclipse.

❖ Notepad ++



Notepad ++ est un éditeur de texte générique codé en C++. Ce logiciel, basé sur la Composante Scintilla, a pour but de fournir un éditeur léger (aussi bien au niveau de la taille du code compilé que des ressources occupées durant l'exécution) et efficace. Il est également une alternative au bloc-notes de Windows (d'où le nom). Le projet est sous licence GPL.

❖ WampServer



WampServer est une plateforme de développement Web de type WAMP, permettant de faire fonctionner localement (sans se connecter à un serveur externe) des scripts PHP.

❖ Firebase



Firebase est un ensemble de services d'hébergement pour n'importe quel type d'application (Android, iOS, Javascript, Node.js, Java, Unity, PHP, C++).

Toute l'implémentation et la gestion serveur de Firebase est à la charge exclusive de la société Alphabet. Les applications qui utilisent Firebase intègrent une bibliothèque qui permet les diverses interactions possibles.

1.3. Langage de programmation et technologies utilisés

❖ JAVA



Le langage Java est un langage de programmation informatique orienté objet créé par James Gosling et Patrick Naughton, employés de Sun Microsystems.

❖ XML



L'Extensible Markup Language (XML, « langage de balisage extensible » en français) est un métalangage informatique de balisage générique qui dérive du SGML.

❖ PHP



PHP : HyperText Préprocesseur, plus connu sous son sigle PHP (acronyme récursif), est un langage de programmation libre, principalement utilisé pour produire des pages Web dynamiques via un serveur HTTP, mais pouvant également fonctionner comme n'importe quel langage interprété de façon locale. PHP est un langage impératif orienté objet.

2. Plateforme Android

2.1. Historique

Android doit son nom à la startup éponyme spécialisée dans le développement d'applications mobiles rachetée par Google en août 2005.

2.2. Architecture Android

Android est basé sur un noyau linux 2.6, au-dessus l'architecture générale :



Figure 13 : Architecture générale Android

➤ Application

Android est fourni avec un ensemble de programmes de base (également nommés applications natives) permettant d'accéder à des fonctionnalités comme les courriels, les SMS, le téléphone, le calendrier, les photos, les cartes géographiques, le Web, pour ne citer que quelques exemples. Ces applications sont développées à l'aide du langage de programmation Java. Pour l'utilisateur final, c'est la seule couche accessible et visible.

➤ Le Framework (Application Framework)

En fournissant une plateforme de développement ouverte, Android offre aux développeurs la possibilité de créer des applications extrêmement riches et innovants. Les développeurs sont libres de profiter du matériel périphérique, les informations de localisation d'accès, exécutez les services d'arrière-plan, définir des alarmes, ajouter des notifications de la barre d'état, et beaucoup, beaucoup plus.

➤ Les bibliothèques (Libraires)

En interne, Android inclut un ensemble de bibliothèques C et C++ utilisées par de nombreux composants de la plateforme Android. Ces bibliothèques sont en réalité accessibles au développeur par l'intermédiaire du Framework Android. En effet, le Framework Android effectue, de façon interne, des appels à des fonctions C/C++ beaucoup plus rapides à exécuter que des méthodes Java standard.

➤ Moteur d'exécution Android (Android Runtime)

Android inclut un ensemble de bibliothèques qui fournit la plupart des fonctionnalités disponibles dans les bibliothèques de base du langage de programmation Java.

3. Présentation des interfaces de l'application

Cette section comporte des captures d'écran de quelques interfaces de l'application réalisée accompagnée par une brève description.

3.1. Les interfaces de l'application web

Cette application permet à l'administrateur de gérer les endroits, les parcours, les événements (insertion, modification, suppression et l'affichage).

3.1.1. Interface d'authentification

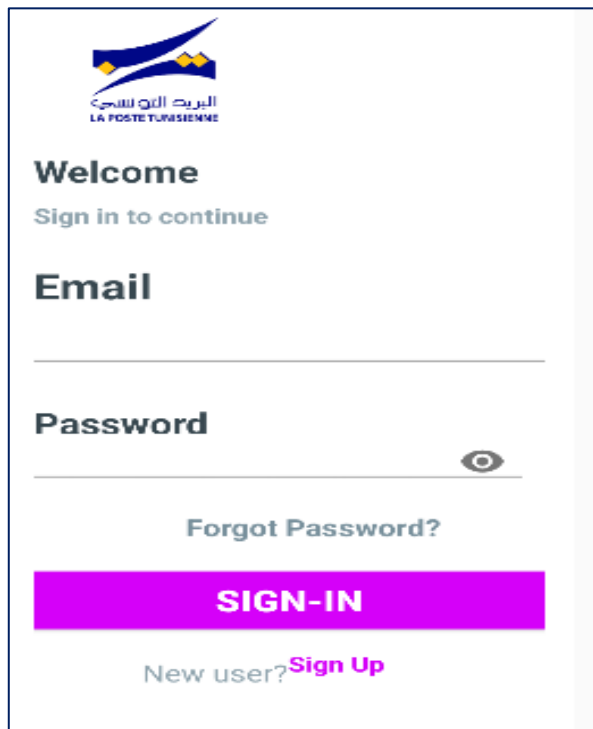


Figure 14 : Interface d'authentification

L'accès à l'application web est protégé par un système d'authentification.

L'administrateur doit saisir son login et mot de passe de passe à fin d'y accéder.

3.1.2. Interface création d'un compte

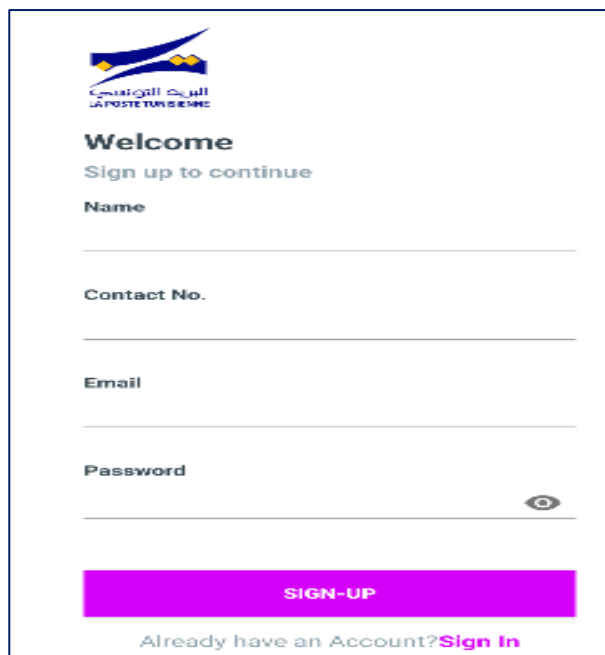


Figure 15 : Interface création d'un compte

3.1.3. Interface restart mot de passe

Dans cette interface l'administrateur peut accéder aux deux fonctionnalités de l'application

- ✓ 1 : Lien vers la page qui doit être écrire votre mail
- ✓ 2 : Lien vers la page de création nouveau mot de passe

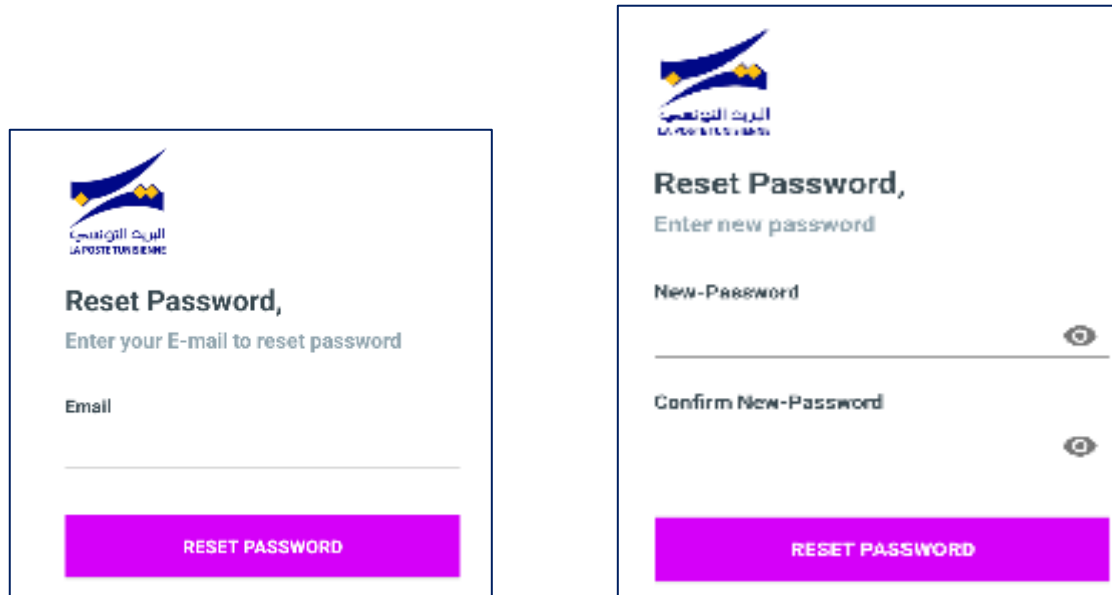


Figure 16 : Interfaces restart mot de passe

3.1.4. Interface page d'accueil

Si l'administrateur doit saisir son login et mot de passe correct

👉 On voit cette interface

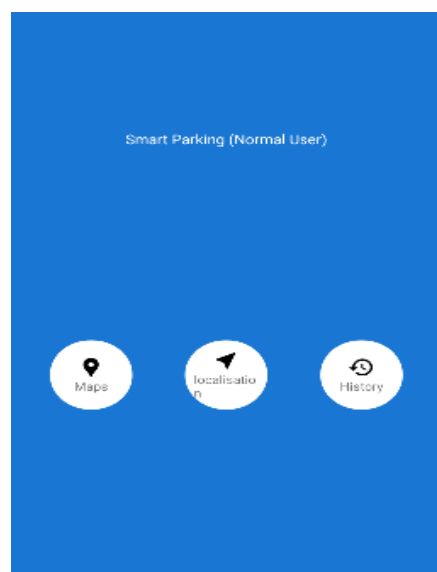


Figure 17 : Interface page d'accueil .

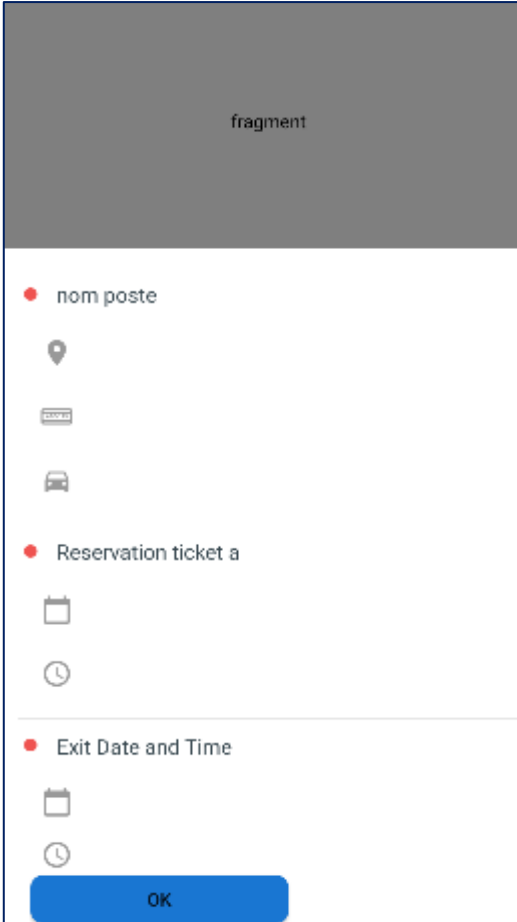
Dans cette interface l'administrateur peut accéder aux plusieurs fonctionnalités de l'application

- ✓ 1 : Lien vers la page de maps pour voir tous les poste
- ✓ 2 : Lien vers la localisation de l'utilisateur qui permet à l'utilisateur de voir le plus proche bureau de poste
- ✓ 3 : Lien vers l'historique

3.1.5. Interface réservation d'un ticket

Depuis cette interface on peut gérer la carte de l'api Google Map, les fonctionnalités sont :

- 1 : Localisation géographique du bureau de poste sur la carte
- 2 : les informations du bureau de poste
- 3 : la date de la réservation d'un ticket
- 4 : la date d'expiration ticket



The screenshot displays a mobile application interface for ticket reservation. It features a grey header with the text 'fragment'. Below the header, there are three distinct sections, each starting with a red dot and a label. The first section, 'nom poste', includes a location pin icon, a bus icon, and a car icon. The second section, 'Reservation ticket a', contains a calendar icon and a clock icon. The third section, 'Exit Date and Time', also contains a calendar icon and a clock icon. At the bottom of the interface is a blue button labeled 'OK'.

Figure 18 : Interfaces reservation d'un ticket

3.1.6. Interfaces de consulter les historiques

Dans cette interface, l'utilisateur voit les historiques :



Figure 19 : Interfaces gérer les endroits

Conclusion

Dans ce chapitre on a mis l'accent sur la description des caractéristiques de l'environnement du travail et décrit les plateformes matérielles et logiciel sur lesquelles nous avons construit notre application d'une part, d'autre part par la représentation des différents interfaces développées tout au long de l'application. Enfin nous avons clôturé ce chapitre par la présentation du chronogramme d'activité.

Conclusion générale

Au bout de notre cursus en licence informatique, nous avons été chargés de réaliser un projet de fin d'études. Notre travail s'est basé sur le développement d'un programme sur les technologies mobiles (Smartphone). Ceci nous a amené à découvrir une nouvelle plateforme de développement et à enrichir notre savoir et notre expérience.

Ce projet se dirige dans le cadre de notre licence appliquée en informatique au sein de l'Institut Supérieur d'Études Technologiques de Gafsa pour le compte de la société postale tunisienne.

Au cours de la phase de réalisation de notre application, nous avons élaboré une étude préalable sur les smart phones et son importance sur le plan social afin de préciser le but principal pour la future application. Cette phase a constitué le point de départ pour l'étape d'analyse et de spécification des besoins. Une fois nos objectifs sont fixés nous avons enchaîné avec la conception afin de mener à bien notre projet. Nous avons procédé à la phase de réalisation au cours de laquelle nous nous sommes familiarisés avec le langage de programmation java.

Pour conclure, notre travail peut être sujet à des extensions. En effet, nous envisageons d'ajouter une application sur Play store dans notre travail, la possibilité développer sous d'autres plateformes.

Bibliographie

- [1] L'art du développement Android 4e édition, 1er décembre 2012, de Grant Allen

- [2] Développez pour Android, 1re édition, 1er mars 2011, de Cyril Mottier et Ludovic Perrier.

- [3] Programmation Android De la conception au déploiement avec le SDK Google Android 2, de Damien Guignard, Julien Chable, Emmanuel Robles.

Netographie

- [1] <http://stackoverflow.com>
- [2] <https://openclassrooms.com>
- [3] <https://developers.google.com>
- [4] <http://android.developpez.com/cours>
- [5] <https://www.youtube.com>
- [6] <http://www.tutorialspoint.com/android>
- [7] <http://stage.lelkol.net>
- [8] <http://getbootstrap.com>
- [9] [http:// www.w3schools.com/bootstrap](http://www.w3schools.com/bootstrap)
- [10] <http://phpmyadmin.ovh.net>
- [11] <http://www.ovh.tn>
- [12] [Wikipédia](#)
- [13] [Firebase — Wikipédia \(wikipedia.org\)](#)

Annexe

❖ Android Studio versus Eclipse

	ADT (Eclipse)	Android Studio
Facilité d'installation	Moyen	Simple
Langue	Nombreuses	Anglais
Performance	Peut-être lourd	Rapide
Système de construction et compilation (build)	Ant	Gradle
Génération de variante et de multiple APK	Non	Oui
Complétion de code et refactorisation	Base	Avancé
Éditeur d'interface graphique	Oui	Oui
Signature d'APK et gestion de Key store	Oui	Oui
Support NDK	Oui	Oui

❖ Installer Android Studio

- Android Studio est disponible pour les systèmes Linux, Mac et Windows à partir de cette page : <https://developer.android.com/sdk/index.html>
- Pour Windows, si vous ratez votre installation, l'installateur n'arrête pas de boucler en affichant des messages d'erreur à propos de composantes manquantes, assurez-vous d'avoir préalablement effacé les fichiers préservés en cache, par la précédente installation, dans le répertoire Temp de votre machine.
- Quelques paramètres à respecter :
<https://developer.android.com/sdk/index.html#Requirements>
 - Minimum 2 GB RAM, recommandé 4 GB RAM
 - Espace disque : 400 Mb
 - Au moins 1 Gb pour le SDK d'Android, les images pour l'émulateur et la cache
 - 1280 x 800 la résolution minimale de l'écran
 - Java Development Kit (JDK) 7
 - Si l'on veut accélérer l'émulateur : processeur Intel avec support pour Intel VT- x, Intel EM64T (Intel 64), et la fonctionnalité « Execute Disable (XD) Bit »