**MINISTERE DE L’ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE UNIVERSITE DE SOUSSE**

**المعهد العالي لإلعالمية وتقنيات االتصال بحمام سوسة**



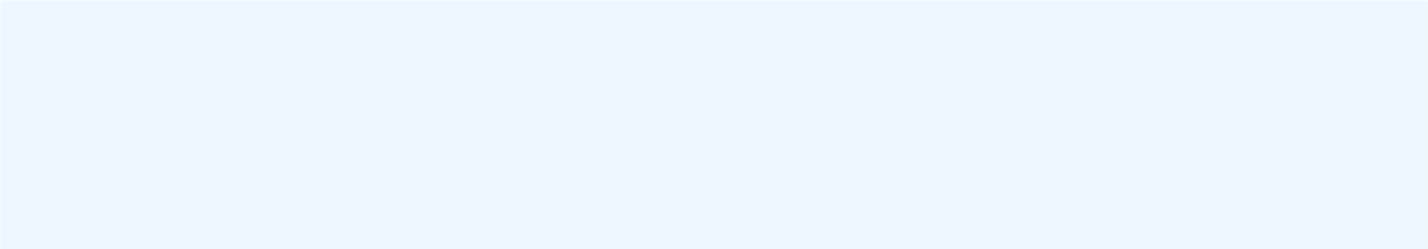
**INSTITUT SUPERIEUR D’INFORMATIQUE**

**ET DES TECHNOLOGIES DE COMMUNICATION – HAMMAM SOUSSE**

**Département Télécommunications**

**MEMOIRE DE STAGE FIN D’ETUDES**

**Présenté en vue de l’obtention du diplôme de Licence Appliquée Technologies des Communications**



**Application Android pour automatiser le traitement des dérangements**

**Réalisé par :** Habiba KORTAS Sarra JABALLAH

**Encadrées par :**

## Mr. Amin BEN AMOR

Mr. Radhwen RWETBI

**Société d’accueil**

## Tunisie Telecom

**MINISTERE DE L’ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE**

**UNIVERSITE DE SOUSSE**

**المعهد العالي لإلعالمية وتقنيات االتصال بحمام سوسة**

**INSTITUT SUPERIEUR D’INFORMATIQUE**

**ET DES TECHNOLOGIES DE COMMUNICATION – HAMMAM SOUSSE**

**Département Télécommunications**

**MEMOIRE DE STAGE FIN D’ETUDES**

**Présenté en vue de l’obtention du diplôme de Licence Appliquée Technologies des Communications**

**Réalisé par :** Habiba KORTAS Sarra JABALLAH

**Encadrant :** *Rwatbi Radhwen* **Date : Signature : Superviseur :** *Ben Amar Mohamed Amin* **Date :** *01/06/2016* **Signature :**

#### Année Universitaire 2015– 2016

Je dédié ce travaille

A ma mère Jalila, ma raison d’être, la lanterne qui éclaire mon chemin et m’illumine d’affection et d’amour.

A mon père abd allah, en signe d’amour, et de gratitude pour tous les soutiens et les sacrifices dont il a fait preuve à mon

égard

A mes chers frères,

A mes chères amies Kouki Khouloud et Gâddes Yamina, Aucun mot ne pourra décrire vos dévouements.

A tous mes proches et mes amis,

Je vous dédie ce travail en vous souhaitant un avenir éclatant et plein de bonnes promesses.

A tous les gens qui ont cru en moi et qui me donnent l’envie d’aller en avant, je vous remercie tous, vos

encouragements et votre soutien me donnent la force de continuer.

…Habiba

A mes chers parents,

Aucun mot, aucune dédicace ne saurait exprimer mon respect, ma considération et l’amour éternel pour les sacrifices que vous

avez déployés pour mon instruction et mon bien être dans les meilleures conditions.

Votre générosité et votre bonté ont toujours été un exemple pour moi.

Trouvez en ce travail le fruit de votre dévouement et l’expression de ma gratitude et mon profond amour.

A mes frères, mes sœurs et mes proches,

A qui je dois ma reconnaissance, je vous remercie vivement pour votre présence et votre soutien.

…Sarra

**Remerciements**

Ce travail n’aurait jamais pu se concrétiser sans l’aide et le soutien de

plusieurs

## personnes que nous souhaitons vivement remercier et à qui nous dédions ce travail.

Monsieur Ben Amor Amin notre encadrant, qui n'a pas cessé de nous prodiguer

## ses conseils et qui n'a épargné aucun effort pour contribuer à la réussite de notre travail.

Monsieur Rwetbi Radhwen, pour les directives qu’il nous a fournies pendant la durée de stage.

## Les personnels et surtout les techniciens de l’entreprise « Tunisie Telecom », pour l’expérience enrichissante et pleine d’intérêt qu’ils nous ont fait vivre.

Tous nos professeurs et plus particulièrement les membres de jury qui ont accepté de juger notre travail.

## Notre Institut qui nous a donné l'occasion d'acquérir une formation professionnelle.

Habiba & Sarra

**Table des matières**

 [................................................................................................................... 1](#_bookmark0)

[.......................................................................................... 4](#_bookmark1)

1. [Présentation de la société 4](#_bookmark2)
   1. [Introduction générale 4](#_bookmark3)
   2. [Logos de Tunisie Telecom 4](#_bookmark4)
   3. [Localisation de Tunisie Telecom 5](#_bookmark5)
   4. [Stratégie de Tunisie Telecom 5](#_bookmark7)
   5. [Organigramme de Tunisie Telecom 6](#_bookmark8)
   6. [Activités de Tunisie Telecom 6](#_bookmark9)
   7. [Organigramme de la direction régional 7](#_bookmark10)
2. [Cadre général du travail 8](#_bookmark11)
3. [Présentation du sujet 8](#_bookmark12)
4. [Présentation du projet 8](#_bookmark13)
5. [Le but à atteindre 8](#_bookmark14)
6. [Travail demandé 8](#_bookmark15)
7. [Planification du projet 9](#_bookmark16)

 [................................................................... 11](#_bookmark17)

[Introduction 11](#_bookmark18)

1. [Étude préalable 11](#_bookmark19)
   1. [Étude de l’existant 11](#_bookmark20)
      1. [Description de l’existant 11](#_bookmark21)
      2. [Critique de l’existant 11](#_bookmark22)
   2. [Solutions envisagées 12](#_bookmark23)
2. [Spécifications des Besoins 12](#_bookmark24)
3. [Les Besoins fonctionnels 12](#_bookmark25)
4. [Les Besoins non fonctionnels 13](#_bookmark26)
   1. [Contraintes ergonomiques 13](#_bookmark27)
   2. [Contraintes techniques 13](#_bookmark28)

[................................................................................................. 15](#_bookmark29)

[Introduction 15](#_bookmark30)

* + 1. [Cycle de vie de développement de projet 15](#_bookmark31)
    2. [Langage UML 16](#_bookmark32)

1. [Présentation du langage UML 16](#_bookmark33)
2. [Intérêt de la modélisation 16](#_bookmark34)
   * 1. [Conception avec UML 18](#_bookmark35)
3. [Outil de modélisation 18](#_bookmark36)
4. [Modélisation avec les diagrammes cas d’utilisation 18](#_bookmark37)
   1. [Le diagramme de Classe 24](#_bookmark45)

 [......................................................................................................... 38](#_bookmark61)

[Introduction 38](#_bookmark62)

* + 1. [Environnement de travail 38](#_bookmark63)
       1. [Environnement matériel 38](#_bookmark64)
       2. [Environnement logiciel 38](#_bookmark65)
    2. [Développement 42](#_bookmark68)
    3. [Choix de développement 43](#_bookmark69)

1. [Le modèle 44](#_bookmark71)
2. [La vue 44](#_bookmark72)
3. [Le contrôleur 44](#_bookmark73)
   * 1. [Les technologies utilisées 44](#_bookmark74)
     2. [Les interfaces graphiques 45](#_bookmark75)

[a. Formulaire de réclamation 45](#_bookmark76)

[....................................................................................................................................................... 46](#_bookmark77)

1. [Interface Authentification 46](#_bookmark79)
2. [Interface d’accueil 47](#_bookmark81)
3. [Interface Consulter la liste des réclamations 47](#_bookmark82)
4. [Interface Consulter la liste des clients 48](#_bookmark84)
5. [Interface afficher la position technique 48](#_bookmark85)
6. [Interface Ajouter un client 49](#_bookmark87)
7. [Interface supprimer un client 49](#_bookmark89)
8. [Interface consulter la liste des personnelles 50](#_bookmark90)
9. [Interface ajouter personnelle 50](#_bookmark91)
10. [Interface Authentification 51](#_bookmark93)
11. [Interface Accueil 51](#_bookmark94)
12. [Interface Contacter 52](#_bookmark95)
13. [Interface de la liste des réclamations 52](#_bookmark97)
14. [Interface d’intervention 53](#_bookmark99)
15. [Interface de statistiques 54](#_bookmark101)

 [.................................................................................................................... 55](#_bookmark102)

[.................................................................................................................................... 56](#_bookmark103)

**Liste des figures**

Figure 1: Logos de Tunisie Telecom 5

[Figure 2: Localisation de Tunisie Telecom 5](#_bookmark6)

Figure 3 : Organigramme de Tunisie Telecom 6

Figure 4: Organigramme de la direction régional 7

Figure 5: Modèle du cycle de vie en cascade 16

Figure 6: Diagramme de cas d'utilisation général 19

[Figure 7: diagramme de cas d’utilisation « Affecter réclamation » 20](#_bookmark38)

Figure 8 : Diagramme de cas d’utilisation «Gérer les clients» 20

Figure 9: Diagramme de cas d’utilisation « Consulter réclamations » 21

Figure 10: cas d'utilisation "Déposer intervention" 22

[Figure 11: Diagramme de cas d'utilisation"consulter statistiques" 23](#_bookmark43)

Figure 12: Diagramme de classe 24

[Figure 13: Diagramme de séquence «Authentification » 32](#_bookmark60)

Figure 14: Diagramme de séquence «Inscription » 33

Figure 15: Diagramme de séquence "Supprimer client " 34

Figure 16: Diagramme de séquence « déposer intervention» 35

Figure 17 : Diagramme de séquence « consulter statistiques» 36

Figure 18: Logo de l'émulateur GenyMotion 39

Figure 19: Logo d’Eclipse JUNO 40

[Figure 20: Logo Eclipse MARS 41](#_bookmark66)

[Figure 21: Logo EasyPHP 41](#_bookmark67)

Figure 22: Logo Java EE 43

[Figure 23: Schéma du Modèle MVC 43](#_bookmark70)

[Figure 24: Formulaire de réclamation 46](#_bookmark78)

[Figure 25: Interface authentification 46](#_bookmark80)

Figure 26: Interface d'accueil 47

[Figure 27: Interface Réclamations 47](#_bookmark83)

Figure 28: Interface liste des clients 48

[Figure 29: Interface de la position technique 48](#_bookmark86)

[Figure 30: Interface Ajouter un client 49](#_bookmark88)

Figure 31: message de suppression 49

Figure 32: Interface consulter la liste des personnelles 50

[Figure 33: Interface ajouter personnelle 50](#_bookmark92)

Figure 34: Interface Authentification 51

Figure 35: Interface inscription 51

Figure 36: interface accueil 51

Figure 37: Rubrique info et aide 51

[Figure 38: Interface contacter 52](#_bookmark96)

Figure 39: Interface de la liste des réclamations 53

[Figure 40: Interface de coordonnées de client 53](#_bookmark98)

[Figure 41: Interface d'intervention 53](#_bookmark100)

Figure 42: Interface de statistiques 54

**Liste des tableaux**

Tableau 1 : planification temporelle 9

[Tableau 2 : Cas d’utilisation «Affecter réclamation » 20](#_bookmark39)

[Tableau 3 : Cas d’utilisation « Gérer les clients » 21](#_bookmark40)

[Tableau 4 : Cas d’utilisation « Consulter réclamations » 22](#_bookmark41)

[Tableau 5: cas d'utilisation "Déposer intervention" 23](#_bookmark42)

[Tableau 6: cas d’utilisation "Consulter les statistiques " 23](#_bookmark44)

[Tableau 7: Classe amorce\_t 25](#_bookmark46)

[Tableau 8 : classe amorce\_d 26](#_bookmark47)

[Tableau 9 : classe client 26](#_bookmark48)

[Tableau 10 : classe couleur\_t 27](#_bookmark49)

[Tableau 11 : classe couleur\_d 27](#_bookmark50)

[Tableau 12: classe distribution 28](#_bookmark51)

[Tableau 13 :classe etat\_derang 28](#_bookmark52)

[Tableau 14: classe technicien 29](#_bookmark53)

[Tableau 15 : classe réclamation 29](#_bookmark54)

[Tableau 16 : classe région 30](#_bookmark55)

[Tableau 17 : classe repartiteur\_général 30](#_bookmark56)

[Tableau 18 : classe sous\_repartiteur 30](#_bookmark57)

[Tableau 19 : classe transfert 31](#_bookmark58)

[Tableau 20: classe admin 31](#_bookmark59)

**Liste des Acronymes**















## **UML :** Unified Modeling Language **ADT :** Android Developer Tools **JRE** : Java Runtime Environment

**MVC** : Model-View-Controller

**GNOME :** GNU Object Model Environment



Depuis l’antiquité, l’homme n’a pas cessé de chercher constamment les différents moyens pour faire véhiculer le message à son correspondant et donc pour communiquer. De ce fait et à travers ces époques successives, il a fourni ses efforts intellectuels et physiques afin de découvrir des méthodes de communication adéquates.

Jusqu’au 20ème siècle, l’émergence d’une révolution pour les télécommunications d’où le développement des technologies et la disponibilité croissante d’internet ont accéléré la transmission des informations entre l’émetteur et le récepteur. Les flux d’information sont dorénavant étendus, diversifiés, réversibles et accessibles.

Aujourd’hui, dans un marché des applications mobiles en est plein expansion, renforcé par l’essor des tablettes informatiques, d’où le nombre d’applications mobiles téléchargées augmente de manière exponentielle chaque année.

Dance ce contexte, de plus en plus d’entreprises du secteur de télécommunications souhaitent investir dans le domaine et semblent intéressées par l’utilisation d’application mobiles.

Faisant partie de l’Isitcom, et dans le cadre de notre projet de fin d’études en vue de l’obtention du diplôme de Licence Appliquée en Technologies des Télécommunications, nous sommes dirigés à développer une application Android pour automatiser le traitement des dérangements au sein de la société d’accueil « Tunisie Telecom».

Ainsi, nous organisons notre rapport autour de quatre chapitres :

* Dans le premier chapitre « cadre général », nous expliquons le cadre de notre projet.
* Dans le deuxième chapitre « analyse et spécification des besoins», nous déterminons les acteurs du futur système, les besoins fonctionnels et non fonctionnels.
* Dans le troisième chapitre «Étude conceptuelle », nous modelons une conception des diagrammes de cas d’utilisation, de diagramme de classe ainsi que le diagramme des séquences complet.
* Dans le dernier chapitre « réalisation », nous présentons l’environnement matériel et logiciel, ainsi que les interfaces principales de l’application réalisée.





Dans ce chapitre nous mettons notre travail dans son contexte général, Tout d’abord, nous décrivons l’environnement du stage à travers une brève présentation de l’organisme d’accueil. Ensuite nous présentons le cadre général de notre projet.

# Présentation de la société

#### Société : Tunisie Telecom

**Adresse : Jardins du Lac II, 1053, Tunis, Tunisie Téléphone : (00 216) 71 901 717**

#### Fax : (00 216) 71 900 777

**Site web :** [**www.tunisietelecom.tn**](http://www.tunisietelecom.tn/)

Dans cette partie, nous présentons l’organisme d’accueil qui a proposé ce sujet.

# Introduction générale

Crée en 1995, « Tunisie Télécom » est une société sous forme d'entreprise publique à caractère industriel et commercial. C’est un opérateur historique des télécommunications en Tunisie.

Avec 24 directions régionales, 95 agences commerciales et 13000 points de vente, « Tunisie Telecom » bénéficie d’une clientèle composée d’environ 8 millions d’abonnés dans le téléphone fixe et mobile. Elle joue un rôle principal dans l’amélioration du taux de pénétration de l’internet en Tunisie. Cette société emploie plus de 9000 agents, dont 45% de cadres. [1].

Tunisie Telecom Leader sur le marché des télécommunications en Tunisie et parmi les pionniers des grands opérateurs de la région, elle est l’opérateur historique global et intégré. Présent sur les segments de la fixe, du mobile, de l’Internet et des services data, l’opérateur s’adresse aussi bien au grand public qu’aux Entreprises et opérateurs tiers.

En effet, Tunisie Telecom dispose de différentes entités commerciales, techniques et administratives**.**

# Logos de Tunisie Telecom :



*Figure 1: Logos de Tunisie Telecom*

# Localisation de Tunisie Telecom :



*Figure 2: Localisation de Tunisie Telecom*

# Stratégie de Tunisie Telecom

La stratégie de Tunisie Telecom s’adresse ainsi autour des principales orientations suivantes :

* + - * Privilégier la croissance du marché mobile par une politique d’approche commerciale fragmentée visant à accélérer les usages nouveaux et existants des services de télécommunications mobiles
      * Participer au développement croissant du département des services de données aux entreprises et dynamiser l’emploi de la téléphonie fixe.
      * Rester le principal fer de lance du développement de l’Internet en Tunisie ;
      * Rassembler sur ses marques et faire de Tunisie Telecom une référence de service clients en Tunisie et en domaine d’engagement qualité.
      * Poursuivre d’investir dans son cœur de réseau dans l’intention de renforcer l’accès au très haut débit fixe et mobile.
      * Devenir un point d’accès inévitable pour les services internationaux (« hub » régional, voire continental) en renforçant sa position de partenaire de référence pour les opérateurs nationaux et internationaux.
      * Améliorer la création de valeur pour ses actionnaires, à travers des relais de croissance de chiffre d’affaires et ordonner l’organisation salariale de la Société aux prérogatives du marché.

# Organigramme de Tunisie Telecom :

*Figure 3 : Organigramme de Tunisie Telecom*

# Activités de Tunisie Telecom :

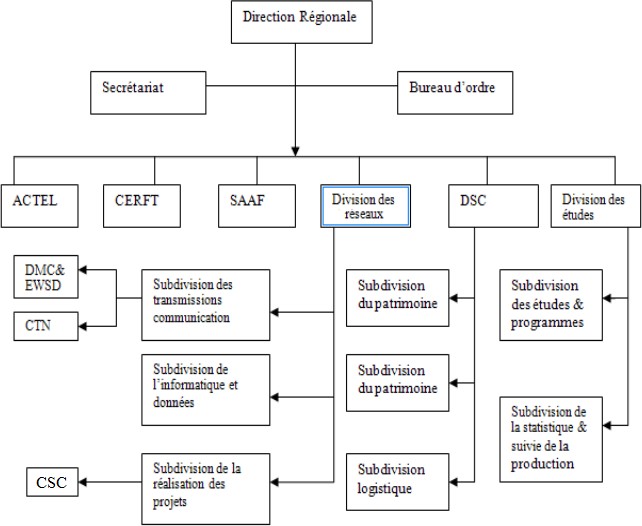
Tunisie Telecom a pour but d’assurer les activités liées au domaine des télécommunications ; pour cela, il est notamment chargé de :

* L’installation, le traitement, l’entretien et l’exploitation des réseaux de téléphones, de transmission de données et de télex.
  + La participation au développement des études et recherches scientifiques attachées au secteur de télécommunications.
  + La contribution à l’effort national d’enseignement supérieur au niveau du secteur de télécommunication et mis en place le 01/01/96 : identifié par son nom commercial Tunisie.
  + L’offre de tous les services publics ou privés de télécommunications correspondant aux

différents besoins à caractère social et économique.

* + L’amélioration des nouveaux services des télécommunications
  + L’application des protocoles et des traités des organisations internationales et régionales spécialisées dans le domaine des télécommunications.
  + L’amélioration de la participation à tous les niveaux dans le domaine des télécommunications [3].

# Organigramme de la direction régional :



*Figure 4: Organigramme de la direction régional*

Le **CSC** est un centre de service clientèle. En effet, il est nécessaire pour le raccordement des abonnés et l’entretien des lignes téléphoniques tout en provenant par l’étude des nouvelles demandes de construction et la surveillance des travaux des constructeurs.

Ce centre a la charge de l’entretien, de l’installation téléphonique et enfin de l’intervention pour le rétablissement d’une ligne qui a déjà été dérangée. Il possède des équipes réparties selon

les zones qui sont équipées afin de manipuler le réseau à partir du répartiteur général jusqu’à l’appareil terminal situé chez l’abonné.

# Cadre général du travail :

Pour ce stage d’insertion dans le monde du travail, l’objectif est la conception et le développement d’une application Android pour automatiser le traitement de dérangements, attachée d’une application Web permet de circuler les services par l’administrateur.

# Présentation du sujet :

# Présentation du projet :

Notre travail est réalisé dans le cadre de stage de projet de fin d’études.

L’application Android est compatible avec la version « Jelly Bean 4.2.2 ». Elle permet aux techniciens d’automatiser le traitement de dérangement et le rendre plus facile, afin d’améliorer leurs rendements de travail.

# Le but à atteindre

Le but de notre projet est de créer et développer une application Android pour :

* + - * **Faciliter** et **organiser** le travail des techniciens.
      * Garantir **le bon service** aux clients**.**
      * **Gagner du temps** et atteindre **la satisfaction** des clients.
      * Une application pour Smartphone est bien souvent **plus ergonomique** que l'utilisation des papiers (les avis). Elle rend l’expérience de l’utilisateur bien meilleure grâce à la navigation.

# Travail demandé :

Notre tâche permet de :

* + - * Apprendre la plateforme de développement mobile sous Android.
      * Réaliser l’interface graphique pour l’application.
      * Développer une application web dynamique
      * Concevoir et développer une base de données et l’intégrer dans l’application
      * Établir la connexion entre l’application et la base de données.
      * Développer une application fonctionnant sur un Smartphone Android.

# Planification du projet :

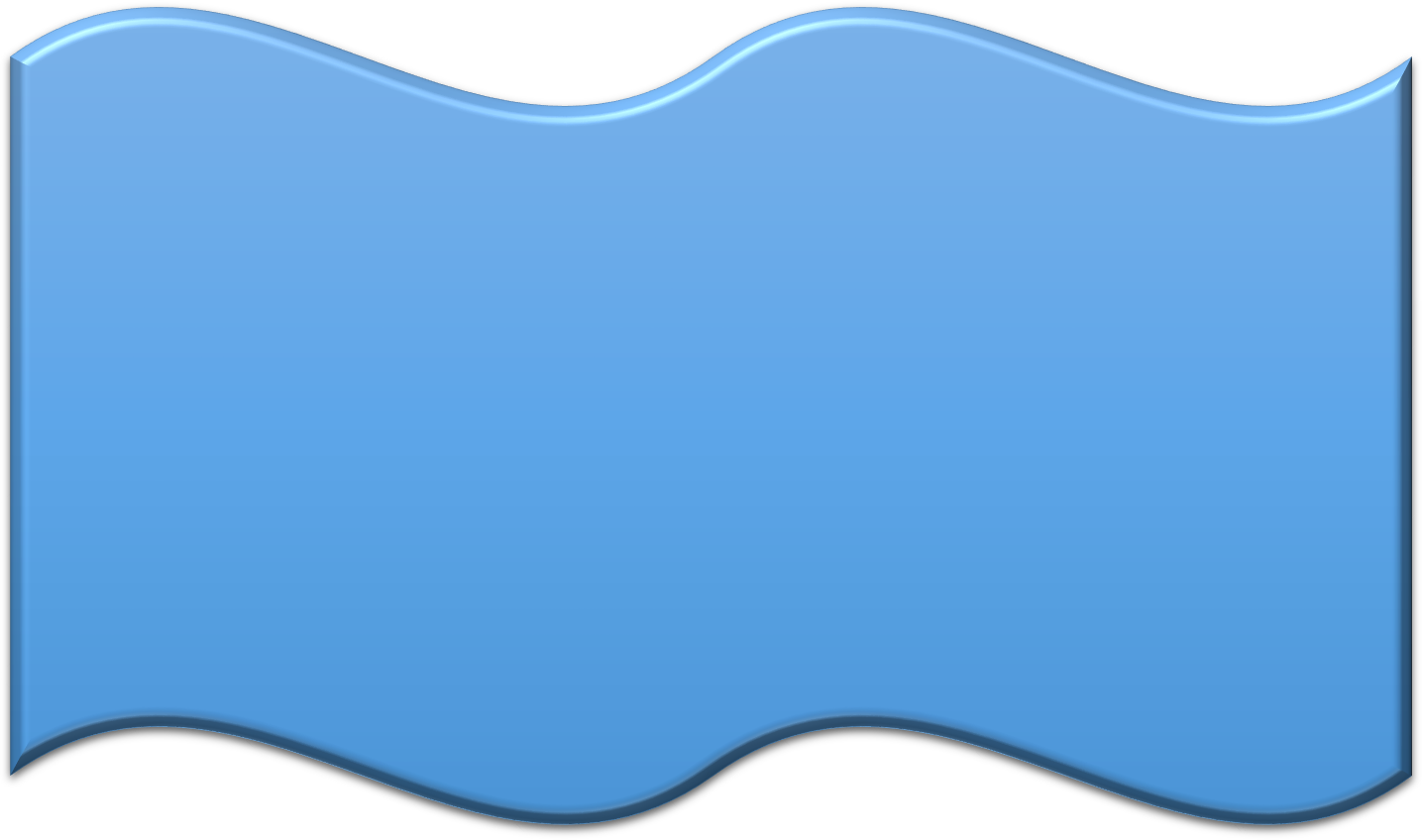
La répartition du temps est un parmi les plus efficaces des outils de développement des aptitudes personnelles en gestion et en performance du travail. C'est un ensemble de techniques claires et d'outils pratiques d’aménagement personnelle et de planification qui permet de diminuer le temps perdu au minimum et d'améliorer la performance personnelle très rapidement ce qui présenté dans la figure ci-dessous :

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Mois** | **Février 2016** | | | | **Mars 2016** | | | | **Avril 2016** | | | | **Mai 2016** | | | |
| **Semaine** | **Semaine 1** | **Semaine 2** | **Semaine 3** | **Semaine 4** | **Semaine 1** | **Semaine 2** | **Semaine 3** | **Semaine 4** | **Semaine 1** | **Semaine 2** | **Semaine 3** | **Semaine 4** | **Semaine 1** | **Semaine 2** | **Semaine 3** | **Semaine 4** |
| **Recherche et documentation** | **■** | **■** | **■** | **■** | **■** | **■** | **■** | **■** | **■** | **■** | **■** | **■** |  |  |  |  |
| **Analyse des besoins et Spécification** |  |  |  | **■** | **■** | **■** | **■** | **■** | **■** |  |  |  |  |  |  |  |
| **Conception** |  |  |  |  |  | **■** | **■** | **■** | **■** | **■** | **■** |  |  |  |  |  |
| **Développement** |  |  |  |  |  |  | **■** | **■** | **■** | **■** | **■** | **■** | **■** | **■** |  |  |
| **Tests et validation** |  |  |  |  |  |  |  |  |  |  |  |  | **■** | **■** |  |  |
| **Élaboration du**  **rapport** |  |  |  |  |  |  |  |  |  | **■** | **■** | **■** | **■** | **■** | **■** | **■** |

*Tableau 1 : planification temporelle*

# Conclusion :

Dans ce chapitre, nous avons mis notre projet dans son contexte général .Nous avons commencé par une description générale de la société «Tunisie Telecom». Nous avons exposé ses logos, sa localisation puis ses stratégies et principales activités et ses organigrammes ensuite nous avons présenté le cadre général de notre projet, une brève présentation, le but à atteindre et le travail demandé en finissant avec l’environnement de développement et la planification temporelle.

**Analyse et spécification des besoins**



# Introduction :

Dans ce deuxième chapitre, nous commençons par la description de système de fonctionnement existant au CSC au sein de la société « Tunisie Telecom ».

L’étude réalisée soulignera quelques problèmes au niveau du rendement et de satisfaction totale de client.Cela nous permettre de proposer une solution totalement informatisée pour améliorer la qualité de service.

# Étude préalable :

Pour assurer les buts à atteindre, il est nécessaire que nous commencions par une vue claire et simple des différents besoins escomptés. C’est pour cela qu’une étude du système existant nous nous permettra de justifier et de dégager une solution.

# Étude de l’existant :

# Description de l’existant :

Un dérangement est tout défaut qui trouble le fonctionnement normal des lignes téléphoniques suite à certains accidents naturels (pluie, vent,…) ou un endommagement accidentel d’un câble ou d’un poteau. Tout abonné ayant une ligne dérangée doit impérativement déclarer l’état de sa ligne en composant le numéro 1298, envoyer un SMS sur le numéro 87117 ou réclamer en ligne sur le site [www.tunisie telecom.com.](http://www.tunisietelecom.com/) Un agent spécifique est destiné pour déterminer le type du défaut à l’aide de quelques tests effectués sur la ligne dérangée.

Une fois terminé, un avis contenant le numéro de la ligne dérangée, l’adresse de client, sa position est reçue par l’orienteur grâce à un minitel. Les avis sont ensuite partagés aux équipes selon la région de travail sachant qu’ils doivent commencer ses travaux directement lors de la réception de ces avis puisque chaque client doit être satisfait pendant 24 heures au maximum.

# Critique de l’existant :

Il est évident qu’il faut garantir un repérage rapide de dérangements, pour assurer le bon service de Tunisie Telecom envers ces clients.

Le problème est que les techniciens perdent beaucoup du temps pour régler ces dérangements à cause de :

* Une mauvaise gestion et un partage de temps inapproprié
* La non-disponibilité des clients pour la plupart du temps
* L’insuffisance de nombre d’équipes
* Manque d’assiduité des employés
* La non-réception des E-mails envoyés par la table d’essai
* La perte de certains avis de dérangement.

Vu le nombre croissant des réclamations, les heures de travail effectives des techniciens ne seront en aucun cas suffisantes pour régler les dérangements aux termes, d’où une insatisfaction non souhaitable des abonnés.

# Solutions envisagées :

Pour résoudre les problèmes dégagés et garantir le professionnalisme des services offerts par Tunisie Telecom, nous visons :

* Concevoir et développer une application mobile sous l’Android qui permet d’automatiser le traitement des dérangements afin d’informatiser la communication entre le technicien et l’administrateur pour garantir la rapidité et la sûreté des transferts des données.

# Spécifications des Besoins

# Les Besoins fonctionnels :

Le futur système permet à :

* « L’administrateur » de :
  + Recevoir les réclamations des clients.
  + Gérer les clients : Ajouter, modifier, supprimer, rechercher, afficher et imprimer la liste.
  + Affecter les réclamations de dérangements : supprimer, transférer imprimer la liste et rechercher à partir les numéros fixes.
  + Consulter les historiques des réclamations.
  + Gérer les personnels ajouter, modifier, supprimer, afficher, rechercher et imprimer la liste.
  + Consulter les interventions effectuées par les techniciens.
* L’utilisateur « TECHNICIEN » de :
  + Contacter l’administrateur et les personnels.
  + Accéder et lister les nouvelles réclamations.
  + Suivre toutes les coordonnées des clients.
  + Suivre l’historique de chaque client.
  + Identifier le cas exact de dérangement et les solutions obtenues.
  + Consulter les statistiques récapitulatives des interventions validées pour bien évaluer son rendement.

# Les Besoins non fonctionnels :

Les besoins non fonctionnels représentent les régles à respecter afin d’assurer une bonne qualité de l’application à achever et le bon fonctionnement du futur systéme.Quant aux besoins non fonctionnels,ils se récapitulent en :

# Contraintes ergonomiques :

Le système doit :

* + Permettre aux utilisateurs de se familiariser rapidement avec le contenu.
  + Être simple et compréhensible.
  + Dispose les boutons de manière logique.

# Contraintes techniques :

* + L’accès à la base de données doit être rapide et sécurisée
  + L’application doit être toujours effective.
  + Temps de réponse minimum.

# Conclusion :

Au cours de ce chapitre, nous avons réalisé une étude au niveau du fonctionnement du système existant et une analyse concernant le déroulement des traveaux de techniciens, afin de trouver une solution acceptable et décrire les besoins fonctionnels et non fonctionnels du nouveau système.



**É**



# Introduction

Penser avant d’agir, faire des plans avant de construire, concevoir d’abord, développer ensuite c’est la démarche qui doit être suivre lors du développement d’une application et pour réussir n’importe quel projet.

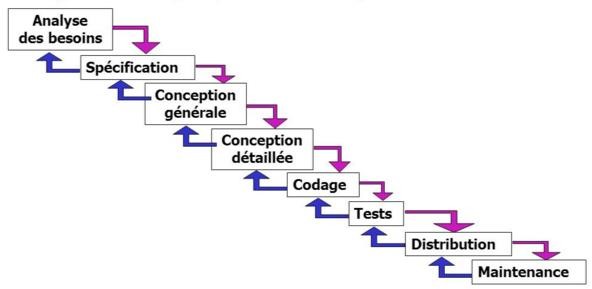
En effet, La conception d’un système informatique est une étape très importante qui va influencer la qualité et la fiabilité de toute application.

D’abord, nous allons commencer ce chapitre par l’explication du modèle de cycle de vie de projet qu’on a choisi. Ensuite nous allons passer à la partie de conception détaillée où nous présentons l’architecture globale de l’application. Enfin nous citons les différents diagrammes de cas d’utilisation, de séquences et le diagramme de classes.

# Cycle de vie de développement de projet :

Le cycle de vie d’un logiciel (en anglais software life cycle), désigne toutes les étapes du développement d’un logiciel, de sa création à sa disparition. L’objectif d’un tel découpage est de permettre de définir des bornes intermédiaires permettant la confirmation du développement logiciel, c’est-à-dire la coïncidence du logiciel avec les besoins exposés, et la vérification du processus de développement, c’est-à-dire la correspondance des méthodes mises en œuvre. [4] Pour fournir une meilleure réalisation, nous avons présenté le modèle en cascade (a été mis au point dès 1996). Il est décrit par la figure 5.

Dans ce modèle le principe est très simple : chaque phase se termine à une date précise par la production de certains documents ou logiciels. Les résultats sont définis sur la base des interactions entre étapes, ils sont soumis à une revue approfondie et on ne passe à la phase suivante que s’ils sont jugés satisfaisants.



*Figure 5: Modèle du cycle de vie en cascade*

# Langage UML :

# Présentation du langage UML :

**UML** (en anglais Unified Modeling Language, « langage de modélisation unifié ») est un langage graphique de modélisation des données et des traitements. C’est une formalisation non- propriétaire de la modélisation objet utilisée en génie logiciel. UML spécifie plusieurs objectifs qui font un outil exact de communication :

* Comprendre et décrire les besoins.
* Spécifier un système.
* Établir l’architecture logicielle.

# Intérêt de la modélisation :

L’utilisation de la modélisation conceptuelle dans le développement des systèmes d’information permet de prendre en compte les besoins des applications d’une façon plus adéquats et de présenter d’une manière abstraite certains aspects des systèmes physiques et humains.

# Les avantages d’UML :

* UML est un langage formel et standardisé.
  + Gain de précision.
  + Motivation à l’utilisation d’outils.
  + Gagne de stabilité et de fixité.
* UML est un support de communication adéquat et compétitif.
  + Il éclaire et facilite la compréhension de représentation abstraite complexe.
  + Son caractère plurivalent et sa souplesse en font un langage universel.
  + UML a pour objectif de spécifier, édifier et documenter les systèmes à base de logiciel.
  + UML n’est pas une méthode mais une notation qui laisse la liberté de la conception.
  + UML est un langage qui permet de modéliser tous les types de systèmes informatiques mais, qui nécessite toutefois une méthodologie de conception.

UML normalises les concepts objet, sa notion graphique permet d’exprimer une solution objet, ce qui simplifie la comparaison et l’appréciation des solutions. UML cadre l’analyse objet, il permet non seulement de représenter les concepts objets, mais il sous-entend une démarche d’analyse qui permet de reproduire une solution objet de manière itérative, grâce aux diagrammes, qui supportent l’abstraction.

Un diagramme UML est une représentation graphique, et à chaque vue correspondent des diagrammes qui sont répartis selon leurs aspects statiques ou dynamiques :

#### Statique :

* + Cas d’utilisation
  + Classes
  + Composants
  + Objets
  + Déploiement

#### Dynamique (comportementaux) :

* + Séquences
  + Activité
  + État-transition
  + Collaboration

#### Fonctionnel :

* + Cas d’utilisation
  + Collaboration

Ces diagrammes ne sont pas nécessairement tous produits à l’occasion d’une modélisation. Les plus utilisables sont les diagrammes d’activités, de classes, de cas d’utilisation, d’objets, d’états transitions et de séquence. Les diagrammes de composants, de déploiement et de communication sont surtout utiles pour la maîtrise d’étude à qu’ils permettent de formaliser les contraintes de la réalisation et les solutions.

# Conception avec UML :

# Outil de modélisation :

http://img.informer.com/icons/png/48/4790/4790677.png Pacestar UML Diagrammer est une alternative moins chère à d’autres programmes

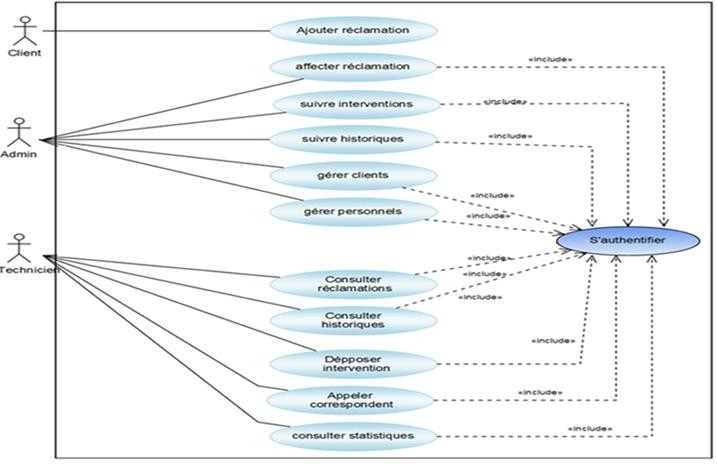
populaires qui aident surtout les programmeurs, les ingénieurs système pour concevoir des diagrammes UML. Il est produit avec un grand nombre de fonctionnalités, ainsi que des options de personnalisation et inclut tous les outils nécessaires pour aider les utilisateurs des schémas de conception à des fins diverses.

# Modélisation avec les diagrammes cas d’utilisation :

Le diagramme de cas d’utilisation permet de déterminer les possibilités d'interférence entre le système et les acteurs, c'est-à-dire déterminer toutes les fonctionnalités que doit fournir le système. Il permet aussi de délimiter ce dernier.

* + - Chaque usage effectué par les acteurs est représenté par un cas d'utilisation.
    - Chaque cas d'utilisation symbolise une fonctionnalité qui leur est offerte afin d’engendrer le résultat attendu.
    - 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.

# Diagramme de cas d’utilisation général :

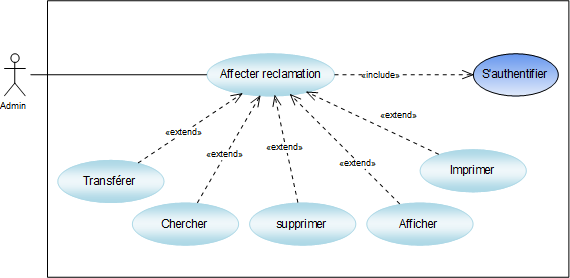


*Figure 6: Diagramme de cas d'utilisation général*

La figure 6 représente le diagramme de cas d’utilisation global de notre projet dont les acteurs sont : Le client, l’administrateur et le Technicien.

* Le client : permet de remplir un formulaire de réclamation
* L’administrateur : permets d’affecter les réclamations, suivre les interventions et les historiques, gérer les clients, gérer les personnels et gérer l’E-mail mais il ne peut pas effectuer ces tâches qu’après une authentification réussie.
* Le Technicien : est autorisé à exécuter les tâches suivantes : contacter l’administrateur ou les personnels, consulter les nouvelles réclamations, consulter les statistiques et l’authentification qui est nécessaire pour accéder au système.

# Diagramme de cas d’utilisation «Affecter réclamation» :

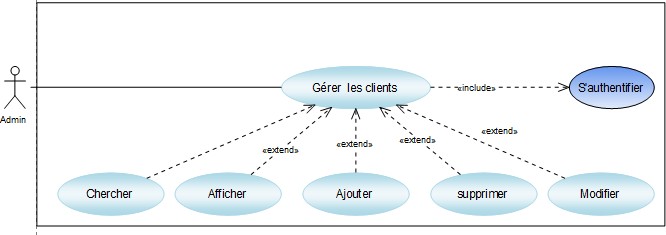


*Figure 7: diagramme de cas d’utilisation « Affecter réclamation »*

|  |  |
| --- | --- |
| **Cas d’utilisation** | **Affecter réclamation** |
| **Acteur** | * Administrateur |
| **Pré condition** | * Authentification |
| **Post condition** | * transfert, suppression, Affichage, et imprime de la liste des réclamations * Chercher une réclamation |
| **Description** | * L’administrateur affiche la liste des réclamations, il peut transférer les réclamations après les vérifications de l’état de factures (payée/impayée). * L’administrateur peut supprimer des réclamations. * L’administrateur peut chercher une réclamation |

*Tableau 2 : Cas d’utilisation «Affecter réclamation »*

# Diagramme de cas d’utilisation « Gérer les clients » :

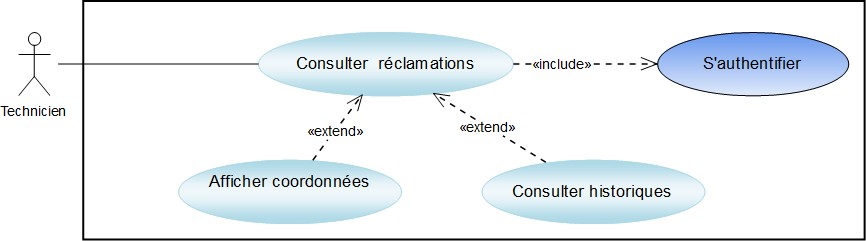


*Figure 8 : Diagramme de cas d’utilisation «Gérer les clients»*

|  |  |
| --- | --- |
| **Cas d’utilisation** | **Gérer les clients** |
| **Acteur** | * Administrateur |
| **Pré condition** | * Authentification |
| **Post condition** | * Affichage, ajout, suppression et modification de client * imprime de la liste des clients * Chercher un client |
| **Description** | * L’administrateur affiche une liste détaillée de clients qui ont déjà une ligne téléphonique. * L’administrateur peut ajouter, modifier et supprimer des clients * L’administrateur peut chercher un client à partir son N° d’appel ou par nom et prénom puis accéder à ses coordonnées * L’administrateur peut imprimer la liste de clients. |

*Tableau 3 : Cas d’utilisation « Gérer les clients »*

# Diagramme de cas d’utilisation «Consulter réclamations» :

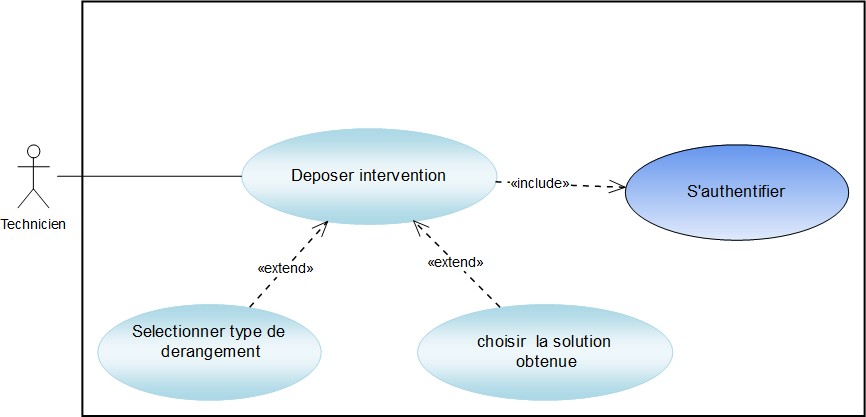


*Figure 9: Diagramme de cas d’utilisation « Consulter réclamations »*

|  |  |
| --- | --- |
| **Cas d’utilisation** | **Consulter réclamations** |
| **Acteur** | * Technicien |
| **Pré condition** | * Authentification |
| **Post condition** | * Etablissement des réclamations de dérangements |
| **Description** | * Le technicien accéder à liste de nouvelles réclamations de dérangements, il sélectionne la référence puis affiche toutes ses coordonnées. * Le technicien peut appeler le client directement puisque le N° d’appel et le N° de contact sont déjà affichés dans les coordonnées. * Le technicien peut accéder à l’historique de client. |
| **Exception** | * Si le client réclame pour la première fois, le système affiche « pas d’historique ». |

*Tableau 4 : Cas d’utilisation « Consulter réclamations »*

# Diagramme de cas d’utilisation «Déposer intervention» :

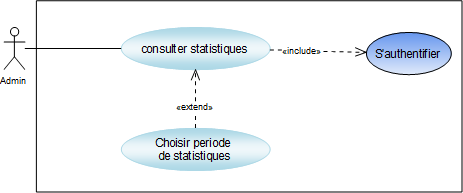


*Figure 10: cas d'utilisation "Déposer intervention"*

|  |  |
| --- | --- |
| **Cas d’utilisation** | **Déposer intervention** |
| **Acteur** | * Technicien |
| **Précondition** | * Consulter les réclamations |
| **Post condition** | * Déposer l’intervention |
| **Description** | * Le technicien choisit le type de dérangement à partir d’une liste. * Le technicien choisit son intervention effectuée le long du traitement. |

*Tableau 5: cas d'utilisation "Déposer intervention"*

# Diagramme de cas d’utilisation « consulter statistiques» :



*Figure 11: Diagramme de cas d'utilisation"consulter statistiques"*

|  |  |
| --- | --- |
| **Cas d’utilisation** | **Consulter statistiques** |
| **Acteur** | * Technicien |
| **Pré condition** | * Authentification |
| **Post condition** | * Affichage de statistiques |
| **Description** | * Le technicien peut accéder aux statistiques après la sélection d’une période. |

*Tableau 6: cas d’utilisation "Consulter les statistiques "*

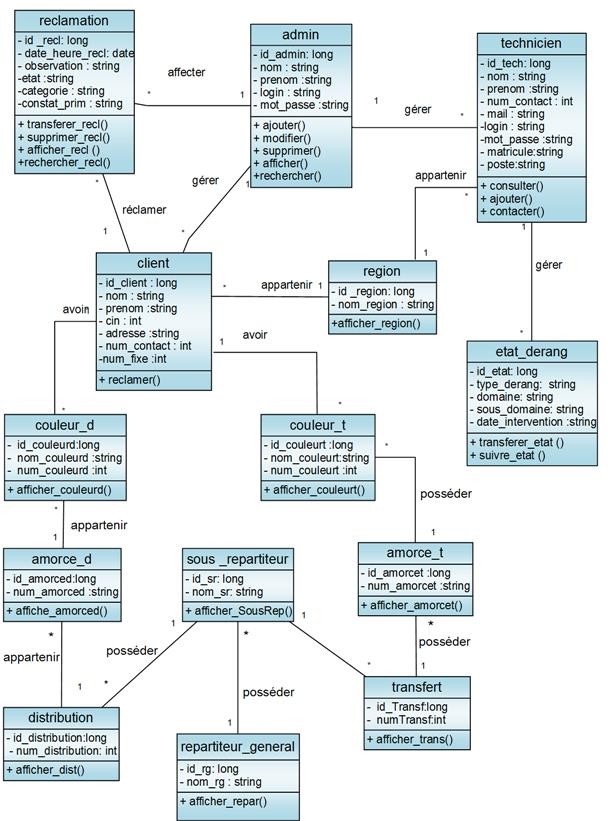
# Le diagramme de Classe :

* 1. **Définition :**

Un diagramme de classe est un diagramme UML qui contient des classes, des interfaces, des packages et leurs relations, et qui fournit une vue logique de tout ou partie d’un système informatique.

On construit un diagramme de classes pour simplifier l’interaction des objets d’un système qu’on est en train de modéliser. Ces diagrammes expriment la structure statique d’un système en termes de classes et de relations entre eux. Une classe décrit un ensemble d’objets et une association décrit un ensemble de liens. Un diagramme de classe n’exprime rien de spécifique concernant les liens d’un objet particulier, mais il décrit, le lien potentiel entre un objet et d’autres objets [5].

La figure 12 décrit le diagramme de classes de notre application



*Figure 12: Diagramme de classe*

# Présentation des classes :

Notre application comporte les classes suivantes :

* amorce\_t : Elle contient les numéros des amorces de transfert
* amorce\_d : Elle contient les numéros des amorces de distribution
* client : elle contient des informations relatives au client.
* couleur\_t : Elle contient le nom et le numéro de chaque couleur de transfert {blanc(1), bleu(2), jaune(3), marron(4), noir(5), rouge(6), vert(7)}.
* couleur\_d : Elle contient le nom et le numéro de chaque couleur de distribution {blanc(1), bleu(2), jaune(3), marron(4), noir(5), rouge(6), vert(7)}.
* distribution : Elle contient les numéros de tête de distribution.
* etat\_derang : Elle contient toutes les informations relatives à l’état de dérangement afin de déposer les interventions par le technicien.
* technicien : elle contient des informations relatives aux techniciens de Tunisie Telecom.
* réclamation : Elle contient les réclamations de dérangements envoyés par le client.
* region : Elle contient toutes les zones de Sousse.
* repartiteur\_general : Elle contient les noms de répartiteurs généraux.
* sous\_repartiteur : Elle contient les noms de sous-répartiteurs généraux.
* transfert : Elle contient les numéros de tête de transfert.
* admin : elle contient des informations relatives à l’administrateur

# Les classes, les attributs et les méthodes :

|  |  |  |
| --- | --- | --- |
| **Classe : amorce\_t** | | |
| **Attributs** | | |
| Nom | Type | Description |
| id\_amorcet | long | Un id unique pour chaque amorce de transfert |
| num\_amorcet | int | Numéro d’amorce de transfert |
| id\_transfert | long | Un id unique pour chaque tête de transfert |
| **Méthodes** | | |
| afficher\_amorcet () | void | Afficher les amorces de transfert |

*Tableau 7: Classe amorce\_t*

\*\*« id\_amorceT» représente la clé primaire de la table amorce\_t

\*\*« id\_transfert» représente la clé étrangère de la table amorce\_t

|  |  |  |
| --- | --- | --- |
| **Classe : amorce\_d** | | |
| **Attributs** | | |
| Nom | Type | Description |
| id\_amorced | long | Un id unique pour chaque amorce |
| num\_amorced | int | Numéro d’amorce |
| id\_distribution | long | Un id unique pour chaque tête de distribution |
| **Méthodes** | | |
| afficher\_ amorced () | void | Afficher les amorces de distribution |

*Tableau 8 : classe amorce\_d*

\*\*« id\_amorced» représente la clé primaire de la table amorce\_d

\*\*« id\_distribution» représente la clé étrangère de la table amorce\_d

|  |  |  |
| --- | --- | --- |
| **Classe : client** | | |
| **Attributs** | | |
| Nom | Type | Description |
| id client | long | Identifiant de chaque client |
| nom | String | Nom de client |
| prénom | String | Prénom de client |
| cin | int | Carte d’identité de client |
| adresse | String | Adresse de client |
| num\_contact | int | Numéro de contact de client |
| num\_fixe | int | Numéro fixe de client |
| id\_region | long | Un id unique pour chaque zone |
| id\_admin | long | Identifiant de l’admin |
| **Méthodes** | | |
| réclamer () | void | Ajouter une réclamation |

*Tableau 9 : classe client*

\*\*« id\_region» représente la clé primaire de la table client

\*\*« id\_admin» représente la clé étrangère de la table client

|  |  |  |
| --- | --- | --- |
| **Classe : couleur\_t** | | |
| **Attributs** | | |
| Nom | Type | Description |
| id\_couleurt | long | Un id unique pour chaque couleur coté transfert |
| nom\_couleurt | String | Nom de chaque couleur |
| num\_couleurt | int | Numéro identifiant de chaque couleur |
| id\_client | long | Identifiant de chaque client |
| **Méthodes** | | |
| afficher\_couleurt () | void | Afficher les couleurs de transfert |

*Tableau 10 : classe couleur\_t*

|  |  |  |
| --- | --- | --- |
| **Classe : couleur\_d** | | |
| **Attributs** | | |
| Nom | Type | Description |
| id\_couleurd | long | Un id unique pour chaque couleur coté distribution |
| nom\_couleurd | String | Nom de chaque couleur |
| num\_couleurd | int | Numéro identifiant de chaque couleur |
| id\_client | long | Identifiant de chaque client |
| **Méthodes** | | |
| afficher\_couleurd () | void | Afficher les couleurs de distribution |

*Tableau 11 : classe couleur\_d*

|  |  |  |
| --- | --- | --- |
| **Classe : distribution** | | |
| **Attributs** | | |
| Nom | Type | Description |
| id\_distribution | long | Un id unique pour chaque tête de distribution |
| num\_distribution | int | Numéro de chaque tête de distribution |
| id\_sr | long | Un id unique pour chaque sous répartiteur |

|  |  |  |
| --- | --- | --- |
| **Méthodes** | | |
| afficher \_dist () | void | Afficher les têtes de distribution |

*Tableau 12: classe distribution*

\*\*« id\_distribution» représente la clé primaire de la table distribution

\*\*« id\_sr» représente la clé étrangère de la table distribution

|  |  |  |
| --- | --- | --- |
| **Classe : etat\_derang** | | |
| **Attributs** | | |
| Nom | Type | Description |
| id\_etat | long | Un id unique pour l’état de dérangement |
| type\_derang | String | Le type de dérangement |
| domaine | String | Le domaine ou l’emplacement de dérangement |
| sous\_domaine | String | La solution obtenue pour le dérangement |
| date\_intervention | Date | La date de l’intervention effectuée par le technicien |
| id\_personnel | long | Identifiant de chaque personnel |
| **Méthodes** | | |
| transferer\_etat () | void | Envoyer un état de dérangement |
| suivre\_etat () | void | Suivre les états de dérangements |

*Tableau 13 :classe etat\_derang*

\*\*« id\_etat» représente la clé primaire de la table etat\_derang

\*\*« id\_technicien» représente la clé étrangère de la table etat\_derang

|  |  |  |
| --- | --- | --- |
| **Classe : technicien** | | |
| **Attributs** | | |
| Nom | Type | Description |
| id\_technicien | long | Identifiant de chaque technicien |
| nom\_technicien | String | nom de technicien |
| prenom\_technicien | String | prénom de technicien |
| num\_contact | int | numéro de contact (téléphone) de chaque technicien |

|  |  |  |
| --- | --- | --- |
| mail | String | Email de technicien |
| login | String | Login de technicien |
| mot\_passe | String | Mot de passe de chaque technicien |
| matricule | String | Code pour chaque technicien |
| id\_admin | long | Identifiant de l’admin |
| **Méthodes** | | |
| consulter () | void | Consulter les données |
| ajouter () | void | Ajouter des interventions |
| contacter () | void | Contacter les personnels |

*Tableau 14: classe technicien*

\*\*« id\_technicien» représente la clé primaire de la table technicien

\*\*« id\_admin» représente la clé étrangère de la table technicien

|  |  |  |
| --- | --- | --- |
| **Classe : réclamation** | | |
| **Attributs** | | |
| Nom | Type | Description |
| id\_recl | long | Un id unique pour chaque réclamation |
| date\_heure\_recl | date | Date et heure de réclamation |
| observation | String | Observation de dérangement observé par le client |
| catégorie | String | Catégorie de la ligne |
| Constat\_prim | String | Constatation primaire de type de dérangement |
| etat | String | L’etat de réclamation |
| id\_client | long | Identifiant de chaque client |
| id\_etat | long | Un id unique pour l’état de dérangement |
| id\_admin | long | Identifiant de l’admin |
| **Méthodes** | | |
| ajouter\_recl () | void | Ajouter la réclamation |
| supprimer\_recl () | void | Supprimer la réclamation |
| afficher\_recl () | void | Afficher la liste des réclamations |
| rechercher\_recl () | void | Rechercher des réclamations |

*Tableau 15 : classe réclamation*

\*\*« id\_recl» représente la clé primaire de la table reclamation

\*\*« id\_client» représente la clé étrangère de la table reclamation

\*\*« id\_etat» représente la clé étrangère de la table reclamation

\*\*« id\_admin» représente la clé étrangère de la table reclamation

|  |  |  |
| --- | --- | --- |
| **Classe : région** | | |
| **Attributs** | | |
| Nom | Type | Description |
| id\_region | long | Un id unique pour chaque zone |
| nom\_région | date | Nom de la région |
| **Méthodes** | | |
| afficher\_reg () | void | Afficher la liste des régions |

*Tableau 16 : classe région*

|  |  |  |  |
| --- | --- | --- | --- |
| **Classe : repartiteur\_general** | | | |
| **Attributs** | | | |
| Nom | | Type | Description |
| id\_rg | | long | Un id unique pour chaque répartiteur général |
| nom\_rg | | String | Nom de chaque répartiteur général |
| **Méthodes** | | | |
| afficher\_repar () | void | | Afficher la liste des répartiteurs généraux |

*Tableau 17 : classe repartiteur\_général*

|  |  |  |
| --- | --- | --- |
| **Classe : sous\_repartiteur** | | |
| **Attributs** | | |
| Nom | Type | Description |
| id\_sr | long | Un id unique pour chaque sous répartiteur |
| nom\_sr | String | Nom de sous répartiteur |
| id\_rg | long | Un id unique pour chaque répartiteur général |
| **Méthodes** | | |
| afficher\_sousRep () | void | Afficher la liste des sous répartiteurs |

*Tableau 18 : classe sous\_repartiteur*

\*\*« id\_sr» représente la clé primaire de la table sous\_repartiteur

\*\*« id\_rg» représente la clé étrangère de la table sous\_repartiteur

|  |  |  |
| --- | --- | --- |
| **Classe : transfert** | | |
| **Attributs** | | |
| Nom | Type | Description |
| id\_transfert | long | Un id unique pour chaque tête de transfert |
| num\_transfert | int | Numéro de chaque tête de transfert |
| id\_sr | long | Un id unique pour chaque sous répartiteur |
| **Méthodes** | | |
| afficher\_transf () | void | Afficher une tête de transfert |

*Tableau 19 : classe transfert*

\*\*« id\_transfert» représente la clé primaire de la table transfert

\*\*« id\_sr» représente la clé étrangère de la table transfert

|  |  |  |
| --- | --- | --- |
| **Attributs** | | |
| Nom | Type | Description |
| id\_admin | long | Identifiant de chaque technicien |
| nom | String | nom de technicien |
| prenom | String | prénom de technicien |
| mail | String | Email de technicien |
| login | String | Login de technicien |
| mot\_passe | String | Mot de passe de chaque technicien |
| **Méthodes** | | |
| ajouter () | void | Ajout des données |
| modifier () | void | Modification des données |
| supprimer () | void | Suppression des données |
| rechercher () | void | Rechercher des données |

*Tableau 20: classe admin*

# Les diagrammes de séquences :

1. **Définition :**

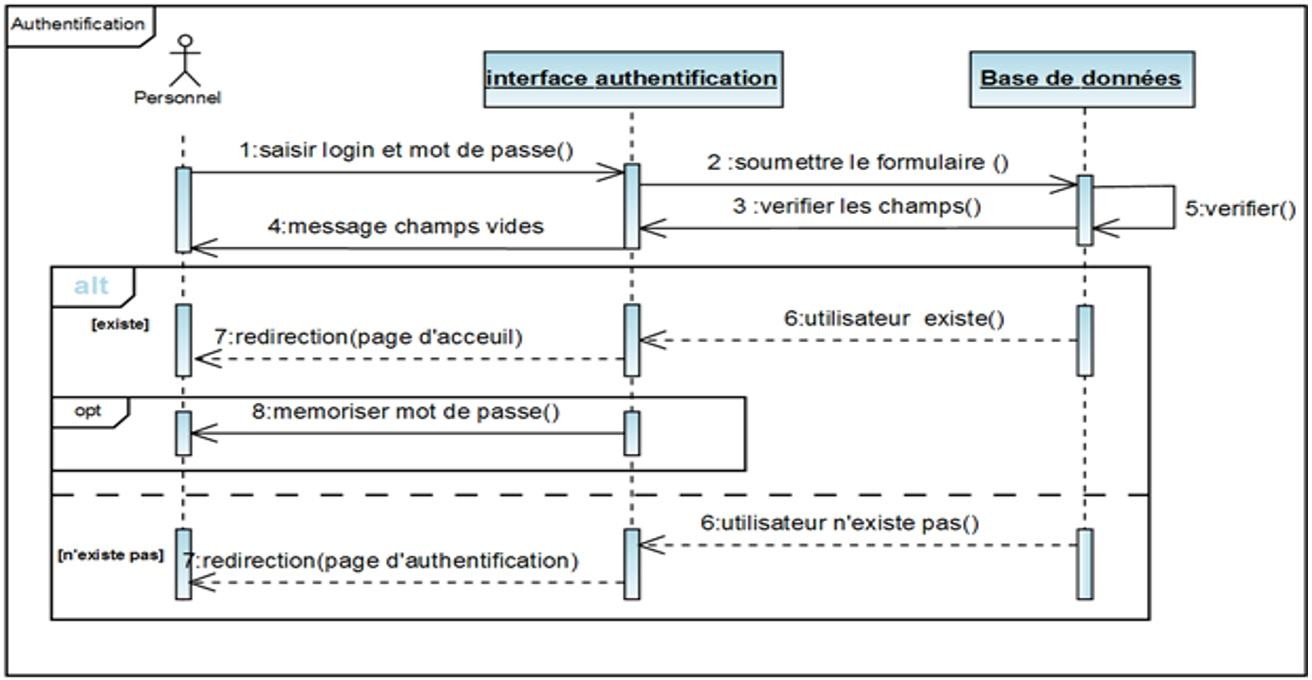
Le diagramme de séquence, décrit les scénarios de chaque cas d'utilisation en mettant l'accent sur la chronologie des opérations en interaction avec les objets. Un diagramme de séquence montre une interférence présentée en séquence dans le temps. En particulier, il montre aussi les objets qui participent à l'interaction par leur "ligne de vie" et les messages qu'ils échangent présentés en séquence dans le temps.

Voici quelques notions de base du diagramme :

* Scénario : une liste d'actions qui décrivent une interaction entre un acteur et le système.
* Interaction : un comportement qui comprend un ensemble de messages échangés par un ensemble d'objets dans un certain contexte pour accomplir une certaine tâche.
* Message : Un message définit une communication particulière entre des lignes de vie (objets ou acteurs) [6].

# Diagramme de séquences « authentification » :

Le diagramme qui suit, figure 13, présente l'enchainement de la phase d'authentification.



*Figure 13: Diagramme de séquence «Authentification »*

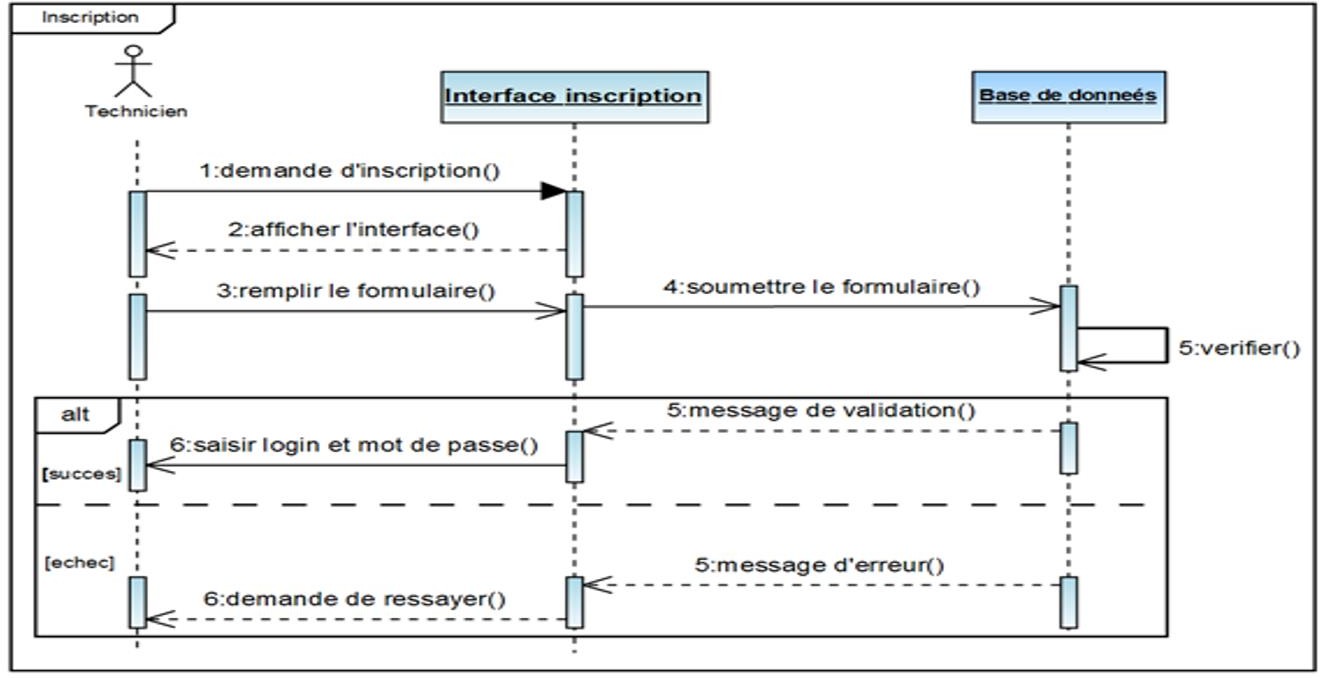
#### Avant d’accéder à l’accueil de l’application et faire l'ensemble des autres scénarios l'utilisateur doit se connecter en utilisant son login et son mot de passe.

* + **Acteur** : Technicien/Admin
  + **Objectif :** la sécurité et la confidentialité de l’accès à l’application
  + **Description :**

Pour accéder à l’application, l’utilisateur doit tout d'abord s'identifier par son login et son mot de passe via le système qui prend en charge de vérifier les champs saisis par le technicien dans la base de données. S'il est accepté, donc il aura accès au système et aux applications du menu correspondant. Sinon, il doit vérifier ses données et s’identifie de nouveau s’il a déjà s’inscrire.

# Diagramme de séquences « Inscription» :

Le diagramme qui suit, figure 14, présente l'enchainement de la phase d’inscription



*Figure 14: Diagramme de séquence «Inscription »*

* **Acteur** : Technicien
* **Objectif** : créer un compte

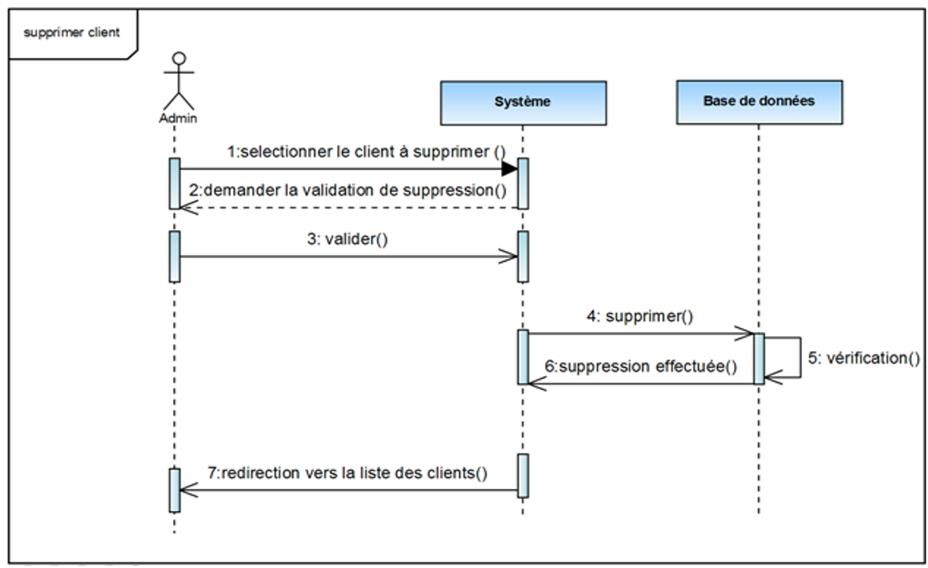
#### Description :

Pour créer un compte, le système vérifie d’abord si les données saisie par l’utilisateur ne sont pas déjà enregistrées, une fois les vérifications terminées avec sucées l’utilisateur peut

accéder à l’application par son login et son mot de passe sinon il doit ressayer à nouveau.

# Diagramme de séquences « Supprimer client» :

Le diagramme qui suit, figure 15, présente l'enchainement de la phase de suppression de client



*Figure 15: Diagramme de séquence "Supprimer client "*

* **Acteur** : Admin
* **Objectif** : supprimer un client

#### Description :

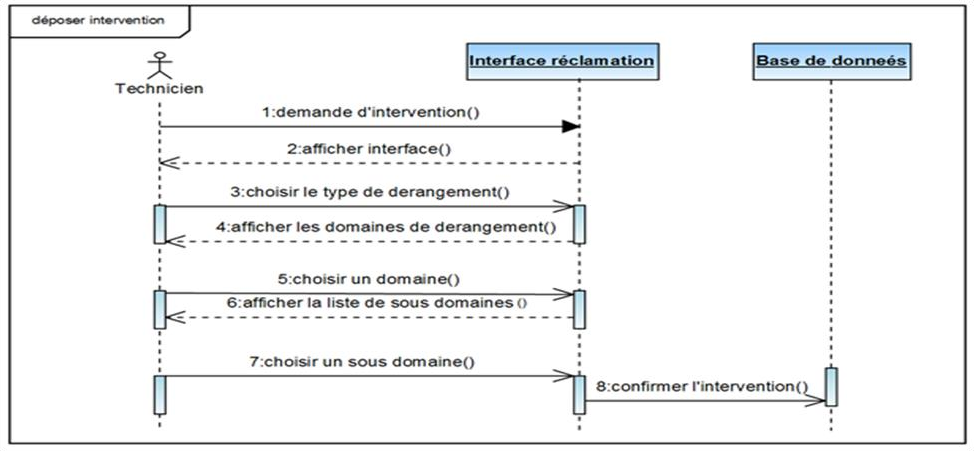
Afin d’établir la suppression de client, l’administrateur doit sélectionner le client approprié et par la suite un message de validation affiché pour la confirmation. La base de données fait la vérification et puis supprime le client.

Client est supprimé avec succès.

**N.B** : Ce diagramme montre la suppression de client, le même principe pour les autres interfaces tels que : personnels et réclamations.

# Diagramme de séquences «Déposer intervention» :

Le diagramme qui suit, figure 16, présente l'enchainement de la déposition d’intervention



*Figure 16: Diagramme de séquence « déposer intervention»*

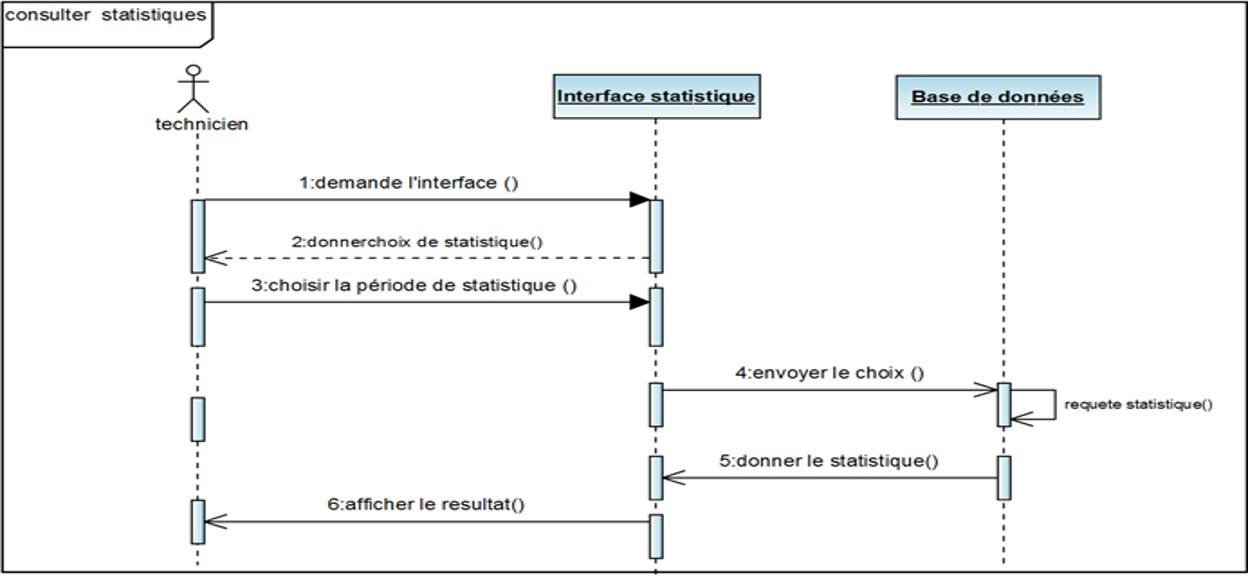
* + **Acteur** : Technicien
  + **Objectif** : déposer l’intervention

#### Description :

Le technicien demande de faire son intervention , il doit choisir le type de dérangement puis selectionne tous les données correspondant à partir d’une liste de choix afin de l’envoyer vers la base de donnéees pour les stockés.

# Diagramme de séquences de système «Consulter statistique» :

Le diagramme qui suit, figure 17, présente l'enchainement de la consultation de statistiques.



*Figure 17 : Diagramme de séquence « consulter statistiques»*

* + **Acteur** : Technicien
  + **Objectif** : Consulter statistique

#### Description :

Le technicien consulter l’interface de statistique et choisit le type correspondant et le valider. Une requête va être envoyé afin d’afficher le résultat correspondant.

# Conclusion :

Ce chapitre a été consacré à la modélisation UML du système .Nous avons présenté différents diagrammes : les diagrammes de cas d’utilisation, de classe , et de séquences afin de spécifier de façon détaillée les aspects fonctionnels, dynamiques et statiques du système. Dans le chapitre suivant, nous présenterons la réalisation et la mise à l’essai de ce système.





# Introduction :

Nous arrivons maintenant à la phase ultime. Cette dernière partie est la plus importante puisqu’elle met en réalité toute la théorie précédente. Dans un premier temps nous présentons l’environnement de réalisation sur le plan logiciel. Dans un second temps nous présentons quelques interfaces de notre site web et de notre application Android ainsi que quelques scripts considérées les plus importantes.

# Environnement de travail :

Dans cette partie, nous présentons les différents outils matériels et logiciels nécessaires pour le développement de notre application.

# Environnement matériel :

Nous mentionnons les caractéristiques de nos ordinateurs sur lesquelles nous avons développé l’application parce qu’elles peuvent donner une idée sur les conditions du travail.

Donc l’application a été développée sur deux ordinateurs portables dont les caractéristiques principales sont :

#### 1er ordinateur

* + **Nom :** ASUS
  + **Processeur :** Intel(R) Coré (TM) i54210U CPU @ 1.70GHz 2.40 GHz
  + **Mémoire installée (RAM) :** 8,00Go (7.89GO utilisable)
  + **Type de système :** Système d’exploitation 64bits

#### 2èm ordinateur

* + **Nom :** DELL
  + **Processeur : Intel** (R) Pentium(R) CPU 2117U@ 1.8 GHz 1.8 GHz
  + **Mémoire installée (RAM) :** 4.00Go (3,88GO utilisable)
  + **Type de système :** Système d’exploitation 64bits

# Environnement logiciel :

Nous avons énuméré au cours de cette partie les différents outils utilisés tout au long de ce projet pour l’étude et la mise en place de notre application.

# Système d’exploitation :

Nous avons utilisé comme système d’exploitation :

* Microsoft Windows 8.1 Professionnel
* Microsoft Windows 7 Edition Intégral.

# Outils de développement :

* + - * JDK 1.8.0\_74

Le Java Développent Kit (JDK) désigne un ensemble de [bibliothèques logicielles](http://fr.wikipedia.org/wiki/Biblioth%C3%A8que_logicielle) de base du [langage de programmation](http://fr.wikipedia.org/wiki/Langage_de_programmation) [Java](http://fr.wikipedia.org/wiki/Java_(langage)), ainsi que les outils avec lesquels le code Java peut être compilé, transformé en [byte code](http://fr.wikipedia.org/wiki/Bytecode) destiné à la machine virtuelle java. [7]

# GenyMobile version 2.6.0 :

Genymotion est un émulateur Android basé sur **Oracle VM Virtual Box** ((anciennement **VirtualBox**) est un logiciel libre de virtualisation publié par Oracle).

Genymotion a l'avantage d'être plus rapide que l'émulateur du Kit de développement Android et fournit de meilleures performances générales. Cet outil requiert une configuration supplémentaire [8].

Figure 18: Logo de l'émulateur GenyMotion

* + 1. Sublime Text :

Sublime Text est un éditeur de texte générique, mais clairement orienté programmation, développé en C++ et Python. À l'origine raisonnée comme une extension de [Vim](http://fr.wikipedia.org/wiki/Vim) (ligne de commande), il est ensuite devenu un logiciel à part entière, et s'utilise dorénavant en interface graphique. Il dispose de toutes les fonctionnalités classiques d'un éditeur de texte, comme la

coloration syntaxique, et l'auto-complétion de code, mais ce qui fait surtout sa force, c'est sa modularité grâce à son système de plugins [9].

# JSON :

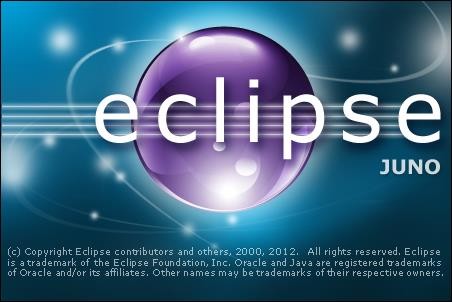
JSON est de « JavaScript Object Notion », car c’est un dérivé de la représentation littérale d’un objet en JavaScript défini par L’ECMAScript Programming Language Standard.

Le JSON n’est pas un langage ni une forme de magie obscure et laborieuse, c’est juste un format de données, c’est-à-dire un texte à plat qui respecte une certaine structure pour véhiculer facilement et légèrement les informations.

JSON est un format texte complètement indépendant de tout langage, mais les conventions qu’il utilise seront familières à tout programmeur habitué aux Language descendant du C, par exemple : C lui-même++, C#, Java, JavaScript, Pert, Python et bien d’autres. Ces propriétés font de JSON un langage d’échange de données idéal.

# Environnement de développement intégré :

* + - * Eclipse

Eclipse est un IDE (environnement de développent intégré) écrit en java, extensible par des greffons, multi-langages et multi-plates-formes, qui s’intègre particulièrement bien à GNOME. C’est un [projet,](http://fr.wikipedia.org/wiki/Projet) organisé en un ensemble de sous-projets de développements logiciels, de la [fondation Eclipse](http://fr.wikipedia.org/wiki/Fondation_Eclipse) visant à développer un environnement de production de logiciels [libres](http://fr.wikipedia.org/wiki/Logiciel_libre) qui soit extensible et universel, en s'appuyant principalement sur [Java.](http://fr.wikipedia.org/wiki/Java_(technique)) On utilise pour l’application Android, **Juno** est la version 4.2 de la plateforme, et est le premier à être construit sur Eclipse4. Pour pouvoir préparer l'environnement de développement sur Android, un plugin nommé ADT - fourni par Google pour cet IDE- doit être installé pour bien faciliter le développement [10].

*Figure 19: Logo d’Eclipse JUNO*

Pour l’application Web on utilise Eclipse Mars : Eclipse 1, baptisé **Mars**, est disponible en téléchargement et promet de séduire encore plus de développeurs.



*Figure 20: Logo Eclipse MARS*

# Outil d’administration de la base de données

L'implémentation de notre application se fera avec MYSQL sous l'environnement **EasyPHP**. Il installe et configure automatiquement un environnement de travail complet sous Windows permettant de mettre en œuvre toute la puissance qu’offrent le langage dynamique PHP et son support efficace des bases de données.

EasyPHP est un environnement comprenant deux serveurs (un serveur web Apache et un serveur de bases de données MySQL), un interpréteur de script (PHP), ainsi qu'une administration SQL **phpMyAdmin.**

*Figure 21: Logo EasyPHP*

# Serveur web Apache-Tomcat 7.0

Apache TomCat est une implémentation open source de Java Servlet et Java Server Pages technologies. Le Java Servlet et Java Server Pages spécifications sont développées dans le cadre du Java Community Process.

# Langage de programmation :

### JAVA :

L’application a été développée en JAVA. En fait, ce langage de programmation est avantageux par rapport aux autres langages car les programmes peuvent être exécutés sur différents systèmes d’exploitation et architectures matérielles. De plus, il permet de créer facilement des interfaces graphiques ergonomiques.

### HTML

HTML (Hyper Text Markup Language / langage hypertexte) est le langage dans lequel sont écrites les pages du web. Un site web est constitué d’un ou plusieurs documents HTML. Pour se déplacer entre les pages dans nos modules on passe par l’intermédiaire d’hyperliens. Pour ajouter des objets graphiques on utilise le HTML d’autre part pour tester des pages web html en local, il suffit d’ouvrir le fichier dans un navigateur. Le HTML n’est pas un langage de programmation comme C++.

### JavaScript

JavaScript est un langage de programmation de scripts principalement utilisé pour les pages web interactives comme les pages HTML. JavaScript est exécuté sur l’ordinateur de l'internaute par le navigateur lui-même. C’est une extension du langage HTML qui est incluse dans le code. Ce langage est un langage de programmation qui permet d’apporter des améliorations au langage HTML en permettant d’exécuter des commandes. Ce code est directement écrit dans la page HTML, c’est un langage peu évolué qui ne permet aucune confidentialité au niveau des codes. Dans l’application nous avons codé plusieurs fonctions JavaScript par exemple : pour l’interaction des pages en envoyant des variables dans l’adresse URL pour filtrer le résultat de la requête en utilisant la méthode POST ou GET.

### CSS

Les CSS, Cascading Style Sheets (feuilles de styles en cascade), permettent de mettre en forme des documents web, type page HTML ou XML. Par l’intermédiaire de propriétés d’apparence (couleurs, bordures, polices, etc.) et de placement (largeur, hauteur, côte à côte, dessus, dessous, etc.), le rendu d’une page web peut être entièrement modifié sans aucun code supplémentaire dans la page web. Les feuilles de styles ont d’ailleurs pour objectif principal de séparer le contenu de la page de son apparence visuelle [11].

# Développement :

### Choix de la plateforme JEE :

Pour le développent de notre application, nous avons utilisé la plateforme JEE (Java Edition Entreprise), est une spécification pour la technique Java de Sun plus particulièrement destinée aux

applications d’entreprise. Ces applications sont considérées dans une approche multi-niveaux. Dans ce but, toute implémentation de cette spécification contient un ensemble d’extensions au Framework Java standard (JSE, Java Standard Edition) pour faciliter la création d’applications réparties.

Les JDK spécifiques à Java EE sont conçus de façon à ce qu'une application réalisée avec Java EE fonctionne sur le même **JRE** qu'une application écrite avec Java SE, mais nécessitera cependant qu'en complément, les bibliothèques exploitées soient fournies par un « conteneur Java » lourd telles **Apache Tomcat** ou léger tel que **Spring** que nous avons déjà l’utilisé.

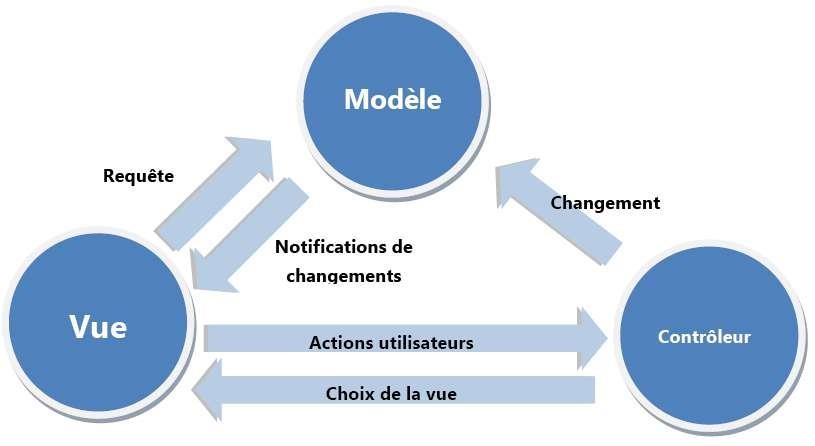


*Figure 22: Logo Java EE*

# Choix de développement :

Pour le développement, nous avons appliqué le modèle MVC. Ce paradigme divise l’IHM (Interface Homme Machine) en un **modèle** (modèle de données) une **vue** (la présentation, l’interface utilisateur) et un **contrôleur** (la logique de contrôle, et la gestion des événements / synchronisation), chacun a un rôle bien précis.

L'architecture MVC ne résout pas tous les problèmes. Elle fournit souvent une première approche qui peut ensuite être adaptée et elle offre aussi un cadre pour structurer une application.

Ce maitre d'architecture impose la séparation entre les données, la présentation et les traitements, ce qui nous donne trois parties fondamentales dans l'application : le modèle, la vue et le contrôleur.

*Figure 23: Schéma du Modèle MVC*

# Le modèle :

Le modèle indique le comportement de l'application : traitements des données, interactions avec la base de données. Il décrit l’emplacement des données manipulées par l'application et assure la gestion de ces données et garantit leur intégrité. Dans le cas spécifique d'une base de données, c'est le modèle qui la contient. Le modèle offre des méthodes pour mettre à jour ces données (ajout, suppression, changement de valeur). Il offre aussi des méthodes pour récupérer ces données. Les résultats renvoyés par le modèle sont dénués de toute présentation.

# La vue :

La vue correspond à l'interface avec laquelle l'utilisateur interagit. Sa première tâche est de présenter les résultats renvoyés par le modèle. Sa seconde tâche est de recevoir toutes les actions de l'utilisateur (sélection d'une entrée, boutons, etc.). Ces événements sont envoyés au contrôleur. La vue n'effectue aucun traitement, elle se satisfait d'afficher les résultats des traitements effectués par le modèle et d'interagir avec l'utilisateur.

# Le contrôleur :

Le contrôleur prend en charge la gestion des événements de synchronisation pour mettre à jour la vue ou le modèle et les synchroniser. Il reçoit tous les événements de l'utilisateur et enclenche les actions à effectuer. Si une action nécessite un changement des données, le contrôleur demande la modification des données au modèle, et ce dernier informe la vue que les données ont changée pour qu'elle les mette à jour.

# Les technologies utilisées :

* ***Couche de présentation :***

Dans la couche de présentation nous utilisons :

* **AngularJS**

AngularJS est un Framework écrit en JavaScript permet d'améliorer, au même titre que [JQUERY,](http://creersonsiteweb.net/page-apprendre-jquery) la syntaxe de JavaScript ainsi que la productivité du développeur. Il se base sur la logique MVC.

* **JQuery**

JQuery est une librairie [JavaScript](http://creersonsiteweb.net/page-jquery-javascript) libre qui fait interagir JavaScript et [HTML](http://creersonsiteweb.net/page-apprendre-html) et dont l'intérêt est de réduire les commandes les plus utilisées. Créée en 2006, elle est depuis utilisée sur de nombreux sites comme Google, Mozilla etc.

* **Bootstrap**

Bootstrap est un ensemble d’outils utiles à la création de sites et d’applications web. C’est un ensemble qui contient des codes HTML et CSS, outils de navigations et autres éléments interactifs, ainsi que des extensions JavaScript en option.

* + ***Couche métier :***
* **Spring**

Dans cette couche nous utilisons le Framework Spring, c’est un [Framework](http://fr.wikipedia.org/wiki/Framework) [libre](http://fr.wikipedia.org/wiki/Libre) pour construire et définir l’infrastructure d’une application javaEE, dont il facilite le développement et les tests. Spring est considéré comme un conteneur dit « léger » c’est-à-dire une infrastructure similaire à un [serveur d’application](http://fr.wikipedia.org/wiki/Serveur_d%27application) [JEE](http://fr.wikipedia.org/wiki/Java_2_Enterprise_Edition) . Il prend donc en charge la création d’objets et la mise en relation d’objet par l’intermédiaire d’un fichier de configuration qui décrit les objets à fabriquer et les relations de dépendances entre ces objets.

* + ***Couche accès aux données :***
* **Les micros-services : Spring boot**

Les micros servis peuvent être créés pour des objectifs très variés, mais ce qui est sûr, c’est que la plupart auront besoin de lire et d’écrire dans une base de données. Spring boot permet de mettre en place l’intégration d’une base de données de manière très simple en offrant un accès Spring Data aux bases de données auto configurables. En incluant simplement le module Spring\_boot\_start\_data\_jpa dans le projet, le moteur d’auto configuration de Spring Boot détecte que le projet nécessite un accès à une base de données, et crée les beans nécessaires dans le contexte applicatif Spring.

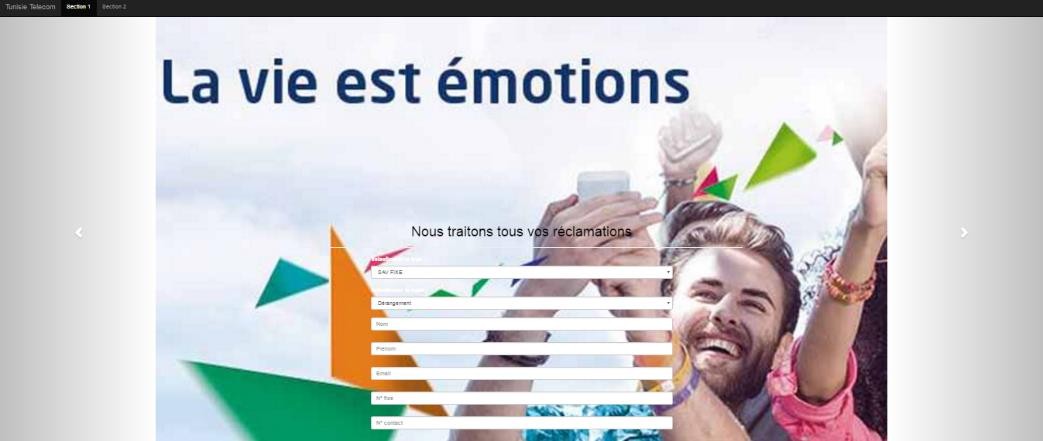
# Les interfaces graphiques :

L’interface graphique est une partie très importante pour la réalisation d’une application convenable offrant un certain plaisir à l’utilisateur lors de sa navigation. Ainsi, ce critère peut faire la différence entre une application et une autre bien qu’elles aient les mêmes fonctionnalités.

Voici un ensemble de captures d’écrans sur les principaux points d’entrées de l’application :

* 1. **Partie client :**

# Formulaire de réclamation :



*Figure 24: Formulaire de réclamation*

Si la ligne téléphonique est dérangée, le client peut réclamer en remplissant un formulaire par toutes les coordonnées nécessaires telles que nom, prénom, numéro fixe …puis en l'envoyant vers l’admin (espace réclamation).

* 1. **Partie administrateur :**

# Interface Authentification :

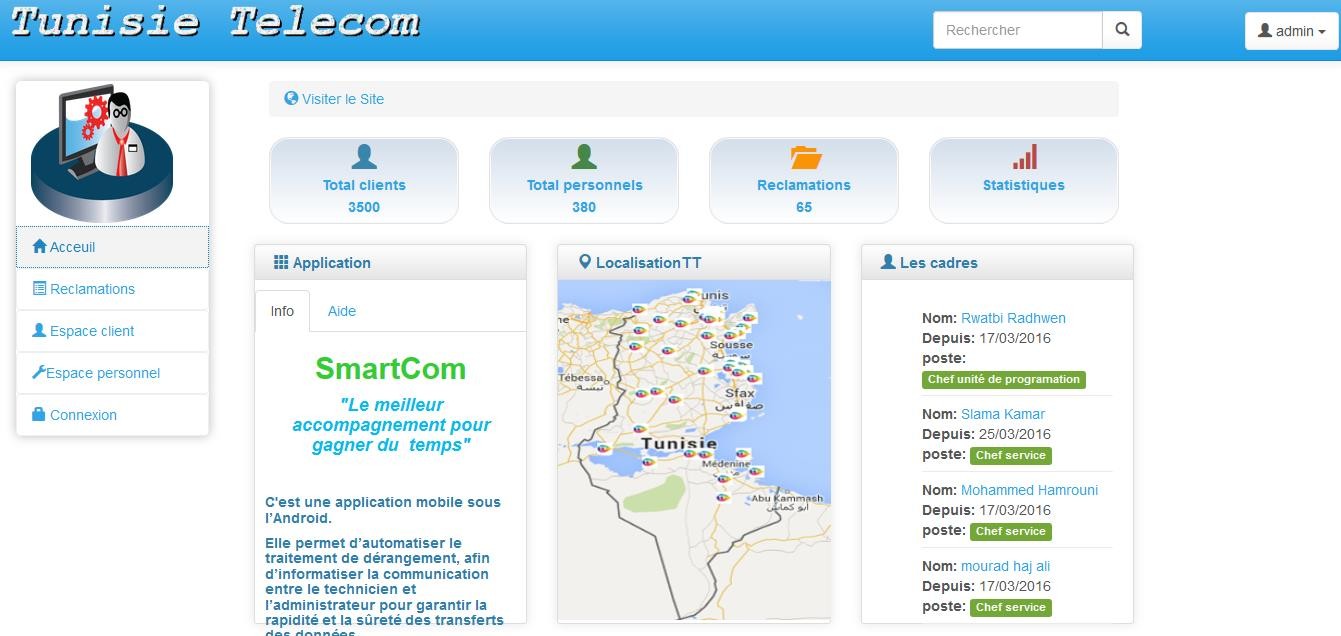


*Figure 25: Interface authentification*

Pour pouvoir accéder aux différentes fonctionnalités de l’application, l’administrateur doit taper son login et son mot de passe dans les champs correspondants. Une fois que le client a cliqué sur le bouton «Login », le système vérifie les données entrées. En cas d’échec, il réaffiche la page

d’authentification avec un message d’erreur. Si le Login et le mot de passe sont acceptables, le système passe au menu principal.

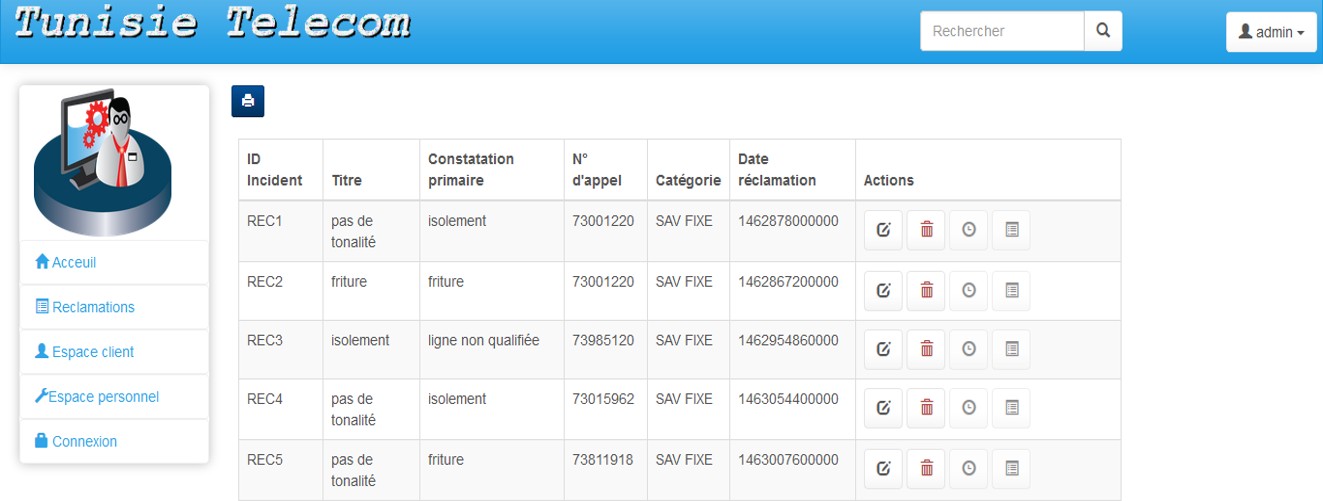
# Interface d’accueil :



*Figure 26: Interface d'accueil*

Cette figure représente l’accueil de l’application.

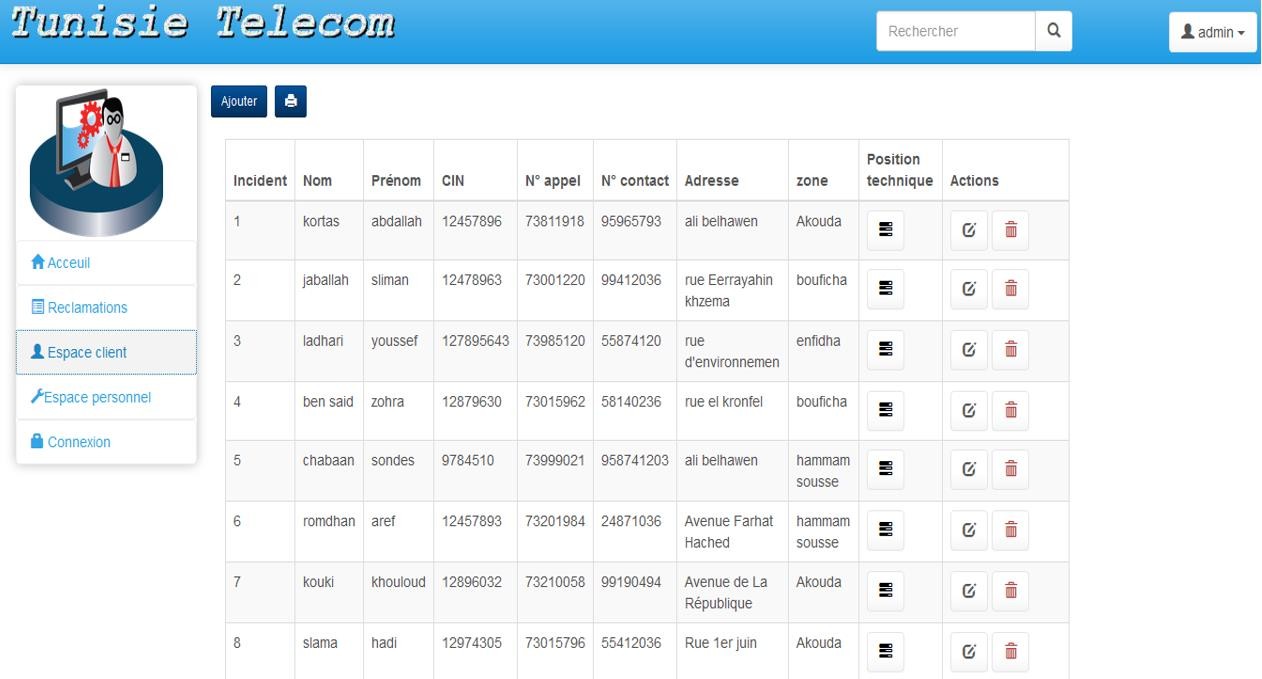
# Interface Consulter la liste des réclamations :



*Figure 27: Interface Réclamations*

Cette interface permet à l’administrateur de consulter, imprimer et supprimer une réclamation. Chaque réclamation comporte les coordonnées nécessaires, Il permet aussi de transférer la réclamation, d’afficher l’historique et de consulter l’intervention de technicien.

# Interface Consulter la liste des clients :



*Figure 28: Interface liste des clients*

Cette interface permet à l’administrateur de consulter, imprimer, modifier et supprimer un client. Pour modifier un employé, l’administrateur remplit les champs à modifier puis il clique sur le bouton valider et pour supprimer un client il suffit de choisir la colonne puis cliquer sur l’icône de suppression.

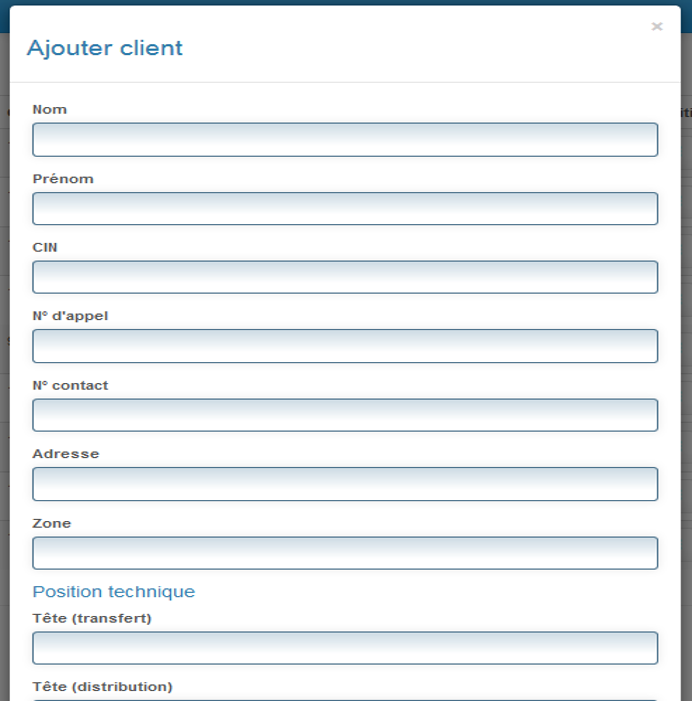
# Interface afficher la position technique :



*Figure 29: Interface de la position technique*

Cette interface permet d’afficher la position technique de chaque client ; c’est-à-dire le nom de répartiteur général, le numéro de sous répartiteur, les têtes, les amorces et les couleurs convenables.

# Interface Ajouter un client :



*Figure 30: Interface Ajouter un client*

Au cours de cette interface, l’administrateur peut ajouter un client en remplissant tous les champs pour gérer les nouveaux clients.

# Interface supprimer un client :

*Figure 31: message de suppression*

L’administrateur peut supprimer un client de la liste. Un clic sur l’image  affiche un message pour confirmation comme la montre la figure au-dessus.

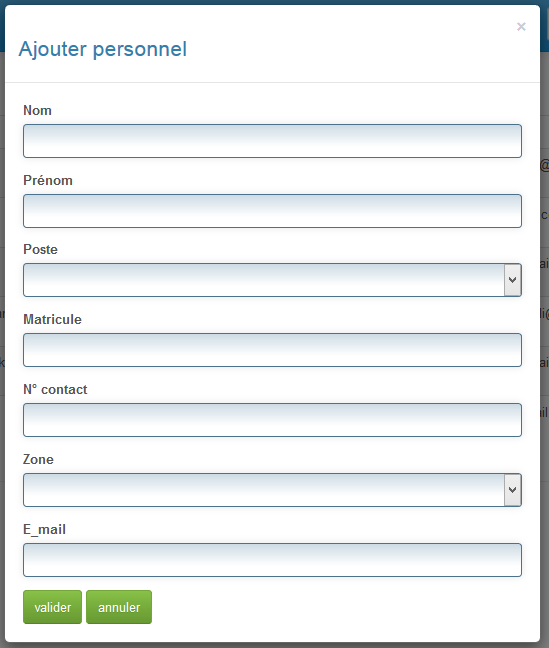
# Interface consulter la liste des personnelles :



*Figure 32: Interface consulter la liste des personnelles*

Cette interface permet à l’administrateur de consulter, imprimer, modifier et supprimer un personnel.

# Interface ajouter personnelle :



*Figure 33: Interface ajouter personnelle*

Cette interface contient toutes les informations pour gérer des nouveaux personnels tels que nom, prénom, poste, matricule, numéro de contact, la zone et l’E-mail.

* 1. **Partie Technicien :**

# Interface Authentification :

*Figure 34: Interface Authentification Figure 35: Interface inscription*

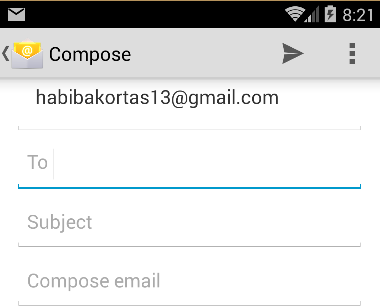
La figure 34 représente l’interface d’authentification dont laquelle le technicien doit saisir le login et le mot de passe pour accéder à l'espace d’accueil de l'application mais il doit passer tout d’abord par l’interface d’inscription comme la montre la figure 35 et remplir tous les champs afin qu’il puisse accéder à l’application.

# Interface Accueil :

*Figure 36: interface accueil Figure 37: Rubrique info et aide*

Après avoir saisir le login et le mot de passe, la figure 36 s’affiche, elle contient le menu principal : un bouton pour contacter l’admin et les personnels, un bouton pour afficher la liste des réclamations et un bouton pour consulter les statistiques. Alors que la figure 37 présente la rubrique info et la rubrique aide.

# Interface Contacter :





*Figure 38: Interface contacter*



Pour contacter l’admin ou les personnels, le technicien doit tout d’abord cliquer sur le bouton pour accéder à l’interface de contact comme l’indique la figure 38 et par la suite, il a le choix de les contacter que ce soit par téléphone ou par mail.

# Interface de la liste des réclamations :

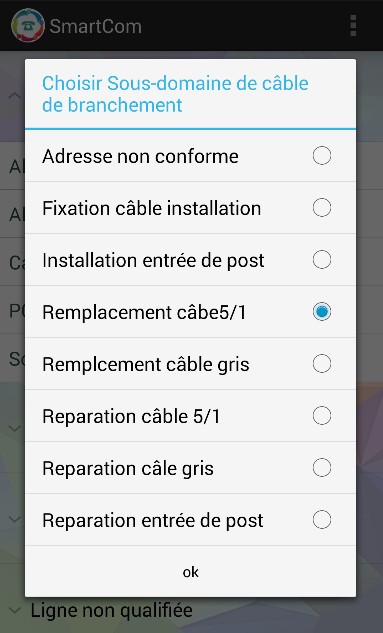


Si on clique sur le bouton , le technicien accède à la liste des réclamations comme l’indique la figure 39 au-dessous, dont elle affiche les références et les heures exactes des chaque réclamation et par la suite il sélectionne l’une des réclamations affichées dans la figure 39 pour consulter les coordonnées de chaque client, qui sont représentées dans la figure 40.

*Figure 39: Interface de la liste des réclamations* *Figure 40: Interface de coordonnées de client*

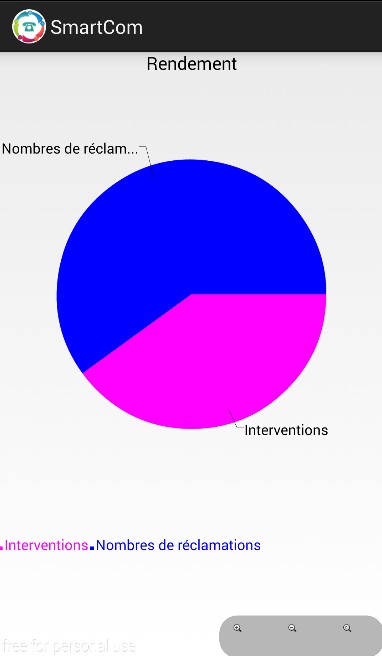
# Interface d’intervention :



*Figure 41: Interface d'intervention*

Après le traitement effectué par le technicien, il va déposer son intervention, donc il sélectionne le bouton de la figure 40, à partir d’une liste affichée, il choisit le type de dérangement puis le domaine, et par la suite une interface s’affiche qui contient tous les sous domaines (solutions obtenues) possibles .il sélectionne l’un d’entre eux et l’envoyer vers la base de données.

# Interface de statistiques :



*Figure 42: Interface de statistiques*

Le technicien choisit les statistiques périodiques à partir d’une liste de choix pour évaluer son rendement, puis une représentation graphique s’affiche (figure 42), qui présente les nombres d’interventions par rapport aux nombres des réclamations.

# Conclusion :

La dernière partie était dédiée à la navigation dans notre application. Elle constitue le dernier volet de ce rapport, elle a pour objet de présenter l’environnement logiciel et matériel de réalisation, et de décrire les principales fonctionnalités implémentées suite au développement par des captures écrans présentant les principes interfaces des applications.



L’élaboration de notre application nous a permis de nous confronter à une expérience professionnelle dans le domaine de développement des applications que nous ne pouvant qualifier que d’enrichissante. En outre, elle nous a offert de l’opportunité de concrétiser nos connaissances théoriques à l’aide d’un cas réel. Ainsi, ce projet nous a apporté le plus concernant les langages de programmations et les plateformes que nous avons découvertes et pratiqués.

Nous avons présenté dans ce rapport toutes les étapes nécessaires pour la conception et le développement d’une application web et d’une application mobile Android pour automatiser le traitement de dérangement. Ces applications sont développées spécialement pour l’utilisation de trois acteurs qui sont le client, le technicien (utilisateur) et l’admin.

Elle leur permet de poursuivre les nouvelles réclamations de dérangement et les traiter aux plus brefs délais afin de réaliser la satisfaction clientèle.

Ce projet nous a donné de plus l’occasion d’acquérir des nouvelles connaissances à propos de Pacestar UML, Eclipse, phpMyAdmin et de maitriser les langages de programmation JAVA, PHP, CSS etc., qui seront certes utile dans nos futures vies professionnelles.

À titre général, ce stage a constitué une expérience professionnelle exceptionnelle. La vie universitaire ne permet que d’avoir un aperçu de son futur métier. On découvre rapidement que l’aspect pratique dépasse souvent les connaissances acquises et qu’il constitue une formation rapide.



1. [https://www.linkedin.com/company/tunisie-t-l-co](https://www.linkedin.com/company/tunisie-t-l-com)m (1503/2016)
2. [http://pf-mh.uvt.rnu.tn/601/1/Etude\_du\_processus\_de\_commande-](http://pf-mh.uvt.rnu.tn/601/1/Etude_du_processus_de_commande-livraison_et_provisionning_de_l%E2%80%99xDSL_au_sein_de_Tunisie_Telecom.pdf) [livraison\_et\_provisionning\_de\_l%E2%80%99xDSL\_au\_sein\_de\_Tunisie\_Telecom.pdf](http://pf-mh.uvt.rnu.tn/601/1/Etude_du_processus_de_commande-livraison_et_provisionning_de_l%E2%80%99xDSL_au_sein_de_Tunisie_Telecom.pdf) (15/03/2016)
3. [http://www.memoireonline.com/11/11/4918/m\_La-sante-financiere-de-Tunisie-Telecom-](http://www.memoireonline.com/11/11/4918/m_La-sante-financiere-de-Tunisie-Telecom-4.html) [4.htm](http://www.memoireonline.com/11/11/4918/m_La-sante-financiere-de-Tunisie-Telecom-4.html)l (16/03/2016)
4. <http://www.commentcamarche.net/contents/473-cycle-de-vie-d-un-logiciel>(27/03/2016)
5. http://infocenter.sybase.com/help/index.jsp?topic=/com.sybase.infocenter.dc31018.1653/doc/ html/rad1232632566113.html (17/04/2016)
6. <http://lipn.univ-paris13.fr/~gerard/uml-s2/uml-cours05.html>(18/04/2016)
7. <https://fr.wikipedia.org/wiki/Java_Development_Kit>(20/05/2016)
8. [https://docops.ca.com/devtest-solutions/8-0-1/fr/installation-et-](https://docops.ca.com/devtest-solutions/8-0-1/fr/installation-et-migration/installation/configuration-de-l-environnement-de-test-d-application-mobile/utilisation-de-genymotion) [migration/installation/configuration-de-l-environnement-de-test-d-application-mobile/utilisation-](https://docops.ca.com/devtest-solutions/8-0-1/fr/installation-et-migration/installation/configuration-de-l-environnement-de-test-d-application-mobile/utilisation-de-genymotion) [de-genymotion](https://docops.ca.com/devtest-solutions/8-0-1/fr/installation-et-migration/installation/configuration-de-l-environnement-de-test-d-application-mobile/utilisation-de-genymotion) (20/05/2016)
9. [http://tristan-jahier.fr/blog/2013/02/presentation-de-sublime-text-un-editeur-qui-a-la-](http://tristan-jahier.fr/blog/2013/02/presentation-de-sublime-text-un-editeur-qui-a-la-megaclasse) [megaclass](http://tristan-jahier.fr/blog/2013/02/presentation-de-sublime-text-un-editeur-qui-a-la-megaclasse)e (05/05/2016)
10. <http://fr.slideshare.net/nextma/gestion-programme-moussanada-avec-openerp>
11. [https://fr.wikibooks.org/wiki/Le\_langage\_CS](https://fr.wikibooks.org/wiki/Le_langage_CSS)S (05/05/2016)