

**Ministère de l'Enseignement Supérieur Et de
la Recherche Scientifique
Institut Supérieur des Etudes Technologiques
de Rades**



Banque de Tunisie

« Conception et réalisation d'un système de « Monitoring » de serveurs »

Réalisé au sein de

Réalisé par :
HTIRA Hatem
MONCER Aymen

Encadré par :
Mme. DHIAB Jihene (ISET)
M.SERGHINI Med Fethi (BT)

Année Universitaire : 2015/2016

Remerciements

Nous tenons à remercier dans un premier temps, toute l'équipe pédagogique de « **ISSET** » et les intervenants professionnels responsables de la formation, pour avoir assuré la partie théorique de celle-ci.

Nous remercions également Monsieur **Serghini Med Fethi** pour l'aide et les conseils concernant les missions évoquées dans ce rapport et l'avancement du projet, qu'il nous a apporté lors des différents suivis.

Nous remercions aussi Madame **Dhiab Jihene** pour l'aide et les conseils concernant les missions évoquées dans ce rapport, qu'elle nous a apporté lors des différents suivis.

Nous tenons à remercier tout particulièrement et à témoigner toute notre reconnaissance aux personnes qui nous ont aidés au cours de ce stage, pour l'expérience enrichissante et pleine d'intérêt qu'elles nous ont fait vivre durant ces mois au sein de l'entreprise **Banque de Tunisie**.

Résumé

L'objectif principal de ce travail de PFE consiste à mettre en place un outil de supervision de serveurs. Une solution web basée sur le Framework .NET et une solution mobile basée sur le Framework Cordova

Cette Application permettra aux les administrateurs à intervenir en mode proactive afin de minimiser l'indisponibilité des serveurs et historiser tous les taches effectués par les superviseurs

Mots clés: Cordova, WCF, Supervision, Authentification à deux facteurs, multi- plateformes, QRcode

Abstract

This final of studies project aims to implement a server monitoring tool. A web solution based on the .NET Framework and a mobile solution based on the Framework Cordova
This Application will allow administrators to intervene in proactive mode to minimize server downtime and historicize all tasks carried out by supervisors

Keywords: Cordova, WCF, Monitoring, Two Factor Authentication, Cross platform, QRcode

Sommaire

Introduction générale	9
Chapitre I : Cadre général Du projet	11
I. Introduction.....	11
II. Présentation de l'organisme d'accueil.....	11
1. Historique de la Banque de Tunisie	11
2. Organigramme de l'organisme.....	12
III. Présentation du projet.....	12
Chapitre II : Etude de l'existant.....	13
I. Introduction	13
II. Etude de l'existant.....	13
1. Critique de l'existant.....	13
2. Solution proposée.....	14
Chapitre III: Etude de l'art.....	15
I. Introduction :.....	15
II. Solutions disponibles sur le marché :.....	15
1. PTRG	15
2. ManageEngine	16
III. Conclusion	16
Chapitre IV : Etude théorique et choix techniques.....	17
I. Introduction	17
II. Notions théoriques.....	17
1. Authentification à deux facteurs :	17
2. WCF Service.....	18
3. Windows Service	19
4. Apache Cordova.....	19
Chapitre V : Analyse et Spécification des besoins	21
I. Introduction	21
II. Spécification des besoins fonctionnels et non fonctionnels	21
1. Définition des acteurs.....	21
2. Spécification des besoins fonctionnels.....	21
3. Spécification des besoins non fonctionnels.....	22
III. Le diagramme des cas d'utilisation.....	23
1. Diagramme de cas d'utilisation global.....	23
2. Affectation des priorités aux cas d'utilisations.....	24

3. Définition des cas d'utilisations	24
4. Raffinement des cas d'utilisations	25
V. Conclusion.....	36
Chapitre VI. Conception	37
I. Introduction	37
II. Conception détaillée.....	37
1. Diagramme de séquence	37
a. Le cas d'utilisation « s'inscrire » :	38
b. Le cas d'utilisation « Récupérer OTP» :	39
c. Le cas d'utilisation « se connecter » :	40
d. Le cas d'utilisation « Superviser Serveur Mail» :	41
e. Le cas d'utilisation « Superviser Serveur Web» :	43
f. Le cas d'utilisation « Superviser Serveur Base de Données» :	44
g. Le cas d'utilisation « Noyau Monitoring» :	45
h. Le cas d'utilisation « Accorder Privilège» :	45
2. Le diagramme de classe	46
3. La base de données de l'application.....	48
III. Conclusion	49
VII. Réalisation.....	50
I. Introduction.....	50
II. Environnement de travail.....	50
1. Environnement matériel et logiciel	50
III. Réalisation partie serveur.....	53
IV. Interfaces de l'application	54
1. Application Mobile	55
2. Application Web	61
V. Conclusion	70
Conclusion générale.....	71
Nétographie	72
Bibliographies.....	73
ANNEXE	74

Liste des figures

Figure 1:Logo de la Banque de Tunisie	11
Figure 2 : Organigramme de la Banque de Tunisie	12
Figure 3:Critique de l'existant	13
Figure 4 : Image simplifiant l'idée de notre projet	14
Figure 5 : PTRG	15
Figure 6 :ManageEngine OpManager	16
Figure 7 : Image simplifiant l'authentification à deux facteurs	17
Figure 8: Architecture WCF	18
Figure 9: Fonctionnement de Cordova	20
Figure 10 : Diagramme de cas d'utilisation global	23
Figure 11: Raffinement du cas d'utilisation « Gérer Serveurs »	25
Figure 12: Raffinement du cas d'utilisation « Gérer Services »	27
Figure 13:Raffinement du cas d'utilisation « S'inscrire »	29
Figure 14: Raffinement du cas d'utilisation « Gérer Superviseurs »	29
Figure 15: Raffinement du cas d'utilisation « Se connecter »	31
Figure 16: Raffinement du cas d'utilisation « Consulter Alertes »	32
Figure 17: Raffinement du cas d'utilisation « Récupérer OTP »	32
Figure 18: Raffinement du cas d'utilisation « Gérer Supervision »	33
Figure 19: Raffinement du cas d'utilisation « Consulter Intervention »	35
Figure 20 : Diagramme de séquence pour le cas d'utilisation "s'inscrire"	38
Figure 21 : Diagramme de séquence pour le cas d'utilisation "Récupérer OTP"	39
Figure 22 : Diagramme de séquence pour le cas d'utilisation "se connecter"	40
Figure 23 : Diagramme de séquence de cas d'utilisation "Superviser Serveur Mail"	42
Figure 24 : Diagramme de séquence de cas d'utilisation "Superviser Serveur Web"	43
Figure 25 : Diagramme de séquence de cas d'utilisation "Superviser Serveur BD"	44
Figure 26 : Diagramme de séquence de cas d'utilisation "Noyau Monitoring"	45
Figure 27 : Diagramme de séquence de cas d'utilisation "Accorder Privilège"	46
Figure 28: Diagramme de classe de l'application.....	48
Figure 29 : Google Authenticator.....	52
Figure 30 : Package de web service de l'application.....	53
Figure 31 : Setup Windows Monitoring	53
Figure 32 : Liste des services	54
Figure 33 : Page d'accueil.....	55
Figure 34 : Dashboard Admin Figure 35 : Dashboard Admin2	56
Figure 36 : Dashboard Serveur.....	57
Figure 37 : Menu Admin Figure 38 : Menu Superviseur	58
Figure 39 : Liste des serveurs	59
Figure 40 : List des services.....	59
Figure 41 : Liste des superviseurs	60
Figure 42 : Historique.....	60
Figure 43 : Page de connexion	61

Figure 44 : Page d'inscription.....	62
Figure 45 : Inscription réussi.....	62
Figure 46:Menu Superviseur Figure 47:Menu Admin	63
Figure 48 : Dashboard Admin	64
Figure 49 : Dashboard Superviseur	64
Figure 50 : Liste des serveurs.....	65
Figure 51 : Historique.....	65
Figure 52 : Table des superviseurs.....	66
Figure 53 : Table Monitoring.....	66
Figure 54 : Table de gestion de privilège	67
Figure 55 : Calendrier.....	67
Figure 56 : Page intervention.....	68
Figure 57 : Control des Superviseurs	68
Figure 58 : Ajouter Serveur	69
Figure 59 : Ajouter Service	69
Figure 60 : Intervenir.....	69
Figure 61 : Liste des services déployés sur le serveur IIS.....	74
Figure 62 : Ajout de l'URL de Service	75
Figure 63 : Liste des services.....	75
Figure 64 : WCF Test Client	76
Figure 65 : Fonction Noyau 1	76
Figure 66 : Fonction Noyau 2	77

Liste des tableaux :

Tableau 1: Affectation des priorités des cas d'utilisation	24
Tableau 2: Raffinement du cas d'utilisation « Ajouter Serveur »	25
Tableau 3: Raffinement du cas d'utilisation « Supprimer Serveur »	26
Tableau 4: Raffinement du cas d'utilisation « Modifier Serveur »	26
Tableau 5: Raffinement du cas d'utilisation « Ajouter Service »	27
Tableau 6: Raffinement du cas d'utilisation « Supprimer Service »	28
Tableau 7: Raffinement du cas d'utilisation « Consulter Liste des Services »	28
Tableau 8: Raffinement du cas d'utilisation « S'inscrire »	29
Tableau 9: Raffinement du cas d'utilisation « Supprimer Superviseur »	30
Tableau 10: Raffinement du cas d'utilisation « Valider Superviseur »	30
Tableau 11: Raffinement du cas d'utilisation « Consulter Liste des Utilisateurs »	31
Tableau 12: Raffinement du cas d'utilisation « Se connecter »	31
Tableau 13: Raffinement du cas d'utilisation « Consulter Alertes »	32
Tableau 14: Raffinement du cas d'utilisation « Récupérer OTP »	33
Tableau 15: Raffinement du cas d'utilisation « Consulter Supervision »	33
Tableau 16: Raffinement du cas d'utilisation « Modifier Supervision »	34
Tableau 17: Raffinement du cas d'utilisation « Supprimer Supervision »	34
Tableau 18: Raffinement du cas d'utilisation « Ajouter Supervision »	35
Tableau 19: Raffinement du cas d'utilisation « Consulter Intervention »	35

Introduction générale

La supervision réseau et le monitoring des services est une nécessité pour chaque entreprise. Cela permet non seulement d'assurer le meilleur fonctionnement possible du système d'information mais également de réduire les coûts en minimisant les interruptions de service.

Il est important de surveiller l'état des serveurs et d'être prévenu si une anomalie survient. Le monitoring de serveurs est une technique de surveillance externe des serveurs. Cela permettra d'être alerté à la moindre indisponibilité d'un des serveurs. Le monitoring assure la surveillance de serveurs 24 heures sur 24 et 7 jours sur 7.

Dans le cadre de notre projet de fin d'étude, au sein de la Banque de Tunisie, il nous a été confié la mission de réaliser une application de « Monitoring des serveurs » qui sera conçu essentiellement pour aider les superviseurs dans le control des serveurs.

Ce rapport présente l'ensemble des étapes suivies pour développer notre application. Il est subdivisé en six chapitres organisés comme suit :

Le premier chapitre intitulé « **Cadre générale du projet** » est consacré à la présentation du contexte du travail ainsi que l'organisme d'accueil.

Dans le deuxième chapitre « **Etude de l'existant** », nous allons essayer de présenter les différents problèmes existants, ensuite, nous allons proposer une solution possible, afin de faciliter, accélérer et bien contrôler la supervision au sein de la Banque de Tunisie.

Le troisième chapitre « **Etat de l'art** » s'articule autour de l'étude de quelques de monitoring d'une manière détaillée, en exposant leurs fonctionnalités. Nous nous sommes basés sur cette étude pour faire la spécification de notre application, qui fera l'objet du quatrième chapitre.

Le chapitre suivant intitulé « **Etude théorique et choix techniques** » contient une étude préliminaire sur les Framework .Net et les Framework dédiés pour le développement mobile multi-plate formes, et par la suite le choix de la solution technique à adopter. Et enfin, une vue globale sur l'architecture du projet.

Dans le cinquième chapitre « **Analyse des besoins et spécification** », nous déterminons les besoins fonctionnels et non fonctionnels de notre application et nous présentons les différents cas d'utilisation.

Le sixième chapitre intitulé « **Conception** » détaille les différents aspects conceptuels de l'application.

Le dernier chapitre intitulé « **Réalisation** » présente l'environnement de travail ainsi que les outils logiciels que nous avons utilisés pour la réalisation de notre projet. Il illustre aussi le travail réalisé avec un ensemble d'interfaces graphiques conçues pour l'application.

Une conclusion clôturera ce rapport dans laquelle nous mentionnons les différents atouts de ce projet et les perspectives d'améliorations possibles à notre application.

Chapitre I : Cadre général Du projet

I. Introduction

Dans ce chapitre, nous exposons le contexte général du projet. On présente en premier lieu, l'organisme d'accueil, le travail demandé et son intérêt.

II. Présentation de l'organisme d'accueil

1. Historique de la Banque de Tunisie



Figure 1: Logo de la Banque de Tunisie

La Banque de Tunisie ou BT est une banque commerciale privée de Tunisie. Fondée le 23 septembre 1884, elle est l'une des plus anciennes banques d'Afrique. En juin 2012, elle compte 103 agences et 165 distributeurs automatiques de billets répartis sur le territoire tunisien.

En 1948, la Banque de Tunisie absorbe la Banque italo-française de crédit et participe en 1951 à la liquidation de la Banca Italiano de Credito dont elle reprend, après l'indépendance, la plupart des agences. Première banque à être installée en Tunisie, elle reste longtemps le seul établissement de crédit bancaire et, pendant longtemps, la seule banque à disposer d'agences dans le pays. En 1963, la Banque de Tunisie ouvre son capital à la Société générale, le Crédit suisse et la Banca Nazionale del Lavoro. En 1968, la Banque de Tunisie rachète les agences tunisiennes de la Compagnie française de crédit et de banque

2. Organigramme de l'organisme

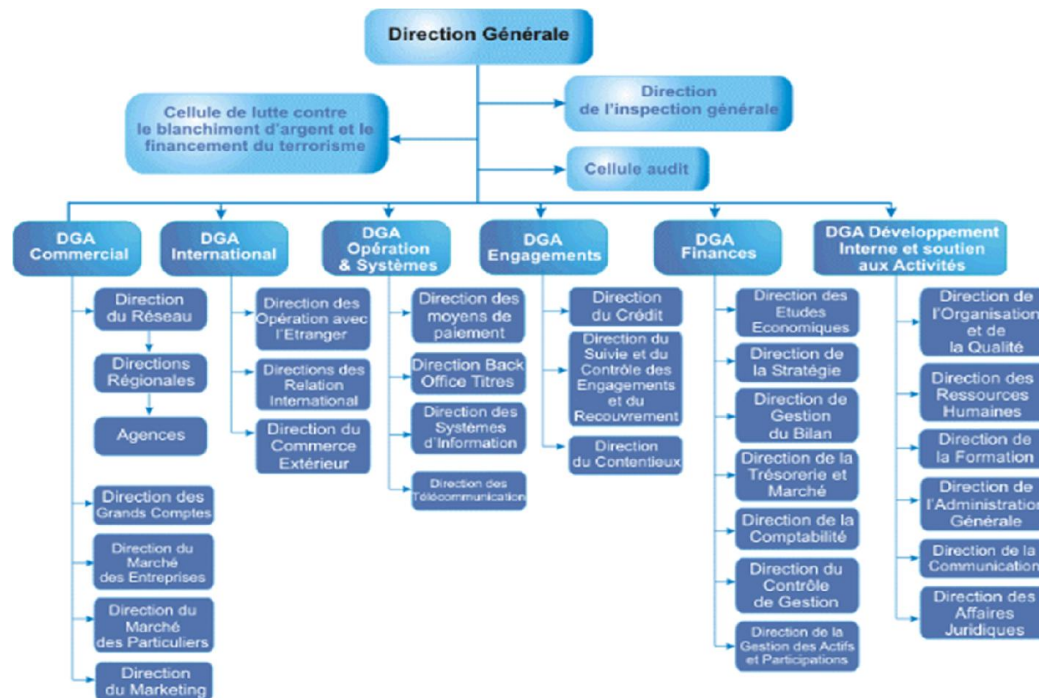


Figure 2 : Organigramme de la Banque de Tunisie

III. Présentation du projet

Dans le cadre de notre projet de fin d'étude, il nous a été demandé de concevoir et développer une application de supervision permettant à aider les administrateurs à intervenir en mode proactive afin de minimiser l'indisponibilité des serveurs et historiser tous les tâches effectués par les superviseurs. En effet, l'enregistrement de l'historique des événements de l'activité sur le système informatique et leur analyse sont des points importants dans le système d'information. Cela permet la construction d'indicateurs pour avoir une vue d'ensemble sur différentes informations (l'état de serveurs, le fonctionnement de services..) et la détection d'anomalies ou d'erreurs par notifications (Envoi SMS, Email ...) afin d'agir plus rapidement en cas de besoin.

Ce projet consiste à développer une application permettant d'analyser les différents comportements des serveurs de la Banque de Tunisie.

Ce travail nécessite comme pré requis de :

- Avoir une connaissance sur l'architecture du réseau étudié.
- Etudier les différents serveurs et services existants.
- Etudier les interactions entre ces serveurs.

A titre d'exemple : dans un même réseau on peut trouver trois serveurs (Web, Messagerie et Base de données) donc la panne d'un seul serveur ou l'un des services qu'il offre peut affecter le reste des serveurs.

Chapitre II : Etude de l'existant

I. Introduction

Dans ce chapitre, nous allons essayer de présenter les différents problèmes existants, ensuite, nous allons proposer une solution possible, afin de faciliter, accélérer et bien contrôler la supervision au sein de la Banque de Tunisie.

II. Etude de l'existant

La Banque de Tunisie est une société qui comporte un nombre important de directions et de services contrôlés par plusieurs serveurs.

Le département informatique de la Banque de Tunisie a comme but de gérer les serveurs tournant dans un réseau LAN qui leur permet de communiquer de manière rapide et sûre. Ces serveurs (Web, Messagerie, Data ...) offrent divers services (le courrier électronique, la sauvegarde de données ...).

On doit noter qu'au sein de la direction informatique, il y'a un centre de supervision qui assure le suivi des différents serveurs, et, qui contrôle et effectue le suivi de leurs états et leurs fonctionnements.

1. Critique de l'existant

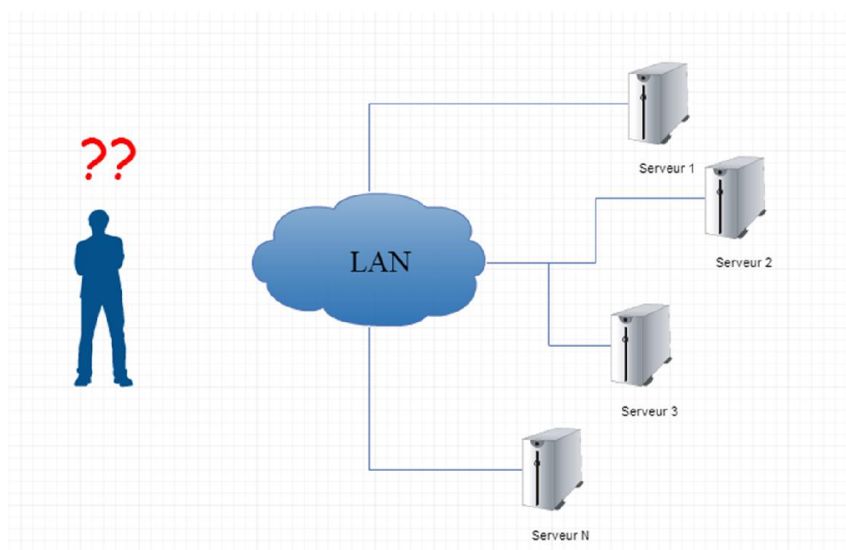


Figure 3: Critique de l'existant

Au sein du centre de supervision, l'administrateur assure une vérification manuelle quotidiennement. Cette vérification nécessite beaucoup de temps et est considérée insuffisante pour des systèmes très critiques.

Ce projet est né dans l'objectif de faciliter la tâche des administrateurs système et d'assurer la supervision des serveurs en temps réel.

2. Solution proposée

Notre application sera subdivisée en deux parties :

- Une application Web qui inclura toutes les fonctionnalités nécessaires pour créer un outil de supervision de serveurs qui permettra de suivre l'état et le fonctionnement de services offerts.
- Une application Mobile qui se limitera sur la consultation des états de différents serveurs et de leurs services.

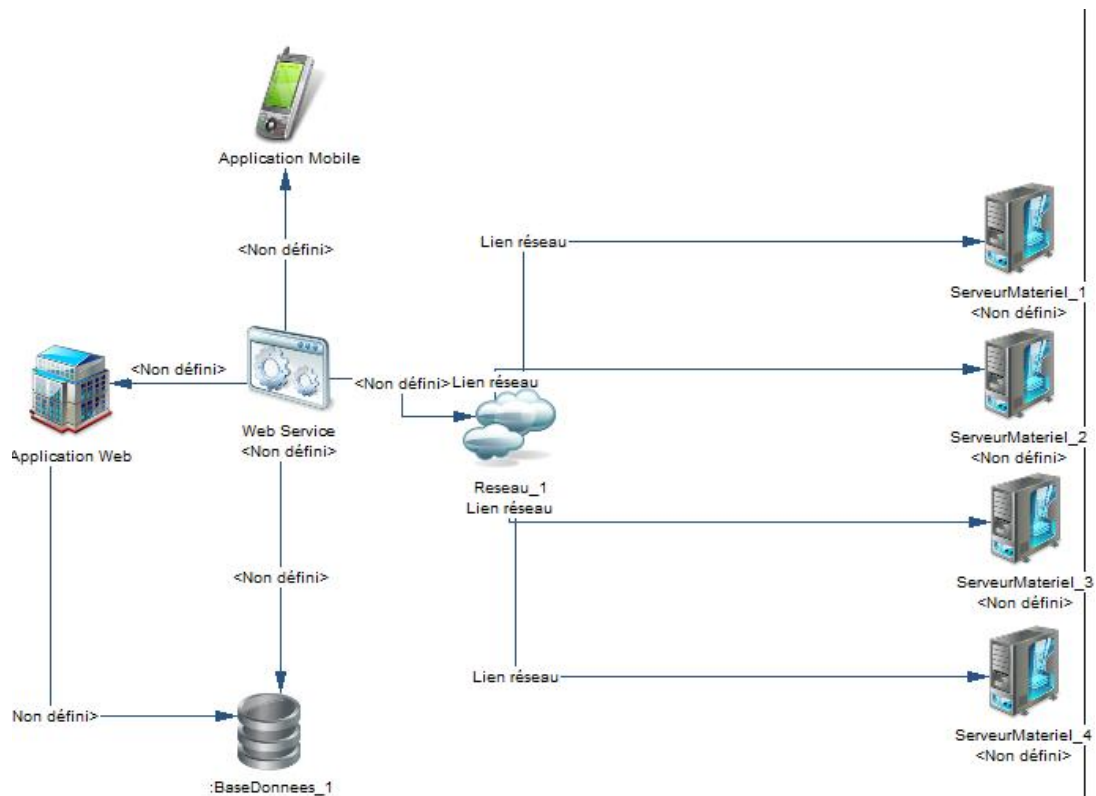


Figure 4 : Image simplifiant l'idée de notre projet

L'évolution du nombre de serveurs chez la Banque de Tunisie provoque une difficulté de plus en plus grande de supervision rapide et efficace. D'où vient l'idée de mettre en place un outil de supervision de serveurs qui permettra de suivre l'état et le fonctionnement de services offerts. Cet outil de supervision devra être adapté au domaine d'application, il permettra aux administrateurs de recevoir des notifications en cas de dysfonctionnement d'un serveur.

Chapitre III: Etude de l'art

I. Introduction :

Ce chapitre a pour objet de présenter quelques applications de monitoring des serveurs.

II. Solutions disponibles sur le marché :

1. PTRG

PRTG est un logiciel qui permet grâce à l'analyse de trames SNMP de créer des graphiques sur le trafic réseau

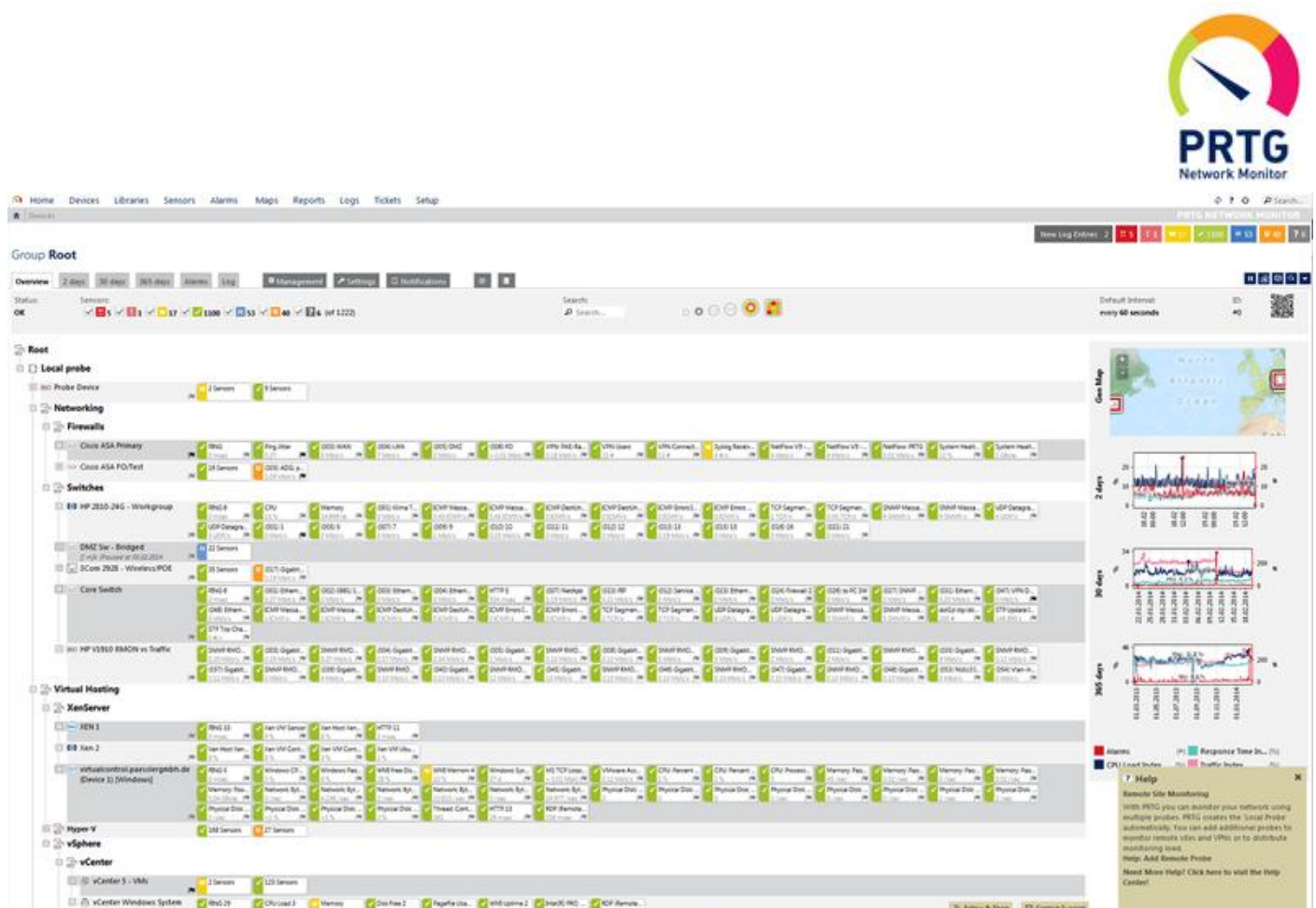


Figure 5 : PRTG

Avantages de PRTG

- ✓ Téléchargement, installation et configuration rapides
- ✓ Facile à utiliser
- ✓ Surveillance réseau complète
- ✓ Des alertes flexibles
- ✓ Interface en plusieurs langues
- ✓ Rapports détaillés
- ✓ Performance et sécurité

2. ManageEngine

ManageEngine OpManager est une plate - forme de gestion de réseau qui aide les grandes entreprises, fournisseurs de services, et les PME à gérer leurs centres de données et l'infrastructure informatique efficace et rentable. Il offre la surveillance du réseau, physique et la surveillance des serveurs virtuels, la gestion des journaux d'événements, services Windows et TCP services de surveillance et de suivi.



Figure 6 :ManageEngine OpManager

Avantages de ManageEngine OpManager

- ✓ Navigation intuitive
- ✓ Suivre les ressources allouées
- ✓ Surveiller les indicateurs de performance
- ✓ Recevoir des alertes via email/SMS
- ✓ Obtenir un rapport détaillé sur l'environnement virtuel

III. Conclusion

Dans ce chapitre nous avons essayé d'étudier quelques applications de control de serveurs dans le but d'avoir une idée sur les fonctionnalités de ces dernières et de ressortir leurs points forts. Sur la base de cette étude, nous allons élaborer la spécification de notre application. Une étude théorique sur les plates formes open source dédiées pour des applications Smartphones fonctionnant sur plusieurs Os mobiles fera l'objet du chapitre suivant.

Chapitre IV : Etude théorique et choix techniques

I. Introduction

Dans ce chapitre, nous allons exposer une étude de certaines notions théoriques qui vont nous rencontrer au cours de développement de ce projet.

II. Notions théoriques :

1. Authentification à deux facteurs :



Figure 7 : Image simplifiant l'authentification à deux facteurs

Un système identifie un utilisateur via un identifiant d'utilisateur, généralement abrégé à l'utilisateur ID. Le processus par lequel un système confirme que l'utilisateur est vraiment celui qui prétend être est appelé authentification. Il y'a trois types d'informations d'identification à travers laquelle un utilisateur peut être authentifié: facteur connaissances (ce que l'utilisateur sait), la propriété facteur (ce qu'un utilisateur possède), et l'inhérence facteur (ce qu'un utilisateur est). Lorsque vous avez un mécanisme d'authentification qui exploite une combinaison de deux de ces facteurs, il est appelé authentification à deux facteurs (TFA).

Prenons l'exemple d'une application bancaire. Lorsque vous essayez de faire quelque chose d'important comme l'ajout d'une nouvelle bénéficiaire pour le transfert de fonds, l'appel d'API correspondante peut exiger un facteur de propriété supplémentaire tout en acceptant un seul facteur pour les autres demandes typiques. L'ajout d'un bénéficiaire est toujours important, car un utilisateur malveillant avec vos informations d'identification peuvent s'ajouter au bénéficiaire et vider votre compte. Ainsi TFA font toujours sens pour ce type de demande. Dans d'autres cas, il faudra TFA sur une base nécessaire, même pour des actions sélectives en fonction de l'entrée ou un autre paramètre. Un exemple d'un tel cas est un transfert de fonds. Si vous transférez 50 \$, vous ne voudrez pas utiliser un jeton pour obtenir le code d'authentification pour effectuer le transfert. La sécurité à deux facteurs pour ce cas d'utilisation devient un irritant. Toutefois, si vous transférez 50.000 \$, vous avez sans doute voudrez la protection supplémentaire offre TFA.

Il va sans dire que, d'un point de vue sécurité TFA est plus robuste que l'authentification à un seul facteur.

C'est pour cela que nous avons opté pour l'authentification à deux facteurs dans ce projet.

2. WCF Service

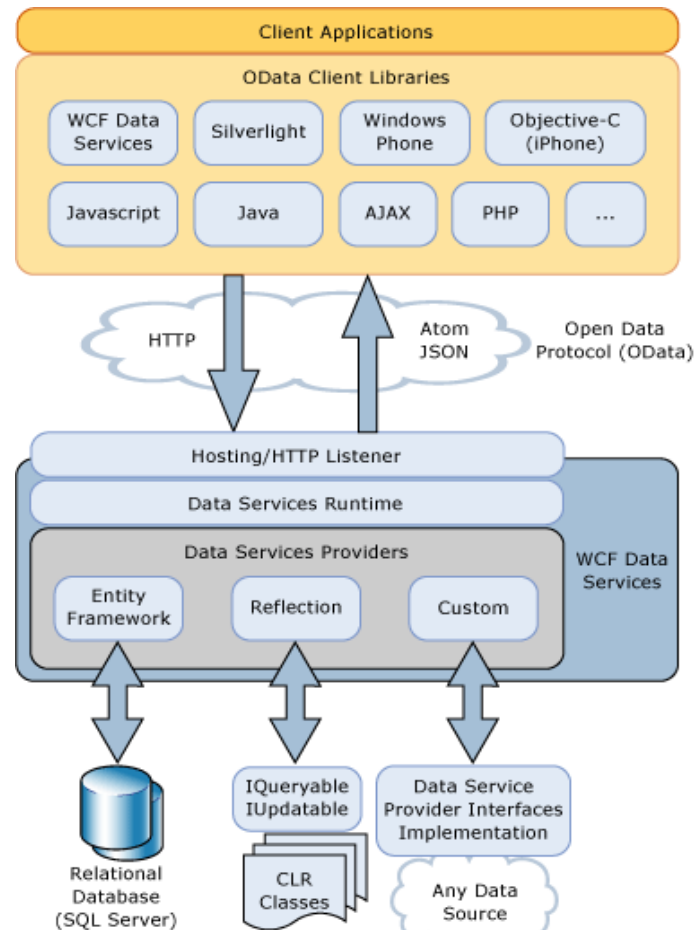


Figure 8: Architecture WCF

Windows Communication Foundation (WCF) est une infrastructure permettant de générer des applications orientées services. Avec WCF, vous pouvez envoyer des données sous forme de messages asynchrones d'un point de terminaison de service à un autre. Un point de terminaison de service peut faire partie d'un service disponible en continu et hébergé par IIS, ou il peut s'agir d'un service hébergé dans une application. Un point de terminaison peut être un client d'un service qui demande des données auprès d'un point de terminaison de service. Les messages peuvent être simplement constitués d'un caractère ou d'un mot unique envoyé au format XML, ou se présenter sous la forme d'un flux de données binaires plus complexe.

WCF comprend l'ensemble de fonctionnalités suivant :

- **Orientation services**

L'une des conséquences de l'utilisation de normes WS est que WCF vous permet de créer des applications *orientées services*. L'architecture orientée services (SOA, Service-Oriented Architecture) dépend de services Web pour envoyer et recevoir des données. Les services présentent l'avantage d'être faiblement couplés, au lieu d'être encodés de manière irréversible d'une application à une autre. Une relation faiblement couplée implique que tout client créé sur une plate-forme peut se connecter à n'importe quel service, dans la mesure où les contrats essentiels sont respectés.

- **Interopérabilité**

WCF implémente les normes industrielles modernes pour l'interopérabilité des services Web.

- **Sécurité**

Les messages peuvent être chiffrés afin de protéger la confidentialité. Par ailleurs, vous pouvez demander aux utilisateurs de s'authentifier avant de pouvoir recevoir des messages. La sécurité peut être implémentée à l'aide de normes célèbres, telles que SSL ou WS-Secure Conversation.

- **Prise en charge d'AJAX et de REST**

REST est un exemple d'une technologie évoluant vers le Web 2.0. WCF peut être configuré pour traiter des données XML « ordinaires » qui ne sont pas encapsulées dans une enveloppe SOAP. WCF peut être étendu pour prendre en charge des formats XML spécifiques, tels qu'ATOM (une norme populaire RSS), et même des formats non XML, tels que JSON (JavaScript Object Notation).

- **Extensibilité**

L'architecture WCF comporte un certain nombre de points d'extensibilité. Si une capacité supplémentaire est requise, il existe un certain nombre de points d'entrée qui vous permettent de personnaliser le comportement d'un service.

3. Windows Service



Un service est un type d'application qui fonctionne en arrière - plan du système. Les services fournissent les fonctionnalités principales du système d'exploitation, comme portion Web, la journalisation des événements, le service de fichiers, aide et soutien, l'impression, la cryptographie, et les rapports d'erreur. Avec les services snap-in, on peut gérer les services sur des ordinateurs locaux ou distants.

Vous pouvez utiliser le composant logiciel enfichable pour:

- Démarrer, arrêter, suspendre, reprendre ou désactiver les services sur des ordinateurs distants et locaux.
- Gérer les services sur les ordinateurs locaux et distants.
- Mettre en place des mesures de rétablissement aient lieu si un service ne parvient pas, par exemple, le redémarrage du service automatiquement ou redémarrer l'ordinateur
- Activer ou désactiver des services pour un profil matériel particulier.
- Afficher l'état et la description de chaque service

4. Apache Cordova

Apache Cordova est un Framework de développement mobile open-source. Il permet d'exploiter les technologies Web courantes telles que HTML5, CSS3 et JavaScript pour développer des applications multiplateformes, évitant ainsi l'utilisation des langages natifs propres aux différentes plates-formes mobiles. Les applications s'exécutent dans des wrappers ciblés pour chaque plate-forme, elles s'appuient sur des API conformes aux standards permettant l'accès aux capteurs de chaque appareil, aux données ainsi qu'à l'état du réseau.

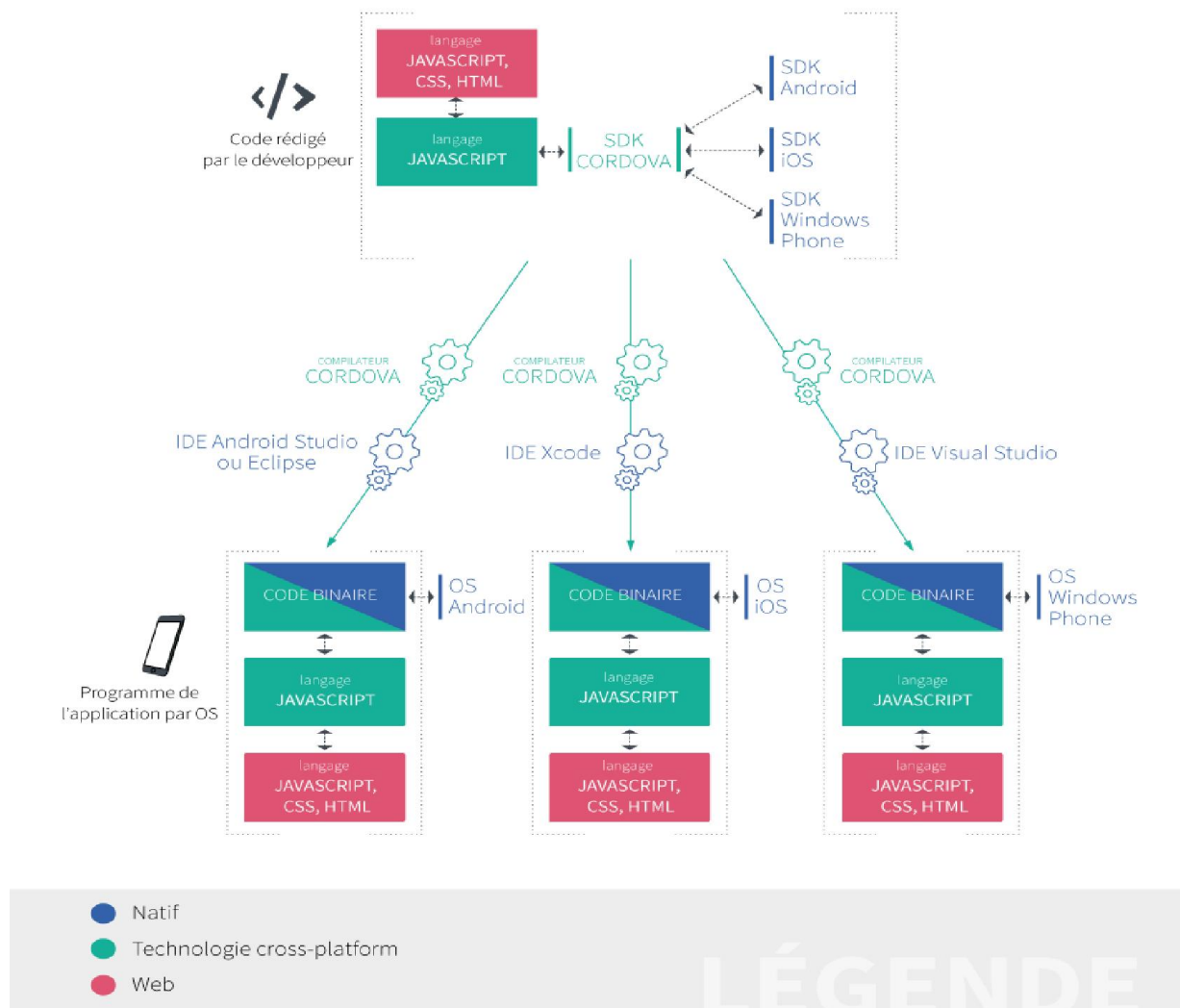


Figure 9: Fonctionnement de Cordova

Avantages

- Ecrire avec des technologies de développement web, au lieu de l'apprentissage de chaque Framework.
- "Write once run every where" (en fait ce n'est pas de cette façon que les choses se passent, la plupart du temps, c'est «Write once debug every where").

Donc, pas de phase d'apprentissage pour chaque plate-forme, et pas de temps de développement supplémentaire pour le portage à chaque plate-forme, seul un petit effort pour un déploiement sur des plateformes différentes.

Chapitre V : Analyse et Spécification des besoins

I. Introduction

Dans ce chapitre, nous allons analyser et présenter les besoins fonctionnels et non fonctionnels de notre application.

Afin de donner une vision globale du comportement fonctionnel du système, la spécification formelle des besoins est représentée à travers des diagrammes de cas d'utilisation.

II. Spécification des besoins fonctionnels et non fonctionnels

1. Définition des acteurs

Pour notre application, nous avons pu distinguer les acteurs suivants :

- L'administrateur : c'est un acteur chargé de contrôler, enregistrer et mettre à jour les serveurs à superviser, les alertes et les tests.
- Le superviseur : c'est un acteur chargé de consulter l'état et le fonctionnement d'un seul serveur.

2. Spécification des besoins fonctionnels

Un besoin fonctionnel est un besoin spécifiant une action qu'un système doit être capable d'effectuer, sans considérer aucune contrainte physique : besoins spécifiant un comportement d'entrée/sortie d'un système. [Jacobson, 1999]

Dans notre contexte le système doit permettre de :

- Gérer les Serveurs

Cette fonctionnalité permettra à l'administrateur d'ajouter un nouveau serveur à superviser, modifier les informations d'un serveur existant, supprimer un serveur existant et consulter la liste des serveurs.

- Gérer les Services

Cette fonctionnalité permettra à l'administrateur d'ajouter un nouveau service à superviser, modifier les informations d'un service existant, supprimer un service existant et consulter la liste des services existants par serveur.

- Consulter l'état des serveurs

Cette fonctionnalité permettra à l'administrateur, ainsi qu'au superviseur de consulter l'état d'un serveur ainsi que l'état des services d'un serveur spécifique.

- Gérer les Supervisions :

Cette fonctionnalité permettra à l'administrateur de créer, démarrer et arrêter des supervisions.

- Consulter les Alertes

Cette fonctionnalité permettra à l'administrateur de consulter les alertes générées.

- Gérer les Utilisateurs

Cette fonctionnalité permettra à l'administrateur d'ajouter un nouvel superviseur, modifier un superviseur existant et de supprimer un superviseur existant.

- Consulter Logs

Cette fonctionnalité permettra au superviseur de consulter les tests effectués.

- Consulter des statistiques

Cette fonctionnalité permettra à l'administrateur de consulter les Statistiques par serveur

3. Spécification des besoins non fonctionnels

Un besoin non fonctionnel est un besoin spécifiant des propriétés du système, telles que les contraintes liées à l'environnement et à l'implémentation, et les exigences en matière de performances, de dépendances de plate-forme, de facilité de maintenance, d'extensibilité et de fiabilité. [Jacobson, 1999]

Dans notre système nous avons pu distinguer les besoins non fonctionnels suivant :

- Le système doit fonctionner en réseau.
- Le système doit présenter des interfaces graphiques bien structurées du point de vue contenu informationnel.
- Le système doit être sécurisé avec l'obligation pour chaque utilisateur de saisir un login et un mot de passe et un One Time Password.

III. Le diagramme des cas d'utilisation

Les diagrammes de cas d'utilisation sont des diagrammes UML utilisés pour donner une vision globale du comportement fonctionnel d'un système logiciel

1. Diagramme de cas d'utilisation global

La figure suivante présente le cas d'utilisation global de notre application :

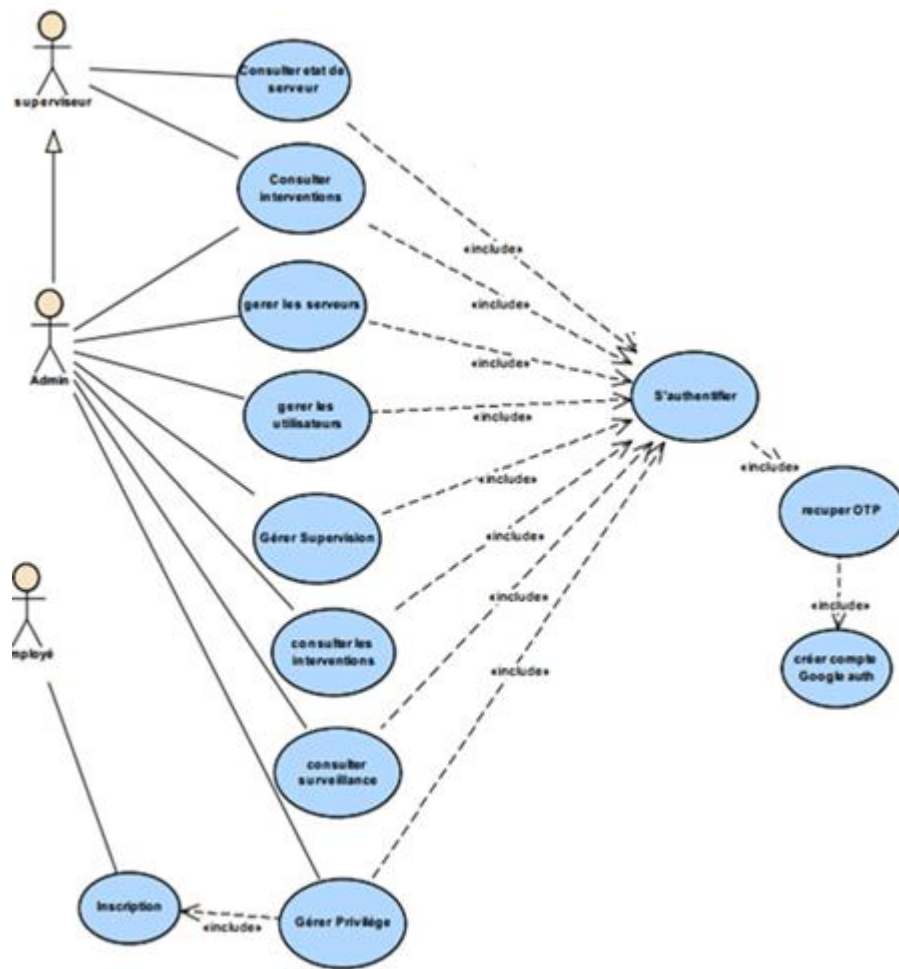


Figure 10 : Diagramme de cas d'utilisation global

2. Affectation des priorités aux cas d'utilisations

Num	Cas d'utilisation	Acteur	Priorités
1	Gérer Serveurs	Administrateur	1
2	Gérer Services	Administrateur	1
3	Gérer Supervision	Administrateur	2
4	Consulter Alertes	Administrateur	2
5	Gérer Privilèges	Administrateur	1
6	Gérer Utilisateurs	Administrateur	1
7	Consulter intervention	Administrateur	2
8	Consulter l'état de Serveurs	Administrateur	2
9	S'authentifier	Administrateur	1
10	Consulter Surveillance	Administrateur	1

Tableau 1: Affectation des priorités des cas d'utilisation

3. Définition des cas d'utilisations

Un cas d'utilisation est une description d'un ensemble de séquences d'actions, avec leurs variantes, effectuées par un système et donnant un résultat satisfaisant pour un acteur particulier. [Jacobson, 1999]

Nous allons dans ce qui suit présenter les différents cas d'utilisations.

Les cas d'utilisations relatifs à l'administrateur :

- La gestion de serveurs (consultation, ajout, modification, suppression)
- La gestion de services (consultation, ajout, modification, suppression)
- La gestion de Supervision (Démarrage, Arrêt)
- La gestion des utilisateurs (consultation, ajout, modification, suppression)
- La consultation des alertes (le suivi des alertes générées).
- L'authentification.
- Le control de toutes les tâches effectuées par les superviseurs

Les cas d'utilisations relatifs au superviseur :

- La consultation des logs (le suivi de supervision effectués).
- La vérification de l'état des serveurs.
- L'authentification.
- L'intervention sur les serveurs en cas de dysfonctionnement

4. Raffinement des cas d'utilisations

❖ Raffinement du cas d'utilisation « Gérer Serveurs » :

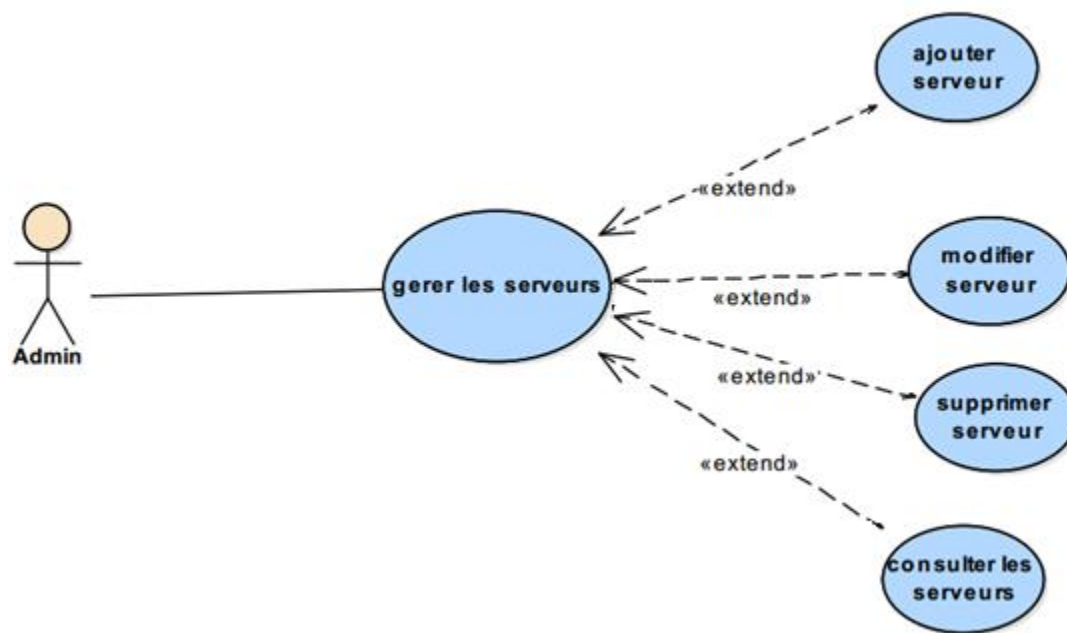


Figure 11: Raffinement du cas d'utilisation « Gérer Serveurs »

- Raffinement du cas d'utilisation « Ajouter Serveur » :

Cas d'utilisation	Ajouter Serveur
Acteur	Administrateur
Pré-Condition	S'authentifier
Post-Condition	Serveur Ajouté
Scénario Principale	L'administrateur s'authentifie Il clique bouton « Gérer Serveur » Le système affiche la liste des Serveurs existants. L'administrateur clique bouton « Ajouter Serveur » Le système affiche l'interface « Ajouter Serveur » Il saisit les données de Serveur L'administrateur clique bouton « Enregistrer »
Exception	En cas de données erronées, afficher un message d'erreur.

Tableau 2: Raffinement du cas d'utilisation « Ajouter Serveur »

- Raffinement du cas d'utilisation « Supprimer Serveur » :

Cas d'utilisation	Supprimer Serveur
Acteur	Administrateur
Pré-Condition	S'authentifier
Post-Condition	Serveur Supprimé
Scénario Principale	<ul style="list-style-type: none"> - L'administrateur s'authentifie - Il clique sur le bouton « Gérer Serveur » - Le système affiche la liste des Serveurs existants. - L'administrateur sélectionne le Serveur à supprimer. - Il clique bouton « Supprimer Serveur »
Exception	En cas d'erreur, afficher un message.

Tableau 3: Raffinement du cas d'utilisation « Supprimer Serveur »

- Raffinement du cas d'utilisation « Modifier Serveur » :

Cas d'utilisation	Modifier Serveur
Acteur	Administrateur
Pré-Condition	S'authentifier
Post-Condition	Serveur Modifié
Scénario Principale	<ul style="list-style-type: none"> - L'administrateur s'authentifie - Il clique sur le bouton « Gérer Serveur » - Le système affiche la liste des Serveurs existants. - L'administrateur sélectionne le Serveur à modifier. - Il clique bouton « Modifier Serveur » - Le système affiche l'interface « Modifier Serveur » - Il modifie les données de Serveur. - L'administrateur clique bouton « Enregistrer »

Tableau 4: Raffinement du cas d'utilisation « Modifier Serveur »

❖ **Raffinement du cas d'utilisation « Gérer Services » :**

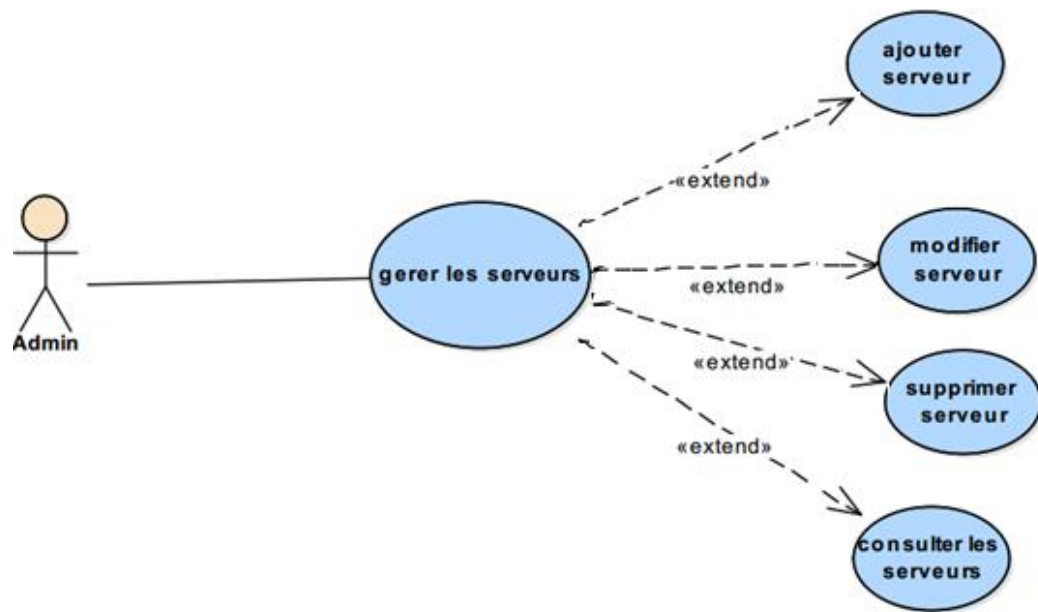


Figure 12: Raffinement du cas d'utilisation « Gérer Services »

- Raffinement du cas d'utilisation « Ajouter Service » :

Cas d'utilisation	Ajouter Service
Acteur	Administrateur
Pré-Condition	S'authentifier
Post-Condition	Service Ajouté
Scénario Principale	<ul style="list-style-type: none"> - L'administrateur s'authentifie - Il clique sur « Gérer Services » - Le système affiche la liste des Services existants. - L'administrateur clique bouton « Ajouter Service » - Le système affiche l'interface « Ajouter Service » - Il saisit les données de Service - L'administrateur clique bouton « Enregistrer »
Exception	En cas de données erronées, afficher un message d'erreurs.

Tableau 5: Raffinement du cas d'utilisation « Ajouter Service »

- Raffinement du cas d'utilisation « Supprimer Service » :

Cas d'utilisation	Supprimer Service
Acteur	Administrateur
Pré-Condition	S'authentifier
Post-Condition	Service Supprimé
Scénario Principale	<ul style="list-style-type: none"> - L'administrateur s'authentifie - Il clique sur le bouton « Gérer Service» - Le système affiche la liste des Services existants. - L'administrateur sélectionne le Service à supprimer. - Il clique bouton « Supprimer Service»
Exception	En cas d'erreur, afficher un message.

Tableau 6: Raffinement du cas d'utilisation « Supprimer Service »

- Raffinement du cas d'utilisation « Consulter Liste des Services »:

Cas d'utilisation	Consulter Liste de Services
Acteur	Administrateur
Pré-Condition	S'authentifier
Post-Condition	Liste de Services affiché
Scénario Principale	<ul style="list-style-type: none"> - L'administrateur s'authentifie - Il clique sur « Gérer Services» - Le système affiche la liste de Services existants.
Exception	En cas d'erreur, afficher un message.

Tableau 7: Raffinement du cas d'utilisation « Consulter Liste des Services »

❖ **Raffinement du cas d'utilisation « S'inscrire »:**

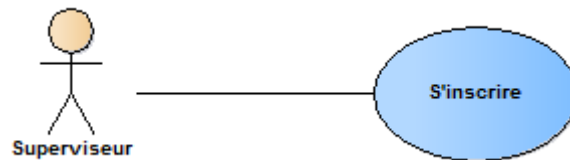


Figure 13: Raffinement du cas d'utilisation « S'inscrire »

Cas d'utilisation	S'inscrire
Acteur	Administrateur
Pré-Condition	-
Post-Condition	Inscription réussite en attente de validation de la part de l'administrateur.
Scénario Principale	<ul style="list-style-type: none"> - Le superviseur accède à la page d'inscription - Il remplit le formulaire d'inscription avec les données adéquates (informations personnelles + paramètres d'accès) - Le système le QR Code et le PSK du
Exception	En cas de données erronées, afficher un message d'erreurs.

Tableau 8: Raffinement du cas d'utilisation « S'inscrire »

❖ **Raffinement du cas d'utilisation « Gérer Superviseurs » :**

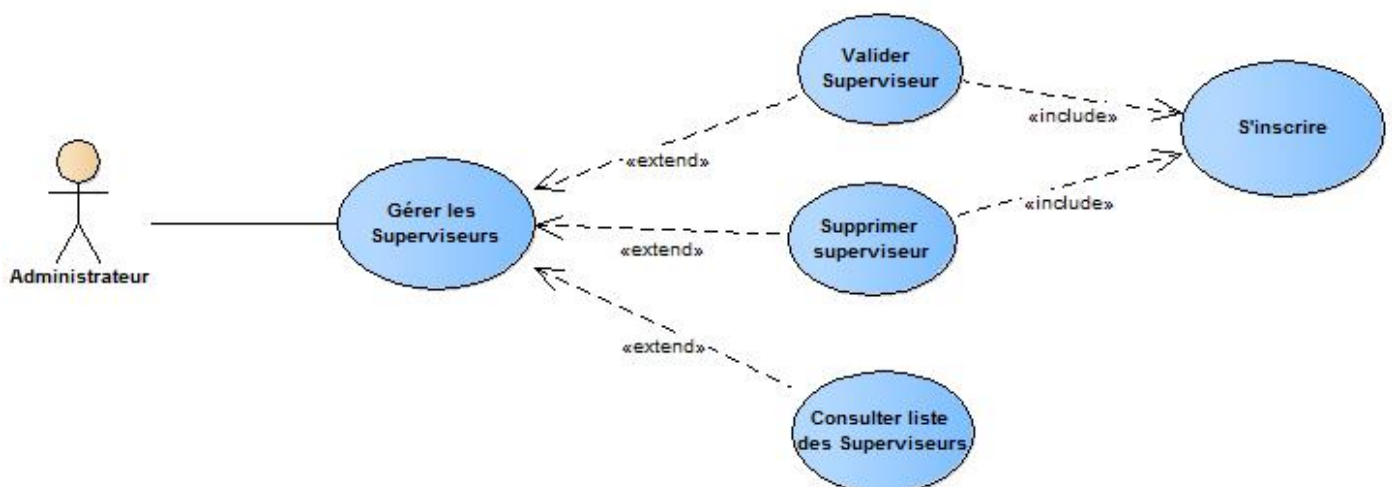


Figure 14: Raffinement du cas d'utilisation « Gérer Superviseurs »

- Raffinement du cas d'utilisation « Supprimer Superviseur » :

Cas d'utilisation	Supprimer Superviseur
Acteur	Administrateur
Pré-Condition	S'authentifier
Post-Condition	Superviseur Supprimé
Scénario Principale	<ul style="list-style-type: none"> - L'administrateur s'authentifie - Il clique sur le bouton « Gérer Superviseurs» - Le système affiche la liste des superviseurs existants. - L'administrateur sélectionne le superviseur à supprimer. - Il clique sur le bouton « Supprimer Superviseur»

Tableau 9: Raffinement du cas d'utilisation « Supprimer Superviseur »

- Raffinement du cas d'utilisation « Valider Superviseur » :

Cas d'utilisation	Valider Superviseur
Acteur	Administrateur
Pré-Condition	S'authentifier
Post-Condition	Superviseur Validé
Scénario Principale	<ul style="list-style-type: none"> - L'administrateur s'authentifie - Il clique sur « Gestion Superviseur» - L'administrateur sélectionne le superviseur à modifier. - Il clique bouton « Valider Superviseur» - Il déverrouille l'accès afin que le superviseur puisse accéder à l'application. - L'administrateur clique sur le bouton « Enregistrer »

Tableau 10: Raffinement du cas d'utilisation « Valider Superviseur »

- Raffinement du cas d'utilisation « Consulter Liste des Superviseurs » :

Cas d'utilisation	Consulter Liste des Superviseurs
Acteur	Administrateur
Pré-Condition	S'authentifier
Post-Condition	Liste des Superviseurs affiché
Scénario Principale	<ul style="list-style-type: none"> - L'administrateur s'authentifie. - Il clique bouton « Gérer Superviseurs». - Le système affiche la liste des Superviseurs existants.

Tableau 11: Raffinement du cas d'utilisation « Consulter Liste des Utilisateurs »

❖ **Raffinement du cas d'utilisation « Se connecter » :**



Figure 15: Raffinement du cas d'utilisation « Se connecter »

Cas d'utilisation	Se connecter
Acteur	Administrateur, Superviseur
Pré-Condition	Récupérer OTP
Post-Condition	Utilisateur connecté
Scénario Principale	<ul style="list-style-type: none"> - L'utilisateur saisit ses données d'authentification (login, mot de passe, OTP) - Il clique bouton « Se Connecter »

Tableau 12: Raffinement du cas d'utilisation «Se connecter»

❖ **Raffinement du cas d'utilisation « Consulter Alertes » :**

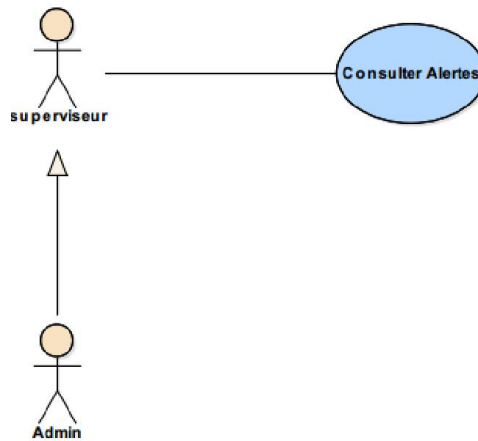


Figure 16: Raffinement du cas d'utilisation « Consulter Alertes »

Cas d'utilisation	Consulter Alertes
Acteur	Administrateur, Superviseur
Pré-Condition	S'authentifier
Post-Condition	Liste des Alertes
Scénario Principale	<ul style="list-style-type: none"> - L'utilisateur clique sur le bouton « alertes » - Un pop-up contenant les alertes s'affichent

Tableau 13: Raffinement du cas d'utilisation « Consulter Alertes »

❖ **Raffinement du cas d'utilisation « Récupérer OTP » :**

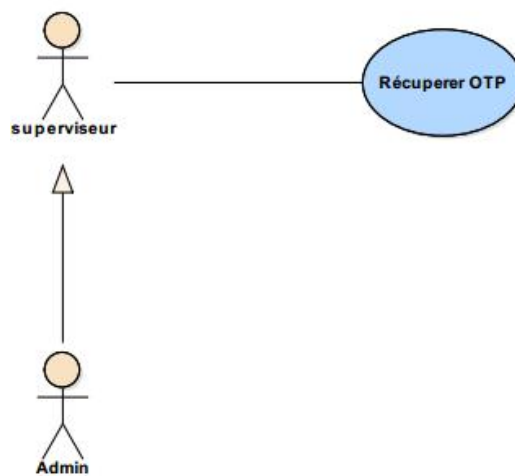


Figure 17: Raffinement du cas d'utilisation « Récupérer OTP »

Cas d'utilisation	Récupérer OTP
Acteur	Administrateur, Superviseur
Pré-Condition	S'authentifier
Post-Condition	OTP Récupéré
Scénario Principale	<ul style="list-style-type: none"> - L'utilisateur ouvre « Google Authenticator ». - Il récupère son OTP

Tableau 14: Raffinement du cas d'utilisation « Récupérer OTP »

❖ **Raffinement du cas d'utilisation « Gérer Supervision » :**

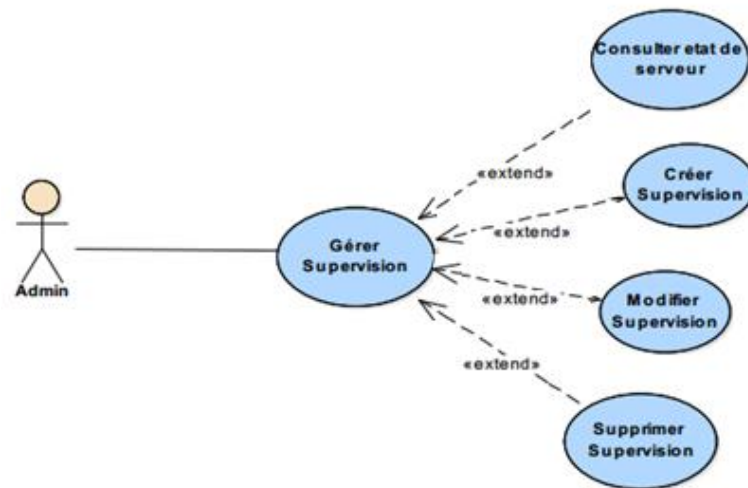


Figure 18: Raffinement du cas d'utilisation « Gérer Supervision »

- Raffinement du cas d'utilisation « Consulter Supervision » :

Cas d'utilisation	Consulter Supervision
Acteur	Administrateur, Superviseur
Pré-Condition	S'authentifier
Post-Condition	Liste des supervisions
Scénario Principale	<ul style="list-style-type: none"> - L'utilisateur ouvre la page « Monitoring ». - Un tableau de toutes les supervisions s'affiche.

Tableau 15: Raffinement du cas d'utilisation « Consulter Supervision »

- Raffinement du cas d'utilisation « Modifier Supervision » :

Cas d'utilisation	Modifier Supervision
Acteur	Administrateur, Superviseur
Pré-Condition	S'authentifier
Post-Condition	Supervision modifiée
Scénario Principale	<ul style="list-style-type: none"> - L'utilisateur ouvre la page « Monitoring ». - Un tableau de toutes les supervisions s'affiche. - Il clique sur le bouton « Modifier Supervision ». - Il modifie les données de supervision. - Il clique sur « Enregistrer ».

Tableau 16: Raffinement du cas d'utilisation « Modifier Supervision »

- Raffinement du cas d'utilisation « Supprimer Supervision » :

Cas d'utilisation	Supprimer Supervision
Acteur	Administrateur
Pré-Condition	S'authentifier
Post-Condition	Supervision Supprimé
Scénario Principale	<ul style="list-style-type: none"> - L'utilisateur ouvre la page « Monitoring ». - Un tableau contenant toutes les données supervisées s'affiche. - Il clique sur le bouton « Supprimer Supervision ». - L'administrateur sélectionne la supervision à supprimer. - Il clique bouton « Supprimer »
Exception	En cas d'erreur, afficher un message.

Tableau 17: Raffinement du cas d'utilisation « Supprimer Supervision »

- Raffinement du cas d'utilisation « Ajouter Supervision » :

Cas d'utilisation	Ajouter Supervision
Acteur	Administrateur
Pré-Condition	S'authentifier
Post-Condition	Supervision Ajouté
Scénario Principale	<ul style="list-style-type: none"> - L'utilisateur ouvre la page « Monitoring ». - Un tableau de toutes les supervisions s'affiche. - Il clique sur le bouton « Ajouter Supervision ».
Exception	En cas d'erreur, afficher un message.

Tableau 18: Raffinement du cas d'utilisation « Ajouter Supervision »

- ❖ **Raffinement du cas d'utilisation « Consulter Intervention » :**



Figure 19: Raffinement du cas d'utilisation « Consulter Intervention »

Cas d'utilisation	Consulter Intervention
Acteur	Administrateur
Pré-Condition	S'authentifier
Post-Condition	Liste des Interventions
Scénario Principale	<ul style="list-style-type: none"> - L'utilisateur ouvre la page « Interventions ». - Un tableau de toutes les interventions s'affiche.
Exception	En cas d'erreur, afficher un message.

Tableau 19: Raffinement du cas d'utilisation « Consulter Intervention »

V. Conclusion

Dans ce chapitre, nous avons énuméré les besoins fonctionnels et les besoins non fonctionnels que l'application doit offrir aux utilisateurs, en plus nous avons identifié les acteurs ainsi que les cas d'utilisations de chaque acteur.

Chapitre VI. Conception

I. Introduction

Dans ce chapitre, nous abordons la partie conception du projet, dans laquelle, nous détaillons les différents éléments de conception, à savoir les diagrammes de séquences, et le diagrammes de classe.

II. Conception détaillée

La conception est une étape importante du cycle du développement du logiciel. Elle se base essentiellement sur la bonne spécification et l'analyse des besoins. Notre démarche a débuté par la compréhension du problème. Ensuite nous avons analysé le problème pour donner une solution adéquate. A présent, nous sommes dans la phase de concevoir cette solution.

1. Diagramme de séquence

Le diagramme de séquence décrit l'aspect dynamique du système. Il modélise les interactions entre les objets ou entre utilisateur et objet, en mettant l'accent sur la chronologie des messages échangés. Dans ce qui suit, nous allons dresser les diagrammes de séquences de chaque cas d'utilisation :

a. Le cas d'utilisation « s'inscrire » :

Le système invite l'utilisateur à choisir ses paramètres d'accès (login, mot de passe, mail,...). Il essaie ensuite de vérifier l'unicité de ses paramètres

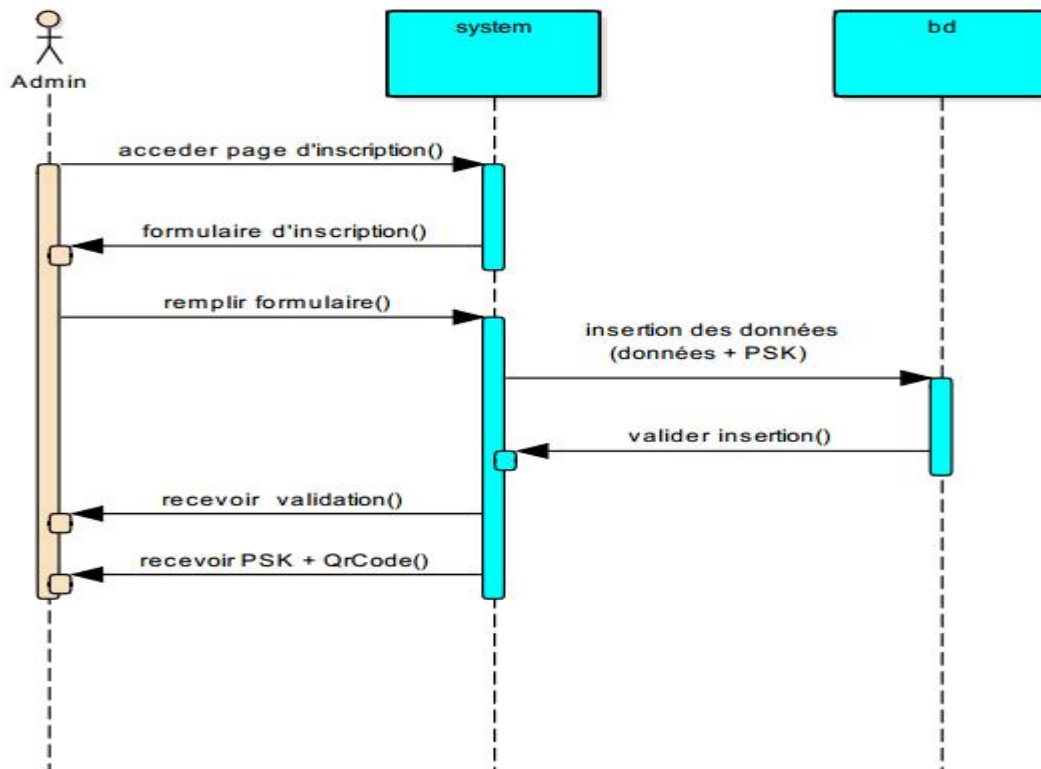


Figure 20 : Diagramme de séquence pour le cas d'utilisation "s'inscrire"

b. Le cas d'utilisation « Récupérer OTP » :

Lorsque tout est réglé, un message de confirmation évoque le succès de l'inscription accompagné par un PSK et un QR Code générés automatiquement s'affiche. Une fois le QR Code ou le PSK sont entrés dans Google Authenticator, L'OTP s'affiche.

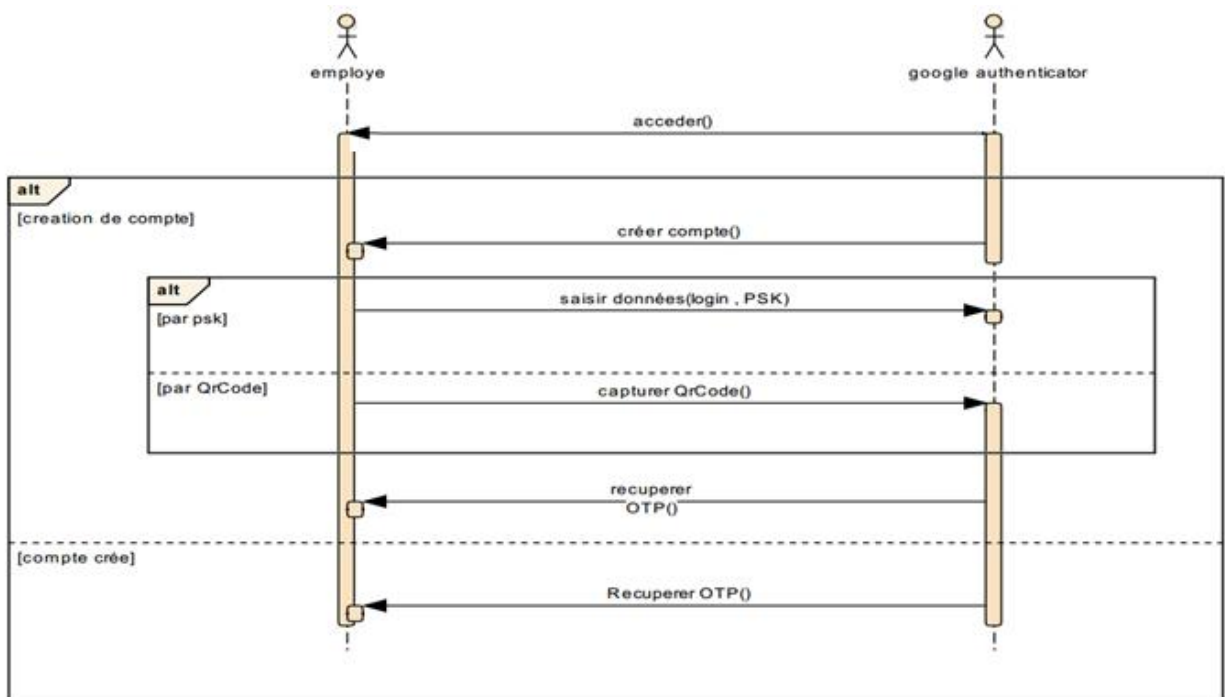


Figure 21 : Diagramme de séquence pour le cas d'utilisation "Récupérer OTP"

c. Le cas d'utilisation « se connecter » :

C'est le premier scénario qui se déroule lors du déclenchement de l'application. Selon cette figure, le superviseur est invité à saisir ses paramètres de connexion(login, mot de passe, OTP).

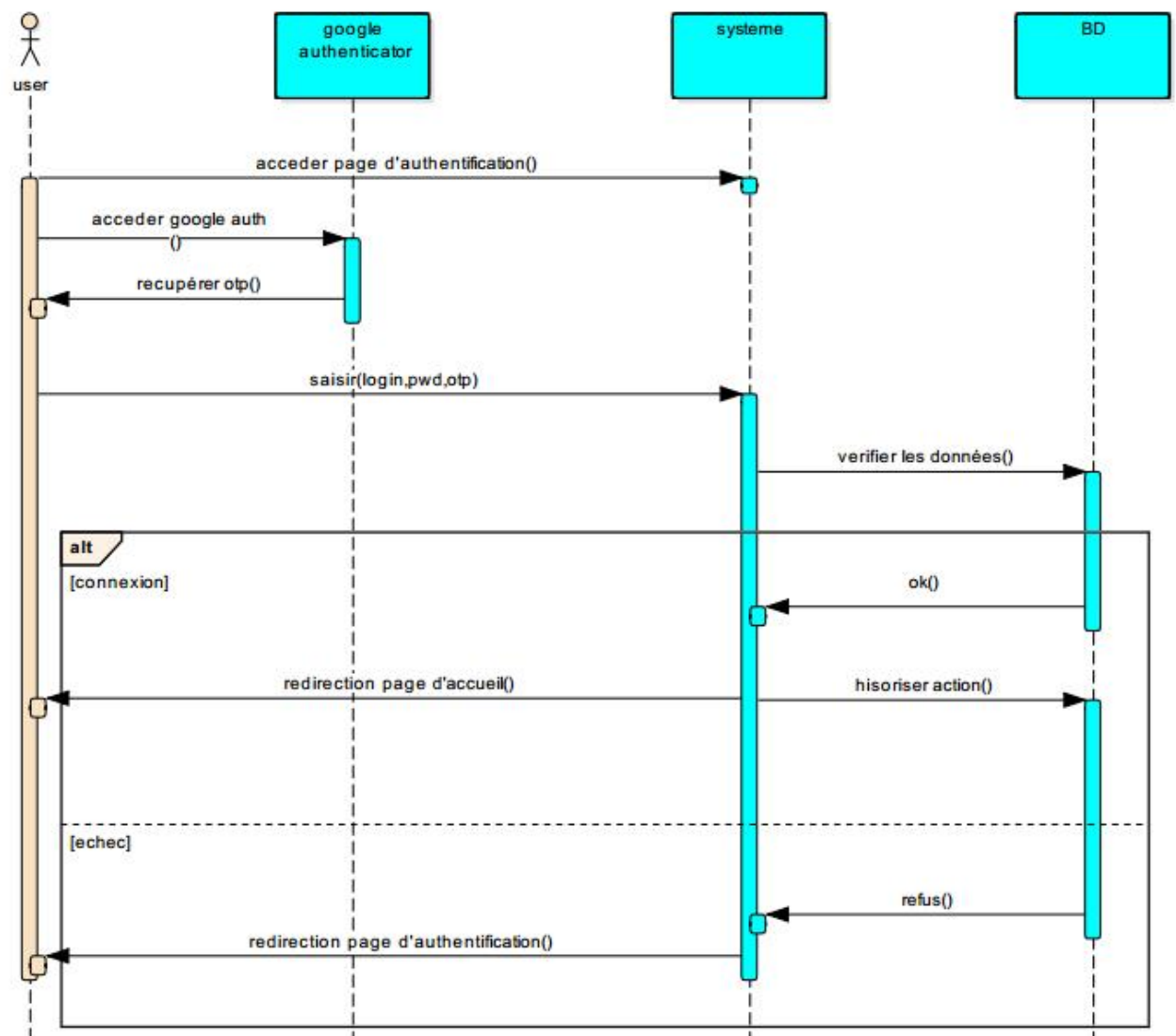


Figure 22 : Diagramme de séquence pour le cas d'utilisation "se connecter"

d. Le cas d'utilisation « Superviser Serveur Mail » :

Le système vérifie le bon fonctionnement du serveur Mail.

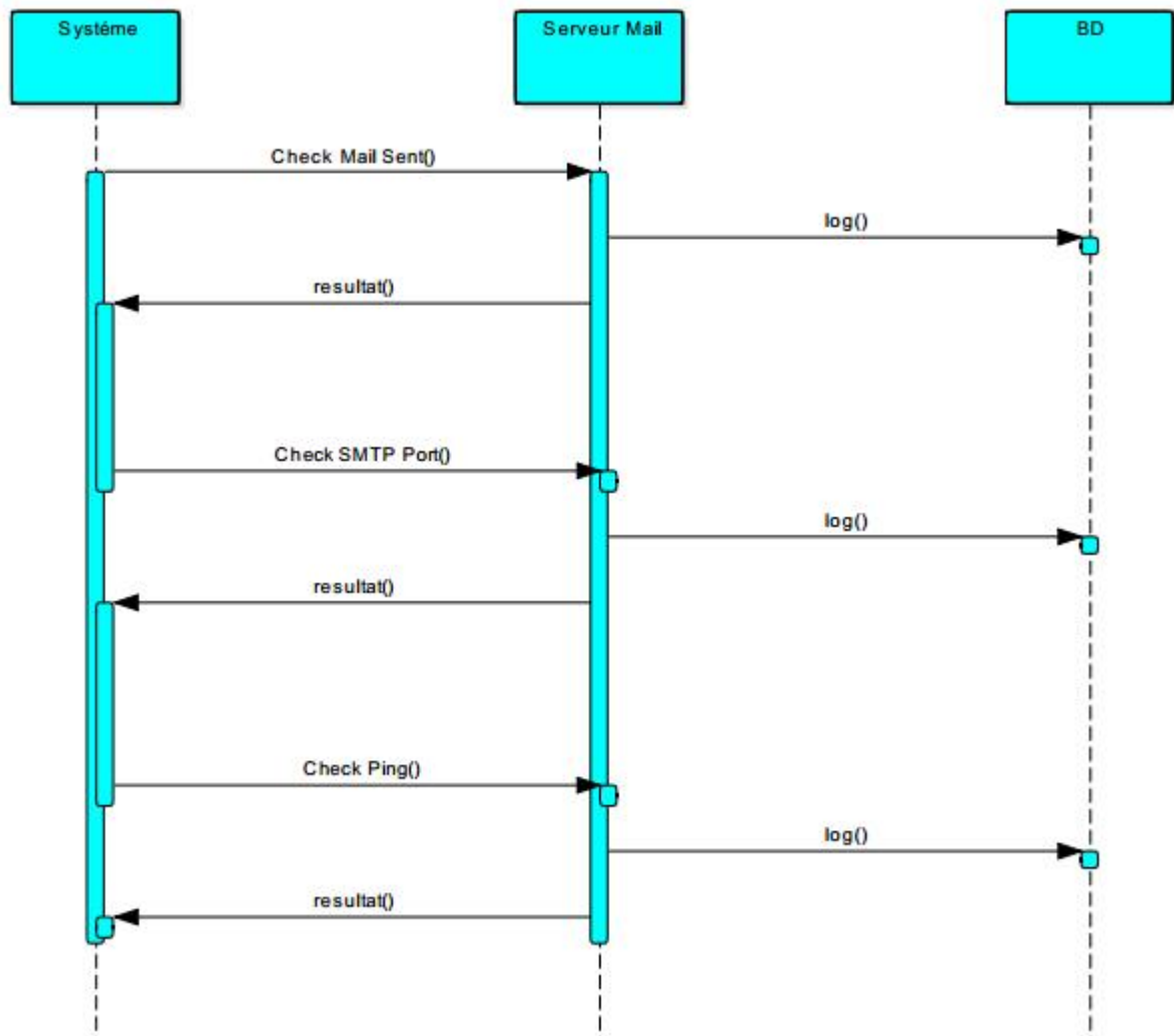


Figure 23 : Diagramme de séquence de cas d'utilisation "Superviser Serveur Mail"

e. Le cas d'utilisation « Superviser Serveur Web » :

Le système vérifie le bon fonctionnement du serveur Web.

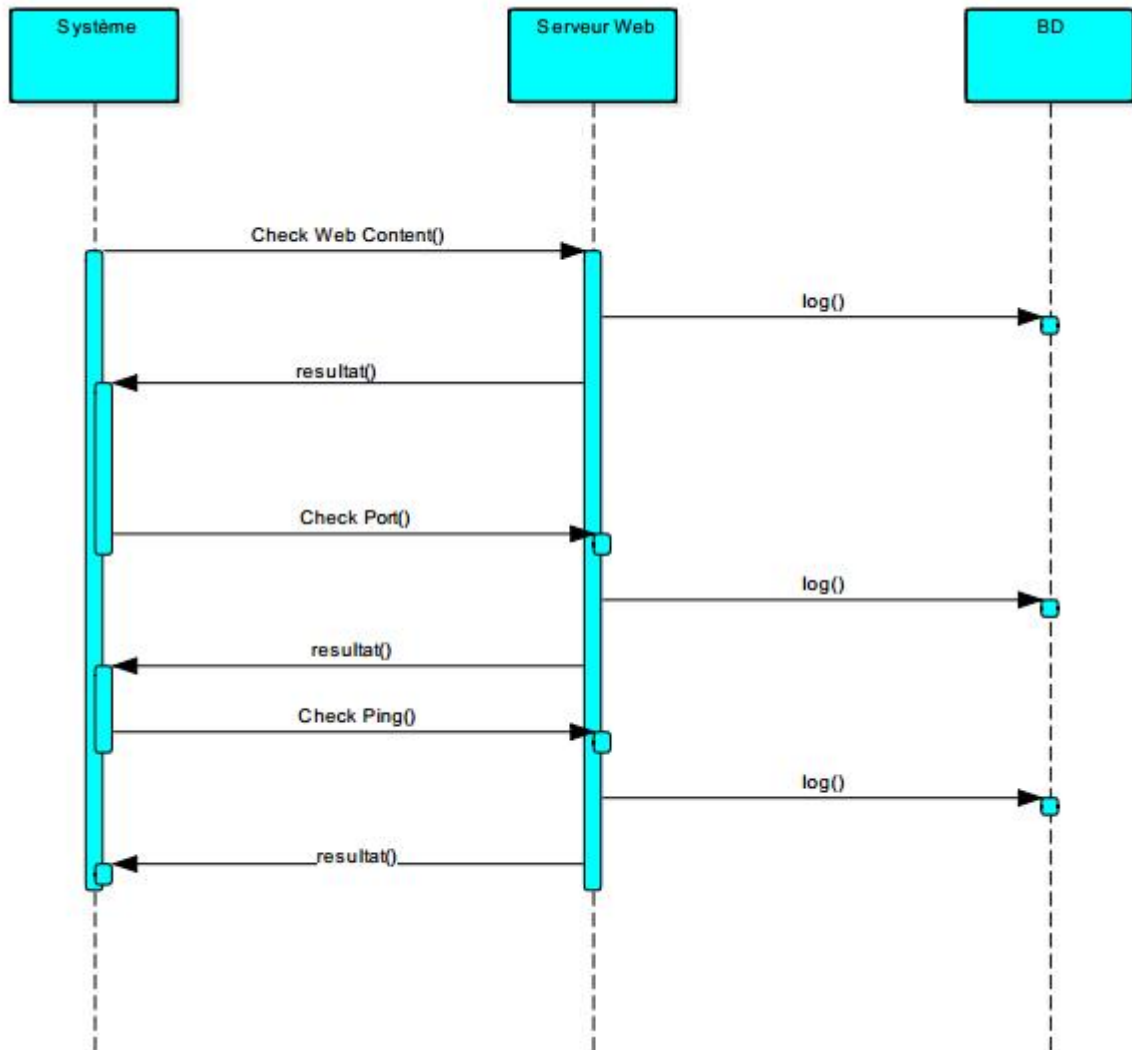


Figure 24 : Diagramme de séquence de cas d'utilisation "Superviser Serveur Web"

f. Le cas d'utilisation « Superviser Serveur Base de Données » :

Le système vérifie le bon fonctionnement du serveur de Base de Données.

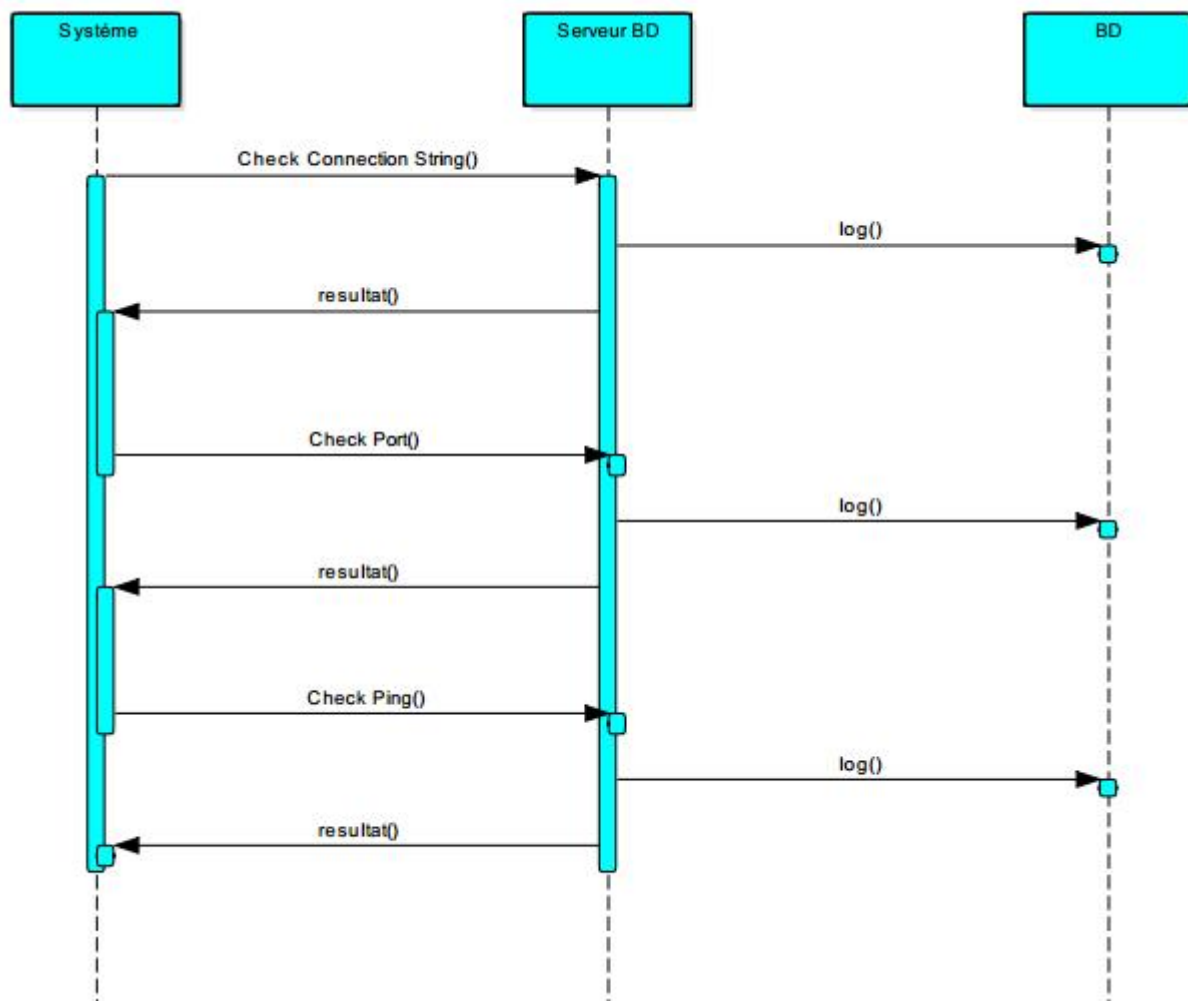


Figure 25 : Diagramme de séquence de cas d'utilisation "Superviser Serveur BD"

g. Le cas d'utilisation « Noyau Monitoring» :

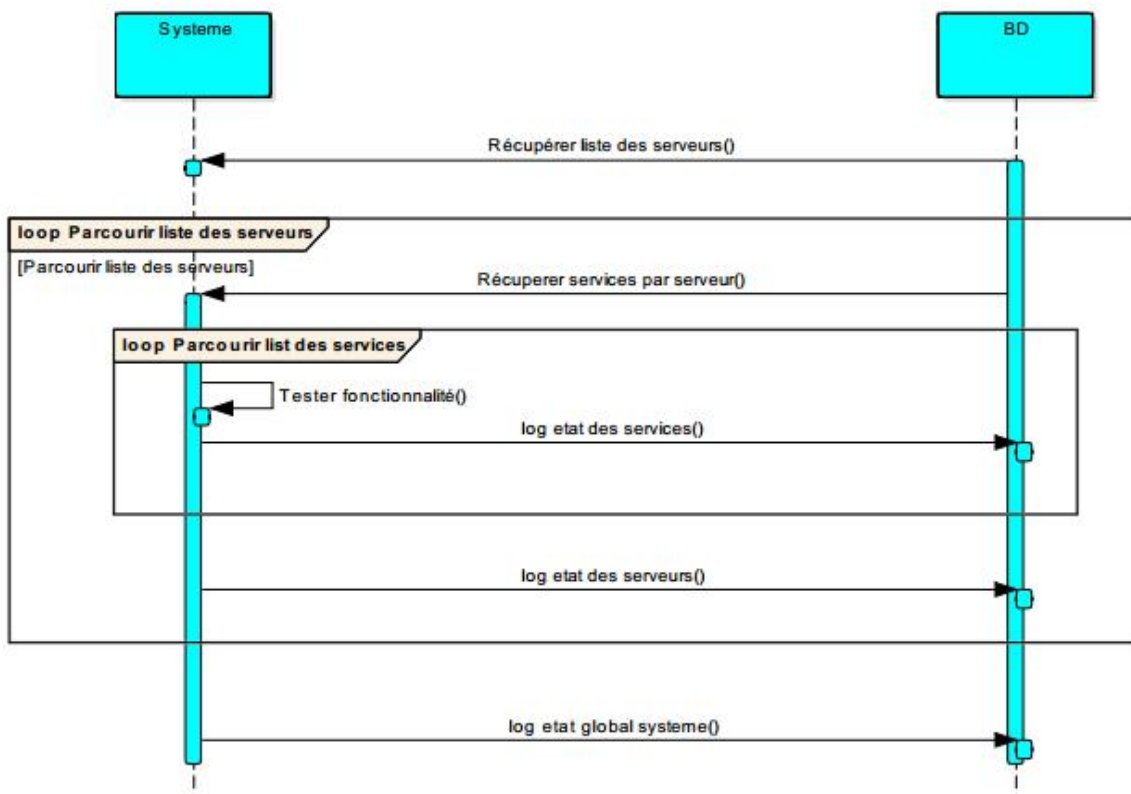


Figure 26 : Diagramme de séquence de cas d'utilisation "Noyau Monitoring"

h. Le cas d'utilisation « Accorder Privilège» :

Le système invite l'administrateur à déverrouiller l'accès au site pour les employés récemment inscrits.

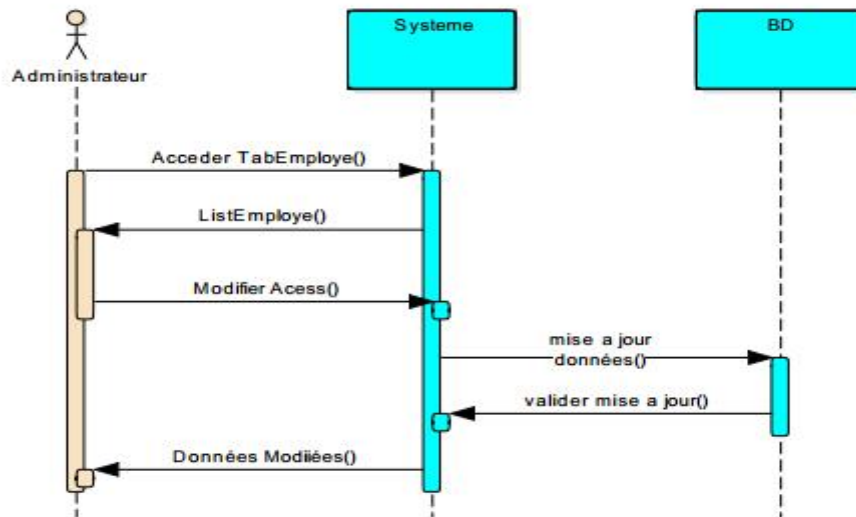


Figure 27 : Diagramme de séquence de cas d'utilisation "Accorder Privilège"

2. Le diagramme de classe

Le diagramme de classes est une modélisation statique du système en termes de classes et de relations entre ces classes. Son intérêt réside dans la modélisation des entités du système d'information.

Dans notre application, nous avons pu distinguer les classes suivantes

- **Superviseur** : C'est la classe contenant toutes les informations concernant le superviseur.
- **Administrateur** : C'est un superviseur ayant tous les droits sur tous les composants de système (serveur, service, supervision, superviseur,...).
- **Serveur** : C'est la classe contenant toutes les informations concernant un serveur.
- **Service** : C'est la classe contenant tous les détails concernant un service.
- **Supervision** : C'est la classe dans laquelle est affecté une priorité à chaque service afin d'assurer une certaine hiérarchie dans l'ensemble de supervision par rapport à un serveur.
- **Log** : elle sert à sauvegarder toutes les résultats des tests et de contrôles effectués.
- **Intervention** : elle représente chaque intervention faite par un superviseur sur un serveur ou un service s'il a un problème.
- **Droit** : représente l'ensemble des droits que possède un superviseur sur un serveur.

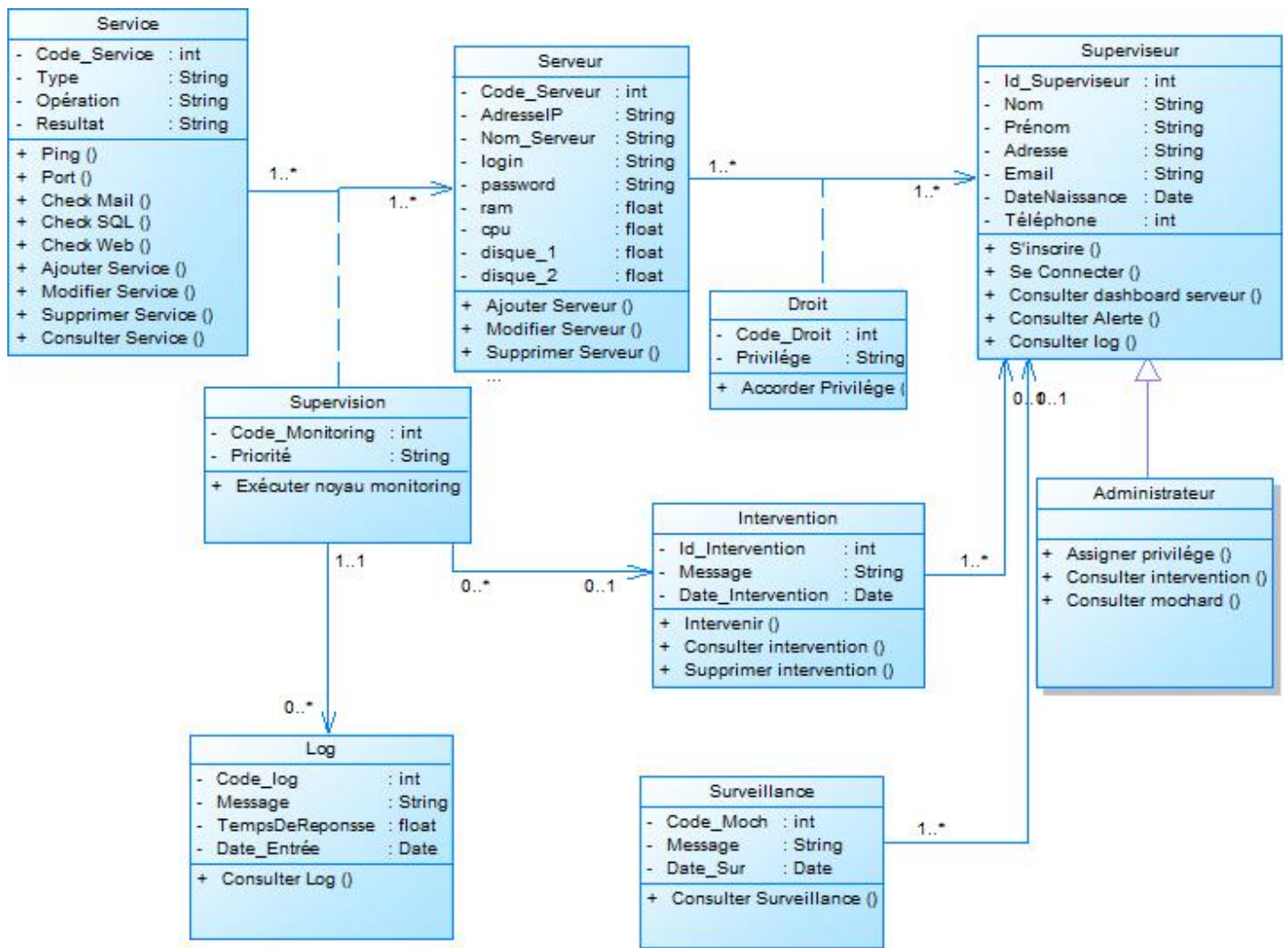


Figure 28: Diagramme de classe de l'application

3. La base de données de l'application

❖ Schéma relationnel de la base de données

Employé

(#IDSuperviseur, Nom, Prénom, Adresse, DateDeNaissance, Email, Téléphone, PSK, Login, Password, Access)

Droit (#UserID, #CodeServeur, IdDroit, Privilège)

Serveur (#CodeServeur, AdresseIP, Nom, Login, Password, RAM, CPU, Disque1, Disque2)

Service (#CodeService, Type, Opération, Résultat)

Monitoring (#CodeMonitoring, #CodeServeur, #CodeService, Priorité, Dépendance)

Log (#CodeLog, #CodeMonitoring, IdScan, Message, Temps, TempsDeReponse, Etat)

Intervenir (#CodeIntervention, #IDSuperviseur, #CodeMonitoring, Problème, Solution, Temps)

Surveillance (#CodeMochard, #UserID, Message, Date)

III. Conclusion

Dans ce chapitre nous avons détaillé les différentes vues conceptuelles de notre application à travers les modèles UML nécessaires. Cette conception est essentielle pour la phase de réalisation qui constitue l'objet du chapitre suivant.

VII. Réalisation

I. Introduction

Dans ce chapitre, nous présentons l'environnement matériel et logiciel du projet. Ensuite, nous nous intéressons à la description de quelques interfaces du système implémenté dans le cadre de quelques scénarios d'utilisation.

II. Environnement de travail

1. Environnement matériel et logiciel

- Système d'exploitation : Windows 10
- Système d'exploitation de serveurs : Windows Server
- SGBD : SQL Server 2014
- IDE de développement : Visual Studio 2015
- Outil pour la conception : Power AMC, Entreprise Architect

2. Choix Technologiques

Pour la réalisation de ce projet nous avons choisi de travailler avec :

✓ *Visual Studio comme IDE*

Visual Studio .NET fournit un environnement de développement intégré (IDE) qui permet aux développeurs de créer des solutions plus rapidement en utilisant des fonctionnalités de productivité clé accessibles avec n'importe quel langage .NET. L'IDE est un environnement personnalisable qui permet aux développeurs les meilleures performances. Il fournit un accès unifié aux concepteurs, éditeurs et outils de Visual Studio à partir de n'importe quel langage .NET.

✓ *.NET comme Framework*

Le Framework .NET s'appuie sur la norme Common Language Infrastructure (CLI) qui est indépendante du langage de programmation utilisé. Ainsi tous les langages compatibles respectant la norme CLI ont accès à toutes les bibliothèques installées (installables) dans l'environnement d'exécution. Le Framework .NET est un sous ensemble de la technologie Microsoft .NET.

✓ *ASP.NET comme langage de programmation*

ASP.NET est un langage de programmation orienté objet conçu par Microsoft, avec un déploiement simplifié autour d'un environnement de développement cohérent qui peut permettre le développement d'applications très puissantes et sophistiquées.

✓ ***Web Forms comme architecture***

- Modèle d'événement : Web Forms prend en charge un style de programmation piloté par événement qui est semblable à celui des applications Windows. De nombreux événements sont disponibles.
- Gestion d'état : Web Forms réduit la complexité de la gestion d'état au moyen de l'état d'affichage et des contrôles serveur.
- Architecture basée sur des pages : Web Forms fournit une architecture qui combine une page comportant un balisage déclaratif (fichier .aspx) et un fichier code-behind ajoutant des fonctionnalités. Cette structure facilite la création des pages qui implémentent des tâches courantes, telles que la réponse aux mouvements de l'utilisateur et le rendu du balisage à partir du code serveur.

✓ ***IIS comme serveur HTTP***

- Le moteur ASP.NET fonctionne avec un pipeline de traitement des requêtes web: Lorsque l'utilisateur navigue sur un hyperlien, une demande est envoyée au serveur web IIS. La demande traverse ensuite le pipeline d'ASP.NET jusqu'au point de sortie où un développeur d'application peut attacher le code nécessaire au traitement de cette demande. Le moteur est indépendant du serveur web IIS.
- Lorsque le serveur web reçoit une demande de page web, il détermine tout d'abord s'il s'agit de contenu statique (enregistré dans un fichier) ou de contenu dynamique (généré sur demande). L'examen tient compte de l'extension du fichier. Selon la configuration, les fichiers qui se terminent par .aspx, .asax, .ascx, .ashx, .asmx, .axd ou .svc seront transmis à ASP.NET via aspnet_isapi.dll, une bibliothèque branchée à l'interface ISAPI de IIS. La requête sera envoyée à un récepteur différent selon son extension⁷. ASP.NET exécute ensuite un programme, puis le résultat du programme est renvoyé au serveur web qui le transmettra au navigateur web à l'origine de la demande.
- Les pages web (.aspx) ainsi que des services web (.asmx) sont mis en œuvre en utilisant deux récepteurs intégrés à ASP.NET. Beaucoup de développeurs d'application se servent uniquement de ces deux récepteurs et le fait qu'il est possible pour les développeurs d'application de brancher leur propre récepteur est souvent inconnu ou pas maîtrisé.

✓ ***Apache Cordova***

Apache Cordova ou plus anciennement Apache Callback ou PhoneGap1, est un Framework open-source développé par la Fondation. Il permet de créer des applications pour différentes plateformes (Android, Firefox OS, iOS, Ubuntu, Windows 8...) en HTML, CSS et JavaScript.

✓ *Google Authenticator*



Figure 29 : Google Authenticator

Google Authenticator est un logiciel open source basé sur l'authentification en deux étapes, développé par Google. Le logiciel fournit un nombre de 6 chiffres que l'utilisateur doit donner lors de son authentification, en plus de son pseudo et de son mot de passe. La validité du numéro à 6 chiffres dure seulement 30 secondes. Après cela, un nouveau numéro est généré et affiché.

Google Authenticator fonctionne dans un téléphone portable ; quelque chose que vous possédez. En exécutant Google Authenticator sur votre téléphone et en entrant le code qu'il génère, vous prouvez la propriété. La propriété que vous prouvez n'est pas celui de votre téléphone portable, mais de secret pré-partagée entre vous et l'application (PSK : PreShared Key).

PSK : Une clé pré-partagée (PSK) est un secret partagé qui était auparavant partagée entre les deux parties en utilisant une voie de communication protégée avant qu'il ne doit être utilisé.

Pour construire une clé de secret partagé, la fonction de dérivation de clé est généralement utilisée. De tels systèmes utilisent presque toujours des algorithmes de cryptographie à clé symétrique

III. Réalisation partie serveur

Cette figure contient les méthodes implémentées au niveau du serveur : (Voir Annexe pour montrer les étapes de déploiement)

« deployment spec » WCF service	
-	rechPSK(string login, String pwd): string
-	inscription(string username, string lastname, string adresse, string birthdate, string email, int tel, string login, string pwd): string
-	ValidOTP(string login, string pwd, String token): boolean
-	noyau(): string
-	wcflog(int codeserveur, int codeservice, int idscan, string msg, int reussi, string tdr): boolean
-	ping(String url): dataset
-	wcfport(String ip, int port): dataset
-	wcfMailSmtP(string operation, string res): dataset
-	wcfSql(int CodeServeur, string operation, string res): dataset
-	wcfUrl(string url, string res): dataset
-	wcfService(string type, int CodeServeur, string operation, string res, string destinataire, int port): dataset
-	StatService(int id, int idservice): dataset
-	StatServeur(int id): dataset
-	StatAdmin(): dataset
-	ConsulterAlert(): dataset
-	VerifAdmin(string id): boolean
-	ConsulterLog(): string
-	ConsulterServeur(): string
-	ConsulterEmploye(): string
-	ConsulterService(): string

Figure 30 : Package de web service de l'application

1. Déploiement de noyau dans Windows Service

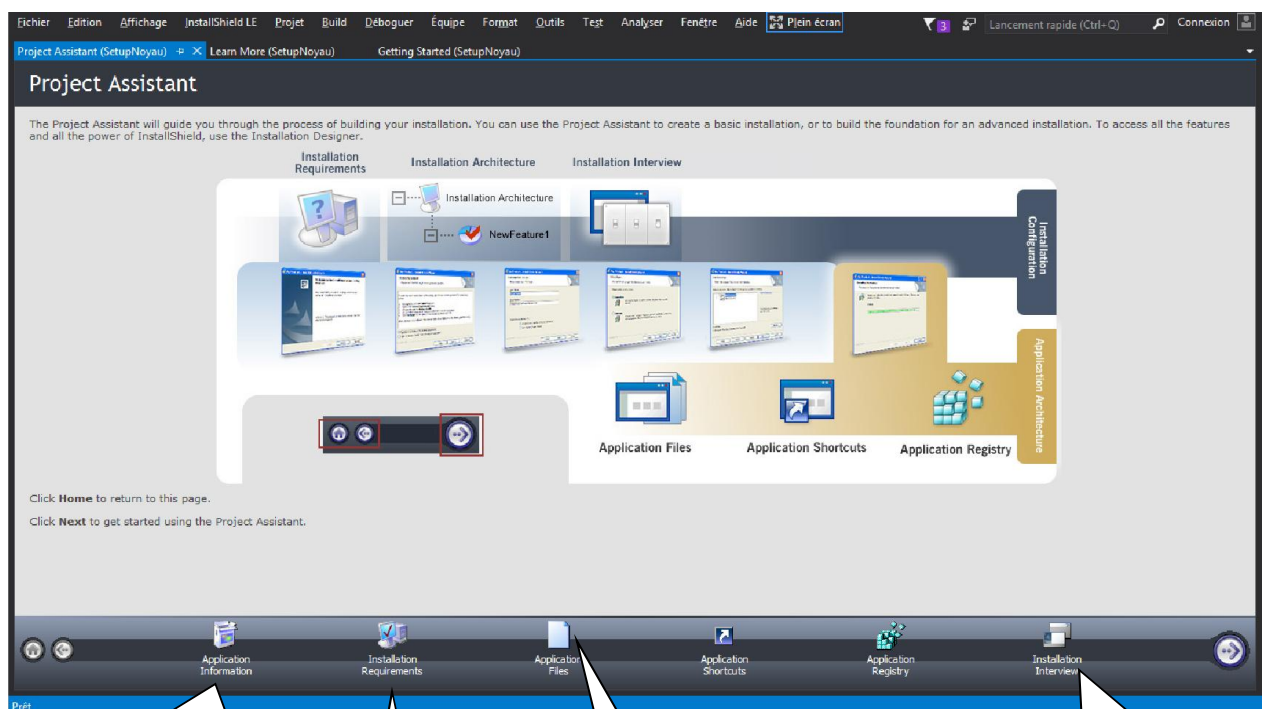


Figure 31 : Setup Windows Monitoring

1-Préciser des informations sur l'application (nom, Company Name, version,...)

2-les exigences d'installation

3-Les fichiers de l'application

4-Fin de Préparation de Setup de l'application

2. Installation de Windows Service de «Noyau Monitoring »

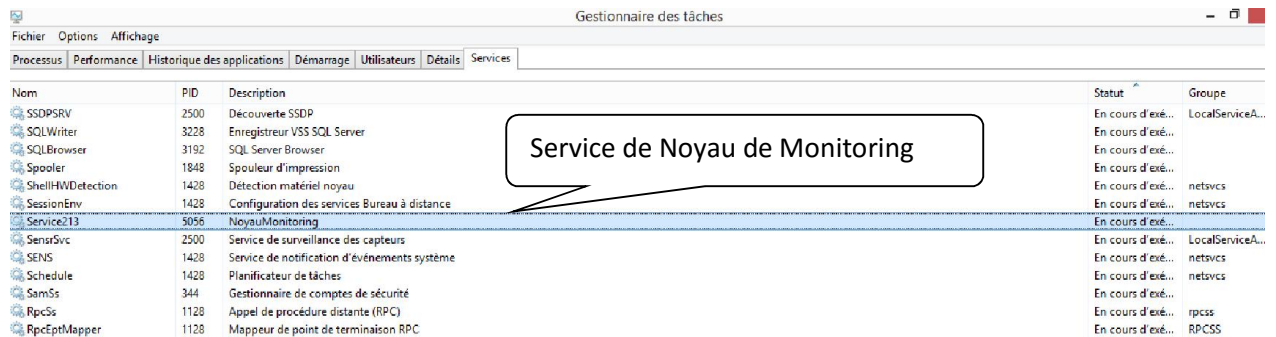
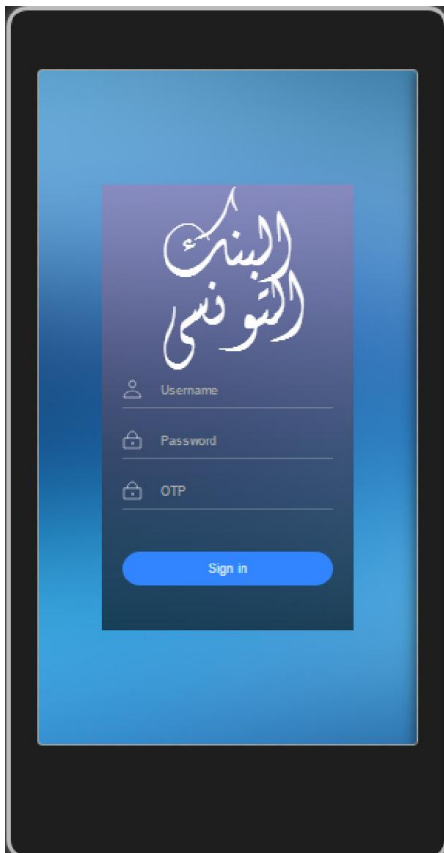


Figure 32 : Liste des services

IV. Interfaces de l'application

Nous exposerons quelques interfaces de notre application, en essayant à chaque fois de décrire les différents objets interactifs mis à la disposition de l'utilisateur

1. Application Mobile



le client est invité à s'authentifier en saisissant ses paramètres d'accès(Login,mot de passe,OTP)

Figure 33 : Page d'accueil



Figure 34 : Dashboard Admin



Figure 35 : Dashboard Admin2

Cette page contient toutes les données générales sur les serveurs tels que l'état actuel, la date du dernier scan, les derniers états des services et l'historique qui contient le résultat de tous les contrôles effectués sur les serveurs



Cette page contient toutes les données concernant un serveur tel que son état actuel, la date du dernier scan, les derniers états de ses services et l'historique qui contient le résultat de tous les contrôles sur ce serveur



Figure 37 : Menu Admin



Figure 38 : Menu Superviseur

Le Menu de l'administrateur lui permet de tout consulter (les serveurs, les services, les employés,...)

Le menu de superviseur lui permet de consulter l'état de serveur auquel il a droit, les données sur ce serveur, l'historique des tests faits sur le serveur.

Code Serveur	Adresse	nom	login	Password
1	128.25.125.2	web	1	1
2	dns	/d56'	125.26.35.2	546*-
5	154.235.60.3	sql	3	3
50	125.66.256.3	url	2	2

Figure 39 : Liste des serveurs

Cette page contient la liste des services, chaque service est caractérisé par son adresse, son nom, son identifiant et son mot de passe

code	type	operation1	operation2	resultat attendu
1	url	google.com	null	google
2	2	2	null	2
3	sql	select version	null	@2015
4	ping	url/ip	null	true
5	9	9	null	9
6	url	www.google.com	null	google

Figure 40 : List des services

Cette page contient la liste des serveurs, chaque serveur est accompagné de son adresse, son identifiant et son mot de passe

The screenshot shows a web application interface titled 'Check Employees'. It features a sidebar menu on the left and a main content area. The main area has a header with a timestamp '09 : 54 : 43'. Below the header is a section titled 'EMPLOYEES LIST' containing a table with the following data:

Id	prenom	nom	adresse	email
1	admin	admin	hazem 44	1
2	user1	user1	test ddd	test
3	lalala	lalal	alkj	ooooo
5	55555	9999	null	email
14	jajajaj	ndndndn	null	kkkkk
15	545	54454	null	45545

Figure 41 : Liste des superviseurs

Cette page contient toutes les données personnelles et les paramètres d'accès de tous les superviseurs. En plus, elle donne la possibilité de gérer (ajouter nouvel superviseur, modifier les données d'un superviseur, retirer un superviseur de la liste)

The screenshot shows a web application interface titled 'History'. It features a sidebar menu on the left and a main content area. The main area has a header with a timestamp '09 : 57 : 36'. Below the header is a section titled 'HISTORY' containing a table with the following data:

Code Serveur	Code Service	Message log	Temps	Etat
5	etat de serveur	problem dans le serveur 5	2016-04-21T10:11:26.203	
5	8	ping false	2016-04-21T10:11:26.187	
5	7	port lalal false	2016-04-21T10:11:26.163	
2	etat de serveur	serveur 2 fonctionne parfaitement	2016-04-21T10:11:26.127	
1	etat de serveur	serveur 1 fonctionne parfaitement	2016-04-21T10:11:26.097	

Figure 42 : Historique

Cette page contient l'historique de tous les tests, chaque ligne de log est caractérisé par son code log, le code de service qu'on a testé, le temps de réponse et la date de ce test, et un bouton qui s'allume en vert si le service fonctionne parfaitement, en jaune si le service nécessite beaucoup de temps pour répondre et en rouge si le service ne fonctionne pas

2. Application Web

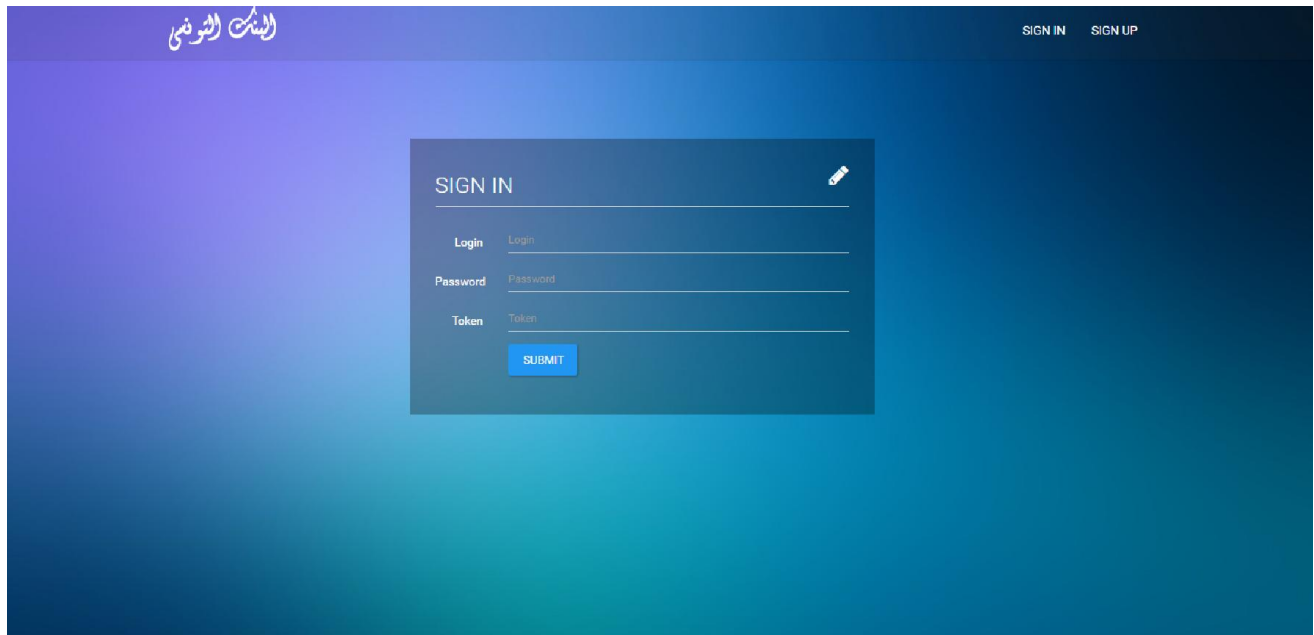


Figure 43 : Page de connexion

Lors de lancement de l'application, une interface apparaît mentionnant le nom l'entreprise éditrice de la solution à savoir Banque de Tunisie. Ensuite le client est invité à s'authentifier en saisissant ses paramètres d'accès (login, mot de Lorsque le client n'admet pas de compte, il pourrait accéder aux fonctionnalités d'ajout d'un membre

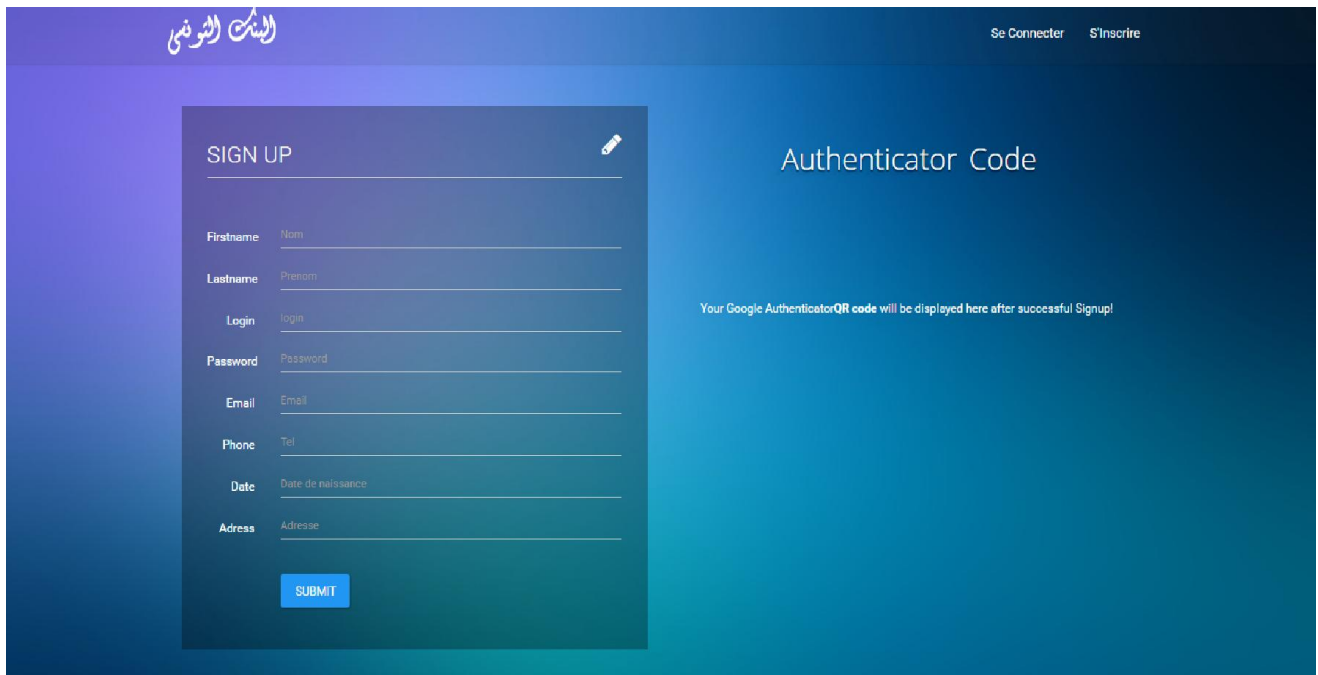


Figure 44 : Page d'inscription

Si un superviseur ne possède pas de compte, il pourrait accéder aux fonctionnalités d'ajout d'un membre tout en saisissant quelques informations personnelles et les paramètres d'accès

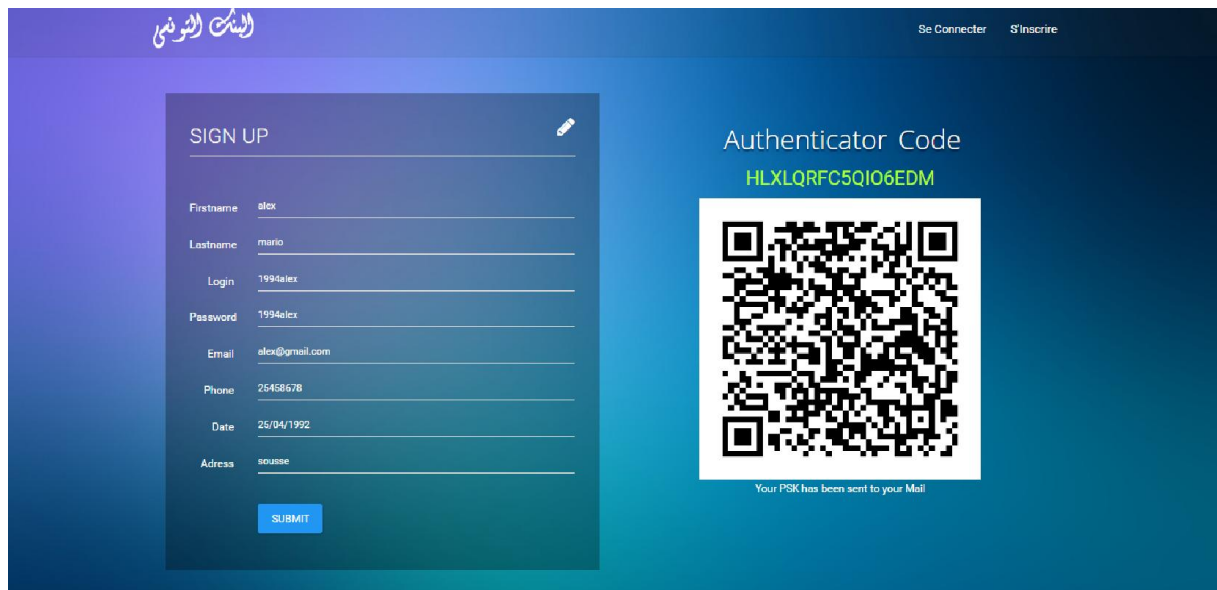


Figure 45 : Inscription réussie

En cas d'une inscription réussie, Un QR Code et un PSK seront générés à l'utilisateur afin qu'il puisse recevoir son OTP



Figure 46:Menu Superviseur

Le menu de superviseur lui permet de consulter l'état de serveur auquel il a droit, les données sur ce serveur, l'historique des tests faits sur le serveur.



Figure 47:Menu Admin

Le Menu de l'administrateur lui permet de tout consulter(les serveurs, les services, les employés,...)

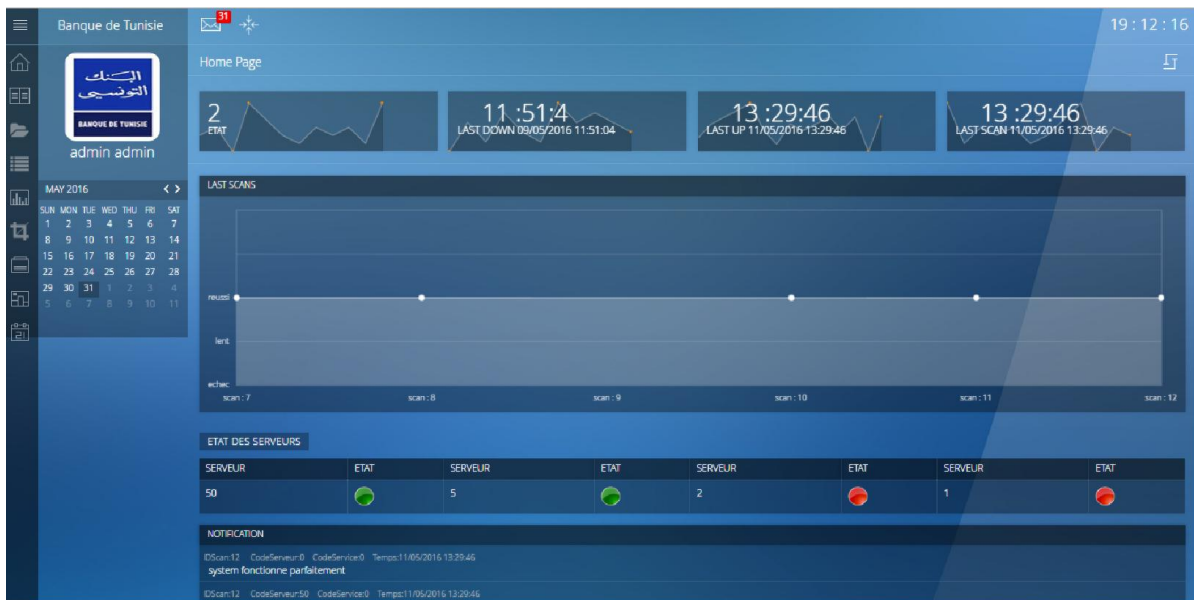


Figure 48 : Dashboard Admin

Cette page contient toutes les données concernant tous les serveurs tels que leur état actuel, la date du dernier scan, les derniers états de ses services et l'historique qui contient le résultat de tous les contrôles sur tous les serveurs



Figure 49 : Dashboard Superviseur

Cette page contient tous les données concernant un serveur tel que son état actuel, la date du dernier scan, les derniers états de ses services et l'historique qui contient le résultat de tous les contrôles sur ce serveur



NO	NAME	ADRESS	LOGIN	PASSWORD	
1	web	128.25.125.2	1	1	delete
2	/rdb	125.26.35.2	dba	546P-	delete
5	sql	154.235.60.3	3	3	delete
50	url	125.66.256.3	2	2	delete

Figure 50 : Liste des serveurs

Cette page contient la liste des serveurs, chaque serveur est accompagné avec son adresse, son identifiant et son mot de passe



CODE LOG	CODESERVICE	MESSAGE	TEMPS DE REPONSE	TIME	
2938	0	system fonctionne parfaitement	11/05/2016 13:29:46	00:00:00	 delete
2937	0	serveur 50 fonctionne parfaitement	11/05/2016 13:29:46	00:00:00	 delete
2936	9	service 9 mail fonctionne parfaitement	11/05/2016 13:29:46	00:00:00	 delete
2935	6	service 6 url fonctionne parfaitement	11/05/2016 13:29:46	00:00:00.6904899	 delete
2934	0	serveur 5 fonctionne parfaitement	11/05/2016 13:29:45	00:00:00	 delete

Page: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21

Figure 51 : Historique

Cette page contient l'historique de tous les tests, chaque ligne de log est caractérisé par son code log, le code de service qu'on a testé, le temps de réponse et la date de ce test, et un bouton qui s'allume en vert si le service fonctionne parfaitement, en jaune si le service nécessite beaucoup de temps pour répondre et en rouge si le service ne fonctionne pas









LIST OF EMPLOYEES										
	ID	Lastname	Firstname	Adress	Phone	Birthdate	Login	Password	PSK	Access
	1	admin	admin	hazem 44	122555	1	55	55	R5FCPDHM5QC5CAV2	unlocked
	2	user1	user1	test ddd	87	test	1	1	RXUXFZ6H2KEOCPIV	unlocked
	3	lalala	lalal	akjllll	546	ssss	ssss	eeeeee	X7WPXWDF5LRYWREZ	locked
	5	55555	9999		0		44	pwd	555	locked
	14	jajajaj	ndndndn	null	11188	null	kkk55k	ahahahah	W22HTWU52LEXB05Z	locked
1234										

Figure 52 : Table des superviseurs






Cette page contient tous les données personnelles et les paramètres d'accès de tous les superviseurs. En plus, elle donne la possibilité de gérer (ajouter un nouvel superviseur, modifier les données d'un superviseur, retirer un superviseur de la liste)

MONITORING TABLE				
	Code	Code Server	Code Service	Priority
	1	50 url	6 url	1
	2	5 sql	7 port	2
	3	5 sql	8 ping	3
	9	5 sql	6 url	1
	1011	50 url	9 mail	2
	1023			1

Add Line

Figure 53 : Table Monitoring

Cette page représente la table supervision, chaque supervision est caractérisée par le nom de serveur à contrôler, le service à contrôler et son priorité dans le système de supervision

AFFECT PRIVILEGE				
	ID	code Employee	code Server	Privilège
	1	1admin admin *	5 sql *	edit *
	2	1admin admin	1 web 2 /d56*	edit
	3	1admin admin	5 sql 50 url	edit
	8	1admin admin	50 url	check
	9	2user1 user1	50 url	edit

Add Privilege

Figure 54 : Table de gestion de privilège

C'est la page dans laquelle l'administrateur accorde des droits tels que la lecture et la modification pour les superviseurs sur des serveurs.

CALENDAR						
< > MAY 2016						
SUN	MON	TUE	WED	THU	FRI	SAT
1	2	3	4	5	6	7
8	9	10	11	12	13	14
	11:33a port lalal false	10:28a serveur 1 fonctionne parfaitement	1:28p serveur 1 fonctionne parfaitement			
	11:33a serveur 1 fonctionne parfaitement	10:28a problem dans le serveur 5 ne fonctionne pas parfaitement	1:28p serveur 2 fonctionne parfaitement			
	11:33a serveur 2 fonctionne parfaitement	10:28a problem dans le serveur 50 ne fonctionne pas	1:29p serveur 1 fonctionne parfaitement			
	11:33a url lalal false		1:29p serveur 2 fonctionne parfaitement			
	11:33a ping false		1:29p serveur			

Figure 55 : Calendrier

Historisation de tous les problèmes sur les serveurs dans un calendrier

CODE LOG	CODESERVICE	MESSAGE	TEMPS DE REPONSE	TIME	
9	2	50	appel au ingénieur responsable problem dans le ping	24/05/2016 00:00:00	delete
8	2	50	55 55	24/05/2016 00:00:00	delete
7	1	50	test test	24/05/2016 00:00:00	delete

Figure 56 : Page intervention

Cette une table qui contient tous les interventions faites par les superviseurs sur les serveurs et ses services.

CODE MOCHARD	CODE EMPLOYE	MESSAGE	DATE	
1	1	log in	16/05/2016 14:14:45	delete
2	1	log in	16/05/2016 15:00:28	delete
3	1	log out	16/05/2016 15:02:00	delete
4	2	log in	16/05/2016 15:02:40	delete
5	0	acces au serveur 50	16/05/2016 15:06:28	delete
6	2	log in	16/05/2016 15:07:10	delete
7	2	acces au serveur 50	16/05/2016 15:07:10	delete
8	2	acces au serveur 50	16/05/2016 15:07:17	delete
9	2	acces au service de serveur	16/05/2016 15:07:20	delete
10	2	acces au serveur 50	16/05/2016 15:07:27	delete

Page: 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8 - 9 - 10 - 11 - 12

Figure 57 : Control des Superviseurs

Historisation de tous les actions faits par les superviseurs dans le site (log in, log out, signaler un problème, ne rien signaler,...)

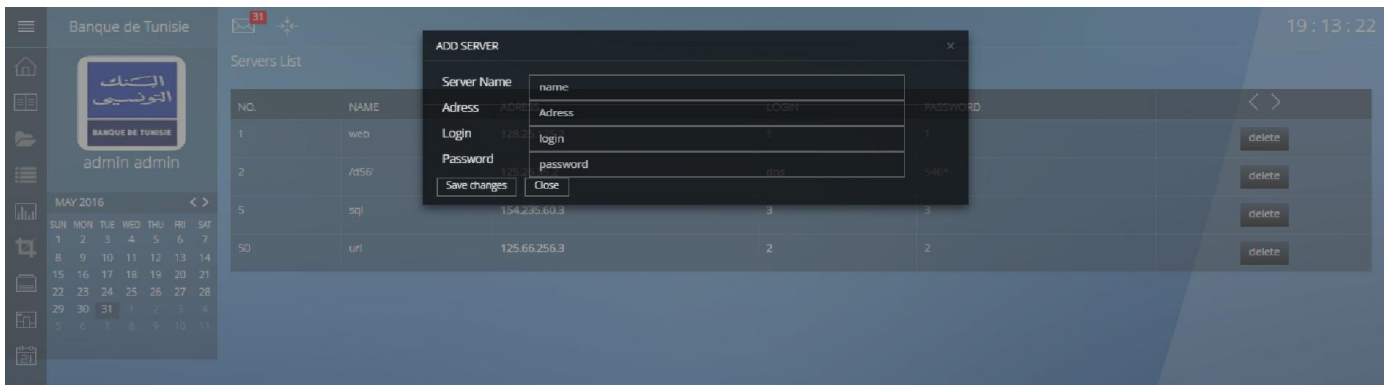


Figure 58 : Ajouter Serveur

C'est un Pop-up qui nous permet d'ajouter un nouveau serveur

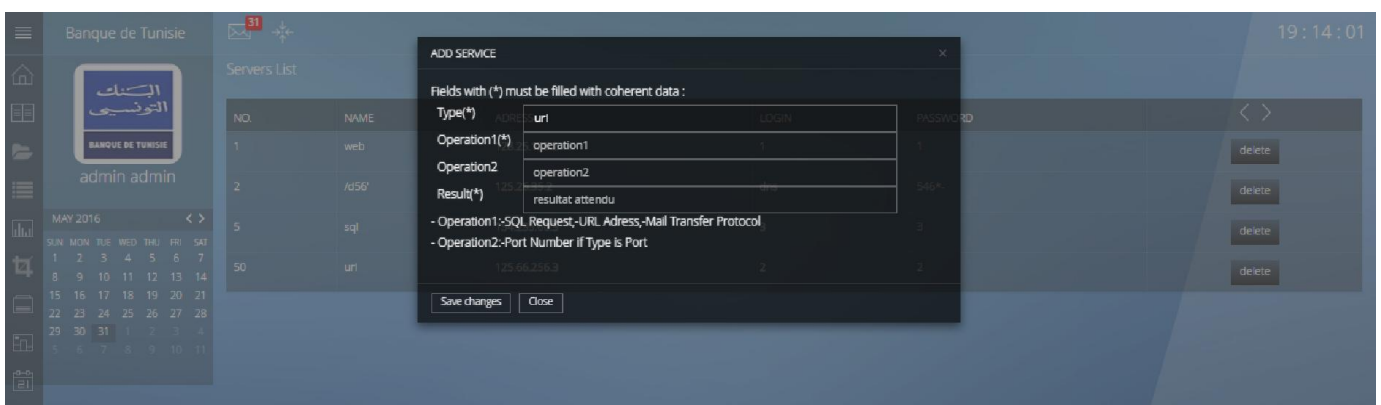


Figure 59 : Ajouter Service

C'est un Pop-up qui nous permet d'ajouter un nouveau service

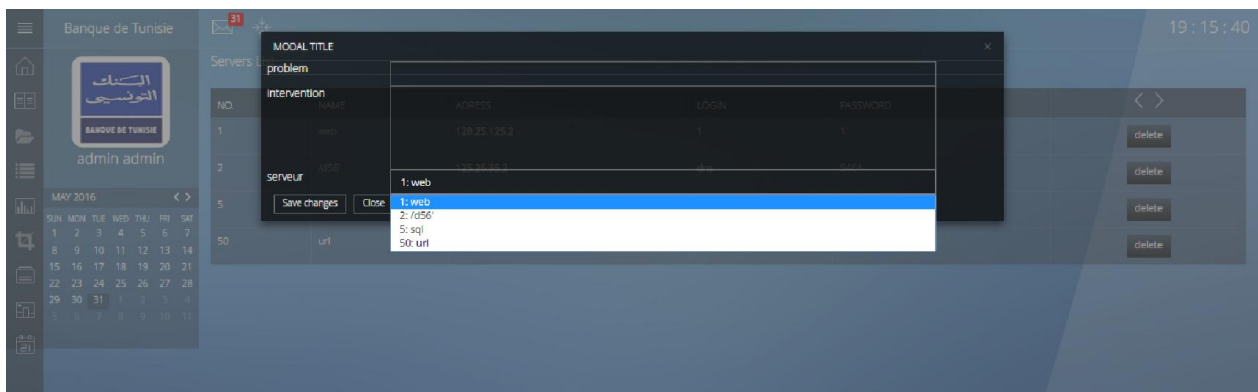


Figure 60 : Intervenir

En cas de dysfonctionnement dans un serveur, le superviseur doit publier une notification contenant le problème rencontré, le nom de serveur en panne et l'intervention qu'il a effectué

V. Conclusion

Dans ce chapitre nous avons détaillé les technologies utilisées pour la réalisation de notre projet ainsi que les fonctionnalités de base de l'application à travers un ensemble de captures d'écran.

Conclusion générale

Comme attendu, le projet de fin d'études a été pour nous, une bonne opportunité afin d'acquérir plus de compétences et faire face à de nouvelles contraintes et enjeux.

En effet, le stage nous a permis de prendre certaines responsabilités, par la suite de consolider nos connaissances théoriques et pratiques. C'est là que réside la valeur d'un tel projet de fin d'études qui combine les exigences de la vie professionnelle aux côtés bénéfiques de l'enseignement pratique que nous avons eu à l'ISET.

Ce travail de conception et de développement d'une application de monitoring au long du stage nous a été bénéfique sur plusieurs plans : il nous a permis de perfectionner nos connaissances acquises en programmation et en conception et de nous intégrer dans la vie professionnelle.

Au début de notre stage, nous avons consacré du temps pour l'étude et le recensement des fonctionnalités de notre application. Et tout au long du développement, nous nous sommes concentrés sur les nouvelles technologies utilisées et les techniques de programmation appliquées.

Tout au long de l'élaboration du projet, nous avons rencontré plusieurs difficultés tant au niveau conceptuel qu'au niveau de la réalisation. Cependant, nous avons essayé de surmonter ces obstacles en redoublant d'effort et en comptant sur la disponibilité et l'aide vailleureux de notre encadreur.

Comme perspective, nous espérons ajouter quelques fonctionnalités à notre application mobile telle que l'ajout d'un outil de notification en temps réel. En ce qui concerne l'application web, nous espérons l'héberger sur le serveur de la banque afin qu'elle soit accessible de l'extérieur.

Nous espérons que ce travail reflètera l'ampleur de nos efforts et vous procurera la satisfaction.

Nétographie

- [URL 1] <http://www.codeproject.com/Articles/304302/Calling-Asp-Net-Webservice-ASMX-From-an-Android-Ap>
- [URL 2] <http://www.jerriepelser.com/blog/using-google-authenticator-asp-net-identity>
- [URL 3] <https://blogs.msdn.microsoft.com/webdev/2014/02/18/adding-two-factor-authentication-to-an-application-using-asp-net-identity/>
- [URL 4] <http://www.asp.net/mvc/overview/security/aspnet-mvc-5-app-with-sms-and-email-two-factor-authentication>
- [URL 5] <https://coding.abel.nu/2014/11/using-owin-external-login-without-asp-net-identity/>
- [URL 6] <http://stackoverflow.com/questions/1916309/pass-multiple-parameters-to-jquery-ajax-call>
- [URL 7] [https://msdn.microsoft.com/en-us/library/bb398995\(v=vs.90\).aspx](https://msdn.microsoft.com/en-us/library/bb398995(v=vs.90).aspx)
- [URL 8] <http://stackoverflow.com/questions/2448472/what-is-the-difference-between-wcf-and-asmx-web-services>
- [URL 9] <https://msdn.microsoft.com/en-us/library/ms972326.aspx>
- [URL 10] http://www.codedigest.com/Articles/ASPNET/347_Pass_Values_from_CodeBehind_to_JavaScript_and_From_JavaScript_to_CodeBehind_in_AspNet.aspx
- [URL 11] <http://halim.canalblog.com/archives/2005/03/28/403805.html>
- [URL 12] <https://cordova.apache.org/plugins/>
- [URL 13] <http://imikado.developpez.com/tutoriels/androidCordova/ma-premier-application/>
- [URL 14] <http://www.iis.net/learn/extensions/introduction-to-iis-express>
- [URL 15] <http://www.w3schools.com/jquerymobile/>
- [URL 16] <https://openclassrooms.com/courses/un-site-web-dynamique-avec-jquery/le-framework-mobile-jquery-mobile>
- [URL 17] [https://msdn.microsoft.com/fr-fr/library/dd483479\(v=vs.100\).aspx](https://msdn.microsoft.com/fr-fr/library/dd483479(v=vs.100).aspx)
- [URL 18] <https://auth0.com/docs/quickstart/hybrid/cordova/wcf-service>
- [URL 19] <http://www.codeproject.com/Articles/706078/Popup-Editing-Adding-Deleting-Records-with-GridVie>
- [URL 20] <https://www.youtube.com/watch?v=4MrCJ7z1ITQ>
- [URL 21] <http://stackoverflow.com/questions/8977821/iis-hosted-wcf-service-sql-queries-using-windows-authentication>
- [URL 22] <http://stackoverflow.com/questions/25449971/hosting-the-wcf-service-on-the-remote-server-with-the-mysql-database-and-consume>
- [URL 23] <http://community.flexerasoftware.com/Responder/Activation.aspx>

[URL 24] <http://www.mobizel.com/2015/08/developpement-dune-application-mobile-hybride-33/>

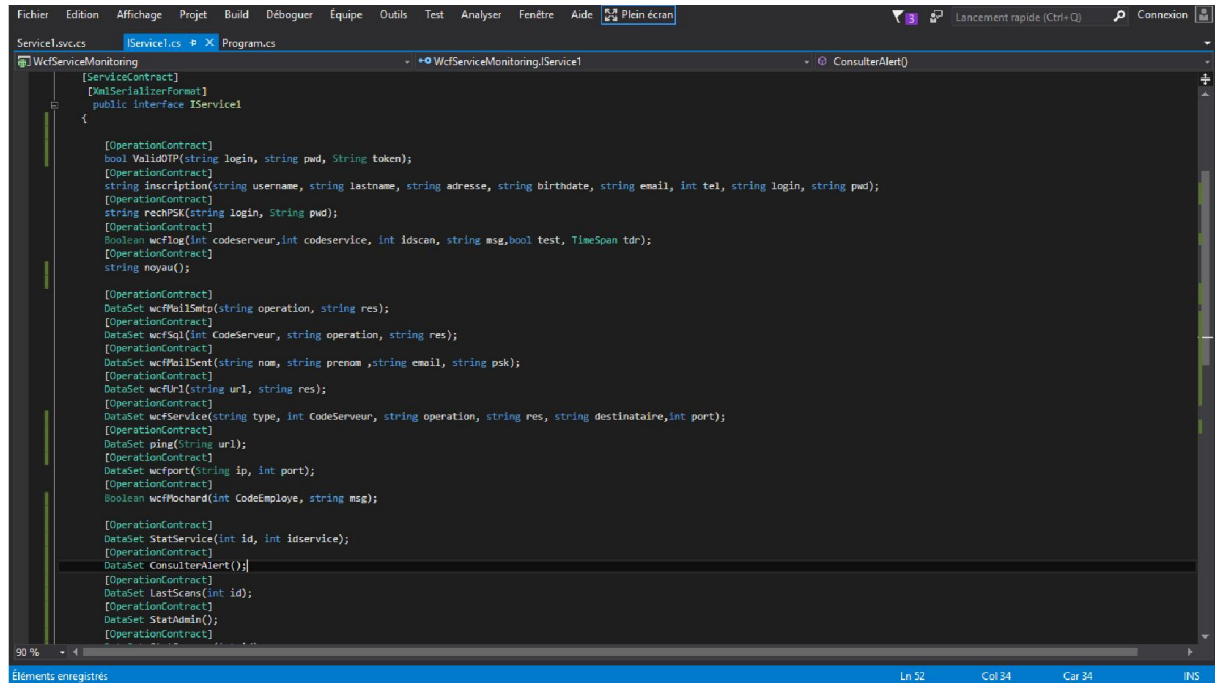
[URL 25] [https://msdn.microsoft.com/fr-fr/library/ms731082\(v=vs.110\).aspx](https://msdn.microsoft.com/fr-fr/library/ms731082(v=vs.110).aspx)

Bibliographies

Nom: Pro ASP.NET Web API Security, Auteur: Badrinarayanan Lakshmiraghavan , maison d'édition: Apress

ANNEXE

WCF Service



```
[ServiceContract]
[XmlSerializerFormat]
public interface IService1
{
    [OperationContract]
    bool ValidOTP(string login, string pwd, string token);
    [OperationContract]
    string inscription(string username, string lastname, string adresse, string birthdate, string email, int tel, string login, string pwd);
    [OperationContract]
    string rechPSK(string login, string pwd);
    [OperationContract]
    Boolean wcflog(int codeserveur, int codeservice, int idscan, string msg, bool test, TimeSpan tdr);
    [OperationContract]
    string noyau();
    [OperationContract]
    DataSet wcfMailSetup(string operation, string res);
    [OperationContract]
    DataSet wcfSql(int codeserveur, string operation, string res);
    [OperationContract]
    DataSet wcfMailSent(string nom, string prenom, string email, string psk);
    [OperationContract]
    DataSet wcfUrl(string url, string res);
    [OperationContract]
    DataSet wcfService(string type, int CodeServeur, string operation, string res, string destinataire, int port);
    [OperationContract]
    DataSet ping(string url);
    [OperationContract]
    DataSet wcfport(string ip, int port);
    [OperationContract]
    Boolean wcfHochard(int CodeEmploye, string msg);
    [OperationContract]
    DataSet StatService(int id, int idservice);
    [OperationContract]
    DataSet ConsulterAlert();
    [OperationContract]
    DataSet LastScans(int id);
    [OperationContract]
    DataSet StatAdmin();
    [OperationContract]
}
```

Figure 61 : Liste des services déployés sur le serveur IIS

Cette page contient toutes les fonctions déployées sur le serveur IIS

Ajout d'un WCF :

1-Ajout de référence URL

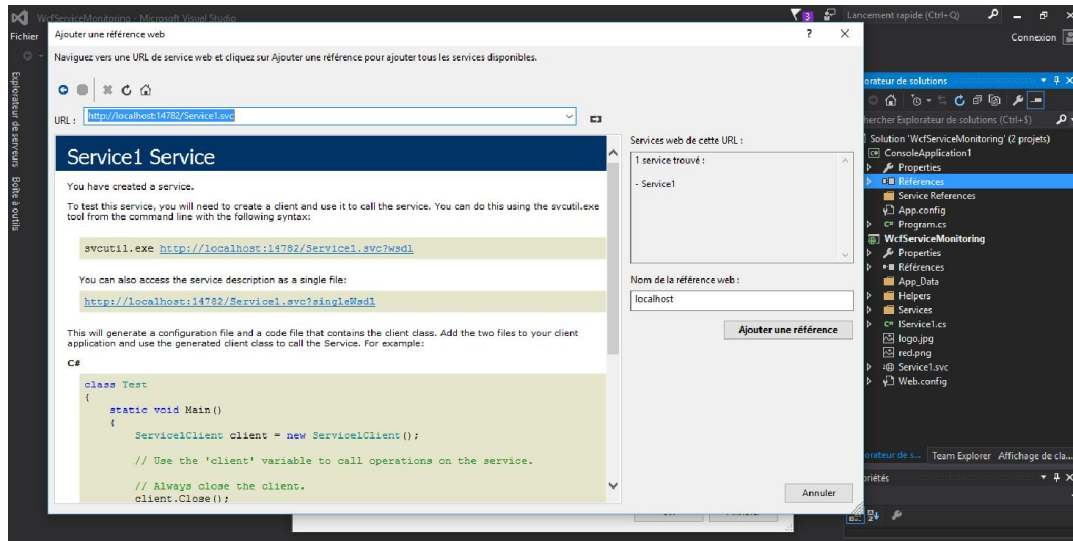


Figure 62 : Ajout de l'URL de Service

2-Nom de référence web

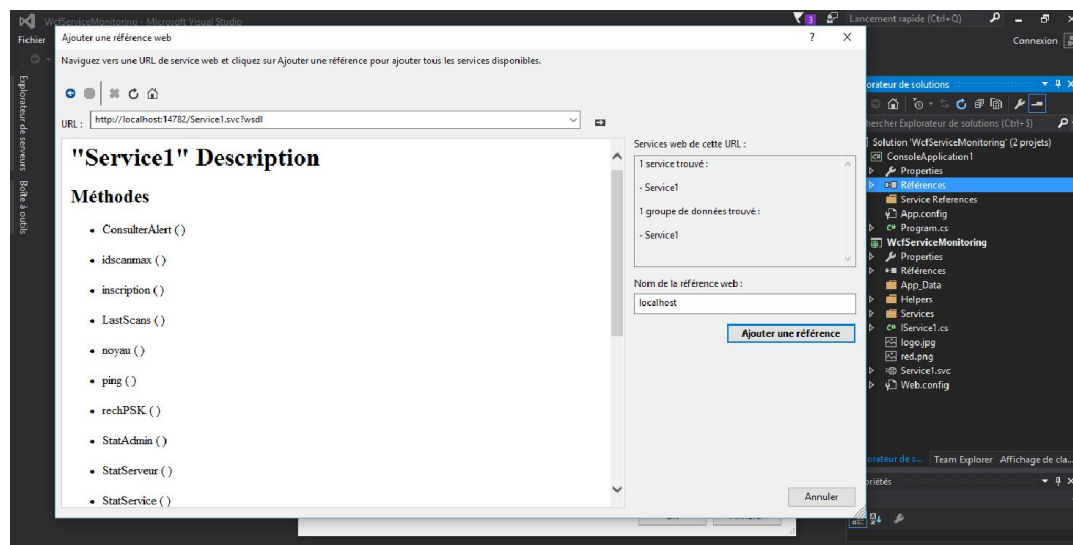


Figure 63 : Liste des services

3- WCF Test Client

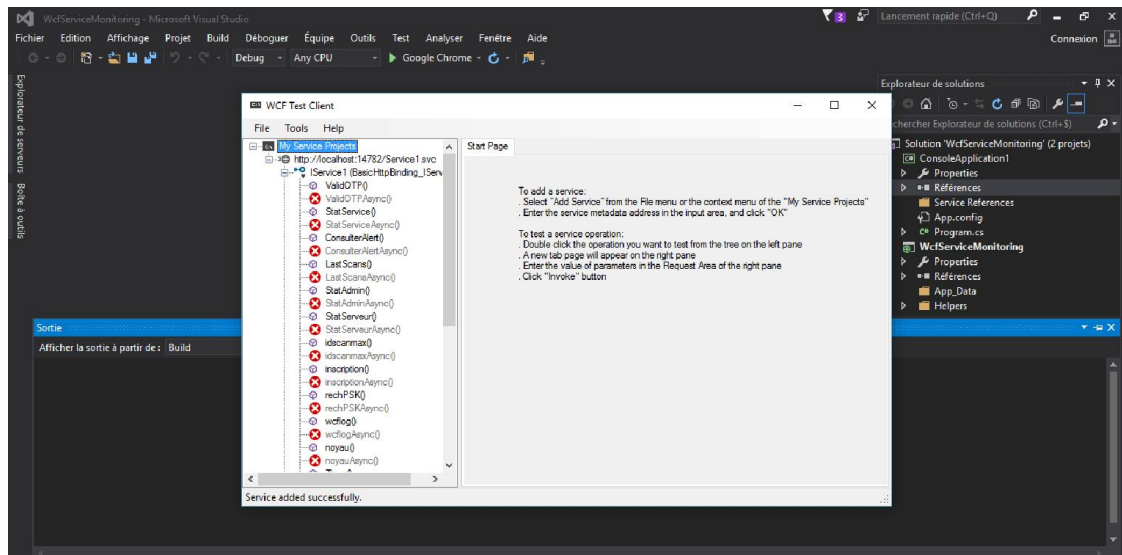


Figure 64 : WCF Test Client

Fonctionnement et Déploiement du noyau de supervision

3. Fonctionnement du noyau

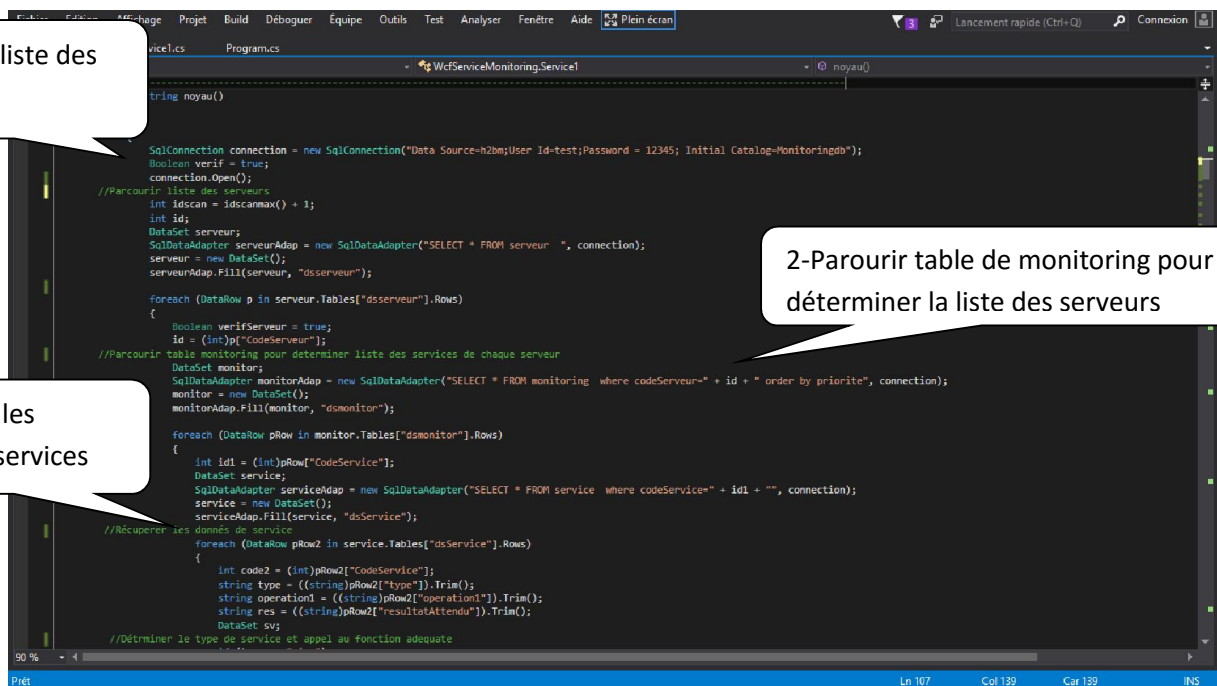


Figure 65 : Fonction Noyau 1

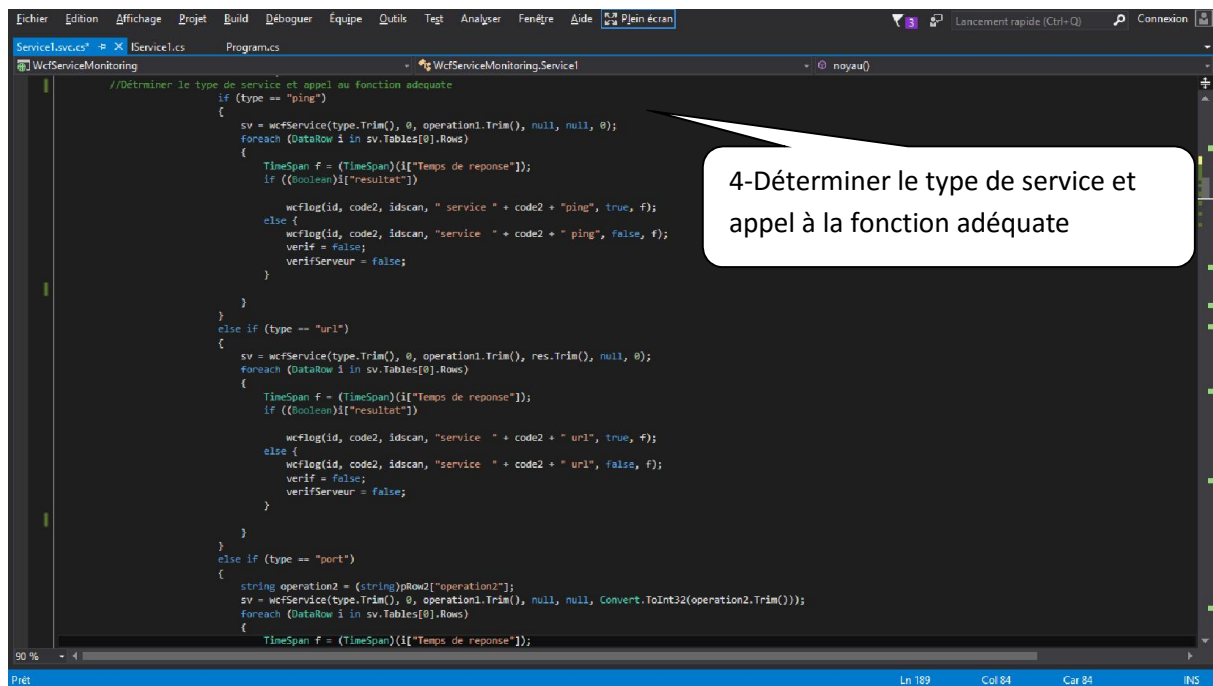


Figure 66 : Fonction Noyau 2