



# Spécification des exigences

Conception d'un logiciel de softphone

Version 1

Martin DELOR

Lauriane LE PAPE

Manon MICHELET

28/10/19

## Historique des révisions

Date	Version	Description	Auteur
28/10/19	Version 1	Première version des spécifications	TriFrogz

## Définitions

Terme	Définition
Téléphonie IP	Téléphonie faite grâce à notre accès Internet plutôt qu'à une ligne terrestre.
Serveur PBX	Autocommutateur téléphonique privé se servant d'Internet pour la gestion des appels téléphoniques d'une entreprise.
Scrum	Framework lié aux méthodes agiles de gestion de projet.

## Abréviations/acronymes

Abré./Acro.	Définition
PC	Personal Computer
SIP	Session Initiation Protocol
IAX	Inter-Asterisk eXchange
PBX	Private Branch eXchange
FAI	Fournisseur d'Accès à Internet

## Table des matières

Historique des révisions	2
Définitions	2
Abréviations/acronymes	2
Table des matières	3
1. Introduction	4
1.1. Objectifs	4
1.2. Portée	4
1.3. Références	4
1.4. Hypothèses et dépendances	4
2. Cas d'utilisation	5
2.1. Diagramme des cas d'utilisation	5
2.2. Cas d'utilisation	5
3. Les acteurs	6
4. Exigences	7
4.1. Fonctionnelles	7
4.2. Non-fonctionnelles	7
4.2.1. Performance	7
4.2.2. Sécurité	7
4.2.3.	7
5. Contraintes de conception	8
6. Composants externes	9
7. Interfaces	10
7.1. Graphiques (GUI)	10
7.2. Matérielles	10
7.3. Communication	10

## 1. Introduction

### 1.1. Objectifs

Ce projet consiste à concevoir un logiciel de softphone afin de pouvoir utiliser le serveur mis en place dans la première partie du projet. Ce logiciel sera conçu à partir de plusieurs langage de programmation tels que Python ou C++ et ouvrira lui-même la communication SIP avec le serveur à l'aide d'un ou plusieurs modules déjà existants. Le but de ce logiciel est de remplir les fonctions demandées dans la partie 1 du projet, c'est-à-dire les boîtes vocales, le renvoi-d'appels et les appels en attente.

### 1.2. Portée

Le produit est un sous-système permettant d'utiliser le serveur installé dans la première partie du projet. Nous devons donc concevoir un logiciel de softphone à l'aide de nos connaissances en programmation mais aussi avec l'aide de modules pré-existant afin de le rendre le plus simple d'utilisation possible. Nous devons donc intégrer toutes fonctions demandées dans la première partie du projet telles que la boîte vocale, la communication avec les télécopieurs et le renvoi d'appel. Notre produit ne gèrera pas la création d'un carnet de numéros ni les appels vidéos.

### 1.3. Références

softphone librairies / software :

<https://www.voip-info.org/open-source-voip-software/>

doc sur le protocole SIP :

[https://fr.wikipedia.org/wiki/Session\\_Initiation\\_Protocol](https://fr.wikipedia.org/wiki/Session_Initiation_Protocol)

<https://www.3cx.fr/blog/guide-protocole-sip/>

Norme ISO 29110

### 1.4. Hypothèses et dépendances

**DEP1** : Le logiciel doit permettre les renvois d'appels, les appels en attentes, les boîtes vocales ainsi que la récupération des numéros actuellement existant de l'entreprise.

**DEP2** : Le logiciel doit pouvoir communiquer avec les télécopieurs.

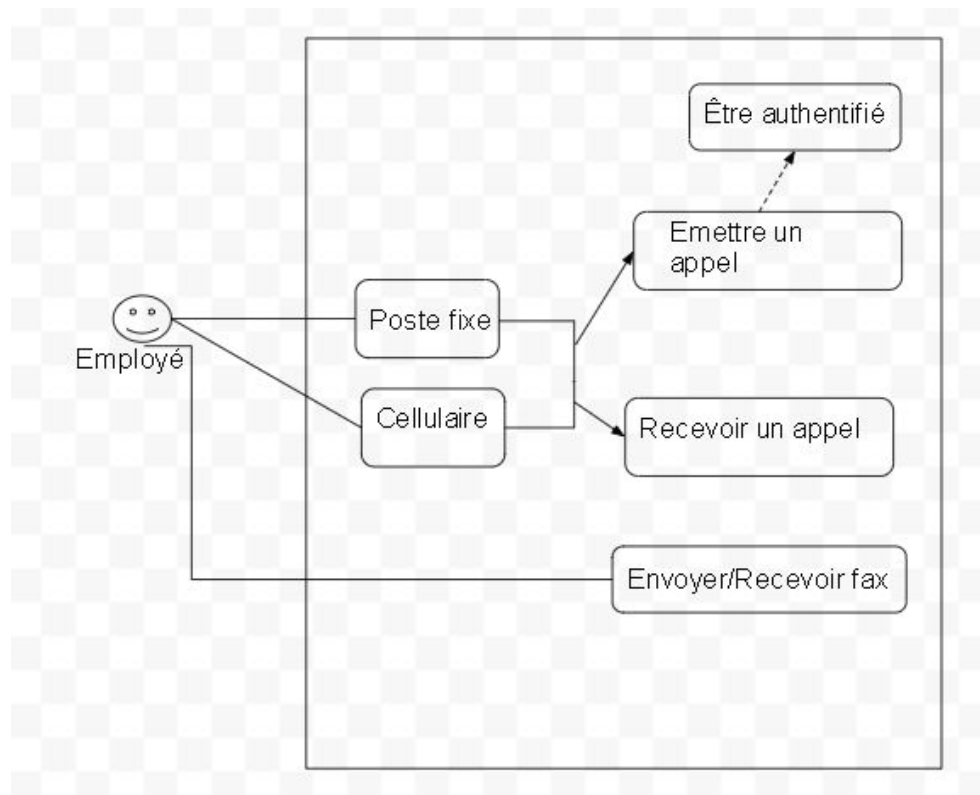
**HYP1** : Nous considérons que les ordinateurs de l'entreprise possède au moins 8Go de RAM afin de faire fonctionner à la fois le serveur, le logiciel et d'autres applications utiles à l'employé.

**HYP2** : Nous faisons l'hypothèse qu'il restera assez de place sur tous les disques durs pour l'installation d'éventuelles mise à jour de Windows, des drivers requis, du logiciel FreeSwitch et du logiciel de softphone conçu.

**HYP3** : Le matériel fourni au cours de la première partie du projet est encore disponible.

## 2. Cas d'utilisation

### 2.1. Diagramme des cas d'utilisation



## 2.2. Cas d'utilisation

**CU1** : Émettre un appel -> Le client doit pouvoir émettre un appel vers un autre membre du réseau IP

**CU2** : Recevoir un appel -> Le client doit pouvoir recevoir un appel provenant d'un autre membre du réseau IP

**CU3** : Envoyer/Recevoir fax -> L'utilisateur est capable d'envoyer/recevoir des documents via le réseau TCP/IP.

## 3. Les acteurs

Dans un premier temps, il y a l'utilisateur, soit la société ACME. Dans un premier temps son rôle sera de nous fournir des indications concernant les besoins primaires que doit combler le logiciel. Au fur et à mesure du projet et du rendu des livrables, le client devra aussi nous faire part de ses suggestions de changements ou des éventuels changements de besoins qui pourraient survenir au cours du projet.

Ensuite, il y a le SCRUM Master appartenant à l'entreprise TriFrogz qui a pour tâche de mener à bien votre projet en réalisant une bonne gestion des ressources humaines, temporelles et matérielles. Pour ce faire, il établira une communication interne entre les employés concernés par le projet, ainsi qu'une communication externe avec votre entreprise. De ce fait, il se chargera de faire passer l'information d'une personne à une autre, permettant donc de faire progresser le projet en prenant en compte l'avis de toutes les parties sur la prise d'une décision de conception.

Les membres de l'équipe ont été choisis en fonction de leurs compétences dans le domaine de la téléphonie IP afin de vous offrir le meilleur services possibles. Leur capacité à travailler en équipe ainsi que leur expérience leur permettra de vous fournir un travail de qualité. Ces membres seront les mêmes que ceux ayant déjà travaillé sur la première partie du projet.

## 4. Exigences

### 4.1. Fonctionnelles

#### Exigences requises par le client :

- Communication bidirectionnelles avec les télécopieurs
- Possibilités de renvois d'appels, de la mise en place d'une boîte vocale, de mise des appels en attente
- Utilisation des numéros téléphoniques actuellement utilisés par l'entreprise

## 4.2. Non-fonctionnelles

### Exigences implicites :

- Les communications doivent être sécurisées entre les acteurs des appels
- Le système ne doit pas surcharger le réseau internet de l'entreprise
- Le logiciel doit permettre l'utilisation d'autres logiciels dans le même temps d'utilisation

### 4.2.1. Performance

La communication entre les acteurs d'un appel en téléphonie IP doit être immédiate, sans délais et ne pas contenir de brouillage qui serait causé par le réseau de l'entreprise en raison d'une surcharge réseau. Le logiciel sera adaptée aux ordinateurs afin de ne pas dépasser les capacités internes de ceux-ci et permettre l'ouverture d'autres logiciels en parallèle.

### 4.2.2. Sécurité

La communication par téléphonie IP installée doit être sécurisée dans la mesure des configurations possibles du logiciel afin de permettre une communication sécurisée entre les acteurs d'un appel.

Nous ne serons en aucun cas responsable des éventuelles vulnérabilités du logiciel si elles sont issues d'une utilisation malveillante ou détournée de notre produit.

## 5. Contraintes de conception

Le logiciel doit pouvoir fonctionner sur Windows 10. Elle doit aussi permettre de récupérer les numéros de téléphones actuels de l'entreprise ainsi que garder la communication possible avec les télécopieurs.

## 6. Composants externes

Pour l'installation du logiciel, aucun composant externe n'est requis.

## 7. Interfaces

### 7.1. Graphiques (GUI)

L'interface minimale sera composée d'une liste des contacts, d'une zone de saisie de numéro ainsi que des boutons permettant de démarrer/stopper l'appel.

### 7.2. Matériel

Le matériel nécessaire est le même que pour la première partie du projet :

- Un **routeur** dans l'entreprise permettant de créer le réseau. Internet est nécessaire dans le cas des appels externes.
- Pour la bonne mise en place du système, il est nécessaire au minimum d'avoir 3 **ordinateurs** Windows 10, dont 1 est le serveur, les 2 autres les postes de communication. Tous devront être reliés au réseau Internet de l'entreprise.
- Pareil pour les **télécopieurs**, qui seront intégrés au système VOIP via une **passerelle VOIP** ou un **ATA** (Analog Telephony Adapter).
- De plus, il faut au minimum 2 **casques d'écoute** avec micro pour les appels.

### 7.3. Communication

Afin d'ouvrir la communication via le logiciel, nous utiliserons le protocole SIP. Le SIP (Session Initiation Protocol) est un protocole de signalement utilisé pour établir une "session" entre deux ou plus de participants, modifier cette session, et finalement terminer cette session.