Etudiant B

BOURBON Kévin

BTS2

Dossier de conception détaillée/réalisation

**Projet 1 :**

**Consignes et assistance vidéolink :assistance téléphonique**

BTS IRIS 2ème année

Immaculée Conception – LAVAL

Sommaire

[I – Le poste IPBX 3](#_Toc421601632)

[I.A – Installation du système 3](#_Toc421601633)

[I.B – Configuration d’Asterisk. 4](#_Toc421601634)

[*I.B.1 – Paramètres généraux* 4](#_Toc421601635)

[*I.B.2 – Configuration des postes téléphoniques sous Asterisk / FreePBX.* 5](#_Toc421601636)

[I.C – Problèmes rencontrés avec le poste IPBX 5](#_Toc421601637)

[II – Le softphone Linphone 6](#_Toc421601638)

[II.A – Installation de Linphone 6](#_Toc421601639)

[II.B – Configuration de Linphone 6](#_Toc421601640)

[III – Messagerie et enregistrement des communications. 7](#_Toc421601641)

[III.A – Configuration de la messagerie. 7](#_Toc421601642)

[*III.A.1- Configuration des paramètres généraux de la messagerie* 7](#_Toc421601643)

[*III.A.2- Modification de la langue par défaut de la boîte vocale* 8](#_Toc421601644)

[III.B – Enregistrement des communications 9](#_Toc421601645)

[III.C – Consultation des enregistrements 10](#_Toc421601646)

[III.D – Problèmes rencontrés avec Linphone 11](#_Toc421601647)

[IV – Module de l’application LC pilotant Linphonec 11](#_Toc421601648)

[IV.A – Développement des classes nécessaires au pilotage de Linphonec 11](#_Toc421601649)

[IV.B – Accès à distance aux consignes par la connexion SSH 15](#_Toc421601650)

[IV.C – Problèmes et difficultés rencontrées pendant le développement 15](#_Toc421601651)

# I – Le poste IPBX

## I.A – Installation du système

Le poste IPBX est un autocommutateur téléphonique privé, il sert à relier entre eux les postes téléphoniques. C’est le logiciel de simulation appelé Asterisk, leader dans le domaine qui a été choisi pour ce projet. Il possède de très nombreuses fonctionnalités pour réaliser une assistance téléphonique très performante, comme l’enregistrement de messages.



Logo d’Asterisk

La configuration d’Asterisk est complexe puisqu’elle s’effectue en ligne de commande. Un outil a été conçu pour faciliter considérablement la partie configuration du système, il s’agit de FreePBX. La documentation Internet, très fournie, explique dans les moindre détails toutes les démarches à suivre pour installer et configurer dans les meilleures conditions le serveur téléphonique que forme dans notre projet le poste IPBX. Site web : <http://wiki.freepbx.org/dashboard.action>



Logo de FreePBX

Il existe deux possibilités pour créer son poste IPBX. La première est d’installer le système entier sur un disque dur. Le fichier .iso est téléchargeable et l’installation peut commencer. Le système est régi par un système d’exploitation de type CentOS. L’installation prend environ une heure. Une autre possibilité est d’installer Asterisk en tant que logiciel sur un ordinateur. L’installation est toutefois beaucoup plus complexe, la première solution est à retenir dans notre cas.



Aperçu d’une copie d’écran au démarrage du poste IPBX.

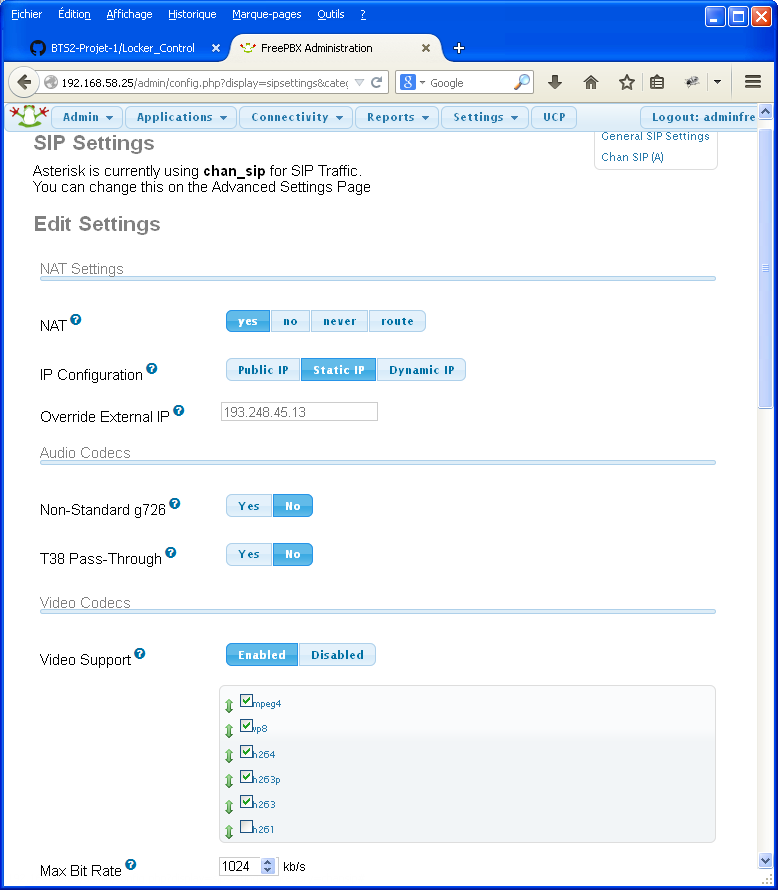
Un nom d’utilisateur et un mot de passe est demandé pour accéder en mode root au système.

## I.B – Configuration d’Asterisk.

### *I.B.1 – Paramètres généraux*

La première chose à faire est de se connecter en entrant l’adresse IP que le système à l’installation a détecté et de la changer pour la rendre statique, puisque c’est un serveur. Ceci se fait en accédant au système par l’interface web. Il suffit d’entrer l’adresse IP correspondante dans la barre d’adresse et l’accès est possible immédiatement après avoir entré son login et son mot de passe. La page qui s’ouvre par défaut est le statut du système. Il y a notamment un graphique qui répertorie les postes téléphoniques en ligne ou hors-ligne. Ce graphique peut aider, pour plus tard à vérifier que les softphones sont bien connectés au poste IPBX.

Dans l’onglet « Settings » > « Advanced SIP Settings », on active la gestion vidéo (on met le champ « Video Support » à « Enabled »). On coche les cases correspondants aux formats d’encodage vidéo tels que h263, h263p et h264 voire mpeg4.



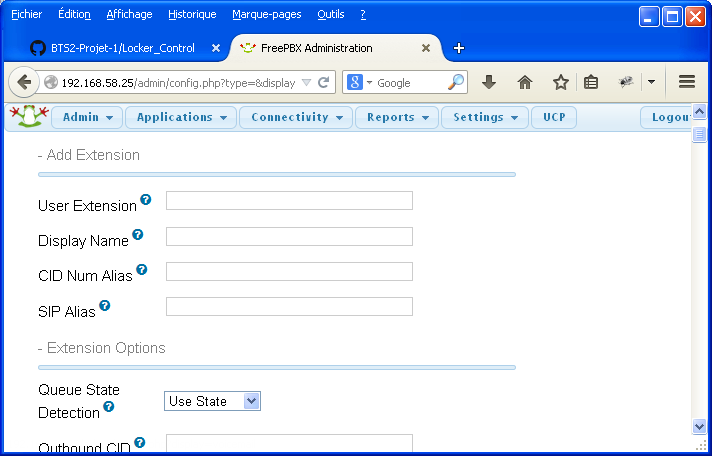
Extrait de fenêtre pour activer la vidéo

### *I.B.2 – Configuration des postes téléphoniques sous Asterisk / FreePBX.*

Le mot anglais correspondant aux postes téléphoniques à configurer est « extension ». Ce qui correspond au mot « poste », mais on trouve également dans de la documentation française le mot « extension ». L’interface proposée permet de créer le numéro affecté au poste téléphonique, par exemple 600. Comme le numéro attribué est communément entre 100 et 900, j’ai choisi le nombre 600.

Un mot de passe est généré de manière automatique et aléatoire. Ce mot de passe sera très utile lors de la configuration de Linphone, nous le verrions par la suite.

Pour configurer un poste téléphonique, il faut cliquer sur l’onglet : « Applications » > « Extensions » et « Add an extension ». Là un numéro est demandé, c’est celui qui sera utilisé pour appeler un poste ainsi que le nom que l’on va lui attribuer. D’autres paramètres seront à modifier sur cette page, ce qui sera développé ultérieurement.



Fenêtre pour ajouter une nouvelle extension (copie d’écran).

## I.C – Problèmes rencontrés avec le poste IPBX

J’avais au tout départ en tête qu’il fallait installer Asterisk sur un poste avec OpenSuSE d’installé dessus. Ce qui est tout à fait possible. Mais il s’agit de la méthode la plus compliquée et même les tutoriels les plus efficaces ne peuvent remédier à la difficulté. Mes professeurs ont corrigé cette erreur d’interprétation et j’ai opté pour la deuxième méthode, telle que je l’ai expliqué précédemment.

# II – Le softphone Linphone

Linphone est un logiciel libre de téléphonie par Internet multiplateforme. Il utilise le protocole SIP et supporte l’intégration vidéo dans les communications.



Logo de Linphone

## II.A – Installation de Linphone

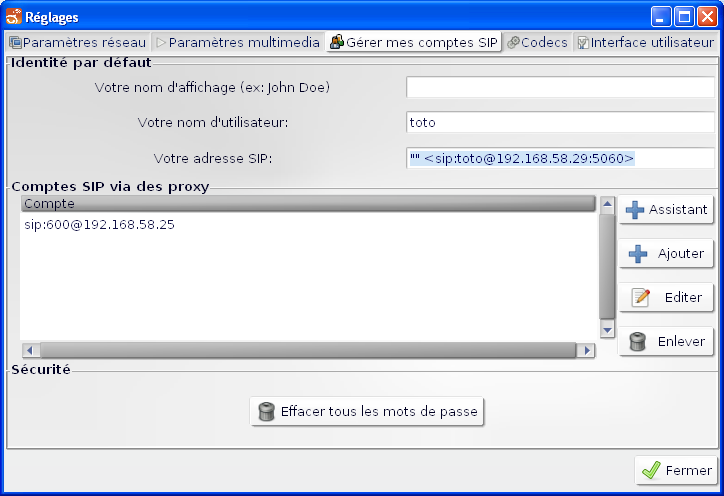
La version installée sur le système est la 3.8.0. Pour plus de facilité, la version est disponible en téléchargement avec YAST2, l’utilitaire de dépôts d’OpenSuSE. Mais attention, pour une meilleure stabilité du logiciel, il est nécessaire auparavant de télécharger le dépôt communautaire Packman. Packman donne à disposition des utilisateurs un ensemble de logiciels que l’on peut installer facilement. Le but est également de rendre compatibles les logiciels qui ne l’étaient pas initialement. Les dépôts Packman qui nous intéressent en particulier sont Essentials qui délivre des formats d’encodage et des logiciels multimédia et Multimedia dans lequel on retrouvera Linphone. Le tutoriel est disponible à cette adresse : <https://fr.opensuse.org/D%C3%A9p%C3%B4ts_de_paquets_suppl%C3%A9mentaires> .

Une fois Packman installé sur le système, Linphone peut être installé. Toutefois cette version présente des problèmes de stabilité et il est nécessaire de vérifier des éléments de la bibliothèque sur laquelle s’appuie Linphone. En particulier : vérifier que la librairie libortp.so.9 est bien celle de Packman et non d’OpenSuSE. De même pour libellesip, la version doit être celle de Packman et la plus récente. Linphone est désormais installé et est prêt à être configuré.

## II.B – Configuration de Linphone

Une fois Linphone lancé, nous allons connecter le softphone au compte créé sous Asterisk / FreePBX.

Dans « Options » > « Préférences », on clique sur l’onglet « Gérer mes comptes SIP », et sur le bouton « Ajouter ». Linphone demande l’identité SIP. La donnée à inscrire doit être sous la forme n°extension@adresseIP\_IPBX. Ensuite, on inscrit l’adresse du proxy SIP : sip : adresseIP\_IPBX. On valide. Linphone demande un mot de passe. Ce mot de passe correspond à la série de numéros et de lettres du champ « secret ». Un fois la donnée rentrée et validée, Linphone doit indiquer le message suivant : Enregistrement sur sip : adresseIP\_IPBX effectué. Les opérations citées précédemment seront à effectuer pour chaque Linphone à enregistrer.



Il s’agit de vérifier qu’au moins les formats d’encodage vidéo h263, h263-1998 et MP4 sont activés pour être assurés de la communication vidéo. Si un format d’encodage ne fonctionne pas ou ne répond pas à temps, Asterisk fait en sorte de basculer sur un autre et ainsi de suite. En ce qui concerne h263-1988, il s’agit du même format d’encodage que h263p. En effet h263p, aussi noté h263+, a été développé et publié en 1998…

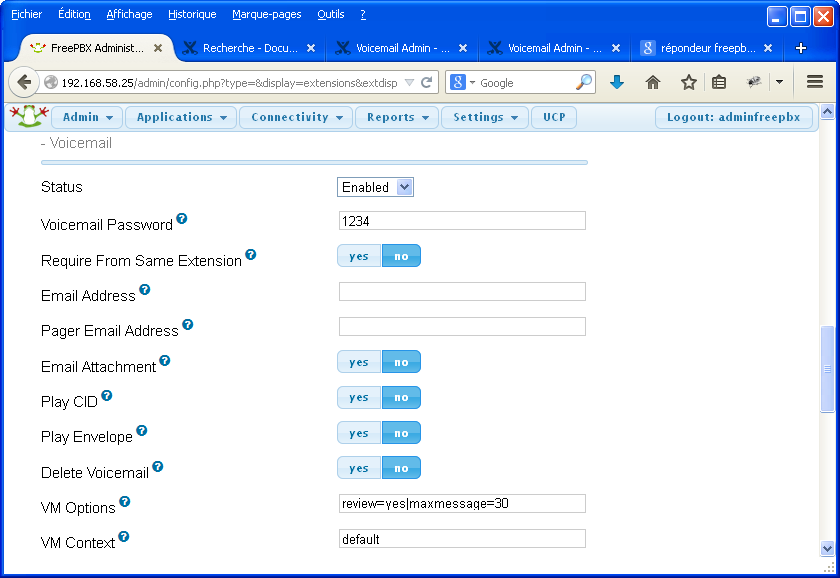
# III – Messagerie et enregistrement des communications.

Le projet stipule de pouvoir laisser un message sur le répondeur de l’assistance. Dans cette partie nous verrons comment configurer la messagerie disponible par Asterisk / FreePBX. De même, on nous demande d’enregistrer la communication. Nous verrons quels sont les outils à notre disposition.

## III.A – Configuration de la messagerie.

### *III.A.1- Configuration des paramètres généraux de la messagerie*

Asterisk possède une messagerie. La messagerie peut être activée sur la page des « Extensions ». La page possède plusieurs catégories de configuration dont l’une est appelée « Voicemail ».



Aperçu de la fenêtre de configuration de la messagerie vocale

Le champ « Status » doit être passé à « Enabled » pour que la messagerie soit activée pour le poste téléphonique sélectionné. Un mot de passe est demandé. Celui-ci est utile si le poste est utilisé par plusieurs personnes et augmente la sécurité, pour plus de simplicité ici, nous laissons de côté cette configuration.

Dans le champ « VM Options » entrer la ligne suivante : review=yes|messagemax=30. Cette ligne indique à la messagerie qu’il est possible de réentendre et de modifier le message enregistré, et que la durée maximale du message est de 30 secondes, conformément à se qui est demandé dans le projet. On clique sur le bouton « Submit » en bas de la page et ensuite sur « Apply Config » pour enregistrer et appliquer les modifications.

En ce qui concerne le numéro à appeler depuis le Linphone pour accéder à la messagerie, il s’agit de \*97. Des instructions sont données au fur et à mesure pour indiquer les options disponibles. Des options avancées, de l’aide ou encore la configuration de la boîte vocale et enregistrement font partie des fonctionnalités disponibles avec le répondeur proposé par Asterisk.

### *III.A.2- Modification de la langue par défaut de la boîte vocale*

Par défaut, la messagerie est en anglais. Il est possible toutefois d’activer la langue française et de revenir à la configuration de départ. Le site <http://www.tux89.com/telephonie/configuration-pour-utiliser-asterisk-en-francais/> propose un tutoriel pour télécharger les fichiers et les installer pour avoir la langue française. Dans Asterisk et la configuration des postes, il faut indiquer dans la zone « Languages » « fr ». De même que dans l’onglet « Applications » > « Languages », indiquer « fr » dans le champ désigné.

## III.B – Enregistrement des communications

 Asterisk offre la possibilité d’enregistrer les communications. La configuration s’effectue d’une part dans « Applications » > « Call Recording ».

Dans cette fenêtre, on nous demande la description des enregistrements. On inscrit l’indication voulue. On met le champ « Call Recording Mode » à « Force » pour enregistrer systématiquement toutes les communications. Enfin, pour la « Destination », on choisit « Extensions » et le poste téléphonique désiré et plus particulièrement celui qui sera apte à pouvoir réécouter les enregistrements des communications, à savoir le poste AEL.

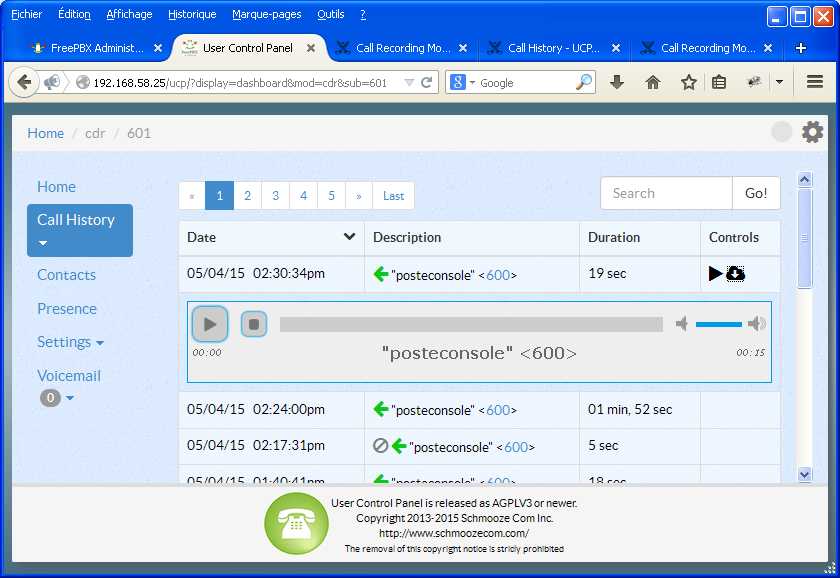
Autre point important, c’est la sélection des enregistrements en fonction des cas. C’est-à-dire qu’il est possible de lancer automatiquement un enregistrement si l’appel vient d’un poste du réseau téléphonique que l’on a créé par exemple. Cette configuration se trouve de nouveau dans la page « Extensions », partie « Call Recording ». On met à « Force » les paramètres « Inbound Internal Call » (appel entre postes téléphoniques internes) et « Outbound Internal Call (appel de l’intérieur vers l’extérieur), ce dernier peut être utile pour enregistrer le répondeur lorsqu’un appel est fait avec le numéro \*97.

Sur la même page, on va dans l’ensemble « Device Options » et le champ « DTMF Signaling ». Le DTMF (dual-tone multi-frequency) est une combinaison de fréquences pour les serveurs vocaux. Lorsqu’une touche est appuyée, le téléphone émet un signal d’une certaine fréquence qui est reconnue par le serveur vocal et il l’attribue à un nombre et il agit en conséquence. Linphone supporte les protocoles RFC 2833 et SIP INFO, toutefois à l‘activation de ces derniers, il s’est révélé que le serveur vocal ne reconnaissait pas les touches, probablement dans un souci de compatibilité de protocoles qui s’y réfèrent. Il faut donc sélectionner « In band audio », même si cela est indiqué comme non recommandé. Répéter l’opération pour chaque poste.

## III.C – Consultation des enregistrements

Il est possible de consulter la liste d’appels téléphoniques passés. Asterisk enregistre toutes les données qui sont affichées dans l’onglet UCP. L’User Control Panel (littéralement « panneau de contrôle de l’utilisateur ») est un outil proposé par Asterisk /FreePBX et qui répertorie entre-autres l’historique des appels en fonction des postes téléphoniques, y compris les messages laissés sur le répondeur.

Pour accéder à ces fonctionnalités, il est nécessaire de créer un nouvel utilisateur. Ceci est accessible dans « Admin » > « User management ». Dans le champ « Linked Extension », assigner au moins un poste, mais il est préférable d’intégrer tous les postes téléphoniques.

Après avoir cliqué sur l’onglet « UCP » et renseigné le login et le mot de passe, la fenêtre apparaît de manière semblable à celle-ci :

Copie d’écran de l’apparence de l’UCP avec la rubrique « call history » sélectionné.

Voici la légende des icônes possibles dans la colonne « Description » :

Screen Shot 2014-06-19 at 1.24.28 PM.png : Appel extérieur

http://wiki.freepbx.org/download/attachments/28180543/Screen%20Shot%202014-06-19%20at%201.24.33%20PM.png?version=1&modificationDate=1403202727000&api=v2 : Appel intérieur

http://wiki.freepbx.org/download/attachments/28180543/Screen%20Shot%202014-06-19%20at%201.25.29%20PM.png?version=1&modificationDate=1403202727000&api=v2 : Appel sur le répondeur

http://wiki.freepbx.org/download/attachments/28180543/Screen%20Shot%202014-06-19%20at%201.25.34%20PM.png?version=1&modificationDate=1403202727000&api=v2 : Appel vers une salle de conférence (que l’on ne rencontrera pas dans notre cas)

De même, la colonne « Controls » indique la possibilité d’une lecture de l’enregistrement directement sur le navigateur, ou par le téléchargement du fichier.

http://wiki.freepbx.org/download/attachments/28180567/Screen%20Shot%202014-06-19%20at%202.52.02%20PM.png?version=1&modificationDate=1403207541000&api=v2 : Lecture sur le navigateur de la communication téléphonique enregistrée.

http://wiki.freepbx.org/download/attachments/28180567/Screen%20Shot%202014-06-19%20at%202.52.07%20PM.png?version=1&modificationDate=1403207594000&api=v2 : Téléchargement sur l’ordinateur du fichier correspondant à l’enregistrement.

## III.D – Problèmes rencontrés avec Linphone

Le poste sur lequel je configurais Linphone n’avait pas le son de configuré. J’ai donc du configurer le son mais mon inexpérience avec OpenSuSE m’a retardé dans ma démarche. Ensuite, j’ignorais comment connecter le Linphone au poste IPBX, et en particulier sur le mot de passe qui est demandé avant de valider. Sur Asterisk / FreePBX, le nombre de champs de configuration pour les extensions est tel que je n’ai pas vu ce qui m’intéressait. De plus il était indiqué devant « Secret » j’ai trouvé le mot peu explicite.

La version de Linphone que je disposais au départ n’intégrait pas la vidéo. J’ai changé de version pour opter vers une version ultérieure, mais le problème était le même. En clair, j’avais dans ma fenêtre vidéo, pendant une communication un écran noir. Il y avait en réalité une incompatibilité au niveau des librairies qu’utilise Linphone. Cette incompatibilité entre les librairies natives d’OpenSuSE et les librairies Packman a été très difficile à déceler. La façon de régler le problème était difficile à effectuer.

# IV – Module de l’application LC pilotant Linphonec

## IV.A – Développement des classes nécessaires au pilotage de Linphonec

L’utilisation de Qt rend plus simple le développement de programmes qui en conception préliminaire paraissent complexes et notamment la notion sur l’utilisation des tubes.

Voici quelques éléments importants dans la définition de la classe :

Dans la méthode run() :

* J’utilise une boucle événementielle, qui effectue une lecture en boucle dans un tableau. Ce tableau récupère la sortie standard et est stockée dans ce tableau de type QString. Cette boucle événementielle est reliée au bouton « raccrocher » de l’interface de l’assistance. Ce qui suit, après la boucle while() sera exécuté quand l’utilisateur va cliquer sur le bouton raccrocher.
* J’utilise la fonction write() pour écrire la commande qui va terminer l’appel.
* La fonction waitForBytesWritten() empêche momentanément, ici une seconde, d’envoyer la commande qui suit.

|  |
| --- |
| void Thread\_linphonec::run() |
| { |
| QString readout; |
| int result; |
| while(m\_loopevent) //Lecture d’un tableau qui reçoit la sortie du processus Linphonec |
| { |
| m\_linphonec->waitForReadyRead(1000); |
| m\_ByteArray = m\_linphonec->readAll(); |
| readout.append(m\_ByteArray); |
| qDebug()<<readout; |
|  |
| }  //Envoi de la commande qui permet de raccrocher et quitter au bout d’un temps de 1s |
|  |
| m\_linphonec->write("terminate\n"); |
|  |
| m\_linphonec->waitForBytesWritten(1000); |
| m\_linphonec->write("quit\n"); |
|  |
| result = m\_linphonec->exitCode(); |
| if(result != 0) |
| { |
| qDebug()<<"error exit process"; |
| } |
| } |

|  |
| --- |
| void Thread\_linphonec::EndThread() |
| { |
| m\_loopevent = false; |
|  |
| } |

Dans la méthode startThread() :

La méthode permet de lancer Linphonec en mode vidéo an ajoutant comme argument dans la fonction start() « -V », stocké dans un QStringList, une liste chaînée. J’envoie la commande de l’appel de l’assistance en ligne qui possède le numéro 601, comme configuré auparavant. Je gère les erreurs au cas où le processus Linphonec retournerait une erreur.

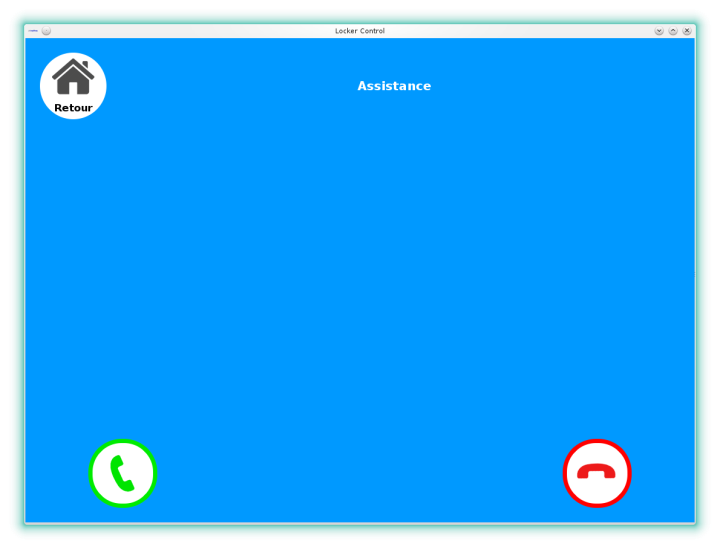
|  |
| --- |
| void Thread\_linphonec::startThread() |
| { |
| //Stockage dans une variable de l’argument à entrer pour ouvrir Linphonec en mode vidéo |
|  |
| QStringList arg; |
| arg << "-V"; |
|  |
| //Démarrage du programme Linphonec en mode vidéo |
| m\_linphonec->start("/usr/bin/./linphonec", arg);  //Vérification que le processus Linphonec est correctement lancé |
|  |
| if (!m\_linphonec->waitForStarted()) |
| qDebug()<<"help";  //Envoi de la commande permettant d’appeler l’assistance en ligne avec gestion d’erreur |
|  |
| if(m\_linphonec->write("call 601\n")==-1) |
| { |
| qDebug()<<"error ecriture"; |
| }  start() ; //démarrage du thread qui va lancer Linphonec |
|  |
|  |
| } |
|  |

Un élément est très important dans Qt : la notion de slot. Un slot est une connexion entre un élément de l’interface, par exemple typiquement un bouton et une action à exécuter (une méthode). J’utilise ce slot pour exécuter d’une part le lancement de Linphonec en cliquant sur le bouton de l’assistance en ligne (cf. image ci-dessous).

Autre précision : l’utilisation de thread. En conception préliminaire, j’avais évoqué la création d’un processus (Linphonec). Cela reste vrai mais la documentation de Qt précise une chose : « pour toutes les fonctions WaitFor, appeler cette fonction depuis le thread principal (GUI) peut faire geler votre interface ». Le thread est utilisé pour éviter tout blocage. Ne pas le faire signifierait potentiellement que l’interface utilisateur de l’assistance en ligne serait « gelée », l’application serait bloquée.



La copie d’écran ci-dessus est la fenêtre affichée par défaut lorsqu’un client vient retirer un colis. Il peut cliquer en haut à droite sur le bouton symbolisant l’assistance en ligne.



Ci-dessus l’utilisateur peut démarrer l’appel à l’assistance en ligne en appuyant sur le bouton vert. C’est à ce moment que le processus Linphonec démarre et que la communication avec ce dernier peut commencer. Normalement au centre apparait la webcam locale à la console mais les contraintes de temps nous on empêché de développer cet aspect du module.

## IV.B – Accès à distance aux consignes par la connexion SSH

Les contraintes de temps m’ont empêché de développer cet aspect du système.

## IV.C – Problèmes et difficultés rencontrées pendant le développement

La prise en main de Qt a été longue, de même que déterminer comment coder en fonction des attentes du logiciel à développer. Le catalogue de fonctions est tel qu’il est difficile de s’y retrouver. Malgré tout la documentation de Qt est précise et permet de mieux s’y repérer. La partie concernant le module Linphonec rest encore fragile, mais du temps supplémentaire aurait permis de terminer correctement.