|  |  |  |  |
| --- | --- | --- | --- |
| Auteur | Date | Version | Commentaires |
| Groupe | 26/01/2016 | V1.0 | Présentation du sujet + Conception de la phase 1 |
|  |  |  |  |
|  |  |  |  |

# Cahier des Charges

Le client nous a demandé de concevoir et réaliser une application d’annuaire d’étudiants.

Un étudiants renseignera les informations suivantes :

* Nom
* Prénom
* Année de diplomation
* Adresse e-mail
* Numéro de Téléphone

Un étudiant doit pouvoir s’inscrire, consulter la liste des étudiants, rechercher un étudiant et consulter la fiche d’un étudiant.

Il peut également modifier les informations de sa fiche et choisir de les partager avec les visiteurs.

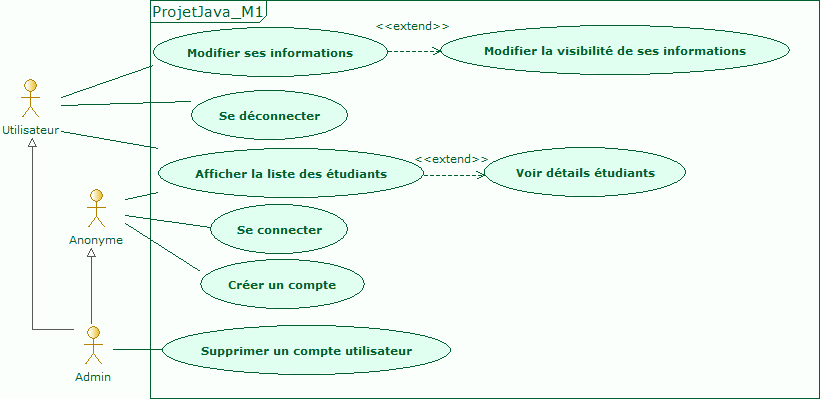
# Architecture

Nous décidons ici de mettre en place une architecture client-serveur.

# Analyse Conceptuelle

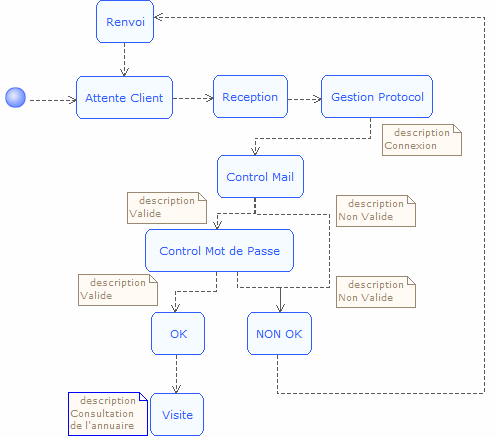
## Diagramme de cas d’utilisation

Il existe trois « Utilisateurs » qui possèdent des droits et des choix différents. Le diagramme de cas d’utilisation permet d’expliciter ces différences.

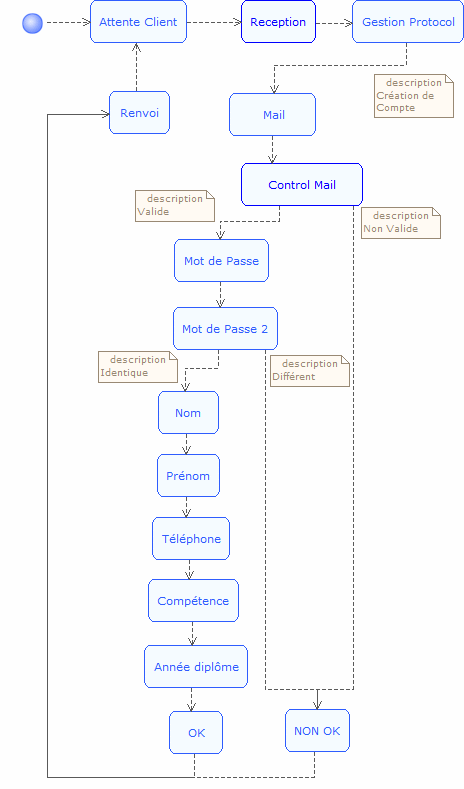


## Diagrammes d’activités.

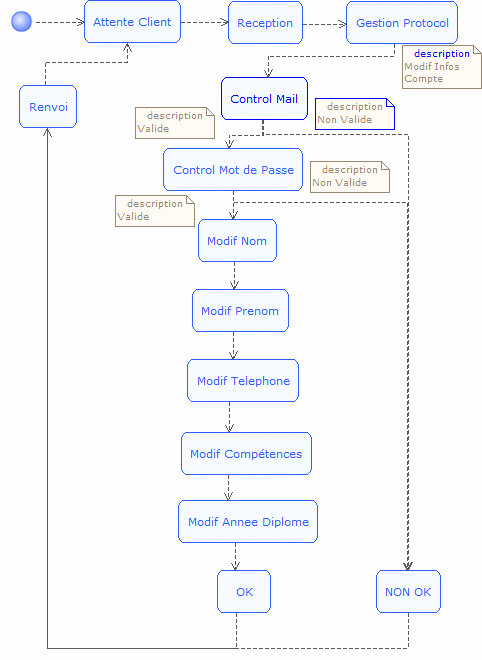
Ce premier diagramme d’activité montre les différentes étapes lors de la connexion.



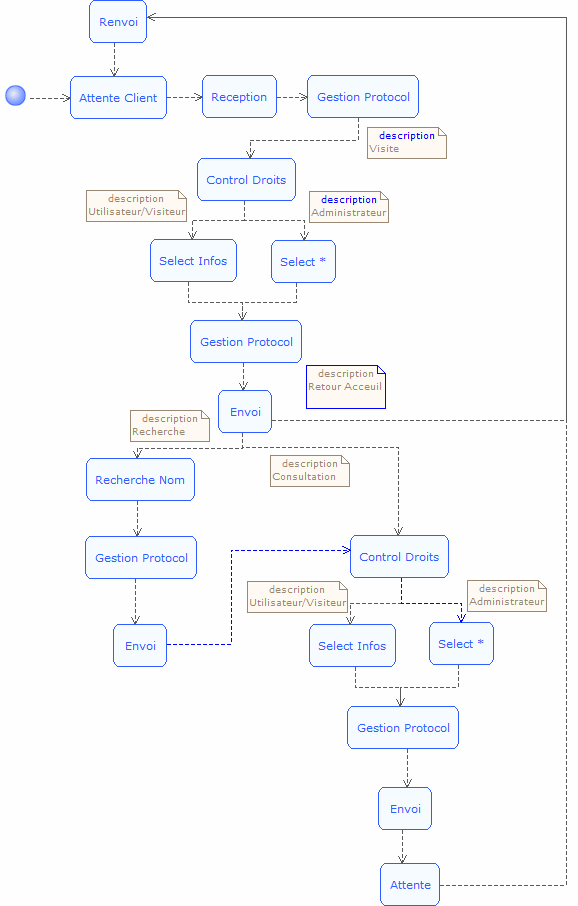
Ce deuxième diagramme d’activité permet de montré les étapes de la création de compte.



Ce troisième diagramme d’activité est consacré à la modification des informations des comptes des utilisateurs.



Ce dernier diagramme explicite toute la visite de l’annuaire des étudiants ainsi que la visite des profils avec la gestion des droits suivant le cas d’utilisation de l’application.



## Diagramme de classe.

## Diagramme de séquence.

# Protocole entre Client/Serveur

## Connexion

Tout d’abord, il faut définir les informations qui seront utilisées par le client pour se connecter.   
Le nom ou le prénom : impossible car il y a des possibilités d’homonymes  
On choisira donc d’utiliser comme identifiant, l’adresse e-mail. Evidemment, on l’associera un mot de passe définit par l’utilisateur lors de l’inscription.

Afin de savoir si un client est connecté ou non, on pourrait assigner une variable dans un champ de la trame émise lorsque ce dernier demande à consulter le profil d’une personne afin de ne lui envoyer seulement les informations auquel il a le droit de regard.   
Il s’agit d’une approche comprenant énormément de risques.

On opte donc pour assigner une variable au niveau du thread, sur le serveur.

La trame sera envoyé comme suit :

Et la réponse du serveur sera :

## Déconnexion

Il y a deux cas pour lesquels une déconnexion est demandée par le client :

* Simple déconnexion du compte.
* Déconnexion du compte suite à l’arrêt du programme client. Dans ce cas il faudra également mettre fin au Thread

Les messages envoyés dans chacun des cas respectivement seront :

## Modification des informations

On récupérera toutes les données de l’étudiants depuis le serveur et afin qu’il puisse les modifier par la suite.

## Inscription

L’étudiant enverra les informations nécessaires à la constitution de sa fiche personnelle lors de l’inscription.