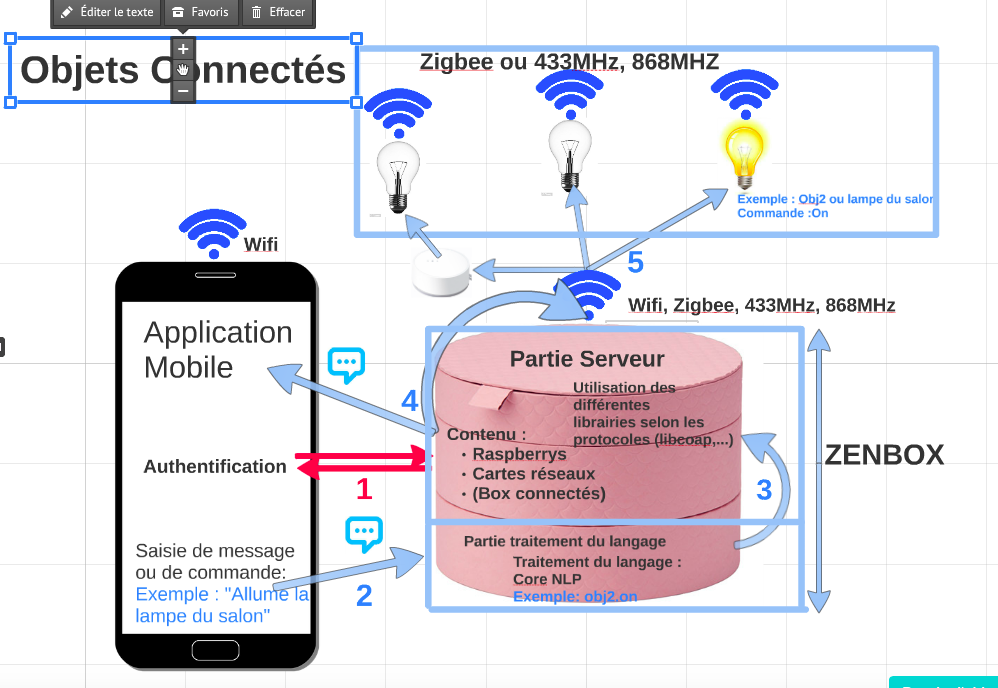
II. Schéma global



Comme vous pouvez le voir sur ce schéma, dans notre projet, il y a 3 grandes parties : l’application mobile du client, les objets connectés et notre box. Dans notre box, on pourra trouver des Raspberry Pi qui feront office de serveur et partie traitement de langage.

Pour commencer, le client s’authentifiera (pseudo et mot de passe) depuis l’application mobile, ces données seront envoyées à la partie serveur qui les traitera et renverra une réponse en fonction du résultat.

Les commandes seront effectuées depuis l’application mobile. Elles seront envoyées ensuite à la partie traitement du langage qui interagira ensuite avec la partie serveur. La partie serveur aura pour rôle d’envoyer les commandes aux différents objets connectés.

Dans la suite du délivrable, nous allons détailler les différentes parties en suivant le chemin que va prendre la commande.

V. Serveur

Une image contenant texte, carte

Description générée avec un niveau de confiance très élevé

Nous avons choisi de coder notre serveur en JavaScript. Ce choix est basé sur le fait que nous avons vu ce langage en cours et c’est donc le langage que l’on maitrise le plus. De plus, dans le cours de WebObject, nous avons fait un serveur de tchat. Nous allons donc réutiliser les bases de se serveur pour concevoir le notre. Cela facilitera donc sa conception. De plus, nous avons vu que node.JS était compatible avec CoreNLP. Il existe des packages (corenlp) et des wrapper (stanford-corenlp) disponible sur node.js. Ainsi, notre partie traitement de langage va pouvoir communiquer sans le moindre problème avec le serveur. De plus, vous verrez ensuite, que le serveur traite également avec le client. Il existe des librairies compatibles avec Android, cela a donc faciliter la communication.

Dans le serveur, il y aura deux parties, une partie traitement des commandes et une partie base de données. Comme vous l’avez vu, le client va devoir s’identifier avec un nom d’utilisateur et un mot de passe. Il va donc falloir stocker ces données. Nous avons choisi d’utiliser sqlite3.

Notre serveur aura deux rôles :

- Envoyer des données au client

- Envoyer des données aux objets connectés

1. Envoyer des données au client

Il y aura deux moments où le serveur enverra des données au client. Le premier est lorsque que le client s’identifiera. Ainsi, lorsqu’il aura reçu les données du client, le serveur stockera ces données dans une base de données puis renverra au client la validité ou non de son authentification.

Ensuite dans un second temps, lorsque le client enverra des commandes via l’application mobile afin d’effectuer des actions sur les objets connectés, le serveur renverra l’état de la commande. C’est-à-dire que le client aura la possibilité de savoir si sa commande a bien été effectuée ou non.

Ces différents retours d’état seront envoyés sous la même forme que les commandes envoyées dans notre application de tchat.

2. Envoyer des données aux objets connectés

Pour les objets connectés, il y aura deux cas de figures. C’est-à-dire qu’il peut y avoir des objets connectés ,tels que la lampe ikea tradfri, qui possèdent une box ainsi que d’autres qui n’en possèdent pas. Ainsi le serveur devra être capable de s’adapter. Lorsqu’une box sera présente, le serveur se servira des librairies préalablement installées permettant de communiquer avec la box (ex : libcoap). Ensuite la box se chargera de communiquer avec l’équipement.

Si il n’y a pas de box, il faudra alors installé une carte wifi, ou un module radio ou zigbee permettant au serveur de communiquer « en direct » avec les différents équipements.