

**PAO *smartcontrol***

-

Document de conception

BOURGEOIS Alex

29 mars 2016

# Chapitre 1

## Introduction

### 1.1 Présentation

Le présent document entre dans le cadre des Projets d'Approfondissement et d'Ouverture (PAO) du semestre 4.2 et est encadré par M. Alexandre Pauchet. L'objectif du projet est de développer un système permettant de contrôler une application Desktop par l'intermédiaire d'interactions avec un smartphone Android.

Le smartphone est un serveur auquel se connecte un PC jouant le rôle de client et récupérant les états des différents capteurs présents sur le téléphone. Dans ce sens le smartphone sera utilisable comme contrôleur pour diverses applications grâce aux données de l'accéléromètre, du gyro- scope, de l'écran tactile, etc. La preuve de fonctionnement du système sera le contrôle d'objet 3D dont les mouvements seront associés à ceux du téléphone.

### 1.2 Spécifications

#### 1.2.1 Généralités

- Le projet consistera à établir une connexion Bluetooth avec un serveur fonctionnant sur un smartphone Android depuis un client situé sur un ordinateur et ainsi il sera possible de récupérer les données de capteur du téléphone.

##### **Pourquoi Bluetooth ?**

il s'agit d'une technologie rapide et plus économe en énergie que le Wifi ce qui n'est pas négligeable sur un appareil mobile. De plus la liaison sera directe et ne sera pas encombrée comme pourrait l'être une liaison Wifi ;

- Le développement du système sera fait sous Qt.

##### **Pourquoi Qt ?**

Qt est multi-plateforme (windows, linux, OSX, android) ce qui facilitera grandement le développement par l'utilisation d'un langage unique : le C++ ;

- La société Qt s'oriente vers la 3D avec la création d'un moteur de rendu 3D donc ce projet permettra d'en explorer quelques possibilités comme la manipulation de caméra via l'utilisation des données capteur issues du téléphone, ceci servira comme preuve de fonctionnement de l'application.

### 1.2.2 Communication

- Le smartphone est un serveur auquel vient se connecter un client qui est l'ordinateur ;
- Le client demande au serveur les données de capteur. Les capteurs disponibles sont communiqués par le serveur lors de la connexion d'un client ;
- Le temps de réponse devant être très court afin d'obtenir une réaction en temps réel, il y a besoin de limiter la taille des trames.  
Pour ne rater aucune information (au cas où le serveur émettrait en continu), le serveur attend que le client lui demande quel capteur il souhaite avant d'envoyer l'information. Cela permet d'économiser de la batterie du serveur et d'avoir une communication propre.

### 1.2.3 Protocole de communication

- Il existera 3 types de requêtes :
- Établir la connexion ;
  - Demander la valeur d'un ou plusieurs capteurs ;
  - Fermer la connexion.

#### Établissement de la connexion

- Le client fait une requête de connexion ;
- Le serveur y répond et si elle est positive, envoie la liste des capteurs dont il dispose en se souvenant de l'ordre ;
- Le client enregistre la liste des capteurs dans le même ordre.

#### Requête sur un capteur

- Le client fait une requête sur des capteurs en envoyant simplement : "c1 c2 cx" en fonction des capteurs qu'il souhaite ;
- Le serveur y répond en envoyant les données dans le même ordre : "31 ;45 ;-8 52 ;10 xx yy ;zz" en fonction des capteurs.

#### Fermeture de la connexion

- Le client fait une requête de fermeture ;
- Le serveur y répond et clot la connexion.

Il est à noter qu'en l'absence de réponse du serveur à 2 requêtes du client la connexion est fermée.

En l'absence de communication du client pendant 5 minutes, le serveur ferme la connexion.

# Chapitre 2

## Architecture

### 2.1 Diagrammes

— Diagramme d'interaction

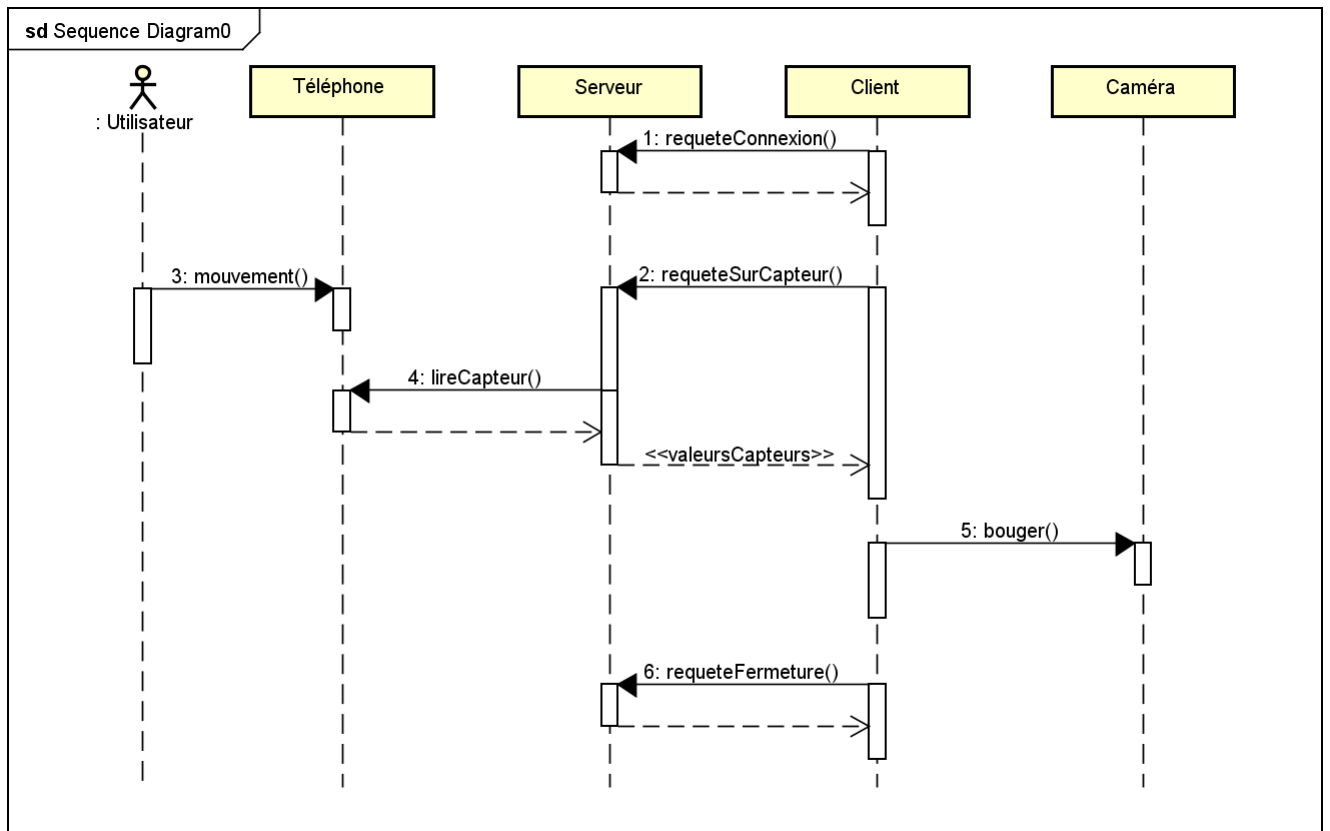


FIGURE 2.1 – Diagramme d'interaction

— Diagramme de classe

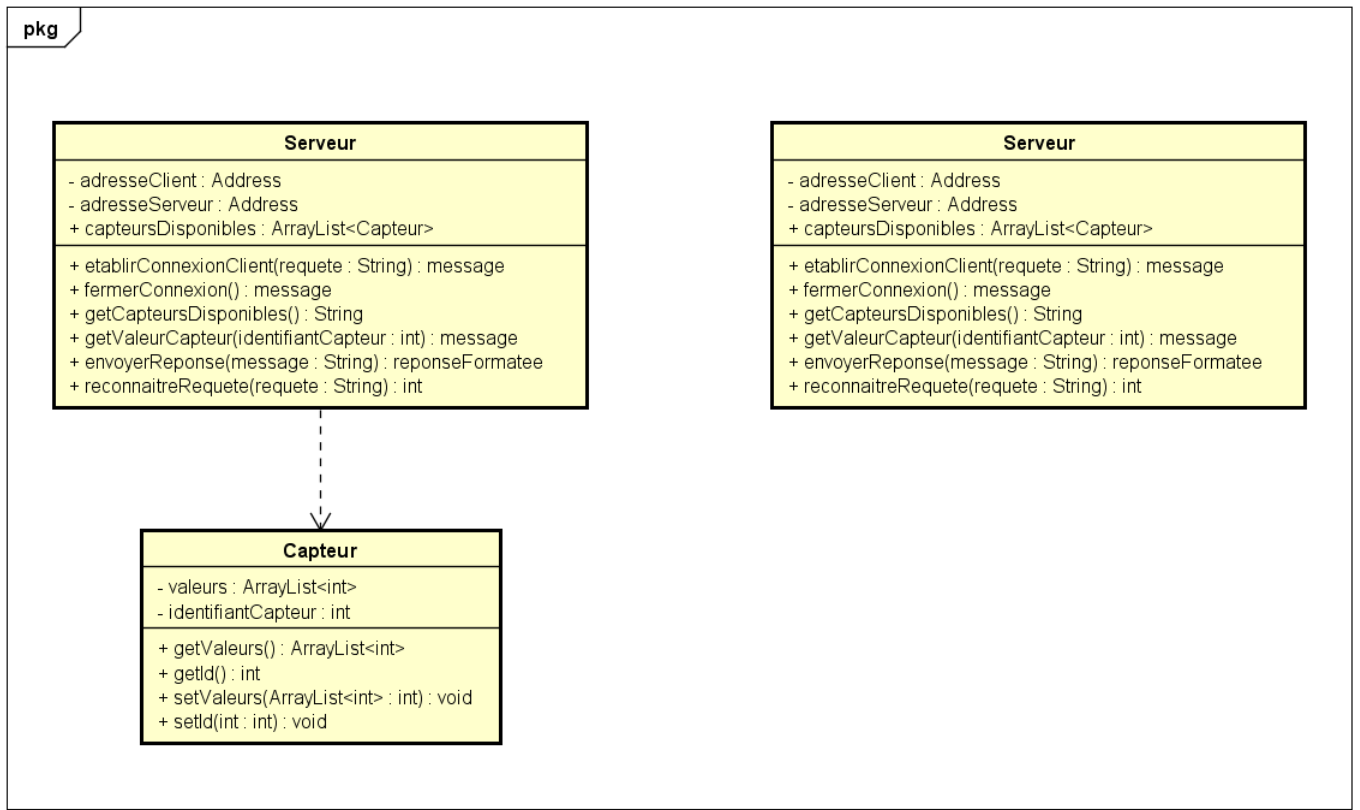


FIGURE 2.2 – Diagramme de classe

## 2.2 Client

Analyse descendante du Client

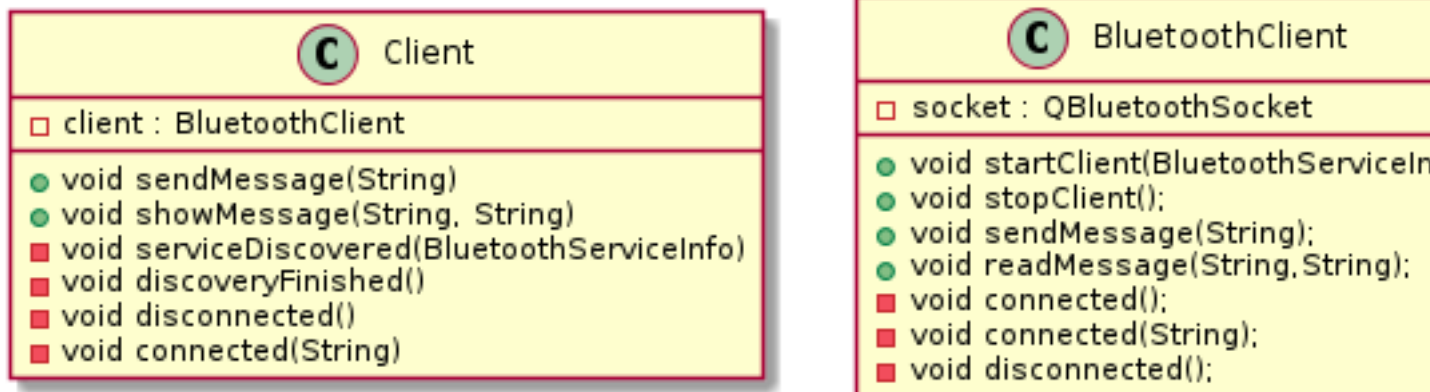


FIGURE 2.3 – Diagramme de classe Client

## 2.3 Serveur

Analyse descendante du Serveur

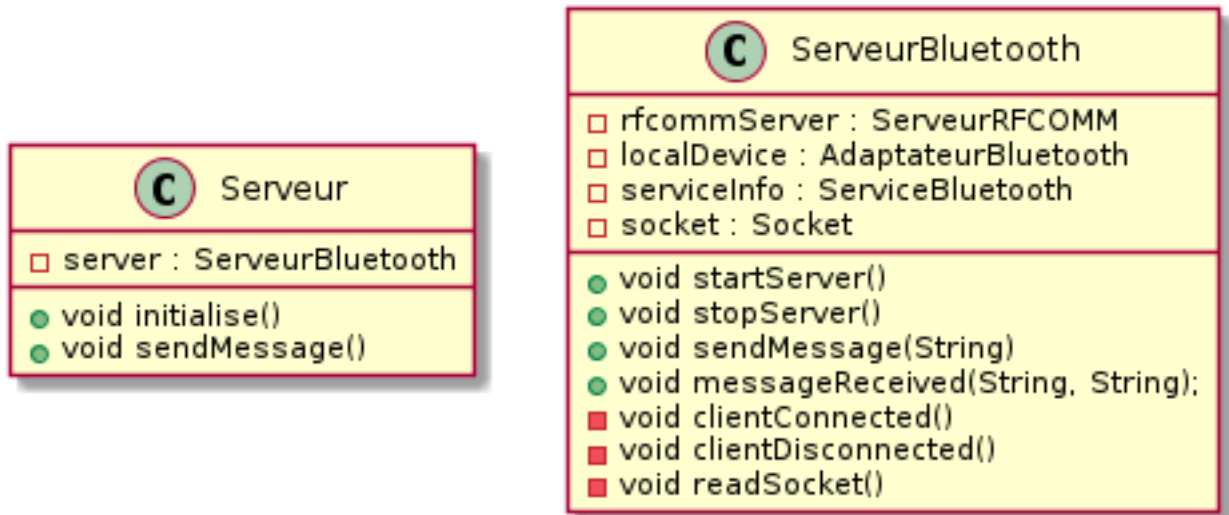


FIGURE 2.4 – Diagramme de classe Serveur

