Project Proposal

Student Name: Barry McCabe

Student Number: x21156131

## OVERVIEW

This document is submitted as an interim submission of CA project of the Distributed Systems module on the Higher Diploma in Computing specialising in Software Development delivered by Yasantha Samarawickrama.

# Domain description

The proposal is to design a Smart Home Automation System, which creates an interconnected environment where smart devices collaborate to enhance the end users living experience within the home.

The System will consist of three primary services:

1. Lighting Service: This will manage smart light bulbs, allowing the end user to turn lighting on/off and adjust brightness.
2. Thermostat Service: This will manage smart thermostats, enabling the end-user precise control over home temperatures allowing them to set and adjust temperatures for different zones in their home.
3. Security Service: This will monitor smart security devices such as cameras and door sensors, providing the end user real-time updates and alerts.

# Service definition and RPC

**service LightingService:**

The Lighting Service controls the lighting, allowing the user to adjust brightness levels and turn lights on or off.

**RPC Definitions:**

1. **GetStatus:**

**Request**: Empty

The client sends an empty request to the server.

**Response:** Returns the current status of all connected light bulbs, including unique IDs, current brightness levels, and on/off status.

1. **SetBrightness:**

**Request:**

int32 light\_id: The unique identifier of the target light bulb.

float brightness: The desired brightness level.

**Response:** Confirms the successful adjustment of the brightness setting.

1. **SetPower:**

**Request:**

int32 light\_id: The unique identifier of the target light bulb.

bool power\_status: The desired power status (true for on, false for off).

**Response:** Confirms the successful change in power status.

**service ThermostatService**

The Thermostat Service controls the temperature within the home, allowing users to set and adjust the temperature.

**RPC Definitions:**

1. GetCurrentTemperature:

**Request:**

int32 thermostat\_id: The unique identifier of the thermostat.

**Response:** returns the current temperature for the specified thermostat.

1. SetTemperature:

**Request:**

int32 thermostat\_id: The unique identifier of the target thermostat.

float target\_temperature: The desired target temperature setting.

**Response:** Confirms the successful adjustment of the target temperature.

**service SecurityService:**

The Security Service manages security devices within the home. The service can be turned on/off by the user and the service.

**RPC Definitions:**

1. GetAlerts:

**Request:** Empty

The client sends an empty request to the server.

**Response:** Retrieves a list of recent security alerts.

1. **ActivateAlarm:**

**Request:**

int32 zone\_id: The unique identifier of the security zone.

**Response:** Confirms the successful activation of the security alarm