Automated Plant Watering System

EID Super Project Proposal by
Bryan Cisneros and Mike Fruge

Concept

Our vision is an automated plant watering system that automatically monitors and gives plants the optimal amount of water. Remote moisture sensors (in a garden, for example) monitor the level of water in the soil and report their measurements to a central gateway that is connected to the internet. The gateway sends this data to the cloud so that the user can monitor the data from anywhere. When a low soil moisture level is detected, the system automatically turns on a water source to water the plant. For this project, this “water source” will be virtualized, meaning we will turn on an LED to represent the water source turning on, but the rest of the system, including the soil moisture sensor, will use physical components.

To interact with the system, the user will have a choice of two separate UIs - one that is QT based and one that is HTML based. These UI designs will look similar, and will allow the user to view recent soil moisture data and configure settings such as alarm thresholds. In a more realized version of this project, these UI’s represent a mobile and desktop interface.

Project Elements

The following elements will be used in our system. A block diagram of their relationships can be found in the next section.

IoT Sensor

We will use a soil moisture sensor, which will report moisture levels to the IoT system.

Communication Protocols

Our system will use the following communication protocols:

- I2C
- Zigbee
- UART
- MQTT
- HTTP

REST API

The AWS API Gateway will host a REST API that the UI clients can communicate with for recent measurements or to configure system settings.

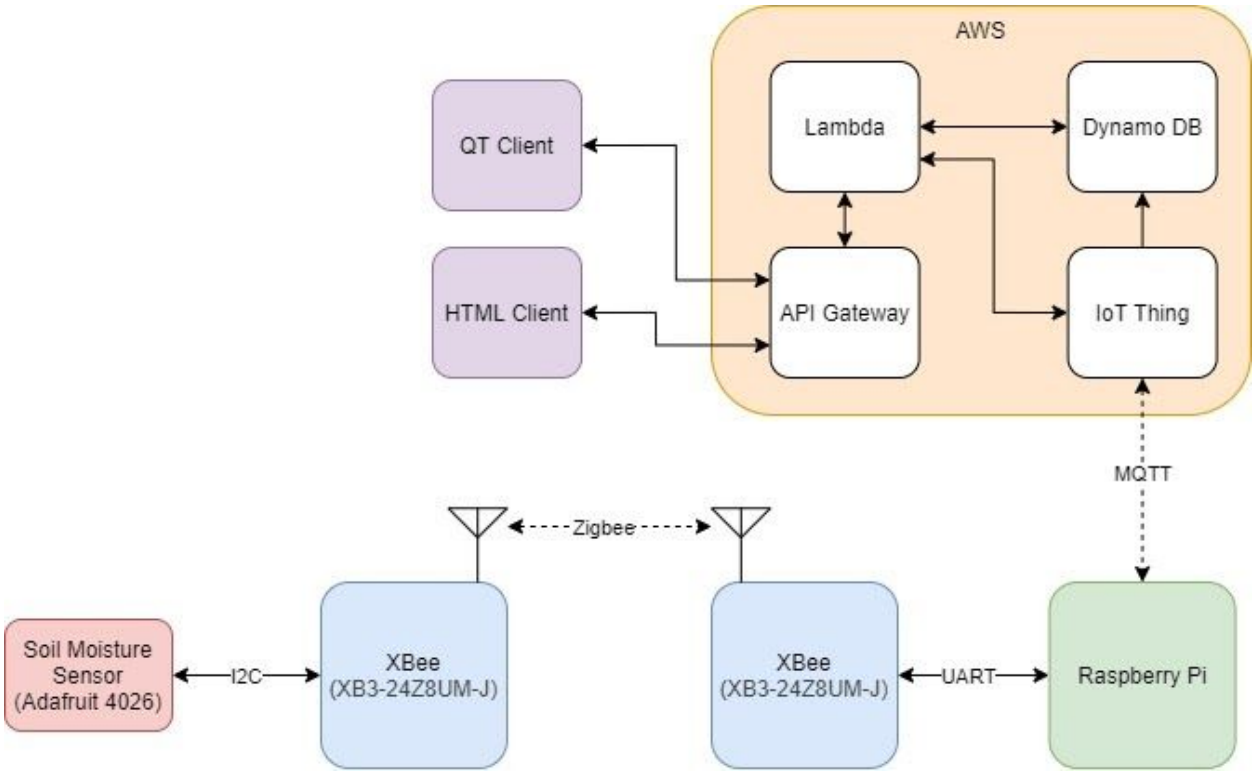
User Interface

There will be two user interfaces, one that is QT based and another that is HTML based. Both will look similar and perform similar functions. Using either UI, the user will be able to to view recent sensor data and configure settings such as alarm thresholds. The UI's will vary in size and layout to be optimized for either a desktop computer or a mobile phone.

AWS

We will use a number of AWS services to store and retrieve sensor data. The gateway will communicate with an AWS IoT Thing to report sensor data, which will be stored in a Dynamo Database. This data can be accessed by using the UI to connect to the REST API running on the AWS API Gateway. When a request to the API is made, Lambda will fetch the requested information from the database and return it to the user.

System Diagram



Hardware

All the hardware needed is represented in the system diagram shown above. This includes the following items:

- Soil Moisture Sensor (Adafruit 4026)

- 2x XBee3 Zigbee (XB3-24Z8UM-J)
- Raspberry Pi

We want to keep our system after the semester is over, so we will acquire the needed hardware ourselves.