Development of Smart Talking Plant with Voice Recognition Function

Mechelle Grace Zaragoza, Haeng-Kon Kim and Ha Jin Hwang

***Abstract*— Plants can tell us what is affecting them with the electrical signals they emit with the help of (Internet of things) IoT.In such ways that is your plant thriving under current light levels? Is your plant thirsty? Or even satisfy with your care? The goal this project is to encourage a happy relationship between plants and people as a lot of people are afraid of plants. Not they are not green handed. The idea of taking good care is a responsibility. This makes it possible for them to have a plant in their lives. As Arduino came into the picture in bringing a better solution for plant care, we proposed an ideal smart plant based on an existing system that will add more features to it to make an all in one architecture. We also presented some useful projects related to this for comparison and future references.**

***Index Terms*— Arduino, Smart talking Plant, Java Voice Recognition**

I. Introduction

This IoT project describes how to build a project that monitors the plant health status by keeping contact with the owner. As a result, the project wants to check some environment parameters like temperature, humidity and light intensity that have effects on the plant. In addition, the project wants to retrieve the soil moisture and at the end of it, tells the owner what it feels by producing a sound produced by a voice enabled device attached to a computer. In addition, a mobile application is possible for the plant to be able to produce test messages, tweets or Facebook posts that tell its owner how thirsty it is by just sending a tweet and of course sending the owner a thank you reply after watering it or removing it from the heat of the sun. More on building an IoT project that monitors plant health status Future plans will be an attached webcam for further monitoring.

II. Related Work

A. Singing Plant

Make Your Plant Sing With Arduino, Touche and a Gameduino.You may have seen a plant used as a musical instrument before, measuring the capacity through the shape of a smooth life and turning it into a tactile sensor. This is something different: it leaves the factory itself expresses communication through sound, using biofeedback to activate the system's life into something audible. If two objects are conductive, it is possible to measure the capacitance between them (for example, objects such as metal, water, human bodies, and plants). By connecting a capacity sensor to a plant, we can detect whether to touch and convert to its contact. Normally, you have to do this with an interface. With the introduction of the frequency principle of Touché Research Disney, the creation of a true stable solution that can be done entirely on the Arduino platform is here. The Touch works on a principle similar to a regular theremin, but a frequency sweeps the place. By hanging the system can the touched measure the capacitive interaction and convert it into sound. [1] and [7].

B. Sigfox Talking Plant

It is a simple project based on the SIGFOX as shown in Fig. 1. The network does an interview on plants on Twitter. The main objective is to demonstrate on making a simple prototype using the Sigfox Network. SIGFOX is dedicated to the Internet connectivity solution objects. The operating network currently operates in +15 countries, on all continents. Focused on small messages (up to 12 bytes) and low energy consumption, currently it feeds 7 million devices. There are several technical solutions compatible with SIGFOX, from different suppliers of silicon. This project uses an Atmel Arduino shield. You can use this project to gather a community around the plant or just to remind your supporters of the water plant. Do not hesitate to share how they used this project. [2]The temperature sensor's provided with the ATMEL shield, thus photocell resistor is only needed (to get the luminosity) and the moisture sensor. What we need is to get the soil moisture, the luminosity and the temperature.

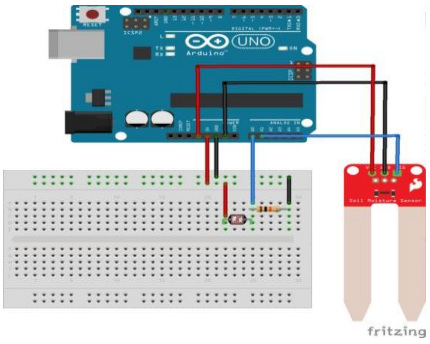

Fig.1. Hardware Connections of Sigfox

Manuscript received July 16, 2017;

Mechelle Grace Zaragoza, is with School of IT, Catholic University of Daegu Korea. (email:mechellezaragoza@gmail.com).

Haeng-kon Kim is with School of IT, Catholic University of Daegu, Korea.(email:hangkon@cu.ac.kr.).

Ha Jin Hwang is with Department of Business Analytics, Sunway University Business School, Malaysia(email:hjwang@sunway.edu.my).

C. PAL PLANT

Presented in Fig. 2, Palplant is also powered by solar energy, you don't even need to charge it. Your first ever tweeting plant. Just stick it in from your mobile phone your plant can talk to you. By putting the device on your plant. Give your plant a name and it will talk to you. [3]

Fig. 2. Pal Plant

III. PROPOSED SMART TALKING PLANT DEVELOPMENT

A. Concept

As shown in Fig. 3, the architecture is for a smart based talking plant. From a user's computer, a running javascript is embedded. As in runs the program, through the detection of the sensors, it sends direct input to the plant owner as it alerts via or social media posts. The mobile Application enables this to happen. Plant display is optional, but if the user might want to visualize the soil moisture or the amount of light a plant it should be considered. The Talking Plant got a light, temperature, soil moisture and two motion sensors for the functionalities. The plant complains if any of the values of the sensors are excessive and is able to interact with people by answering simple questions with the help of JavaScript voice recognition. The Project got a simple face in the form of an 8×8, red LED board that can display basic expressions as well as turn his eyes in the direction where people are approaching from.

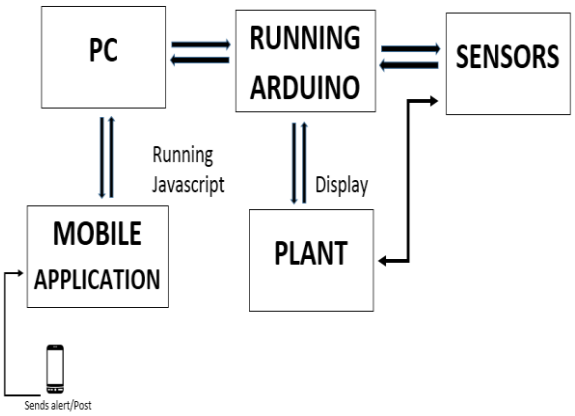

Fig. 3. Proposed Architecture for Smart Talking Plant

A. ARDUINO EXPLAINED

As shown in Fig. 4, Arduino can be used to develop stand-alone interactive objects or can be connected to software on your computer.

(e.g. Flash, Processing, MaxMSP).Arduino is an open-source physical computing platform based on a simple i/o board and a development environment that implements the Processing/Wiring language. [4] and [8]

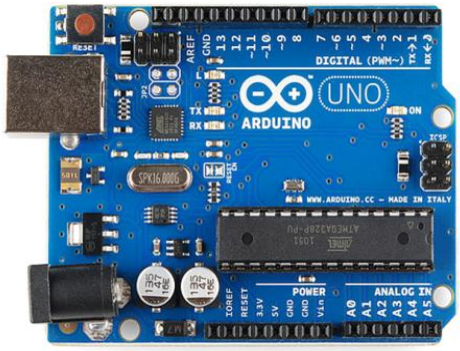

Fig. 4. Arduino Components

B. Hardware Components

Using an 8×8 LED board for facial expression with a built in MAX7219 controller chip. Also add the LM35 temperature, a simple soil moisture, and a light sensor and finally for motion detection we could use two PIR sensors. The project moves its eyes where you come from and only talks if someone is around. The sensor data is concatenated into a JSON string which is then sent to Node.js through the USB. Using web sockets the Node server publishes this data for the browser. [5]

D. IMPLEMENTATION

To begin with, this previous system was programmed using API and series of if/elses conditions. Setting this is easy as the vocabulary is extendable. Needless to say, everything in javascript code is stored in an object for each event making it randomly picks the desired code output.

Listed below:

List 1. This part of the program gives the ability for the plant to speak on the current status/situation through evaluating its environment. Example on temperature. Plant might be able to say "It's too hot in here" if the plant can't bare too much heat.

```
//LIST 1
  plant = {
    complain: {
      hot: [
        "It's too hot in here.",
        "It's really warm in here",
        "I can't take this temperature",
        "My leaves are burning",
        "Can someone turn off the heating please?"
      ],
      cold: [
        "It's too cold in here.",
        "It's really cold in here",
        "It's freezing",
        "I can't take this temperature",
        "Can someone turn up the heating please?"
      ]
```

List 2. Anoher set of codes that allows the plant to talk every 3 minutes. This boolean value is then used in the speak function:

// This part of the program allows the plant to speak every 3 minutes.

```
//LIST 2
if(data.temperature < app.temp_min) {
  // I'm cold
  app.speak(
    plant.complain.cold[app.rdm(5)] + " It's " +
data.temperature + " degrees.", false
  );
}
if(data.temperature > app.temp_max) {
  // I'm hot
  app.speak(
    plant.complain.hot[app.rdm(5)] + " It's " +
data.temperature + " degrees.", false
  );
}
```

List 3. This part is the Limiting speech:

Actions like someone tends to water the plant, it allows the program to disable its delay allowing the plant to use the waiting time for 3 minutes until the voice feeds back.

The setInterval is needed to have a time range within the soil moisture level has to increase by 10% to trigger the event.

```
//LIST 3
  app.spokenTimer = setInterval(function(){
    app.recentlySpoken = false;
  }, 3 * minutes);
```

List 4. The program can understand simple questions and answers them using the HTML5 Web Speech API. example of Hello World example of voice recognition is coded: HTML5 SpeechRecognition API

```
//LIST 4
if(app.moisture + 10 < data.moisture) {
  // I'm being watered
  app.moisture = parseInt(data.moisture);
  app.speak(
    plant.thank.water[app.rdm(5)] + " The last time I
was watered was " + watered.returnDate(), true
  );
  localStorage.setItem("lastWatered", new Date());
}

// Sampling moisture
app.moistureTimer = setInterval(function(){
  app.moisture = parseInt(app.plantData.moisture);
}, 30 * seconds);
```

List 5:There are two helper functions, added to make talking a very simple process:app.matchWords([array], string) - This needs to pass in an array of words as the first attribute and a piece of string as a second.

```
//List 5
var recognition = new webkitSpeechRecognition();
recognition.onresult = function(event) {
  console.log(event);
}
recognition.start();
```

List 6: The function then returns true or false whether any of the words in the array has been found in the string.

app.answer(string) - Pass in the piece of string as an answer and the browser will read it out loud.

An example that would answer you if you asked: "What's the temperature?" or "Are you hot?":

```
//LIST 6
// Temperature
if(app.matchWords(
  ["temperature", "hot", "cold", "warm"], text
)){
  app.answer(
    plant.answer[temp][app.rdm(2)] + " " +
    app.plantData.temperature +
    " degrees"
  );
} [6]
```

F. Sequence Diagram

As shown in Fig. 5, As the user arrives and asks questions to the plant, the plant gives it feedback as what the user asks. Based on the program given and depending on the questions asked by users, the plant will respond to the user like “My leaves are burning” by the question the user asks. By then the user could transfer the plant into a shaded area if it’s under the heat of the sun for too long.

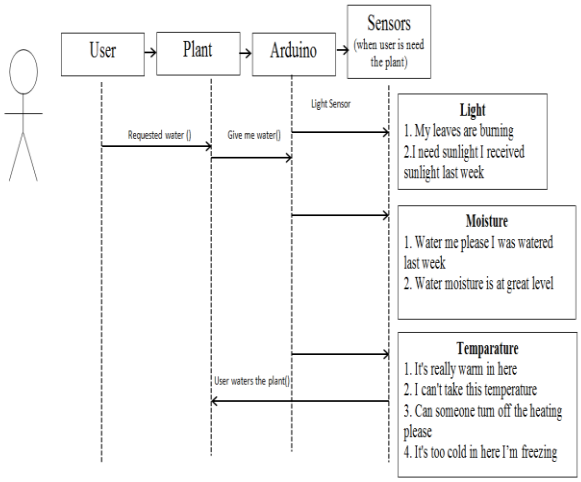

Fig. 5. Sequence Diagram of Smart Talking Plant

IV. Evaluation

As presented in the existing system, it lacks proper boxing as the wires are all over the place. The need of final HTML5 web speech library is also needed to see if it is reliable at sometimes when it hears its own voice, the system tends to speak and listens at the same time and picks up and reacts to his own voice. It does not have online tweet and has no webcam on it to be able to monitor if the owner wants a day to day activity. This was only programmed using Arduino. This is a JavaScript project so the microphone and speaker are attached to the computer and the browser. A brand new version of this with an Arduino MP3 player is being made to completely eliminate the need of a computer. However, with that voice recognition is gone: it can only speak and not listen. As presented below in Fig. 6, represents a typical set-up of a talking based arduino plant. It has sensors such as light and moisture sensors. Display screen and the arduino circuits.

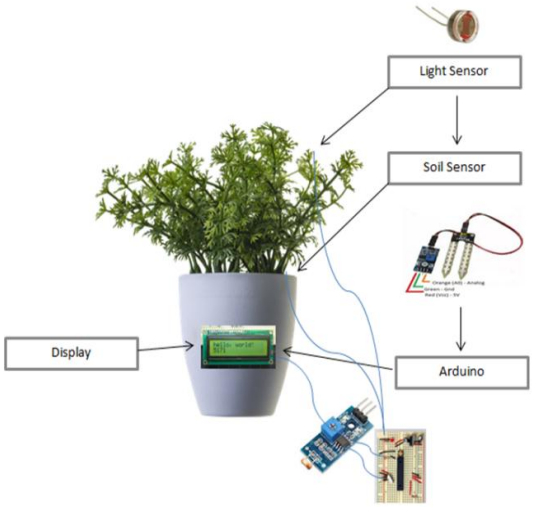

Fig.6. Proposed Set-up for Arduino based Smart Talking Plant

V. Conclusion

Based on the myth, people might want to chat with their plants more often. This technique allows the plant grow more rapidly. Studying this new realm of plant interaction, despite of not having eyes, ears, or a nervous system, plants are anything but uncommunicative. This project will enable plants to clearly let its owner understand how it feels. The goal is to encourage a happy relationship between plants and people. In addition, in the second part of IoT project, we will explore how to enable triggers on the sensor values stored. As mentioned, a proposed mobile application is needed for the plant to be able sent messages via posts or even sent SMS on mobile phones. Moreover, we will send an alert to owner via social media posts when some parameter value is out of the range like water, soil moisture etc.

Acknowledgment

This Research was supported by the MSIP (Ministry of Science, ICT and Future Planning), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2017-2013-0-0087) supervised by the IITP (Institute for Information & Communication Technology Promotion).

References

[1] Bjorner D., Domain Engineering: A Software Engineering Discipline in Need of Research. In: Hlaváč V., Jeffery K.G., Wiedermann J. (eds) SOFSEM 2000: Theory and Practice of Informatics. SOFSEM 2000. Lecture Notes in Computer Science, Springer, Berlin, Heidelberg vol 1963, (2000).

[2] Kim, Haeng-Kon. "Model Driven Engineering for Crop Monitoring Applications."International Journal of Software Engineering and Its Applications,10.6 (2016), pp.125-140.

[3] Carnegie Mellon University, "Domain Analysis," Feb. (1999), http://www.sei. cmu.edu/domain-engineering/domain-anal. Html.

[4] Hassan Gomaa and Larry Kerschberg, Domain Modeling for Software Reuse and Evolution, http://mason.gmu.edu/~kersch/KBSE_folder/KBSEE_folder/RioCASEConference.html

[5] Object Oriented Database Systems, http://www.upriss.org.uk/db/42033lecture_one.html

[6] Bjørner, Dines. "Domain engineering." Formal Methods: State of the Art and New Directions. Springer London, (2010), pp. 1-41.

[7] Carnegie Mellon University, "Domain Analysis," http://www.sei. cmu.edu/domain-engineering/domain-anal. html

[8] Preto-Diaz, "Domain Analysis:An Introduction," http://www.galaxy,net/kpt/VA/Domain,Analysis Into.html