Combination of MIT App Inventor, Arduino Yun and Temboo: An Innovative Approach for Final Year Project based on Embedded System

Hasnim Harun
Politeknik Kota Kinabalu, Sabah Malaysia, hasnim@yahoo.com

ABSTRACT

Final year students in Diploma in Electronic Engineering (Computer) in Politeknik Kota Kinabalu, Sabah Malaysia are compulsory to design and develop an embedded system project. Most of the time, the student have difficulty to determine and selecting which embedded system platform to to use and the programming language involved. The most popular embedded system platforms such as PIC and Arduino also have many type and variants. To construct an innovative project, the project should include other features and capability such as capability to send sms or using smartphone app to control electronic devices. There for, by recommending and assisting the students using the combination of MIT App Inventor to develop android Apps, Arduino Yun for the hardware and Temboo as API provider will accelerate the design and development of the project. In this paper, in the 2nd topic will explain why the hardware and software above are recommended to use. In the 3rd topic will show how easy and fast to design and develop "A Speech Recognition App for Senior Citizens".

Key words : MIT App Inventor, Arduino Yun, Temboo, E-Learning, Engineering Education, Final Year Project

1. INTRODUCTION

For most of the engineering programs, it's a compulsory for their student to take Final Project to complete their study. For Diploma in Electronic Engineering (Computer) student in Polieknik Kota Kinabalu, it's a compulsory for them to design and develop embedded system project. Some student will use android application together with embedded system to develop their project. Some will combine with gps, sms, e-mail, social media and etc.

At the early stage for designing their project, a student have a dilemma to determine and selecting a platform to use, because there are so many platform in the market can be used. For example, embedded system can vary from PIC, Arduino and others. Android App can be develop by many platform, SDK and languages. There for, a student will be confuse and will take much time to choose. After the process of choosing, there will be the a learning time to learning the selected platform or App SDK.

2. AN INNOVATIVE APPROACH USING COMBINATION OF SOFTWARE/HARDWARE

There are 3 software and hardware chosen to be recommended to be used as a final project tools and each will be explain below.

MIT App Inventor

MIT App Inventor as shown in Figure 1 is a tool that is capable to open up new mobile world to a broad audience which allow people to not just use the new technology, but also produce something out of it. The idea was to democratize app building and not leave it to the purview of a small group of technically elite. It runs in a browser and permits us to create apps by plugging blocks together in tinker toy fashion, rather than by typing code. The apps created are not toys or demonstrations, but full-fledged apps. It is a very much benefit as we can share them, upload them to Google Play, and even sell them[1].

We could trace the origins of the App Inventor to 2008 when Hal Abelson taught an MIT class on mobile computing using the very first publicly available Android devices. There are 82,000 people actively using the service each week as in April 2014[2]. In total, there are 1.7 million registered users from 194 countries, who together have created (or at least started) 4.4 million apps[3].
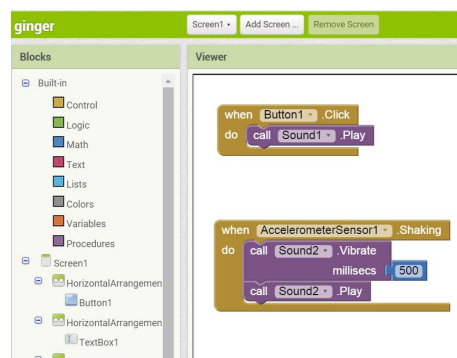

Figure 1: Mit App Inventor Block Programming

Blocks are simply drop into the program then add the blocks that are to run when the event occurs. There is no "main" program and App Inventor hides all the behind-the-scenes event-dispatching code. If there are changes made by

developer, either in the User Interface Designer or the Blocks Editor are immediately reflected in the developer's phone, such that we can sees the app and can interact with and test it instantly at every stage of its development. This have been called the "live development"[3,4].

Arduino Yun

The Arduino Yún as shown in Figure 2 is a microcontroller board based on the ATmega32u4 and the Atheros AR9331. The Atheros processor supports a Linux named OpenWrt-Yun. The board has built-in Ethernet/WiFi support, a USB-A port, micro-SD card slot, 20 digital I/O pins (7 can be used as PWM outputs and 12 as analog inputs), a 16 MHz crystal oscillator, a micro USB connection, an ICSP header, and a 3 reset buttons[5].
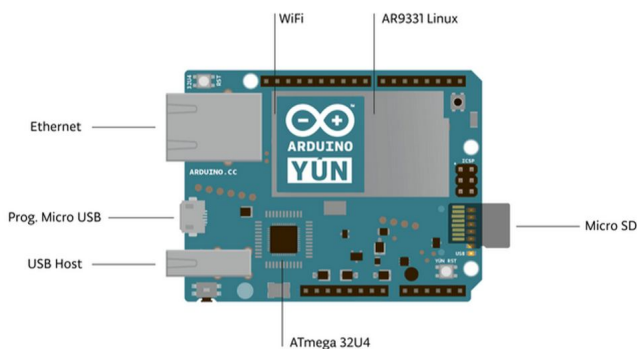

Figure 2: Arduino Yun

The Yún have its own specialty as it differ from other common Arduino boards in that it can communicate with the Linux distribution onboard, Offer a powerful networked computer with the ease of Arduino. Thus, we can write our own shell and python scripts for robust interactions as in addition to Linux commands like cURL.

These have allow Arduino sketches the ability to run shell scripts, communicate with network interfaces, and receive information from the AR9331 processor as The Bridge library facilitates communication between the two processors. The USB host, network interfaces and SD card are not connected to the 32U4, but the AR9331, and the Bridge library also enables the Arduino to interface with other peripherals.

Temboo

There is a web services that help to make it easy for the Arduino Yún that is Temboo (www.temboo.com). It facilitate to connect to over 100 web-based resources and services (e.g. Faceboo and Dropbox) by standardizing how we interact with their Application Programming Interfaces (APIs). We can connect to over 2000+ API codes. With Temboo you don't have to worry about the details[6].

The only thing required to use Temboo is, once we know how to use one API, we will know how to work with any API in it's Library and that what makes Temboo uniquely powerful and so much useful not only for masters but also to the beginners.

The other powerful Temboo features are,

Conditions – the capability to generate code that triggers an API call based on sensor values, e.g., send an email when a sensor hits a certain threshold. Also make our hardware react to specific values in API responses, e.g., turn on an LED when an API call is successful.

Profiles - keep our API credentials organized, save memory on our resource-constrained device, and reprogram our device remotely.

Output Filters - use server-side API response filtering to return only the part of an API response that our application needs.

3. CASE STUDY: DESIGN AND DEVELOPMENT OF A SPEECH RECOGNITION APP FOR SENIOR CITIZENS

To understand the design and development process using combination of 3 recommended software/hardware which are MIT App Inventor, Arduino Yun and Temboo, a case study has been done by developing "A Speech Recognition App for Senior Citizens". It's an Android App for Senior Citizens to use at home if emergency happen such as heart stroke or other emergency and to seek help when needed.

When the senior citizens want to seek help, he/she will use the app, click a button and say "Help" and the alarm or siren will be activated and the sms will be send to their son/daughter alerting that their parent may be in emergency and need help.

To connect to Arduino using Bluetooth. MIT App Inventor has a block to connect to Bluetooth. We do need to write some code at Arduino side. The block in Figure 3 below showing block programming for Speech Recognition part for android programming , while Figure 4 to connecting to Bluetooth module HC-05 from MIT App Inventor.
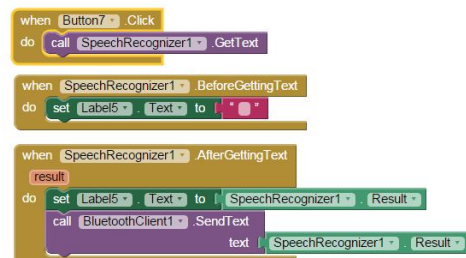

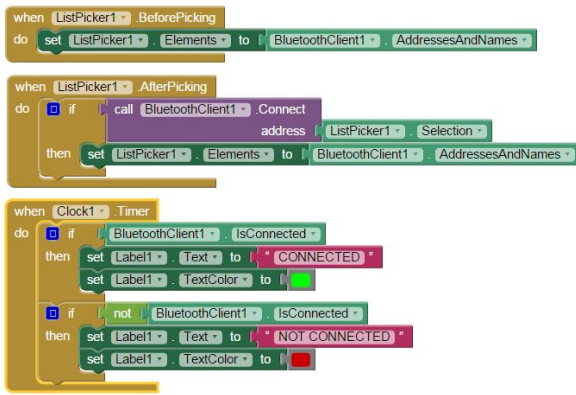Figure 3: Speech Recognition block programming

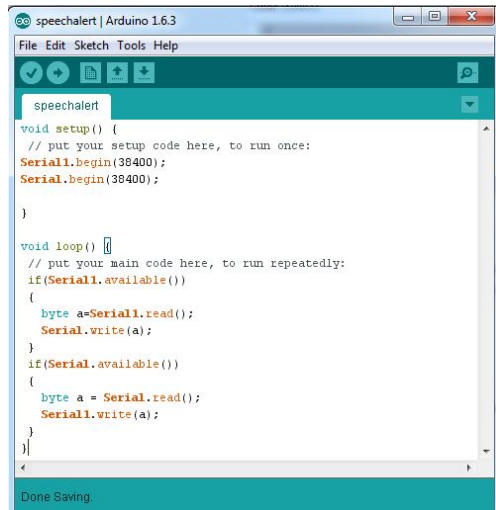Figure 4: Connecting To Bluetooth module HC-05 from MIT App Inventor

Figure 5: Arduino Yun programming to received Bluetooth

At the hardware side, we need to some coding as shown in Figure 5 to received Bluetooth signal. from App. Arduino will then be connected to electrical device through relay.
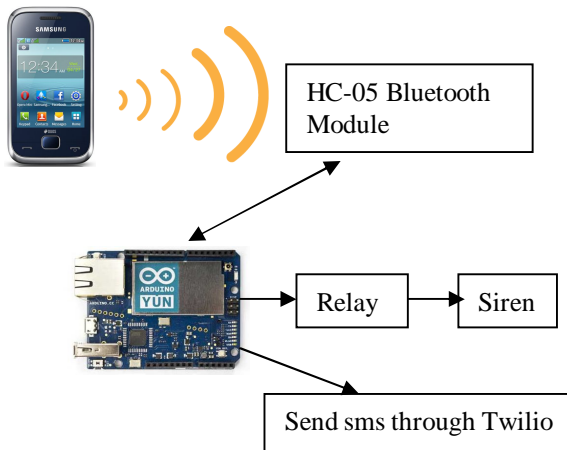

Figure 6: Design of the system

Temboo will provide a services to send sms feature. In this project, we will be using Twilio AP1. First we must register as Twilio user.

Figure 7: Twilio Account ID and Auth Token

And we will get Twilio Account ID and Auth Token as shown above.

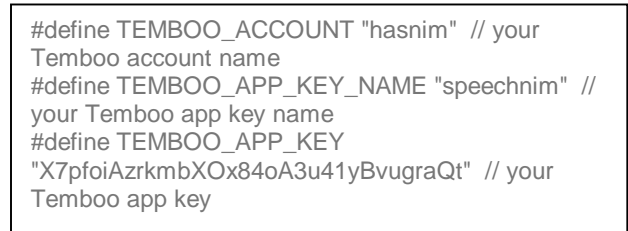Temboo will generate a Arduino sketch as shown in Figure 8 and Figure 9.

```
#define TEMBOO_ACCOUNT "hasnim"  // your
Temboo account name
#define TEMBOO_APP_KEY_NAME "speechnim"  //
your Temboo app key name
#define TEMBOO_APP_KEY
"X7pfoiAzrkmbXOx84oA3u41yBvugraQt"  // your
Temboo app key
```

Figure 8 : TenbooAccount.h

```
#include <Bridge.h>
#include <Temboo.h>
#include "TembooAccount.h"  //
contains Temboo account information

// the Account SID from your Twilio account

const String TWILIO_ACCOUNT_SID =
"xxxxxxxxxx";

// the Auth Token from your Twilio account

const String TWILIO_AUTH_TOKEN =
"xxxxxxxxxx";

// your Twilio phone number, e.g., "+1
555-222-1212"

const String TWILIO_NUMBER = "xxxxxxxxxx";
```
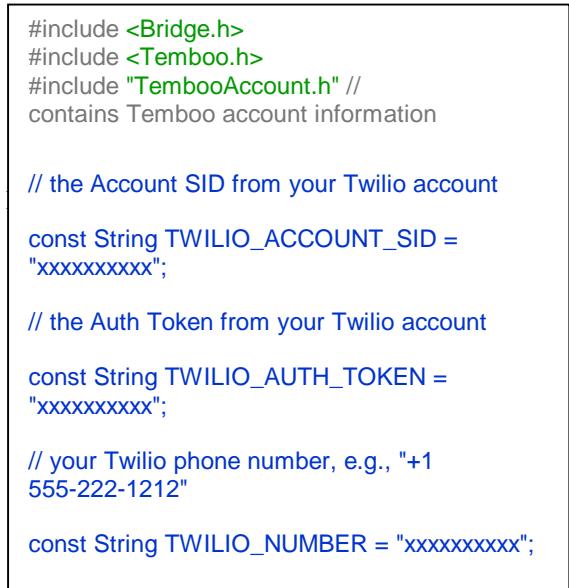
Figure 9: To include in Arduino code.

We need to replace our registered Twilio account to Figure 9.

That's it. It show that using the combination of the recommended software and hardware, the prototypes process will be much faster.

31

4. DISCUSSION

Combination of MIT App Inventor, Arduino Yun and Temboo will open many possibilities for invention because of the easy and fast development time frame. Temboo provide 2000+ API to be used. Although Temboo is paid services, it has free plan for student to develop a project. Connecting to cloud and media social is a breeze. MIT App Inventor also have a webURI features where we can request a services outside of MIT App Inventor, also Tinywebdb if want to use database. Arduino Yun also has a Linux machine and wifi provide full capability to be used in many final year project in Politeknik Kota Kinabalu, Sabah Malaysia.

5. CONCLUSION

There are many platform to choose when we want to design and develop embedded system. Through experience, this paper recommending combination of MIT App Inventor, Arduino Yun and Temboo as an innovative approach for design and development of final year project for Diploma in Electronic Engineering (Computer) based on embedded system and for development of Android App.

REFERENCES

1. D. Wolber, Harold Abelson, Nark Friedman, **Democratizing Computing with App Inventor** in GetMobile: Mobile Computing and Communications Volume 18 Issue 4, October 2014, Pages 53-58, New York, NY, USA, ACM
2. D. Wolber. **App Inventor and Real-World Motivation.** In Proceedings of the 42nd ACM Technical Symposium on Computer science education (SIGCSE '11), New York, NY, 2011. ACM.
3. E.S. David Wolber, Harold Abelson and L. Looney. **App Inventor: Create your own Android Apps.** O'Reilly, 2011.
4. B. MacKellar and M. Leibfried. **Designing and Building Mobile Pharmacy Apps in a Healthcare IT Course**. In Proceedings of the 14th annual ACM SIGITE conference on Information technology education (SIGITE '13)., pages 188{197. ACM, 2013.
5. Marco Schwartz, **Internet of Things with the Arduino Yún**, *2014*, Packt Publishing, Birmingham B3 2PB, UK.
6. Natalia Rodriguez, **This Killer New Service Lets You Connect To Any Popular API—Without Wrappers,** http://www.fastcolabs.com/3020690/ , Oct 2013.