ILR 03 – Progress Review 2

Richa Varma

Team Daedalus

Members: Pranav Maheshwari, Shivam Gautam, Mohak Bhardwaj, and Dorothy Kirlew

October 29, 2015

1. Introduction

In keeping with our goals, different members of the team worked on separate subsystems to start off with setting up basic functionalities and testing them. We were able to demonstrate the following in the Progress Review:

- Kinect was procured from the inventory and was interfaced with ROS by setting up the environment, dependencies and installing libraries. Depth map and point cloud were visualized.
- Arduino Mega was interfaced with ROS as a node.
- Functionality was added to the buttons on the basic layout of the smartphone app. Messages were sent from mobile to laptop via Bluetooth.
- A ROS node was enabled to receive serial commands via Bluetooth and publish them on a ROS message.
- The first set of communication equipment, Xbee Pro, was acquired.
- Final mobile platform, Oculus Prime was ordered.

INDIVIDUAL PROGRESS

For the past week, I had taken up the task of getting familiar with android app development using Android Studio, making the buttons functional and sending the information via Bluetooth to the laptop.

I was working with Dorothy, who has some prior experience with android app development. So I decided to first get familiar with the software. To accomplish the same, I watched tutorials on Udacity, which proved to be very helpful in getting a beginners level knowledge about android app development. In the process of learning, I experimented with code and tried out different layouts for apps. It was a good learning experience.

After completing the tutorials, Dorothy and I started working together on attaching functionality to the basic layout that was created in the previous week.

This required making changes to the Java code in Android Studio. Although I was a beginner in Java, I could understand the basics and gather what was going on. User-friendly nature of the language, coupled with adequate support documentation helped Dorothy and me in performing this task with relative ease.

We began by making the ‘Park’ and ‘Return’ buttons active and displaying different status messages for different states of the buttons:

When the user presses ‘Park’: Status changes to ‘Parking’, message field changes to ‘Car is parking’ and Estimated Time to Park (ETP) to ‘May 2016’. If the button is pressed twice, the message field displays ‘It’s still parking’ and when pressed for the third time, the button is disabled.

When the user presses ‘Return’: Status changes to ‘Returning’, message field changes to ‘Car is returning’ and Estimated Time of Arrival (ETA) to ‘One Month’. If the button is pressed twice, the message field displays ‘It’s still returning’ and when pressed for the third time, the button is disabled.

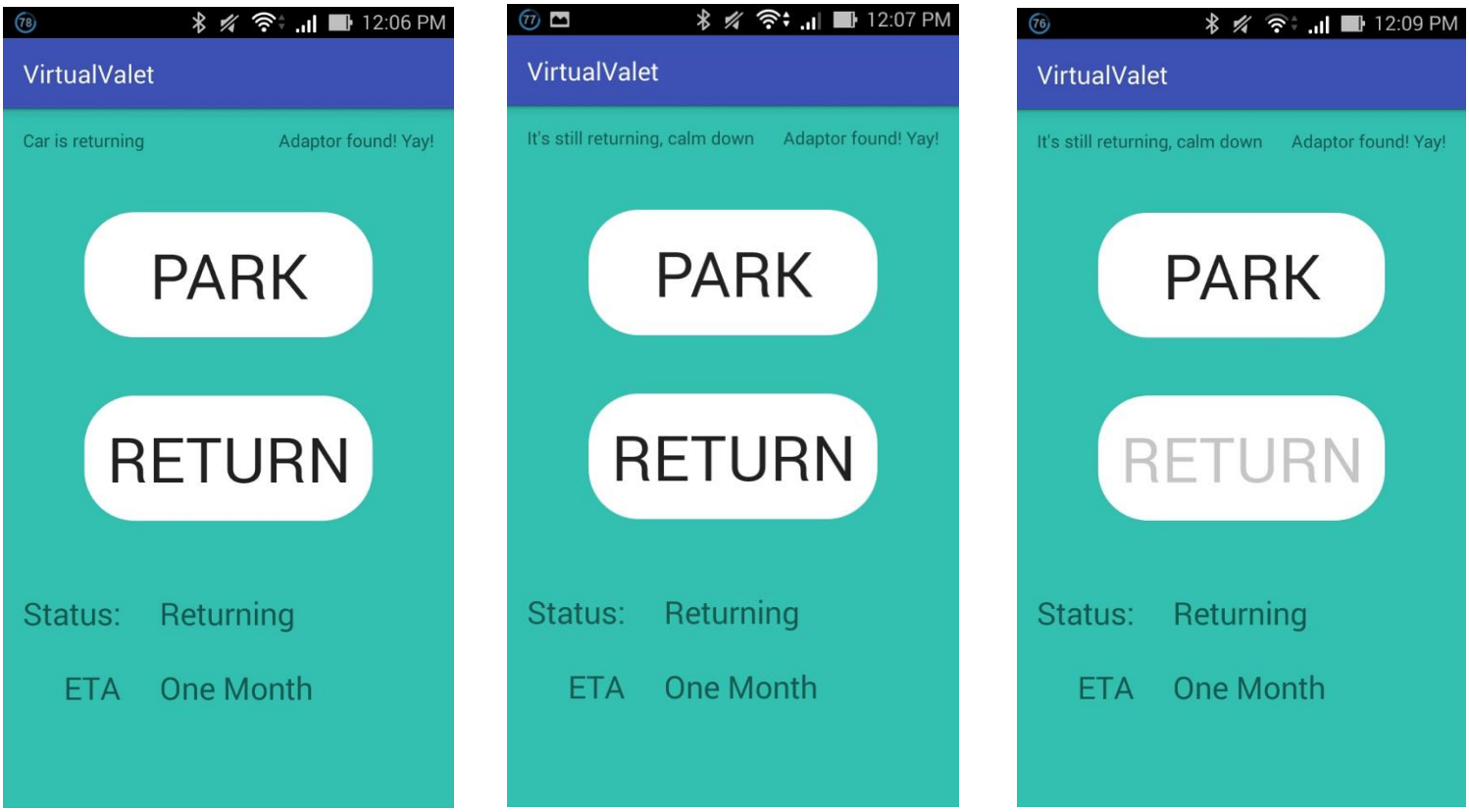

Figure 1. The different states

After successfully testing these functions, we moved on to configuring the Bluetooth API. This can be enable din Android Studio using the android.bluetooth package in the java program:

```
android.bluetooth.BluetoothAdapter;
android.bluetooth.BluetoothDevice;
android.bluetooth.BluetoothServerSocket;
android.bluetooth.BluetoothSocket;
```

and by declaring the Bluetooth permission in the application manifest file:

```
<manifest ... >
  <uses-permission android:name="android.permission.BLUETOOTH" />
  ...
</manifest>
```

We proceeded by writing the code for authenticating a Bluetooth connection on the device and establishing a connection with paired devices. The app now asks the user for permission to turn on Bluetooth and make the phone visible to other devices for a specified amount of time.

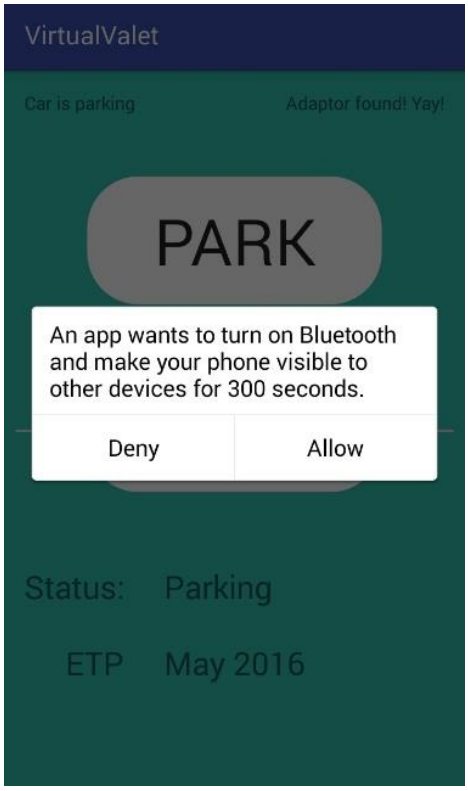

Figure 2. Permission

Thereafter, we decided to test the connection by converting the press button commands to messages in the form of text files to be transmitted via Bluetooth to the laptop.

For this purpose, a new directory was created named 'Notes' on the phone. Inside this folder, a new text file (.txt) is created every time a button is pressed on the app. This text file, called Message.txt is sent

over the Bluetooth connection. The contents on this file can be either ‘Park’ or ‘Return’. No file is created once the button is disabled.
There is another message on the top right corner of the application notifying the user of the status of the Bluetooth connection.

Conclusively, we were able to successfully transfer the commands from the android app to the laptop via Bluetooth.

2. Challenges

Being a beginner in android app development and Java, I had to read a lot to be able to understand the concepts involved. With the limited time available due to continuous deadlines for assignments, it gets really difficult to go out of the way and get familiar with things one is new to. I still have a long way to go in Java programming, so I can continue to work towards improving the app functionality.

While creating a file to send over Bluetooth, we had difficulty understanding how to create the file in the code. After reading up documentation and java file handling, we could handle this by:

```
import java.io.File;
import java.io.FileWriter;
```

3. Teamwork

The other members of the team were working on the following subsystems:

- Pranav Maheshwari: Worked with Mohak on creating a ROS node to receive serial messages via Bluetooth and with Shivam on interfacing Arduino Mega as a node in ROS.
- Shivam Gautam: Worked with Pranav on interfacing Arduino Mega as a node in ROS and with Mohak on Kinect interfacing and depth map/PCL visualization.
- Dorothy Kirlew: Worked on the Android app with me.
- Mohak Bhardwaj: Worked with Shivam on Kinect and with Pranav on creating a ROS node for receiving Bluetooth messages.

As a team, we decided on a mobile platform that best suits our needs for the long term. After considering a number of platforms that would work well with ROS, we narrowed down on the Oculus Prime. It comes with a lot of peripherals that will make integration efficient and comes with a LifeCam camera, which could be useful for our application. It is expected to arrive in two weeks’ time.

My team-work with Dorothy on the app proved very constructive and helped us achieve a lot of functionality in a short time.

Apart from this, I worked with Shivam and Mohak in creating the conceptual design of the Power Distribution Board for our project. We discussed the power needs of our system and decided which of our subsystems need a controlled power supply. We are in the process of deciding components for the schematic and I will soon come up with the schematic design of the board.

4. Plans

For the next Progress Review, the team members intend to:

- Set up and test the mesh network using Digi networking components – Shivam and Pranav
- Integrate the mobile app with the ROS node created to communicate data via Bluetooth – Dorothy, Richa and Mohak
- Show progress regarding obstacle detection algorithm implementation using Kinect – Shivam and Mohak
- Acquire Oculus Prime and integrate it with an SBC – Richa and Pranav

Personally, I intend to continue working on sending serial data from the app to the laptop so that the two separate processes that we've created can be successfully merged. Apart from this, I will try to set up Ubuntu and ROS on the Odroid XU4 along with Pranav. I shall study ROS in detail over the next week to gain the required proficiency as our entire system is going to be ROS-based.