

REV3 (Remote EV3 Control) Application Project Requirements

February 5, 2016

**Bryan Walsh
Team Name: bpwalsh
ECE 573**

1. Project Overview

The REV3 (Remote EV3 Control) application will allow the user to remotely control a Lego EV3 Robot, running the open source, debian based EV3Dev (www.ev3dev.org) operating system, from an Android smart phone. This application will include the ability to control the EV3 via GUI or by use of the phone's accelerometer and gyro sensors.

2. Project Scope

This project will focus on the REV3 Android application itself. It does not include the software to be hosted on the EV3 required to accept and interpret commands being sent from the phone. The messaging protocol specification between REV3 and the EV3 will be developed as part of this project. The protocol will be verified via recordings/playback of the relevant protocol messages. EV3 hosted software will be developed at a later point in time as part of a follow-on effort.

3. Functional Requirements

3.1 Startup and Configuration

3.1.1 Upon startup the REV3 application shall immediately enter a configuration screen. The configuration screen shall allow the user to enter the IP address or hostname and port of the EV3 robot that they wish to connect to. The screen shall also have a connect button that upon clicking will establish connectivity to the EV3 robot. (Difficulty 1)

3.1.2 If the REV3 application loses connectivity to the EV3 the application shall reinitialize and return to the configuration screen. (Difficulty 1)

3.2 Drive Controls

3.2 The REV3 application shall have a “drive” screen from which the user can control of the EV3 robot.

3.2.1 From the “drive” screen the user shall be able to adjust the commanded forward/reverse (+100%/-100%) speed via touch screen controls. The touchscreen commands will be translated to the proper EV3 protocol message necessary to turn or accelerate/decelerate. (Difficulty 2)

3.2.3 From the “drive” screen the user shall be able to adjust the commanded turn rate (+100% turn left/-100% turn right) speed via touch screen controls. The touchscreen commands will be translated to the proper EV3 protocol message necessary to turn or accelerate/decelerate. (Difficulty 1)

3.2.3 The REV3 application shall be capable of receiving the change in heading (since configuration time) over the EV3 protocol. The application shall display the change in heading via an indicator arrow on the “drive” screen. (Difficulty 1)

3.2.3 From the “drive” screen the user shall be able to adjust the commanded speed (+100%/-100%) speed by accelerating the phone forward or backward. The user interaction shall be captured by the phones accelerometer sensor. The accelerometer outputs will be translated to the proper EV3 protocol “set speed” message. (Difficulty 1)

3.2.5 From the “drive” screen the user shall be able to adjust the commanded turn rate (+100% turn left/-100% turn right) speed by turning the phone left or right. The user interaction shall be captured by the phones gyro sensor. The gyro outputs will be translated to the proper EV3 protocol “turn” message. The commanded turn shall be proportional to the quickness of the user's motion. The message will be recorded for requirements verification. (Difficulty 2)

3.3 EV3 Protocol (Overall Difficulty 1)

3.3.1 The EV3 protocol shall contain a message to command the speed of the EV3 by specifying the percent power level. +100% shall be interpreted as full power forward. -100% shall interpreted as full power backwards.

3.3.2 The EV3 protocol shall contain a message to command the turn rate of the EV3. +100% shall correspond to a hard left turn. i-100% shall correspond to a hard right turn.

3.3.3 The EV3 protocol shall contain a message to receive the delta change in heading (since configuration) from the EV3. The delta change in heading shall be expressed in degrees.

5. Functional Requirements Dependencies

Requirement:	Depends On:
3.1.1	None
3.1.2	3.1.1
3.2	3.1.1
3.2.1	3.1.1, 3.2, 3.3.1
3.2.2	3.1.1, 3.2, 3.3.2
3.2.3	3.1.1, 3.2, 3.3.3
3.2.4	3.1.1, 3.2, 3.3.1
3.2.5	3.1.1, 3.2, 3.3.2
3.3.1	None
3.3.2	None
3.3.3	None

6. Critical Path Requirements (B Grade) versus Objective Requirements (A Grade)

The following critical path requirements (B Grade) are necessary for basic functionality of the REV3 application. Basic functionality includes device configuration, speed control via touchscreen, and steering via the phones gyro. Requirements necessary to fulfill this basic functionality include: 3.1.1, 3.2, 3.2.1, 3.2.5, 3.3.1, and 3.3.2.

The following objective requirements (A Grade) are required to meet the objective system functionality. This additional requirements would give the user fault tolerance due to lost network connectivity, the ability to steer via touch controls, display of delta heading information, and control of speed via use of the phone's accelerometer. Objective requirements include: 3.1.2, 3.2.3, 3.2.4 and 3.3.3.

7. Non Functional Requirements

7.1 REV3 application response time between user interaction and the generation of the command to be transmitted by the EV3 protocol command shall be within 50 milliseconds.

7.2 The REV3 application shall be targeted for the Android Marshmallow 6.0 operating system.

7.3 The REV3 application shall be written using java 1.7.

7.4 The EV3 Protocol TCP/IP stack shall not be implemented for this project. For verification purposes a file-based testdriver will be used to stimulate necessary inputs and outputs across the EV3 protocol. All transmitted message will be recorded for verification purposes. Received messages will be stimulated by prerecorded files.