CPE656TL-72 Created by Corey 21 Oct 2015 14:44 PeerReview

Review Project Management Document Initial Draft

Here is basically the document for tracking the risks for the project.
I have populated it with the risks that I have to date. It should be in sync with the Operation Document Review that I sent out earlier.

Priority: **Normal**
Type: **Task**
State: **Fixed**
Assignee: **Corey**
Subsystem: **ProjectProgress**
Fix versions: **Unscheduled**
Affected versions: **Unknown**
Fixed in build: **Next Build**
Estimation: **?**

Updated by Corey 03 Nov 2015 14:25

Comments (8) History

Rashad — 26 Oct 2015, 00:23

Looks good only thing else i can think of is isn't there a risk asssociated with the uncertainities wtih the hardware? For example the IMU's we are purchasing are built with a G force range typically intended for objects with much higher G forces not train testbeds. Isn't there risk associated with how reliable it is going to be able to get data from there?

THe other risk I gueess would be with what language we use, if we decide to use Java what's the learning curve and how easy will it to build a C++/Java interface between the system components that require C (sensor collecting) and the rest that we are using for Java?

Corey — 26 Oct 2015, 14:14

Thanks. You brought up a good point. The hardware is a risk, so I will be sure to mention that one. Collecting data early is about the only way I know of that we can mitigate the risk. I don't have using Java as a risk, since 2/3 team members already are familiar with it. Also, there isn't a need for interoperating between C++ and Java since the components that a single language is being used for each module, and there's a communication interface being used between them. Specifically about sensor collecting, the measurements are coming across using Zigbee or some other wireless network, which will then come in likely as a network packet or serial port traffic. So the Java code will never interoperate with the C code directly as the C code is what will be generating the traffic.

Corey — 26 Oct 2015, 14:25

Attached is the revised version that now includes the risk of acceleration measurements.

Rashad — 30 Oct 2015, 02:56

Ok so we are going to communicate with each component via some socket like communication or something? Then just sending the data that through it? For example the interface between the Navigation Library which I assume is the one getting and computing the location information through the RFID tag etc. Say The Navigation Library would be in C++. If it needs to send information such as current position speed, etc it can send that through the socket to the database and the GUI that may both be coded in Java? Not certain that socket would be our communication means or that the nav library is in C++ just trying to use an example to make sure I understand what you are saying.

Rashad — 30 Oct 2015, 02:58

Reviewed and Accepted Changes.

Corey — 01 Nov 2015, 01:31

The GUI I figured would be the same language as the library. It would habe its own api to talk through. We would use the api of some communication library to talk to the motion detection unit. It could be sockets oe something else. A builtin api for a database driver could also be used. It would likely use SQL.

Stephen — 01 Nov 2015, 22:32

Reviewed:
Looks good. But on a formatting note, might want to try and re-size the table in 2.2 so that the titles and some of the entries don't word wrap part way through the word.

Corey — 03 Nov 2015, 14:24

The formatting will be addressed in CPE656TL-80.