

MCTX3420 Team 4: Progress Report #3 (Summary)

Sam Moore, Rowan Heinrich, Callum Schofield, James Rosher, Justin Kruger, Jeremy Tan

Sam:

- 1. Multithreaded framework for server side software**
 - a. Using dummy functions to simulate sensor reads
 - b. Using dummy function to simulate request from remote client
- 2. Collaboration with other teams**
 - a. Combined group meeting 2013-08-14
 - b. Started work on list of tasks for the project from the point of view of the software team
- 3. Experiment with data transfer storage & transfer**
 - a. Binary file is fastest, and very simple to use for transferring data between threads
 - b. CSV is human readable, but difficult to transfer data between threads
 - c. Database (sqlite) is extremely slow. Better than CSV for data transfer.

Rowan:

- 1. Arduino coding**
 - a. Control servo position using servo libraries
 - b. Communicate between Arduino and RPi using wire library
- 2. Consider GUI Layout requirements**
 - a. What operations the user will need
 - b. What devices the GUI will need to control

Callum:

- 1. Image Processing in OpenCV**
 - a. Create block diagram
 - b. Pseudo-code for process of capturing image with camera
 - c. Work on installing OpenCV under Ubuntu for testing

Justin:

- 1. Consider safety requirements for system**
 - a. Software control: argument checks, error handling, visual/GUI safety, data safety, etc.
 - b. Security: single authorised login at a time, limit system operation via safety signals.
 - c. Hardware failsafes also needed. If software fails, relinquish control to hardware.
 - d. Created features checklist and flowchart to coordinate software team.

James:

- 1. Importing jQuery library into html files**
- 2. Calling JavaScript files from an external source**
 - a. Calling and integrating external JavaScript code into the html file at the run time
 - b. Running the JavaScript and producing an visible output
- 3. Identified key components for the GUI and Prioritised**
 - a. Identified possible inclusions to the GUI
 - b. Ranked inclusions in terms of Priority to the user

Jeremy:

- 1. Consider database with PHP approach**
- 2. Design and begin implementation of server side HTTP API**
 - a. Now using FastCGI program, running under nginx, instead of custom HTTP server
 - b. Using dummy functions to represent sensors
 - c. Work on secure logon / authentication with nginx

TODO:

- 1. Finish list of tasks and timeline; get agreement with other teams**
- 2. Get agreement from sensors/software/electronics teams on server/microcontroller hardware**
 - a. Beaglebone or RPi
 - b. ADC/DAC modules available for Beaglebone/RPi or Arduino with USB communication
- 3. Start to combine software into a single system instead of using separate test programs**
 - a. Get OpenCV working; transfer images to basic GUI
 - b. Look into controlling ADCs and digital I/O on the Beaglebone
 - c. Merge dummy function in Sam's code with sensor handler in Jeremy's code
 - d. Have basic jQuery GUI able to query the server's HTTP API