**Capstone Project: Protobuf viewer**

**Brainstorm:**

How to parse in code: With java you can pass in compiled code and show the GUI framework

How about C++? With C++ we can use Qt (available at <http://qt.nokia.com/products/>). With Java we can use swing, swt, etc.

Message Types: Need to handle whole numbers (integers and longs), strings, enums, and recursion (messages within messages).

Ex: message{ id, message{first name, last name}}.

Version Control tool: We will be using Github, which is open-souce and available through the net to control software changes. A link for our Github repository will be made available in the documentation.

Build Environment: Need to get the necessary libraries (Protocol libraries, QT Gui libraries) into the environment as well as to have a list of protocol buffers, utilizing reflection. The libraries and tools we need should be gathered first before development begins.

Plan on having (tentative) releases every Friday (documentation, software milestones, prototypes, etc) with status meetings either in person on through online conference on Sunday at 2 pm. This will begin, tentatively, Juy 15th.

**Preliminary Schedule:**

Weekly Status meetings: Sunday, beginning July 15th.

July-August:

Mainly used to create documentation:

Proposal:

The project and project details (google protocol buffers and viewer).

This could be taken from the email sent to Sharba

Reasoning for project, motivation from the L245 program, would be good to get some input from manager, schedule of development, testing, and documentation. Also need to get a list of references (protocol buffers, library reference, etc).

Requirements:

Build environment, what the project must accomplishment (languages and message types to support), additional section for design goals (such as the proto converter).

Tools:

Spreadsheet of tools, such as Git/Github and how we’re gonna use them, as well as why.

Software Design Document:

Used to describe the development cycle of choice (Agile like development), the architectural pattern of choice (Model-View Controller) and any other details. I propose creating this document and filling in the details during development, using the first few pages to describe agile development and model-view-controllers.

September-November:

Development and Testing- Each prototype release will be checked for testability and if so, will be tested.

As stated above: Each week Friday will have a prototype release.