**Chapter 4: Project Plan**

Like any project, careful planning and organization is paramount to the success of the project. More importantly however, is the methodical execution of the plan. Although we originally developed a roadmap for success as well as implemented a number of project management systems, we did not follow the plan as intended. This section outlines our proposed plans for making ChemLAB happen and the actual process that we went through.

**/subsection Proposed Plan:**

We had originally planned to use the waterfall model in our software development process in which we would first develop a design for our language, followed by implementation, and finally testing. The idea was for all team members to dedicate complete focus to each stage in the project. Especially since we only had three members on our team, our roles were not as distinct and everyone had the chance to work, at least in some capacity, in all the roles. We intended to meet consistently each week on for at least two hours. During our meetings, each member was suppose to give an update about what he or she had been working on the past week as well as plans for the upcoming week and any challenges he or she faced that required the attention of the rest of the group. To help facilitate communication and the planning of meetings, we used Doodle to vote on what times were best for meetings. Also, in order to improve team dynamics, we planned to meet at least once every two weeks outside the context of school in order to hang out and have fun. Development would occur mostly on Mac OS and Windows 7, using the latest versions of OCaml, Ocamllex, and OCamlyacc for the compiler. We used Github for version control and makefiles to ease the work of compiling and testing code. The project timeline that we had laid out at the beginning was as follows:

Sept 24th: Proposal Due Date

Oct 2nd: ChemLAB syntax roughly decided upon

Oct 23th: Scanner/Parser/AST unambiguous and working

Oct 27th: LRM Due Date

Nov 9th: Architectural design finalized

Dec 5th: Compile works, all tests passed

Dec 12th: Project report and slides completed

Dec 17th: Final Project Due Date

**/subsection What Actually Happened:**



This graph was pulled from Github reflecting the number of commits being made over the span of this semester. Due to schedule conflicts and a false sense of security, we did not start intensely working on the project until after Thanksgiving break. Since we did not coordinate the development of the Scanner, AST, and parser with the writing of the LRM, our language did not have as concrete a structure as we had hoped. Furthermore, we did not have enough time to implement some of the features in our language such as object-orientation or more built-in functions. As we were developing the software, we did make sure to allow testing at all steps in the design process. In the test script, we had identifiers for how far in the compilation process we wanted the program to run. Thus, we were able to maintain testing capabilities even before all of our code was ready. We discuss the testing procedure in more detail in a subsequent section.

**/subsection Team Responsibilities**

This subsection describes the contributions made by each team member:

Project Proposal - Gabriel L/Alice C/Martin O

Scanner - Gabriel L

AST - Alice C/Gabriel L

Parser - Alice C

LRM - Gabriel L

Code Generation - Alice C

Semantic Analyzer -Gabriel L/Martin O

Testing - Martin O

Final Report - Gabriel L/Martin O

**/subsection Project Log**

*We add later*