Outline of Bug Fix Project
Peter C Rigby

I expect students doing both a research project and bug fix project to fix 1 to 2 bugs and people doing just bug fixes to fix 3 to 4 bugs. The difficulty of the bug and the number of people on your team will also be considered. Search for domains that interest you (eg “day planner”) on sourceforge, github, and OpenHatch.

Milestones and dates

1. (September 27) Describe why you think the project is interesting (Fill in the new version of ProjectNameMilestone1.doc)
   1. Provide a general overview of the project and why it interests you (approx one paragraph)
   2. What is the maturity of the project (e.g., alpha, stable)? How old is the project? How many developers are listed as working on the project? What is the size of the project?
   3. For each group member, describe the experience and skills that he or she will contribute and try and relate this to the chosen project.
   4. Provide a link to a potential bug(s) or feature(s) that you feel that you should be able to fix
      1. in a paragraph describe why you feel this should be an appropriate amount of work
2. (September 29) Contact the developer and subscribe to the developer's mailing list
   1. It is required that you contact one of the developers on the project. Explain your situation. For example: We are masters computer science students who are required to fix a bug and/or implement a feature in an OSS project. Is there any small bugs or features that you feel we would be able to implement. We have looked over your documentation and system architecture, we are considering the following bugs or features. Can you provide us with any information that you think may be helpful to us?
   2. The core developers are often very busy, make sure you subscribe to the developer's mailing list and any other lists you think may be useful! If you get ignored by the core developer, try posting to the list.
   3. Provide greater detail and links to the bugs or features you plan to implement
3. (October 4) Provide a preliminary plan with completion dates and start working. The events must be in a google calendar that you share with rigbypc@gmail.com I will hold you to your plan, so try to be realistic.
   Example events include:
   1. understand system architecture
   2. understand API X related to bug #Y
   3. describe why bug #Y exists (the evolution that lead to the bug)
   4. write change log and create patchset (iterate)
   5. submit the patch and implement any fixes that the project's developers may require
4. At any stage you can get feedback from me, but once you are finished implementing a change, you must submit it to me for review, and then submit it to the project. If you schedule slips, you need to justify the slip. For example, bug X was unexpectedly coupled to class Y and Z, which added additional work (describe the work) ...
5. As the term progresses, you will summarize your experience and the practices of the project. See ProjectNameExperiencePractices.doc