**WEB CRAWLERMAN**

**By Team Amurica**

**(Ben Nelson, Tucker McKnight, Carlos Brenneisen, Scott Stromberg)**

**Executive Summary**

(Summarize who you are, the date, and the design purpose of and requirements for the project. Summarize how well your code to met these requirements.)

**Technical Summary and Conclusion**

(Detail how well the goals were met and justify how you know this. Additionally list any goals that **were not realized.** You should create a table to summarize important criteria (such as computed results and time required for the program to run) computed over several domains.

Charts and figures would be especially appropriate here. )

**Extras**

(You will be required to add some extras of your own choice to the project. You should describe them here, summarize what they tell you about the web pages, describe the technical challenge in computing your extra, and describe why you chose to implement each extra.)

**Memory Management and Efficiency**

(Include a section on memory management and algorithm efficiency. You should describe how much memory your program used and how efficient it was at visiting and analyzing web pages. Please speak in terms of "web pages processed per second". Additionally, discuss your use of threads and what effect this has (non-threaded vs threaded runs).

Charts and figures would be especially appropriate here. )

**Testing and Verification**

(Discuss how well tested, and in what manner, the project was evaluated and verified to do the correct job.)

**Code Reuse**

(If you modified any of your past code for use in the project, you must describe (in high level detail) what changes were made and why. You should also discuss how you discovered the weakness of the previous version of the code.)

**Person Hours**

(Describe how long the project took (person hours for the team) and comment on how efficiently your team used this time.)

**Team**

(First, both in prose and with a tabular summary, describe how well each member of the team completed their assigned section of the project (assigned during the design document). If for some reason a team member did not do their job, list why not and if appropriate who "picked up the slack".

Discuss Teamwork and the teams "reflections". How well did the team work together and **what changes need to be made in the future?** This section should include results from the post project recap.

Also make sure to discuss and write up how well your team communicated with each other as a group. )

Meeting Overviews

Planning Meeting:

Our first meeting as a group was to plan out the project, review on the last project and for each member to make suggestions as to how to better our group performance for the web crawler project. During this productive meeting session we had each member come up with ideas for possible classes and code that would be used in the project as well as other development ideas to make the project go smoother. After hammering out what we believed was necessary for the project in terms of classes we focused more on use cases for the project and drew up some rudimentary ones on the white boards in the CADE lab (which is where we met for all our group meetings). The photographs of the use cases are included in Appendix A along with the team notes that were taken by Tucker (who was designated as scribe for this meeting).

The team decided to move from the Google Wave software that was used in the previous project for team management to Microsoft Groove due to the fact that Google Wave would not be continued into 2011 as stated by the Wave project and the Groove software was freely available to all engineering students, as well as the Groove software being more powerful than the simple Wave web interface. However, this proved to add a bit of a challenge when working across different operating systems and the migration process was not as smooth as was hoped (it took a little bit of time for everyone to become acclimated to the software and its use). Additionally, during this meeting it was decided on by the group to use CPPUnit to try and do unit testing on the project and each member was given an assignment of code to be worked on before the next meeting as some members had more time than others outside of meeting and would be more productive during those times. These individual homeworks would be completed by each member and brought to the next meeting and the harder parts of the project in addition to problems with the code would be worked on by the group as a whole. The project would be worked on by teams of two during full group meetings where one person would test the code and one person would try and debug and complete coding. All of this was then recorded and put in the design document that was handed in.

Working Meetings:

During the work meetings the teams met for about fifteen minutes at the beginning of the meeting to discuss the progress made at the last meeting and to review what still needed completion. A To-Do list would be compiled by the group members and then posted to the issue tracker on the Groove as well as on the Wave so all members could access it at all times. After this the team would break off into their respective extreme programming groups and continue working on their projects. The meetings would last 3 hours, from 6pm to 9pm each time on Tuesdays and Thursdays each week as well as on Presidents Day for major code finishing. The Presidents day meeting lasted from approximately 10am until 5PM and resulted in much of the code being completed. These meetings differed from what was presented in the design document because of schedule conflicts that arose over the course of the project with several members and also due to illness (a severe case of the flu) hitting both Ben and Carlos over the course of the project.

Review Meeting:

The review meeting took place on the last day of coding and the members reviewed what they believed were successes in the project and what they believed required more work. Each person vented what they believed needed to change and what they liked about the project and in addition they put input for whatever special things needed input in the final report.

Software Successes

Classes:

(include changes)

Collaborative Effort:

Extras:

Final Code Presented:

Web Crawling Statistics

(add pertinent graphs/stats for at least 3 domains here)

Software Limitations

Threading:

Extras:

Unit Testing:

Problems in Development

(1/2 page)

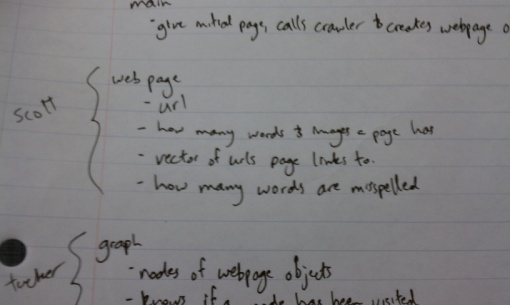
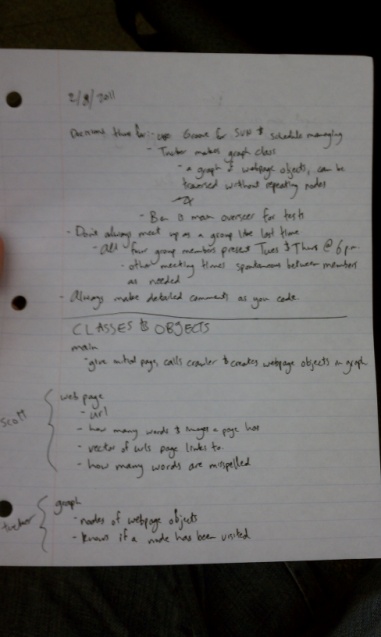
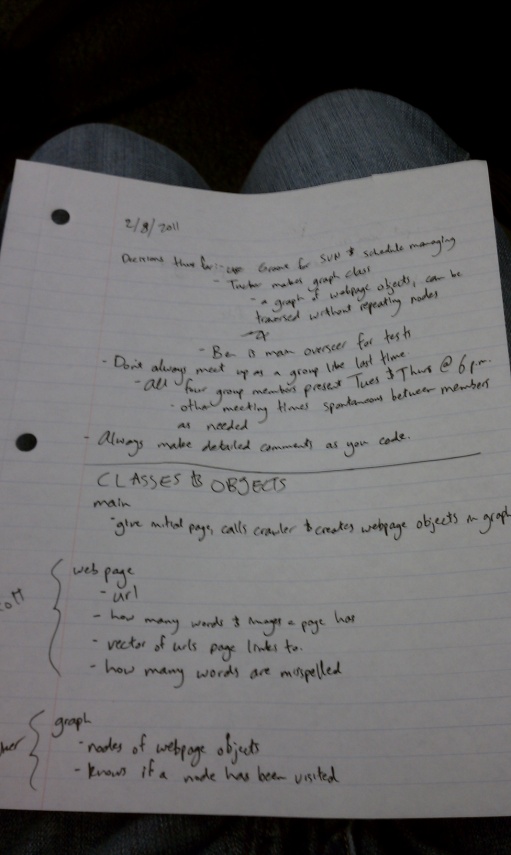
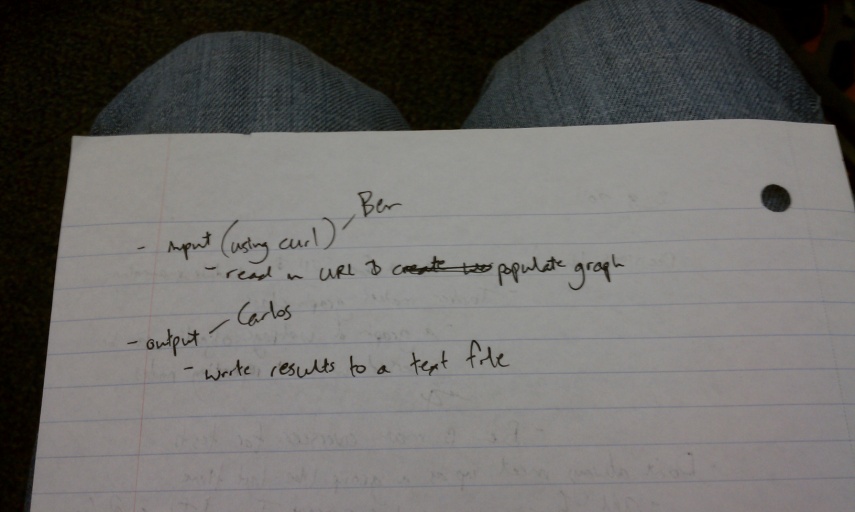
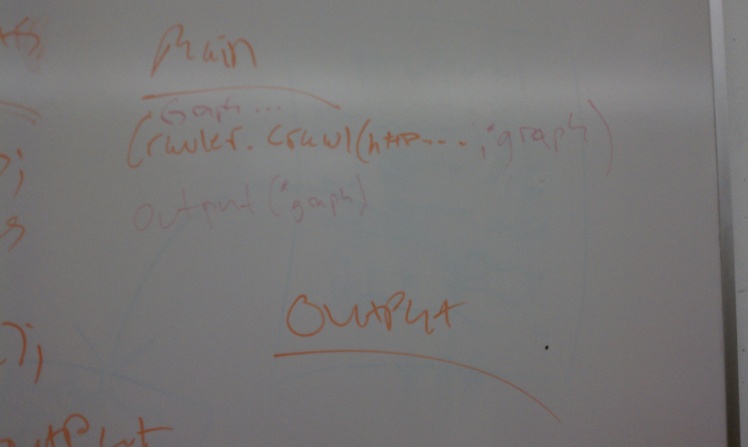
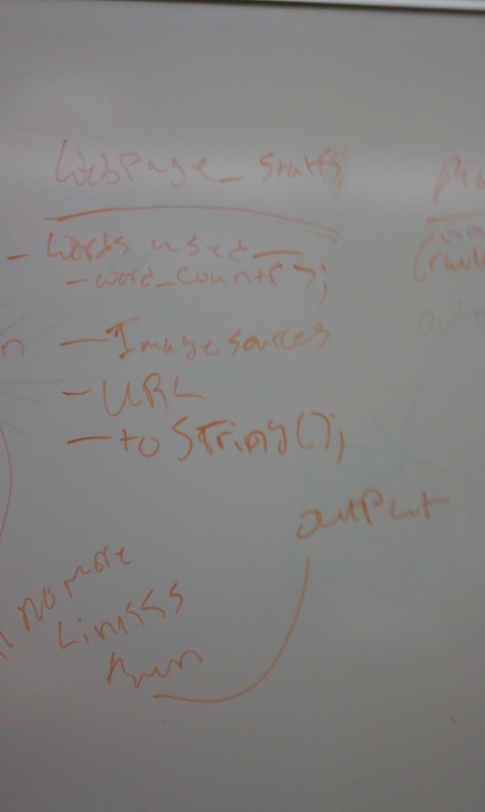
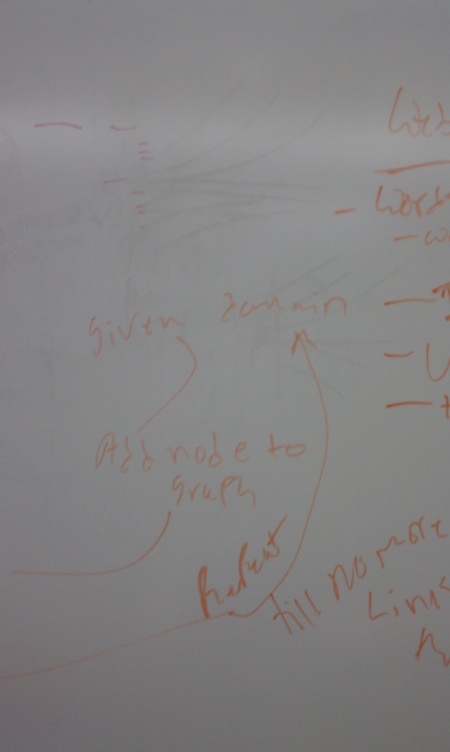
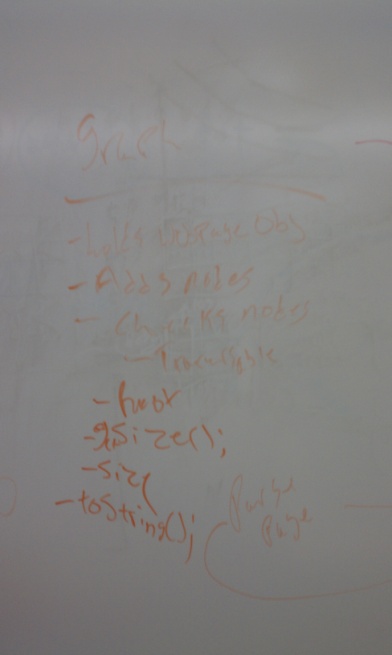
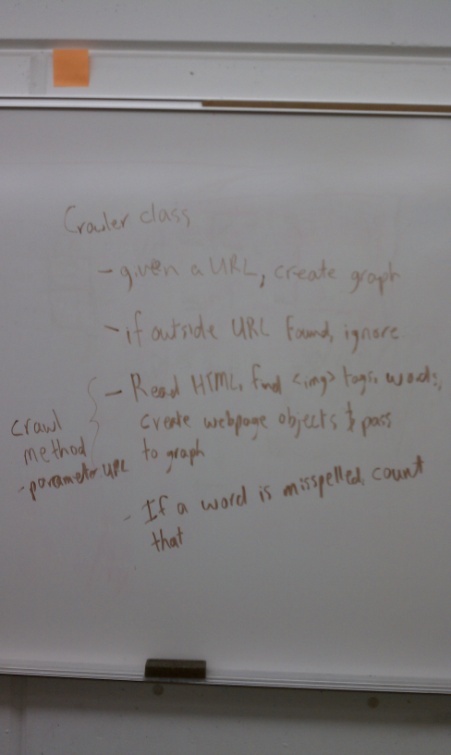
Successes in Development

(1/2 page)

Plans for Future Development

(1/2 page)

APPENDIX A (Use case photographs)



APPENDIX 1 – Example Results