http://users.csc.calpoly.edu/~csturner/courses/308w09/PostMortemExamples.html

**Ryan Management:** Division of labor, Team meetings / minutes, risk mitigation, design layout too late, testing, adapting to change, sage process

* **Quality & Risk Plan**

Our team met a lot, twice a week and for 2 or sometimes 3 hours a meeting on work days. We kept on task and really dealt with any troubles each other were having. These meeting allowed us to make sure we were all on task and to have hard deadlines for eachothers work. However we did not take very good minutes on these meetings. The meetings tended to stay on task but no record of what we talked about just a general idea of what we did and the time we spent on it for the time log.

Risk mitigation was an issue the team did fairly well on. The team did run into a risk and mitigated it well. We did not get a server by our deadline so the team decided to develop using XAMPP to simulate the server allowing us to port over to a real server if one ever became available and if one did not the client could use XAMPP and have all the functionality we proposed. Our other risk we reduced the likely hood of occurrence was the lack of mobile development. We got started on this risk before the end of the semester although we didn’t initially plan on dealing with it until next semester. The team used PhoneGap and decided that we were confident in our ability to produce an app with our current knowledge.

Division of labor was a problem in our management. The work didnt seem to be evenly divided amongst group members because we did not pull enough work from the backlog. The team should have been working on the back end of the web interface for more sprints than we did to give more work to other members of the group.

The team also messed up in its scheduling of tasks. Specifically the team did not do an official design of the interface template until it had already been started. We initially did this to get a feel of what could be done but in reflection that was a poor choice.

The team Did not do much testing this semester. The only testing done was that of the database and that was more so just seeing if the data was in the database not rigorosly testing the database. This will certainly be changed next semester as next semester is focused on quality, since most of the core components are close to finished.

The team did adapt to change quite well. The team did not initially have any mobile development in the plans for this exit strategy but talking with the Oversight we added mobile development training and exploration to the exit strategy. The team fit that into our goals for the end of the semester and completed that goal. The team also had to develop without a server, the team did not plan on the client not providing a server so late so the team adjusted and kept plugging along.

The SAGE process was followed by the team with the bi-weekly scrum meetings and the planning of every sprint based off what backlog items were deemed more important by the team members and which were completed last sprint. Status reports were made after each sprint and given to the Oversight with a few typo's but accurate none-the-less. Also the team set our CS425 exit strategy to reduce the likely hood of our 2 biggest risks.

**Mike Communication:** inner and external communication, oversight communication, team collaboration, server

**Nate Tools:** Hangouts, Dropbox, Bitbucket, PhoneGap, boot-strap, xammp

**Mike CS 425 Exit Strategy:** Changing it, thoughts on how it went (did we meet it, should we have more), was it too easy?

**Matt Progress:**

While we were successful in getting a lot of work done including the completion of our 425 exit strategy we still have a way to go in order to fully complete the application. Currently we have implemented a fully functioning database, the general layout of each HTML page using boot-strap and accompanying CSS, PHP functionality to interact with the database, and a mobile app prototype using PhoneGap.

Generally things went well during this semester such as our progress in research and the team’s ability to meet deadlines on time. There were several topics that the team was somewhat unfamiliar with, mainly mobile development, but also things such as boot-strap on web pages and database implementation strategies. We managed to find time in our sprints to research these topics while still completing all of the goals we had assigned for each team member. The deadlines that the team set either for what needs to be completed at the end of a sprint or during a sprint were almost always met on time. But there are a few things that we could have done differently that would’ve made the implementation of this application run smoother. Especially early on in the process, during our first couple of sprints, we had poor time estimations. These estimations improved slightly as we worked into later sprints, but still need a lot of improvement for next semester. To remedy this issue we plan on spending a little more time at the beginning of each sprint to set out well defined goals. The time needed to complete these goals will be discussed as a group to get the most accurate estimate. Another major issue we had was spending too much time working on the documentation. We didn’t divide the documents up, and tended to work on them all together as a group. This caused them to take much longer than they actually needed to. Towards the final sprints we were more successful in splitting them between group members and we plan on using his strategy for all future sprints.

~~Where we are now~~, how we would’ve done things differently, ~~estimations~~, ~~meeting deadlines~~, ~~research~~, database design, ~~too much time on documentation~~

**Nate Customer representative report:**