**NAU-CS Team Project Self-Reflection Worksheet**

**Team Name:** Team Panda

**Team members:** Kimberly Oyama, Daren Rodhouse, Blayne Kennedy, Chihiro Sasaki

**Course number and name:**

CS486C Senior Capstone Design

**Semester:** Spring 2013 **Date this reflection completed:** 04/30/2013

## Software DESIGN PROCESS

**How did your team structure** the software development process? Did you choose a particular formal model (SCRUM, Agile, etc.). If so, which one and why? If not, did you explicitly agree on an informal process…or was it just pretty random. Explain briefly.

The software development process we used was an Agile methodology that was very similar to the Scrum framework. The main difference between our process and Scrum was that we did not have a Scrum master. Instead, we decided as a group how to prioritize our tasks each week.

Our sprints were two to three days long and took place between team meetings. Meetings were held three times a week, Tuesdays, Thursdays, and Sundays, where specific tasks were assigned to each person to complete by the next meeting. We came up with a list of tasks at the beginning of the semester; we modified this list and re-prioritized the tasks as we developed.

**How did it go?** Now briefly discuss how satisfied you were with this process. Did it work well for this project? Why or why not?

The process worked well for our (project). It worked well because we had a small team and it was easy to modularize our project. We were able to have our applications continuously evolve without much backtracking, and complete each deliverable for the class. However, we should have made our design specification document a priority before we got too far into prototyping.

**What changes might you make** in your development process if you have it to do again? More structure? Less? Different process model?

We would make the development process more structured, especially if the design specification document was made beforehand. Because we did not a strict plan to follow for our modules, we added on features to the apps each week. This worked fairly well for this project, but if we were to redo the project, we would have a stricter plan to reduce the amount of coding and recoding that went into the finished product.

## Software DEVELOPMENT TOOLS

**What software tools or aids**, if any, did your team members use to support or organize software development? For each of the following categories, list the tool(s) used, and briefly describe how the tool was actually used. If you didn’t use a formal tool, explain how you handled the matter with informal means.

* Source creation tools: IDEs, text editors, plugins, anything used to edit/create source.
  + Eclipse (Java and XML)
    - ADT Plugin - for Android development
  + xCode (Objective-C)
* Version control: How did you manage your codebase?
  + Git, Github
* Bug tracking: How did you keep track of bugs, who was working on them, and their status
  + YouTrack
* UML modelers and other miscellaneous tools:
  + Doxygen
  + Microsoft Word
  + Google Drive
  + Dropbox

**How did it go?**  Comment on any problems or issues related to organizing the coding process. How might you have managed this better? Were some tools you used superfluous or overkill? What tools or mechanisms would you try next time to deal with those issues better?

The organization of the coding process worked well for the Android app development. There were a few merges that were problematic, but two people could work on the different apps (customer or merchant) at the same time.

There was a problem with version control on the iPhone apps. When using git and making commits, there were permission issues when two or more people would both commit and then make a pull for any work. Thus, the two developers working on the iPhone apps ended up using xCode’s own version control and Dropbox.

During the later parts of the semester, we hardly used YouTrack to track all of our bugs. We manually wrote down bugs in our notebooks and recorded changes in the commit messages in GitHub.

## TEAMING and PROJECT MANAGEMENT

Without getting caught up in detailed problems or individual blame, take a moment to think about how your team dynamics worked overall. Here are a few questions to guide you:

**How did you organize your team?**  Did you have some clear distribution of team roles (leader, technical lead, documentation lead, etc.) up front? Or was it more just “everyone does everything as needed”?

We distributed different roles to members of the team at the beginning of the semester, as seen below. However, we did not really stick to the specific roles as we started development in the second semester.

We originally planned the following roles:

* Kimi - team leader and client communicator
* Daren - recorder
* Blayne - architect
* Chihiro - release manager

At the end of the semester, the only roles that stuck were Kimi’s team leader and client communicator roles. The other team members all became full-time developers.

**How did you communicate within the team?** Comment on each of the following communication mechanisms:

* Regular team meetings? If so, how often?

We had regular team meetings three times a week. We met every Tuesday at 1:00 P.M., Thursday at 2:20 P.M., and Sunday at 1:00 P.M. Each meeting was 2 - 4 hours long.

Additionally, we had a meeting with our mentor, Dr. Georgas, every Tuesday at 10:50 A.M. to go over our progress and upcoming deliverables.

* Impromptu team meetings? If so, roughly what percent of total team meetings were of this sort?

We did not have any impromptu meetings for our team, but we had semi-impromptu meetings with our client that were scheduled a day or two in advance.

* Emails to all members? If so, explain briefly: about how often, what used for?

Kimi, our team leader would send out emails to remind team members of our tasks due by the next meeting. She would also send out emails to coordinate meetings with our project sponsor.

* Software tools? Were any of the software tools you mentioned above (e.g. bug/issue tracking) using to communicate and organize tasks, e.g., in lieu of emails or other discussion?

We used Google Drive to create first drafts of all documents. After edits have been made to documents, they have been exported to Word Documents and further formatting changes have been made.

* Other communication channels used? Facebook, wiki, text messages, phone conferences, etc.

Kimi would send out group texts to communicate any changes in our meeting plans.

**How did it go?** Did you feel that intra-team communication overall went well? Were there breakdowns, e.g., where someone didn’t know something was due, didn’t realize a task had been assigned to him/her, did not know about a deadline, etc.? Without getting into details, simply comment on whether such breakdowns occurred, what the overall cause was, and how serious (if at all) the consequences were.

Team communication went extremely well within our group. There were no deadlines or meetings missed due to a breakdown in communication. Every time someone was late, they would notify the others by texting them.

**What could you do better?** More structured leadership? A more formal task assignment/tracking system? Using better/other communication mechanisms? Generally just think about what you all would do next time to improve communication and avoid breakdowns mentioned.

We could have spent more time on the overall project schedule and planning out the assignments due dates to better reflect our gantt chart. This was just due to the team members schedules and an optimistic planning early on in development. Also, we should have worked on our design specification document before getting too far into our prototypes. A formal design specification document would have made it easier to know which application modules still needed implementing.