Doug Richardson

Project Management report on Project Kintsugi

# Project Overview

Project Kintsugi involves improving the Common Weaknesses Enumeration (CWE) web site. The current web site has poorly implemented search engine that only allows a given CWE to be searched for by ID number. This information is something knowable only to people who already know what CWE to look for. This severely curtails the potential use and value of a database such as this. Our project entails making a web application with a superior search engine.

# Overall strategy

I see the development of this system done in the following phases:  
  
  
-Phase 1 (August-September):  
--Team being formed, project being decided  
--Research and group consensus on tools and techniques chosen  
--Drafting of requirements  
--Conceptualization and visual representation of physical and logical architecture  
--Conceptualization and visual representation of database schema  
--Begin development of the “skeleton” of the web site with means chosen by group consensus

-Phase 2 (October):  
--Creation of database  
--Populating the database (either using or creating a mechanism to convert the XML data)  
--Completion of the web site meant for users to access the web site  
--Creation of a rudimentary search engine that has greater functionality then the current model.  
--All absolute (“must have”) requirements of the security sub-project will be completed

-Phase 3 (November):  
--Creation of a superior search algorithm   
---Creation of a search algorithm that can do a full text search of CWEs  
---Creation of an accompanying prioritization algorithm to give precedence to finding results in more relevant fields  
---Creation (or implementation of an already existing) text mining tool to purge common terms (“a”, “the”, etc.) from the search.  
--Creation of a “breadcrumb trail” so that one can trace their navigation path   
--Completion of all high priority non-absolute (“should have”) requirements of the security sub-project THAT DO NOT REQUIRE ADMIN MODE (admin mode is a “C” task for the main project).

Should the project go ahead of schedule or should developers not have any core tasks to complete (i.e. because of dependency bottlenecks that only one or two team members can work on without diminishing results), mid-priority tasks will be worked on. This will be a case-by-case basis, and will be dropped as soon as a more important task is available unless the task can be finished reasonably quickly from that point.

# Team

(I feel weird writing this “behind the team’s back” because they are inevitably going to read this, but this is supposed to encapsulate my thoughts in terms of the team so far, untainted by fear of observation).

Apart from me, the team consists of Jon von Kampen, Chris Fibich, and Nick Watkins.

Jon seems to have the greatest understanding of the project of all the team members. His experience with information assurance and his contacts in the NUCIA faculty made this project possible. He and I agreed that a project that prioritizes web security would be ideal for our post-academic goals. He has a strong sense of what is needed and how it’s going to be accomplished. Also, of all of the team members (apart from myself), he has the greatest sense of what role he wants to play in the development of this project. Also, he and I have a pre-existing friendship dating back to 2009, and have collaborated on academic work before.

Chris has made his specialties known early on. He has expressed an interest in being assigned the visual tasks if possible. This “visual imagination” and artistic skill has proven valuable for the conceptualization phase. I don’t know what he will want to do once the conceptualization phase is done. I am thinking of suggesting web-application development to him after conceptualization is over, since it’s the most visually oriented. If we get ahead of schedule and wish to start the visualization of the CWE sub-tasks, I might suggest he is put in charge of graphics. I do not know his preferences apart from visual tasks. However, I believe that will come to light through the tasks he chooses to do.

(I am only “suggesting” these things because we assign ourselves tasks, so long as we are consistently contributing to the project and have something to do).

Nick remains a mystery. Has shown drive and works hard, and he obviously knows what he’s doing. However, he hasn’t shown any active preference in any task yet. This may change once phase 1 is done and we are more actively coding. I don’t want to relegate any group member to “everyone else has stronger preferences, and this is what’s left” if at all possible. However, I do not feel lacking preferences is a character fault. It may be that he prefers it this way (I have been on a few projects where I was the same way), and that is perfectly fine.

# Problems

## Time and availability constraints

The most consistent problem our team has is extreme scheduling constraints. All team members (except for me) have an outside job. Also, all team members have other classes with other work requirements. For each team member (including me), at least one of the classes being taken apart from the capstone course has a relatively heavy work load or otherwise time-consuming style of progress. This problem is so prevalent that it’s factored into all assumptions and estimations about project delivery time. I predict this problem will get worse as the semester progresses and external work load increases. My ability to abate this as Scrum Master is limited by the terms of the UNO code of conduct (I can’t do their work for them; but if it’s a class I have completed, I can help them).

These scheduling constraints are starkly different then my previous project I did in Norway, where each group member only had one other class which was designed accompany the bachelor project (and thus, provide a relatively light amount of extra work). Also all working group members only had to work on Friday. We also had four months to complete the project instead of two and a half.

My experience in Software Engineering has showed me that scheduling conflicts and additional outside work are two of the most significant threats to a given project. Our group did decently well on our project, but the vast majority of the challenge was because it was so difficult for our group to find time to meet up and get project work done. In addition to that, I had a very heavy work load that semester (arguably the most work heavy semester in my undergraduate career), and started a temporary job doing political polls. While these are not the only things that lead to increased challenge, or even the main ones (one group member dropped the course, and another was very hard to get in contact with\*), this showed me that time and availability are key components to a successful project.

\*Early in the project he very slowly responded to e-mails and was hard to get him to commit to tasks. Late in the project he was temporarily incapacitated due to circumstances beyond our control (he was shot, fortunately he survived).

## Other Problems

The only other problems I can think of at the present are my recent sleeping problems, and that the first major deliverable (the Database Schema) is a major bottleneck.

My sleeping problems combined with increased travel time have resulted in a decreased amount of time to do project work. In addition, I have taken longer to get into the “swing” of scholastic work then I normally do, but that should self-correct before the month is out.

The current major task that is due by the end of the month is the schema of the database. The XML file is very large and the schema is rather intricate. The schema is very well designed, which will ease transition, but the mapping of the database has to take this into account as well as data normalization factors. Since this entire project hinges on the creation of the database, little major development tasks can be undertaken until that is finished. This type of task can only have two people on it max. Should this task take longer then I calculated, it will be difficult to allocate the remaining team member to tasks that significantly contribute toward the core system.

My last resort measure will be to abandon the database and work directly from the XML. However, because a more traditional relational database will greatly facilitate searchabiltiy (i.e. use of classic SQL commands instead of searching the XML document raw), I wish to avoid this if at all possible.