SCORE reviews and PATH responses

1.   Introduction

Purpose of this document is to provide the PATH answers to the SCORE Report [1] reviews done by SCORE team of ICSI, Canada 2009. All the changes needed is done to the main report and here is a repetitive of the content we added to the report to make it easier for audiences to evaluate our work. All the comments and answers are put within Blue color.

1.1 Intended Audience

SCORE reviewers and also some others who like to know the professional reviewers point of view about our project.

1.2 Scope

The scope of this document is limited to the answers to the three reviews we got from SCORE team at the end of the first project submission and before the second submission. We had about two weeks to prepare for the second round and answer the questions as well.

2.   First Reviews

2.1 Summary of the submission

2.1.1   The team of 6 members was distributed, 3 in Sweden and 3 in Croatia.

RE: Although the course was distributed but because teams were grouped on the students' preferences; two groups out of six groups were locally and with remote supervisor.   PATH was a local team in Vasteras, Sweden that only has its supervisor on the remote site (Croatia).

2.1.2   The team started with a RUP approach, but switched to a method that allowed more team communication half way through the project.

2.1.3   There was some requirements gathering by collecting survey information from 30 people, it appears fellow students.

2.1.4   Testing was not discussed in depth. I did not see any actual user testing beyond being able to operate the interface widgets.

2.1.5   The architecture was a basic 3-tier approach.

2.2 Positive

2.2.1   The team did use an existing API in the form of Google Maps. They do note that it was difficult to learn and they probably could have programmed it themselves with less effort. I am dubious about this conclusion.

2.3 Negative

2.3.1   They got a start on eliciting requirements. But 30 is a small number and it appears that responders were other students.

RE: Mostly because of the limited time we had (10 weeks), the requirement engineering done in short time but the feedback that are mentioned in other documents was better and more.

2.3.2   I did not see the use of the conflict manager.

RE: This part added as a section 1.3 to the "SCORE Report".   More details are in "Design Document".

 Users of PATH are able to add advice entries. Also it is possible to review the existing advices and edit them. So there is a probability that users do not agree on the statement and change some content of an advice for several times in one day. To avoid any problem, PATH defined conflict concept and moderators to manage conflicts. If advice changes occur more than Conflict Threshold (CT), then conflict would happen. CT is the number that PATH Administrator defines. This part is
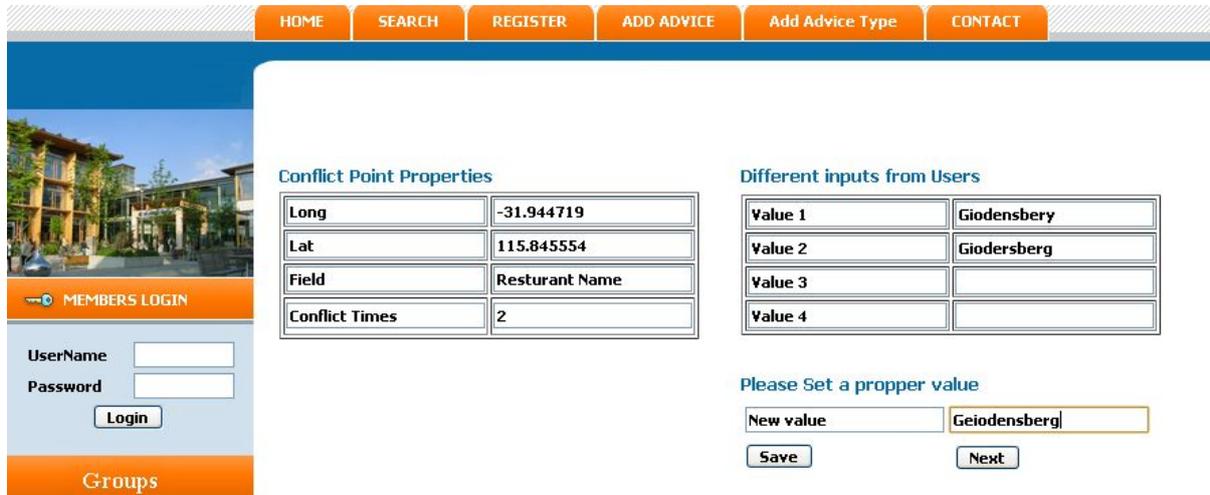
completed partially.

Figure 1: Conflict Management page. PATH Moderators are able to manage conflicts. A conflict happens when users change an advice entry for several times and more than CT.

2.3.3 I never got a clear picture of how the team functioned at a distance, nor at the same site.
//This part added as section 1.1 to the "SCORE Report"
RE: This project was performed as a part of "Distributed Software development" (DSD) course at Mälaradalen University, Västerås, Sweden in collaboration with University of Zagreb, Croatia. DSD course duration was 10 weeks. The project team consisted of six members with each member to contribute about 20 hours weakly according to course load. Figure 2 shows how PATH team was organized. All the team members were located in Sweden while the main supervisor was in Croatia. Teams were grouped according to student interests and in PATH team there was no student from Zagreb.
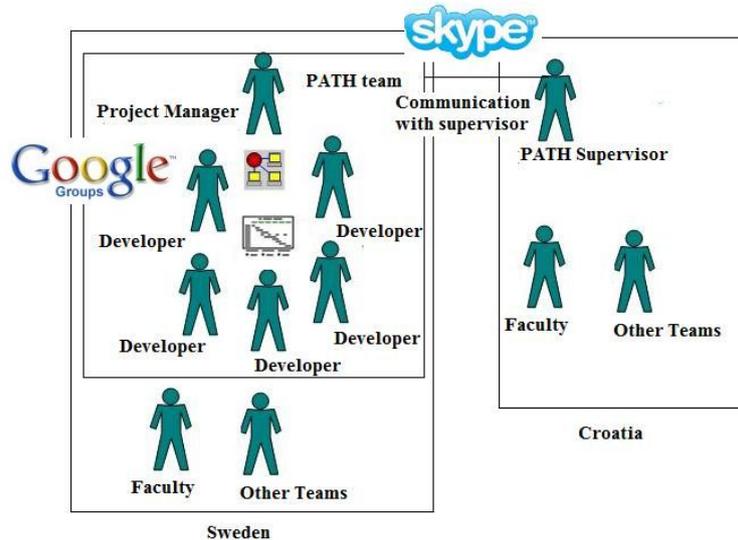

Figure 2: PATH was a group in DSD course that is done in collaboration of Mälardalen University, Sweden and University of Zagreb, Croatia.

2.3.4   The team notes that they moved away from RUP to "agile" in the middle. But what I see described is more along the lines of choosing to work more closely together. This is really orthogonal to agile, and has more to do with distributed nature of the project. They mention tools they used by don't give concrete examples.

// Edited on "SCORE Report"

RE: Initially we used the Rational Unified Process (RUP) model for development. This is a continuous iterative process which let the teams get back to each phase when needed. Due to the limited time we had, one member was assigned to do some implementation and code practicing at the beginning and one team member focused on Google API technology and other similar alternatives while other members started with project requirements and design. Also we had early testing. The other feature of our RUP development was the "Responsibility". Team members' Responsibility[1] emphasized on the RUP Role they had and their tasks were about a unit of work. As we had only our supervisor on a different site, planning for several meetings was not a problem. We fixed two meetings per a week and meeting duration was about two hours. We developed the project with this model for about four weeks. Each member spent about 16 hours individual work and 4 hours during the meetings, but the result was not good enough and development process was slow. It was because of short meetings and long time working separately. There was a delay in understanding the problem or requirement while working alone. These problems mostly were software installation, configuration and primary project environmental need. So we decided to change the development process into agile development. After that, meetings duration extended to 5 hours each and team members worked less at home (about 10 hours) and instead put more time in agile meetings (10 hours meeting in two sessions per week). Also we had a very good collaboration and support from our supervisor. We used long time Skype meeting with her during our sessions, several times. As a result we could cooperate more with each other, project requirement and project design became more clearly understandable and the progress of the project improved significantly. Furthermore all team members shared their knowledge and experiences more comfortable in the new approach. Agile meeting supported changes better and therefore we could easily change classes and database design when it was needed. The table below shows the improvement in methodology changes. After week 5, although the number of assigned tasks increased but the number of missed tasks are reduced. Also less number of posts was needed in the second approach.

Table 1: RUP used for first five weeks to develop PATH and in last three weeks Agile was used. The project progress improved within Agile more.

Weeks Number	No of Tasks Completed	No of missed Tasks	Number of meetings held	Number of documents posted in group	Number of post in group	Weekly hours of team	Methodology
1	6	2	2	10	31	139.5	RUP
2	8	3	2	13	30	146	RUP
3	7	2	1	15	28	150	RUP
4	5	3	2	6	31	116	RUP
5	9	2	4	7	30	130	RUP
6	12	2	1	3	20	131	Agile
7	11	4	1	4	10	140	Agile
8	7	1	1	1	5	120	Agile

2.3.5   Ttable 5 is a style of functional testing - can the user carry out individual actions. It does not test

---

[1] Members Responsibility is described in Project Plan

the actual usefulness of the system. To do this, you need real users with real needs.

RE: More details added on "Test Plan"

2.4 Conclusion

As noted on the project description, this problem is more about requirements engineering and user testing, and less about writing code. From the write-up, it appears that you spent minimal time working with real users and testing in realistic scenarios.

3.   Second Review

This is a well written (although sometimes verbose) and well structured document about a distributed software engineering project. The students seem to have invested a large amount of energy into making this an actually usable project and have succeeded in doing so. The requirements were collected through a survey, which is actually a good solution in the absence of an actual stakeholder. What I especially like is that the students have monitored themselves throughout the project, and this allows them now to reflect on the development process. Very nice.

3.1 Evaluation

> 3.1.1 I am a bit puzzled about the discussion of starting with RUP and then moving to agile development. I would have never used RUP in the first place, the project is too small for that.

RE: We started on 3 different phase at the beginning and after one day studying about the subject. One member assigned to work on Google API as we understood it is more popular and one member was assigned to work on other GIS alternatives except Google. One member was assigned to work on the software and tools needed. Other three members concentrated on project requirement. As you see different phases were picked at the beginning and we followed an iterative model. If we knew some technologies required, we would follow waterfall model. Also we defined "Responsibility" for each member to keep him or her active and more concentrated on one subject and area. Finally even we need more interaction and we moved to Agile from RUP.

> 3.1.2 What is missing in the document is some reflections on the system itself, i.e., since this is now on the web I would have liked to hear something about the way it is being used, if at all..

RE: The web site link sent to 50 end users across the globe. We got about 40 responses and 15 comments. The advertisement to this web site to being used by so many people was not handled due to the limited time we had and the already known improvement we should do. Web Poll is [2].

3.2 Third Review

The problem of BTW proposed by Steve Fickas is chosen. The project is conducted by a six-member team. Three-layered architecture is adopted, consisting of the user interface layer, the business object layer and the data access layer. For implementation, the Google API is used.

3.3 Evaluation

The report is fairly well written. The strong point of this project is that the implementation seems to be done at least at a certain level. It was a good idea to try to collect requirements from real potential customers. In the requirements process, it is described that you have found three types of users: general users, managers and administrators.

There are explanations on general users and managers but I could not find it for administrators.

// this was added as section 2.2 in "SCORE Report". Also we changed the "Manager" name to "Moderator" in order to make it clearer.

RE: PATH Users

As shown in Figure 3, we recognized three users for PATH. The first one is so called User, which is a simple user of the system. Most of the functionalities are available for User. User Use Cases are the outcome of studying requirement tables. Users can change existing advices several times. Two other PATH users are Moderator and Administrator. Moderator is a promoted user and able to decide about the conflicts that was described in section 2.3 on "SCORE Report". Administrator is the one who could decide to set a Conflict Parameter and is not a promoted user. Figure 3 shows three mentioned type of PATH's users.
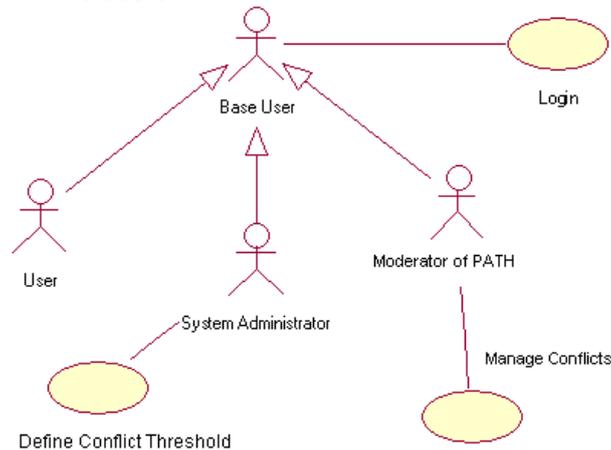

Figure 3: Three types of user for PATH. Most of the PATH functionalities are available for User

It is stated that the manger's role is to decide about conflicts. As the conflict management is mentioned at many places of the report, it must be an important feature of this system, although it is not explicitly described in the given problem document. But what it exactly means is not sufficiently explained.

RE: This part was added as a separate part and as mentioned earlier in this document.

Shifting the process from RUP to agile in the middle of the project is given a special notice. It certainly appears to be an interesting lesson learned but it is also probable that in the latter part of the project, the project members got used to the problem, the development environment and the group so that the productivity was enhanced regardless of whatever process taken. More analysis had better be conducted to make the argument more convincing.

RE: This part is also added as mentioned earlier in this document.

4.   References

[1] SCORE Report available at http://www.fer.hr/rasip/dsd/projects/btw_2/documents
[2] Feedback Poll for PATH application