KTH Royal Institute of Technology
Department of Information and Communication Technology
MSc. Software Engineering of Distributed Systems (TSEDM)

Course: ID2207, Modern Methods in Software Engineering
Teacher: Mihhail Matskin

Final Project

Group 7
Vasileios Charalampidis
Bernardo González Riede

Stockholm, 16 October 2017

Contents

1. Developed stories with time estimates

Core

Login Any employee in SEP can access the system through the login screen where she/he enters her/his username and password. After verification and based on the logged in user's authorization level, she/he will be able to access different functionalities
Estimated time: GUI: 0.5h Logic: 1h

Worklist Every employee has access to a worklist, which lists the pending tasks. A task's scope defines the expected result which is needed in the company.
Estimated time: GUI: 6h Logic: 2h

Workflow for event request

Initiate request The CSO can initiate requests by entering the client's details and information regarding the event. After finish entering the details, the request will be redirected to SCSO.
Estimated time: GUI 3h Logic: 2h

SCSO Process event request The event request is processed by SCSO by either rejecting or approving it.
Estimated time: GUI 2h Logic: 1h

FM Process event request The event request is processed by FM by writing a feedback for it.
Estimated time: GUI 2h Logic: 1h

AM Process event request The event request is processed by AM by either approving or rejecting it.
Estimated time: GUI 1h Logic: 1h

Summary During the first meeting, SCSO creates a summary, refining the initial event request's details.
estimated time: GUI: 3h Logic: 1.5h

Workflow for task distribution

STM Process event request The event request is processed by STM by creating an application related to it and updating its status.
Estimated time: GUI 5h Logic: 2h

Tasks for sub team STMs initiate tasks for the subteams after creating the application. These are sent to a specific employee.
Estimated time: GUI: 2 Logic: 2h

Plan Subteam employees process a task related to an application by creating a plan on how they plan on realizing the task at the event. Optionally, they may include comments regarding the budget. This is sent to their respective STM.
estimated time: GUI: 0.5h Logic: 1h

Staff recruitment

Staff request every department can request staff in an outsourcing or temporary form.
Estimated time: GUI: 1h Logic: 1h

Staff request update HR team updates requests for hiring or outsourcing new employees.
Estimated time: GUI: 1h Logic: 0.5h

Financial request

Budget request AM, STM or FM request a budget adjustment by filling out a form, which gets redirected to FM.
Estimated time: GUI: 1.5h Logic: 1h

Budget request processing FM processes a budget request by creating a budget negotiation record. It stores the requested amount, the amount agreed upon with the client (after contacting him) and a discount if FM offers one.
estimated time: GUI: 1h Logic: 0.5h

Supporting user stories

Retrieve event requests SCSO can search and view all event requests, i.e. active and inactive requests.
estimated time: GUI 1h Logic: 1h

Retrieve employees' records AM, FM & HR team can search and view employee's records.

Estimated time: GUI: 0.5h Logic: 1h

Reports about clients AM, Marketing team and Vice president are able to generate reports about the clients and view already generates ones.
Estimated time: GUI: 1h Logic: 1h

Reports about events AM, Marketing team and Vice president are able to generate reports about events and view already generates ones.
Estimated time: GUI 1h Logic: 1h

Reports about employees AM and Vice president are able to generate reports about employees' utilization.
estimated time: GUI: 1h Logic: 1h

Financial report request Vice president can request financial reports from financial department about the situation of the company.
estimated time: GUI: 1h Logic: 0.5h

Staff schedule & assignments STMs can search and view staff schedules and assignments.
estimated time: GUI: 1h Logic: 1.5h

Return a task in the even request workflow If a person in the workflow notices an error in the requests, it can be returned to the person responsible for it. If there's some error in the budget request, it will be returned to FM, while errors related to the event request's capturing are sent to SCSO.
Estimated time: GUI: 0.5h Logic: 1.5h

2. Release Planning

First release

User Story Name	Value	Risk
Login	Medium	Medium
Worklist	High	Medium
Initiate request	High	Low
SCSO Process event request	High	Low
FM Process event request	High	Low
AM Process event request	High	Low
STM Process event request	High	Low
Summary	Medium	Low

Tasks for sub team	High	High
Plan	High	Medium
Staff request	High	Medium
Staff request update	Medium	Low
Budget request	High	Low
Budget request processing	High	Medium

Combinations

	High value	Medium value	Low value
High risk	1	0	0
Medium risk	4	1	0
Low risk	7	2	0

Theses user stories have been selected since they provide the functions required for the company in the current form to be able to generate revenue.

3. Iteration Planning

Iteration	**Covers functionality**	**User stories**
1	• Workflow of event requests • Client data management	• Login • Worklist • Initiate request • SCSO process event request • FM process event request • AM process event request • Summary
2	• Workflow of tasks distribution for production & services	• STM process event request • Tasks for sub team • Plan
3	• Staff recruitment • Financial requests	• Staff request • Staff request update

		• Budget request • Budget request processing

4. The metaphor

As a metaphor, it can be used the case of a PhD candidate asking from his assigned Professor to expand his research. The professor will examine his request and he will decide whether to approve or immediately reject it. If he approves it, he will forward it to the committee of the University. In the committee there are several people that will go through the request regarding the first approval (while giving notice to the Professor and to the PhD candidate), the extra fund, and the extra staff and material that may be needed for the research group.

Metaphor	**System**
PhD candidate	Client
PhD's research request	Event request
Professor's secretary	Customer Service employee
Research request form	Event request form
Professor	Senior Customer Service Manager
Employee responsible for the funds (FE)	Financial Manager
Employee responsible for the approval of the research request regarding first money estimation (RRR)	Administration Manager
Employee responsible for the resources in the department of PhD's section (PRE)	Production Manager
Employee responsible for the material resources in the lab (LRE)	Services Manager
Employee responsible for hiring new research staff and buying materials for the lab	HR Manager
The Professor will be informed about the decision of the committee and will inform in his turn the PhD candidate	Senior Customer Service Manager will be informed about Administration's Manager decision and will inform the client accordingly

The summary of the meeting between the PhD candidate, the Professor, the FE, the PRE, and the LRE	Summary of Business meeting
Depending on the summary of the meeting the PRE and the LRE may send request for more research staff and some new materials for the lab	Staff request
The FE will post the needs for staff and extra material providers in the website of the University	Job advertisement
The PRE will assign specific people from the department regarding the request. The LRE will also assign the needed materials for the research	Tasks for sub teams
The assigned research group will coordinate their actions and needs. The materials for the lab will be properly given for the lab needs	Plan
According to the needs of the research group and the possible limitations of the extra materials, a negotiation may take place between the PRE and LRE wit the FE for extra funding	Budget request
The FE will process the requests for extra funding, and if he finds out that the PhD candidate has already accomplished his goals from previous funds, then he will give even more money to him for his needs for better results	Budget request processing

5. Pair programming, Refactoring, Iteration Management

Pair programming

While developing the platform, we mostly divided the work, so that one of us to be responsible for the front-end and the other for the back-end. Most of the time, we worked at the same place so that to avoid any possible errors at the passing of variables between the two parts. Of course, we tried pair programming in some sections of the flow of events, so that to have a clearer view of what is happening at the code. We realized that the guidance of the observer eliminated many possible bugs for the system. It helped us a lot to finish things earlier from what was initially planned.

Refactoring

At the beginning we started writing code, having several files in our repository, without taking care of how the code looks like and if it is easily readable. We proceeded with several changes: a) we

reduced a lot the number of front-end files, by introducing similar templates into one, b) we included some comments in the main JavaScript code that handles the worklist, to make it clearer how the flow works, c) we created separate php files for each specific functions of the system, to be able to locate easily possible bugs.

Iteration 1:

- Generalization of *employee* and insertion of role
- Creating *preferences* catalog
- Creating status level with a catalog for *eventRequest*
- Integrating elements of *BudgetNegotiation* into *FinancialRequest* which gets updated
- Changing usage of *eventRecord* to *idTask* to process the workflow
- creating status catalog for *eventRequest*

iteration 2:

- connecting *staffRequest* to a task and then use that to display the tasks to
- changed connection from *financialRequest* to task instead of to *eventRequest*
- changing connections from plan to task

iteration 3

- separating status from task to use the status catalog
- changing sending of *eventRecord* to *idTask* to carry only this one and select the *eventRecord* from it
- test cases: added get task list wrapper

Iteration management

While started writing the first release of the program, we were a bit out of schedule. We had difficulties of succeeding the ‘today’s’ goals. That was because we included many things that we wanted to accomplish the same day. Especially the first days, we also needed some time to refresh our programming skills in web development. After a while, we managed to finish anything on time, without any considerable issues.

6. Acceptance Tests

Successful Login

Test Case Name	**Login**
Expected Actions	1. Navigate to the login page of the website 2. Enter username: 'employeename' 3. Enter password: 'employeename' 4. Click the Submit button or when finished with typing the password, just press Enter in the keyboard
Expected Results	Everyone, except Sarah, will be redirected to their task list. Sarah will be redirected to the new event request form.
Test Result	Successful

Initiate new event request

Test Case Name	**Initiate Event Request**
Expected Actions	1. Sarah fills up the Event Request form with details provided by an already existing client. 2. As a client record, she has to either type 1 or 2. It is considered that there are only two stored clients in our database. 3. With the click of the Send button, the event request is initiated.
Expected Results	The new event request must be successfully forwarded to Janet (SCSO)
Test Result	Successful

Administration Manager's process of event request

Test Case Name	**AM process of event request**
Expected Actions	1. The AM check his worklist with only new tasks that are created because of FM's feedback on new event requests 2. If he clicks in one of them a pop up window appears with details about the event as well as the feedback of FM 3. He has two options with corresponding buttons, either to Approve or Reject the event request

Expected Results	If the AM clicks Approve, the event must be forwarded to SCSO's worklist, so that to call for a business meeting. If the AM clicks Reject, the event request is deleted.
Test Result	Successful

7. Daily stand-up meetings

Meeting 1

Date:	2017-10-04
Participants	Vasileios Charalampidis Bernardo Gonzalez Riede
Meeting notes	Summary of activities yesterday a. Writing of user stories b. Sorting user stories c. Release planning d. Iteration planning Expected actions today a. Finish stand up meeting report b. Add missing users stories c. Metaphor d. Decide procedure for implementation, includes selection of programming language & frameworks, version control and storage solution e. Discuss refactoring f. Architecture of program Issues a. There aren't any issues so far.
Comments	Goal is to have initial work done so programming can start.

Meeting 2

Date:	2017-10-16
Participants	Vasileios Charalampidis Bernardo Gonzalez Riede
Meeting notes	Summary of activities yesterday a. Finishing up with the last JavaScript and PHP files. Expected actions today a. Fix possible bugs b. Perform several tests to avoid any malfunctions in the operation of the system c. Perform unit tests d. Finish up what it is left from the final report Issues a. There were difficulties defining interfaces between presentation and logic. Using the test cases helped in determining on which sides the issues were created.
Comments	The goal is to have everything ready and uploaded on time

8. Comparison with Object-Oriented analysis and design approach

Comparing the approach we followed now and the Object-Oriented analysis and design approach from previous assignments, we realized that we were more productive and we were more capable of doing changes while new demands were coming on the surface. Also, we reduced a lot the errors by introducing the method of pair programming. Generally, we enjoyed more this way of working than the one we followed in the previous assignments.

Visions

- The system in it's current form assumes the omission of ' in all textboxes and areas. It is not protected agains SQL injections.
- Remodel the connections to event request, instead using a project which might be connected to an event request, but doesn't have to. For example for internal projects.
- Implement a functionality to return a task to the previous assignee for corrections.