Modern methods in Software Engineering ID 2207 KTH

SEP Swedish Event Planning And Swedish United Solutions USS

Project report

Course Coordinator: Mihhail Matskin
Teacher: Shatha Jaradat
Teacher: Cosar ghandeharioon

Students
Haseeb Aslam Butt
Adam shafai

Table of contents

Introduction

An internal system management is implemented using the (Agile) XP-programming approach.
Instruction of how the program should be implemented was given by the course instructor which focused on the following business.

Swedish event planning SEP is a company which organise business events and they need an internal system management to be developed in order to automate their system.
The organization have different departments such as customer service CS which register the event requests into the system.
The idea is that every employee should login to the system with some authentication and the employees have different accessibility and responsibility depending on their position in the organization.
Since the time and scope of this project was limited we were instructed to focus on some workflows as follows.

We will focus on the following functionalities in SEP business:
1. Workflow of event requests.
2. Workflow of tasks distribution to services/production departments.
3. Staff recruitment management (through HR)
4. Financial requests management.

Client management, reports management, the employees' records, scheduling issues, salaries and other parts of the problem are not required for the project.

The main elements of XP in this project will be:
1. Developing user stories.
2. Release planning.
3. Iteration planning.
4. Selecting a system metaphor that could be suitable for the problem solving.
5. Developing a system in a test-driven fashion for selected user stories.
6. Refactoring the programs.
7. Pair programming.
8. Daily Stand-up meeting.

User stories

Some GUI parts are excluded from user stories but exits in code and in the iteration plan
Table 1

User Stories	Descriptions	Time in Minutes
login	The user login to the system user credentials checks by the system authentication	20
EventRequest	CS open and fills the form for new event request and submit to the system	30
Client	initiate new event request The details form fills by the SCS and submits to the system	20
Client records	Check the database/list if client is already registered in the system	40
User	The employees in SEP username and password should be used in order to login	30
Position	The employees have different rank or position in SEP	20
ResourceRequest	if resource needed the department subteam should fill the form resource request and submit the system must view this to FM and the FM is able to accept or reject the request.	30
StaffRequest	The department managers should have ability to fill the form staff request and submit	30
Status	The system should view the status of requests and also view if a request accepted or rejected	40

Task	The department manager should have the ability to assign tasks to sub team employee	30
Team	The department manager should have ability to build sub team in order to assign them certain task to complete	30
UserRecords	save all user information in database/ lists	40
EventRecords	the system accept many events and save all events in a database/lists	20
ListClientView	The CS, SCS,AM and FM can search and view for clients records	50
ListUpdaterview	should update and view the status of a request	30
ListStaffRequest	The staff request may come from PM or SM which should view as a list	30
Edit Task	The department manager have ability to edit the task, The employee may also have access to modify the task	10
MainView	GUI	
New account	The system may have functionality which HR can create new user account for the new employee	20
LoginView	GUI	10
ListResourceRequest	The resource may can be send from PM or SM or Sub team which should be view as list	30
createEventRequest	the system may accept many create request action which may view as a list	30

Controller	handle the interaction between model class and view classes	30

Release plan

The following user stories were been chosen among those in table 1.1. The choice is based within the scope of the project timing,some user stories which involve data base and front end development are excluded in order to deliver the project in time.

Login
EventRequest
Client
Position
ResourceRequest
StaffRequest
Status
Task
Team
User
CreateClientView
CreateEventView
ListClientView
ListStaffRequest
ListResourceRequest
ListUpdaterView
LoingView
MainView
Controller

Iteration plan

The architectural style MVC is used to group the user stories in three category. The implementation follows the test driven approach TDD. The user stories as planned were categorised and implemented in three iteration.
The implementation is based in Gradle and the environment used is intellij IDEA with Junit for testing purpose.

Iteration 1

LoginTest Login LoginView UserTest User ClientTest Client Position MainView EventRequestTest EventRequest
Implementation took 2 days

Iteration 2

ResourceRequest StaffRequestTest StaffRequest Status TaskTest Task TeamTest

Team Controller
Implementation took 3 days

Iteration 3

CreateClientView CreateEventView CreateResourceRequest CreatStaffRequest CreateTaskView ListClientView ListStaffRequest ListResourceRequest ListTaskView ListUpdaterView
Implementation took 3 days

Metaphor

A **mailbox:** An internal system management which focus on communication and interaction between employees in a company, the communication is time independent. Employees have the ability to **Create/View/update** and **Submit/send** the messages.

Test

Used test driven programming TDD

Project Design Methodology

We used Agile Methodology for developing this project. Pair Program Technique was used to implement the project. Based on this technique, we both developers worked simultaneously on one computer. One person was coding at a time and the other person was observing and giving live feedback. Moreover, the role was also switched multiple times

during the working day. While working using this technique we realized that it helped in enhancing our productivity as the errors were spotted very early and corrected instantly which saved a lot of time and effort.

Comparison

The object oriented and analysis design approach is more “formal” approach. In this approach clients requirements are gathered in the start and they are considered fix and change in requirements at the later stage become very difficult. Moreover, alot of formal documentation is produced during different phases of the project.
The Agile approach is more flexible approach and welcomes changes. Also Agile approach emphasises more on communication between clients and programmers. Moreover, in Agile approach, focus is on developing the working release of the software with some critical business functionality implemented earlier in the timeline.

Meetings

Project Day 1
Participants: Haseeb & Adam
Agenda: To logically divide the intended system in different components. Discuss an outlook on the schedule and the possible buildup for timely completion of all the required functionalities. Moreover, possible mechanisms for Authentication using Username & passwords were also discussed.

Project Day 2
Participants: Haseeb & Adam
Agenda: Discuss the progress made in the implementation of Authentication mechanism in previous day. Contemplate on the User Interface that can fulfil our non-functional requirement of user-friendly UI. Design details for the Implementation of the functionality of entering new event request by CS representative was discussed. Moreover, the method to test the correct functionality of this use case was also discussed.

References

1. Lectures slides of the course MMSE 2017
Minhail Mitskin, KTH
2. SEP Business Case Description
3. http://www.extremeprogramming.org/