

Software Engineering Lab Report I

Infrastructure Management

For: Professor K. Chandrashekharan

Submitted By:

Pavan Vachhani (16CO151)

Kaushik I (16CO120)

January 23, 2018

1. Process Model

The process model chosen for our project is Agile model. We have considered this model as it suits the closest for our requirements. We considered following criteria while deciding our model:

1. Unclear User Requirement

The customer, here in our case doesn't provide his exact requirements as, it might need user feedback to completely analyse the requirement.

2. Familiar Technology

The technology used in the process of infrastructure management is familiar to us.

3. Low complexity of system

The complexity of the problem is very low, as there exist no conflicts between the entities and the system is well defined.

4. Reliability

The software must be reliable at all times, as we cannot afford the management department to hit a crisis due to the software. Here the software plays a crucial role.

5. Short time schedule

The time schedule to solve this problem is too short. This is rather the project duration assigned to us.

6. Cost limitation

Since this problem is about infrastructure management and the institute always wants the developers to try to limit the cost we have considered this specific factor.

7. Visibility of the customers

The customers need to keep regular contact with the developers as there is a continuous change in the requirements.

8. No component reusability required

Here we are not going to use any component again for developing a new feature in our software.

9. No need for detailed planning

As the concept is familiar to us as mentioned in 2, much detail planning is not required in the development of the software.

10. Flexibility for the developers

We as developers need flexibility as there are changing requirements of the customer and we need to adopt to their demands. So, flexibility plays a crucial role.

11. No need for extensive documentation

This project is a centred around the customer and his/her demands. So, there is no need of extensive documentation, if we are able to fulfil those demands at regular intervals.

1.2 Explanation

Agile model is a combination of iterative and incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product. Agile Methods break the product into small incremental builds. These builds are provided in iterations. Each iteration typically lasts from about one to three weeks. Every iteration involves cross functional teams working simultaneously on various areas like –

1.2.1. Step / Stage 1 Concept

In this stage we will discuss the rough idea about the project with the stakeholders and procure their requirements. We will take notes of their expectations for the project. We will use MS Office tools like MS Word and MS Excel to take down the requirements of the stakeholders and analyse them.

1.2.2. Step / Stage 2 Inception

In this stage we will design the ER diagrams, Use case diagrams, Context diagrams, and DFDs to get precise idea of the requirements. Then we will discuss with the stakeholders and get their comments on our design.

We also will specify the system requirements to the stakeholders.

We are using tools like NetBeans UML for constructing UML diagrams, Use case diagrams. ERDPlus for designing ER diagrams. Draw.io for DFDs.

1.2.3. Step / Stage 3 Construction/Sprint Cycle

In this stage we will discuss the project with the developers and prepare detailed design. The developers will begin programming the modules of the project. UX designers and developers will create a model of the project with barely minimum functionalities.

NetBeans IDE, JDK will be used for the project. We will use MySQL as DBMS and GlassFish for storing application, and Apache Web Server for Online database.

1.2.4. Step / Stage 4 Release/Testing

Here will release the beta version of the software and take the feedback of the customer. As per the comments of the stakeholders, we will add more functionalities and modify the software. We will keep repeating the sprint cycle till all the customer requirements are fulfilled. Finally we will move our sprint to production.

Selenium will be used for finding defects and TestComplete to produce test cases for the software.

2. Tools

| S. No | Step | Tool Name | About the tool |
|-------|-------------------------------|----------------------|--|
| 1. | Concept | MS Office | MS office tools such as excel and word are used to create report about requirements of the customer |
| 2. | Inception | NetBeans UML | This tool is used to create diagrams that will help in a better understanding of the project |
| | | ERD+/Draw.io | This tool helps in the construction of ER and context diagrams |
| 3. | Construction/ Sprint Cycle | NetBeans IDE | IDE is used in the designing and construction the software and its UI |
| | | Java Development Kit | JDK is the background of the programming structure being used in the development of the code for the modules in the software. |
| | | MySQL | This DBMS is used to create and manage the database that is required by the software. |
| | | Glassfish | Glassfish is an application server that helps users to access the database through our application on the web server. |
| | | Apache Web Server | APS is used to store the created database in a remote location so that users of the program can access the data from anywhere. |
| 4. | Release/Testing | Selenium | These tools are used to test and debug the program by creating test cases and running the program for bugs. |
| | | TestComplete | |

3. Summary

Infrastructure Management software development project is made using Agile Process Model with Extreme Programming methodology. There are many reasons for choosing the Agile model over others which were listed above. They can be summarised in a tabular form as below:

| <i>Factors</i> | <i>Waterfall</i> | <i>Spiral</i> | <i>Agile</i> | <i>Code-n-fix</i> | <i>Prototyping</i> | <i>Incremental</i> |
|--------------------------------------|-------------------------|----------------------|---------------------|--------------------------|---------------------------|---------------------------|
| <i>Unclear User Requirements</i> | X | ✓✓ | ✓✓ | X | ✓ | ✓ |
| <i>Familiar Technology</i> | ✓✓ | X | ✓✓ | ✓✓ | ✓ | X |
| <i>Low system complexity</i> | ✓ | X | ✓✓ | ✓✓ | X | ✓ |
| <i>Reliability of system</i> | ✓ | ✓✓ | ✓ | X | X | ✓ |
| <i>Short time schedule</i> | X | X | ✓✓ | ✓ | ✓ | ✓✓ |
| <i>Cost limitation</i> | X | X | ✓✓ | X | X | ✓✓ |
| <i>Visibility of customer</i> | ✓ | ✓✓ | ✓✓ | ✓✓ | ✓✓ | ✓ |
| <i>No component reusability</i> | X | ✓✓ | ✓✓ | ✓ | ✓✓ | X |
| <i>No need for detailed planning</i> | X | X | ✓ | ✓✓ | ✓ | X |
| <i>Flexibility for developers</i> | ✓ | ✓ | ✓✓ | X | ✓ | ✓ |
| <i>No need for documentation</i> | X | ✓ | ✓ | ✓✓ | X | X |

*Here symbols represent: X- Poor ✓- OK ✓✓- Excellent

From this table, we can observe that Agile process model best suits the project.

Under Agile process, we have Scrum and Extreme Programming methodologies. We will be following Extreme Programming framework as it provides the option of incorporating the changes requested by customer during the short development cycles. It also follows a strict priority order, so that we can work on Room booking module firstly. We were also interested in following some practices of XP like automated testing, pair programming, simple design, refactoring, and so on. Thus XP under Agile is our process model.

The list of tools required for development, testing, debugging etc. phases are listed on page 3.