INFRASTRUCTURE MANAGEMENT

For Professor K. Chandrashekharan

Report – 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 the 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.

Considering all these factors of our project we made the analysis considering all the models available in a tabular form as below:

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

*Here symbols represent: ✗ - 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.

Tools required to design and develop the software:

- **Requirements Tool**
  - rmToo
- **Design Tools**
  - *Unified Modelling Language*
    - NetBeans UML
  - *Refactoring Tool*
    - NetBeans IDE
- **Development Tools**
  - *Integrated Development Environments(IDE)*
    - NetBeans IDE
  - *Application Development Framework*
    - Java Development Kit (JDK)
  - *Database Tools*
    - MySQL
  - *Application Server*
    - Glassfish
  - *Web Server*
    - Apache Web Server
- **Testing Tools**
  - *Selenium*
  - *TestComplete*