Software Requirements Specification

For

Agile based Product Development System

Version 0.1 draft

Prepared by Saurav Singh

13-September-2016

Table of Contents

Table of Contents ii

Revision History ii

1. Introduction 3

1.1 Purpose 3

1.2 Document Conventions 3

1.3 Intended Audience and Reading Suggestions 3

1.4 Product Scope 3

1.5 References 3

2. Overall Description 3

2.1 Product Perspective 3

2.2 Product Functions 3

2.3 User Classes and Characteristics 3

2.4 Operating Environment 5

2.5 Design and Implementation Constraints 5

2.6 User Documentation 5

2.7 Assumptions and Dependencies 5

3. External Interface Requirements 6

3.1 User Interfaces 6

3.2 Hardware Interfaces 6

3.3 Software Interfaces 6

3.4 Communications Interfaces 6

4. System Features 6

4.1 Creating a Task 6

4.2 Modify,View,Delete Task 7

4.3 Creating a Module 7

4.4 Modify,View,Delete Module 7

4.5 Assigning the module to PDT 7

5. Other Nonfunctional Requirements 7

5.1 Performance Requirements 7

5.2 Safety Requirements 7

5.3 Security Requirements 8

5.4 Software Quality Attributes 8

5.5 Business Rules 8

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Saurav Singh | 13-08-16 | Initial Draft | 0.1 Draft |
|  |  |  |  |
|  |  |  |  |

**1. Introduction**

**1.1 Purpose**

The main agenda of agile based PDS is to make the communication and interaction between employees, managers and customers easier. Hence, moving forward to accomplish a perfect final product. Agile-PDS will allow managers to keep track of their employees working on a particular product. The customers can be in directly touch with the employees working on their required product.

This document will attempt to provide with all the basic needs for the development of Agile-PDS.

**1.2 Document Conventions**

This document is based on IEEE standard with the font ‘TIMES NEW ROMAN’ and heading size of 20, subheading size of 14 and content of size 12.

**1.3 Intended audience and reading suggestions**

This document is intended for the particular sectors like – IT and Software Industry.

**1.4 Product scope**

If a software development company is having this tool then it can help that particular firm to manage the development of a product by dividing it in modules and assigning that module to the different departments present in the company’s organizational structure. Manager of that particular firm can have hold over his employees that in how much time and individual group is completing the assigned task and what are the changes getting made during that interval of time frame.

**1.5 References**

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications.* IEEE Computer Society, 1998.

**2. Overall Description**

**2.1 Product Perspective.**

The Agile based Product Development Software (Agile-PDS) is a unique system which will divide things in organizational hierarchy, and helps in forming the development timeline required for the completion of the product. The use-case diagram in Figure 1 depicts the external entities and system interfaces for release 0.1.

**2.2 Product Functions.**

Agile-PDS is responsible for the organizational hierarchy made for the development phase of the abstract product. The firm’s manager is responsible for creating the task, defining modules of the product and assigning the modules towards its product development team. Agile-PDS provides a user-interface to the firm’s manager, employees working under product development team and its customer to see the production timeline of the final product.

**2.3 User Cases and Characteristics.**

Customer A customer is the actor who is defining the final goals to the

Manager/HRD, which in turn is defining modules for the

product. Customer can do changes in modules and is

the only one in the system who giving the Manager final objectives.

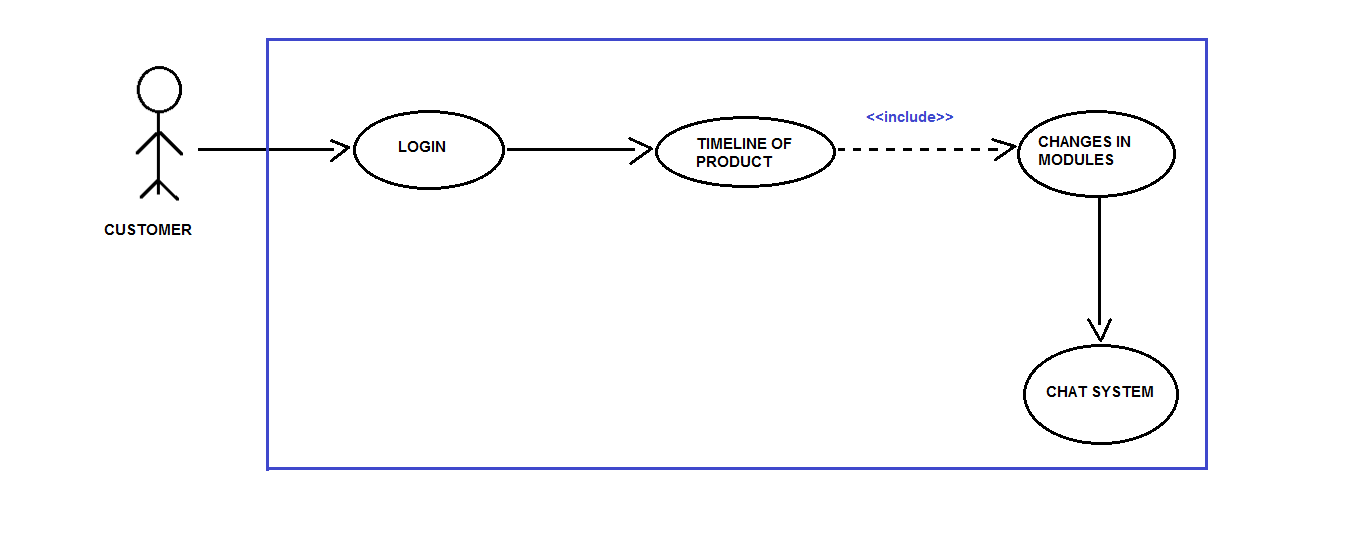


Figure 1.1

Manager/HRD

Manager/HRD is the actor who is responsible for the commencement

of the product development phase, which he does through the system’s

interface provided to him. Manager is responsible for creating a task,

objectifying it into smaller modules according to the customer and finally

assigning it to his product development team.

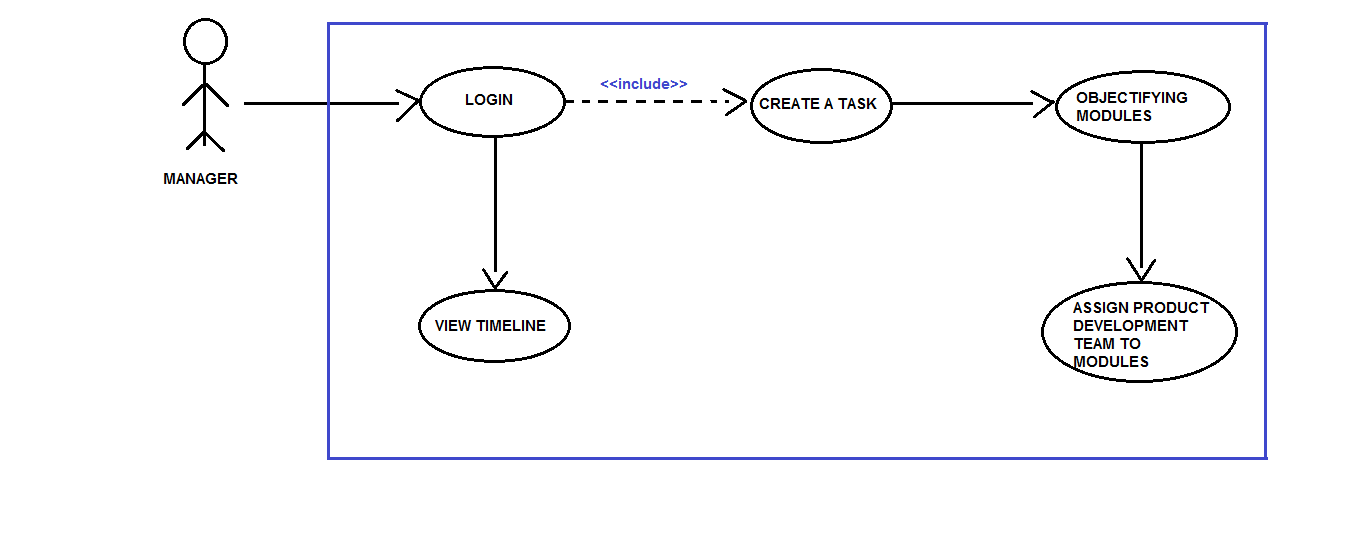


Figure 1.2

Employee The employee working in the product development team is the actor who is

responsible for the weekly reporting on his assigned module by the

Manager/HRD. Employee comprehends the changes which are done to

modules according to the customer’s needs. An employee can get its

objectives much clearer by communicating to the customer by our

Chat/Commenting service.

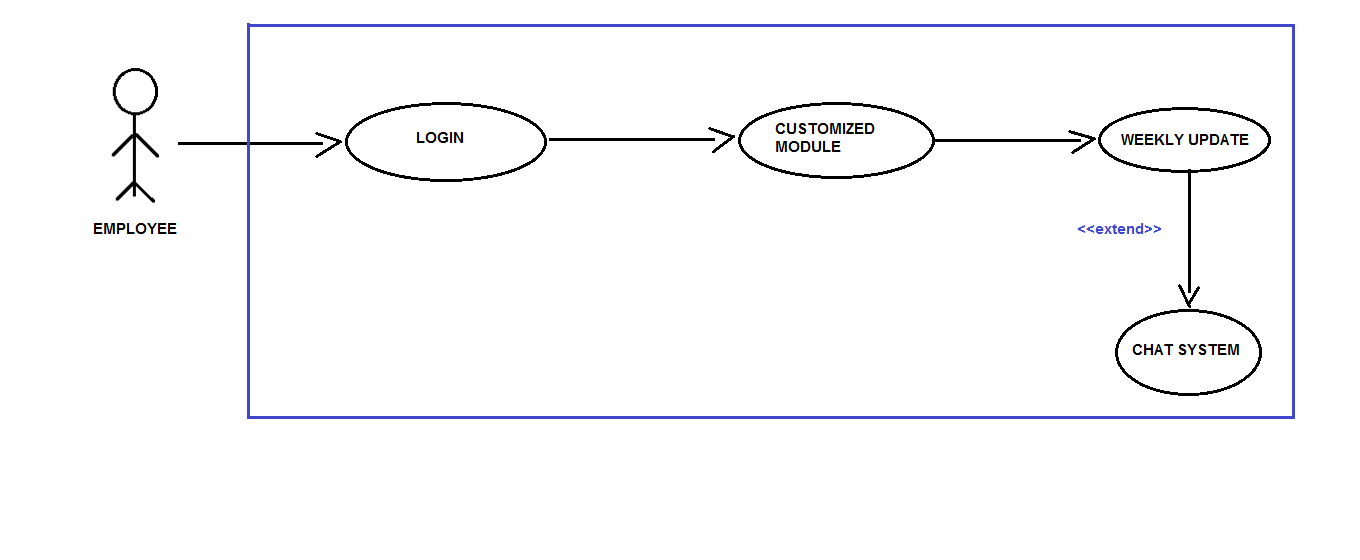


Figure 1.3

**2.4 Operating Environment.**

OE-1: The Agile-PDS shall operate with the following web browsers:

Microsoft Edge version-38 or latest, Google Chrome Version-53 or

Latest.

OE-2 The Agile-PDS shall operate on a server running the current corporate

approved versions of Tomcat Apache Web Server 8.0.

OE-3 The Agile-PDS shall have a hosted interface on the internet for the

customer, managers as well of product development team.

**2.5 Design and Implementation Constraints.**

CON-1 The system shall use the current corporate standard of Mysql engine InnoDB.

CON-2 The system shall use the current corporate standard of Nosql with MongoDB.

CON-3 All HTML code shall conform to HTML 5.0 standard.

CON-4 All backend scripts shall be written in JavaScripts.

CON-5 The system’s design,code and maintenance shall follow the latest version of Eclipse Neon Java EE tools.

**2.6 User Documentation.**

UserDoc-1 The system shall provide an online hierarchical and cross-linked help

system in HTML that describes and illustrates all system functions.

UuseDoc-2 The first time customer when accesses the system, on demand it will

be given Information regarding the User Interface and how to use it.

**2.7 Assumptions and Dependencies.**

AS1 The final product made by the firm is feasible according to their

standards.

DE1 The operation of Agile-PDS depends on the firm’s Manager/HRD who

is responsible for the creation of the task and defining its modules, later

assigning those modules to the product development team.

DE2 The operation of Agile-PDS depends on the employees working in

product development team inputting their work progress weekly which helps in framing the timeline of the final product.

DE3 The operation of Agile-PDS depends on the customer in regular

commentation and communication about the final product’s

modules/goals.

**3. External Interface Requirements**

**3.1 User Interfaces:**

UI 1: The INDEX screen will display a login page with a checkbox, where there can be three different types of users entering their login credentials.

UI(for Customers): The customers can see the progress and the timeline for the respective project. Also there will be a **chat system,** if there are any messages or changes regarding any module for any of the team, the same can be conveyed through the chat system.

UI(for manager):  The manager/HRD can create a new project, set the modules, assign teams to these modules and specify the deadline.

UI(for Development team): The personnel in the Development team can update their project progress/report, weekly. They can also send and revert messages through the **chat system**, if needed any clarification regarding any module.

**3.2 Hardware Interfaces.**

No hardware interfaces have been identified.

**3.3 Software Interfaces.**

SI1 Chat System/Commenting Service.

SI1.1 Chat System is responsible for sending the comments provided by the

Customer to its designated employee working on that particular

module.

SI1.2 Chat System can also be utilized by the employee in clearing its

objectives while designing/implementing its module.

**3.4 Communication Interfaces.**

CI-1 The Agile-PDS should send an email to the customer on creation of the

task and about its modules and should ask for their suggestions.

CI-2 The Agile-PDS should send an email to the customer to report any

problems with the modules(objectifying goals).

**4. System Features**

**4.1 Creating a Task**

**4.1.2 Functional Requirements**

|  |
| --- |
| **Task.name**             **:** The name of the task must be decided and shall be           posted by manager whenever he wish to create his new task and shall be shared among the PDT.  **Task.customer\_name :** the name of the customer who has given the task must be written while creating the modules.  **Task.company\_name :**name of the company to which customer belongs shall be provided. |
| **Task.create\_modules :** manager shall create the modules based on the product specification which that company requires. |
| **Task.changes\_in\_modules :**  Customer shall ask for some changes if it’s not according to his requirements.  **Task.chat\_system :** Using this chat system customer can give update about the changes whether it's okay or not and relating to his response the team can look after that query. |
| **Task.customized\_module :** employee shall work on the allotted module related to his profile.  **Task.weekly\_update :** employees should weekly update the modules on the basis of the **Task.chat\_system** according to the customer requirements. |

**4.2 Modify, view and delete task**

**4.3 Creating a module**

**4.4 Modify, view and delete module**

**4.5 Assigning modules to PDT**

**5. Other Non-Functional Requirements**

**5.1 Performance Requirements**

**User Friendly Environment:** The system will easily understandable and user friendly so that it can be easily maintained.  
**Reduction in the Bugs:** The primary objective of will be to make the UI error free i.e. debug the program as soon as an issue is brought to notice.

**Parallelization:** The UI should be able to run on multiple systems regardless of its configuration. It should be a kind of universal.

**5.2 Safety Requirements**

**Periodic Backups/Recovery:** In case if any situation arises where the data is being completely lost, we will ensure that a backup of the entire project is taken into the server, preferably the Cloud, beforehand. The recovery of this data will be done in an order to reduce the time and effort of the firm. This backup will be done periodically such as weekly or on a monthly basis, depending on the requirements of the firm and the volume of data being generated during this process.

**Encryption:** The login details will be encrypted with SHA-256 or MD5 Hash for the security and safety of the critical data information.

**5.3 Security Requirements**

**Preventing Data Leakage:** There would be a strong requirement to secure the data and protect the privacy of the users as well as the developers at the same time.

**Encryption:** The login details will be encrypted with SHA-256 or MD5 Hash for the security and safety of the critical data information.

**5.4 Software Quality Attributes**

**Availability:** The system should be available to the users as well as the developers anytime during the day, so as to view their progress report and the working statistics of the system.  
**Consistency and Reliability:** The system will be designed to become consistent and not fail at any given point of time, even if it does then we will have counter measures to rectify the system, then and there itself.

**5.5 Business Rules**

* Every individual must be self-reliant and self-directed, they should also have enough trust amongst fellow colleagues in order to complete the project.
* Regular meetings will be held to discuss the prospects of improvement at any stage of the project and choosing the right pace for the team members.