Software Requirement Specification (SRS) Document

Scrum Product Management

Sprint-1

Project Timeline: 07-12-2022 to 12-12-2022

Capgemini

High Level Design & Low Level Design

The purpose of this document is to provide with a template for documenting both HLD & LLD

Document Control :

Project Revision History

Date	Version	Author	Brief Description of Changes	Approver Signature
06-Dec-2022	0.1	Rahul Tarkunde	Start	
06-Dec-2022	0.2	Manas	Added Acronyms	
06-Dec-2022	1.0	Shantipriya	Corrected spellings and approved.	Kanchan

1. Introduction:

A Release backlog manager for Scrum Master is a console-based utility to add/modify Features for a particular release.
Stakeholders (Release Management/Product Management and Scrum master) can review the features desired in the Product backlog and select them for a particular release based on priority, the capacity available, and the efforts involved for a particular feature.
This software requirement specification provides an overview of the entire software. The entire SRS contains an overview, purpose, scope, tools used, and basic description.
This document aims to gather, analyze and give an in-depth insight into the complete feature Backlog Management for Product and Release Management.
The detailed requirements of the Release Backlog Manager application are provided in this document.

1.1. Intended Audience

This document is intended to be read by Development Team and Release Manager. Since this is general-purpose Software hence anyone who is associated with development can use this software.

Project Manager	Development team
Release Manager	Maintenance team

1.2. Acronyms/Abbreviations

PM	Project Manager
RM	Release Manager
Struct	structure
Fname , Lname	First Name, Last Name

1.3. Project Purpose

The purpose of this software or project is to show the requirements for Scrum Product Management which creates direct contact between the Product Manager and Release Manager. And it will provide an efficient and handy platform for the development of the proposed product or software. It applies proven methodologies and uses current software tools so that one can plan, control, and monitor people, processes, and other components.

1.4. Key Project Objectives

1) Allow user registration
2) Allow users to login as Product Manager or Release Manager
3) Validates username and password
4) Allow Product manager to add, delete, display features in backlog
5) Allow Release manager to import, delete, display features from backlog

1.5. Project Scope and Limitation

This project aims to create the development of Scrum Product Management. This system consists of an application that will serve as a platform for the development team and Product Manager to add/modify Features for a particular release.
This System provides a reliable platform for the developer team and Product Manager to follow the Agile methodology in an efficient manner. It provides a platform for the interaction of the Development team and Product Management

1.5.1 In Scope

It can be implemented into the next step that involves Scrum Product Management for example in Sprint Backlog. Another user like Scrum Master can also be integrated with the future proposed system.

1.6. Functional Overview

The following functions are included in the program:

Registration process: The Release Manager and the Product manager/Manager who are going to involve in the process of Development of Product need to register themselves along with Username and Password.

Choose Category/User List: After Registration process it will prompt a category list to select how would the user want to login into. Either user wants to login as Product Manager or Release Manager.

productManagerRegister();

After choosing the Product manager as the user, the first step is to register itself for the authentication process.

releaseManagerRegister();
Similarly, if the user chooses the category as Release Manager then he/she has to register itself for the authentication process.

productManagerLogin();
After Successful registration the product manager need to login with the help of credentials provided during registration process like Password and Username.

releaseManagerLogin();
The Release Manager needs to login with the credentials provided like password and username.

addProductBacklog();
This function allow Product Manager to add feature in the Backlog file. the Product Manager needs to add the proposed features into the Backlog list along with the shirt size and Priority.

deleteBacklog();
This Function will delete the Release Features one by one if Release Manager want to remove/delete.
displayFeatureBacklog();

This Function will display all the Feature backlog added by Product Owner.

importFeature();
This Function will select the features one by one from Backlog list to release list.

deletefeature();
This method is to delete a particular Backlog feature which
Product Owner wanted to delete. This function allow Product Manager to delete feature in the Backlog file.
displayFeatureRelease();
This Function will display all the Release Feature selected by Release Manager**.**

1.7. Assumptions and Dependency

–

- The system should have any Distribution of Linux installed.
- The Proposed System is intended to work on Terminal GUI.
- The system should have either 8GB or more RAM.

- The service is used preferably on a desktop or laptop.

2. 2.Design Overview

Name of the Module	User registration and login
Handled by	Akanksha Singh
Description	User registration: new user details store in file. Login: search the user given input in user details file and allow login if details match.

Name of the Module	User validation
Handled by	Akanksha Singh
Description	Compare the entered username and password with registered username and password.

Name of the Module	Password Encryption
Handled by	Santipriya Bodeboina
Description	This function will Encrypt the password by which a readable message is converted to an unreadable form to prevent unauthorized parties from reading.

Name of the Module	Password Decryption
Handled by	Santipriya Bodeboina
Description	This function will Decrypt the password of converting an encrypted message back to its original (readable) format. The original message is called the plaintext message.

Name of the Module	Display Feature Backlog
Handled by	Divya Rathore
Description	This function will display all the Feature backlog added by Product Owner.

Name of the Module	Display Feature Release
Handled by	Divya Rathore
Description	This Function will display all the Release Feature selected by Release Manager**.**

Name of the Module	Add Product Backlog
Handled by	Manas Behra
Description	This function allow Product Manager to add feature in the Backlog file. the Product Manager needs to add the proposed features into the Backlog list along with the shirt size and Priority.

Name of the Module	Import Feature
Handled by	Manas Behra
Description	This Function will select the features one by one from Backlog list to release list.

Name of the Module	Delete Feature
Handled by	Sakshi Verma
Description	This method is to delete a particular Backlog feature which Product Owner wanted to delete. This function allow Product Manager to delete feature in the Backlog file

Name of the Module	Delete Backlog

Handled by	Sakshi Verma
Description	This Function will delete the Release Features one by one if Release Manager want to remove/delete.

2.1. Design Objectives

You and your team will create a sprint backlog during your sprint planning meeting. The exact frequency will vary based on how long your sprints are, but you'll likely be doing this in every two weeks or once a month. If you're running the Scrum agile method, the scrum master —with the help of the Scrum team—chooses product backlog items to tackle that week. If you're running a different form of Agile, this could be done by a product Manager or product manager.

As the Scrum master or product manager, it's your responsibility to create the sprint backlog and distribute it to all project stakeholders. With the help of the product owner, you will choose backlog items based on priority. Then, document each task's needs in the form of user stories. These are software features written from the perspective of the end user within a workflow.

2.1.1. Recommended Architecture

In this option the architect is part of the Scrum team, and this is the most embedded way that an architect would fit into a Scrum project.

2.2. Design Alternative

We have used file handling instead of linked list because linked list is a dynamic which resizes itself when insertion and deletion operations are performed. Insertions are performed at the end and in deletion when element gets deleted every element moves next to the position but it will increase the complexity of program.

2.2.1. User Interface Paradigms

This project allow user to register and login as product manager or release manager. The Product Manager can add any number of features which he wants to introduce for his product. The features added by the Product Manager will be stored into the Product backlog list which can be later remove or modify by the Product Manager itself. They ensure that the expectations of the completed product are communicated and agreed upon.

After the addition of features into the Backlog list by the Product Manager, the role of the release manager comes into play. Now the Release Manager needs to select the features from the Backlog list according to the Priority marked by Product Manager.

2.2.2. Error Detection / Exceptional Handling

New users should register before login or else it displays the no user found. Registered users have to login with valid credentials. otherwise, they will get invalid username or password.

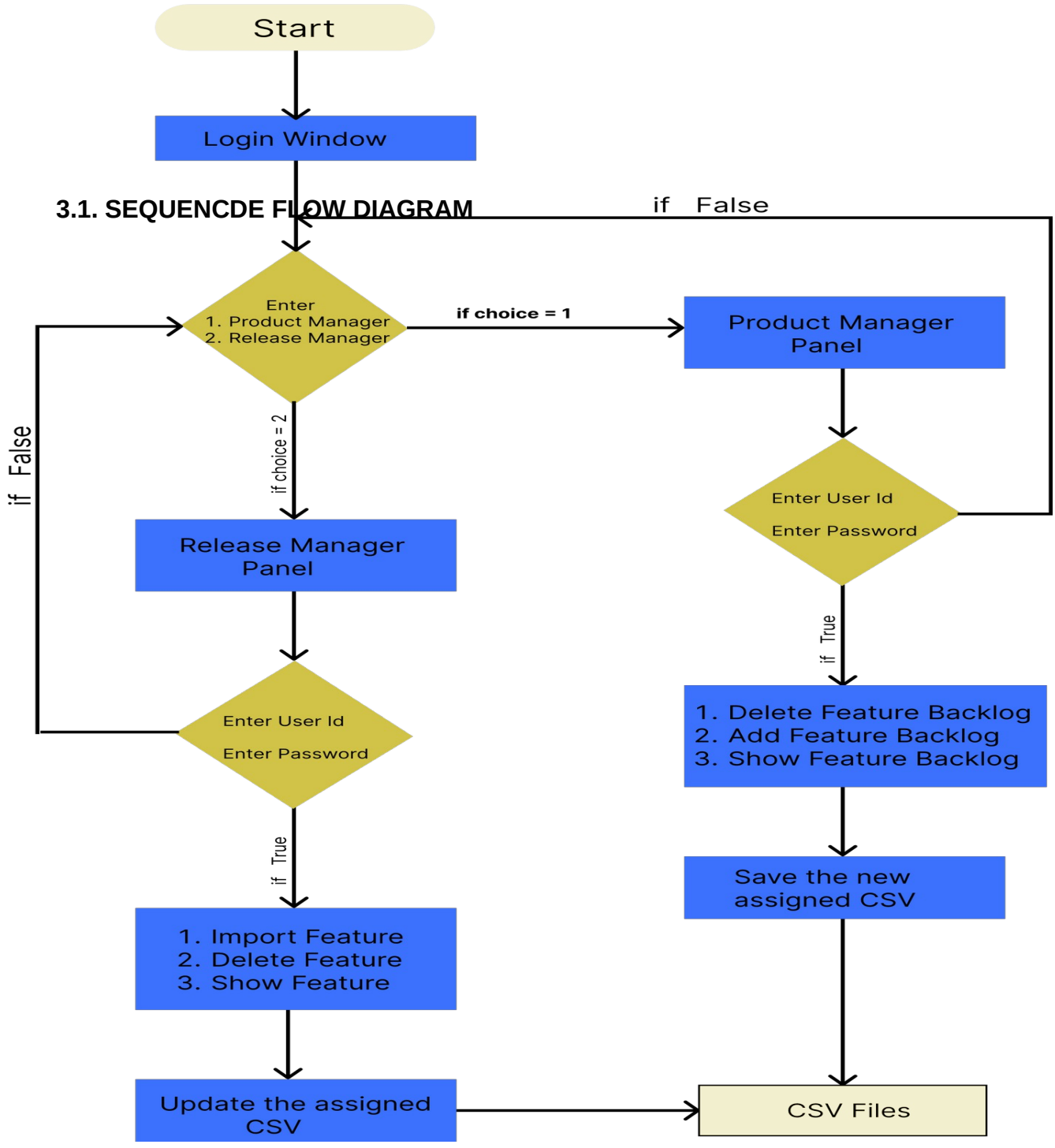
2.2.3. Performance

The system will work on the client terminal. The performance depends on the hardware component of the user's system.

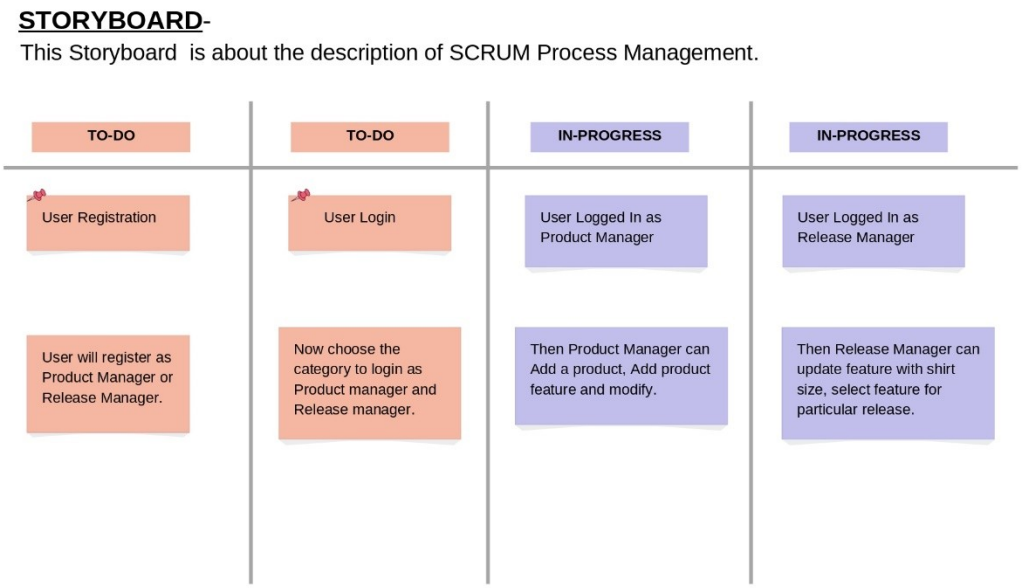
3.System Architecture

- Supportability: The system is easy to use.
- Design Constraints: The system is built using only C language.
- Usability: The Scrum Process Management system will cut down the gaps between the Product Manager and Release Manager and will be more helpful by providing a platform for efficient interaction between both.
- Reliability & Availability**:** The system is available 24/7 that is whenever the user would like to use the system, they can use it up to its functionalities.
- Performance: The system will work on the user's terminal**.** It is Terminal based Software.

3.1. SEQUENCDE FLOW DIAGRAM



3.2. STORY BOARD OF PROJECT



The Product Manager can add any number of features which he wants to introduce for his product. The features added by the Product Manager will be stored into the Product backlog list which can be later remove or modify by the Product Manager itself. They ensure that the expectations of the completed product are communicated and agreed upon.
After the addition of features into the Backlog list by the Product Manager, the role of the release manager comes into play. Now the Release Manager needs to select the features from the Backlog list according to the Priority marked by Product Manager.

The Shirt size and Capacity of Development team should also be taken into consideration while selecting Features for release.

4. DETAILED SYSTEM DESIGN

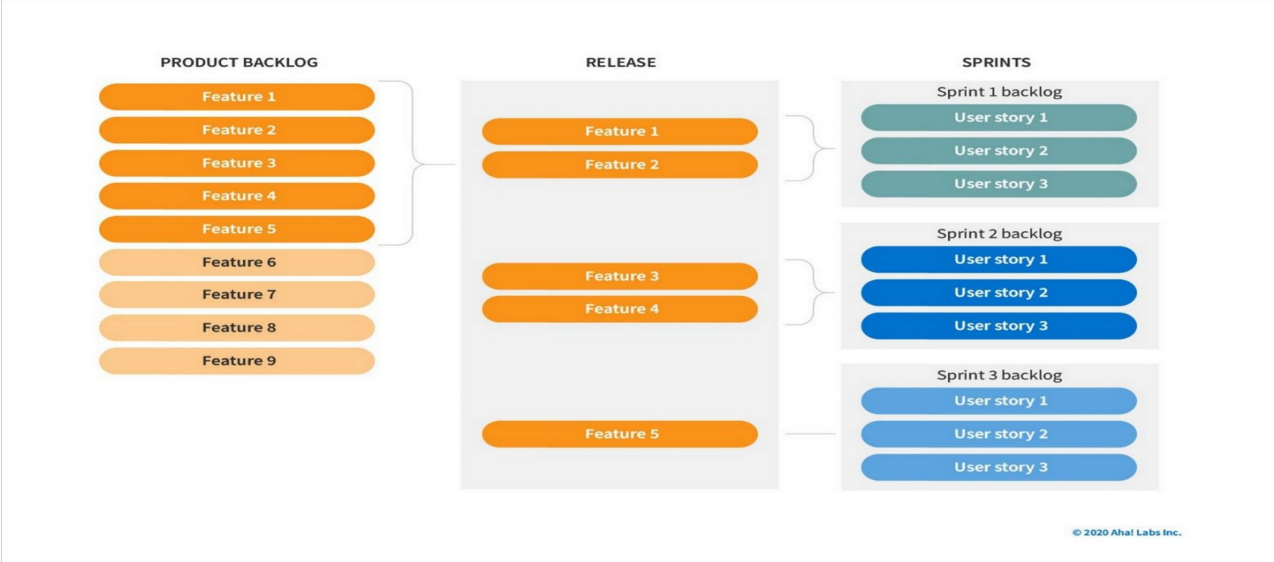
The Scrum Product Management is a Software which helps to manage the process of software development.

It allows:

- It allows Product Manager to orders the work for complex problem into Product Backlog and can add any numbers of features which he wants to add into its product.

- The scrum Teams turns a selection of the work into an increment of value during the selection.

- The Release Manager and Stakeholders inspect the result and adjust the features according to the Shirt size and capacity as well as Priority.



Role:

The Product Manager can add any number of features which he wants to introduce for his product. The features added by the Product Manager will be stored into the Product backlog list which can be later remove or modify by the Product Manager itself. They ensure that the expectations of the completed product are communicated and agreed upon.

After the addition of features into the Backlog list by the Product Manager, the role of the release manager comes into play. Now the Release Manager needs to select the features from the Backlog list according to the Priority marked by Product Manager.

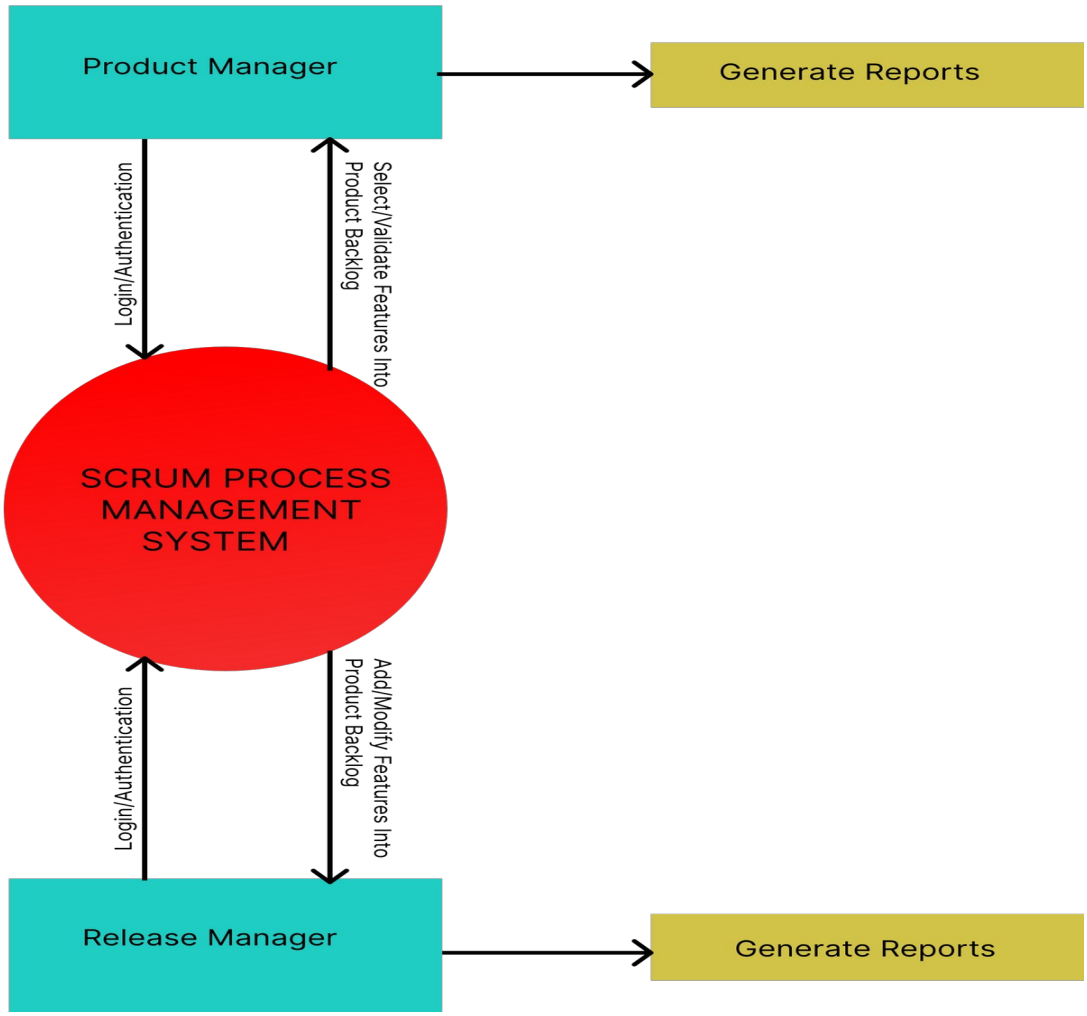
The Shirt size and Capacity of Development team should also be taken into consideration while selecting Features for release.

4.2. DATA FLOW DIIAGRAM

The DFD or Data Flow Diagram maps out the flow of information for the given software. The given DFD diagram depicted the flow of data and information of the proposed system.

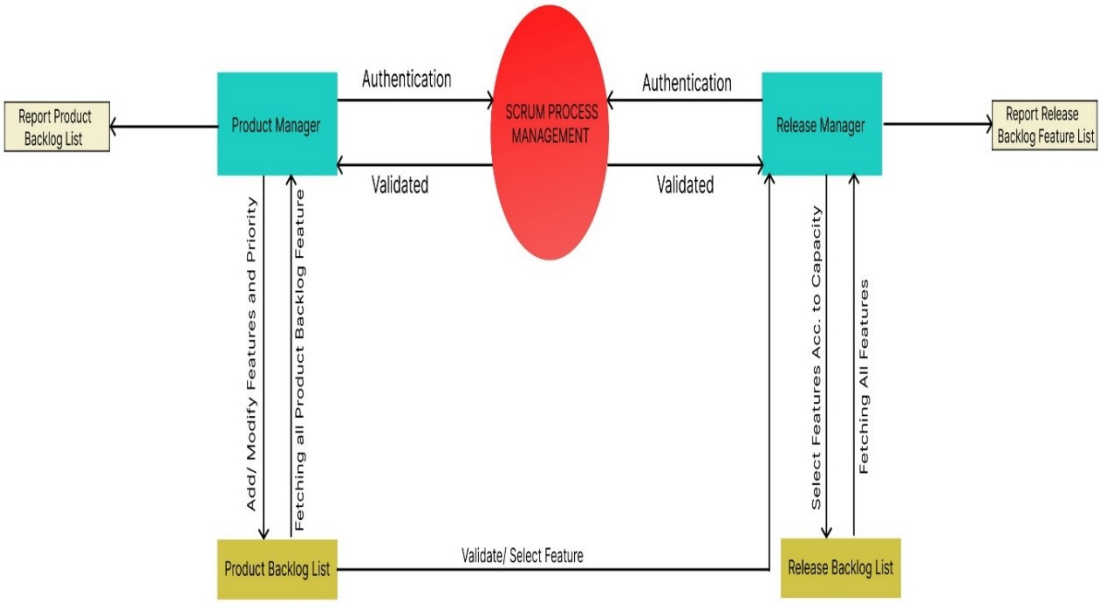
4.2.1. DATA FLOW DIIAGRAM(LEVEL-0)

This is the DFD level 0 diagram of given software:



4.2.1. DATA FLOW DIIAGRAM(LEVEL-1)

This is the DFD level 1 diagram of given software:



LEVEL 1 DFD

5. ENVIRONMENT DESCRIPTION

5.1.Time Zone Support

- **IST- Kolkata, IST-Mumbai**

5.2. Language Support

- **English**

5.3. User Desktop Requirements

- 64-bit processor, 1 GHz or faster
- At least 2 GB free hard drive space
- At least 1 GB RAM

5.4. Server-Side Requirements

- 64-bit processor, 1 GHz or faster
- At least 1 GB free hard drive space
- At least 1 GB RAM

5.5. Deployment Considerations

- Easy setup
- Local storage is used.
- No network latency to consider.
- To scale buys a bigger CPU, more memory, larger hard drive, or additional hardware.

5.5.1 Application Server Disk Space

No such disk space is required as the program is fully functional on online IDE(s) as well. The Local Operating System is required and one text file to store the records of processes.

5.5.2 Database Server Disk Space

No such disk space is required as the program is fully functional on online IDE(s) as well. The Local Operating System is required and one text file to store the records of processes.

5.5.3. Integration Requirements

No such disk space is required as the program is fully functional on online IDE(s) as well. The Local Operating System is required and one text file to store the records of processes.

5.5.4 Jobs

We can establish connections between users who are connected to the server. And we can search the song history of the user.

5.5.5. Network

- **End to End**

3. 5.6. Configuration

5.6.1. Operating System

- **Linux environment**

Change Log

QMS Template Version Control (Maintained by QA)

Date	Version	Author	Description
28-May-2015	1.0	QA Team	Initial Version