< Project Name >

Software Requirements Specification

Version 1.0

Group Id: <<S17….>>

Supervisor Name: <<Sir/Miss ….>>

Revision History

Date	Version	Description	Author
Current Date	1.0	SRS document includes Scope of the project, Functional Requirements, Non Functional requirements, Use Case Diagram, Usage Scenarios, Adopted Methodology, Work Plan/ Gantt Chart	Student Name VU Id

Table of Contents

1. Scope (of the project)

Project scope is the part of project planning that involves determining and documenting a list of specific project goals, deliverables, features, functions, tasks, deadlines, and ultimately costs. In other words, it is what needs to be achieved and the work that must be done to deliver a project.

2. Functional Requirements Non Functional requirements

Functional Requirements

A functional requirement describes what a software system should do,

Functional Requirements: specify the functionality of the system.

Functional requirements are the activities that a system must perform

Examples

Add customer

Print invoice

Display the name

Non Functional Requirements

Non-functional requirements describe how the system works,

how the system should behave and that it is a constraint upon the systems behavior.

Examples

Performance:

Application should perform all of its functions excellently and effectively without any inconvenience.

Availability:

This application will be available free for everyone

Portability:

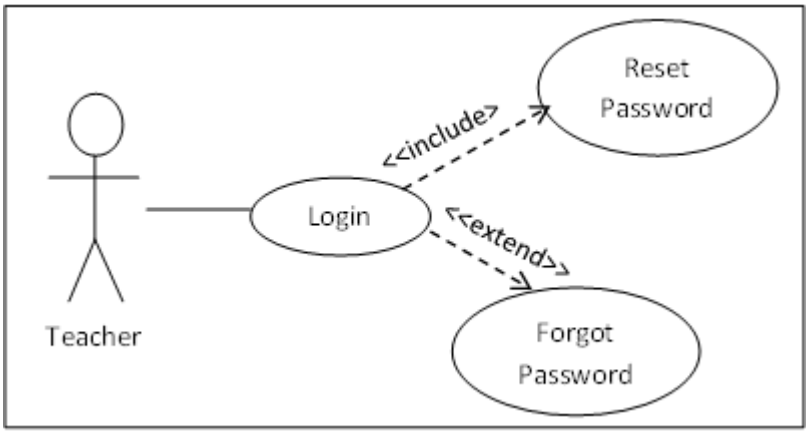
The application can easily portable from one desktop application to another

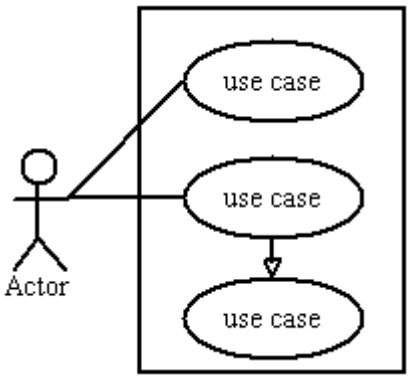
User Friendly:

Application will have a user friendly interface.

3. Use Case Diagram

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved.

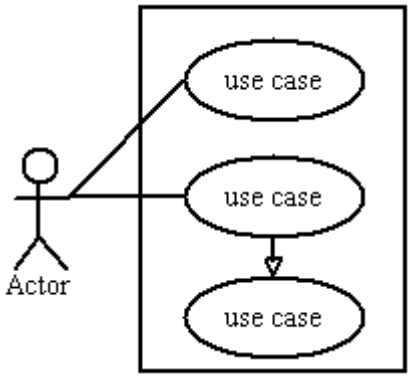




Basic Use Case Diagram Symbols and Notations

System

Draw your system's boundaries using a rectangle that contains use cases. Place actors outside the system's boundaries.

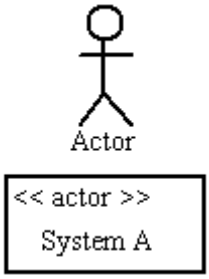


Use Case

Draw use cases using ovals. Label with ovals with verbs that represent the system's functions.

Actors

Actors are the users of a system. When one system is the actor of another system, label the actor system with the actor stereotype.



Relationships

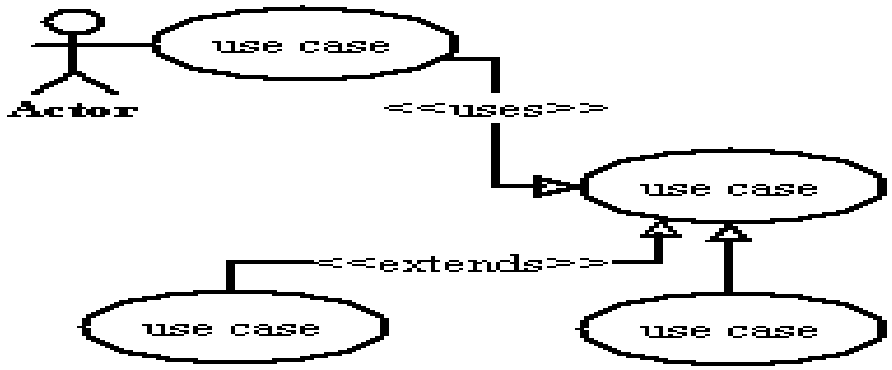
Illustrate relationships between an actor and a use case with a simple line. For relationships among use cases, use arrows labeled either "uses" or "extends" or "includes

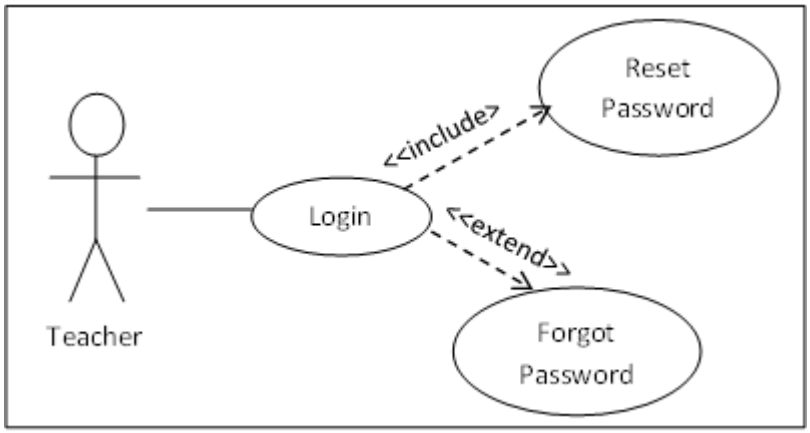
A "uses" relationship indicates that one use case is needed by another in order to perform a task.

An "extends" relationship indicates alternative options under a certain use case.

An include relationship is a relationship between two use cases It indicates that the use case to which the arrow points is included in the use case on the other side of the arrow. This makes it

possible to reuse a use case in another use case.





The following is a sample use case diagram representing the order management system. So if we look into the diagram then we will find three use cases (Order, Special Order and Normal Order) and one actor which is customer.

The Special Order and Normal Order use cases are extended from Order use case. So they have extends relationship. Another important point is to identify the system boundary which is shown

in the picture. The actor Customer lies outside the system as it is an external user of the system.

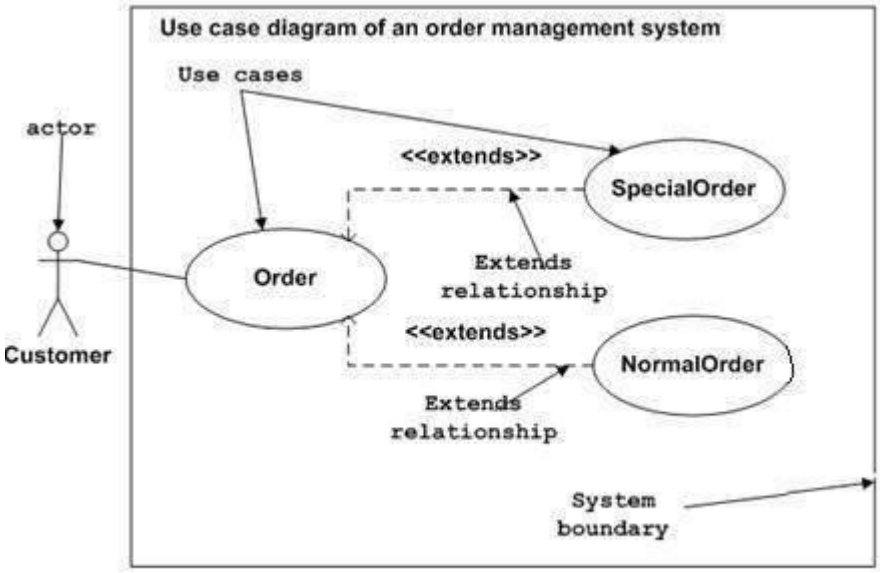
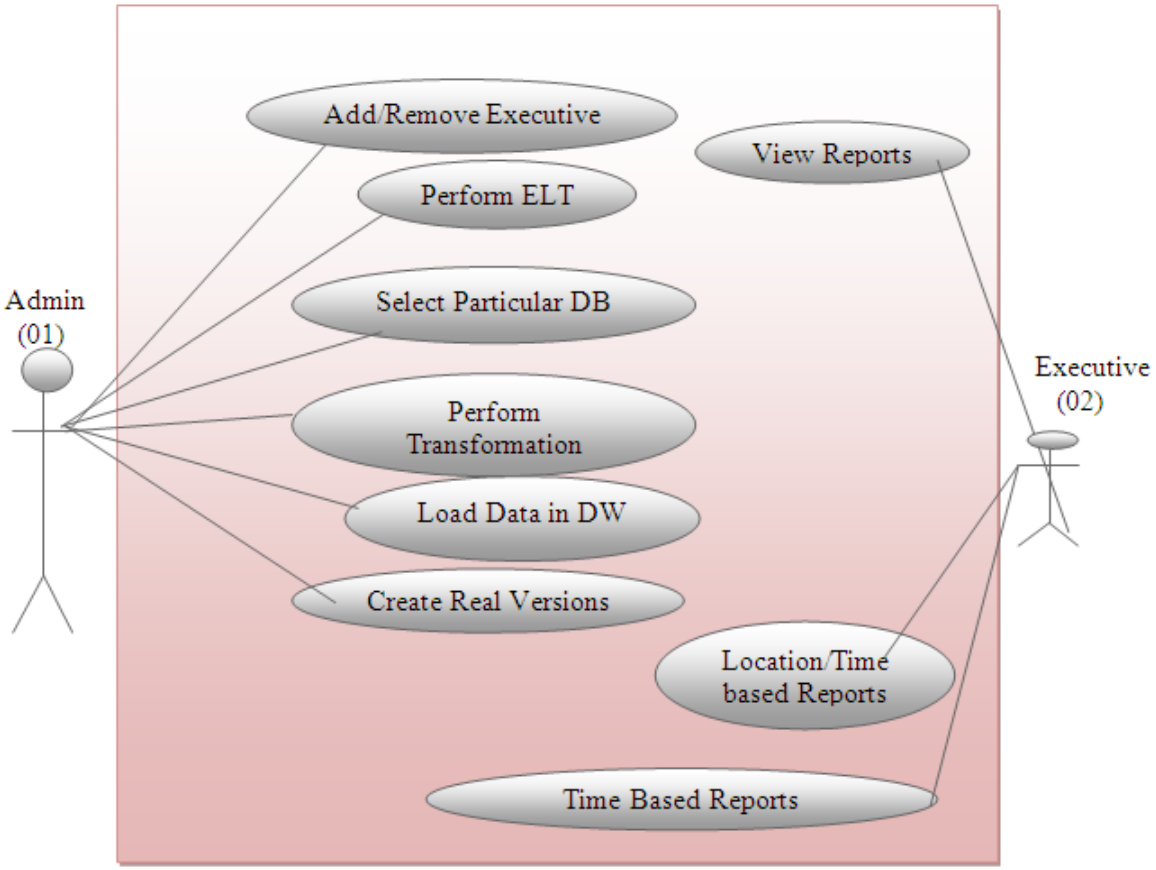


Figure: Sample Use Case diagram

Sample Use Case Diagram



4. Usage Scenarios

A usage scenario, or scenario for short, describes a real-world example of how one or more people or organizations interact with a system. They describe the steps, events, and/or actions which occur during the interaction.

Use Case Title	**Add User**
Use Case Id	1
Requirement Id	3
Description: This use case is about adding a new user to existing system with the privileges defined at time of user account creation.	

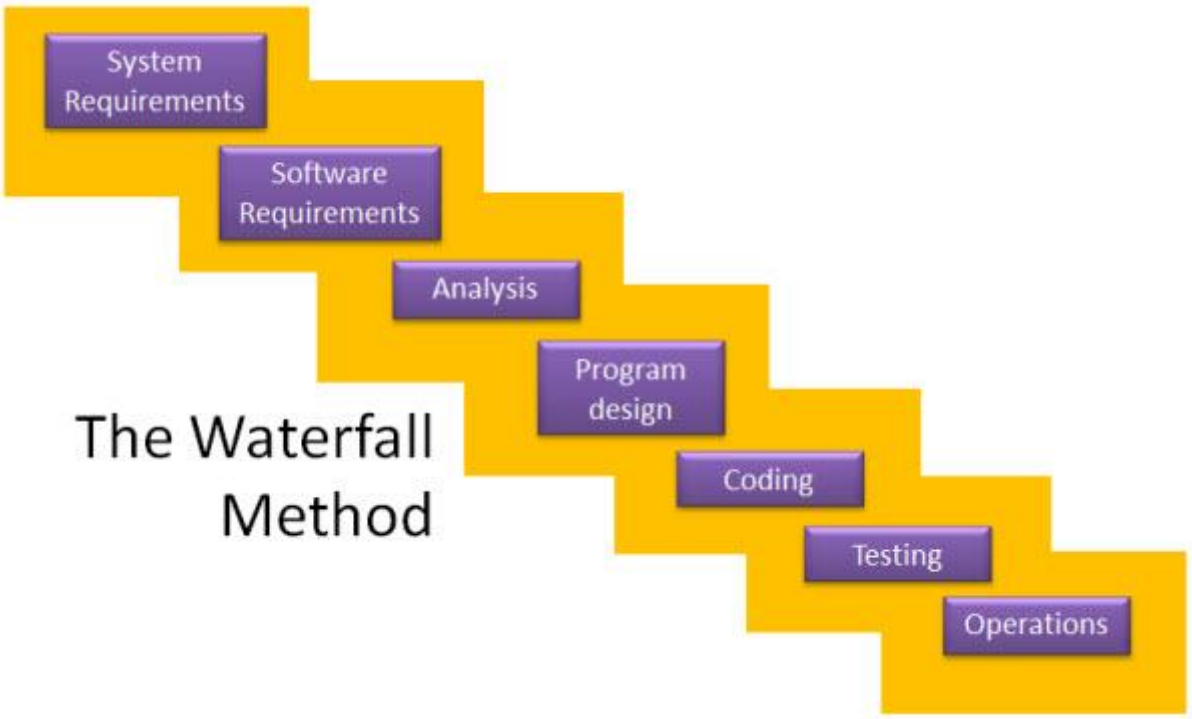
Pre Conditions: 1. All must-required information about the new user should be available. 2. Database should be available in online mode.	
Task Sequence	**Exceptions**
1. Administrator opts to add a new user account.	
2. System asks for necessary information.	
3. Administrator provides all the required information and opts to complete the operation.	
4. There is a problem in the data provided; some data needs to be corrected. – Administrator checks the available information and corrects the error. – Administrator continues from the step 3.	
5. System after confirmation adds the new account.	
6. System sends the account creation email to the administrator's email id and user's email address.	
Post Conditions: – A new user account is successfully created.	
Unresolved issues:	
Authority: Administrator	
Modification history: 1.0 **Author:** <Project or Group ID> **Description:**	

5. Adopted Methodology

It is a method to use development of project. We have adopted the VU Process Model which is the combination of Waterfall methodology and Spiral methodology. VU Process Model will combine the benefits of Waterfall and Spiral methodologies.

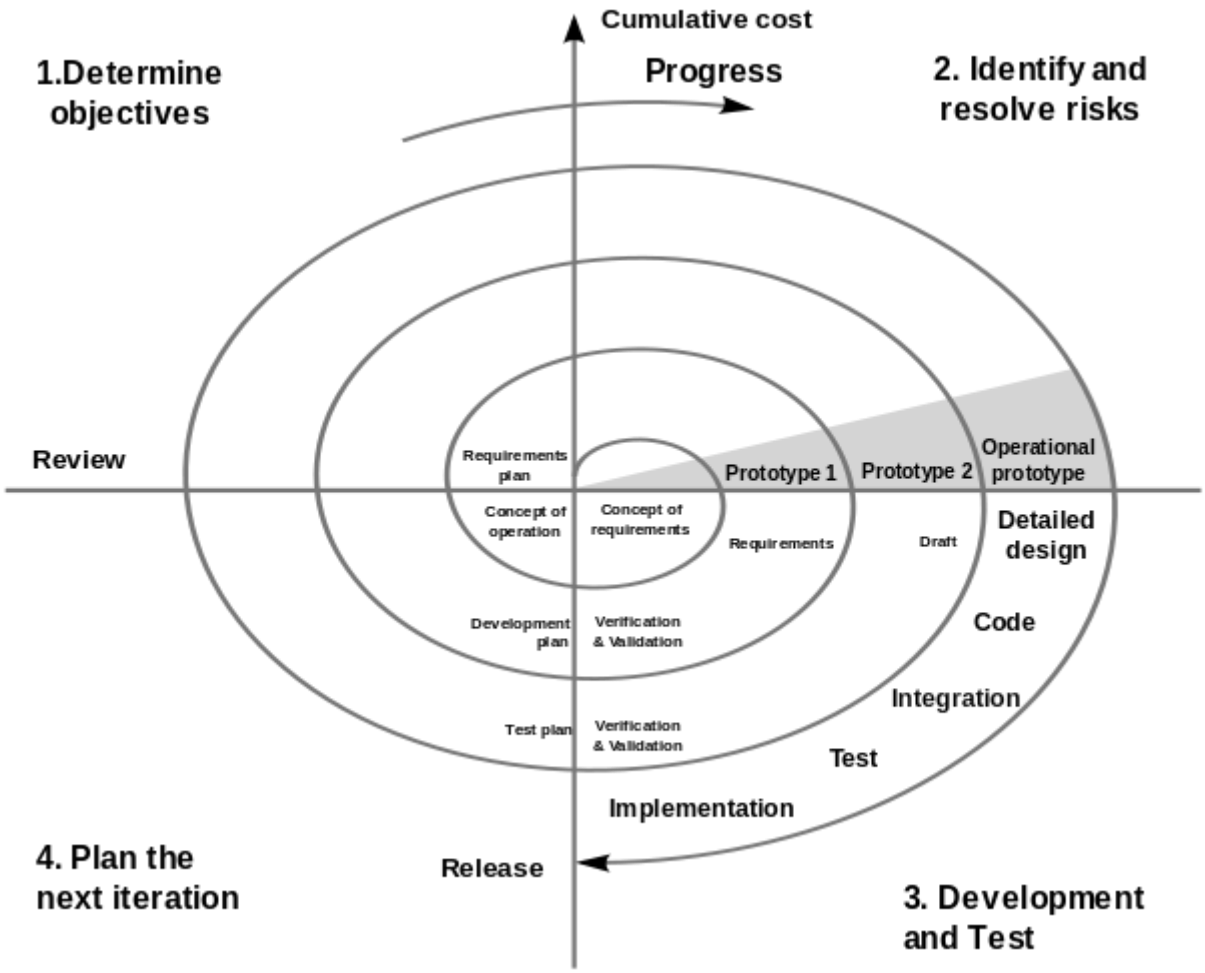
The waterfall model

The waterfall model is a sequential design process, used in software development processes, in which progress is seen as flowing steadily downwards (like a waterfall) through the phases of Conception, Initiation, Analysis, Design, Construction, Testing, Production/Implementation and Maintenance.

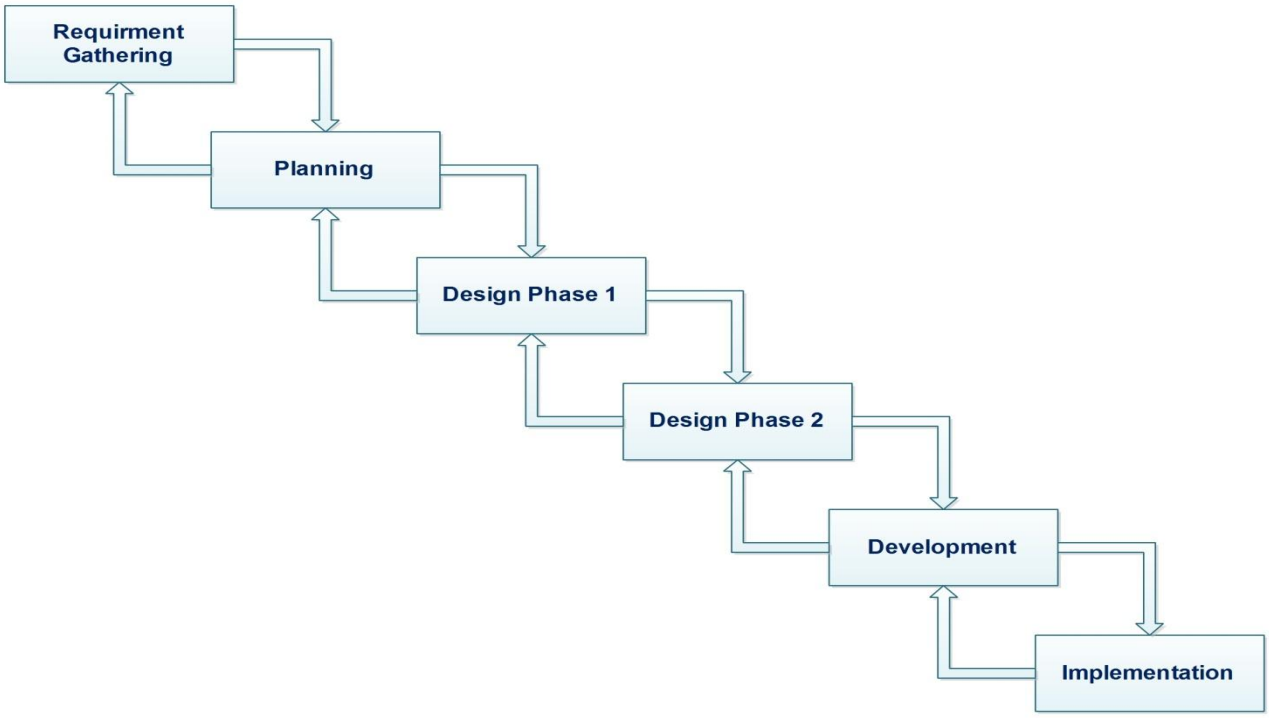


The spiral model

The spiral model, also known as the spiral lifecycle model, is a systems development method (SDM) used in information technology (IT). This model of development combines the features of the prototyping model and the systems development life cycle (SDLC).



VU process Model.



It is decided to select VU process Model that is a Combination of Water Fall model and Spiral Model.

Reasons for Choosing VU process Model.

1. The main idea to choose is to get the benefits of both these models.
2. It is sequential model with backward repetition.
3. We want to achieve the linear nature of waterfall and iterative plus risk reduction nature of spiral model.
4. In vu process model we will be work in phases to complete the given project.
5. All the activities are performed in a sequence in VU Process Model.

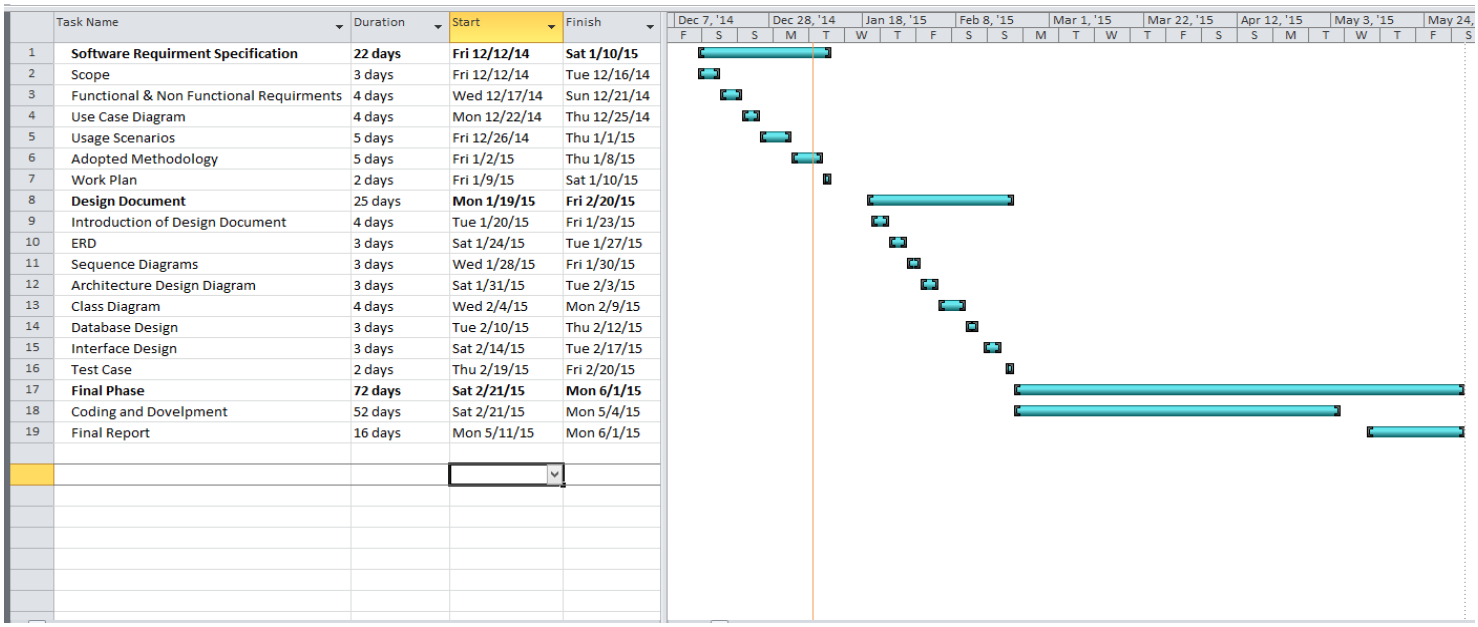
If we want to do correction or betterment at any stage and repetition whenever necessary then we can choose VU process model

6. Work Plan / Gantt Chart (Use MS Project to create Schedule/Work Plan)

The work plan is a document that consulting firms use to organize a project. It outlines the plan by which the company plans to complete a quality project within a given amount of time and in compliance with a set budget.

Gantt charts are useful tools for planning and scheduling projects. They allow you to assess how long a project should take, determine the resources needed, and lay out the order in which tasks need to be carried out. They are useful in managing the dependencies between tasks.

When a project is under way, Gantt charts are useful for monitoring its progress. You can immediately see what should have been achieved at a point in time, and can therefore take remedial action to bring the project back on course. This can be essential for the successful and profitable implementation of the project.



	Task Name	Duration	Start	Finish
1	**Software Requirment Specification**	**22 days**	**Fri 12/12/14**	**Sat 1/10/15**
2	Scope	3 days	Fri 12/12/14	Tue 12/16/14
3	Functional & Non Functional Requirments	4 days	Wed 12/17/14	Sun 12/21/14
4	Use Case Diagram	4 days	Mon 12/22/14	Thu 12/25/14
5	Usage Scenarios	5 days	Fri 12/26/14	Thu 1/1/15
6	Adopted Methodology	5 days	Fri 1/2/15	Thu 1/8/15
7	Work Plan	2 days	Fri 1/9/15	Sat 1/10/15
8	**Design Document**	25 days	**Mon 1/19/15**	**Fri 2/20/15**
9	Introduction of Design Document	4 days	Tue 1/20/15	Fri 1/23/15
10	ERD	3 days	Sat 1/24/15	Tue 1/27/15
11	Sequence Diagrams	3 days	Wed 1/28/15	Fri 1/30/15
12	Architecture Design Diagram	3 days	Sat 1/31/15	Tue 2/3/15
13	Class Diagram	4 days	Wed 2/4/15	Mon 2/9/15
14	Database Design	3 days	Tue 2/10/15	Thu 2/12/15
15	Interface Design	3 days	Sat 2/14/15	Tue 2/17/15
16	Test Case	2 days	Thu 2/19/15	Fri 2/20/15
17	**Final Phase**	**72 days**	**Sat 2/21/15**	**Mon 6/1/15**
18	Coding and Dovelpment	52 days	Sat 2/21/15	Mon 5/4/15
19	Final Report	16 days	Mon 5/11/15	Mon 6/1/15