PRACTICAL – 2

Practical – 2.2

Aim: To perform the system analysis task of your system :

Prepare Software Project Management Plan (SPMP) Document.

SPMP – Password Manager

Table Of Contents:

SPMP – Password Manager

Software Project Management enables a group of Software developers to work efficiently towards the successful completion of the project.

1. Introduction

This document is a formal plan for a Password Management System. The following section give a basic background of the project including an overview, deliverables, milestones and a glossary of terms and acronyms.

1.1 Overview:

The chief intent of this project is to provide the users, an efficient, safe and secure means of storing their personal or professional passwords all in one place, which can be accessed anytime using a single master password after thorough verification. Thus, the hassle of remembering several passwords and the risk of generating easy-to-remember but weak passwords is eliminated.

Following is the high level functionality that the Password Manager will provide:

- Maintain a secured database of all usernames and portals/websites as well as their corresponding passwords in the Password Manager database.
- Provide a convenient Graphical User Interface (GUI) so that the users can get access to the database after verification and perform the following functions:

  ✓ Register to generate a strong and unique master password.
  ✓ Have access to his account and perform desired functions after verified log in.

  After valid master password and OTP verification:
  ✓ Add a portal address and desired username corresponding to the password to be generated and stored.
  ✓ Retrieve a stored password.
  ✓ Update a stored password.
  ✓ Discontinue the use of the application and delete his account.

The team members have agreed that Password Manager will be developed as a web – based application, since a web – based solution will provide many benefits including the following:

- The application only needs to be installed and maintained on one desktop/laptop or android phones.
- The application can be accessed from any computer or android device which has Internet access and compatible web browser.
- The system is available 100 % for the user and is used 24 hours a day and 365 days a year. The system shall be operational 24 hours a day and 7 days a week.

1.2 Deliverables:

The table below lists the deliverables that will be provided as part of the system. Both product and project deliverables are included. Product deliverables are those items that are created to produce the system. Project deliverables are those items are created to support the project.

Deliverables	Description	Dates
Software Project Management Plan (SPMP)	A complete formal project plan, including technical and managerial processes that will be implemented in the development and delivery of the system	
Software Requirement Specification (SRS)	A formal document detailing the functional and non-functional requirements of the system	
Test Scenarios	Formal documentation detailing scenarios that must be followed in order to ensure that the product software is satisfactorily tested.	
User Guide	A formal document describing, for each user type, how to use the entire system	
Final Presentation	A demonstration of the product software and a presentation of the project experience	

Table 1.2.1 Deliverables

1.3 Evolution Of the SPMP:

All members on the project team have agreed upon this SPMP. Any team member may make changes to the SPMP, but the entire project team must approve all changes. This document may be periodically updated during the lifecycle of the project to reflect changes in the project schedule. All changes will be documented in order to keep the SPMP current.

1.4 Reference Materials:

- http://libcircsys.sourceforge.net/SPMPv1_06082004.htm
- https://www.geeksforgeeks.org/software-engineering-software-project-management-plan-spmp/

1.5 Definition and Acronyms:

This section provides a quick reference for the acronyms used throughout this document.

✓ SQL → Structured Query Language
✓ GUI → Graphical User Interface
✓ IDE → Integrated Development Environment
✓ SRS → Software Requirement Specification
✓ SPMP → Software Project Management Plan

2. Project Organization

The following sections explain the organizational structure for the project, the specific roles and responsibilities for each team member, and the process model that will be used.

2.1 Process Model:

The reason for choosing Agile model for our Password Manager app was that the process follows a continuous improvement cycle, exposing the flaws and faults of the system faster. It also simplifies the task by breaking it into smaller iterations and hence it is a quick method.

The meaning of Agile is swift or versatile. "Agile process model" refers to a software development approach based on iterative development. Agile methods break tasks into smaller iterations, or parts do not directly involve long term planning. The project scope and requirements are laid down at the beginning of the development process. Plans regarding the number of iterations, the duration and the scope of each iteration are clearly defined in advance.

Each iteration is considered as a short time "frame" in the Agile process model, which typically lasts from one to four weeks. The division of the entire project into smaller parts helps to minimize the project risk and to reduce the overall project delivery time requirements.

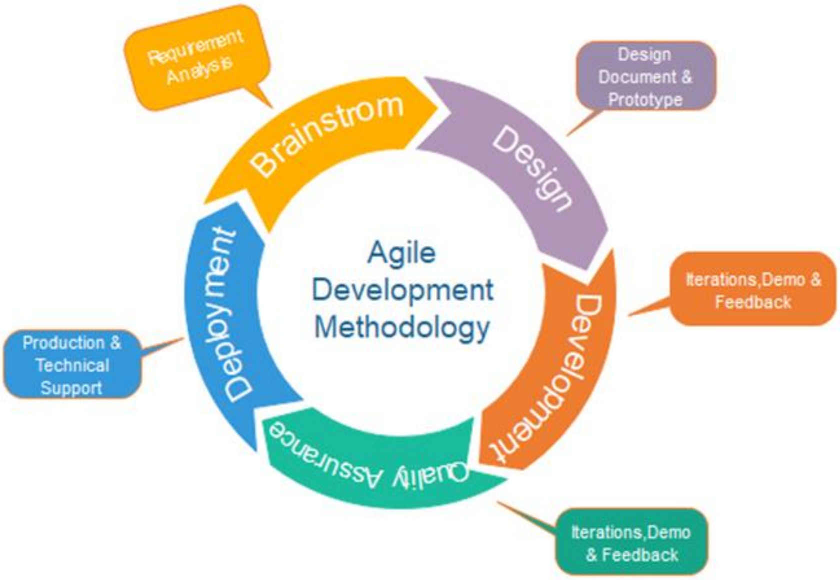

Requirement Analysis
Brainstrom
Design Document & Prototype
Design
Iterations,Demo & Feedback
Development
Iterations,Demo & Feedback
Quality Assurance
Deployment
Production & Technical Support
Agile Development Methodology

Figure 2.1.1 Agile Model

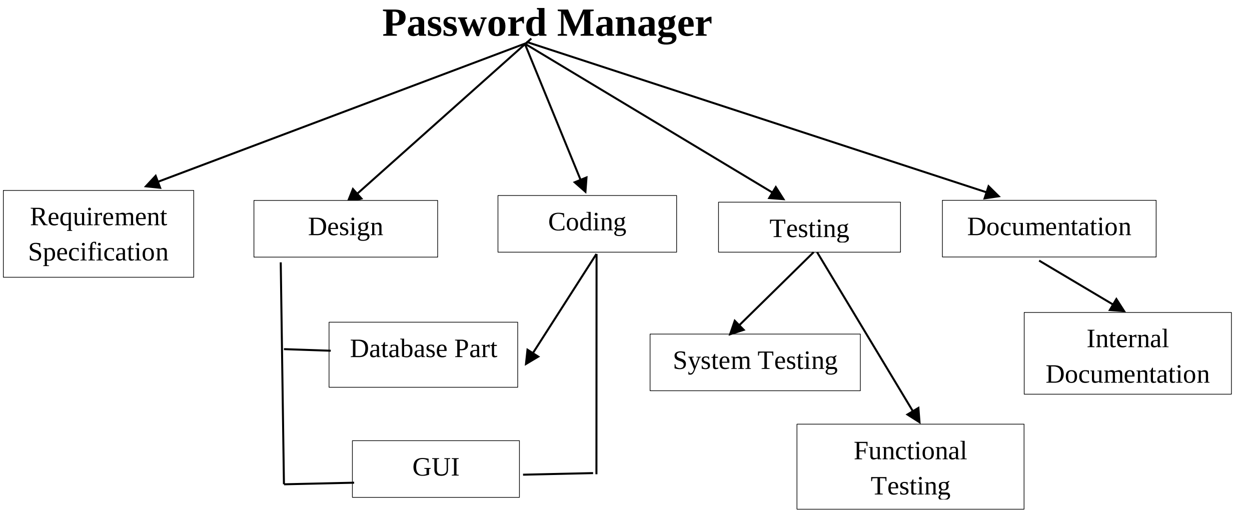
Advantages:

- Customer satisfaction is rapid, continuous development and delivery of useful software.
- Product is developed fast and frequently delivered (weeks rather than months.)
- It continuously gave attention to technical excellence and good design.
- Daily and close cooperation between business people and developers.
- Regular adaptation to changing circumstances.
- Even late changes in requirements are welcomed.

Disadvantages:

- It is not useful for small development projects.
- There is a lack of intensity on necessary designing and documentation.
- It requires an expert project member to take crucial decisions in the meeting.
- Cost of Agile development methodology is slightly more as compared to other development methodology.
- The project can quickly go out off track if the project manager is not clear about requirements and what outcome he/she wants.

2.2 Work breakdown structure:



2.3 Roles and Responsibilities:

The following table lists the roles and responsibilities necessary to complete the project.

Roles	Responsibility
Project Manager (Bhavya Desai)	Plans, organizes, and coordinates the team project. Schedules and prepares team meetings. Resolves conflicts. Works as a liaison between team members. Monitors and reports the weekly status of the team. Ensures that project deliverables are met.
Requirement Analyst (Yash Mistry)	Identifies, prioritizes, and documents requirements that satisfy the customer's needs. Clearly communicates the requirements to the Application Developers. Defines acceptance criteria for the completion of the solution.
Application Designer (Bhavya Desai)	Designs a solution to the problem that satisfies the requirements. Determines the data needs for the solution. Determines what hardware and tools are necessary. Assists the Technical Writers in documenting the design.
Application Developer (Yash Mistry)	Develops the software solution. Determines the data needs for the solution. Determines what hardware and tools are necessary. Fixes bugs found by the Quality Assurance Testers. Assists the Technical Writers in creating online help, and writing the user manual.

3. Project Estimates

3.1 Estimation Technique Used:

We have used Constructive Cost Model (COCOMO) and it is procedural software cost estimation model and also used for estimating effort and schedule for software projects.

3.2 Efforts, Resource Cost and Project Duration Estimation:

(Assumption: Our project falls in the category of semi-detached software project)

INPUT: The no. of lines: 20,000

Suppose LOC (line of code) =20,000

Effort E = $a_{b\ k\ LOC}{}^{b}{}_{b}$
= 3.0 (20) 1.12
= 86

Where E = Effort in person – month
D = Development time in months
kLOC = Estimated numbers of Line of Code

Duration time D = c_b * $(E)^{d}{}_{b}$
= 2.5 (86) 0.35
= 12 months

Person Estimation P = (E / D)
= (86/12)
= 7 per month

4. Managerial Process

4.1 Project Dimension:

Project Dimension	Fixed	Flexible
Resources	×	
Hours Worked		×
Schedule	×	
Scope		×

Table 4.1.1 Project Dimension

The number of resources for the project is fixed. The scope and the hours worked are the dimensions that can be adjusted if necessary.

4.2 Assumptions, Dependencies, and Constraints:

The project will be developed under the following assumptions and dependencies:

- It is assumed that the hardware design will work correctly with the 3rd party operating system and the developed software.
- A web browser to test the source code's output is required.
- A deployment platform.

The project will be developed under the following constraints:

- All the inputs should be checked for validation and message should be given for the improper data. The invalid data are to be ignored and error messages should be given to the user.
- Details provided by the user during his sign up should be stored in database. While adding the products to the system, mandatory fields must be checked for the validation whether the user has filled appropriate data in this mandatory fields or not.
- The user must have provided valid and legal identity proof.
- The full payment must have been done at the time of registration.

4.3 Risk Management

The project doesn't contain any major risk which may crash the system but though there are some risk which may affect the system. The risks which may affect the system are as below:

1. **Project Risk**:

Project risks concern differ forms of budgetary, schedule, personnel, resource, and customer-related problems. It is very tough to control something which cannot be identified such as increase in personnel required or budget due to new customer requirements.

2. Technical Risk:

Technical risks concern potential method, implementation, interfacing, testing, and maintenance issue. It also consists of an ambiguous specification, incomplete specification, changing specification, technical uncertainty, and technical obsolescence. For example, the development of new more secure technologies which pose a potential risk of making the system gradually obsolete.

❖ **Risk Analysis**:

1. Security
2. Maintenance
3. Access Loading
4. Database Crash

4.4 Project Tracking and Control Plan:

Sr. No.	Person	Task	Target
1.	Yash Mistry	Design Canvas	3 Days
2.	Bhavya Desai	Prototype Designing	10 Days
3.	Yash Mistry	Prototype Review	2 Days
4.	Bhavya Desai	Web Design	3 Weeks
5.	Yash Mistry	Web Interaction	4 Weeks
6.	Bhavya Desai	Composition	3 Weeks
7.	Yash Mistry	Database functionality	2 Months

Table 4.1.2 Monitoring and Controlling

5. Technical Process

Password Manager will be developed using an Agile model. It is very flexible, speedy and its processes follows a continuous improvement life cycle. In it, each iteration involves a team working through a full software life cycle including planning, requirement analysis, design, coding and testing.

The system will be a web- based application supporting the following browsers:

- Google chrome
- Mozilla Firefox
- Internet Explorer
- Opera
- Safari

It will be developed using the following technologies:

Front end:

- React.js
- HTML
- CSS
- Bootstrap

Back end:

- Node.js and Express.js for server-side scripting.
- It will run on MySQL Database, which is the world’s most popular open source database. MySQL is freely available, easy to use and cross platform.

6. Task Dependency, Scheduling and Other Plans

6.1 Dependencies:

The following sections explain what tasks are to be done and by whom. This plan may become more detailed as the project progresses.

Task	Dependencies
Software Requirement Specification (SRS)	User requirements
Software Project Management Plan (SPMP)	SRS
Coding	SPMP, SRS, Final design of all Modules
Test Scenarios	SPMP, SRS, Testable Prototype
User Guide	SPMP, SRS
Final Presentation	SPMP, SRS, Working prototype

Table 6.1.1 Task Dependency

6.2 Resource Requirement:

The project team will be do all of the work on the project. No additional resources will be required. The following table details the expected person hours for each major task.

Task	Efforts in person hours
Software Project Management Plan (SPMP)	50
Software Requirement Specification (SRS)	70
Coding	75
User Guide	32
Final Presentation	23
Total Effort	**250**

Table 6.1.2 Resource Requirement

6.3 Quality Assurance Plan:

Sr. No.	Person	Module	Duration	Report	Submit
1.	Bhavya Desai	1	2 Day	Quality Report of modules	Yash Mistry
		2	5 Day		
		3	10 Day		
2.	Yash Mistry	4	2 Day		Bhavya Desai
		5	3 Day		
		6	1 Day		
3.	Bhavya Desai	7	2 Day	Quality Report of Functionality and modules	Yash Mistry
		8	4 Day		
		9	5 Day		
4.	Yash Mistry	10	6 Day		Bhavya Desai
		11	7 Day		
		12	10 Day		

Table 6.1.3 Quality Assurance Plan

6.3 Schedule:

The following **Gantt chart** details the major tasks along with their expected start and finish dates. The duration on the chart is shown in days. The duration doesn’t necessarily reflect the amount of effort that will expended for that task. The effort was shown previously.

These dates make up the baseline against which future progress will be measured. Actual start and finish dates will be recorded as progress occurs on the project. These dates will be compared to the original baseline to confirm that project deliverables and milestones will be completed in accordance with the project plan.

This plan will be closely monitored by Bhavya Desai and Yash Mistry to ensure that the project is completed on time. Both team members will be involved with any decisions to reassign resources.

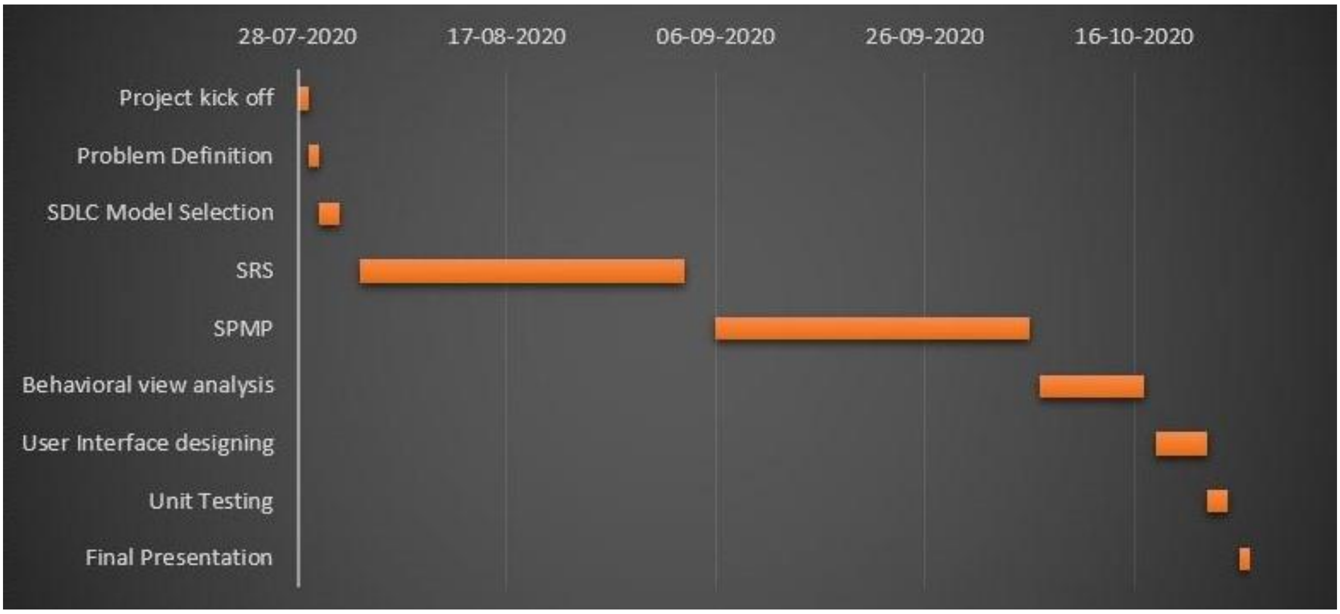


Figure 6.3.1 Task Schedule (Gantt Chart)