A PROJECT REPORT

ON

“Password Manager”

(Final Report)

BY:- PRAKHAR SINGH

Domain:- PYTHON

DECLARATION

I, **Prakhar Singh**, hereby declare that the presented report of the internship titled “**Password Manager”** is uniquely prepared by me after the completion of one months’ work at Unicoverge Technologies Pvt. Ltd.

I also confirm that the report is only prepared for my academic requirement, not for any other purpose. It might not be used in the interest of the opposite party of the corporation.

Signature of Student

ABSTRACT

"Password Manager" is a small-scale project used for storing your login information for all the websites you use and generating random passwords for users. The user will have the access to add, update and remove his details and he can fetch his data whenever needed.

It's an GUI based application simple and easy to use. It has a lot's of feature which makes user experience better. In this application Passwords can be stored for various applications and details required for maintaining the Passwords are application or website name, username, password and email id.

You would not get any problem while using it because it's simple and user friendly UI will make your work easy, faster and gives you a better result.

TABLE OF CONTENTS

Table of Contents

CHAPTER 1

INTRODUCTION

INTRODUCTION

1.1 Basic Introduction:

Password Manager is an GUI based application that allows users to store and generate random passwords. It is created in Python using Tkinter and Mysql database to manage passwords for several application. A password manager is a program that houses all your passwords, as well as other information, in one convenient location with one master password.

The benefits of using Password Manager are:

- A Password Manager will do the work of creating the complicated passwords you need to help protect your online accounts.
- You need to remember only the password manager's password. That single password will give you access to all of your others.

Not only do password managers help securely house your passwords, but they can also generate passwords that are unique and complex, which makes them more difficult to crack or guess. It also simplifies your life by making account access easier for you and more diffifcult for hackers. You don't have to memorize any passwords except for the password to your password manager. That means you can actually follow unpleasant butuseful security advice, like never reusing a password and always using long, strong and complex passwords.

1.2 Objectives and scope:

In this project, we will build an application which will store the User's password, as well as other information, in one convenient location with one master password. Python Features and methods are used to implement in this project. Today, people have a large number of passwords for social media sites, work logins, shopping pages, online banking and much more. While it is important to use strong passwords and to use different passwords on each site, it can be a difficult task to remember all them. With a password manager, you simply enter the GUI app, provide the master password you set for the password manager in that software, then log in to the GUI app and the username and password will be stored for you. However, the primary purpose of this project to make Password Manager application user-friendly so that any individual can interact with the system.

The main scope and deliverables of the project would be to:

- Understand and prepare detailed user requirement and specifications.
- Prepare high level and detailed design specifications of the system.
- Prepare Test Plan & Test cases.
- Develop the system and coding.
- Perform unit testing, integration testing and system testing.
- Demonstrate a bug free application after suitable modification, if needed.
- Develop a GUI application using Python Tkinter.

1.3 Tools And Technologies:

Python: -

Python is an interpreted high-level and general-purpose programming language. Its language constructs and object-oriented approach aim to help programming write clear, logical code for small and large-scale projects.
Python is dynamically-typed and garbage-collected. It supports structured, Object-oriented and functional programming.
Python was created in the late 1980's, and first released in 1991, by Guido Van Rossum at Centrum Wiskunde & Informatica in the Netherlands as a successor to ABC programming language.

Python GUI Framework:

A Graphical User Interface (GUI) is the first thing your user sees and interacts with when he opens your application or website. Having a good GUI goes a long way in increasing your platform's reputation and user count. A user interface usually includes a host of visual elements like icons, buttons, graphics, displayed text, and several other forms of input, like checkbox, text input boxes, and such.

Tkinter :

Often referred to as the go-to GUI toolkit by a majority of Python developers, **Tkinter** was created to equip modern developers with a standard interface to the Tk GUI toolkit with its Python bindings. The name Tkinter comes from Tk interface. Tkinter was written by Fredrik Lundh and it is free software released under a Pyhton license.

a) **Window**:

This term has different meanings in different contexts, but in general it refers to a rectangular area somewhere on the user's display screen.

b) **Widget**:

The generic term for any of the building blocks that make up an application in a graphical user interface.

1. Core widgets: Containers: frame, labelframe, toplevel, paned window. Buttons: button, radiobutton, checkbutton (checkbox), and menubutton. Text widgets: label, message, text. Entry widgets: scale, scrollbar, listbox, slider, spinbox, entry (singleline), optionmenu, text (multiline), and canvas (vector and pixel graphics).
2. Tkinter provides three modules that allow pop-up dialogs to be displayed: tk.messagebox (confirmation, information, warning and error dialogs)

c) **Themed Tk**:

This allows Tk widgets to be easily themed to look like the native desktop environment in which the application is running, thereby addressing a long-standing criticism of Tk (and hence of Tkinter). Some widgets are exclusive to ttk, such as the combobox, progressbar and treeview widgets.

d) **Frame**:

In Tkinter, the Frame widget is the basic unit of organization for complex layouts. A frame is a rectangular area that can contain other widgets.

e) **Pack**:

Pack it into position so it becomes visible. Developers also have the option to use ‘ .grid() ’ (row=*int*, column=*int* to define rows and columns to position the widget, defaults to 0) and ‘ .place() ’ (relx=*int or decimal*, rely=*int or decimal*, define coordinates in the frame, or window) .

Tkinter is included with standard Linux, Microsoft Windows and Mac OS X installs of Python.

MySQL:

MySQL is an open-source relational database management system (RDBMS). language for relational database management systems. It is used for storing, manipulating and retrieving data in databases. Its name is a combination of "My", the name of co-founder Michael Widenius's daughter, and "SQL", the abbreviation for Structured Query Language. A relational database organizes data into one or more data tables in which data types may be related to each other these relations help structure the data. SQL is used to create, modify and extract data from the relational database, as well as control user access to the database.

CHAPTER 2
SYSTEM ANALYSIS

SYSTEM ANALYSIS

2.1 Preliminary Analysis & Information Gathering :

System analysis is the process of gathering and interpreting facts, diagnosing problems and using the information to recommend improvements on the system. System analysis is a problem-solving activity that requires intensive communication between the system users and system developers.
System analysis or study is an important phase of any system development process. The system is viewed as a whole, the inputs are identified and the system is subjected to close study to identify the problem areas. The solutions are given as a proposal. The proposal is reviewed on user request and suitable changes are made. This loop ends as soon as the user is satisfied with the proposal.

2.2 Input /Output:

Input Design:

Input design is the link that ties the information system into the world of its users. The input design involves determining the inputs, validating the data, minimizing the data entry and provides a multi-user facility. Inaccurate inputs are the most common cause of errors in data processing. Errors entered by the data entry operators can be controlled by input design. The user-originated inputs are converted to a computer-based format in the input design. Input data are collected and organized into groups of similar data. Once identified, the appropriate input media are selected for processing. All the input data are validated and if any data violates any conditions, the user is warned by a message. If the data satisfies all the conditions, it is transferred to the appropriate tables in the database. In this project the student details are to be entered at the time of registration. A page is designed for this purpose which is user friendly and easy to use. The design is done such that users get appropriate messages when exceptions occur.

Output Design:

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system, can use easily and effectively.

2. Select methods for presenting information.

3. Create document, report, or other formats that contain information produced by the system.

4.The output form of an information system should accomplish one or more of the following objectives.

i. Convey information about past activities, current status or projections of the future.
ii. Signal important events, opportunities, problems, or warnings.
iii. Trigger an action. Confirm an action.

2.3 Feasibility Study:

Feasibility study is most important and it concerns great to the student and faculty which new system is to be developed. It includes the study of success and level of satisfaction of the user will get from the system. Feasibility means possibilities (to some extent), i.e., this study the possibility of implementation of new system as a substitute to old system removing all shortcomings and including all new requirements.

2.4 System Requirement Specification:

Hardware Requirements

Memory Space:

Minimum	-	32 MB
Recommended	-	64 MB

PROCESSOR - Intel Pentium IV, 1GHZ or above

RAM - 256MB or above

Software Requirements

OPERATING SYSTEM - WindowsXP Professional or above

DEVELOPING LANGUAGE -

FRONT END	-	PYTHON (Tkinter)
BACKEND	-	PYTHON

DATABASE - MY SQL

2.5 SOFTWARE ENGENEERING MODEL USED:

This document plays a vital role in the development of life cycle (SDLC) as it describes the complete requirement of the system. It means for use by developers and will be the basic during testing phase. Any changes made to the requirements in the future will have to go through formal change approval process.

SPIRAL MODEL was defined by Barry Boehm in his 1988 article, “A spiral Model of Software Developed and Enhancement. This model was not the first model to discuss iterative development, but it was the first model to explain by the iteration models.

As originally envisioned, the iterations were typically 6 months to 2 years long. Each phase starts with a design goal and ends with a client reviewing the progress thus far. Analysis and engineering efforts are applied at each phase of the project, with an eye toward the end goal of the project.

The steps for Spiral Model can be generalized as follows:

i. The new system requirements are defined in as much details as possible. This usually involve interviewing a number of users representing all the external and internal users and other aspects of the existing system.

2. A preliminary design is created for the new system.

3. A first prototype of the new system is constructed from the preliminary design. This is usually a scaled-down system, and represents approximation of the characteristics of the final product.

4. A second prototype is evolved by a fourfold procedure:

5. Evaluating the first prototype in terms of its strengths, weakness, and risks

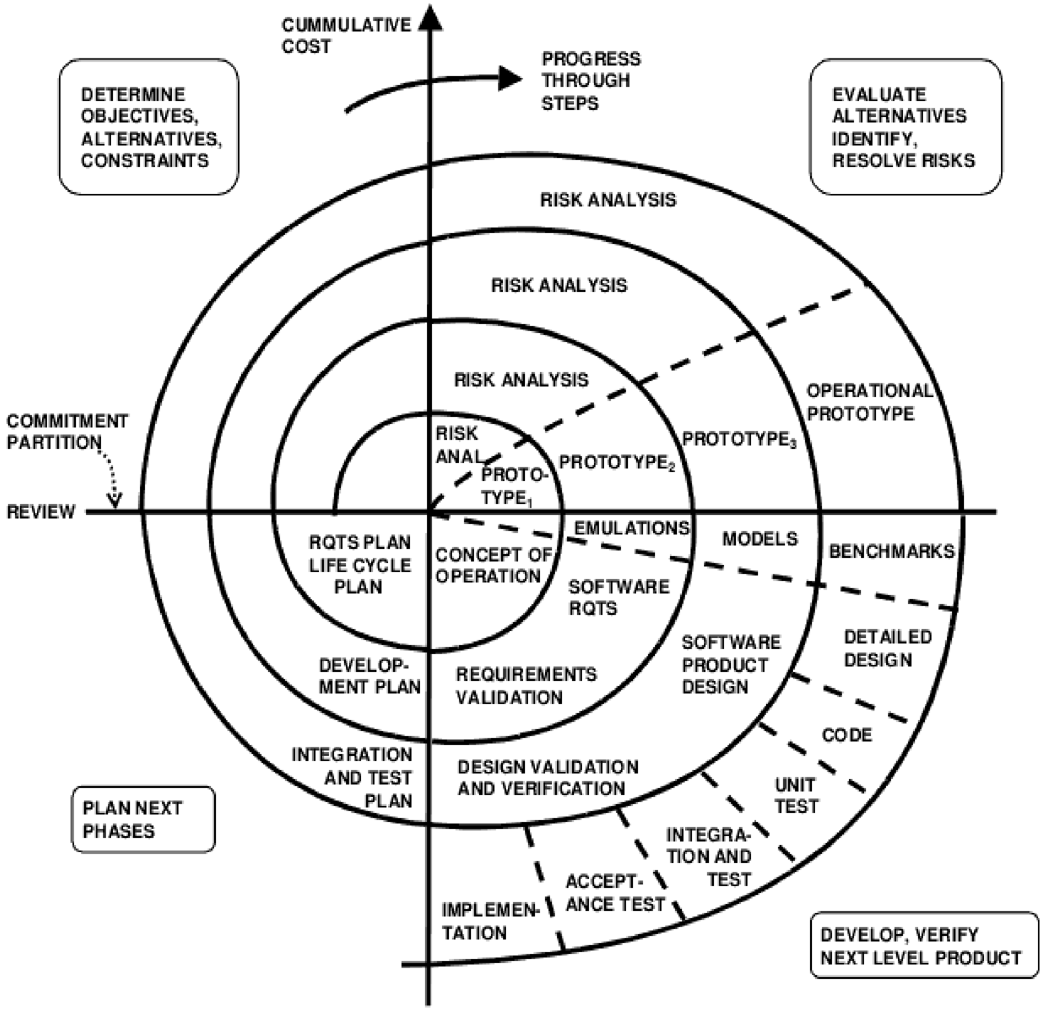
6. Defining the requirements of the second prototype.

7. Planning and designing the second prototype.

8. Constructing and testing the second prototype.

9. The final system is constructed, based on the refined prototype.

The following diagram shows how a spiral model act like:



2.6 Cost Estimations

Economic analysis is the most frequently used technique for evaluating the effectiveness of a proposed system. It is commonly known as cost/benefit analysis: the procedure is to determine the benefits and savings that are expected from a proposed system and compare them with cost. If benefits outweigh cost, a decision is taken to design and implement the system. Otherwise, further justification or alternative in the proposed system will have to be made if it is to have a chance of being approved. This is an ongoing effort that improves in accuracy at each phase of the system life cycle. Total estimated cost of the project.

Financing of the project in terms of its capital structure, debt e/unity ratio and promoters share of total cost.

Existing investment by the promoter in any other business

Projected cash flow and profitability.

2.7 Project scheduling

Project scheduling is a mechanism to communicate what tasks need to get done and which organizational resources will be allocated to complete those tasks in what time frame. A project schedule is a document collecting all the work needed to deliver the project on time.

A project is made up of many tasks, and each task is given a start and end (or due date), so it can be completed on time. Likewise, people have different schedules, and their availability and vacation or leave dates need to be documented in order to successfully plan those tasks.

Project scheduling provides the following benefits:

- Assists with tracking, reporting on, and communicating progress.
- Ensures everyone is on the same page as far as tasks, dependencies, and deadlines.
- Helps highlight issues and concerns, such as a lack of resources.
- Helps identify task relationships.
- Can be used to monitor progress and identify issues early.

CHAPTER 3
SYSTEM DESIGN

SYSTEM DESIGN

3.1 Project planning:

Project management begins with planning, which perhaps the single largest responsibility of the project management Proper planning is recognized as a critical ingredient for a successful project. The project plan provides the fundamental basis for project management.

A software plan is usually produced before the development activity begins and is updated as development proceeds and data about progress of the project become available. The major activities of project planning is :

- Cost estimation
- Schedule and Milestone determination
- Project-Staffing
- Quality control plans
- Controlling and monitoring plan

Cost and schedule estimation is the amount of total cost and time nedded for successfully completing the project. In addition, cost and schedule for the different activities of development process to be used are also estimated, as the development process only specifies the activities, not the cost and time requirement for them. In this project, the cost is calculated on the effort of the hardware and software cost.

Activities carried out are scheduling the project according to the time limit and obtaining the correct output.

The project plans for all the software quality assurance activities that needed to be performed in order to ensure the quality objectives are met. A project plan also provides methods for handling change and methods for monitoring a project. Project planning is the single most important management activity and output of this forms the basis of monitoring and control.

3.2 Modules:

a) **Pyperclip:** Pyperclip is the cross-platform Python module which is used for copying and pasting the text to the clipboard.

b) **Random:** The random module is a built-in module to generate the pseudo-random variables. It can be used perform some action randomly such as to get a random number, selecting a random elements from a list, shuffle elements randomly, etc.

c) **PIL**: PIL is the Python Imaging Library adds image processing capabilities to your Python interpreter. This library provides extensive file format support, an efficient internal representation, and fairly powerful image processing capabilities.

d) **SignUp:** With the SignUp module, you can quickly and easily create a Account for Password Manager GUI app. So you can access its feature.

e) **Login:** This module validates the user, and returns an error statement if it doesn’t find the specified user in the data base.

f) **User:** User privileges are provided to the user, where he is able to to store and generate random passwords.

i. **Add Data:** This module provides features for adding the password or other details of user in the database. The user must register/signup himself first, so that he is able to login and store or generate a random password for his online accounts.

ii. Update Data: This module enables the user to update his existing data. If he wants update his data that he has already stored in the database.

iii. Remove Data: When an user wants to remove his data from his database, he is able to do it with the help of this module and is able to successfully remove that specific data.

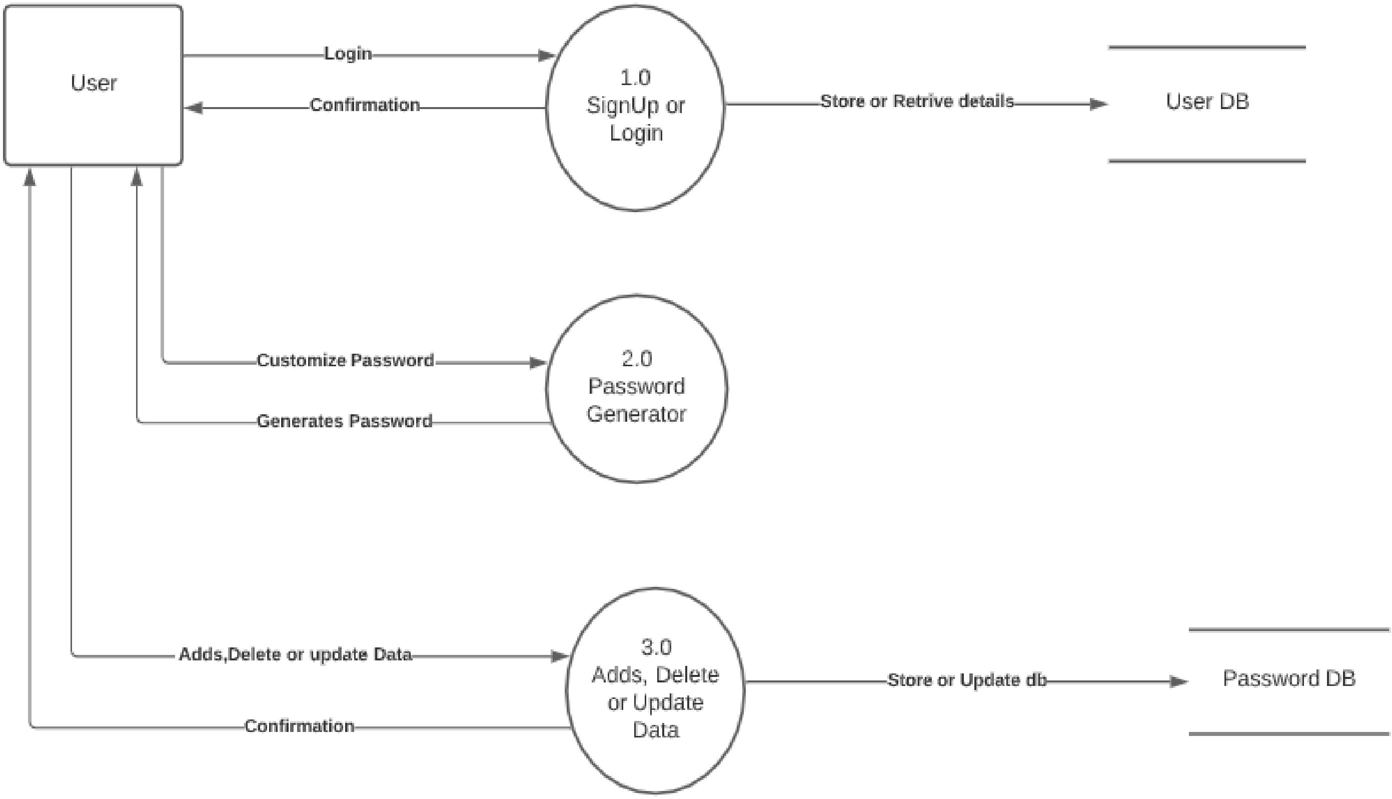
g). Password Generator: This module generates passwords based on the customization that you set to create an unpredictable strong password for your accounts. It helps to create a random and customized password for users that helps them to create a strong password which provides greater security.

h). MySQL: MySQLdb is an interface to the popular MySQL database server for Python.

The design goals are:

- Compliance with Python database API version 2.0
- Thread-safety
- Thread-friendliness (threads will not block each other)

3.3 Data Flow Diagram:



3.4 E-R Diagram:

The Graphical representation of organizational system elements and the association among the elements is called ER Diagram.

a) The elements make up a system referred to entities.

b) A relationship is the association that describes the interaction between entities.

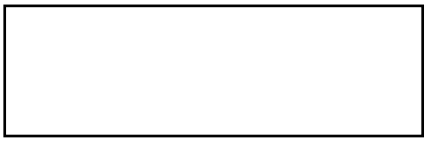
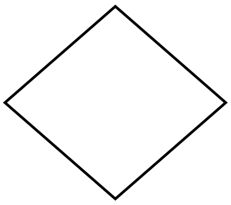
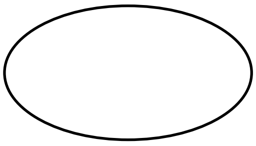
c) An ER Diagram may also indicate the cardinality of the relationship.

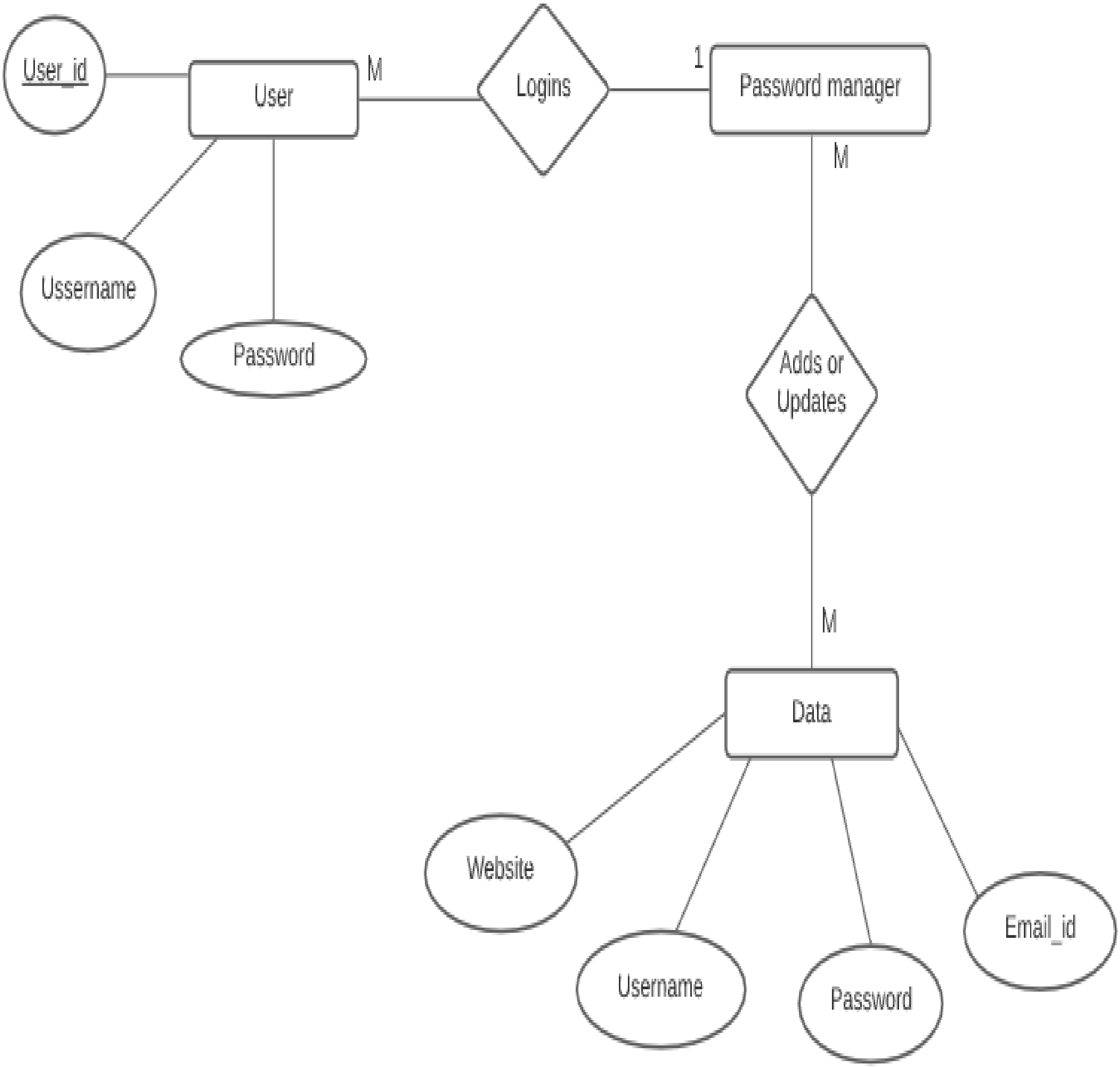
d) The ER notation also allows for distinguishing different types of entities.

e) A plain rectangle is used for what is termed as Fundamental Entity which is a person, place or thing

f) Attribute is a property an entity uniquely.

g) Relationship specifies how two entities share information in the database.

SYMBOL	MEANING
	***ENTITY***
	***RELATIONSHIP***
	***ATTRIBUTE***


User_id
User
M
Logins
1
Password manager
M
Ussername
Password
Adds or Updates
M
Data
Website
Username
Password
Email_id

3.5 Database Design:

useraccounts

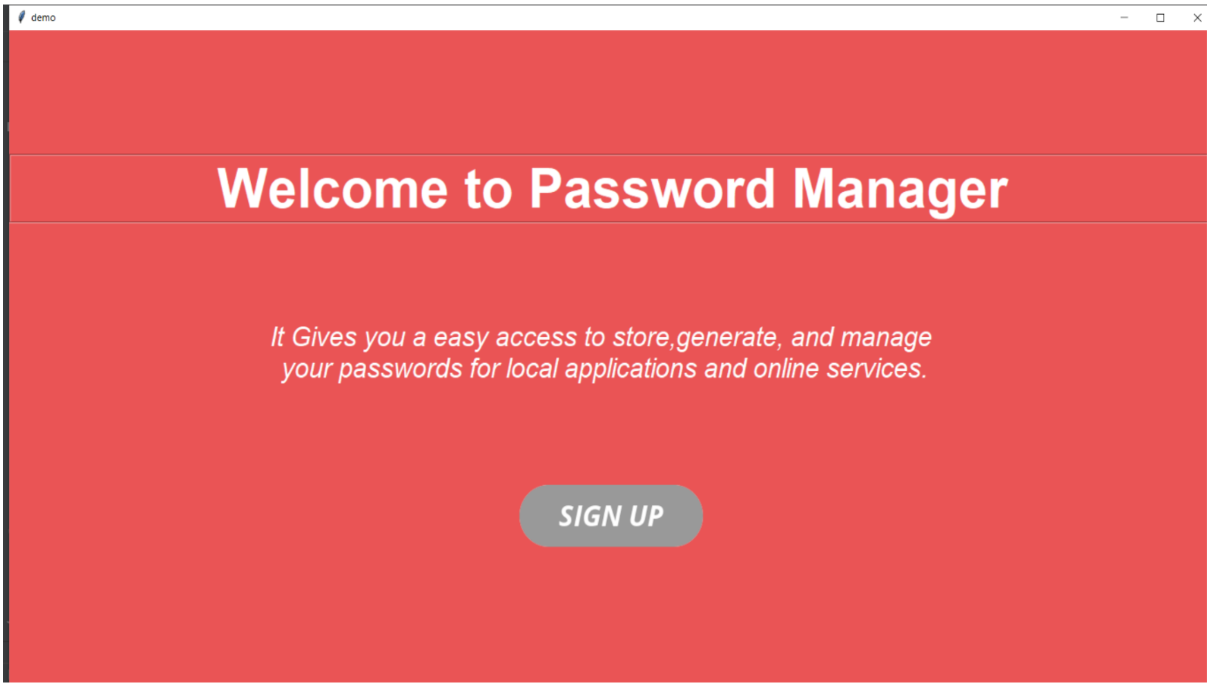
user_id	int
Username	varchar(50)
Password	varchar(30)

userdata

Website	varchar(50)
Username	varchar(50)
Password	varchar(50)
Email_id	varchar(50)

3.6 Screenshots:

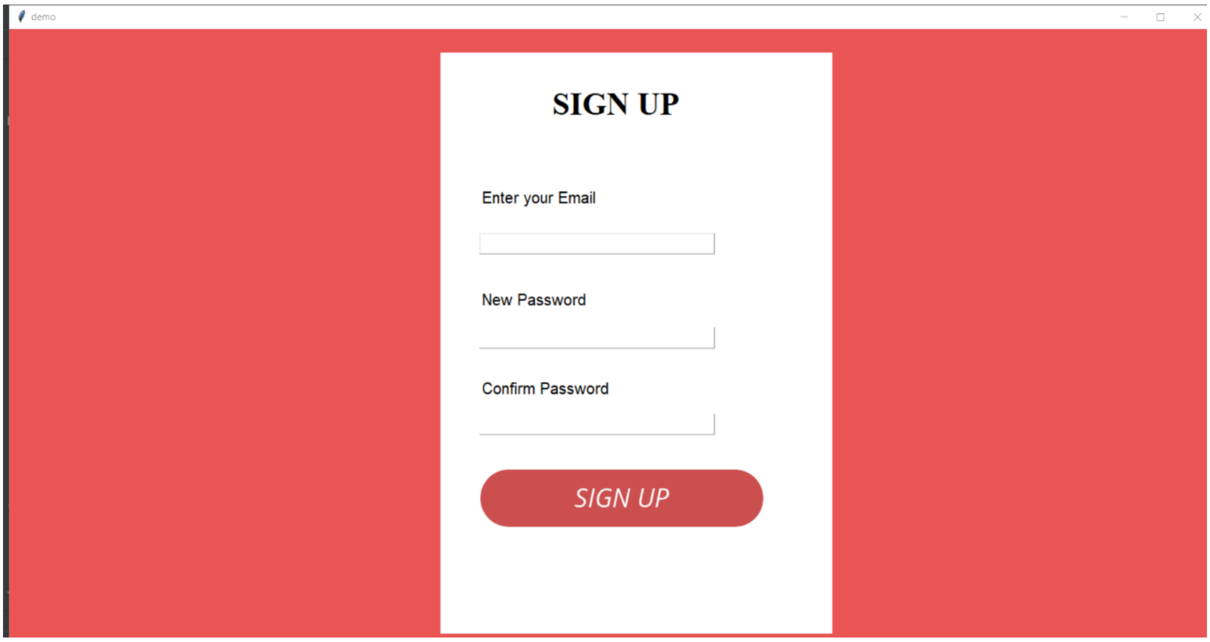
Welcome Page:



Welcome to Password Manager

It Gives you a easy access to store,generate, and manage your passwords for local applications and online services.

SignUp Page:



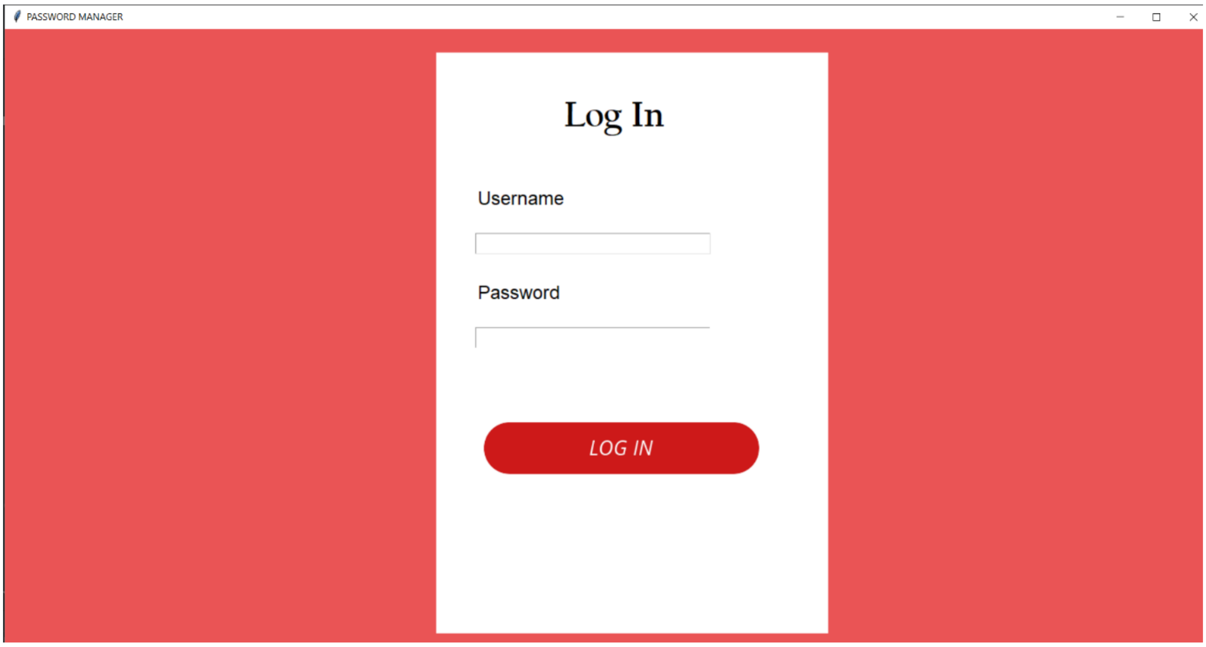
SIGN UP

Enter your Email

New Password

Confirm Password

Login Page:

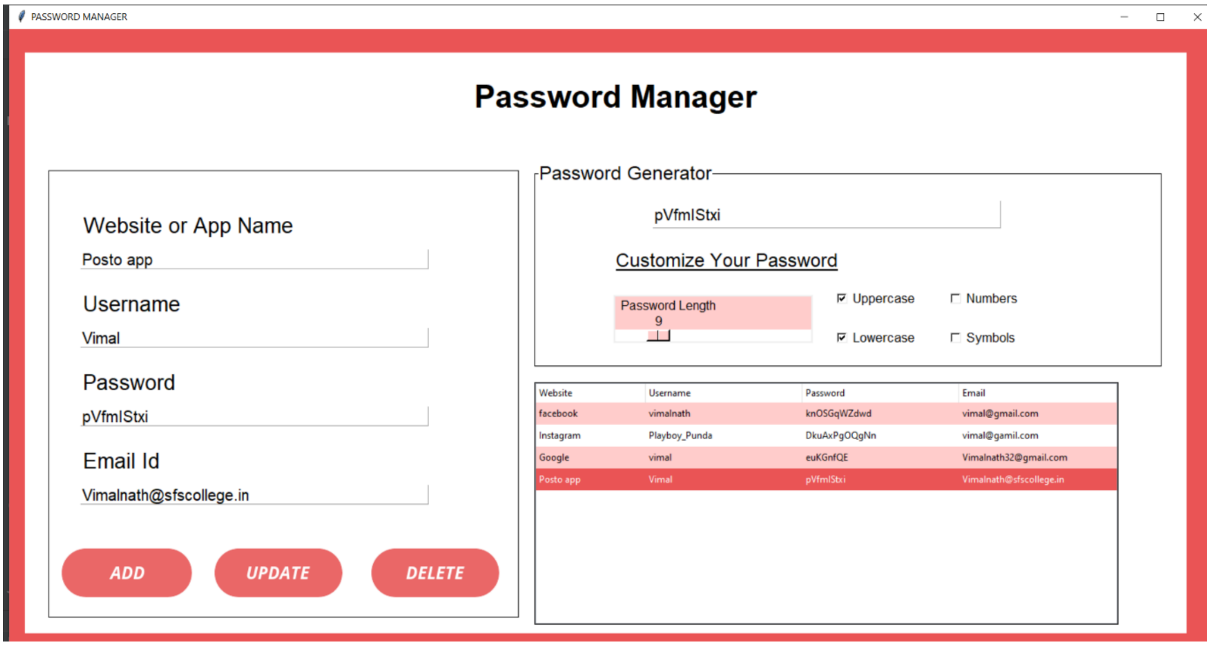


Log In

Username

Password

Password Manager:



Password Manager

Website or App Name

Posto app

Username

Vimal

Password

pVfmlStxi

Email Id

Vimalnath@sfscollege.in

Password Generator

pVfmlStxi

Customize Your Password

Password Length
9

☑ Uppercase ☐ Numbers

☑ Lowercase ☐ Symbols

Website	Username	Password	Email
facebook	vimalnath	knOSGqWZdwd	vimal@gmail.com
Instagram	Playboy_Punda	DkuAxPgOQgNn	vimal@gamil.com
Google	vimal	euKGnfQE	Vimalnath32@gmail.com
Posto app	Vimal	pVfmlStxi	Vimalnath@sfscollege.in

CHAPTER 4

TESTING

SYSTEM TESTING

Testing is an important phase encountered in any developed product or framework is the testing phase. It is because, the developed product should be free from errors and it should be validated for accuracy. The product should work under normal conditions as long as the user gives proper inputs and therefore it should be checked for its robustness and should withstand and inform the users about the erroneous input.

The testing phase involves testing the system using various test data. Preparation of test data plays a vital role in the system testing. After preparing the test data, the system is tested using those test data. Errors are found and corrected by using the following testing steps and corrections are recorded for future reference. Thus a series of testing is performed on the system before it is ready for implementation.

Testing is applied at different levels in the development lifecycle. Each level of testing done is different in nature and has different objectives at each level. The focus of all levels of testing is to find errors, but different types of errors are looked for at each level.

The quality of system is confirmed by the thoroughness of its testing. Duration and cost of testing and debugging is a significant fraction of the system development cycle and hence influences overall productivity during the development. In this phase, the errors in the programs or modules are localized and modifications are done to eliminate them. The testing makes a logical assumption that all parts of the system work efficiently and the goal is achieved.

VERIFICATION & VALIDATION TESTING

The purpose of the requirement phase is to verify all behavioral characteristics and performance requirements. Validation is achieved through a series of testing that demonstrate conformity with requirements. Both the plan and the procedures are designed to ensure that all the functional requirements are satisfied in the program.

- All behavioral characteristics are achieved.
- All performance requirements are attained.

BLACK-BOX TESTING:

Black-box testing is an approach to testing where the tests are derived from the program or component specification. The system is a ‘block box’ whose behavior can only be determined by studying its inputs and the related outputs. Another name for this is functional testing because the tester is only concerned with the functionality and not the implementation of the software. The following methods are commonly used:

The following methods are commonly used:

- Equivalence partitioning
- Boundary-value analysis
- Error guessing.

Equivalence Partioining:

Equivalence partitioning is a systematic process that identifies, on the basis of whatever information is available, a set of interesting classes of input conditions to be tested, where each class is representative of a large, set of other possible tests.

Boundary-value analysis:

Boundary-value analysis is a test case design technique that complements equivalence portioning. Rather than selecting any element of an equivalence class, BVA leads to the selection of text cases at the "edges" of the code.

Error Guessing:

Error guessing is an ad hoc approach, based on intuition and experience, to identify tests that are considered likely to expose errors. The basic idea is to make a list of possible errors or error-prone situations and then develop tests based on the list.

- Empty or null lists/ strings
- Zero instances/ occurrences
- Blanks or null characters in strings
- Negative numbers.

Features to be tested

- Verify that the entries are of the correct format.
- No duplicate entries should be allowed.
- All links should take the user to the correct page.

WHITE-BOX TESTING

White box testing is conducted to ensure that internal operations are performed according to specifications and all internal components are adequately exercised. This type of testing is called as white-box testing or glass-box testing. White-box testing of software is predicated on close examination of procedural detail. Providing test cases that exercise specific sets of conditions and/or loops tests logical paths through the software. The "status of the program" may be examined at various points to determine if the expected or asserted status corresponds to the actual status.

WHITE BOX TEST IS DONE TO

- Guarantee that all independent paths within a module have been exercised at least once.
- Exercise all logical decisions on their true or false sides
- Execute all loops at their boundaries and within their operational bounds.
- Exercise internal data structure to assure their validity.

Testing was done for the grouping row and column pixels.

THERE ARE SOME BASIC FORMS OF LOGIC COVERAGE

- Statement coverage: Each statement is executed at least once.
- Decision coverage: Each statement is executed at least once; each decision takes on all possible outcomes at least once.
- Condition coverage: Each statement is executed at least once; each

condition in a decision takes on all possible outcomes at least once.

- Decision/condition coverage: Each statement is executed at least once; each decision takes on all possible outcomes at least once; each condition in a decision takes on all possible outcomes at least once.
- Multiple/condition coverage: Each statement is executed at least once; all possible combinations of condition outcomes in each decision occur at least once.

- Whether correct statistical values are getting from the given grouped rows, grouped pixels.

UNIT TESTING

Unit testing focuses on the verification effort of the smallest unit of design module. Attention is diverted to individual modules, independently to locate errors. This has enabled the detection of errors in coding and logic. The various modules of the system are tested in unit testing method. Using the detailed description as a guide, important control parts are tested to uncover errors within the boundary of the module. The relative complexity of tests and the error detected as a result is limited by the constrained scope established for unit testing. This test focuses on each module individually, ensuring that it functions properly as a unit, and hence the name Unit Testing.

INTEGRATION TESTING

The program as a whole is integrated into a single component and checked with test data. This is to check that the whole program works together correctly. The various modules are integrated first. Then the system is tested as a whole. The behavior of each module with another is identified in this testing. Any misbehavior will lead to dangerous problems or errors in the system. All these problems can be filtered off using this testing.

We combined all the modules into a single component and tested for integrity. We found the crude form of our project is meeting the basic requirements like functioning of the application based on the requirement.

- Bottom-up Integration: low-level components are integrated and tested before the higher-level components have been developed.
- Top-down Integration: The high-level components of a system are integrated and tested before their design and implementing has been completed.

The steps in top-down integration are:

- Begin with the top module in the execution hierarchy.
- Stub modules are produced, and some may require multiple versions.
- Stubs are often more complicated than they first appear.
- The next module to be tested is any module with at least one previously tested super ordinate module.

After a module has been tested, an actual module and its required stubs replace one of its stubs.

The following points must considered by a software engineer to clear the potential interfacing problems in the process of designing the software.

1. Design error-handling paths that test all information coming from other elements of the system.

2. Conduct a series of tests that simulate bad data or other potential errors at the software interface.

3. Record the results of tests to use evidence if finger pointing occurs.

4. Participate in the planning and design of system testes to ensure that software is adequately tested.

ACCEPTANCE TESTING

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

TEST RESULTS:

All the test cases mentioned above passed successfully. No defects encountered.

CHAPTER 5
IMPLEMENTATION
&
MAINTENANCE

SYSTEM IMPLEMENTATION & MAINTENANCE

Implementation means converting a new system or revised system into an operational one. Conversion is main aspect of implementation. It is the process of designing from the old system to the new one. Several procedures and documents are carried during the conversion phase. Implementation of a modified application to replace an existing one. Using the same computer this type of conversion is relatively easy to handle, provided there are no major changes in the files.

In implementation, the training for the system user, how to work with this software should be given. The development of operating procedure to repair and enhance the system should be done. The installation new computers with required hardware and software specification are to be performed, if the present system is working manually. If the hardware specification of the present system is below the recommended specification the existing system should be replaced with new one or up gradation of the existing system should be done. It is necessary to check whether the required application should be working properly.

Implementation is the stage in the project where the theoretical design is turned in to a working system and is giving confidence on the new system for the users, which will work efficiently and effectively. It involves careful planning investigation of the current system and its constraints. If the implementation neither is nor carefully planned it can cause confusion. Thus it can be considered as the most crucial stage in achieving a successful new system and in giving the users confidence that the new system will work be effective. It is during the implementation phase the system takes shape.

The main stages in the implementation are

- Planning
- Training
- System Testing

MAINTENANCE

Software development efforts result in the delivery of a software product which satisfies user requirements. Accordingly, the software product must change or evolve. Once in operation, defects are uncovered, operating environments change, and new user requirements surface. Software maintenance is an integral part of a software life cycle. The software maintenance is defined as the totality of activities required to provide cost effective support to software.

The software life cycle processes essentially depicts maintenance as one of the primary life cycle processes, and describes maintenance as the process of a software product undergoing modification to code and associated documentation due to a problem or the need for improvement. The objective is to modify the existing software product while preserving its integrity.

Software maintenance is the totality of activities required to provide cost-effective support to a software system. Activities are performed during the pre-delivery stage as well as the post-delivery stage. Pre-delivery activites include planning for post-delivery operations, supportabilty and logistics determionation

Post-delivery activities include software modification, training
and operating a help-desk.

There are number of reasons, why modifications are required like

- Client requirement
- Host Modifications
- Organization changes

Need for maintenance:

Maintenance is needed to ensure that the software continues to satisfy user requirements. Maintenance is applicable to software developed using any life cycle model (for example – spiral). The system changes due to corrective and non-corrective software actions. Maintenance must be performed in order to:

- Correct faults
- Improve the design
- Implementation enhancement
- Interface with other systems
- Perfecting existing functions

CONCLUSION

This project is designed for the users who surf through internet a lot and create a number of online accounts and they can't keep track or memorize all of their online acconts login details. That's where our Password Manager comes in use it helps them to houses all their passwords in encrypted database. Even it helps in generating a stronger, random and complex passwords which makes them more difficult to crack or guess. Thus, you can have easy access for your accounts and more difficult for hackers.

Future Enhancement:

This project has very vast scope in future. As the user's requirement is always going to be changed which is not static. Project can be updated as per the user requirement and it is very flexible in terms of expansion. The enhancement of the project can easily be made without changing the programming structure.

The following are the future enhancement of the project

- Enhancement of UI of application
- Converting the Desktop application into Web application.

REFERENCES

1) https://www.python.org/doc/

2) https://docs.python.org/3/library/tk.html

3) https://www.youtube.com/watch?v=YXPyB4XeYLA&t=15620s&ab_channel=freeCodeCamp.org

4) https://www.youtube.com/watch?v=3vsC05rxZ8c&t=12s&ab_channel=TechWithTim

5) https://www.udemy.com/course/100-days-of-code/

6) https://stackoverflow.com

SUBMISSION LINK

- **Code Submission Link:**

  https://github.com/Sinprakhar01/Password_manager