CONTENTS

ABSTRACT **I**

ACKNOWLEDGEMENT **II**

1. INTRODUCTION

1.1. PROBLEM DEFINITION **1**

1.2. OBJECTIVES **1**

1.3. METHODOLOGY TO BE FOLLOWED 1

1.4. EXPECTED OUTCOMES **2**

1.5. HARDWARE AND SOFTWARE REQUIREMENTS **2**

2. FUNDAMENTALS OF PYTHON

2.1. FUNDAMENTALS OF PYTHON **3**

2.2. CLASS AND OBJECT **4**

3. FUNDAMENTALS OF DBMS

3.1. INTRODUCTION 5

4. DESIGN

4.1. DESIGN GOALS 6

4.2. FLOW CHART 7

4.3. ALGORITHM FOR ADDING NEW APPLICATION 8

4.4. ALGORITHM FOR RETRIEVING USERNAME AND PASSWORD 9

5. IMPLEMENTATION

5.1. MODULE 1 : HOME PAGE 10

5.2. MODULE 2 :ADD AND RETRIEVE 11

6. RESULTS

6.1. LAUNCH SCREEN 12

6.2. ADD AND VIEW APPLICATION 13

6.3. RETRIEVE USER NAME AND PASSWORD 14

7. CONCLUSION 15

8. REFERENCES 16

9. PLAGIARISM CERTIFICATE 17

CHAPTER 1

INTRODUCTION

1.1 PROBLEM DEFINITION :

In this modern world, we create many accounts on so many websites. If we give the same username and password to all the websites then it will affect privacy and security, we also can not remember all the account usernames and passwords we created. This project gives the solution for this problem where u can save all the account details in one place and the information will be encrypted and secure in one place.

1.2 OBJECTIVES

- We create an account on any website and forget all of them when we try to use it again and then we use to forget the password frequently this project provides the solution for this.
- Easy to save account information in one place.
- As the username and password is encrypted it is safe to save it.

1.3 METHODOLOGY TO BE FOLLOWED

- HTML and CSS for Front-end
- Python-Django for Back-end
- SqlLite for Database.

1.4 EXPECTED OUTCOMES

A password manager can provide peace of mind by securely storing and managing all of your login credentials in one place, so you don't have to worry about memorizing them or storing them in an insecure location. Additionally, many password managers include features such as copy and direct past, which can help users create strong, unique passwords for each account, providing an extra layer of security. With a password manager, you can also easily access your login information from any device, making it more convenient to use the multiple applications.

1.5 HARDWARE AND SOFTWARE REQUIREMENTS

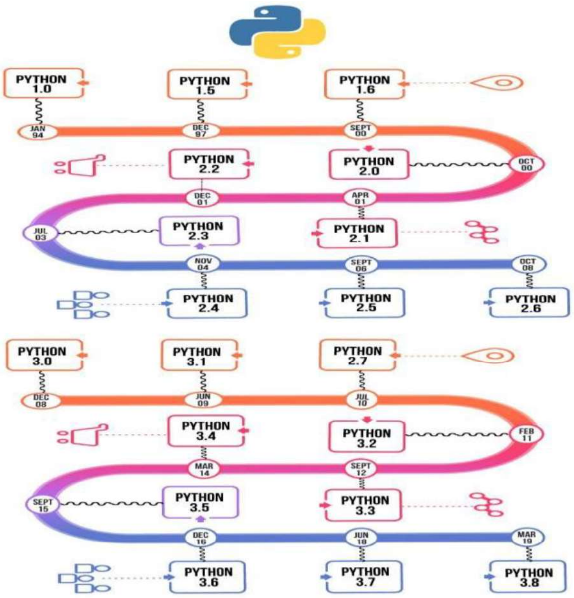
- **Hardware**
  - A personal computer or laptop.
  - RAM minimum of 4GB.
  - Graphics Card.
  - Memory: 512GB SSD.
- **Software**
  - Latest version of python.
  - Atom(for HTML , CSS , JS)
  - Sqlite3(to create database).

CHAPTER 2

PYTHON FUNDAMENTALS :

2.1 FUNDAMENTALS OF PYTHON

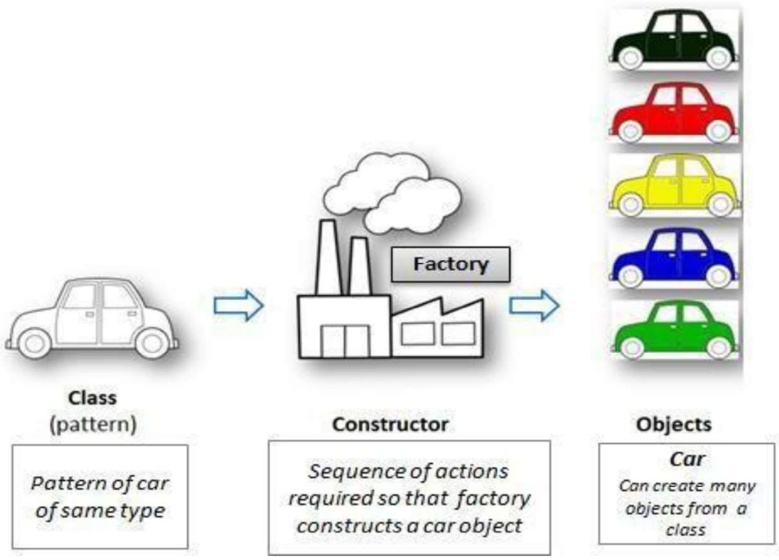
Python is a great language for beginners, as it is relatively easy to learn and use, and the syntax is straightforward and intuitive. Additionally, it is a powerful language, with many possibilities for advanced users., make it very attractive for Rapid App development, as well as for use as a scripting connect existing components together. It is often used for web development, scientific computing, data analysis, artificial intelligence, and more. One of the main benefits of Python is its dynamic nature, which allows for more flexibility and ease of use in comparison to languages that use static typing. Python is open source, which means there are many libraries andresources available for other developers. Additionally, Python has a number of built- in features such as garbage collection, making memory management simpler.



2.2 Classes and Objects

In object-oriented programming, a class is a template for creating objects (also called instances) which share the same attributes and methods. The class defines the structure of the object, while the object is an instance of that class.

An object has two characteristics: state and behavior. An object's attributes or properties, which are the data kept inside the object, serve as a representation of the object's state. The behavior of an object is represented by its methods, which are the actions the object can perform. An object is a specific example of a class, while a class is a template or blueprint used to create objects. Objects contain specific values and methods, while a class defines the structure and behavior that all objects of that type share.


Factory
Class
(pattern)
Pattern of car of same type
Constructor
Sequence of actions required so that factory constructs a car object
Objects
Car
Can create many objects from a class

CHAPTER 3

FUNDAMENTALS OF DBMS

3.1 INTRODUCTION:

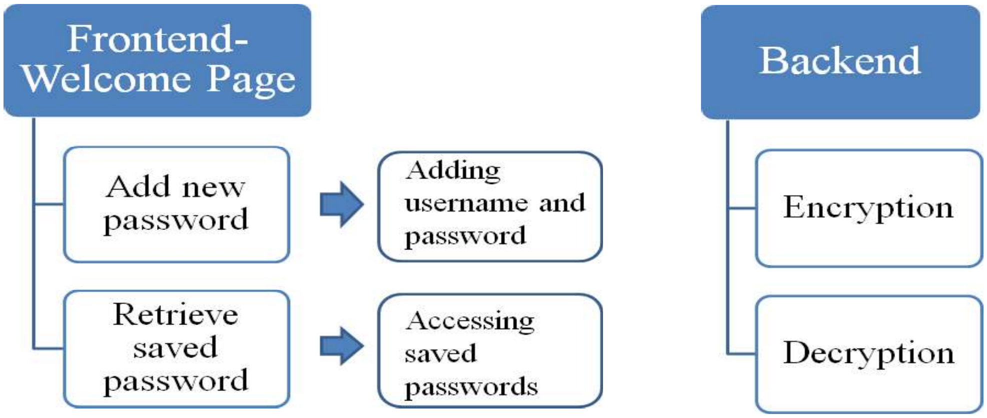
A database management system is a software system used to maintain and interact with the database.. It is typically used to organize, store, and retrieve large amounts of data in an efficient and organized manner. Databases are widely used in various applications such as financial systems, inventory management, customer relationship management, e-commerce websites, and more. There are different types of databases such as relational databases, NoSQL databases, document databases, graph databases, and more. Relational databases, such as MySQL and Oracle, organize data into tables with rows and columns and uses a structured query language (SQL) to access and manipulate the data. NoSQL databases, such as MongoDB and Cassandra, do not use the traditional table structure, but instead store data in flexible, unstructured formats like JSON or BSON. A DBMS is a software that interacts with the database, allowing users to create, read, update and delete data in the database.

CHAPTER 4

DESIGN

4.1 DESIGN GOAL

My project's design mainly includes these following operations:

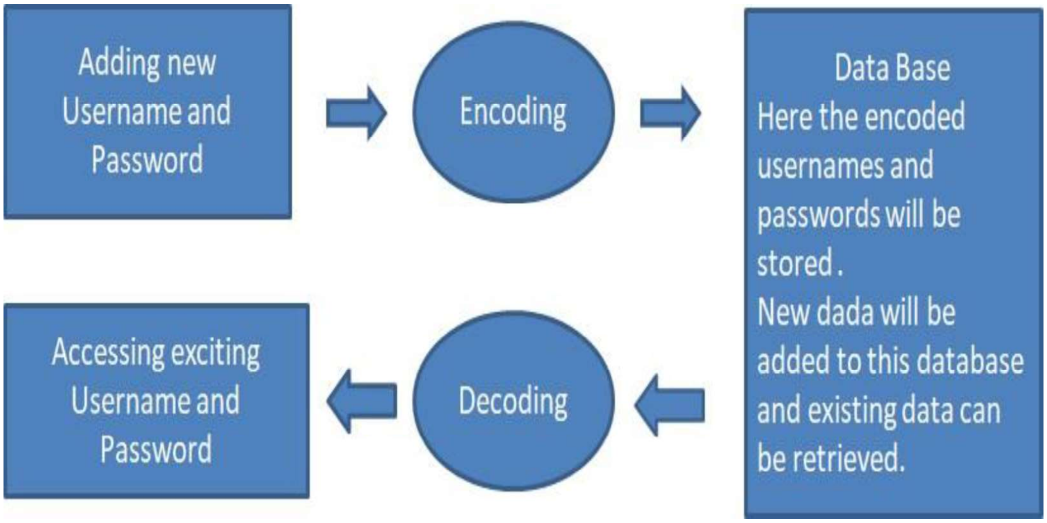


HTML, CSS, and JavaScript are commonly used to create the front-end user interface of web applications. HTML provides the structure of the page, CSS is used to define the layout and styling, and JavaScript provides interactivity and dynamic functionality.

SQLite3 is a lightweight, file-based relational database management system that can be used to store and manage the data for the project. It is a popular choice for small to medium-sized projects because it does not require a separate server, and is easy to set up and use.

Python is a powerful programming language that can be used for both front-end and back-end development. The Django framework is a popular web framework for building web applications using Python. It provides a high-level, reusable set of components for building web applications such as user authentication, database models, and an admin interface.

4.2 FLOW CHART


Adding new Username and Password
Encoding
Data Base
Here the encoded usernames and passwords will be stored .
New dada will be added to this database and existing data can be retrieved.
Decoding
Accessing exciting Username and Password

When the user provide account username and password it will be encoded and saved in the database this is provided for the privacy of the user as the admin also can niot see the info of the user.

When the user wants to see the details of his/her project then the encoded info is taken and then its decoded and shown for the user

Its works on the flow of the provided flowchart.

4.3 ALGORITHM FOR ADDING NEW APPLICATION

STEP 1: Start (Run server from command Prompt)

STEP 2: WELCOME PAGE opens.

STEP 3: Select “ADD NEW APPLICATION” on the welcome page. This will redirect you to the next page

STEP 4: On this page you can see a simple form of 3 fields: “Name of the application, User name and Password” fill in the details and press “SAVE”.

STEP 5: Once the “SAVE” button is pressed these details are passed to the backend where the data is encrypted and saved.

STEP 6: In the backend, first an unique key is generated for the application, with the help of this key username and password is encrypted using the “Fernet” module. Now the palindrome of the key is found.

STEP 7: Application name is stored in one table of the database and the palindrome of the key along with the encrypted username and password and a link to the application name is stored in another table of the database.

STEP 8: Repeat steps 4,5,6,7 are repeated whenever the user wants to add a new application.

STEP 9: Stop

4.4ALGORITHM FOR RETRIEVING USERNAME AND PASSWORD OF EXISTING APPLICATION

STEP 1: Start (Run server from command Prompt).

STEP 2: WELCOME PAGE opens.

STEP 3: Select “RETRIVE EXISTING PASSWORDS” on the welcome page. This will redirect you to the next page.

STEP 4: On this page you can see name of all the applications present in the database in block format. These blocks will be highlighted once the cursor hovers over it and can be clicked.

STEP 5: Click on the application you want to retrieve the username and password of this will redirect you to a new page.

STEP 6: Once the application is selected in the backend the palindrome of the key, encrypted username and password is retrieved from the database.

STEP 7: Original key is found by calculating the palindrome of the palindrome of the key with the help of this key the username and password is decrypted using the “Fernet” module.

STEP 8: Finally the Name of the selected application along with the decrypted username and password is displayed to the user.

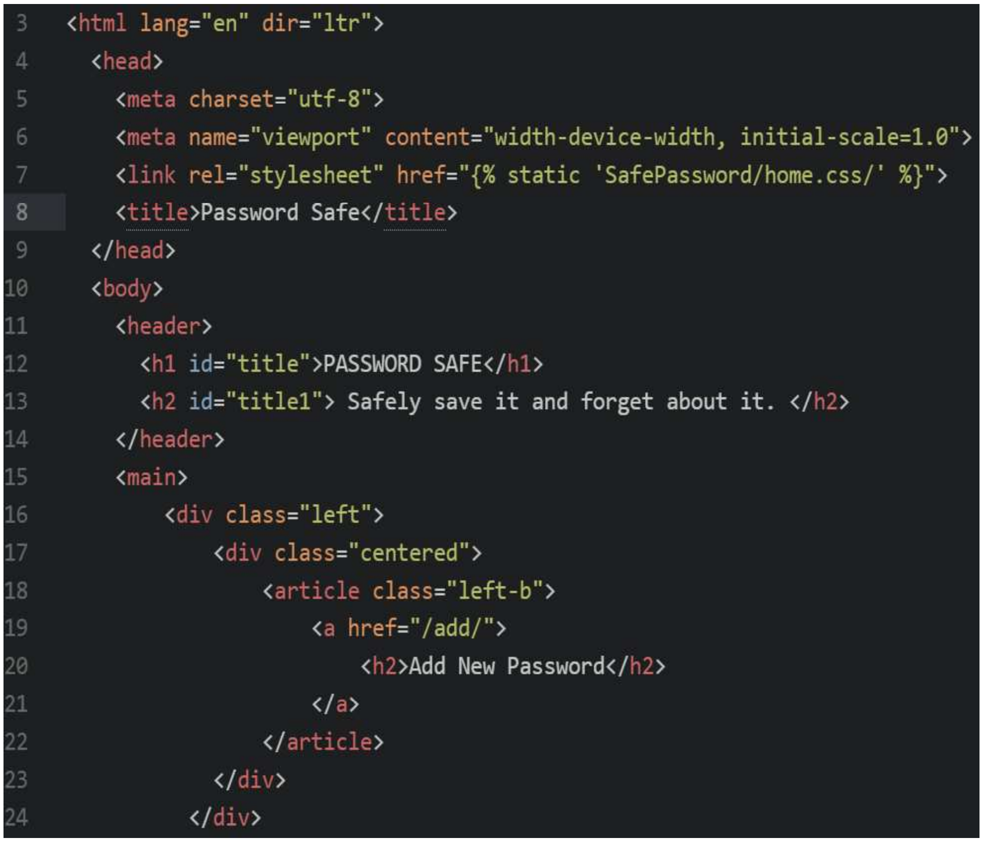
STEP 9: Repeat steps 5,6,7,8 are repeated whenever the user wants to retrieve the username and password of any existing application.

STEP 10: Stop

CHAPTER 5

IMPLEMENTATION

5.1 MODULE 1:HOME PAGE



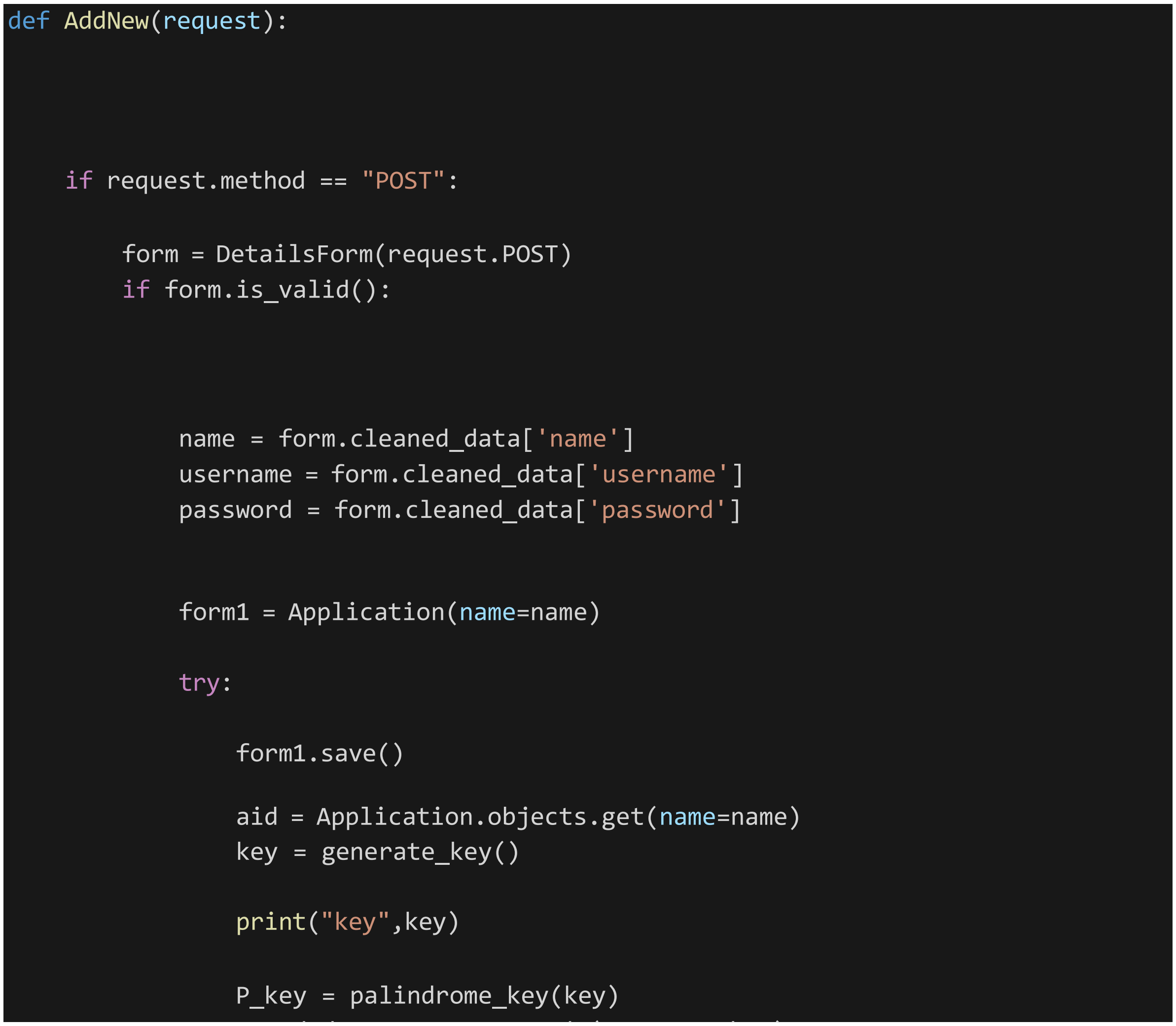
```
<html lang="en" dir="ltr">
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width-device-width, initial-scale=1.0">
    <link rel="stylesheet" href="{% static 'SafePassword/home.css/' %}">
    <title>Password Safe</title>
  </head>
  <body>
    <header>
      <h1 id="title">PASSWORD SAFE</h1>
      <h2 id="title1"> Safely save it and forget about it. </h2>
    </header>
    <main>
        <div class="left">
            <div class="centered">
                <article class="left-b">
                    <a href="/add/">
                        <h2>Add New Password</h2>
                    </a>
                </article>
            </div>
          </div>
```

This is the first module of the code which is helpful to create GUI for the frontend using HTML (tags,links, etc) and CSS(for styling).

Below Modules are GUI interface same as the above but they are used to add and retrieve the passwords .

5.2 MODULE 2:ADD AND RETRIEVE

Figure 5.2.1: ADD page using HTML CSS(Styling)

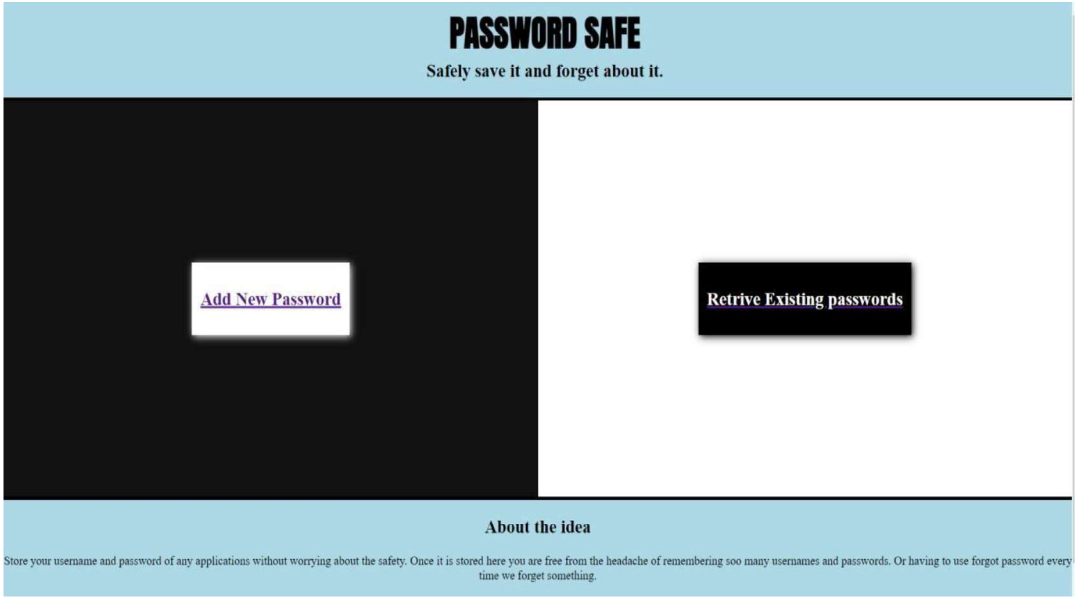


```
def AddNew(request):



    if request.method == "POST":

        form = DetailsForm(request.POST)
        if form.is_valid():



            name = form.cleaned_data['name']
            username = form.cleaned_data['username']
            password = form.cleaned_data['password']


            form1 = Application(name=name)

            try:

                form1.save()

                aid = Application.objects.get(name=name)
                key = generate_key()

                print("key",key)

                P_key = palindrome_key(key)
```

CHAPTER 6

RESULTS

6.1 LAUNCH SCREEN


PASSWORD SAFE
Safely save it and forget about it.
Add New Password
Retrive Existing passwords

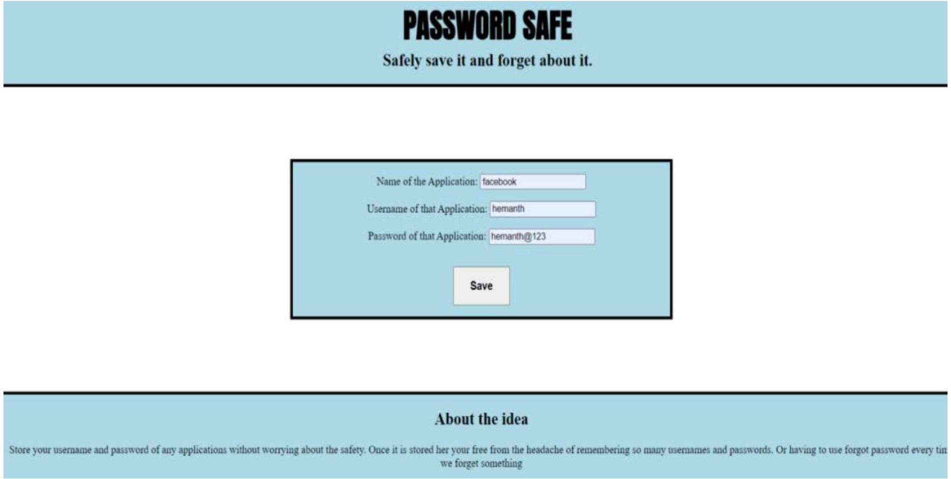
About the idea

Store your username and password of any applications without worrying about the safety. Once it is stored here you are free from the headache of remembering soo many usernames and passwords. Or having to use forgot password every time we forget something.

Figure 6.1: welcome page using HTML CSS(Styling)

This is the first page , when we do launching of the server welcome page will be loaded with the above interface showing the 2 main options to use , first one is to add new new password and the second one is to retrieve the password.

6.2 ADD NEW PAGE VIEW APPLICATION



PASSWORD SAFE

Safely save it and forget about it.

Name of the Application: facebook

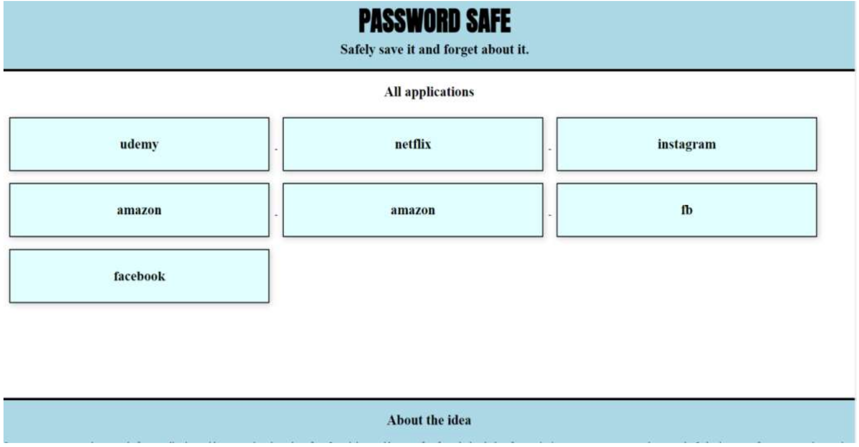
Username of that Application: hemanth

Password of that Application: hemanth@123

About the idea

Store your username and password of any applications without worrying about the safety. Once it is stored her your free from the headache of remembering so many usernames and passwords. Or having to use forgot password every tin
we forget something

Figure 6.2.1: ADD new page using HTML CSS(Styling)



PASSWORD SAFE

Safely save it and forget about it.

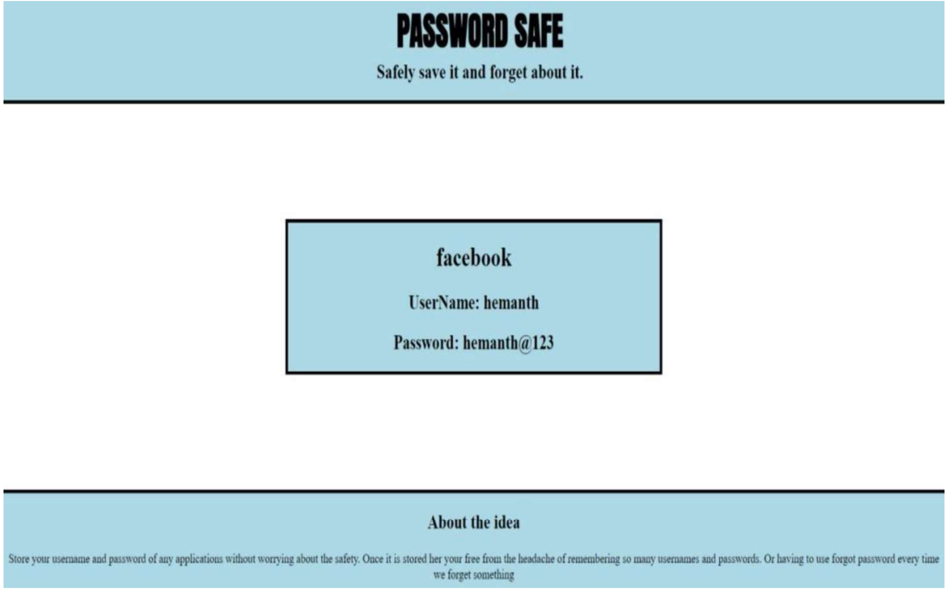
All applications

About the idea

Figure 6.2.2: view application page using HTML CSS(Styling)

This page will be loaded when we use the option add new password,in this there are three main fields to store the data first is name ,second is login details and password details.when we click the save button it will be saved in the encoded form in the database .In main page while accessing we have to select the particular application name.

6.3 RETRIEVE USER NAME AND PASSWORD



PASSWORD SAFE

Safely save it and forget about it.

facebook

UserName: hemanth

Password: hemanth@123

About the idea

Store your username and password of any applications without worrying about the safety. Once it is stored her your free from the headache of remembering so many usernames and passwords. Or having to use forgot password every time we forget something

Figure 6.3: retrieve page using HTML CSS(Styling)

This page will be loaded when we select the particular application name those are available in the database , login details will be displayed in this interface page.

CHAPTER 7

CONCLUSION

This Project can provide peace of mind by securely storing and managing all of your login credentials in one place, so you don't have to worry about memorizing them or storing them in an insecure location. Additionally, this password managers include features such as encoding and decoding, which can help users to protect strong, passwords for each account, providing an extra layer of security. With a password manager, you can also easily access your login information from any device, making it more convenient to use the multiple applications.

REFERENCES

Head-First Python by Paul Barry (O'Reilly, 2016)

Invent Your Own Computer Games with Python by Al Sweigart Effective Python by Brett Slatkin

Murach's HTML5 and CSS3 by Anne Boehm and Zak Ruvalcaba

Django 3 Web Development Cookbook by Aidas Bendoraitis and Jake Kronika High Performance

Django by Peter Baumgartner and Yann Malet

6620786

ORIGINALITY REPORT

1 %	%	1 %	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Mauricio Osorio, Claudia Zepeda, Hilda Castillo, Patricia Cervantes, Jose Luis Carballido. "My University e-Partner", 2019 International Conference on Inclusive Technologies and Education (CONTIE), 2019 Publication	1 %

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		