Password Manager Website - Detailed Project Plan

1. Introduction

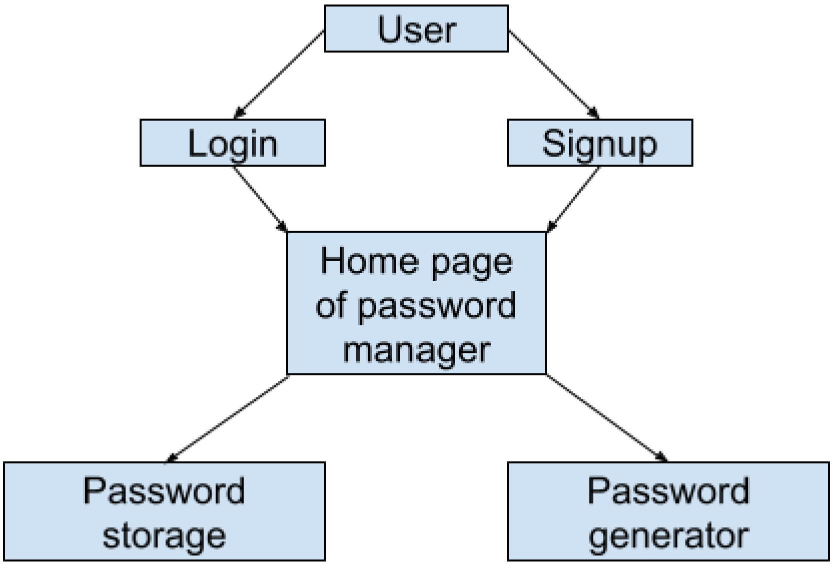
My website will allow users to save multiple passwords, manage them by adding their criteria, generating new random passwords and storing the saved passwords forever. This website will be easy to use with multiple features and the features will help the user to find the passwords easily. This website will also ensure the users privacy, safety and security.

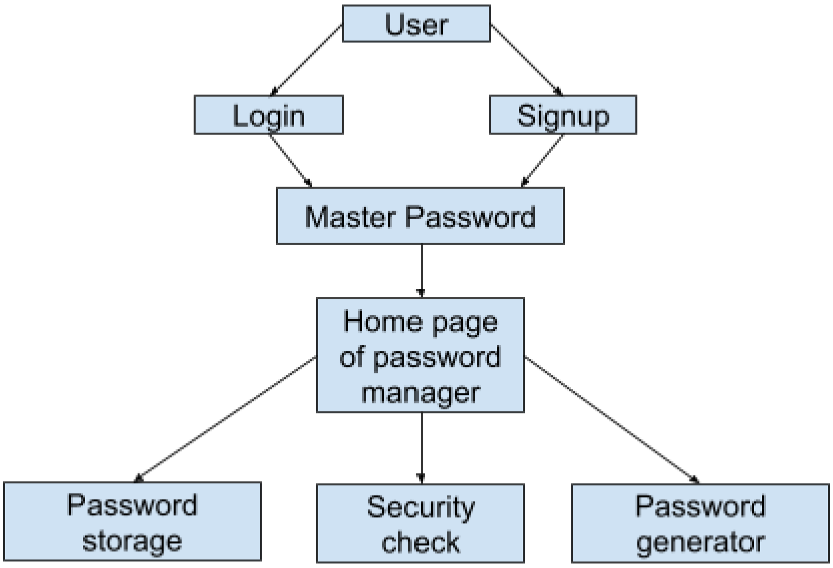
- **Targeted Audience:** This Website is for the people struggling to manage their passwords and keep forgetting their passwords and it is also for the businesses to handle multiple passwords. This website will help them and they don't need to worry about saving their passwords anymore.
- **Core Functionality:**
    - Store passwords securely and insuring users privacy
    - Generate random strong passwords
    - User can view, edit, delete, and copy passwords in the dashboard
    - Provide additional security features like auto logout and two step verifications.

---

2. ER Diagrams

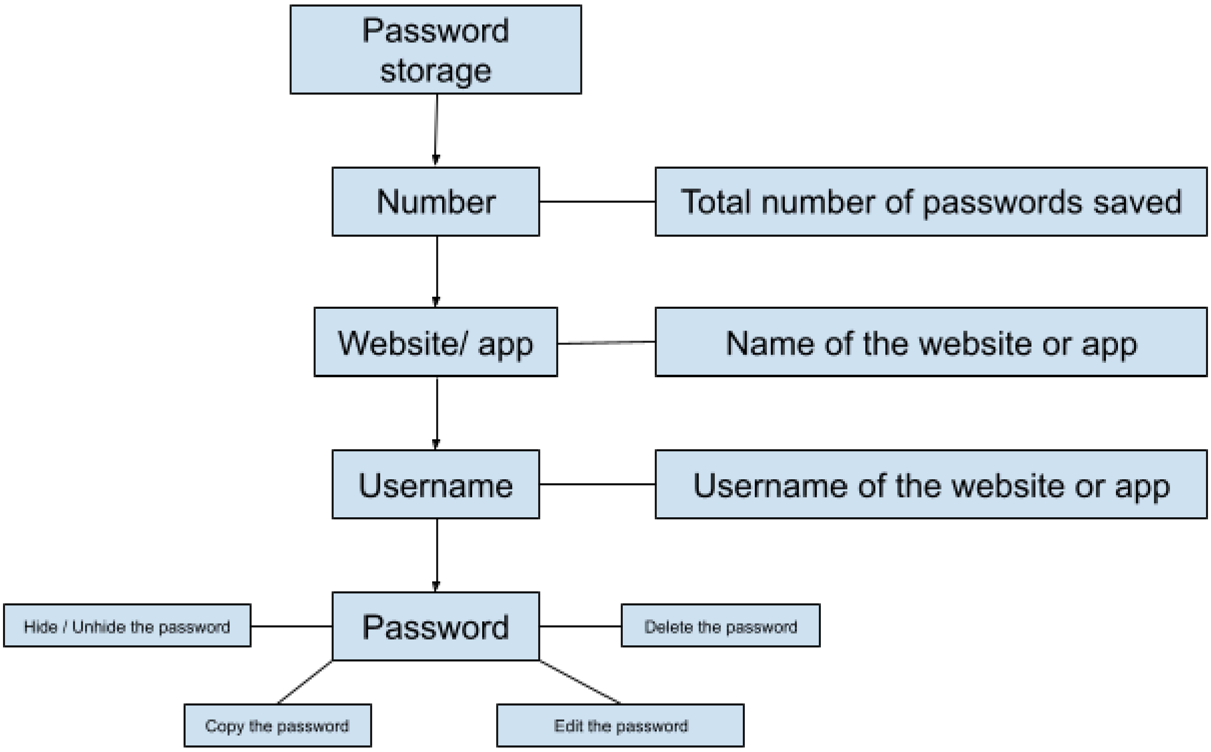
- **Users interaction**



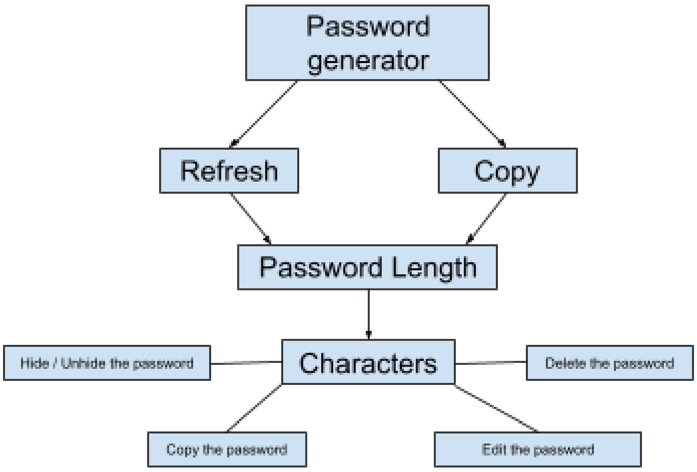


I added a master password page after getting to the vault which I planned after I made the website for extra safety. I also added a few more features like security features where the user can see how strong their password is. There are multiple other small features which are not very important but they are there.

- **Password storage page**


Password storage page
Number
Total number of passwords saved
Website/ app
Name of the website or app
Username
Username of the website or app
Hide / Unhide the password
Password
Delete the password
Copy the password
Edit the password

- **Password Generator page**



---

3. Website Layout & Design

Homepage (Login/Signup Page)

The homepage will allow users to either log in or sign up and there will be a different page for sign up which you need to click to get access by clicking the signup button. The layout includes:

- **Login Form:**
  - Username
  - Password
  - Forgot Password
  - Login Button and Signup button next to it
  - 2 step verification if enabled by the user

- **Signup Form:**
  - Username
  - Email
  - Password (with confirmation)

- **Colours and Fonts:**
  - Light colours:
    - Purple
    - Light purple
    - Silver

        - Gray
        - white
    - Fonts:
        - Arial
        - Tapioka

Dashboard Page (Post-login)

After logging in, users will be redirected to the **Dashboard** where they can manage their passwords. It will display passwords in a table format which will include the following :

This is the basic idea of how the table is going to look like

#	Website/App	Username	Password	Edit	Delete
1	Gmail	abc@gmail.com	XsaAhd8 👁‍🗨📋	✍️	🗑️
2	Netflix	abc	********** 👁📋	✍️	🗑️

There will also be buttons to add a new password and options for logging out or switching users.

Password Generator Page

A Password Generator feature will allow users to create secure passwords. It will let users customise:

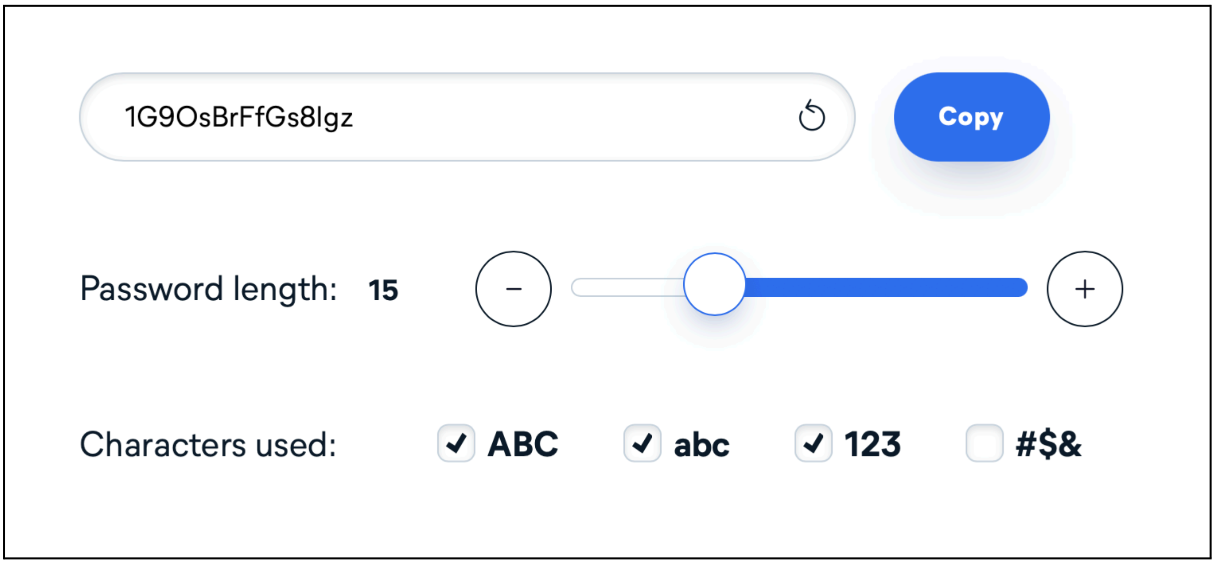
- Length of Password
- Include/exclude special characters
- Generate and copy options

Password Details Page

When viewing a password, users can see:

- Website Name
- Username
- Password (hidden by default)

- Buttons to Copy, Edit, or Delete

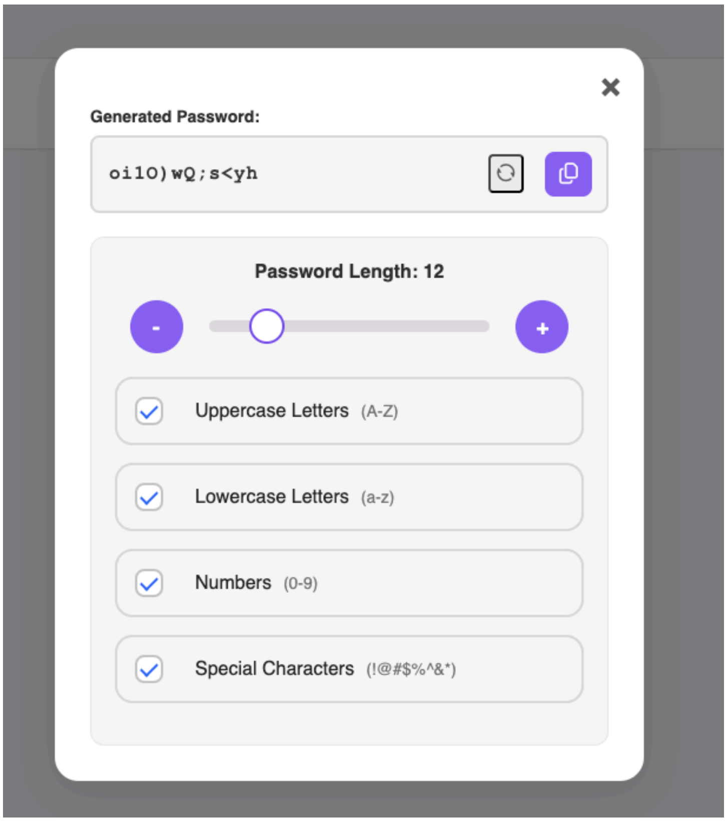


1G9OsBrFfGs8lgz

Password length: 15

Characters used: ☑ ABC ☑ abc ☑ 123 □ #$&

The password generating page will have features like the one given above but the design will be different.



Generated Password:

oi1O)wQ;s<yh

Password Length: 12

☑ Uppercase Letters (A-Z)

☑ Lowercase Letters (a-z)

☑ Numbers (0-9)

☑ Special Characters (!@#$%^&*)

This is the password generator page which I made which has exactly the same features as given above and it also changes live using some javascript.

---

4. Justifications for Design

Security Best Practices

1. **Password Encryption:** All passwords will be encrypted using Fernet **encryption**, a symmetric encryption method that is both secure and simple to implement.
2. **Master Password Authentication:** This ensures that even if someone gains access to the database, they cannot view the passwords without the master password.
3. **Auto Logout:** To prevent unauthorized access after inactivity, a timeout-based auto logout will be implemented.

User Experience (UX) Considerations

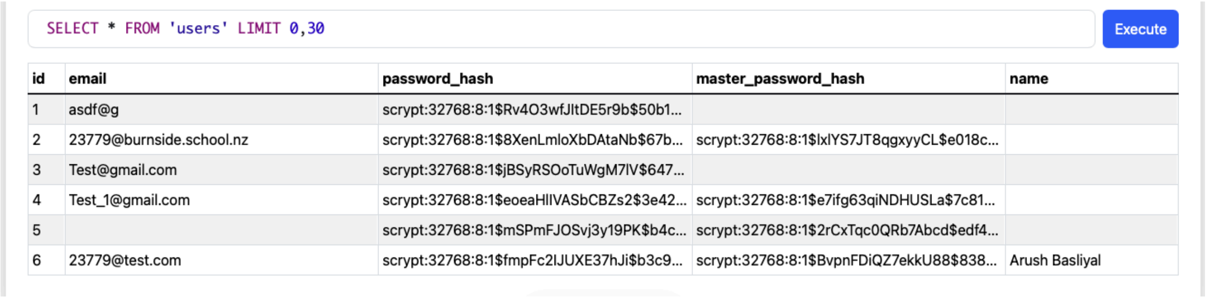
1. **Password Generator:** Allows users to create strong passwords, reducing the risk of weak passwords.
2. **Mobile Responsiveness:** The design will be responsive, ensuring it works well on both desktops and mobile devices.

---

5. databases

I added 3 databases in SQLite for different reasons.

1. For the users who are making their ID in the signup page.

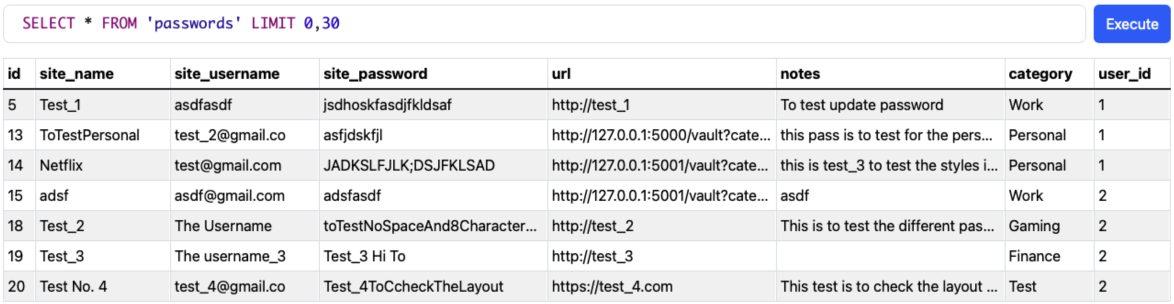


```
SELECT * FROM 'users' LIMIT 0,30
```

id	email	password_hash	master_password_hash	name
1	asdf@g	scrypt:32768:8:1$Rv4O3wfJltDE5r9b$50b1...		
2	23779@burnside.school.nz	scrypt:32768:8:1$8XenLmloXbDAtaNb$67b...	scrypt:32768:8:1$lxlYS7JT8qgxyyCL$e018c...	
3	Test@gmail.com	scrypt:32768:8:1$jBSyRSOoTuWgM7lV$647...		
4	Test_1@gmail.com	scrypt:32768:8:1$eoeaHlIVASbCBZs2$3e42...	scrypt:32768:8:1$e7ifg63qiNDHUSLa$7c81...	
5		scrypt:32768:8:1$mSPmFJOSvj3y19PK$b4c...	scrypt:32768:8:1$2rCxTqc0QRb7Abcd$edf4...	
6	23779@test.com	scrypt:32768:8:1$fmpFc2lJUXE37hJi$b3c9...	scrypt:32768:8:1$BvpnFDiQZ7ekkU88$838...	Arush Basliyal

This had basic id starting from 1 password for logging and a master password for double security. I added multiple users although emails can be anything but if someone made an id with 1 email that email cant be used to make a different account and finally added a name section which has been added in the last as decided to add it later for showing the name in the profile popup.

2. For all the passwords saved in the website with all the users.

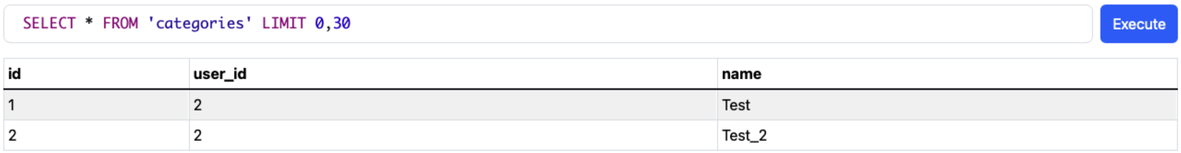


```
SELECT * FROM 'passwords' LIMIT 0,30
```

id	site_name	site_username	site_password	url	notes	category	user_id
5	Test_1	asdfasdf	jsdhoskfasdjfkldsaf	http://test_1	To test update password	Work	1
13	ToTestPersonal	test_2@gmail.co	asfjdskfjl	http://127.0.0.1:5000/vault?cate...	this pass is to test for the pers...	Personal	1
14	Netflix	test@gmail.com	JADKSLFJLK;DSJFKLSAD	http://127.0.0.1:5001/vault?cate...	this is test_3 to test the styles i...	Personal	1
15	adsf	asdf@gmail.com	adsfasdf	http://127.0.0.1:5001/vault?cate...	asdf	Work	2
18	Test_2	The Username	toTestNoSpaceAnd8Character...	http://test_2	This is to test the different pas...	Gaming	2
19	Test_3	The username_3	Test_3 Hi To	http://test_3		Finance	2
20	Test No. 4	test_4@gmail.co	Test_4ToCcheckTheLayout	https://test_4.com	This test is to check the layout ...	Test	2

This is for all the passwords saved by every user in the website with all different accounts. I added an id for each password and then the site name then the username of the site in optionals I choose url and notes and also added a category so people can also save their category and all the passwords will be found using the user_id which is the ids the users have and can be used to make the passwords easier to find.

3. For the categories which users can make if they want to store other passwords.



```
SELECT * FROM 'categories' LIMIT 0,30
```

id	user_id	name
1	2	Test
2	2	Test_2

This also has an id to find each category and it also has a user id so it will be easy to find the category with the given user id to the user and their new category can be found easily and lastly the name of the category. This table was mainly added so the category doesn't get lost and can be saved on the website. The user can also delete the category if they don't want to have it or created it accidentally.

---

6. Problems

1. **Password generator** - The password generator popup was the most difficult task to do mainly because of the slider which must be properly working with the password length and the number showing the length. I was not able to match the slider and the adding and minusing buttons with the length of the password and if I made it happen it was not live and had to set it then changing the length which

was not good for user experience.
How I fixed it - I finally had to use javascript to make the slider live working with the length of the password and also made other buttons live so the page didn't have to refresh again and again while selecting or deselecting the options.

2. **Space in password -** While I was putting the password I realized that anyone can put their password as spaces only which are not good for both security and the quality of the website.
   How I fixed it - I simply added a backend which was validation function `_valid_password()` and also added `no_spaces=True` to reject all the passwords with the spaces and saying no spaces are allowed.

3. **Max/min Length and required -** There was an issue where anyone was able to change the required and minimum and maximum length by just using the inspect option which was also not good for the safety and the quality of my website.
   How I fixed it - I changed the required, max and min from html to the backend so no one is able to change it from inspect making the site more safer.

4. **Password in database -** Any one who had access to the website's database was able to access the passwords of all the users as it was directly saved to the database.
   How I fixed it - I used `werkzeug.security` to change the password from normal password to hashed password and stored it inside the database so the password was different but it stored a hash password in the database instead of the real one.

7. Testings

I tested the website by simply following the path of the website like going to the sign up page then login then master password and then to the vault. I did it multiple times and tried to do something new everytime I entered the website. firstly tried it normally to see if it works without any issues. Then I tried to be tricked like typing wrong passwords and wrong emails and trying to create a new id with the same gmail. I then tried to play with the insects and tried to remove the required or put a different code in the console to kill mine and the website was well secure to protect from small issues.

- **Authentication & Session Tests**

What did I test	**How I tested it**	**What was the expected result**	**What actually happened**
Correct Login	I tried to login with the correct login and password to the website.	I will be able to enter the website without any issues.	I entered the website without any issues.
Incorrect Login	I tried to login with an incorrect login and password to the website.	The program will not let me enter the vault and say incorrect password or username.	I was not able to enter the vault and it actually gave me an alert for the wrong username or password.
Empty login fields	I tried login with empty login fields and also tried typing multiple spaces.	The program should alert for not writing the field and saying no spaces are allowed.	The program stopped me in the login page telling me to put the fields and not write spaces.

Session timeout	I stayed on the website for longer than 20 min and did other tests while I was idle.	It should log me out automatically when logged in for more than 20 minutes.	It logged me out automatically after 20 minutes.
Direct access to vault	I tried going directly to the vault using the search bar and writing /vault directly.	It should not let me enter the vault instead redirect me back to the sign/login page.	It redirected me back the the page I tried entering to the vault.
Multiple logins	I tried logging with different browsers and adding things to the database.	It should allow adding things to the database and update the added things in every browser.	It updated the things added from different accounts and just had to refresh the page.

- ## Signup & Password Tests

What did I test	How I tested it	What was the expected result	What actually happened
Password length less than 8	I tried putting the password less then 8 digits and tried changing the min value in inspect mode.	It will not let me set the password less then 8 characters even after changing the value in	It didn't let me change the value and write less than 8 characters.

		inspect mode.	
Password hashing	I tried to sign up properly with correct details and wrote the password correctly.	It should hash the password and change it in the database to the random characters.	It hashed the password to random characters and the password I wrote was not in the database.

---

8. Conclusion

In conclusion I think the website was not exactly what I expected but it was mostly what I had expected. Everything works nicely without any crashes and mostly every security is inside the backend so normal users can't have access to it and can't change the sensitive options like required options and max / min. There are other features like password generator which works very well and it works live so the page will not refresh every time the option is being selected and the slider also works live with the length. The options for the password generator are also working and the user has to select at least 1 option out of 4 in order to generate the password so the program will not let the user to deselect all the options. There is also another feature which is more for fun which is security check. It allows the user to see the strength of their password and lets the user see the estimated time of hacking the password the user has selected. I should have added more features in the website but I decided to stick with the main features and prioritise those features first. That's why I came up with main features and 1 fun and useful feature.

---

9. Feedback

Who	**What did they say**	**Should or should not**
Non programmer (Brother)	This website is good, it looks amazing and all the features work very well. But the category option dont have delete option if someone want to delete it and want to be more clear so you should add a button to delete the category as well	I definitely added a button to delete the category as I forgot about that and it's a very important feature and must have in this website as if the user is not able to delete their added category the website will be soo bad.
Non Programmer (Mother)	This website is very simple and easy to understand and works well. I really like the password strength check and the password generator. I think almost everything is good but you can add a forgot password button under the password if someone forgot their password.	It's a very important feature and very useful as well as if someone forgot their password they would not be able to have access to the page and will be stuck but currently my website takes any gmail if it exists or not and I don't know how to make it work with real emails and the codes that appear in gmail so I would not be able to include this feature in the website.
Programmer (Friend)	Good looking websites with very good styles but the security is not good and anyone can break the website by just going to inspect or just misleading the passwords or the	As a password manager website this website should be very good at security and that's why I already fixed all the security issues and there will be no or minimal security

	usernames so should work on the security to make things safer.	breaches so my website is mostly safe now.
Non programmer (Friend)	The ui is good but you should use the strips to store the passwords basically in table form instead of blocks. They are much cleaner and easy to understand whereas currently your website doesn't look very clean and doesn't even tell all the information properly.	That's a good idea as I thought about it before and also tried it but its soo difficult to keep it look good according to the screen size because when the screen size is big the right side will be blank which dont look good where as when using in card form we can add more cards in 1st row or make the size big it work in both way so I am not be able to taking this for my website.

---

10. Repository and link

This is the repository of my website in github.

https://github.com/Arush2008/Password-manager.git

This is the public live website which everyone has access to.

https://password-manager-1-i14u.onrender.com/login