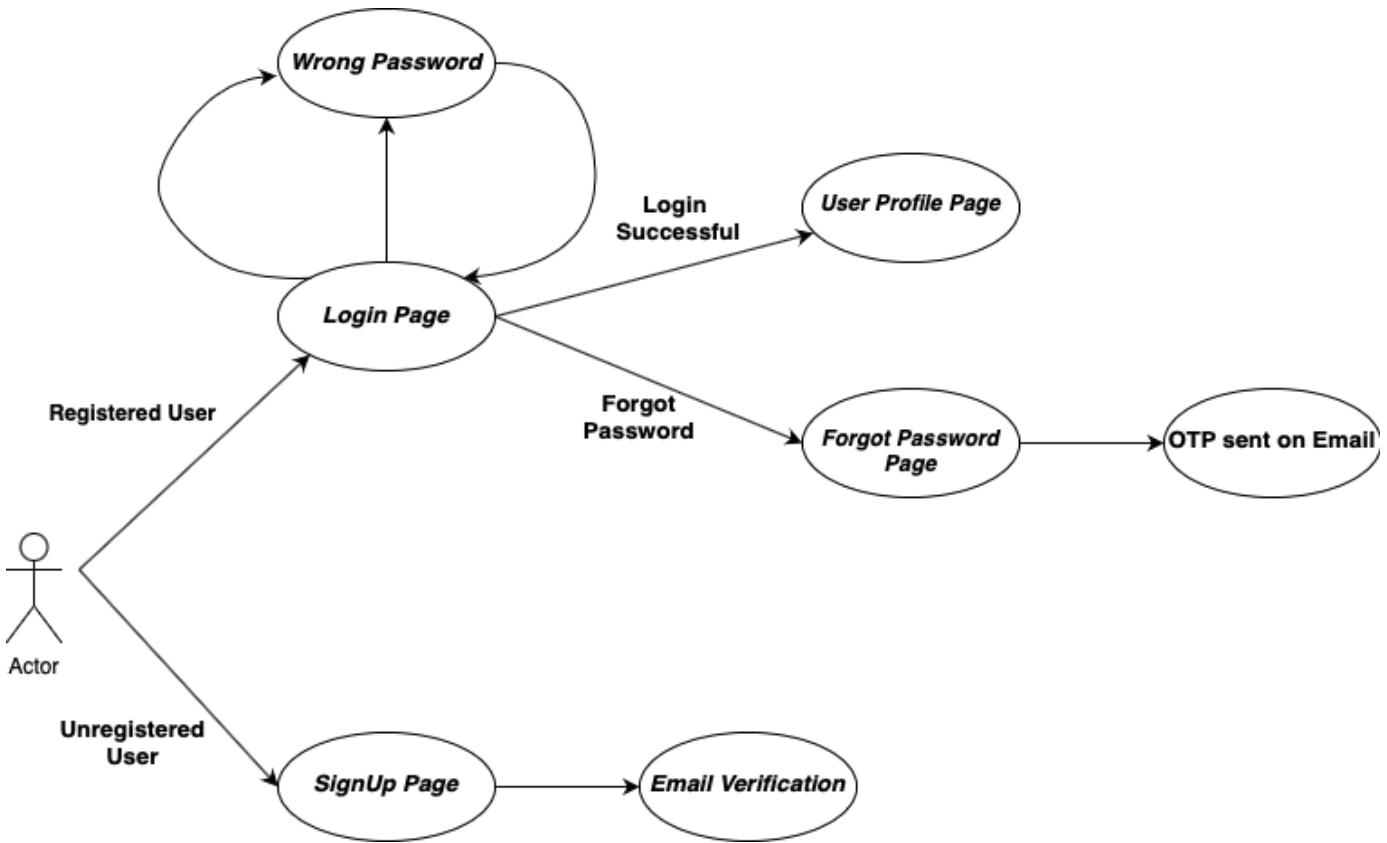


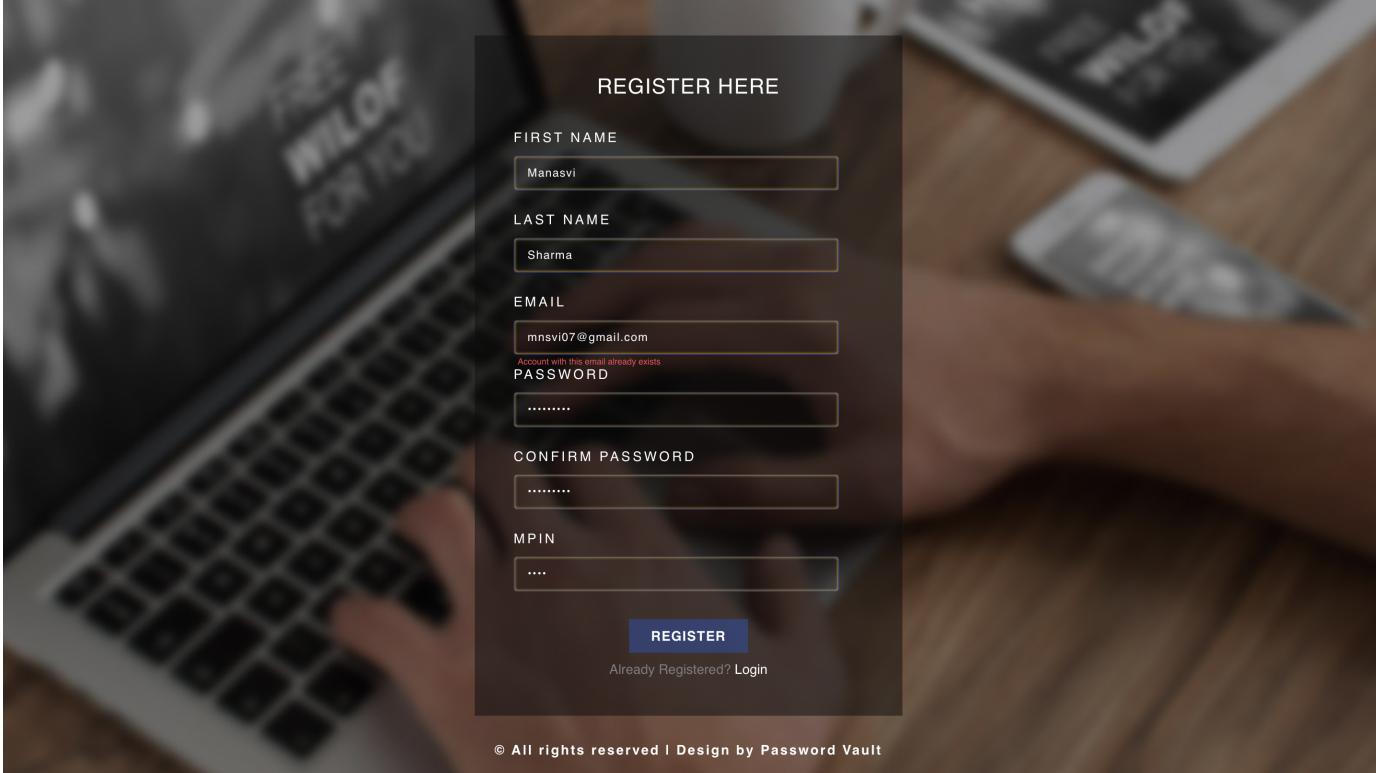
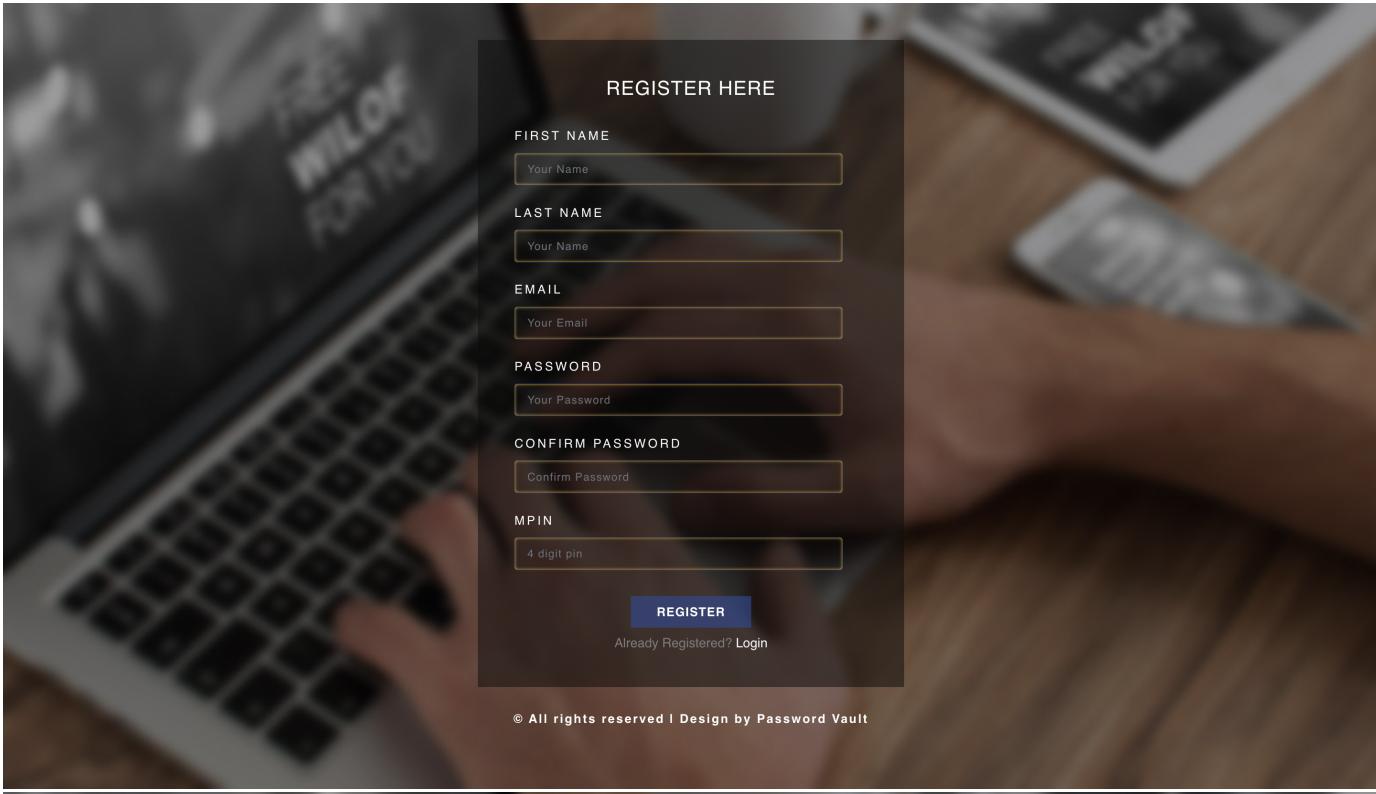
# User Scenarios

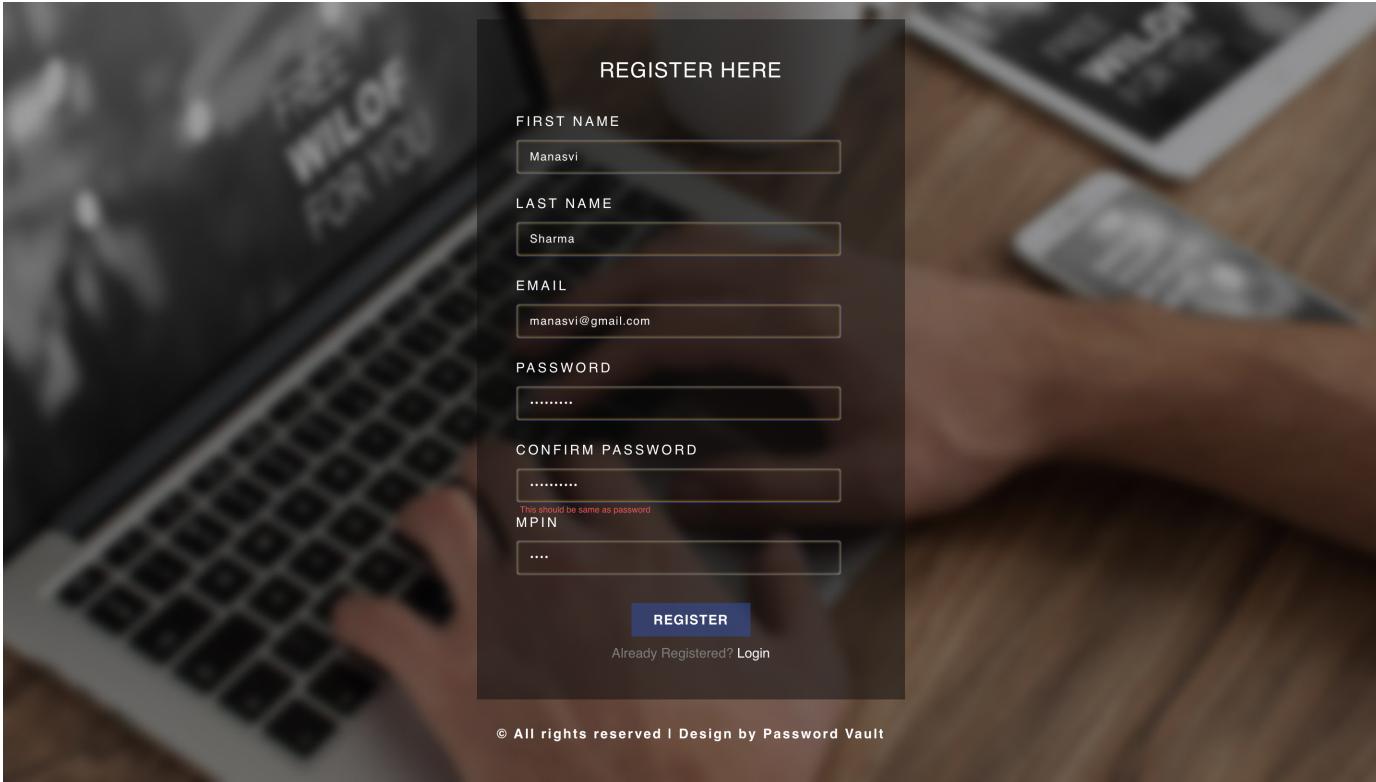
## 1. User Management



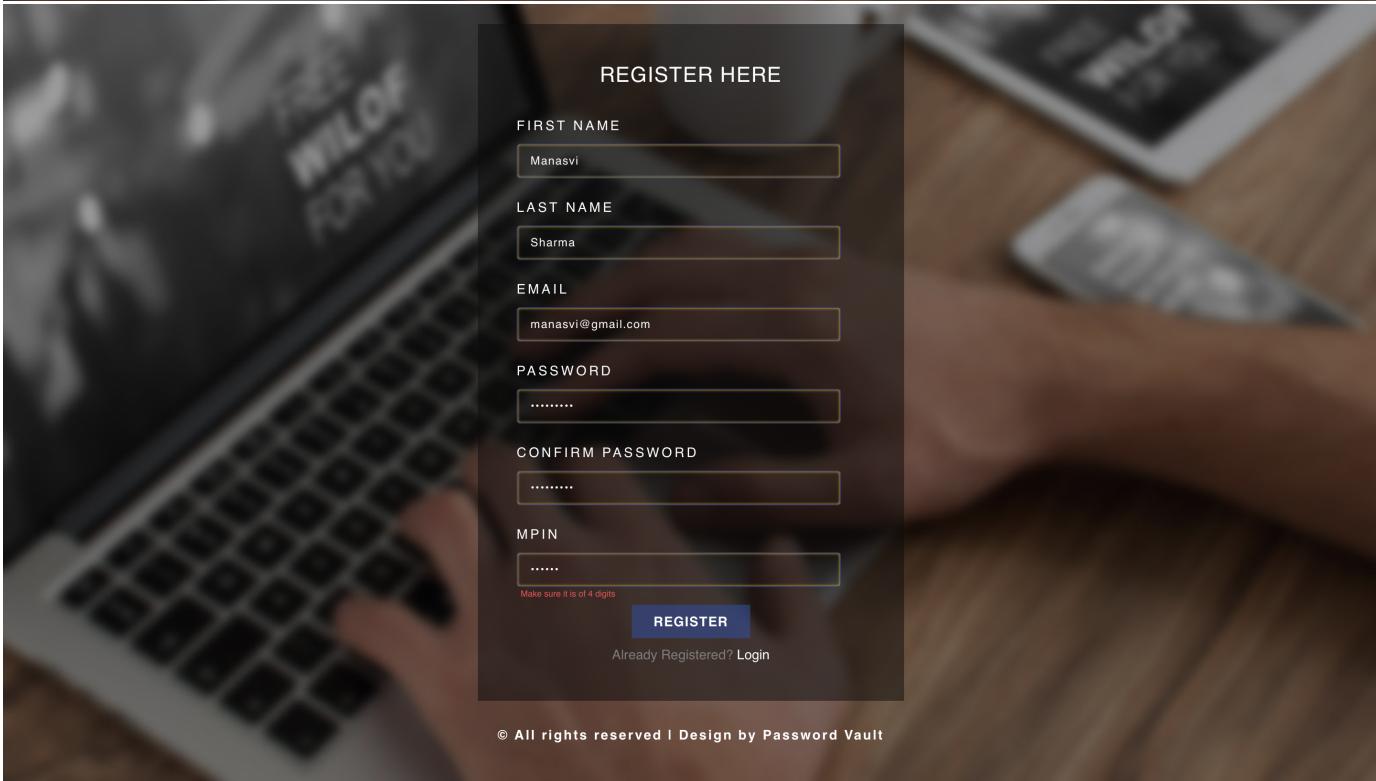
## a. User Registration

Use Case Number	1
Use Case Name	User Registration
Input	First Name, Last Name, Email, M-PIN, Password, Confirm Password
Output	User should be successfully Registered
Description	Before user can experience all the features of Password Vault, user should create their account. To register, user should click on the SIGN-UP button on the Home page which will redirect them to SignUp page and ask them to fill the required details. After successful registration, one email link will be sent to the respective email id for email verification. After successful verification, user is good to Login in the application.
Task on Gitlab	<a href="https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/20">https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/20</a>

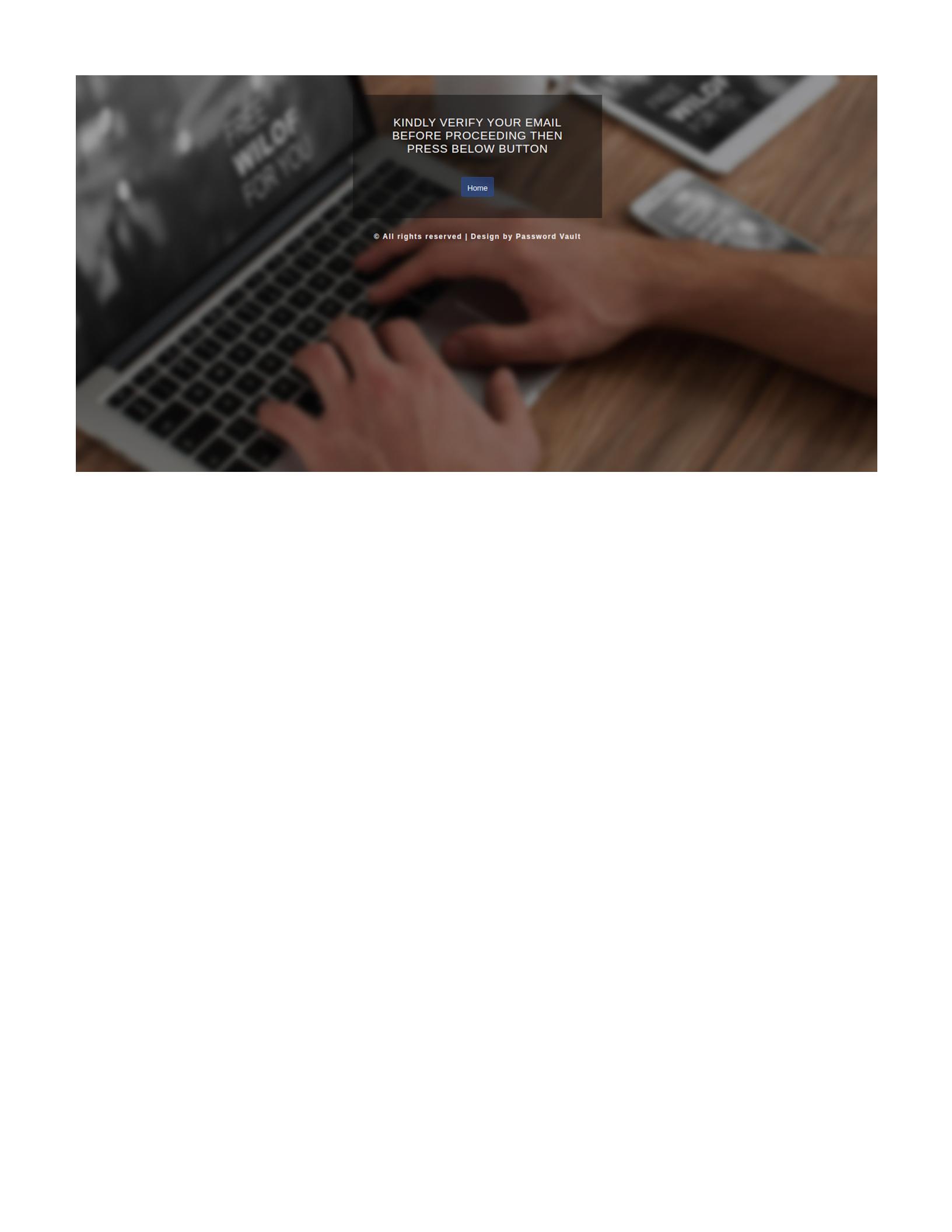




© All rights reserved | Design by Password Vault



© All rights reserved | Design by Password Vault



KINDLY VERIFY YOUR EMAIL  
BEFORE PROCEEDING THEN  
PRESS BELOW BUTTON

[Home](#)

© All rights reserved | Design by Password Vault

S

signupdonotreply@gmail.com

Welcome to password vault

To: mnsvi07@gmail.com

Inbox - Google Yesterday at 6:55 PM

## Welcome Email!



### Welcome Manasvi!

Thank you for signing up with Password vault.

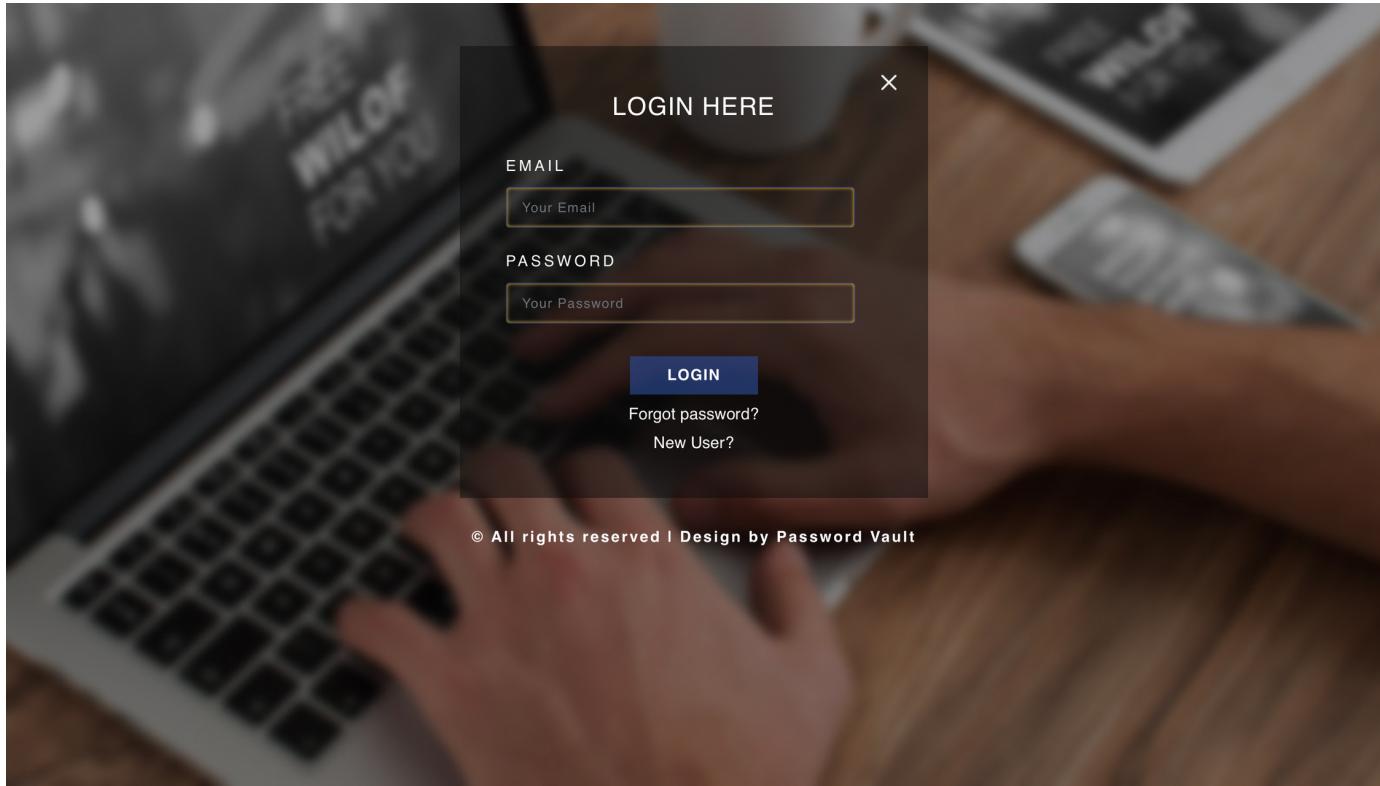
You're receiving this email because you have created user account at Password vault.

[Click to Validate](#)

© Copyright 2022 PasswordVault. All Rights Reserved.

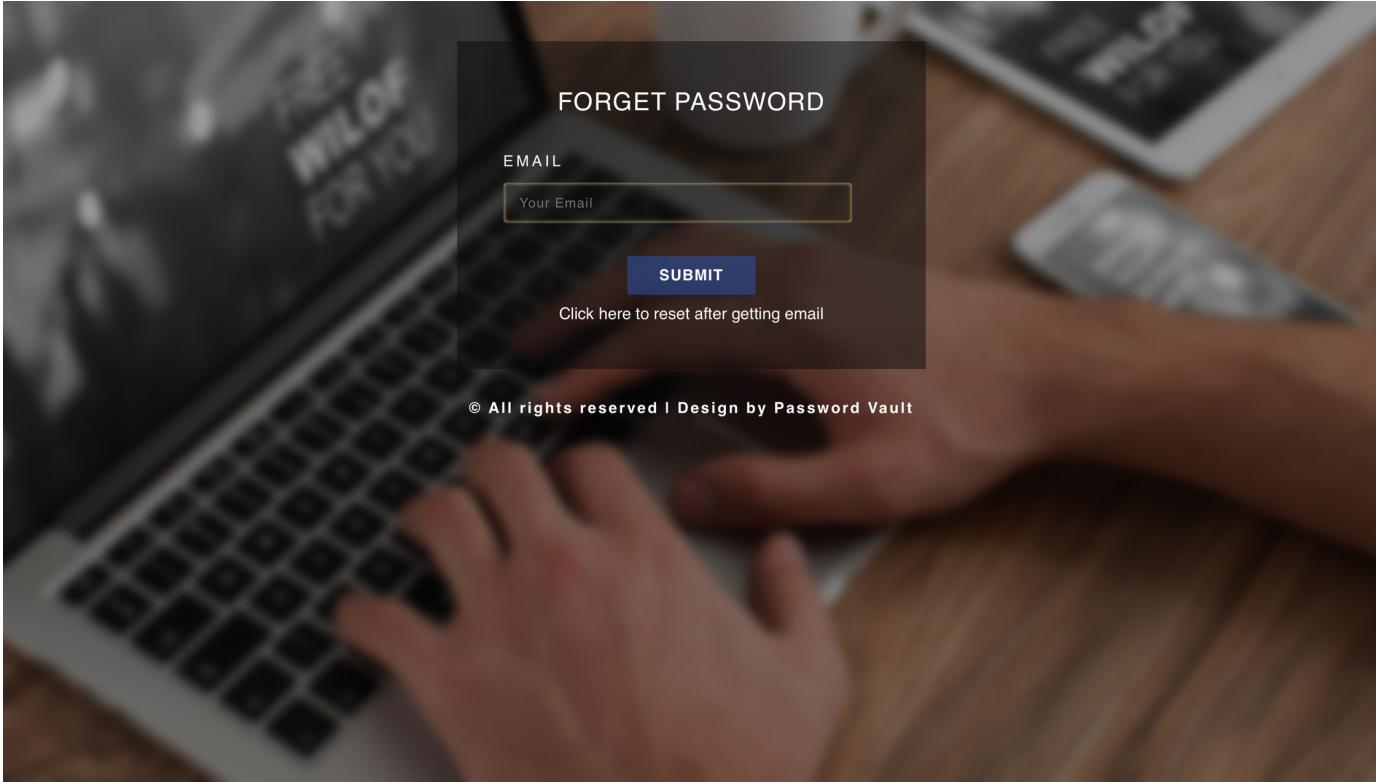
### b. User Login

Use Case Number	2
Use Case Name	User Login
Input	Email, Password
Output	User should be successfully Logged In
Description	To view the profile, user must login to the application. To login, user should click on the LOGIN button on the homepage which will redirect the user to the login page. User should enter their respective credentials, which we will authenticate from the backend. If the credentials are correct, user will see the profile page, otherwise an error message will be displayed.
Task on Gitlab	<a href="https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/21">https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/21</a>



### c. Forget Password

Use Case Number	3
Use Case Name	Forget Password
Input	Email
Output	User should be able to change the password
Description	If the user forgets the password, user will not be able to login in the application. The user has the option to change the password by clicking on Forgot Password link on Login Page. User should enter the email id, and an email will be sent with one time password. User should be able to login one time with the sent password and can change the password again by navigating to user profile.
Task on Gitlab	<a href="https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/21">https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/21</a>



© All rights reserved | Design by Password Vault

S

signupdonotreply@gmail.com

Reset password request

To: Manasvi Sharma

Inbox - Google 10:40 AM

## Forget Password Email!



Dear, manasvi!

This is your one time password: 404464

You're receiving this email because you requested a password reset for your user account at Password vault.

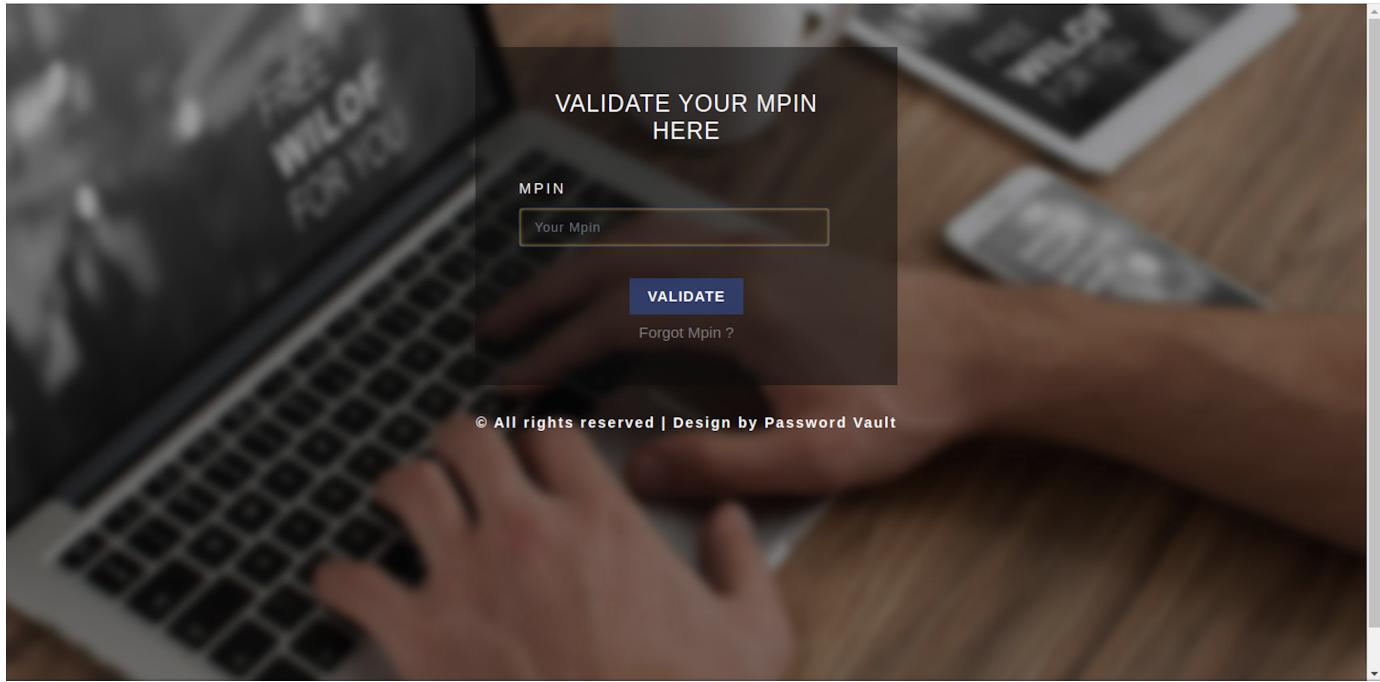
-The Password vault team

© Copyright 2022 PasswordVault. All Rights Reserved.

## 2. M-PIN Authentication

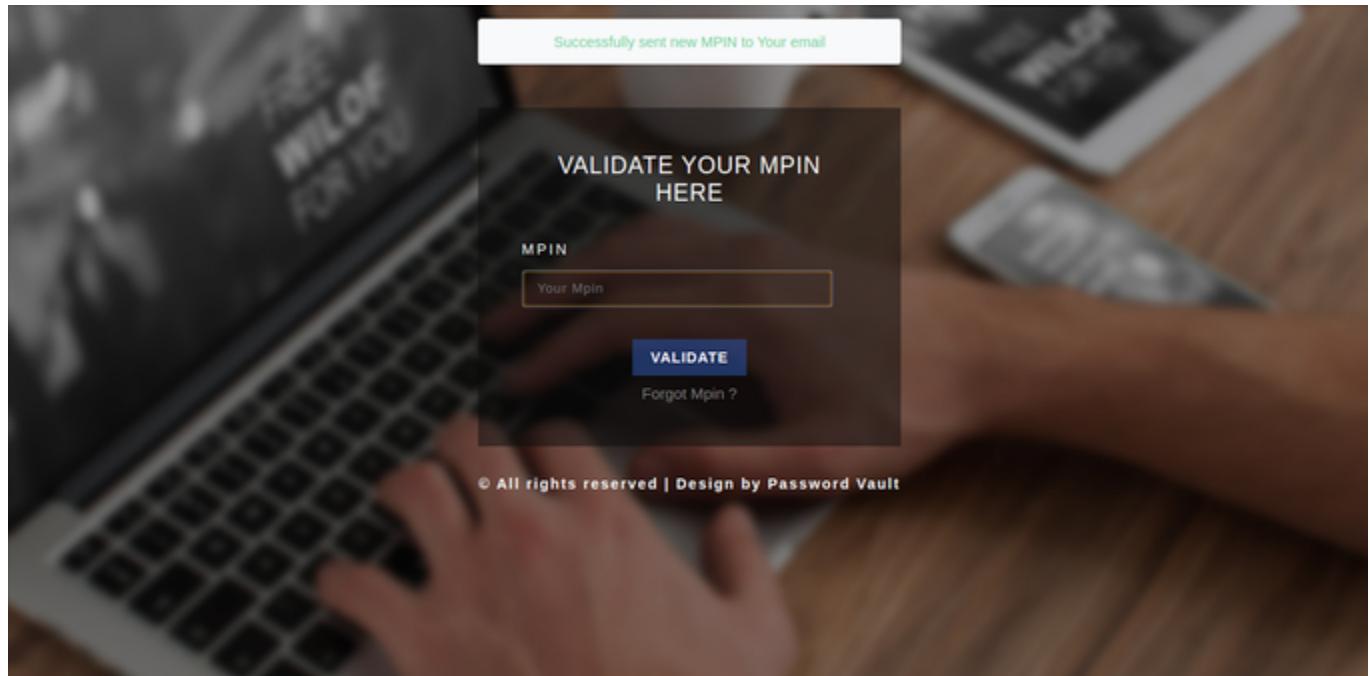
### a. M-PIN Authentication

Use Case Number	4
Use Case Name	M-PIN Authentication
Input	M-PIN
Output	User should be able to successfully see the dashboard page
Description	Since our application is storing the sensitive data like all the passwords, it is mandatory to have one more layer of authentication. User should enter the valid M-PIN in the provided textbox and click on <b>Validate</b> button. User should be able to see the profile page only after the successful authentication of M-PIN.
Task on Gitlab	<a href="https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/14">https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/14</a>



### b. Forgot M-PIN

Use Case Number	5
Use Case Name	Forgot M-PIN
Input	Click on <b>Forgot Mpин?</b>
Output	User should be able to get the new M-PIN on email
Description	If user forgets the M-PIN and not able to proceed further, user should have the flexibility to get new M-PIN via email. On the validate M-PIN page, user should click on Forgot Mpин link, and an email will be sent with the new M-PIN. The email will be sent on the email Id with which the user is already logged in.
Task on Gitlab	<a href="https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/15">https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/15</a>



S

signupdonotreply@gmail.com

Reset MPIN request

To: mnsvi07@gmail.com

Inbox - Google Yesterday at 7:13 PM

Hello Manasvi

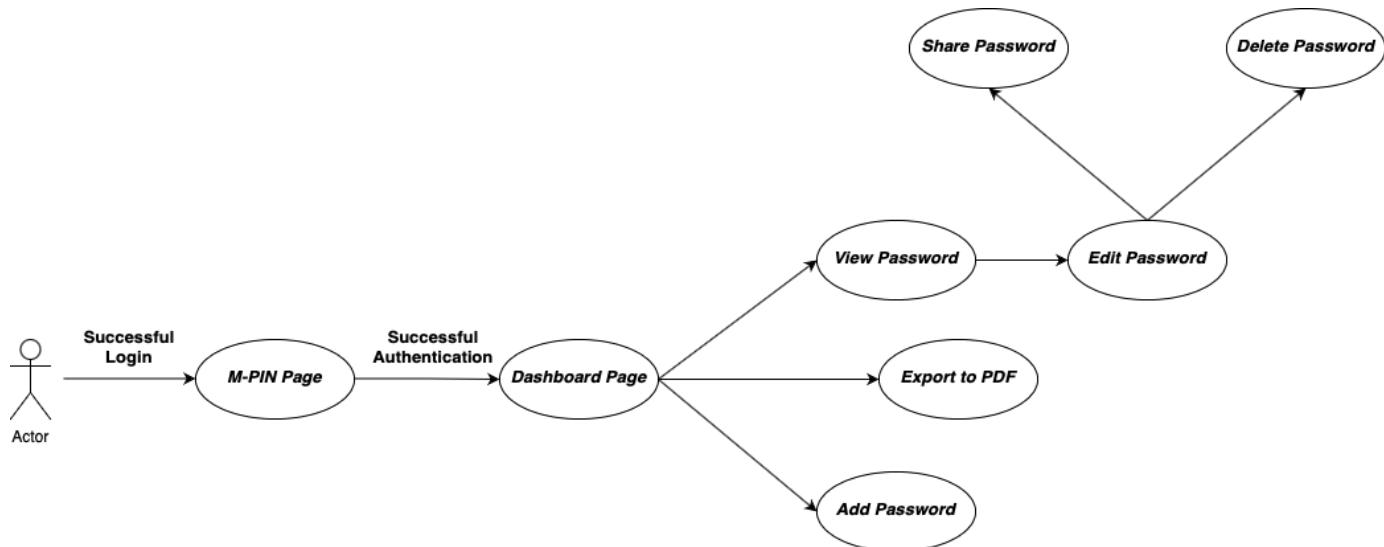
This is your new mpin: **2471**

Thanks for using our site!

You're receiving this email because you requested a password reset for your user account at Password vault

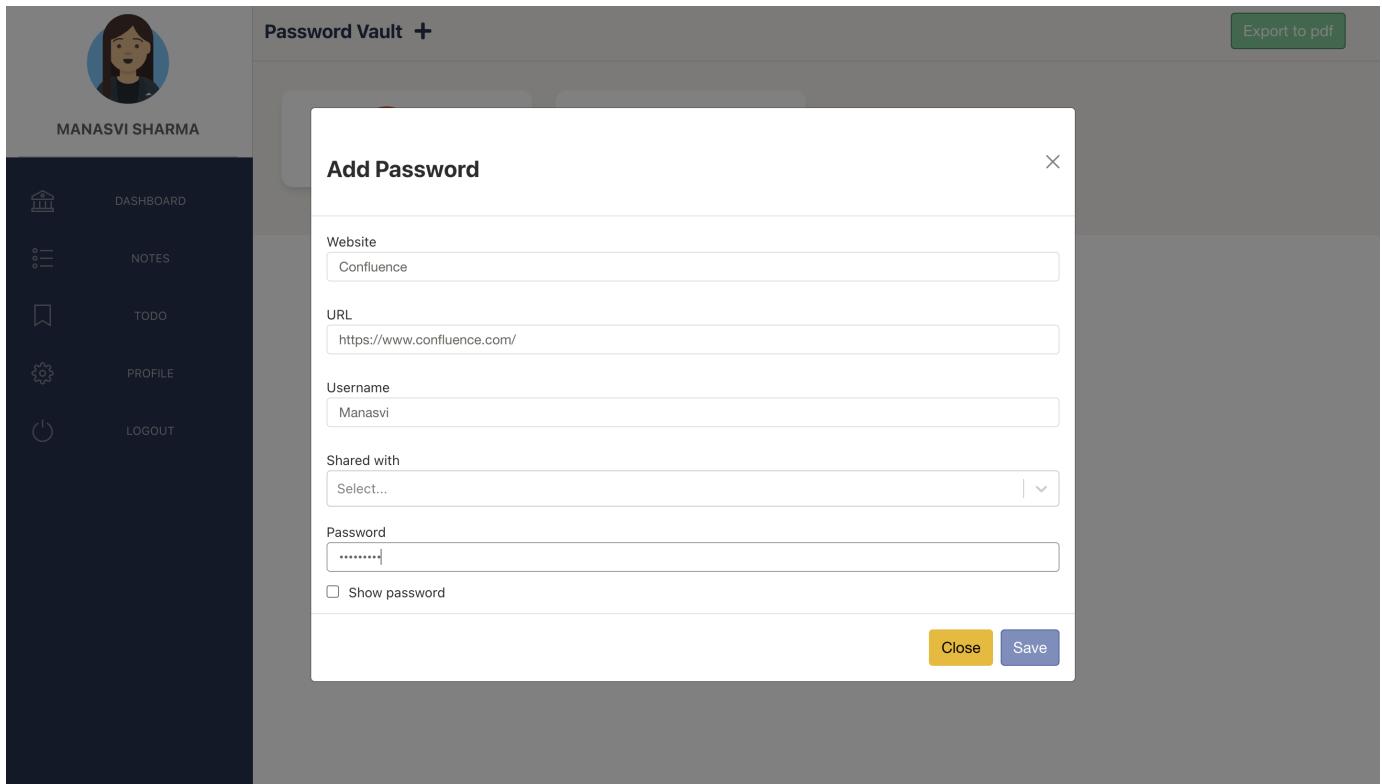
The Password vault team

### 3. Password Vault



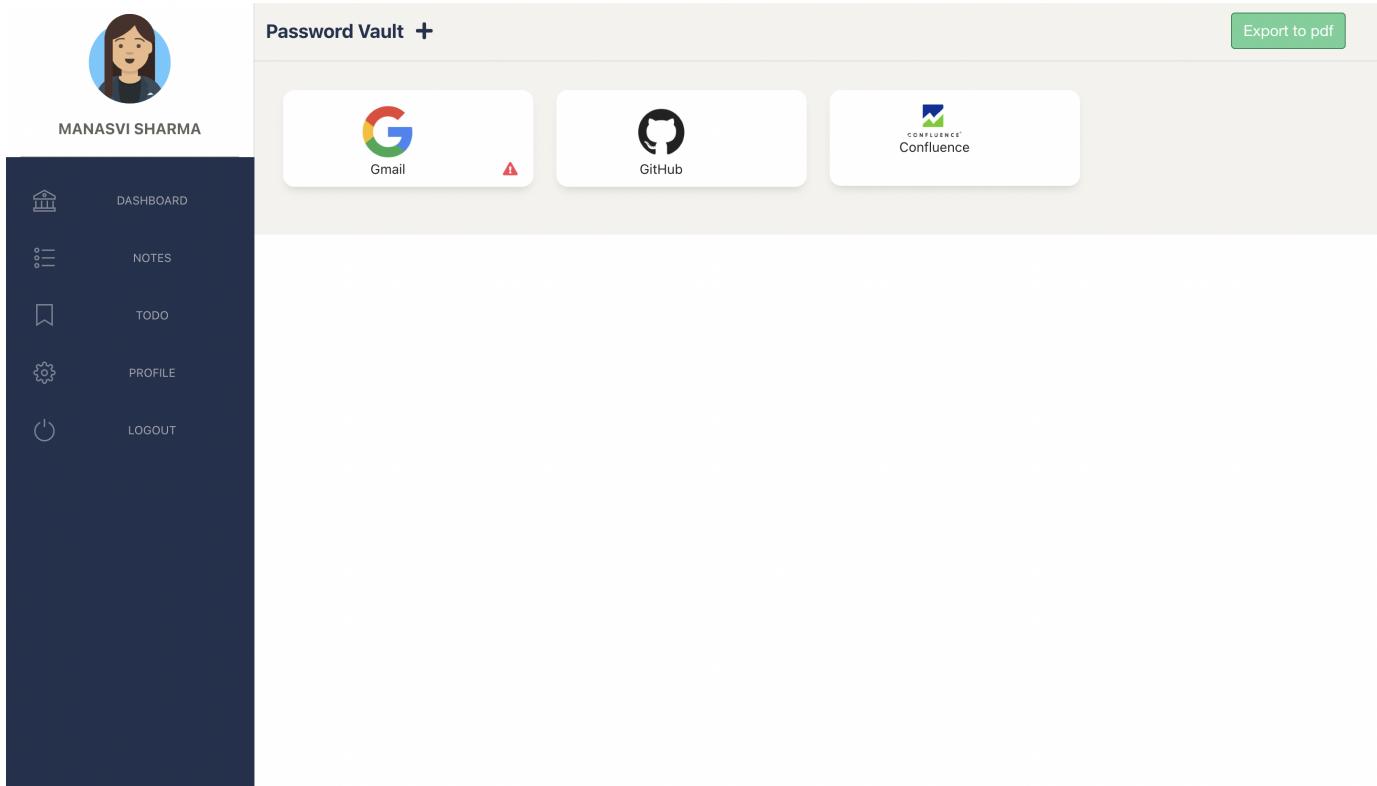
### a. Add Password

Use Case Number	6
Use Case Name	Add Password
Input	Website Name, URL, Username, Shared With, Password
Output	Password will be added.
Description	The main functionality of the application is the user should be able to store passwords for different websites. User should be able to navigate to the <b>DASHBOARD</b> page after successful login. User should click on + icon to be able to add password. A popup will appear. User should be able to fill all the required details and click on <b>Save</b> button. To cancel the creation, user should click on <b>Close</b> button.
Task on Gitlab	<a href="https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/16">https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/16</a>



### b. View Passwords

Use Case Number	7
Use Case Name	View Passwords
Input	Navigate to DASHBOARD
Output	Saved Passwords will appear
Description	Once the passwords are stored, user should be able to view all the passwords on the DASHBOARD page. Each password will appear in the form of tile with the respective website logo.
Task on Gitlab	



### c. Warning Message for Pwned Passwords

Use Case Number	8
Use Case Name	Warning Message for Pwned Passwords
Input	
Output	Warning Message will appear
Description	User should be able to view the stored passwords. If the password is a pwned password, then a warning message will appear on the particular tile: "This Password is previously exposed and very likely to be breached. we suggest to change it."
Task on Gitlab	<a href="https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/35">https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/35</a>

MANASVI SHARMA

DASHBOARD

NOTES

TODO

PROFILE

LOGOUT

Export to pdf

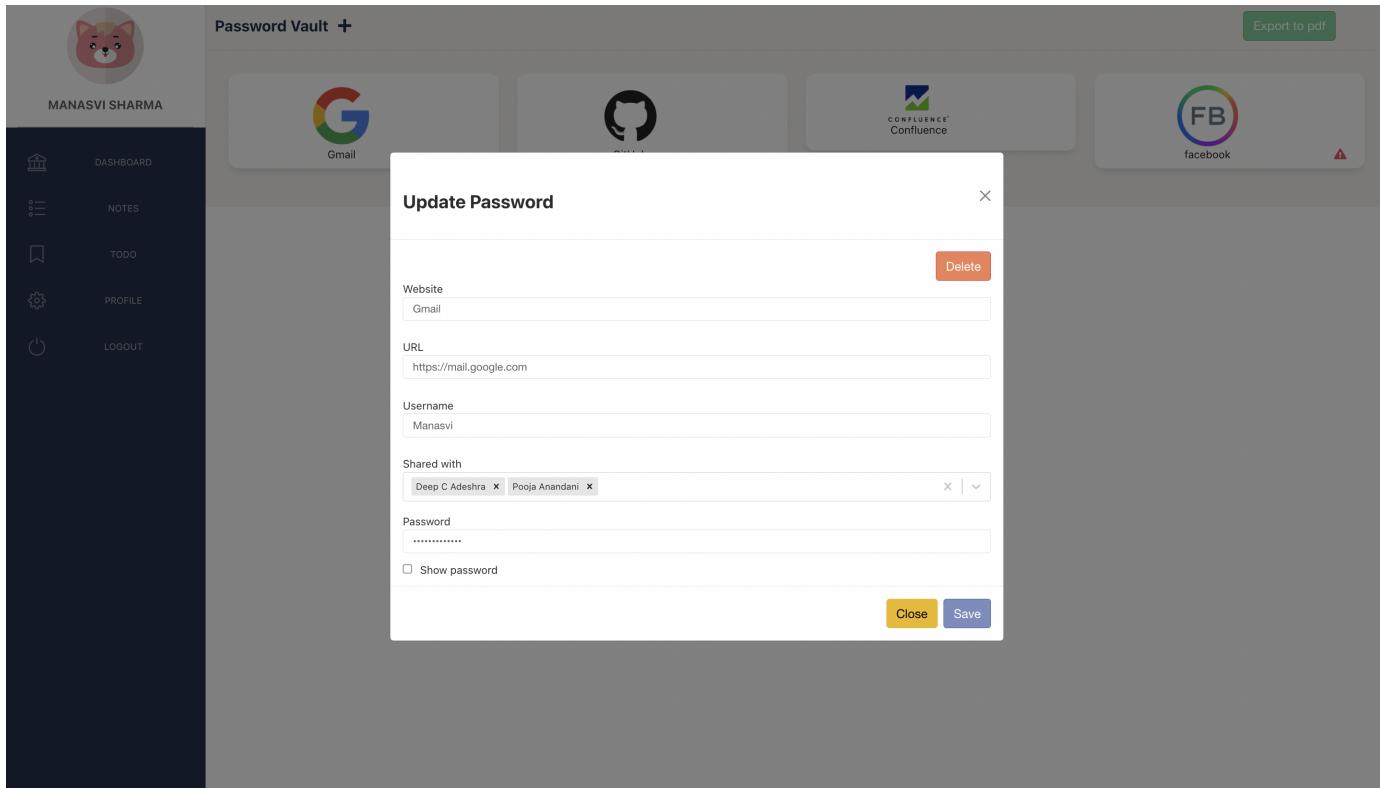
Password Vault +

This password is previously exposed and very likely to be breached.  
We suggest to change it.

Gmail GitHub

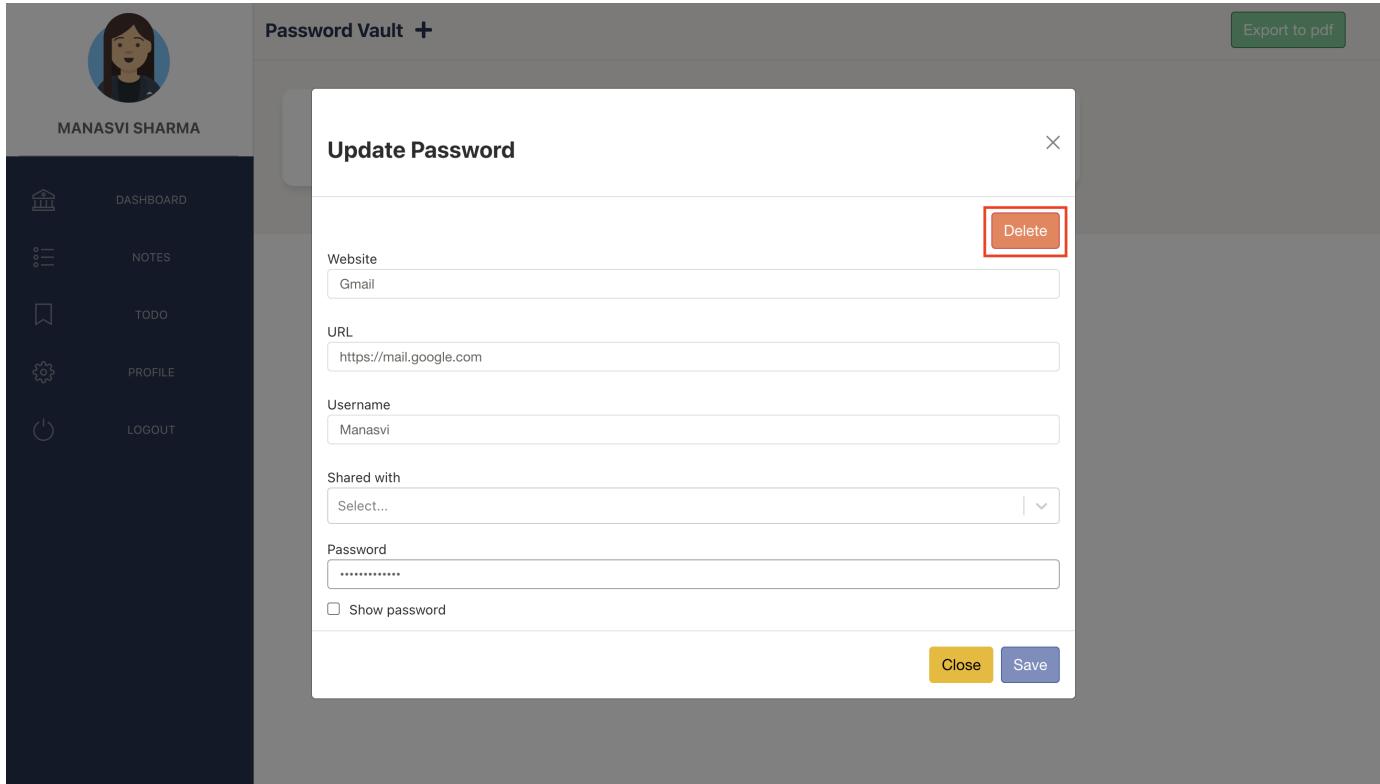
#### d. Edit Password

Use Case Number	9
Use Case Name	Edit Password
Input	Website Name, URL, Username, Shared With, Password
Output	Password will be updated
Description	User should be able to edit the passwords stored. User should navigate to DASHBOARD page and should be able to click on the password tile. User should be able to edit the required fields and click on <b>Save</b> button for the changes to get reflected. Click on <b>Close</b> button to cancel the changes.
Task on Gitlab	<a href="https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/17">https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/17</a>



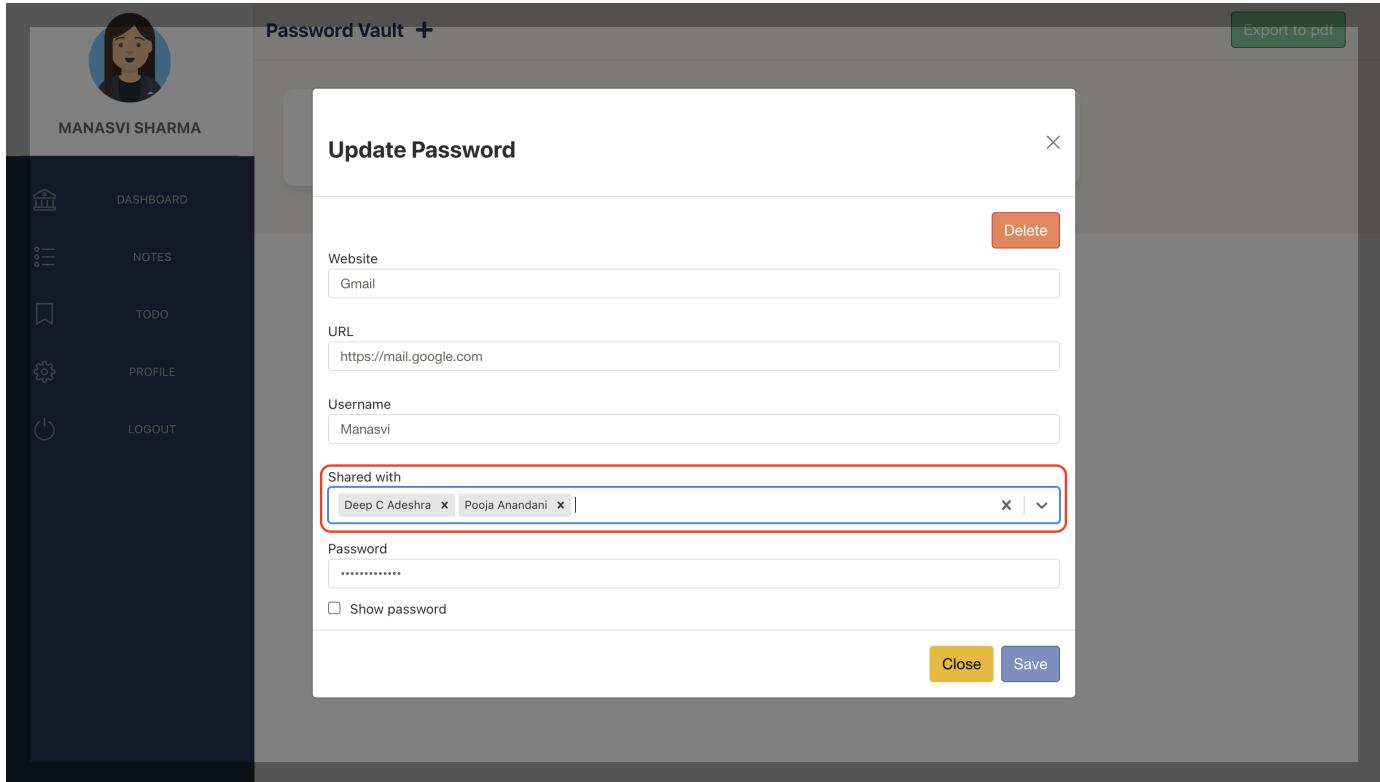
### e. Delete Password

<b>Use Case Number</b>	<b>10</b>
<b>Use Case Name</b>	Delete Password
<b>Input</b>	Click on Delete button
<b>Output</b>	Stored Password will be deleted
<b>Description</b>	User should be able to delete the stored password which is not required anymore. User should navigate to DASHBOARD page and click on the tile of the password to be deleted. One popup should appear and user should click on <b>Delete</b> button. This should delete that specific password.
<b>Task on Gitlab</b>	<a href="https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/17">https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/17</a>



## f. Password Sharing

<b>Use Case Number</b>	11
<b>Use Case Name</b>	Password Sharing
<b>Input</b>	Email ID/First Name/Last name of the user
<b>Output</b>	Password will be shared
<b>Description</b>	<p>Another feature of the password vault is the user should have the ability to share the password with other users within the platform. User should navigate to DASHBOARD page and click on the tile of the password you want to share. A popup should appear and inside the popup there should be one field <b>Share Password With</b>. User should be able to search for the person either through email id or first name or last name. User should click on <b>Save</b> button and the password will be shared with the respective user. Once the password is shared, the user with whom the password is being can only view the password not edit the password.</p>
<b>Task on Gitlab</b>	<a href="https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/37">https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/37</a>



### g. Export to PDF

<b>Use Case Number</b>	<b>12</b>
<b>Use Case Name</b>	Export to PDF
<b>Input</b>	Click on the button <b>Export to PDF</b>
<b>Output</b>	A password protected PDF file will be sent on the email ID containing passwords.
<b>Description</b>	Another interesting functionality our application provides is that user should be able to export the list of all the passwords in the form of password protected PDF. This PDF will be downloaded by the user and should be protected with password. User should navigate to DASHBOARD page and a green coloured button should be visible on the top right corner of the page <b>Export to PDF</b> . Once user clicks on the button, PDF should be downloaded. The password to open the PDF should be sent to the registered email id. User should enter the same password to open the PDF. Now they are good to view all the passwords they have stored.
<b>Task on Gitlab</b>	<a href="https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/34">https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/34</a>



MANASVI SHARMA

DashboardNotesTodoProfileLogout

### Password Vault +

Successfully exported pdf.

GmailGmailGitHubConfluence

facebook

passwords (1).pdf | Show all | X

S **signupdonotreply@gmail.com** Secure File Password To: Manasvi Sharma

Inbox - Google 2:48 PM

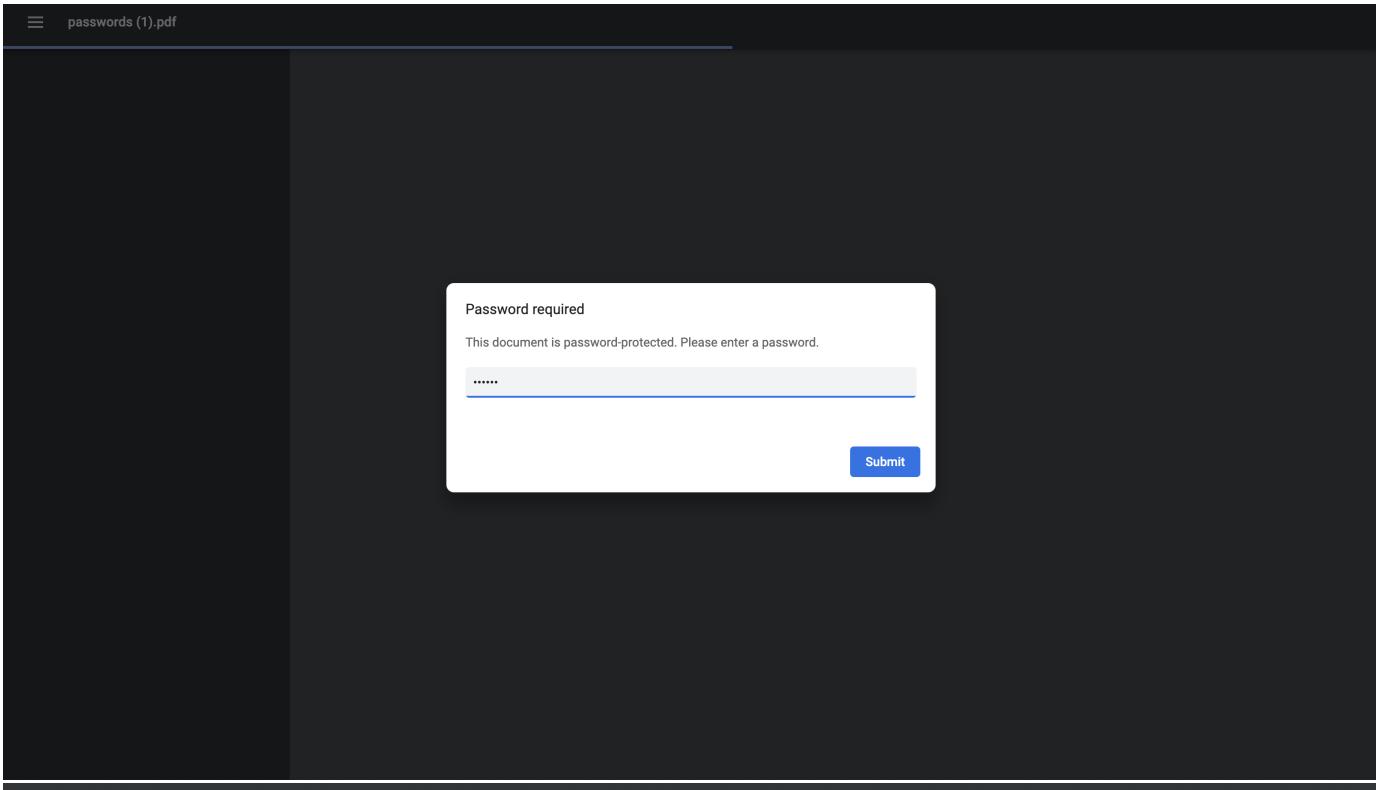


**Thank you for downloading your file Manasvi!**

This is the password for your encrypted file GZ7LT0.

You're receiving this email because you have downloaded file from Password vault.

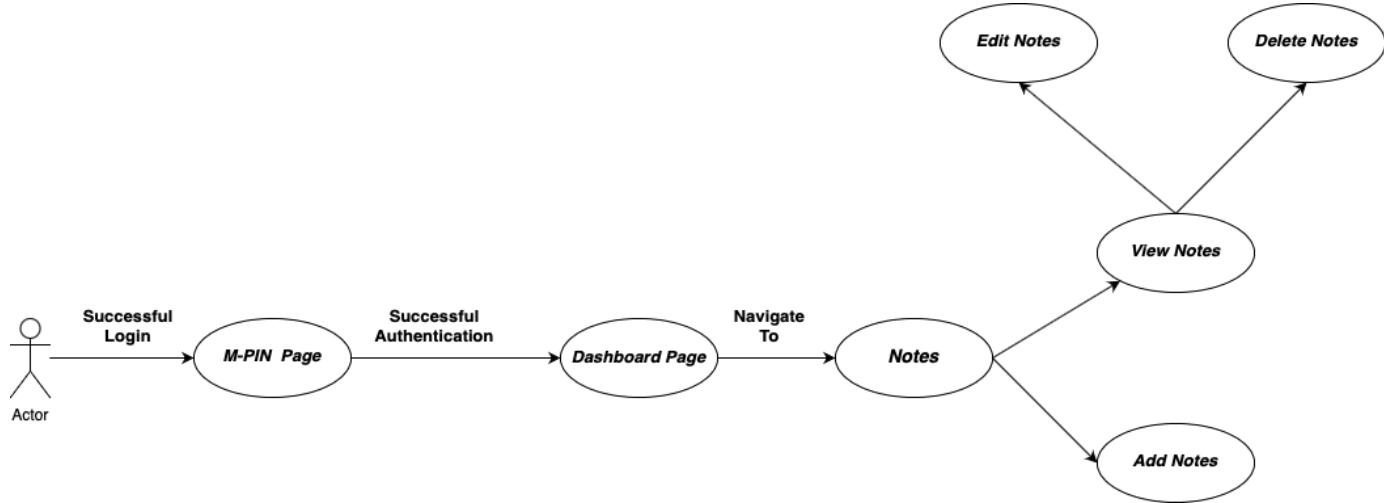
© Copyright 2022 PasswordVault. All Rights Reserved.



This screenshot shows the same PDF document after a password has been entered. The dark theme remains. At the top, the navigation bar is visible with "passwords (1).pdf", page count "1 / 1", zoom level "100%", and other document controls. The main content area now displays the first page of the PDF, which contains a table with three rows of login information. The table has columns for website URL, website username, and password.

	website_url	website_username	password
0	https://mail.google.com	Manasvi	@Tree#include
1	https://github.com/	Manasvi	@Manasvi1996
2	https://www.confluence.com/	Manasvi	@Kshi1996

#### 4. Notes



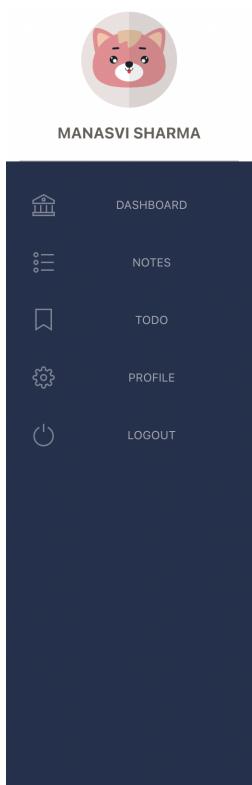
### a. Add Notes

Use Case Number	13
Use Case Name	Add Notes
Input	Title, Content
Output	User should be able to add the notes
Description	Our application gives user the flexibility to add the notes of any important thing. After login, user will see the profile page. User should be able to navigate to the <b>Notes</b> page from the sidebar of the profile page. User should be able to add the notes by clicking on the + button, which will open the page to add the required details. User should click on <b>Create</b> button after adding all the details which will add the Notes.
Task on Gitlab	<a href="https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/10">https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/10</a>

The screenshot shows the application's interface. On the left is a dark sidebar menu with icons for DASHBOARD, NOTES, TODO, PROFILE, and LOGOUT. The main area is titled 'Notes' and contains a form for creating a new note. The 'Title' field is filled with 'My Notes'. The 'Content' field contains a list: '1. ASDC Project Deadline' and '2. CSCI 5408 Project'. Below the content area are standard rich text editing tools (Normal, Bold, Italic, Underline, etc.). At the bottom right of the main area are 'Cancel' and 'Create' buttons.

## b. View Notes

<b>Use Case Number</b>	14
<b>Use Case Name</b>	View Notes
<b>Input</b>	Navigate to NOTES page
<b>Output</b>	User should be able to view all the notes.
<b>Description</b>	Once the notes are added, user should be able to view all the notes displayed on the Notes page. For each note, a tile should be displayed containing two buttons, <b>Edit</b> and <b>Delete</b> .
<b>Task on Gitlab</b>	



Notes +

My Notes

- 1. ASDC Project Deadline
- 2. CSCI 5408 Project

[Edit](#)
[Delete](#)

## c. Edit Notes

<b>Use Case Number</b>	15
<b>Use Case Name</b>	Edit Notes
<b>Input</b>	Title, Content
<b>Output</b>	User should be able to edit the details of the notes.
<b>Description</b>	User should have the flexibility to edit the details of the notes. On the view page of Notes, user should click on <b>Edit</b> button. After clicking on button, user should be able to update the details and save it.
<b>Task on Gitlab</b>	<a href="https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/11">https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/11</a>

DASHBOARD

NOTES

TODO

PROFILE

LOGOUT

Notes

Title

My Notes

Content

Normal **I** U ~~C~~ **B** **I** U ~~C~~ **T<sub>x</sub>**

1. ASDC Project Deadline
2. CSCI 5408 Project
3. Communication Assignments

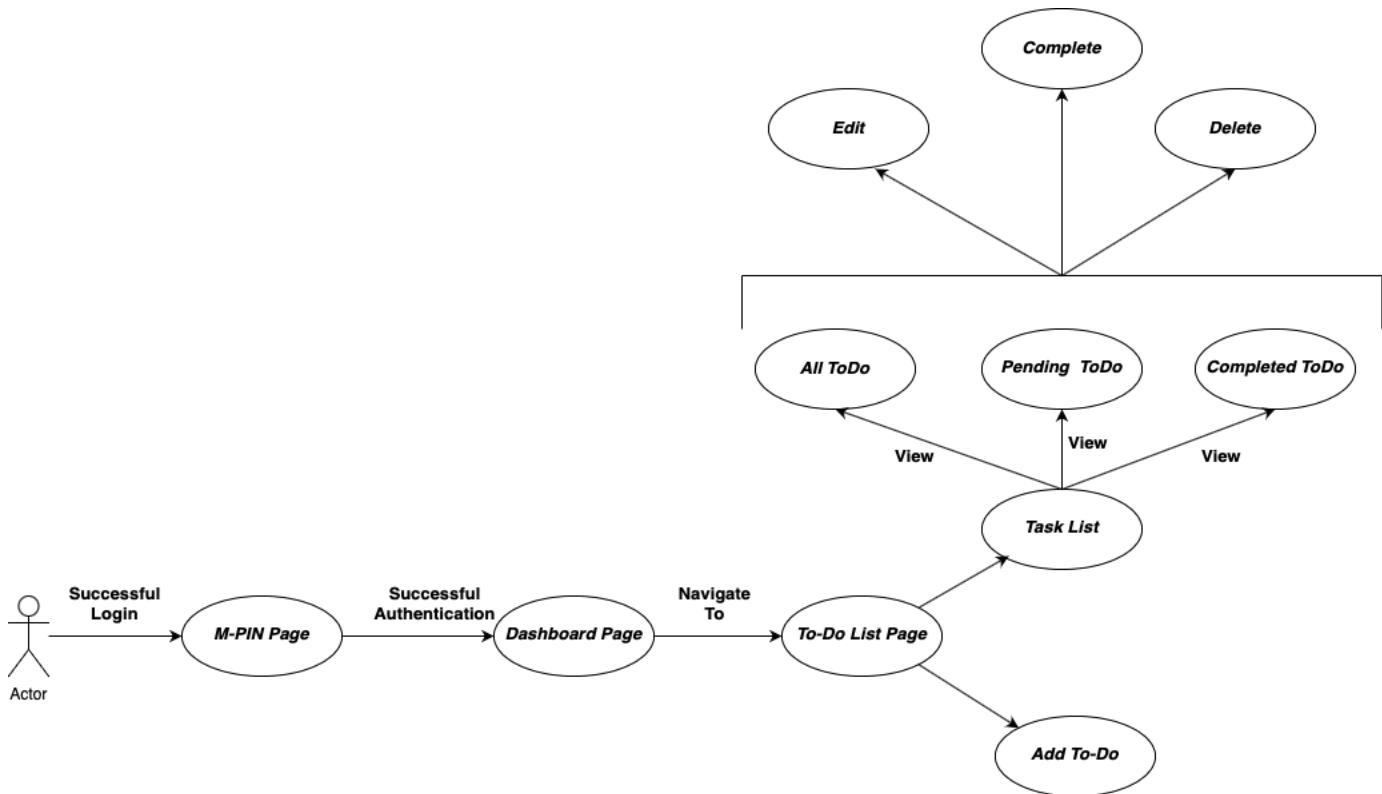
Cancel Update

#### d. Delete Notes

Use Case Number	16
Use Case Name	Delete Notes
Input	Click <b>Delete</b> button
Output	User should be able to delete the notes.
Description	User should have the flexibility to delete the notes. On the view page of Notes, user should click on <b>Delete</b> button. After clicking on button, user should be able to delete the added notes.
Task on Gitlab	<a href="https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/12">https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/12</a>



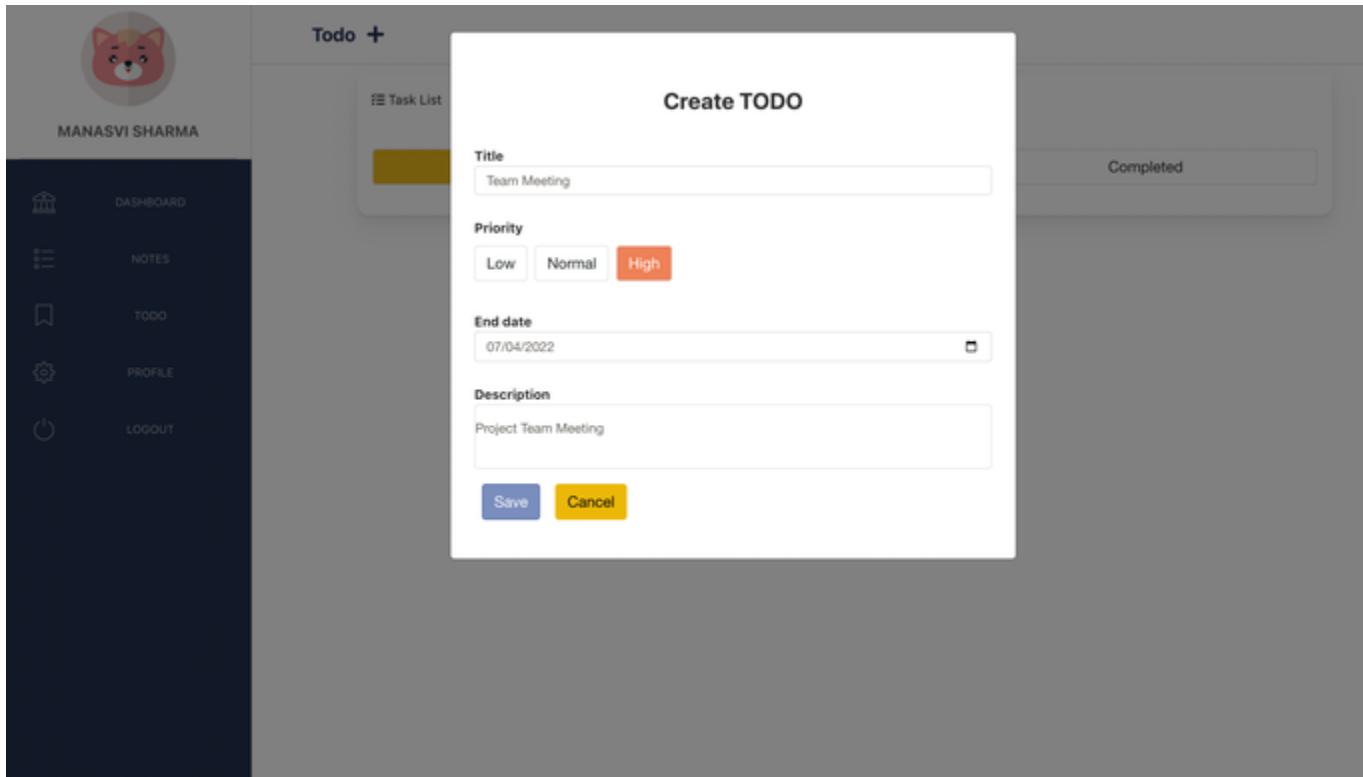
## 5. ToDo List



### a. Create ToDo

Use Case Number	17
Use Case Name	Create ToDo

<b>Input</b>	Title, Priority, End Date, Description
<b>Output</b>	User should be able to add ToDo list.
<b>Description</b>	In the application, user should be able to create list of ToDos. After logging in, user should navigate to <b>TODO</b> page, and click on <b>+</b> button. This should open a popup to add the required fields. User should enter the Title, priority (if the task is of high priority or medium priority or low priority), date of when the task should be ended and a small description. After that, user should click on <b>Save</b> button. This will create a Todo. If user does not want to proceed, they should click on <b>Cancel</b> button.
<b>Task on Gitlab</b>	<a href="https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/2">https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/2</a>



## b. View Task List

<b>Use Case Number</b>	18
<b>Use Case Name</b>	View Task List
<b>Input</b>	Navigate to TODO page
<b>Output</b>	User should be able to view the ToDo List
<b>Description</b>	After creating the required ToDo List, user should be able to view the list of tasks. To view, user should navigate to <b>TODO</b> page. The view section is partitioned into 3 sub-sections. First is <b>All</b> where all the lists whether pending or completed is being displayed. Second section is <b>Pending</b> which should display all the ToDos which are pending. Last one is <b>Completed</b> section, which should display the tasks which are completed.
<b>Task on Gitlab</b>	<a href="https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/7">https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/7</a>



MANASVI SHARMA

-  DASHBOARD
-  NOTES
-  TODO
-  PROFILE
-  LOGOUT

## Todo +

Task List

All	Pending	Completed
<p>Team Meeting <span style="background-color: #f08080; border-radius: 5px; padding: 2px 5px;">High</span> Due on 06 April 2022</p> <p>DMWA Project Deadline <span style="background-color: #ffd700; border-radius: 5px; padding: 2px 5px;">Normal</span> Due on 12 April 2022</p>	 	 



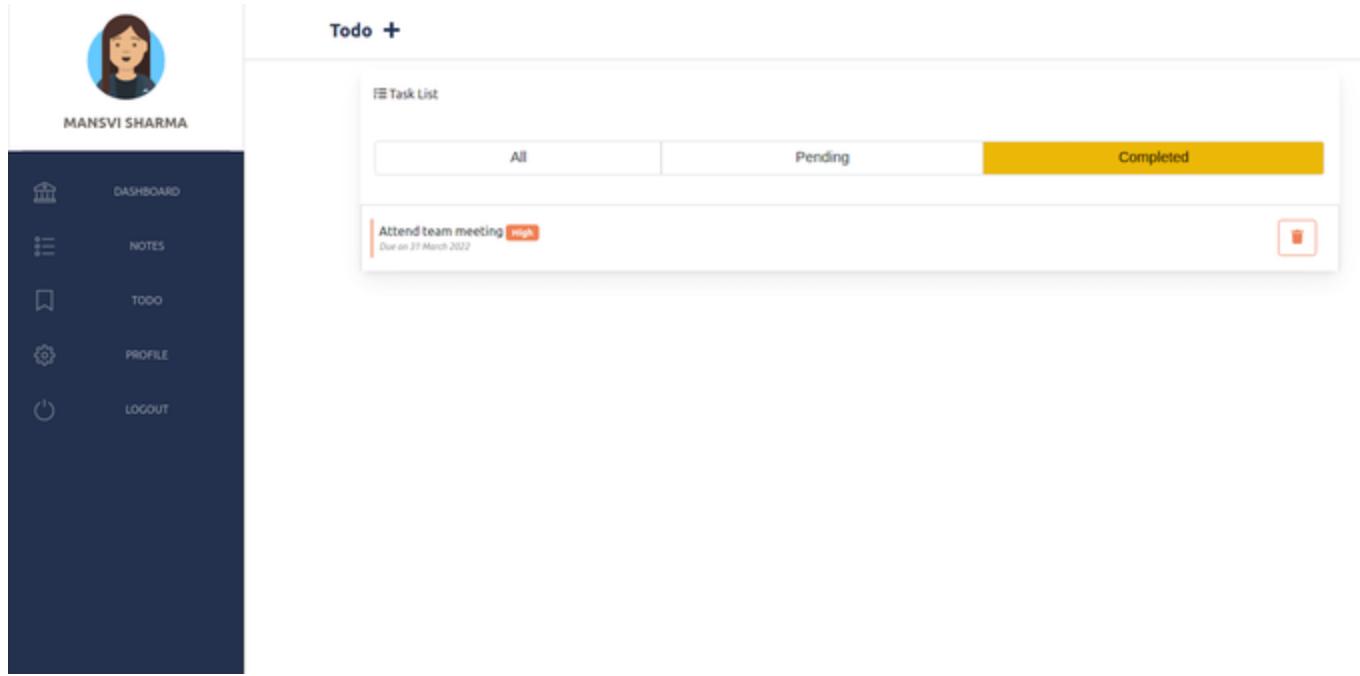
MANASVI SHARMA

-  DASHBOARD
-  NOTES
-  TODO
-  PROFILE
-  LOGOUT

## Todo +

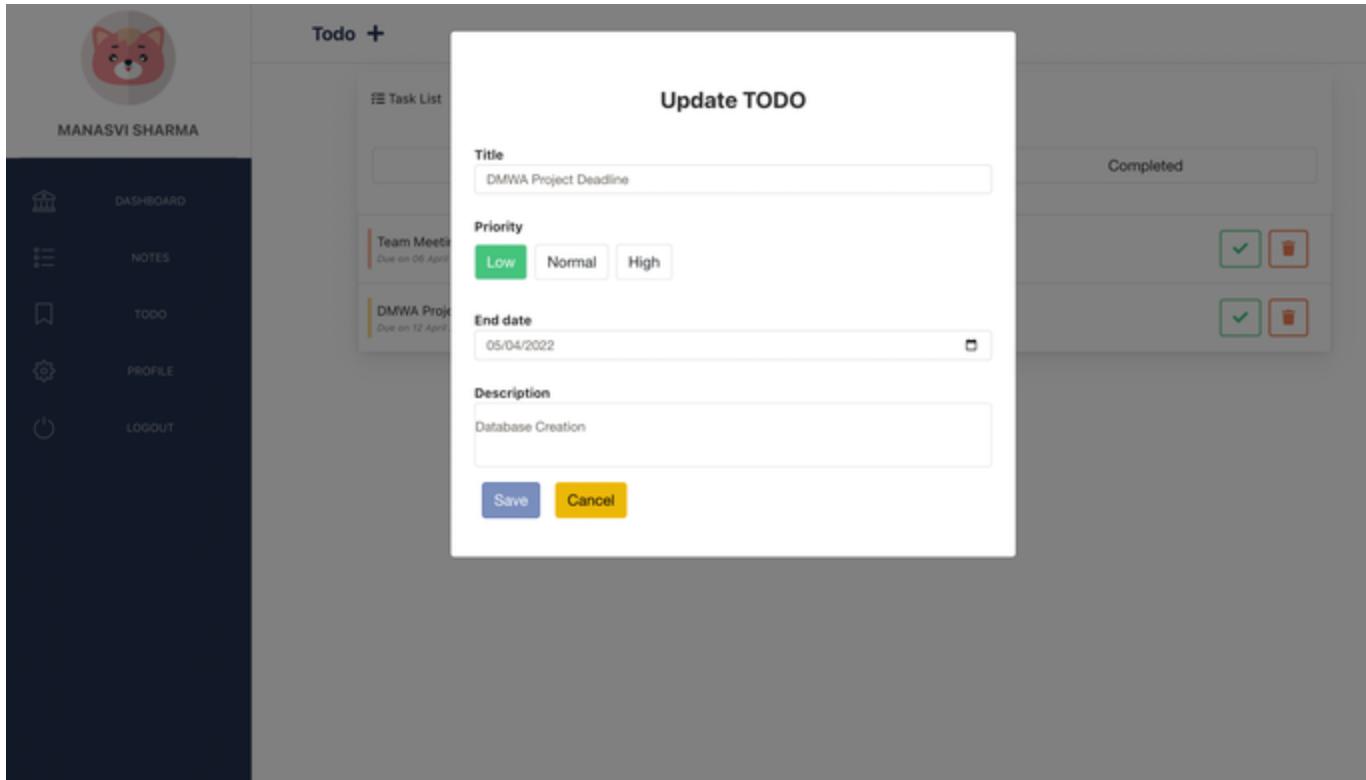
Task List

All	Pending	Completed
<p>Team Meeting <span style="background-color: #f08080; border-radius: 5px; padding: 2px 5px;">High</span> Due on 06 April 2022</p> <p>DMWA Project Deadline <span style="background-color: #ffd700; border-radius: 5px; padding: 2px 5px;">Normal</span> Due on 12 April 2022</p>	 	 



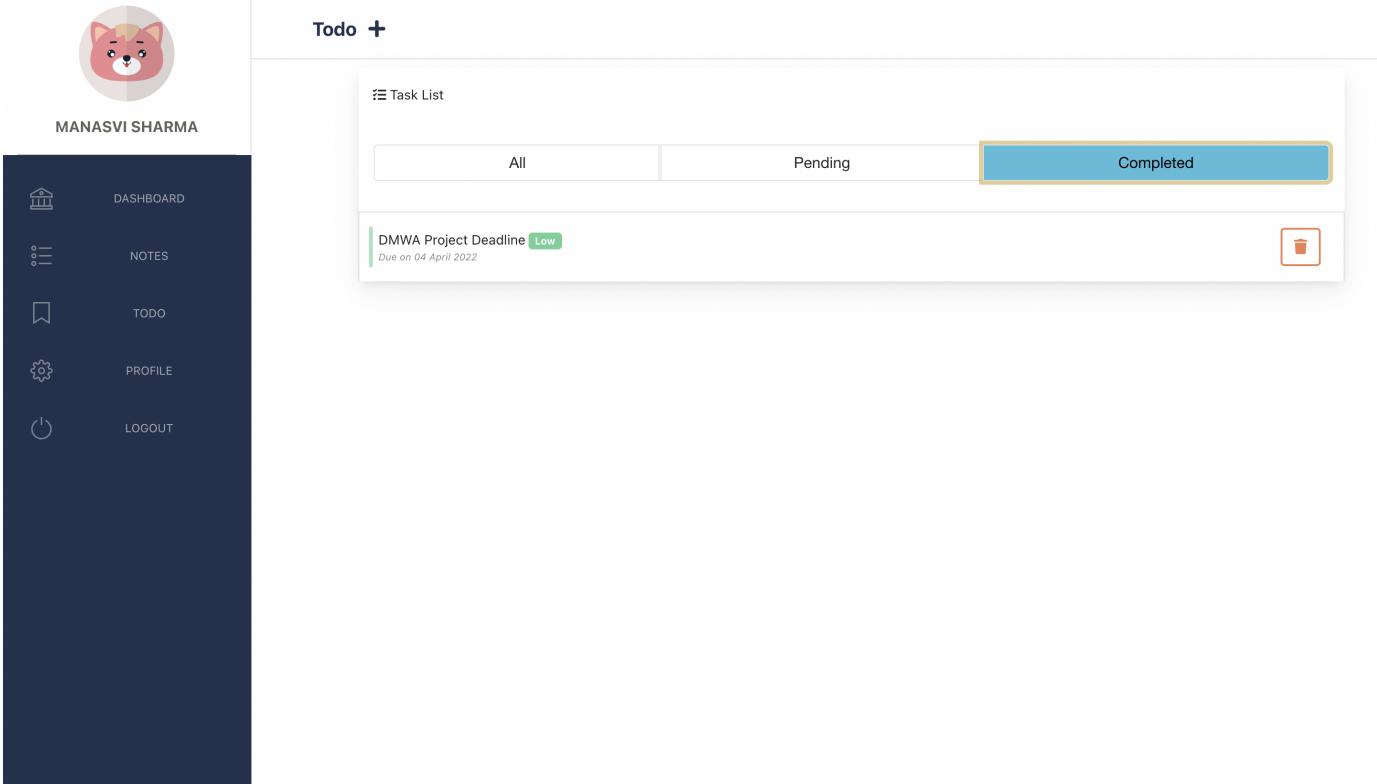
### c. Edit ToDo

Use Case Number	19
Use Case Name	Edit ToDo
Input	Title, Priority, End Date, Description
Output	User should be able to edit the ToDo List
Description	Once the user has created a ToDo list, user should also be able to edit those lists. While editing, user can edit Title, Priority, End date as well as Description. Once updated as required, user should click on <b>Save</b> button. If user wants to cancel those changes, user should click on <b>Cancel</b> button.
Task on Gitlab	



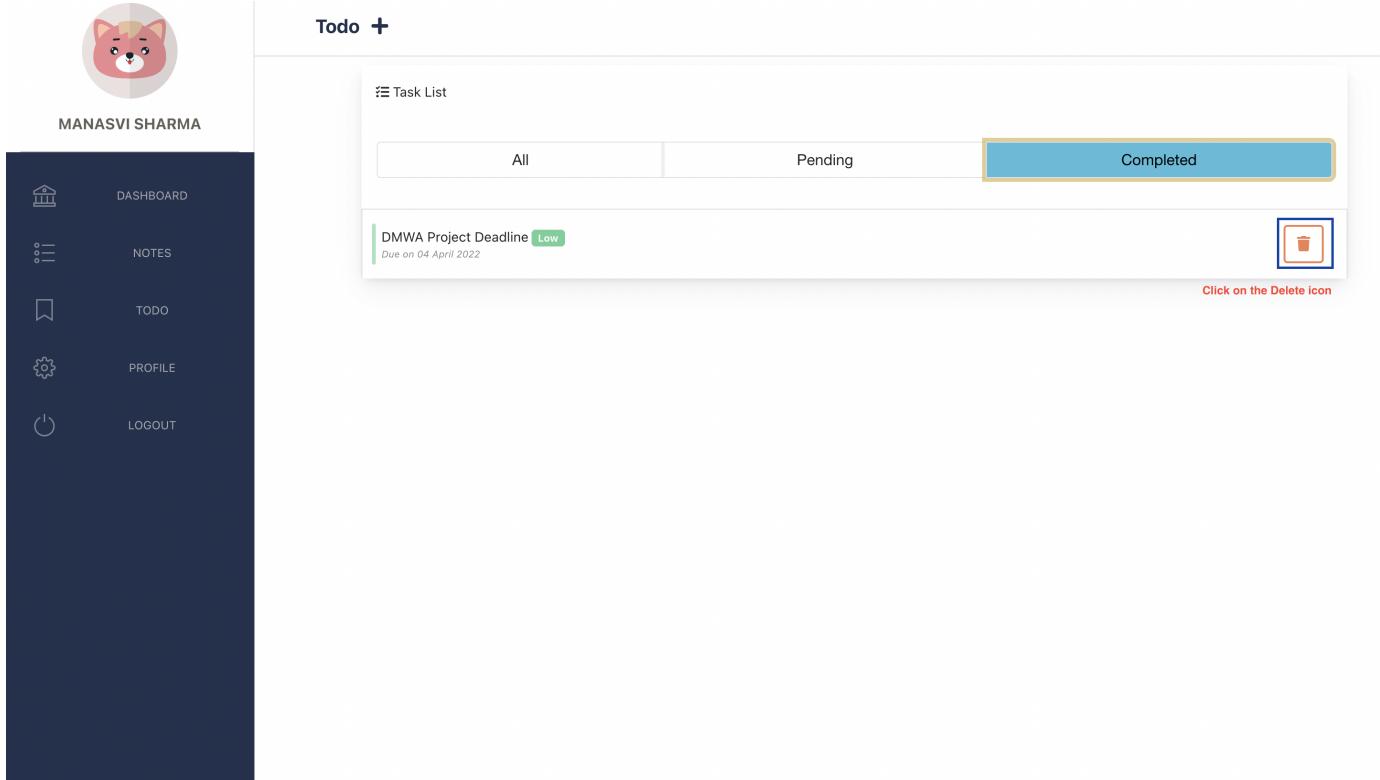
#### d. Complete ToDo Task

<b>Use Case Number</b>	<b>20</b>
<b>Use Case Name</b>	Complete ToDo
<b>Input</b>	Click on Complete Icon
<b>Output</b>	The task should move from pending tab to Completed tab
<b>Description</b>	Once the task listed is completed, user should be able to move the completed task from <b>Pending</b> tab to <b>Completed</b> tab. For this, user should navigate to TODO page and go to the particular task either in Pending or All tab. Then user should click on completed icon  . This should move the task under completed tab. Once under Completed tab, the icon  will be removed.
<b>Task on Gitlab</b>	



e. Delete ToDo Task

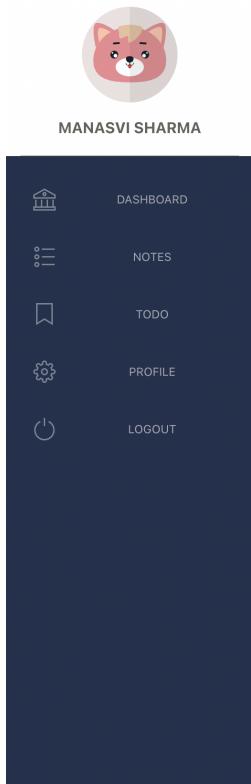
<b>Use Case Number</b>	21
<b>Use Case Name</b>	Delete ToDo
<b>Input</b>	Click on Delete Icon
<b>Output</b>	The task should be removed.
<b>Description</b>	If user wants to delete any task which is no longer required, user should click on the red delete icon. On clicking, the task should be removed from the list.
<b>Task on Gitlab</b>	<a href="https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/5">https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/5</a>



## 6. Account Settings



Use Case Number	22
Description	User should be able to update any profile related details from Account Settings. The user should navigate to PROFILE page from the side bar. The Account details should appear and user should be able to edit any details they want to. After updating details user should click on <b>Save Changes</b> button. The required details will be updated.
Input	First Name, Last Name, Email, New Password
Output	Profile Details will be updated
Task on Gitlab	<a href="https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/25">https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/25</a>
Use Case Name	Account Settings



MANASVI SHARMA

## Account Settings

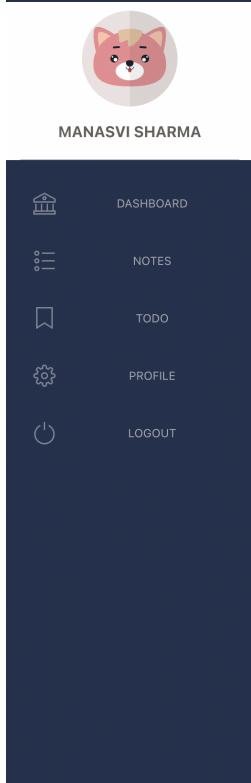
First name

Last name

Email

New Password

Save changes



MANASVI SHARMA

Profile updated ✓

## Account Settings

First name

Last name

Email

New Password

Save changes

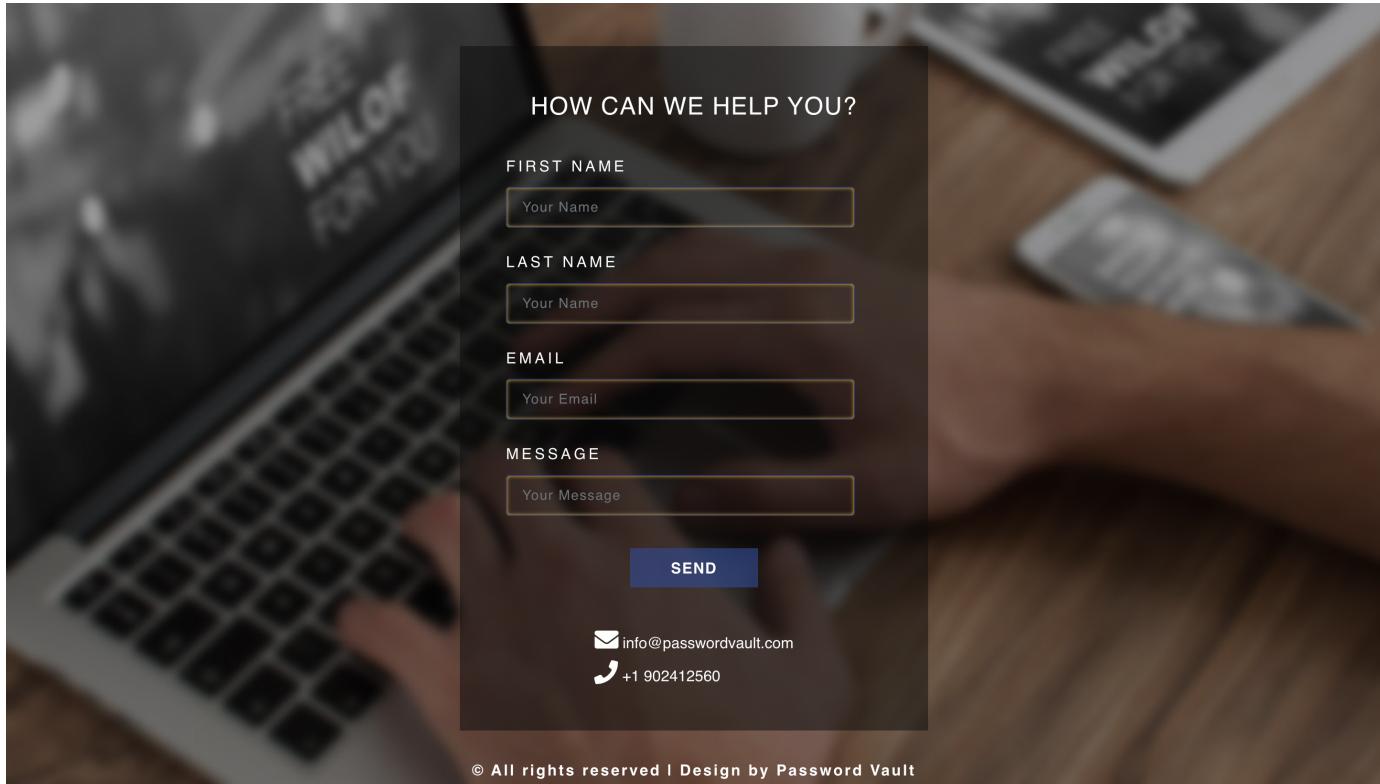
## 7. Logout

Use Case Number	23
Use Case Name	Logout

<b>Input</b>	Click on <b>LOGOUT</b>
<b>Output</b>	User should be logged out from the application
<b>Description</b>	After logging into the application, user should be able to end the session as well. User should navigate to <b>LOGOUT</b> from the side bar. After clicking on the link, user should be able to log out from the application.
<b>Task on Gitlab</b>	<a href="https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/23">https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/23</a>

## 8. Contact Us

<b>Use Case Number</b>	<b>24</b>
<b>Use Case Name</b>	Contact Us
<b>Input</b>	Click on <b>CONTACT</b> on Home Page
<b>Output</b>	User should be able to contact us.
<b>Description</b>	User should be able to contact the password vault team either through email or phone or sending direct message. User should navigate to <b>CONTACT</b> page via Home page. If user wants to send a direct message, then user should provide details like First Name, Last Name, Email, Country and Message. User should be able to email the team directly as the email id is mentioned. Moreover, user is provided with contact number as well on page.
<b>Task on Gitlab</b>	<a href="https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/40">https://git.cs.dal.ca/courses/2022-winter/csci-5308/group5/-/issues/40</a>



Someone contacted us ! ➔ Inbox x Print Compose

 [signupdonotreply@gmail.com](#) to me ▼ 2:05 PM (0 minutes ago) Star Reply Forward ⋮

Hello Admins! You have a new query from the customer. Please find details attached

Name: Manasvi Sharma  
Email: [manasvi@gmail.com](mailto:manasvi@gmail.com)  
Message: [manasvi@gmail.com](mailto:manasvi@gmail.com)

## Third-Party API

In the application, we have used 2 third-party APIs:

- <https://logo.clearbit.com/> : This API is used to easily find the company logo by simply just appending the Domain. In the application, we are using the API to fetch the logo of the website for which the user has saved the password.
- <https://api.pwnedpasswords.com/range/> : This API is used to check if the password can easily be breached or not or if the password is pwned or not. In this application, we are using the API to check if the password that user has stored for any particular website can be pwned or not. If the password can be compromised, then a warning message will be given to the user. For doing this we send the first 5 latters of 5 SHA-1 hash of the password and the API gives us matching hashes that starts with payload request we sent. from the response that API sent us, we can find the exact match of our hash which can tell us password is breached earlier or not.