PASSWORD VAULT

A Software Requirement Specification (SRS) Report

Submitted by

117553062 - Balaji Bharatwaj Manikandan

University of Maryland, College Park

Table of Contents

List of Figures

List of Tables

List of Abbreviations

CRUD Create, Read, Update, Delete

SRS Software Requirement Specificationq

Chapter 1

Introduction

1.1 Purpose

This document provides a detailed description of a Password Vault web application. This document will provide details like use-cases from the user perspective, security requirements to make the web application more secure and safe for use. This document also addresses the system requirements and the steps that users need to take without fail for efficient use of this web application. Use-case diagrams and step by step process for each use-cases will be provided in this document.

1.2 Scope of the Project

This is the era for web application. There are multiple web applications that has taken over us to provide valuable services to save us time. But that comes at a cost. This makes us remember passwords for multiple sites. This web application will be like one-stop to save all of the passwords. This web application will start with a register web page to register new users. Users must use a login password to login into accounts, and a master password to unlock the password for a particular website. The user will be able to do all the CRUD operations inside the web application, and

to be secure, the master password is required to do all of the operations.

1.3 Overview of the Document

The flow for the document is as follows. Chapter 2 - Overall Description, which will tell about the system environment, the use-cases, and misuse-cases. Chapter 3 - Requirement Specification, which will tell about the functional requirements of each use cases and it will touchbase on Non-functional requirements. 4 - Security Specification will tell about the security requirements of the web application for safe and secure password store. It will also tell about the steps that users must take to have a safe experience.

Chapter 2

Overall Description

2.1 System Environment

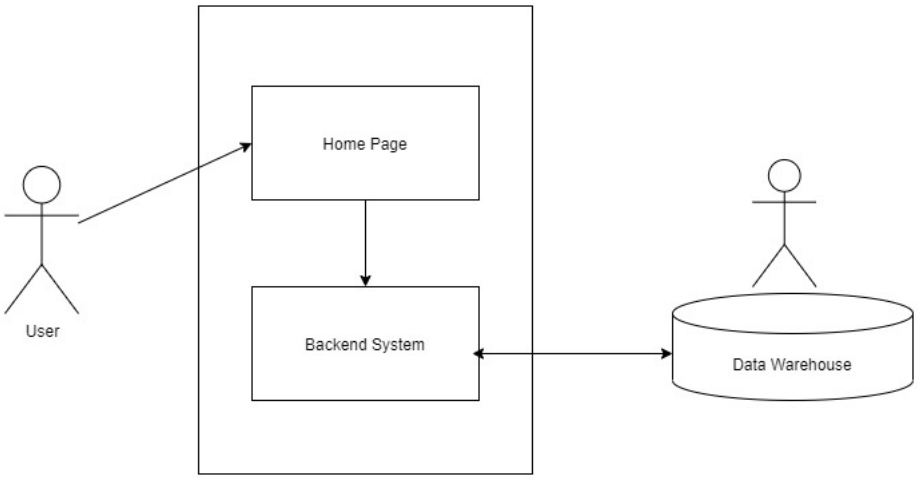

Figure 2.1: System Environment

The password vault web application has two actors. One is New/Existing user and another one is data warehouse. The new user will create an account to store passwords. The credentials for that particular account will be stored in the data warehouse. The login validation will be done from the data stored in the data warehouse.

The user can do CRUD operations. Any changes to the database can only be done when the user enters the master password. The user can copy the password directly from the web application to paste it in any other websites.

2.2 Functional User Specification

This section will contain detailed view of the User Function Requirement. This section will explain in detail about the action that a user takes for a particular use case. The actors in this system are the user and the Data warehouse only.

2.2.1 User Use-case

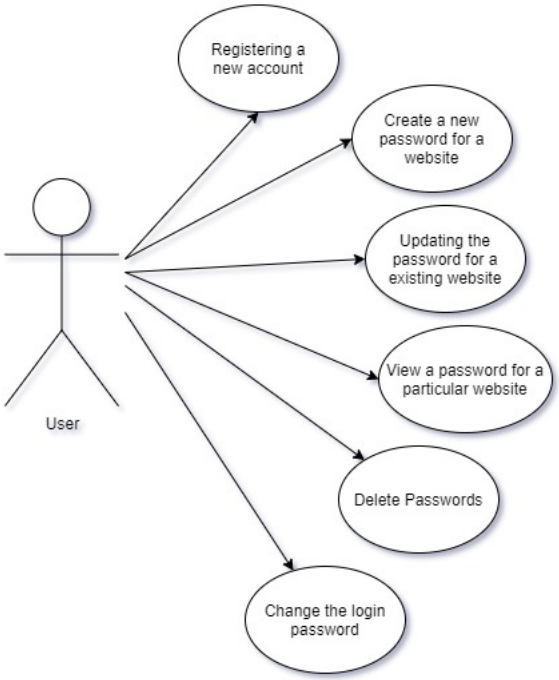

Figure 2.2: A diagram of all use cases in the User side

2.2.1.1 Registering A new Account

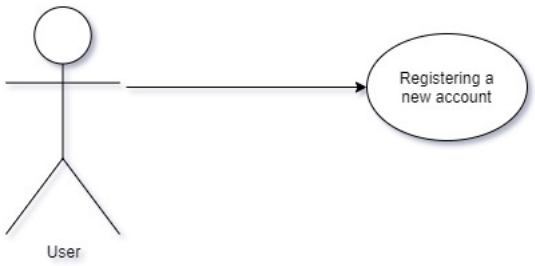

Figure 2.3: Use case - Create a new account

Description: New users are required to create a new account in order to use the Password Vault Service.

Step-by-Step Process

1. The new user searches for the web page.
2. The User reaches the landing page, the user selects "Create New account"
3. The User will enter the following details in the registration page
   (a) Full Name
   (b) Email ID (For authentication)
   (c) Login Password
   (d) Master Password
   (e) Verify Master Password
   (f) Mobile Number
4. The User selects Create account
5. The User will be taken to the home page

2.2.1.2 Store a new password for a website

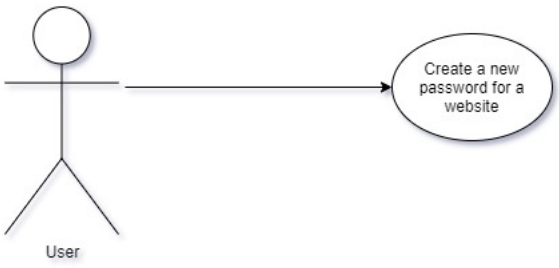

Figure 2.4: Use case - Store New Password for a website

Description: The user will store a new password for a particular website.

Step-by-Step Process

1. The User will click the "Plus" Sign on the top right corner to store a new password
2. The User will enter the following Details
   (a) Website Name
   (b) Email Address
   (c) Password
3. The User will select store password
4. The User will enter the master password upon the prompt from the web application
5. The password gets stored when the master password is verified successfully.

2.2.1.3 Update the password for a website

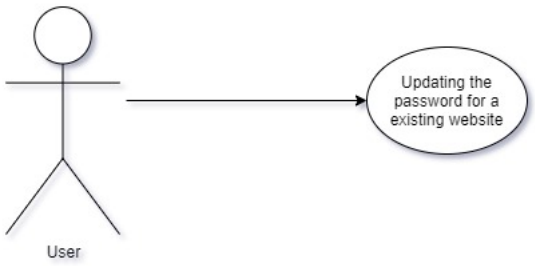

Figure 2.5: Use case - Update New Password for a website

Description: The user will Update a new password for a particular website.

Step-by-Step Process

1. The User Clicks on the card for which they wish to update the password
2. The User Clicks "Update Password"
3. The User enters the Master Password
4. Upon successful authentication, the User will enter the following details
   (a) The new password
5. The User Clicks "Update Password"
6. The password gets updated.

2.2.1.4 View password for a website

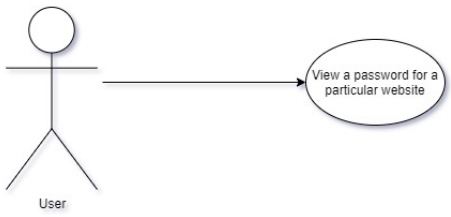

Figure 2.6: Use case - View Password

Description: The user will view password for a particular website.

Step-by-Step Process

1. The User Clicks the card for which they would like to view the password
2. The User clicks the "eye" icon to view the password
3. The User enters the master Password
4. Upon Successful authentication, the User can view the password

2.2.1.5 Delete password for a website

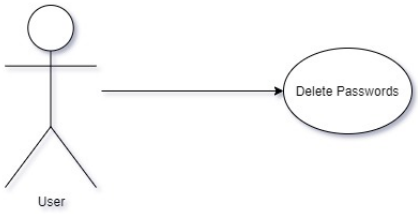

Figure 2.7: Use case - Delete Password

Description: The user will delete the password for a particular website.

Step-by-Step Process

1. The User Clicks the card for which they would like to delete the password
2. The User clicks the "dustbin" icon to delete the password
3. The User enters the master Password
4. Upon Successful authentication, the User can delete the password by clicking "Confirm" button upon pop-up.

2.3 User Characteristics

The user must know how to operate an computer and should know how to browse the internet. The user must be able to remember the master password to perform any operation in the web application. The user must not share the master password as encryption happens along with the master password (More in Chapter 4)

2.4 Non-functional Requirements

The user must have a decently powered system in order to run the website and get smooth animations on the website.The user must use a password that is in decent length and strong enough by using multiple characters. The database must be connected to high speed internet access to get faster updates and data fetch from the data warehouse. The database must be powerful enough to handle all the incoming requests without any crashes.

Chapter 3

Requirement Specification

3.1 External Interface Requirements

Connecting to database to store the data is the only External interface requirements. Connection to the database is required for the following operations.

1. Add new users
2. Add a new password for a website
3. Update a password for a website
4. Delete password for a website
5. Change login password
6. View Password for a website

3.2 Functional Requirements

3.2.1 To Create new user

Use case Name	Create a new User
Reference	Register a new Account (Section - 2.2.1.1)
Trigger	The user goes to the landing page
Pre-Condition	The user shouldn't have an existing account in this web application
Basic Path	1. The user enters the web address in the address bar 2. The browser will take to the landing page of the website. 3. The user clicks the "Create new account" button on the top-right hand corner of the navigation bar 4. The system will fetch the document that contains a form to create a new account 5. The user will enter the details asked in the form (Details are specified in the User-Use cases) 6. The user will click submit 7. The system will take the values and store it in the database
Post Condition	The user will be taken to the dashboard upon successful account creation
Exception Paths	The user may abort the new account creation

Table 3.1: To create a new user account

3.2.2 To Add a new password

Use case Name	Create a new password for a website
Reference	Store a new Password (2.2.1.2)
Trigger	The user clicks "Add Password" which is available in the dashboard
Pre-Condition	The user is present in the dashboard to add a new password
Basic Path	1. The user selects "Add Password" button available in the top-right hand corner 2. The system will fetch the form to create a new password for a website 3. The user will enter the details asked in the form (Details asked in the form is present in the User-Use case) 4. After the details are entered, the user enters the master password for confirmation, and then clicks submit. 5. System takes the form values and stores it in the database
Post Condition	The user will be taken to the dashboard. Upon successful creation the password will be available in the dashboard for use.
Exception Paths	The user may abort the new password creation

Table 3.2: To create a new password

3.2.3 Update Password for a particular website

Use case Name	Change the password for a website
Reference	Update the password for a website (2.2.1.3)
Trigger	The user selects the "Update" button inside the card that is available in the dashboard
Pre-Condition	The user must be present in the dashboard and click the card that contains the data.
Basic Path	1. The user clicks a card that contains details for that particular website. 2. The user clicks "Update" present in the top-right hand corner of the card 3. The user enters the master password. 4. The system validates the master password 5. Upon successful validation, the user enters the updated password. 6. The system takes the form value and updates on the database
Post Condition	Upon successful update, the updated password will be available in the card for the particular website.
Exception Paths	The user may abort the updating the password session

Table 3.3: To update a new password for a website

3.2.4 Viewing the password for a website

Use case Name	View Password for a webiste
Reference	View Password for a website (2.2.1.4)
Trigger	The User clicks the eye icon on the card, which is available in the dashboard.
Pre-Condition	The user must be present in the dashboard and click the card that contains the data.
Basic Path	1. The user clicks a card that contains details for that particular website. 2. The user clicks "eye" icon present in the bottom-right hand corner of the card 3. The user enters the master password. 4. The system validates the master password 5. Upon successful validation, the user can view the password for further use
Post Condition	Upon successful update, the updated password will be available in the card for the particular website.
Exception Paths	The user may abort the viewing the password session

Table 3.4: To view the password for a website

3.2.5 Deleting the password for a website

Use case Name	Delete Password for a webiste
Reference	Delete Password for a website (2.2.1.5)
Trigger	The User clicks the "dustbin" icon on the card, which is available in the dashboard.
Pre-Condition	The user must be present in the dashboard and click the card that contains the data.
Basic Path	1. The user clicks a card that contains details for that particular website. 2. The user clicks "dustbin" icon present in the bottom-right hand corner of the card 3. The user enters the master password. 4. The system validates the master password 5. Upon successful validation, the user is asked to click confirm delete
Post Condition	Upon successful Validation, the password gets deleted from the system
Exception Paths	The user may abort the deleting the password session

Table 3.5: To delete the password for a website

Chapter 4

Security Specification

4.1 Preliminary Security Requirements

This section covers a detailed view of the preliminary security requirements that should be implement as a part of the web application. The security requirements are shown in terms of user's perspective and as well as system perspective.

4.1.1 Passwords

This web application uses two kinds of passwords. One is a login password, which is used to authenticate the web application. Another password is called the master password which is used for two reasons.

1. The master password is used as a key to encrypt all the passwords that are entered in the password vault.
2. The master password is used for any operations that is done on the web application. CRUD Operation, changing login password requires a master password for extra protection.

The user is expected to use passwords with the following Rubric

1. Both Login and Master Password must be more than 8 characters in length
2. Login password must have a Uppercase letter, a special character, and numbers
3. Master password can have the same rubric as login password.

The master password is really important for this web application. Because the master password is used to encrypt the passwords added to the password vault. The user is also not expected to give save password on their respective web browsers for extra security

4.1.2 Web Application Security Requirements

This application runs using PHP with MySQL database. With MySQL, SQL injection is one of the known attacks for a very long time. In order to combat that, the back end will have prepared statements and the form values taken from the front end will be sent as a parameter to the prepared statements. This will make sure the backend and frontend doesn't directly interact and there will be an middleman to handle the incoming data. The prepared statements are compiled which means the form values will be parameters to the statements just like entering inputs to an already compiled C program. This way SQL injection can be averted.

All the CRUD operations will have its own separate file. These files will be included in the main file, where the incoming form values are captured and the operations are performed. One of the reason doing this way is to make sure each function as its own separate file for easy categorization. But also in the event of any errors, it will be easier to find the heart of the error.

4.1.3 Encryption

Encryption is an important component when it comes to storing passwords securely. But implementing the encryption from scratch requires expertise in cryptography and data handling. But luckily there are GitHub Repositories that provide precise, highly efficient PHP based encryption. ADD REFERENCE HERE has a PHP based encryption which this web application is going to use.

4.1.3.1 Procedure

The user will enter the login password and a master password. When a user tries to add a new password to the vault, instead of storing the password directly to the database, the password is encrypted using the master password. The encrypted password is then stored in the database. The login password is encrypted with the master password. When the user changes the login password, the encryption of login password using the master password will happen again and gets stored in the database.

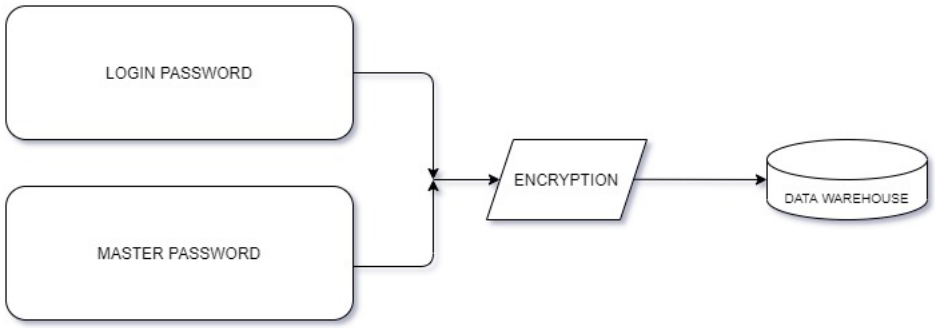

Figure 4.1: Encryption of Login password with Master Password

4.2 Session Management

Session Management is really important for a web application that has login credentials. This web application will implement session management.When a user logs out of the web application, the user cannot go to the dashboard by clicking back button on the web browser. Once the user log in to the application, a new session is created. And the session is carried out till the user is log-ed in to the application.

Chapter 5

Bibliography

[1] PHP Encryption Github Repository:https://github.com/defuse/php-encryption

[2] Use Case Diagram Tool: draw.io

[3] SQL Parameterized Queries: https://stackoverflow.com/questions/60174/how-can-i-prevent-sql-injection-in-php