CPE 334 Software Engineering

Term Project Report

TOPIC : PassBrain Password Manager & Generator Web Application

Website URL : https://pass-brain.vercel.app/

By

1. https://github.com/Asch-sys
2. https://github.com/Cloclodudu

Abstract

In today's digital landscape, where online security is paramount, the need for robust password management solutions is more pressing than ever. Our project, a password generator and manager website, directly addresses this need by offering a dual-functionality platform. Primarily, it focuses on the generation of strong, unique passwords with a selected and expensive set of settings for the users to freely modify, an essential feature in safeguarding online identities and sensitive data. Secondary to this, the website provides secure storage and management of these passwords, further enhancing user convenience and security.

This website stands out in the field of cybersecurity tools by offering a user-friendly interface that simplifies the often complex process of creating and managing passwords. While many existing solutions offer either password generation or management, our project effectively combines both, ensuring a comprehensive approach to password security. This integration is especially crucial in an era where the average user juggles numerous online accounts, making the management of different passwords a significant challenge.

Developed using HTML, CSS, and JavaScript, with Firebase as a backend, the website's architecture is designed to be both robust and accessible. Our team employed the Scrum framework, a testament to our commitment to agile, iterative development and continuous improvement. This approach facilitated a dynamic and responsive development process, enabling us to rapidly adapt to user feedback and evolving security standards.

The project's significance lies not only in its technical prowess but also in its contribution to the broader conversation about online security. By making strong password practices more accessible, we aim to empower users to take proactive steps in protecting their digital lives, a goal more critical now than ever before.

Summary

Introduction

In an era where our digital footprints intertwine with our real lives, the significance of robust cybersecurity measures will soon be as significant as our lives. The complacency towards password security, the most fundamental element in safeguarding our digital identities, is a ticking time bomb in our interconnected world. The habitual use of weak, recycled passwords across multiple platforms is akin to leaving our digital doors unlocked, inviting unforeseen threats. This negligence not only jeopardizes individual security but also becomes the weakest link in the larger chain of data protection.

The alarming ease with which passwords can be deciphered, owing to their predictability and repetition, underscores a critical vulnerability. Databases are incessantly breached, spilling passwords like sand through a sieve, and dictionaries of common passwords are widely distributed in the darker corners of the internet. This landscape vividly echoes the narratives of Perlroth's book: "This is how they tell me the world ends", where the fragility of our digital fortresses is laid bare. If a password is the fragile link, it becomes a gateway for exploitation and so often in the most random and untargeted attacks. Conversely, a strong, unique password can be the first line of defense against these opportunistic breaches.

In our project, we confront this ubiquitous challenge head-on. By deploying a password generator and manager, we aim to fortify this first line of defense, making sure that it will not be this time the weak link in the system, and ensuring the user to stay safe in face of the random attacks that exploit human error and predictability. Our technology while not edge-cutting, aims firstly to push user to takes a first critical step towards cybersecurity, addressing a glaring gap in its armor: the human factor. This approach is not just about creating stronger passwords; it's about nurturing a culture of security consciousness, where every keystroke in our password creation process counts towards fortifying our digital defenses.

Next, we will aboard the problem statement, followed by our project objective and scope, outlining the possibility and boundaries of our project. We will continue with the navigation map that will give a general overview of the functioning of the website. The wireframe design will offer a visual and structural blueprint of our system, while the user interface design will show the result. This will be followed by the use case diagram and the use case narratives which illustrate the various user interactions and scenarios our system is designed to handle. The component diagram, the class diagram and the sequence diagram delve deeper into the technical anatomy of our solution. The technology and development part will explain the technical aspect chosen to

operate this project. As we near the end, the self-evaluation & problems encountered offers a reflective look at our journey, acknowledging the hurdles we encountered and the lessons learned. And finally the conclusion & opening let us summarize our endeavor but also cast our gaze forward on the possible evolution of this project.

Problem Definition

In today world, it has been estimated that the average Internet user has around a hundred business and personal accounts, adding to this the studies done by Ellisphere.com that reported an augmentation of cyberattacks since the Covid pandemic of 400% making it easier to understand the scary picture that get draw in front of our eyes. This alarming rise is not just a statistic; it's a clarion call for a change in how we manage our digital keys. While cybersecurity engineers handle these issues for businesses, who takes care of the security of the personal online user's accounts ? The answer is pretty much, the personal user himself.

In this present digital world, individuals encounter the challenge of having to remember numerous passwords to access online services. This task presents a lot of challenges, risks, and inefficiencies. If we take weak or repetitive passwords, they are often created by users who don't know a lot about computer, making them vulnerable to cyber-attacks. There's to many passwords that may also be forgotten or misplaced, resulting in multiple and severe consequences including the loss of personal information, money, or time. And managing multiple passwords can be bothersome and may lead to storing them in insecure or impractical locations such as text files, post-it notes, or web browsers. So the aim of this project is to address the problem of secure and efficient password creation and management.

Passbrain, our response to this call, is conceived to tackle this multifaceted challenge. It is more than a password manager; it is a guardian of digital identities. With Passbrain, the creation of robust, unique passwords for each account becomes effortless and secure. By eliminating the need to memorize these passwords, we address the core of the problem: the human tendency to opt for convenience over security. Our tool is designed for the layperson and the expert alike, ensuring that everyone, regardless of their technical prowess, is equipped to fortify their digital presence against the relentless tide of cyber threats.

In this digital epoch, where each online account is a gateway to personal and professional realms, our project stands as a bulwark against the vulnerabilities that plague our online experiences. We aim not just to provide a tool but to usher in a shift in mindset – from passive users to active defenders of our digital selves. This is our commitment in the face of an escalating cyber onslaught: to empower every individual with the means to secure their digital world, one password at a time.

Project Objective & Scope

Our project's primary objective is to encourage and facilitate the adoption of robust password practices among users. We aim to achieve this by providing a user-friendly, dual-function platform comprising a password generator and a password manager. This tool is designed not to outperform established giants in the cybersecurity field but to serve as a vital step towards personal cybersecurity responsibility. It is a bridge for users to transition from weak, repetitive password habits to a regime where every password is a unique, strong, and secure gatekeeper of their digital identity.

1) Password Generator: The password generator component of our project is engineered to create complex, hard-to-crack passwords. It is designed with simplicity in mind, allowing users of all technical proficiencies to generate secure passwords that meet established security criteria.

2) Password Manager: The password manager serves as a secure vault where users can store and manage their passwords. This component aims to eliminate the need for users to reuse passwords across multiple sites, a common practice that significantly undermines digital security.

3) User Education: An integral part of our project's scope is to educate users about the importance of strong, unique passwords. Through simple and intuitive interface design, we seek to instill a heightened awareness of cybersecurity practices.

4) Accessibility and Ease of Use: The platform is designed to be accessible and easy to use, ensuring that users are not deterred by complexity from adopting better password habits.

5) Data Security: While we do not claim to offer the most advanced security features, the integrity and security of user data are still paramount in our system design. Our platform employs adequate security measures to protect user data from unauthorized access.

6) Continuous Improvement: Acknowledging the ever-evolving nature of cyber threats, our project includes a roadmap for continuous improvement and adaptation to emerging security challenges.

Through this project, we do not envision a complete overhaul of the existing cybersecurity infrastructure but rather seek to plug a critical gap – the human element in password security. By making it simpler and more intuitive for users to generate and manage strong passwords, we take a significant step towards mitigating the risks posed by human error and complacency in digital security practices. Our project, therefore, stands as a testament to the idea that every step, no matter how small, counts in the journey towards a more secure digital world.

Functional Requirements

- System must allow a logged out user to create a new account using a username and a password.
- System must allow a logged out user to log into their account using the username and the password.
- System must allow a logged in user to generate one or multiple password(s) according to their chosen settings.

- System must allow a logged in user to choose the length of the generated password
- System must allow a logged in user to choose the symbols that will be used in the generated password
- System must allow a logged in user to choose the quantity of password that should be generated.
- System must allow a logged in user to choose whether to include or not numbers in the generated password.
- System must allow a logged in user to choose whether to include or not lowercase characters in the generated password.
- System must allow a logged in user to choose whether to include or not uppercase characters in the generated password.
- System must allow a logged in user to choose whether to begin or not with a letter in the generated password.
- System must allow a logged in user to choose whether to include or not symbols in the generated password.
- System must allow a logged in user to choose whether to include or not similar characters in the generated password.
- System must allow a logged in user to choose whether to include or not duplicate characters in the generated password.
- System must allow a logged in user to choose whether to include or not sequential characters in the generated password.

- System must allow a logged in user to edit a saved record in their password manager.
- System must allow a logged in user to delete a saved record in their password manager.
- System must allow a logged in user to add a saved record in their password manager.
- System must allow a logged in user to logout of their account.
- System must allow a logged in user to show the password of a record in their password in clear.
- System must securely encrypt the users usernames and passwords.
- System must securely encrypt the users records using the master password concept.

Navigation Map

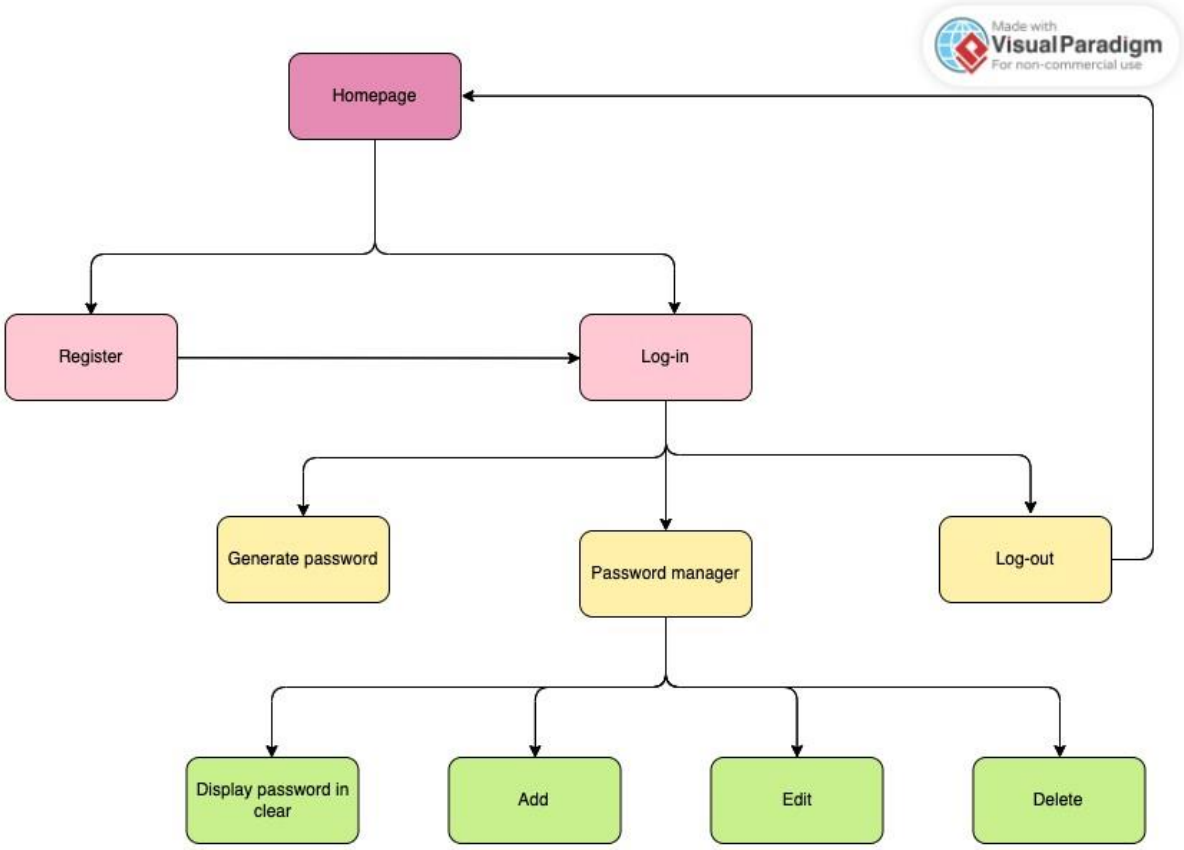

figure 1. Navigation map

Wireframe Design

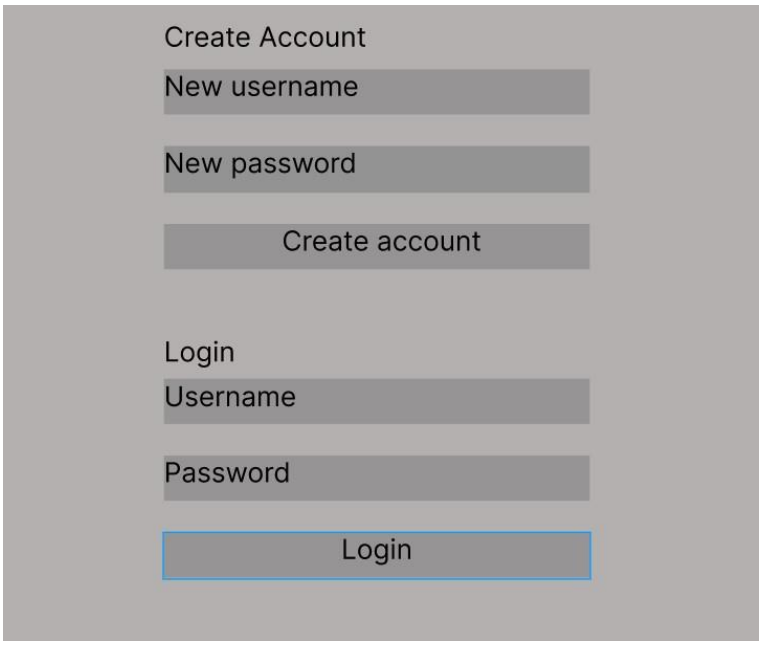

figure 2. Wireframe design n°1

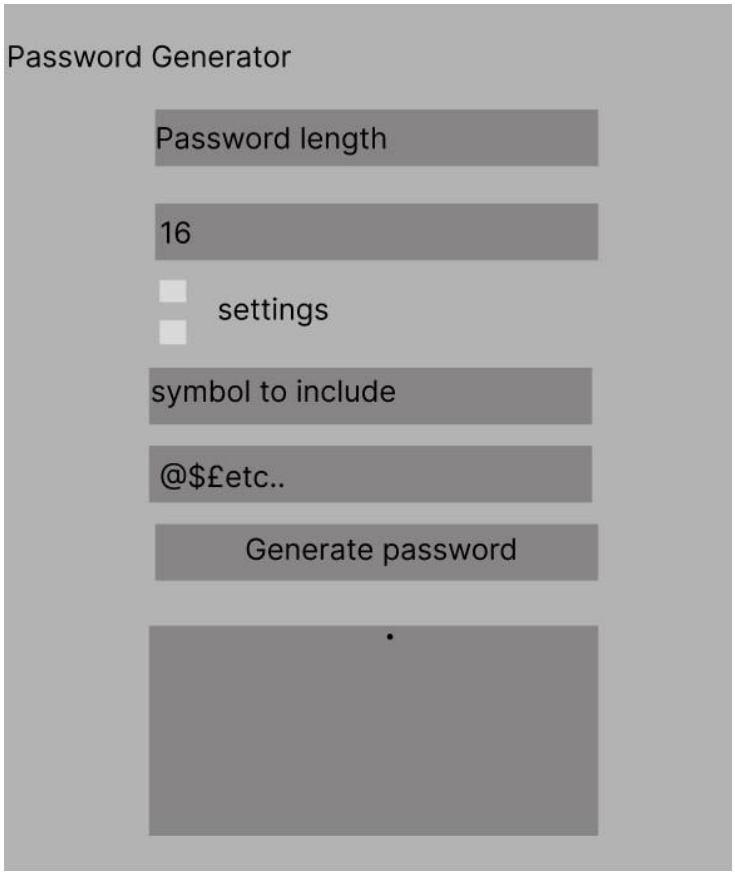

figure 3. Wirerfame design n°2

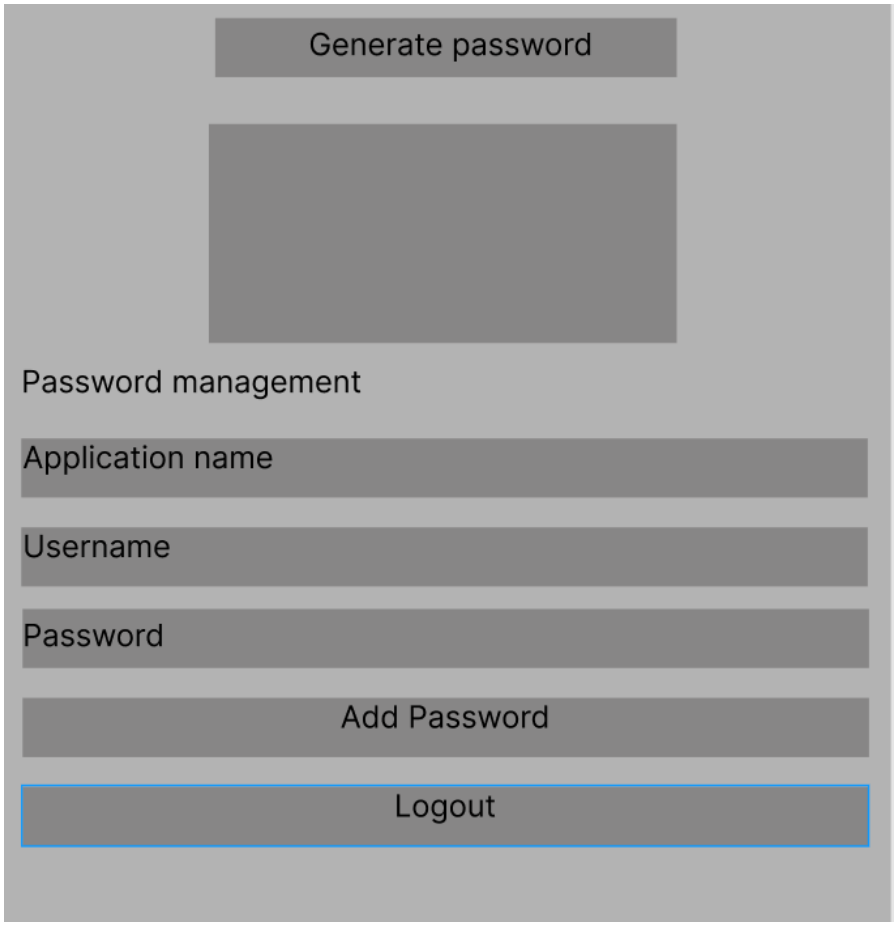

figure 4. Wireframe design n°3

User Interface Design

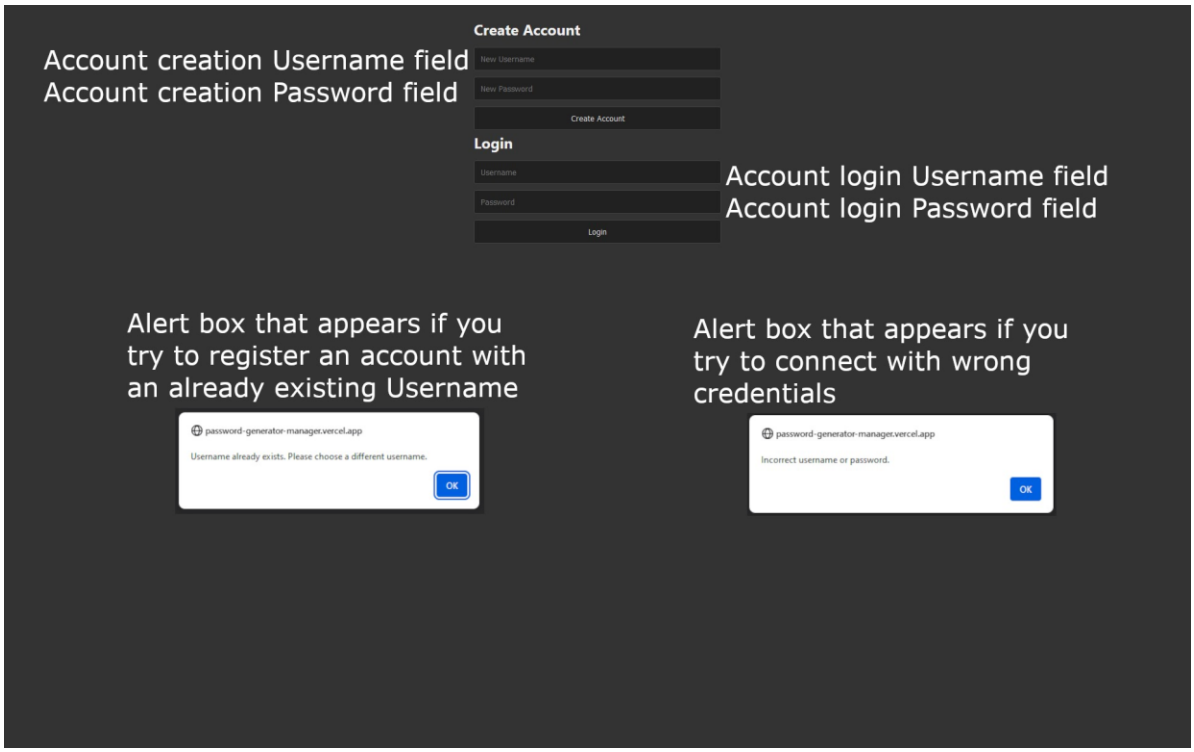

figure 5. User interface design n°1

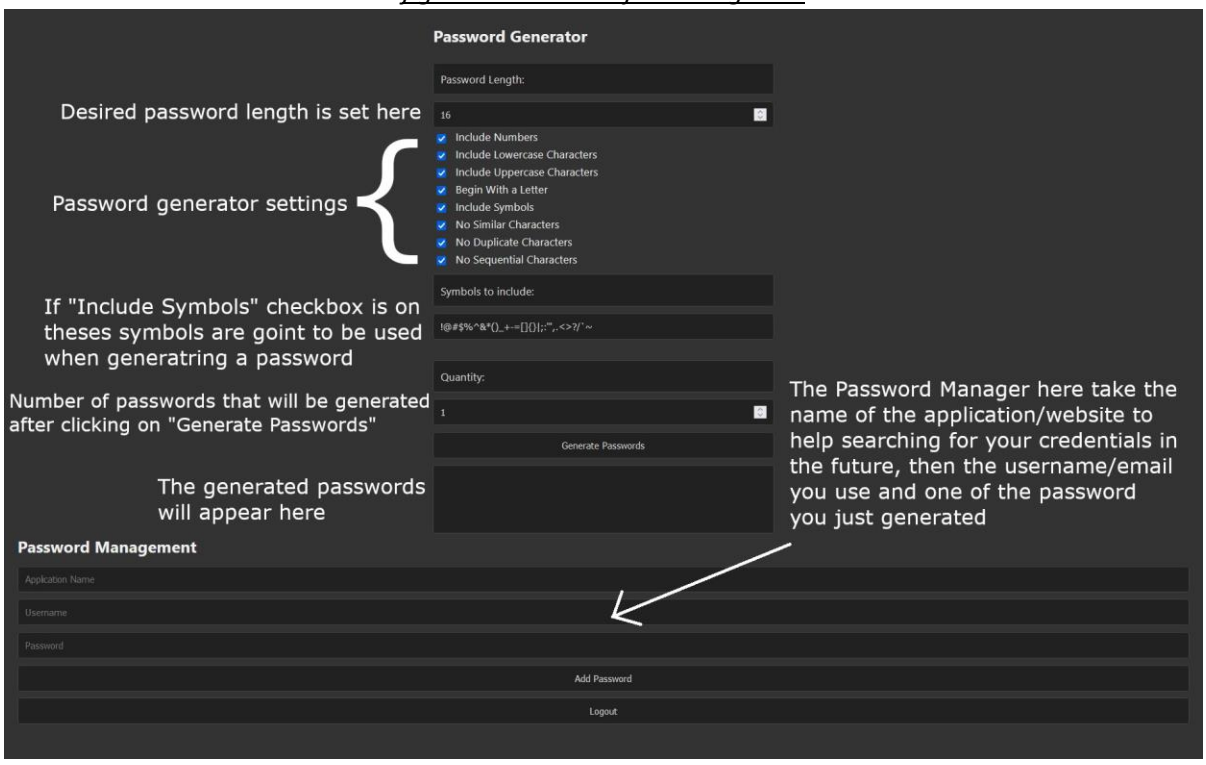

figure 6. User interface design n°2

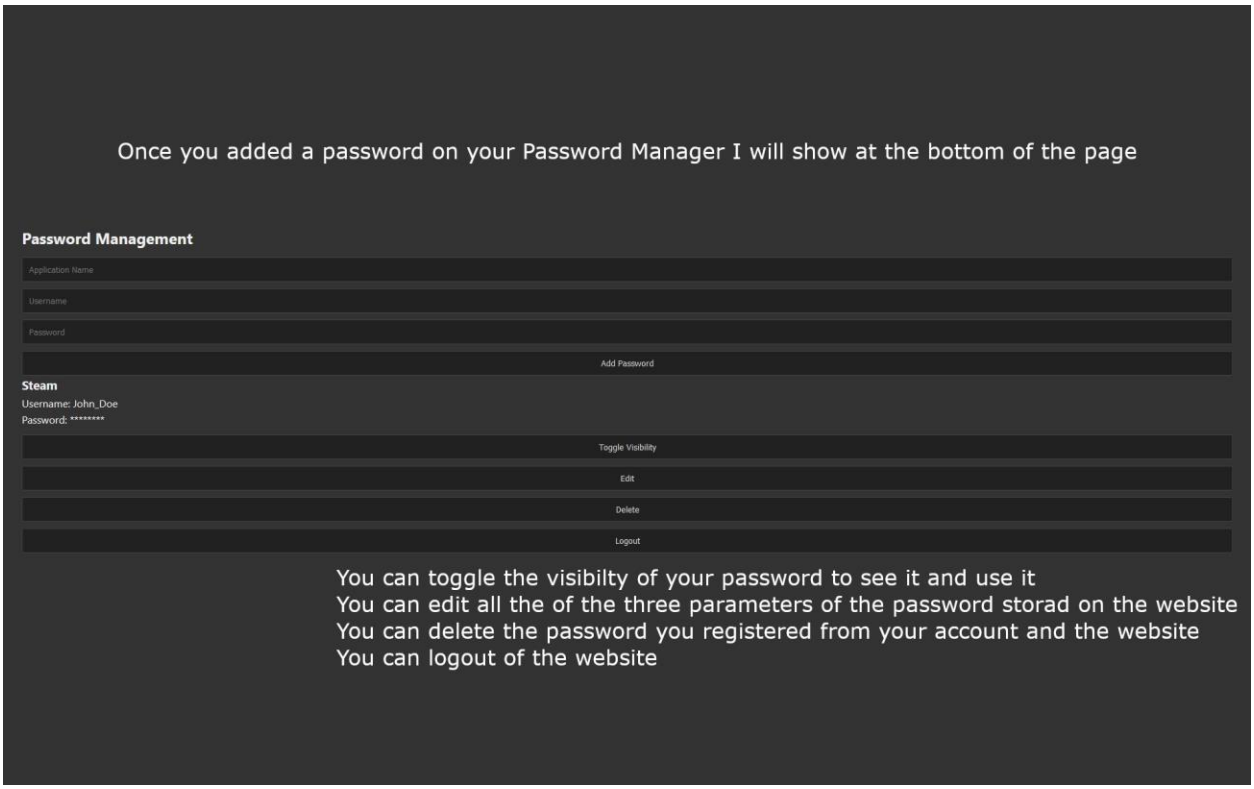

figure 7. User interface design n°3

Use Case Diagram

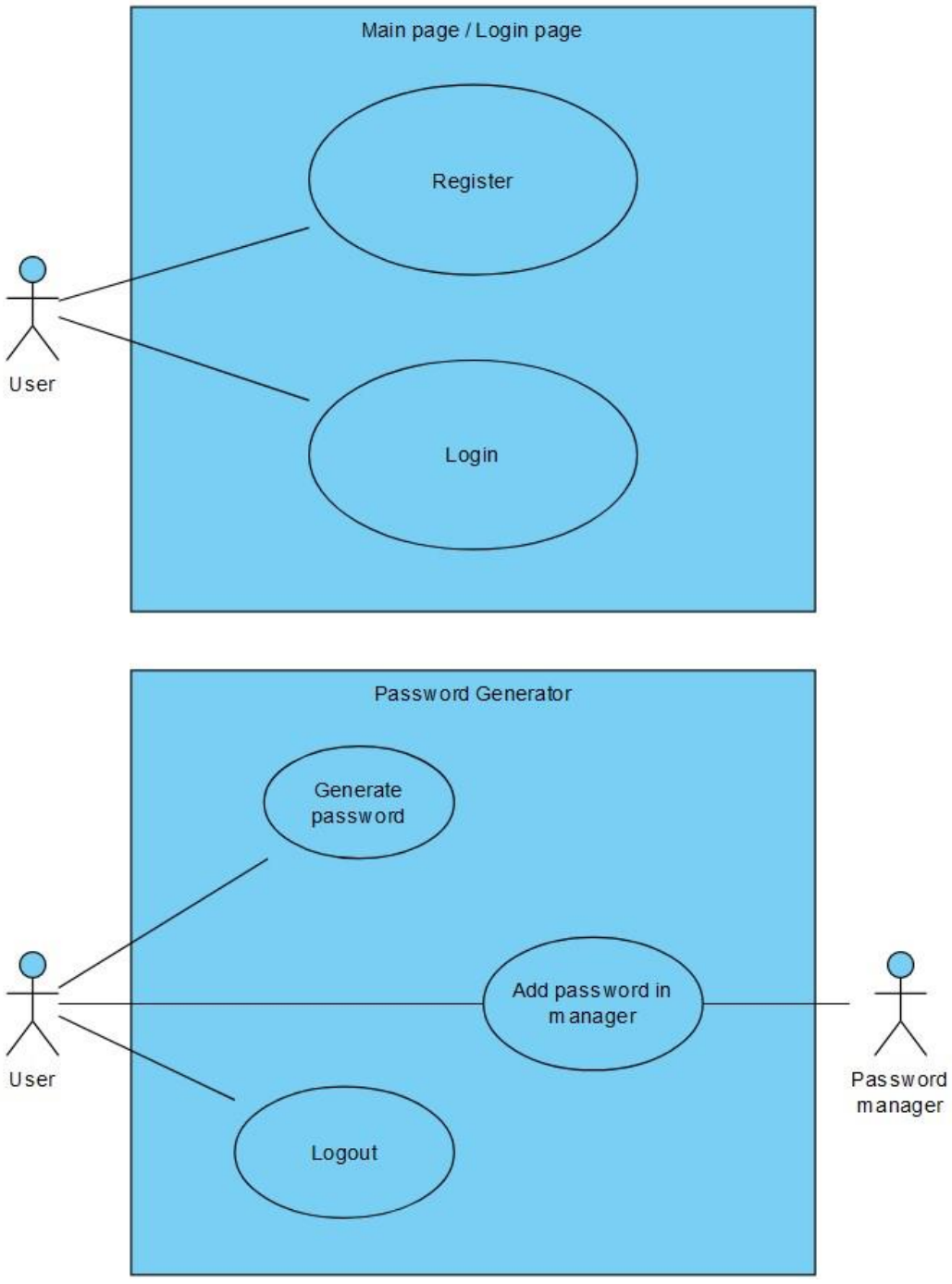

figure 8. Use Case Diagrams

Use Case Narratives

As the website has a lot of functionalities, we have chosen three use case narratives from the functionalities of the website to add in the report. One is the generation of a password use case narrative, the second is adding a new password to manager use case narrative, the third is the login use case narrative.

Name: Generate New Password
Actor(s): Logged-in User
Goal: Generate a new password.
Preconditions: User is logged into his account.

Main success scenario:
1- User doesn't want to change settings
2- User click on generate password
3- System generate and display passwords according to standard settings

Extension scenario A
1a- System displays standard settings
2a- User want to change setting(s)
3a- User un-check one or multiple boxes settings or/and modify one or multiples of the fields values settings
4a- User click on generate password
5a- System generate and display passwords according to modified settings

Postconditions: The password(s) are displayed.

Name: Add New Password to Manager
Actor(s): Logged-in user
Goal: Add a new password to the password manager of the user
Preconditions: The user is logged into his account

Main success scenario:
1- User want to add a new record in the password manager
2- System ask for the name of the application, for the username of the application, for the password of the application
3- User complete the application name field, the username field, and the password field.

4- User click on add password button
5- System display the new record in the password manager

Extension scenario A
4a- User click on add password button while on the three field or more is still empty
5a- System ask user to complete all the fields.
6a- Return to step 3

Extension scenario B
4b- User click on add password button while one or more of the three field has more than 50 characters
5b- System ask user to keep all fields under 50 characters.
6b- Return to step 3

Postconditions: The new password is successfully added to the password manager and visible.

Name: Login
Actor(s): User
Goal: Log into his user account
Preconditions: The user has a valid account in the system, and the user is on the login page.

Main success scenario:
1- The system prompts the user to enter their username and password.
2- The user enters the correct username and password.
3- The system check that the username and password match an existent account
4- The system log the user into his account and display the logged in page.

Extension scenario (a):
2a- The user enters an incorrect username or/and password.
3a- The system check the username and password and doesn't find any matching account.
4a- The system displays an error message indicating that the username or password is incorrect.
5a- Return to step 2

Postconditions: The user is logged in into his account.

Component Diagram

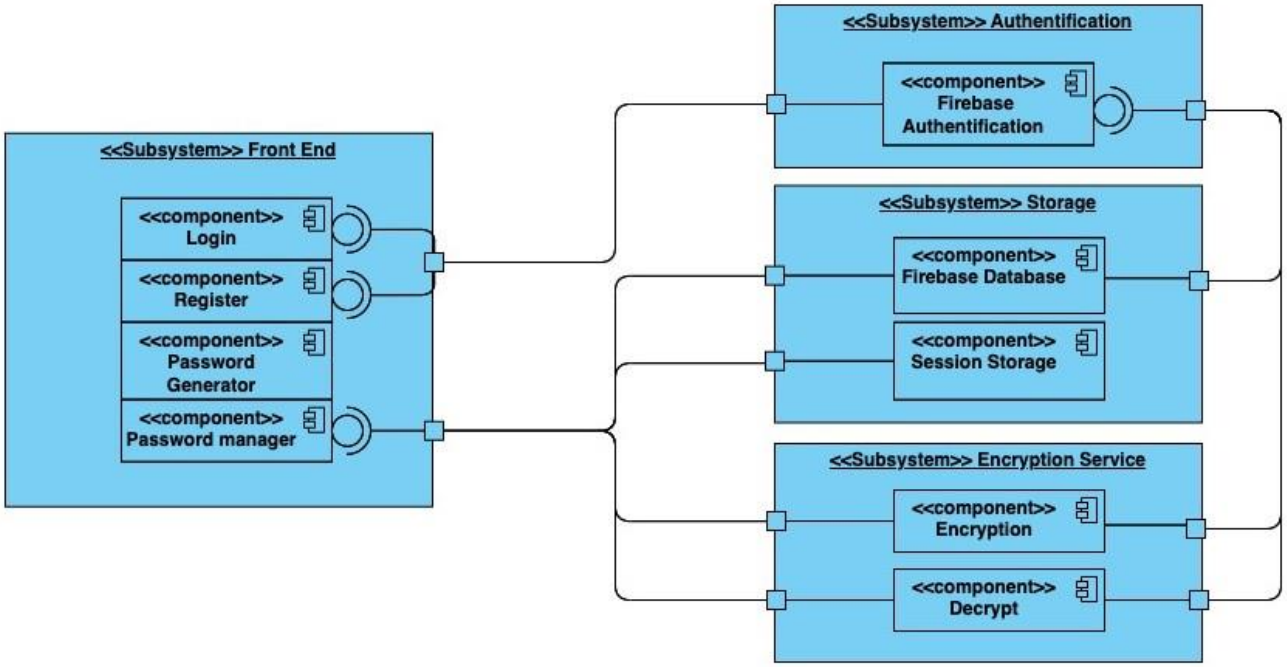

figure 9. Component diagram

Class Diagram

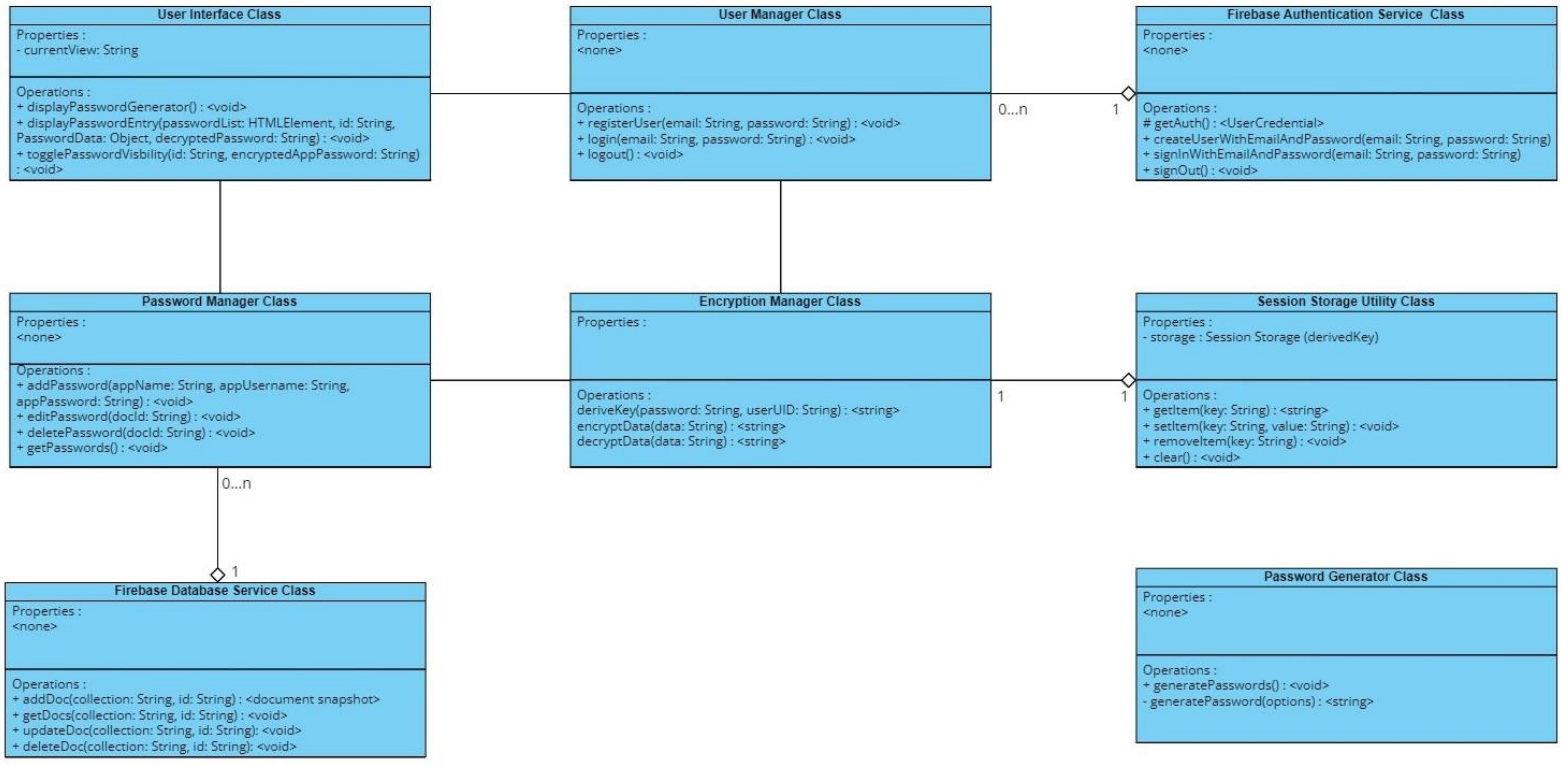

figure 10. Class Diagram

Sequence Diagram

As the website has a lot of functionalities, we have chosen four of the functionalities sequence diagram of the website to add in the report. One is the login sequence diagram, the second is the registering sequence diagram, the third is the generation of password sequence diagram and finally the fourth is adding a new record to the password manager sequence diagram.

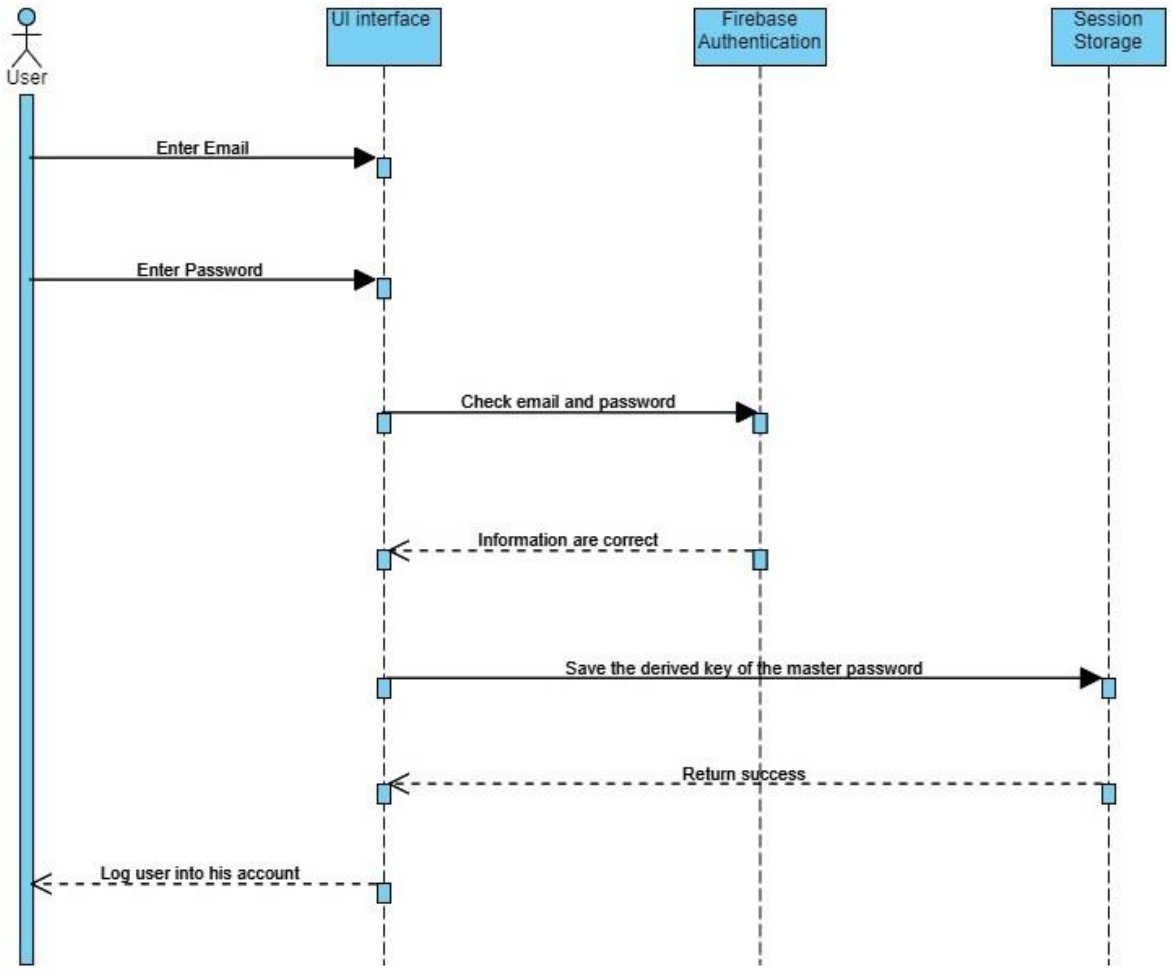

figure 11. Login sequence diagram

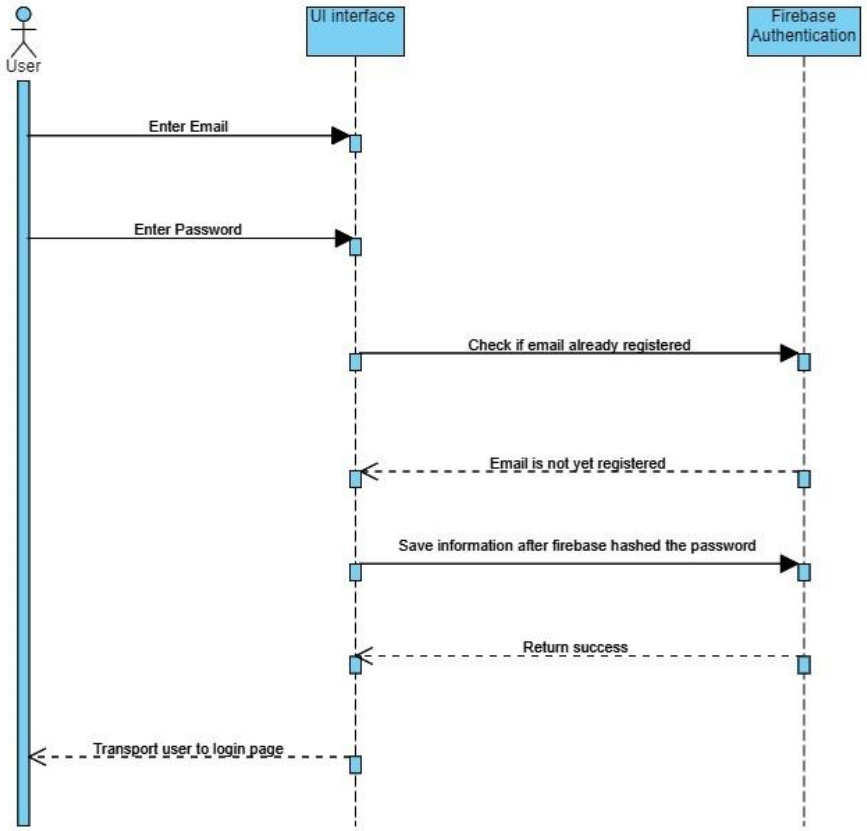

figure 12. Registering sequence diagram

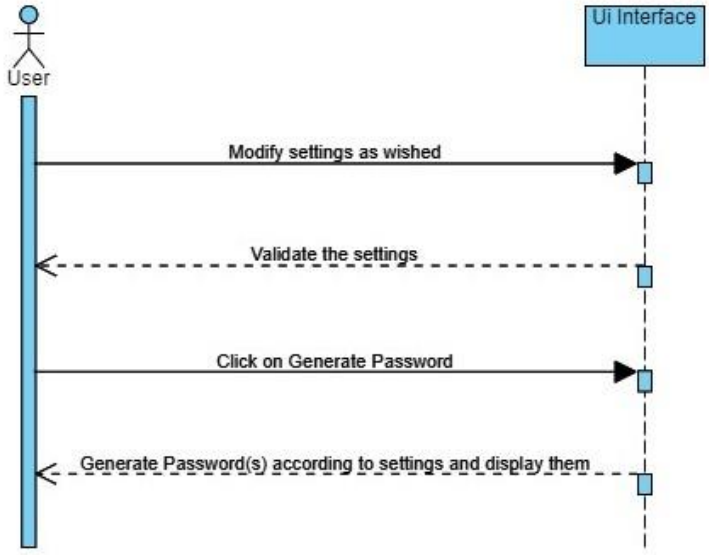

figure 13. Generate password(s) sequence diagram

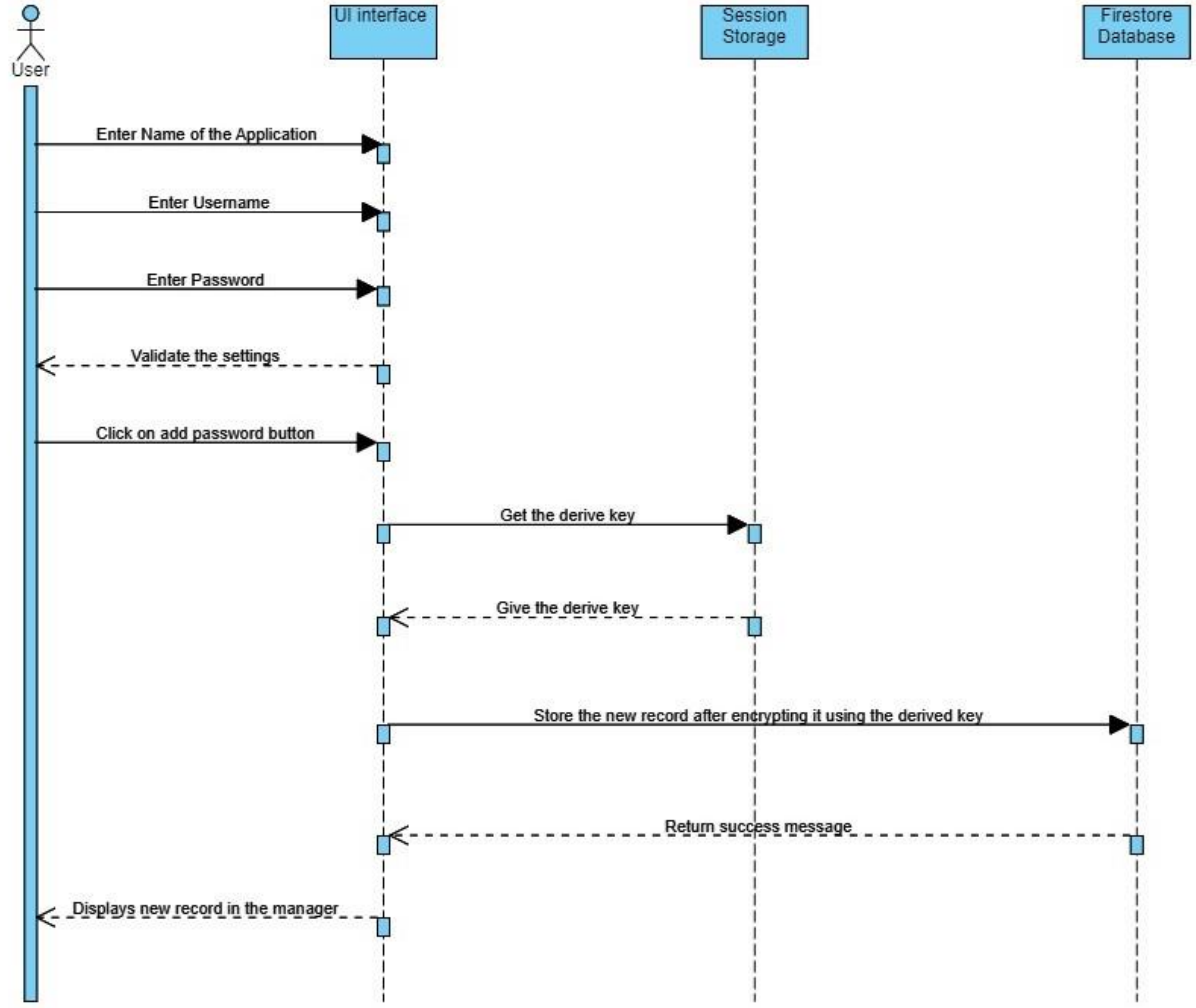

figure 14. adding new record in the password manager sequence diagram

Technology and Development tools

In the development of the password management website, the following technologies and development tools were strategically selected to construct a robust, secure, and user-centric application:

HTML (Hyper Text Markup Language): HTML is the cornerstone markup language utilized for structuring web content. It facilitates the creation of the foundational layout and elements within the software interface, including interactive forms and buttons crucial for user registration and password management functionalities.

CSS (Cascading Style Sheets): CSS was employed to define the aesthetic presentation of the HTML-structured content. Its implementation guarantees a visually consistent and engaging user interface, enhancing usability and ensuring adherence to contemporary web design standards.

JavaScript: JavaScript's role in the project is multifaceted, encompassing the creation of dynamic user interface components, event handling, and the manipulation of the Document Object Model (DOM). Its use is pivotal in enabling real-time interaction within the software, allowing immediate responsiveness to user actions without the need for page reloads.

CryptoJS: The CryptoJS library provides robust cryptographic functionalities, including secure encryption and hashing capabilities. Within the scope of this project, CryptoJS is instrumental in safeguarding sensitive user data, particularly by encrypting the stored passwords of the manager.

Web Storage API (Session Storage): The Web Storage API serve to store the client derived key and is derived from the master password, this key is only generated and stored client side, making sure it will never be stored on the server side, adding security against data breach. The session storage is furthermore purged upon session termination, enhancing security by preventing long-term key storage.

GitHub: We use the GitHub web application to store the files of our website in a private repository, this include the html file, the JavaScript file and the CSS file. It has enabled us to develop our project easily, giving us functionalities such as git add, git commit, and git push for easier organization and development.

Vercel: We use Vercel web application to deploy and host our website linking it directly with our GitHub repository for immediate deployment, it has also enable to establish a domain name that makes sense with our project name.

Server Database: We use the Firebase server to store our database, including the users main passwords but also the passwords in their password manager.

Server Authentication: We use the Firebase server to authenticate our users, this enable us to have an additional layer of security concerning the storage of the passwords since it uses a hashing encryption.

The rationale for the technology stack encompasses several facets:

-Universal Compatibility: The triad of HTML, CSS, and JavaScript ensures that the application is operable across all contemporary web browsers without necessitating additional software, enhancing the software's accessibility.

-End to End Encryption: Thanks to the client side encryption, it is made sure that the password manager passwords never leaves the users device to reach the server before being fully encrypted. In addition the derived key used to create a specific and unique user encryption never leaves the client side and is erased as soon as he log off. This make sure that the server only stores the encrypted version of the passwords and never receives plain text passwords or the derived key.

-Zero Knowledge Architecture : The server lack any means to decrypt the data it stores, as the server (Firebase) doesn't have access to the derived keys and as such will not be able to decrypt the user's passwords. Making sure the decryption can only occur client side, this also ensure that in case of a server breach, the attacker would only find encrypted data.

-Enhanced Security: The adoption of CryptoJS for cryptographic operations ensures the secure handling of sensitive data, a non-negotiable requirement for password management software. We can add the hashing encryption of Firebase for our database and finally the derived key encryption that operate only on the client side. All of this provide our users a sense of security which is justified and necessary in this field.

-Development Efficiency: The prevalent use and community support for HTML, CSS, and JavaScript accelerate the development process and problem resolution, given the abundant resources and documentation available. The GitHub technology enabled us to work as a team and build the website together, while Vercel enabled us to host and deploy continuously our website with immediate updating across the hosted website, and finally Firebase enable us to integrate a secure and scalable database easily.

-Scalability and Maintainability: The selected technologies afford scalability and ease of maintenance, ensuring that the application can be seamlessly enhanced or modified to meet evolving requirements.

For the security objective of our website information’s, it's separated in two different steps : the security of the user's account password, and the security of the stored passwords.
For the user's password we use firebase as our backend authentication system, enabling use to have an hashing encryption automatically done on the passwords, thus making sure that those information are only stored on the database in its hashed form, keeping it secure.
Concerning the encryption of the password manager passwords, we use the PBKDF2 with the cryptoJS encryption library to create a derived key from the master password with a particular salt for each user, the derived key is then used to encrypt the new stored password of the password manager of the user before it's stored on the firebase database.
Using this method we make sure that even with a leaked firebase, the passwords stored of the users will still be fully encrypted, keeping it fully safe.

Conclusion

The project was very challenging and rewarding for all of us, it was the occasion to go beyond our limit and show us that we could achieve this time of project we were to put the time for it. Thanks to this project we have been able to apply and understand the different teachings and theory we learned in class, their difficulties, effectiveness, but also their strengths and weaknesses. The project was also an occasion for us to aboard new coding language or technical tools and put them to practice. It also enabled us to improve our working methods, work on our efficiency, and learn how to manage the work together in such a big team. This project was without a doubt a real challenge, whether it was in time, in technical difficulties (as a lot of it was totally new to us), at applicating theory, using software process and staying rigorous and others. We were able to create a website that made us proud, proud of the progress and effort that was put into it, but also a bit sad, sad because we wanted to do more, to implement more and better. We certainly took a liking to the project and it's future.

As an opening we would firstly want to say that we would love to push this project further and continue to implement every idea that was detailed in the last part, whether it's the evolvement of the UI in something more interactive and beautiful, or in the security of our web application, but also in it's different functionalities and singular features we wanted to create. As for the future it's important to note that our journey does not end with the creation of a more secure password system; it is merely a stepping stone in the constant pursuit of fortifying our digital lives. The future is full of promises concerning advanced technologies, between AGI technology and quantum computing, we could easily see our traditional passwords become obsolete. However, until that day dawns, our vigilance and commitment to evolving our security practices remain our strongest shield against the consistent cyber-threats as users. To close this report, let us not forget that in the realm of cybersecurity, complacency is the adversary we must constantly outwit. The password, in all its simplicity and complexity, is not just a guardian of our digital identities but a reminder of our enduring responsibility in the cyber world.