iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 12 Issue: XI Month of publication: November 2024

DOI: https://doi.org/10.22214/ijraset.2024.65164

Notes and Password Manager

Jasleen Kaur[1], Kirti[2], Dharmesh Gidwani[3]
Department of Computer Science and Engineering, Apex Institute of Technology, Chandigarh University, Mohali, Punjab, India

Abstract: In today's digital age, keeping passwords safely and effectively is a continual challenge because there are so many credentials to maintain for several platforms. While retaining a high degree of security, the focused on users Password Manager system demonstrated in this study makes it simpler to save, retrieve, and handle passwords. Through the integration of a front-end React-based interface with a Node.js and Express backend, the proposed solution allows users to securely keep and access their login credentials. The system uses token-based authentication to ensure that each user's information is kept secure and that only they may access it. Features like encoding, edit/delete options, copy-to-clipboard functionality, and password visibility switching further enhance the user experience. The backend links to a database called MongoDB for persistent credential storage, and authentication middleware secures all actions (save, modify, and delete). Future improvements may include cloud synchronization, multiple-factor authentication (MFA), and AI-driven password generating tools to increase accessibility across devices. In the increasingly complex digital environment, this password manager aims to promote secure password management, reduce user cognitive load, and enhance security.

I. INTRODUCTION

People use many different websites in the modern digital era, and each one requires a particular set of security credentials. Managing a large number of passwords may be challenging and often leads to poor password practices, including password reuse, which increases a system's susceptibility to attacks. A secure and user-friendly tool called Password Manager fixes these problems by storing users' encrypted login information so they just need to recall a master password or securely authenticate. This project presents a Password Manager system built with the the MERN (MongoDB, Express, React, and Node.js) stack, with a focus on secure storage, ease of use, and exclusive access for users via authentication tokens. The primary objective of this system is to provide consumers with a seamless experience while prioritizing security when saving, retrieving, and managing their credentials.

Token-based authentication (JWT) to ensure that only those with authorization may access stored data, visibility toggling, password encryption, and CRUD (Create, Read, Update, Delete) activities on saved credentials are some of the features. Using the MERN stack, this password manager combines speed and adaptability to provide a responsive and easily scaled solution.

With future plans to incorporate biometric authentication, multi-device synchronization, and AI-powered password creation, the Password Manager aims to address the evolving security concerns of today's digital world.

A. *Hardware Specifications*

1) CPU: 2.0 GHz or faster AMD CPU or Intel Core i5 processor
2) RAM: 8 GB at the very least (16 GB is advised for seamless development). 250 GB of SSD or HDD storage (for database, dependencies, and code)
3) Network: Reliable internet access for server-client interactions and library downloads
4) System compatibility: Ubuntu 20.04+, macOS 10.15+, and Windows 10/11

B. *Software Specifications*

1) Frontend: To create the user interface, utilize React.js
2) Tailwind CSS: For contemporary and responsive design React Router
3) DOM: To manage protected routes and navigation
4) React Toastify: To deliver toast alerts for comments

C. *Backend*

1) Express.js: Used to create RESTful APIs:
2) Node.js: Used to create the backend server.

3) MongoDB: To safely save user information in a NoSQL database The mongoose To work with MongoDB and model data Protection
4) Verification: JSON Web Token (JWT): To control user sessions and prevent unwanted access Passwords are encrypted using Bcrypt.js before being stored in a database.
5) Axios: to make it easier for the client and backend to communicate via API.

D. Problem Overview

In today's linked world, most people have several online accounts that require complex, strong passwords. Keeping track of these passwords by hand has several issues:

1) Ineffective Password Procedures

Password Reuse: Many people use the same passwords for many accounts in order to save their energy and time while memorizing multiple login credentials. On the other hand, credential stuffing attacks expose several accounts at risk by breaching a single compromised account.

Simple Passwords: Users usually create weak passwords (such "123456" or "password") to make them simpler to remember. Dictionary or brute-force assaults can easily break these passwords. Passwords that are still the same: Individuals are more susceptible to long-term vulnerability in the case of a data breach because they seldom update their passwords.

2) Weaknesses in Security

Insecure Storage Practices: If users keep their passwords in text files, spreadsheets, or notebooks, they may unintentionally or accidently reveal them.

Phishing attacks: Without an appropriate password manager, users could inadvertently enter their login information into phony websites, making them vulnerable to phishing attacks.

3) Complexity of Access and User Dissatisfaction

Too Many Credentials to Maintain: When a person has accounts on several of websites, it might be difficult for them to remember or keep track of all their passwords.

Frequent Password Resets: Because people forget their passwords so easily, it is necessary for them to be changed on a regular basis. This process might be time-consuming and tiresome.

4) Current Solutions and Their Drawbacks

Password managers with a membership model: Some password managers discourage a wider audience from using them by charging for all services, including cross-platform synchronization.

Usability Problems: Many password management applications are problematic for non-technical users since they are hard to grasp.

5) Risks of Unauthorized Access and Data Privacy

Multi-User Devices: Users who share computers with relatives or colleagues face the risk of having their saved credentials viewed by unauthorized persons if proper session administration and user isolation procedures are not in place.

Inadequate Session Management: Without robust token-based authentication, data leakage might happen if cached credentials remain accessible after a session timeout or logout.

E. Objectives

Safe Data Storage: Store all Bcrypt-encrypted passwords in MongoDB to ensure data security.
User Authentication: Use JWT-based authentication to ensure that users only has access to their personal data.
Create separated user sessions to prevent unapproved access to other members' login information.
CRUD Functions: For adding, editing, viewing, and removing credentials, provide a seamless user experience.
Make sure that passwords are shown in a toggleable manner and that copy-to-clipboard functionality is enabled.
User-Friendly Interface: Utilize Tailwind CSS for adaptive layouts and React.js to build an intuitive front end.
Provide toast alerts to users to inform them of accomplishments or errors.
Real-Time API Integration: Front-end and the back end interactions may be seamlessly managed using Axios.
To get, edit, and delete user data, utilize Express routes.

II. LITERATURE REVIEW

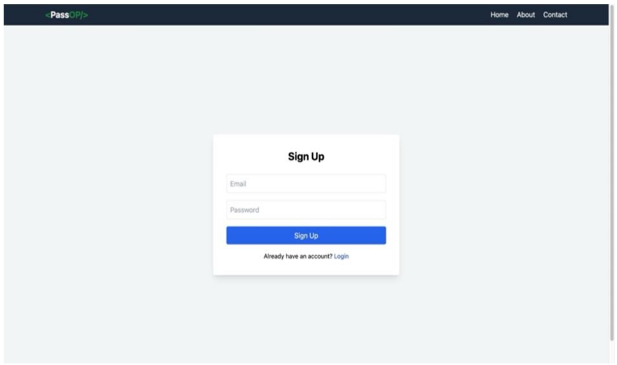
1) Password managers have become essential tools in today's digital world to address the growing problems of compromised credentials and password reuse. Preliminary study indicates that users often struggle to properly manage several accounts, leading to poor practices such as reusing passwords. To combat this, password managers come with features including password generation, secure storage, and auto-fill. However, research shows serious problems with usability and adoption—many users are reluctant to use password management programs due to concerns about complexity and trust. Modern password managers employ encryption techniques like AES-256 and multi-factor authentication (MFA) to secure data, but they are still vulnerable to master password leaks, clipboard attacks, and phishing.
2) Comparative Usability and Security Research: Johnson et al. (2017) conducted a research that evaluated password managers based on their encryption and other security aspects. One important lesson learnt was the need to combine security and usability. Adoption by Users: Smith and Brown (2018) investigated how users viewed password managers and discovered that although users appreciated its security capabilities, non-users were concerned about trust and usability. Raising awareness of the advantages might lead to a rise in adoption. Two-factor authorization (2FA): Chen et al. (2019) shown that adding 2FA significantly increases password manager security by adding an extra layer of authentication.
3) Security: KeePass, Password Safe, Bitwarden, and other password managers' encryption techniques were evaluated. KeePass provides a great deal of versatility by supporting several encryption schemes like AES and Twofish, whereas Bitwarden uses AES-256 encryption and relies on servers in the cloud for security. Password Safe, which uses Twofish for database encryption, now supports Yubikey two-factor authentication. Usability: The research emphasized the contrasts in user interfaces between Bitwarden and Password Safe, which have simpler interfaces, and KeePass, which offers more flexibility and plugins from third parties but has a more sophisticated UI. KeePass was commended for its extensive feature set, which includes customizable entries and password generation, making it more appropriate for more seasoned users.
4) Chaitanya Rahalkar and Dhaval Gujar's "A Secure Password Manager" (2019): The weaknesses of conventional password storage systems and data breaches are covered in this study. It suggests an offline password management that uses a state master password technique to create safe passwords instantly. By ensuring that passwords are never kept anywhere, this method lowers the possibility of data exposure. The suggested approach incorporates PGP-inspired file encryption features and maximizes attack resistance by using Argon2d for key generation. The usage of conventional password managers, like 1Password and LastPass, which normally store credentials in encrypted vaults and might cause security problems if the vaults are stolen, is one example of the material studied.
5) Sonia Chiasson et al.'s "A Usability Study and Critique of Two Password Managers" (2006): PwdHash and Passwords Multiplier are two password management systems whose usability and security consequences are assessed in this study. It draws attention to important usability issues that have a direct impact on security, such as users' inability to accurately visualize how the tools operate. Common misconceptions regarding how password managers create and secure passwords were among the inconsistencies the investigation found between the original authors' usability claims and observed user behavior. Previous studies on cognitive load and security of passwords are included in the literature listed.
6) The document titled "Introduction to MERN Stack & Comparison with Previous Technologies," published in the *European Chemical Bulletin* in June 2023, offers a literature review on the MERN stack, comparing it to prior technologies like HTML, CSS, SQL, and NoSQL. It covers the structure, advantages, and technical differences of MERN components (MongoDB, Express.js, React, and Node.js), highlighting why the stack is popular for modern web applications.The authors argue that the MERN stack allows developers to use JavaScript across both client and server sides, creating a cohesive development experience. The paper also addresses the advantages of MongoDB's flexibility for unstructured data, the lightweight nature of Express.js for backend management, the efficient, component-based architecture of React, and Node.js's ability to make JavaScript asynchronous, allowing for high scalability and real-time applications
7) Optimization strategies for e-commerce systems developed with the MERN stack are covered in the paper "Performance Optimization using MERN Stack on Web Application," which was published in the International Journal of Engineering Research & Technology in June 2021. The components of the stack—Express.js, React, Node.js, and MongoDB—contribute to an effective, scalable design that is appropriate for high-performance online applications, as this paper highlights. The authors review the literature on the benefits of the MERN stack for e-commerce, such as the scalable, document-oriented data storage of MongoDB, the streamlined middleware support of Express.js for creating APIs, the virtual DOM of React for effective rendering, and the event-driven, asynchronous environment of Node.js, which encourages a non-blocking request model that is crucial for managing high-traffic web applications.

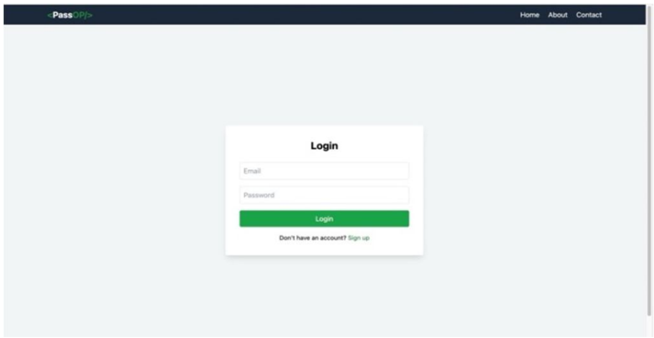
8) A thorough analysis of several authentication techniques, such as password-based, smart card, biometric, and digital certificate-based processes, may be found in the paper "A Review of Authentication Methods," which was published in the International Journal of Scientific & Technology Research in November 2016. This essay highlights each method's application in security systems by examining its advantages, disadvantages, and solutions. The authors divide authentication techniques into three categories: biometrics, which are based on innate characteristics, smart cards, which are based on ownership, and knowledge-based (passwords). Despite being widely used, password authentication is vulnerable to social engineering and brute-force attacks, thus using complicated passwords with high unpredictability is crucial. Smart cards, which use PINs to increase security, are vulnerable to phishing and loss.

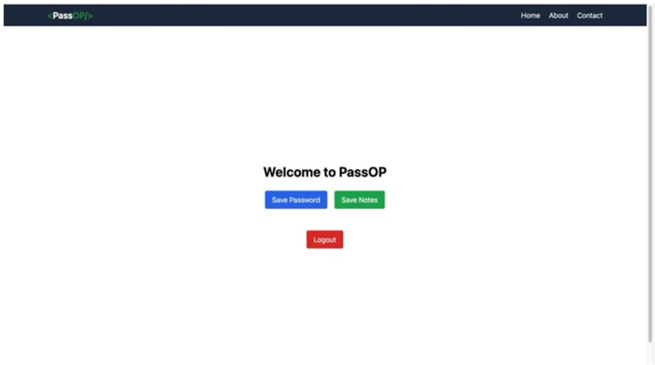
9) Christina Braz and Jean-Marc Robert's 2006 conference paper, "Security and Usability: The Case of the User Authentication Methods," discusses the difficulty of striking a balance between security and usability in user authentication techniques. The authors examine how increased security frequently comes at the expense of usability, resulting in a less-than-ideal user experience. They contend that designing authentication systems with human considerations in mind is crucial to producing more efficient security features that users can easily understand and control. The study examines usability concerns unique to public key infrastructure (PKI), passwords, PINs, fingerprints, and other authentication techniques. For example, password systems are criticized for failing to adhere to certain usability guidelines, including making users memorize lengthy sequences and offering little to no feedback.

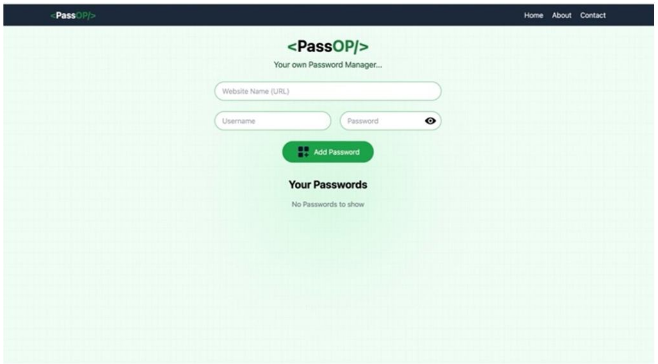
10) The article "A Study for an Ideal Password Management System," which was published in January 2022 in the International Journal for Research in Applied Science & Engineering Technology, explores several password management strategies and lists the qualities of the perfect password manager. Shivam K. Shinde and Mohit V. Deshpande, the authors, concentrate on examining current password creation, storage, and retrieval systems in order to pinpoint factors that can successfully strike a balance between security and usability. Because the ordinary user must manage a large number of online accounts, the study addresses the shortcomings of conventional password methods and the growing demand for trustworthy password managers. By safely creating, storing, and autofilling passwords, password managers reduce the mental strain of memorizing complicated, one-of-a-kind passwords.

III. All RESULTS AND OUTPUTS









REFERENCES

[1] Luevanos, Carlos & Elizarraras, John & Hirschi, Khai & Yeh, Jyh-haw. (2017). Analysis on the Security and Use of Password Managers. 17-24. 10.1109/PDCAT.2017.00013.

[2] Patel, Neel & Kalra, Aryan. (2023). "SECURE PASSWORD MANAGER".

[3] Master, A. (2023) Password managers: Secure passwords the easy way [Preprint]. doi:10.5703/1288284317618.

[4] Chaitanya Rahalkar, Dhaval Gujar . A Secure Password Manager. International Journal of Computer Applications. 178, 44 (Aug 2019), 5-9. DOI=10.5120/ijca2019919323

[5] Chiasson, Sonia & Oorschot, P & Biddle, Robert. (2006). A usability study and critique of two password managers. 15th USENIX Security Symposium.

[6] Kadam, Prof & Goplani, Akhil & Mattoo, Shubit & Gupta, Shashank & Amrutkar, Darshan & Dhanke, Jyoti & Kadam, Yogesh. (2023). Introduction to MERN Stack & Comparison with Previous Technologies. European Chemical Bulletin. 12. 14382-14386. 10.48047/ecb/2023.12.si4.1300.

[7] Master, A. (2023) Password managers: Secure passwords the easy way [Preprint]. doi:10.5703/1288284317618.

[8] Farik, Mohammed & Lal, Nilesh & Prasad, Shalendra. (2016). A Review Of Authentication Methods. International Journal of Scientific & Technology Research. 5. 246-249.

[9] Braz, Christina & Robert, Jean-Marc. (2006). Security and usability: the case of the user authentication methods. 199-203. 10.1145/1132736.1132768.

[10] Shinde, S.K. and Deshpande, M.V. A study for an ideal password management system. Available at: https://www.ijraset.com/research-paper/an-ideal-password-management-system

[11] Ambarish Karole, Nitesh Saxena, and Nicolas Christin. A Comparative Usability Evaluation of Traditional Password Managers. In Kyung-Hyune Rhee and DaeHun Nyang, editors, Information Security and Cryptology - ICISC 2010, Lecture Notes in Computer Science, pages 233–251. Springer Berlin Heidelberg, 2011.

[12] Zhiwei Li, Warren He, Devdatta Akhawe, and Dawn Song. The Emperor's New Password Manager: Security Analysis of Web-based Password Managers. pages 465–479, 2014.