S-Secure

Project Overview

Problem Statement:

In today's technology-driven world, the security of personal information is of utmost importance. However, many people, due to their busy schedules, tend to create weak passwords, often using easily guessable information like their names or dates of birth. This poses a significant security risk, making it essential to address the issue of creating strong and secure passwords.

Proposed Solution:

To enhance security, I developed a website that generates strong, random passwords based on user preferences. This tool simplifies the process of creating secure passwords, allowing users to generate passwords with varying patterns and complexities according to their convenience.

Concept Flow:

1. **User Registration:**
    - Users begin by signing up on the website.
2. **Data Storage:**
    - User data is securely stored in Firebase Realtime Database.
3. **Password Generation:**
    - After signing up, users are redirected to the password generator page.
4. **Password Algorithm:**
    - Passwords are generated using a combination of uppercase letters, lowercase letters, symbols, and numbers.
5. **Password Usage:**
    - Users can easily copy and use the generated password.

Technology & Tools Used:

1. **Firebase:** Realtime Database for storing user data.
2. **GitHub:** Used for website hosting.
3. **Git:** Version control system to manage project changes.
4. **JavaScript, HTML, CSS:** Core technologies for developing the website.
5. **Visual Studio Code:** Integrated development environment (IDE) used for coding.

Project Description:

1. **Random Character Generation:**
    - **Lowercase Letters:** The getRandomLower function generates random lowercase letters using the ASCII code for 'a' (97) and a random number between 0 and 25.

- **Uppercase Letters:** The getRandomUpper function generates random uppercase letters starting from the ASCII code for 'A' (65).
- **Numbers:** The getRandomNumber function uses window.crypto.getRandomValues to generate a secure random number between 0 and 9, converting it to its corresponding ASCII character.
- **Symbols:** The getRandomSymbol function selects a random symbol from a predefined set of special characters.

2. **Password Construction:**
    - The generatePassword function combines selected character types (lowercase, uppercase, numbers, symbols) to create a password of the desired length. The process involves iterating through each selected type and appending random characters to construct the final password.
3. **Secure Randomness:**
    - The secureMathRandom function uses window.crypto.getRandomValues to ensure that the random numbers generated are secure and less predictable compared to using Math.random() alone.
4. **Dynamic UI Updates:**
    - The UI dynamically updates based on user interaction. Sliders control the password length, and checkboxes allow users to choose which character types to include. Real-time feedback, such as displaying the current password length and adjusting the slider's background, enhances user interactivity.
5. **Preventing Errors:**
    - The algorithm ensures that at least one checkbox is always selected. If only one type is selected, the checkbox is disabled to prevent the user from deselecting it and creating an invalid password.
6. **Clipboard Functionality:**
    - The generated password can be easily copied to the clipboard by clicking a button. This is implemented by creating a temporary text area element, selecting the password text, copying it, and then removing the text area.

Conclusion:

The website provides a simple yet effective solution for generating strong passwords, addressing the critical need for secure personal information in the digital age. By leveraging modern web technologies and ensuring a user-friendly interface, the project aims to make password generation both accessible and secure for all users.

Website URL: **https://sasidharan1415.github.io/S-Secure/**

Database URL: Carrom - Authentication - Users - Firebase console (google.com)

Source Code URL: sasidharan1415/S-Secure (github.com)