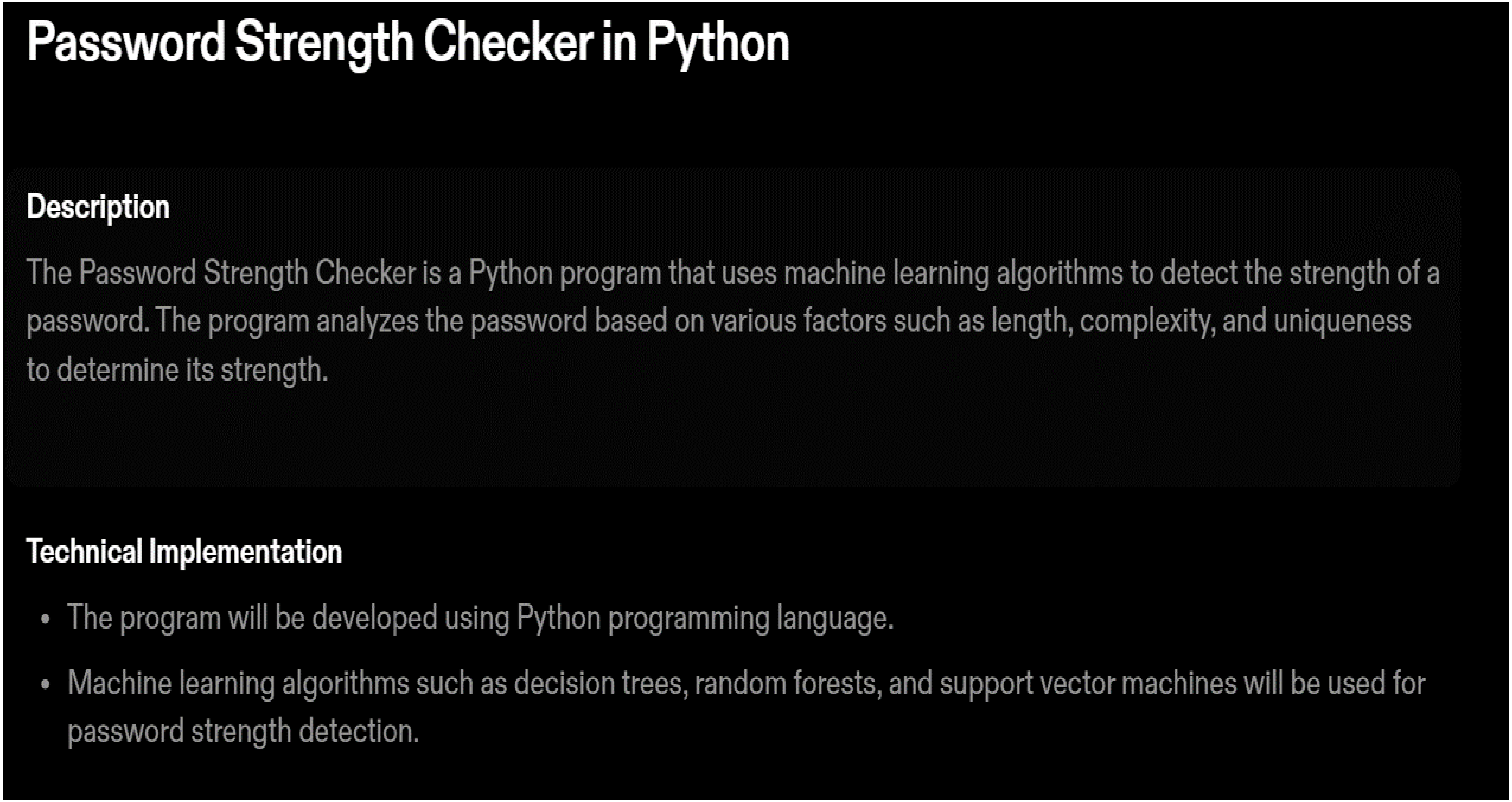
PASSWORD STRENGTH CHECKER

SUNMATHI R

7376222IT267

INFORMATION TECHNOLOGY

Building a Password Strength Checker in Python



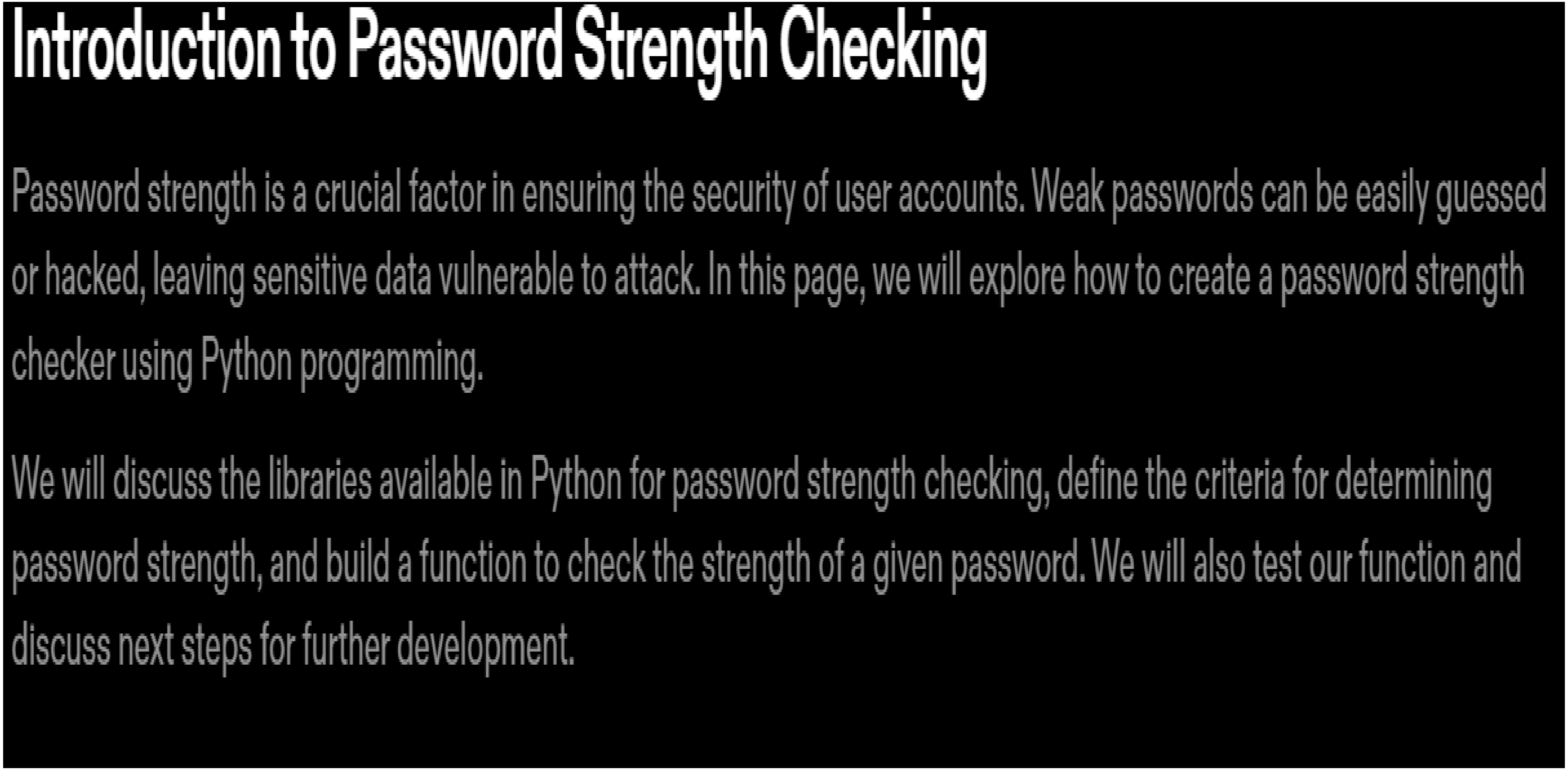
Password Strength Checker in Python

Description

The Password Strength Checker is a Python program that uses machine learning algorithms to detect the strength of a password. The program analyzes the password based on various factors such as length, complexity, and uniqueness to determine its strength.

Technical Implementation

- The program will be developed using Python programming language.
- Machine learning algorithms such as decision trees, random forests, and support vector machines will be used for password strength detection.



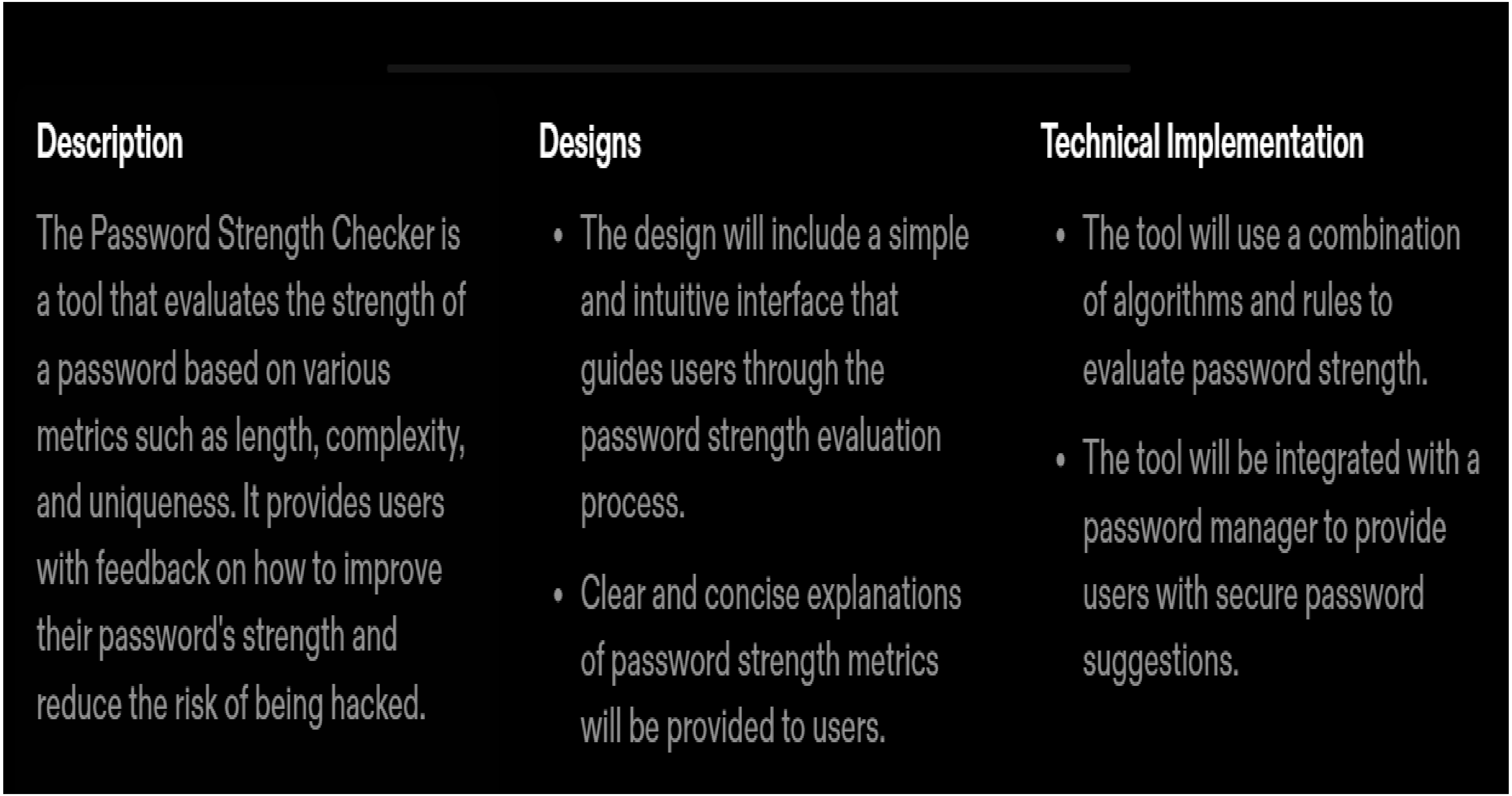
Introduction to Password Strength Checking

Password strength is a crucial factor in ensuring the security of user accounts. Weak passwords can be easily guessed or hacked, leaving sensitive data vulnerable to attack. In this page, we will explore how to create a password strength checker using Python programming.

We will discuss the libraries available in Python for password strength checking, define the criteria for determining password strength, and build a function to check the strength of a given password. We will also test our function and discuss next steps for further development.

Common Password Mistakes

Using easily guessable passwords such as 'password' or '123456', reusing the same password across multiple accounts, and not changing passwords frequently are some of the common mistakes people make when it comes to password security.



Description

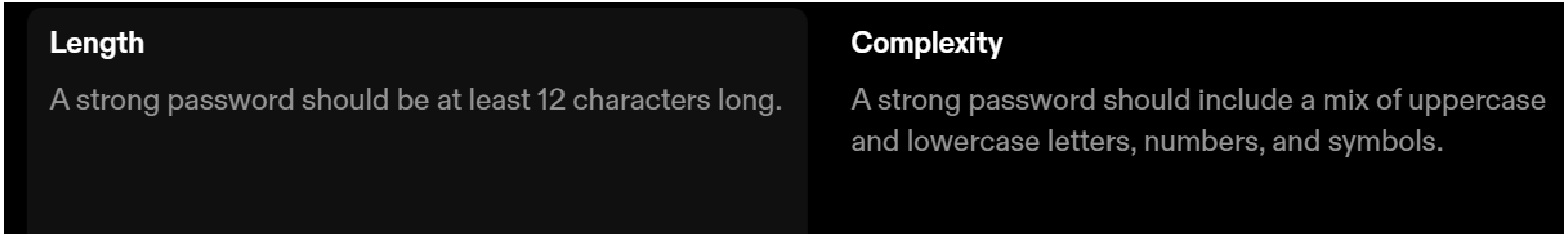
The Password Strength Checker is a tool that evaluates the strength of a password based on various metrics such as length, complexity, and uniqueness. It provides users with feedback on how to improve their password's strength and reduce the risk of being hacked.

Designs

- The design will include a simple and intuitive interface that guides users through the password strength evaluation process.
- Clear and concise explanations of password strength metrics will be provided to users.

Technical Implementation

- The tool will use a combination of algorithms and rules to evaluate password strength.
- The tool will be integrated with a password manager to provide users with secure password suggestions.

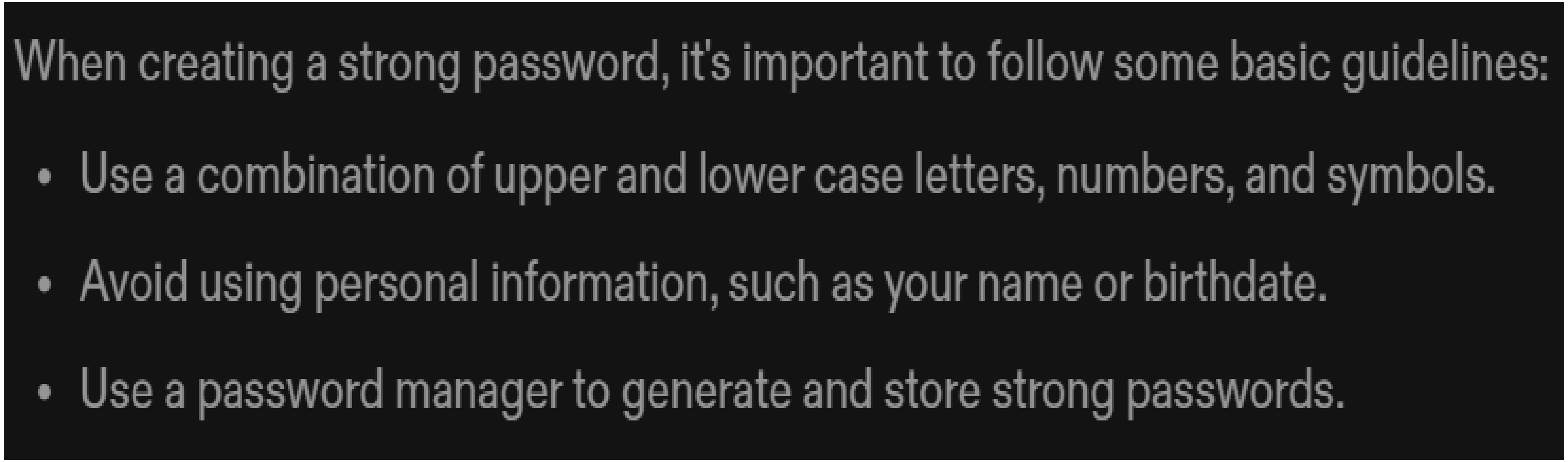


Length

A strong password should be at least 12 characters long.

Complexity

A strong password should include a mix of uppercase and lowercase letters, numbers, and symbols.

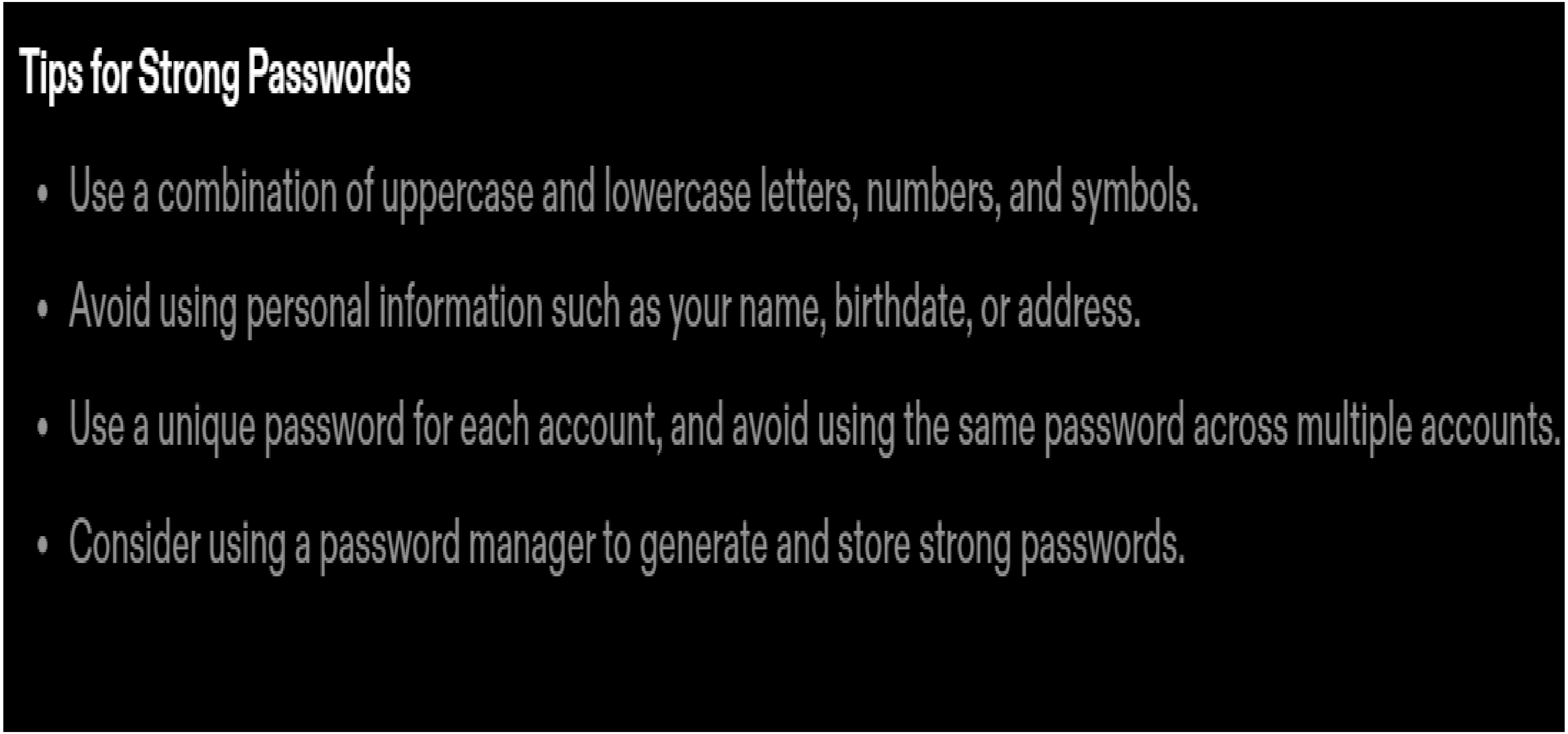


When creating a strong password, it's important to follow some basic guidelines:

- Use a combination of upper and lower case letters, numbers, and symbols.
- Avoid using personal information, such as your name or birthdate.
- Use a password manager to generate and store strong passwords.

Stay Secure

Protect yourself online by using strong passwords and multi-factor authentication. Take the next step by implementing these practices in your daily routine.



Tips for Strong Passwords

- Use a combination of uppercase and lowercase letters, numbers, and symbols.
- Avoid using personal information such as your name, birthdate, or address.
- Use a unique password for each account, and avoid using the same password across multiple accounts.
- Consider using a password manager to generate and store strong passwords.