Designing and Implementing a Random Password Generator using Flask

Mikyle Jones

# Introduction

Validating the identity of individuals has become ever more important in the society we live in today, as we live more of our lives on the internet now. To validate the identity of an individual, many mechanisms has been put into place, one such mechanism, the most popularly used one, is the use of passwords. With so many people having different account for different sites, this means they are most likely re-using the same passwords, which is not good for security. You need to use different passwords for each site to avoid attackers from getting one password and having access to all of your accounts. But remembering different passwords can be difficult and coming up with different passwords can be even more difficult.

## Purpose

The objective of this project is to build a Flask application, that would serve as a Random Password Generator. Here, users would be able to generate a unique random password through the click of a few buttons. It would take away the hassle, from the user, of coming up with unique passwords to remember.

## Functional and Non-Functional Requirements

* User should be able to access the page.
* User can give the password a name, for easy reference. Also, specify the characteristics of the password, i.e. length of the password, if the password should contain digits, letters or symbols etc.
* User can save or copy the password or generate a new one.
* System displays the entire page.
* System allows the user to enter in a password reference name, and the password characteristics.
* System provides the user with a unique password once the user generates the password.
* System allows the user to save the password.

# System Architecture

## Hardware Requirements

Any device with an internet connection and browser should be sufficient.

## Software Requirements

### Display Screen

Display the home page where the user would generate the password.

### Handle User Input

User enters a string to reference the password. Failure cases – Input is left empty so use default.

Length of the password needs to be entered. Failure cases – User did not enter a size so use a default length.

The user chooses if they want to include numbers, letters etc. Failure cases – No preference is given so use default.

### Generate Button

Display unique password, when clicking on generate button. Failure cases – Password can’t be generated due connection issues.

### Save Password

User can click on a button to save the password. Failure cases – The password has not been generated yet so it can’t be saved.

# Design

#### Activity Diagram: Random Password Generator

