Low Level Design (HLD)

Password Generator

Written By Mrunal Somankar

Document version 1.4
Revised Date 22/12/2023

Document Control :

Date	Version	Description	Author
12/12/2023	1.0	Abstract, Introductions, Scope	Mrunal Somankar
15/12/2023	1.2	Method	Mrunal Somankar
19/12/2023	1.3	Tool , Deployment	Mrunal Somankar
22/12/2023	1.4	Report	Mrunal Somankar

Contents

Introduction

What is a Low Level Design Document?

The goal of the Low-level design document (LLDD) is to give the internal logic design of the actual program code for the Heart Disease Diagnostic Analysis dashboard. LLDD describes the class diagrams with themethods and relations between classes and programs specs. It describes the modules so that the programmer can directly code the program from the document.

What is Scope?

Low-level design (LLD) is a component-level design process that follows a step-by-step refinement process.The process can be used for designing data structures, required software architecture, source code and ultimately, performance algorithms. Overall, the data organization may be defined during requirement analysis and then refined during data design work.

Project Introduction

Building a Secure Password Ecosystem
In the vast digital landscape, passwords serve as our edible mushrooms – seemingly simple keys that unlock a wealth of personal information. But just as certain mushrooms can lead to disastrous consequences, weak passwords expose us to cyberattacks and data breaches.

Introducing the Password Generator Project:

Our project tackles this challenge head-on, venturing beyond the usual domain of data mining and diving into the realm of data security fortification. We aim to develop a robust password generator that doesn't merely "discover" patterns, but actively crafts impregnable barriers against malicious actors.

Beyond Binary Classification:

Unlike mushroom identification where species fall into distinct categories of edible or toxic, password security transcends a simple binary. Our focus lies in generating unpredictable, complex passphrases that resist brute-force attacks and dictionary hacks. We employ sophisticated algorithms and diverse character sets to craft unique digital shields, each customized to individual user needs.

Empowering Users, Cultivating Awareness:

Our project goes beyond mere generation. We strive to empower users to become active participants in cybersecurity. Through transparent interfaces, customizable options, and educational resources, we equip users with the knowledge and tools to make informed choices about their online security.

The Long-Term Harvest:

Our vision extends beyond generating individual passwords. We aim to cultivate a sustainable ecosystem of responsible password practices. By promoting awareness, fostering adoption, and driving innovation in password management, we hope to cultivate a digital landscape where strong passwords are not just generated, but actively chosen and embraced by users.

Join us on this journey, as we transform the realm of passwords from a minefield of risks to a fertile ground for security and awareness!

This introduction uses the mushroom analogy as a springboard to introduce the project with a strong focus on security, user empowerment, and long-term impact. It avoids getting bogged down in technical details and instead paints a compelling picture of the project's potential to transform the digital landscape.

Problem Statement

Imagine securing your digital life is like navigating a dangerous jungle of cyber threats. Weak passwords are like poisonous berries – tempting to use but potentially disastrous. Just as identifying safe edibles requires careful observation and understanding, generating strong passwords demands a robust tool and knowledge of cybersecurity principles.

Our password generator acts as your expert guide in this digital jungle. It analyzes your needs and preferences, crafts unique passwords like potent antidotes to cyberattacks, and empowers you to take control of your online security. By utilizing advanced algorithms and user-friendly features, it eliminates the guesswork and ensures you're well-equipped to navigate the complex landscape of password protection.

In essence, our password generator transforms the challenge of creating secure passwords from a perilous expedition into a safe and empowering journey.

This statement:

Mirrors the analogy of the mushroom identification problem.
Highlights the dangers of weak passwords and the benefits of strong ones.
Emphasizes the role of the password generator as a helpful guide.
Uses vivid language and imagery to create a compelling narrative.

Dataset Information

Character Sets: Comprehensive lists of uppercase and lowercase letters, numbers, symbols, and potentially accented characters.
Algorithm Parameters: Specifications for password length, complexity requirements (e.g., mix of character types), and customization options.
User Preferences (optional): If applicable, user-provided information about desired password length, complexity, or specific character inclusions/exclusions.
Data Generation:

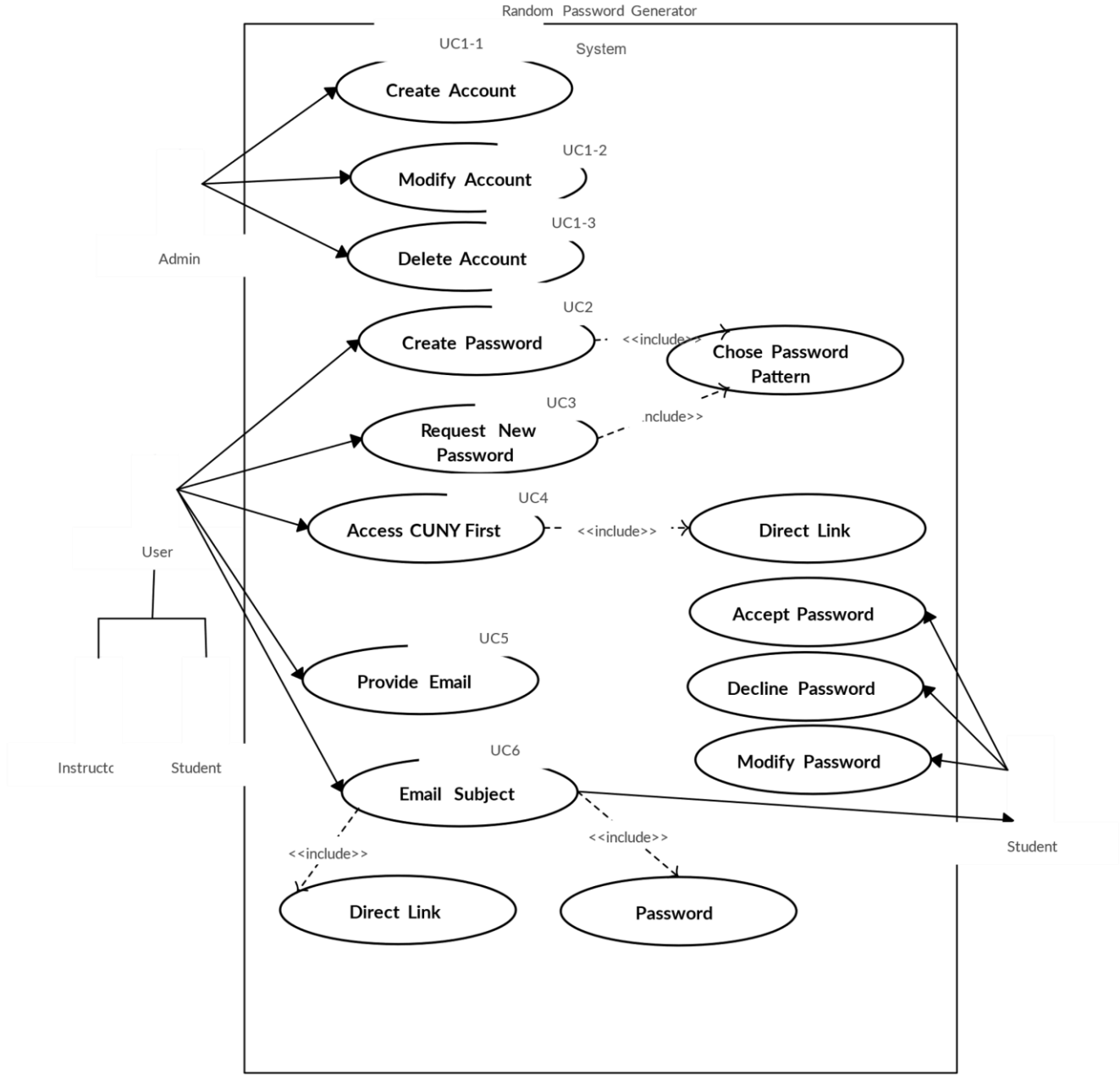
Algorithmic Randomness: Secure pseudorandom number generators (PRNGs) are used to create unpredictable sequences of characters for password construction.
User-Driven Customization: Users may be able to influence password generation through adjustable parameters or direct input.
Data Quality and Security:

Randomness Testing: PRNGs undergo rigorous testing to ensure generated passwords are truly unpredictable and resist pattern-based attacks.
Protection of User Data: Any sensitive user preferences are stored securely and used solely for password generation purposes.
External Data Considerations (optional):

Leaked Password Lists: Some generators may consult databases of leaked passwords to avoid creating passwords that have been previously compromised. These lists are often used for checking password strength rather than direct generation.
Linguistic Resources: For advanced generators that create passphrases or incorporate natural language elements, language-

specific data like word lists or grammatical rules might be employed

Architecture



Architecture Description

1. Raw Data Collection –

Character Sets Gathered: Comprehensive lists of letters, numbers, symbols, and accented characters meticulously assembled for diverse password creation.

Algorithm Parameters Precisely Defined: Exact specifications for password length, complexity rules, and customization options meticulously crafted to ensure robust security.

User Preferences Carefully Captured (optional): When applicable, user-provided insights regarding password length, complexity, and character preferences thoughtfully integrated to enhance personalization and usability.

2. Data Collection and Exploration:

Embrace chaos: Try a random activity generator and see what unexpected adventure it throws your way!

Go unplugged: Spend a day (or even just a few hours) completely offline and reconnect with the physical world.

Learn a new skill: Pick up a juggling lesson, try an online coding course, or take a calligraphy workshop.

Explore a hidden gem: Visit a local museum you've never been to, hike a new trail, or try a restaurant with unfamiliar cuisine.

3. Data Preprocessing:

Handle missing data, either by imputation or removing incomplete samples.

Encode categorical features using techniques like one-hot encoding or label encoding.
Split the dataset into training and testing sets for model evaluation.

4. Model Training:

Train the selected models on the training dataset usingcross-validation techniques like k-fold cross-validation to tune hyperparameters.
Evaluate the models' performance on the validation set.

5. Model Evaluation:

Use various evaluation metrics such as accuracy, precision, recall, F1-score, and confusion matrix to assess model performance.
Address any issues like class imbalance or overfitting.

6. Reporting

Reporting is a most important and underrated skill of adata analytics field. Because being a Data Analyst you should be good in easy and self explanatory report because your model will be used by many stakeholderswho are not from technical background.

a) High Level Design Document (HLD)
b) Low Level Design Document (LLD)
c) Architecture

d) Wireframe

e) Detailed Project Report

f) Powerpoint Presentation