Institute of Geographical Information Systems

CS 212 - Object Oriented Programming

Semester: Fall 2025 | **Class: SCEE-IGIS - 2024**

Group Member 1: Ali Nawaz — **CMS ID : 00000526123**
Group Member 2: Eiman Irshad — **CMS ID : 00000502562**
Group Member 3: Muniba Emaan — **CMS ID : 00000511952**

Submitted to: Ma'am Alvina Anjum | **Due Date: Dec 28, 2025**

Complex Engineering Problems – Project Report

Project Title: Password Manager GUI Application

1. Introduction

The Advanced Password Manager is a desktop application built using **Python and Tkinter**.

It helps users generate strong passwords, check password strength, and securely save login details.

The system is designed using **Object-Oriented Programming (OOP)** concepts so that each part of the application has a clear responsibility.

---

2. Purpose of the System

The main goals of this project are:

- To generate **strong and random passwords**
- To **check password strength** using rules
- To prevent saving weak passwords unless the user allows it
- To store credentials safely in a file
- To provide a **simple and user-friendly interface**

---

3. Main Features

- Generate strong passwords automatically
- Copy generated password to clipboard
- Validate password using multiple rules
- Warn user if password is weak
- Allow forced saving if user wants
- Store credentials with date and time
- Simple graphical interface (GUI)

---

4. System Flow (How the App Works)

1. The application starts and initializes all components.
2. The user sees the password manager window.
3. The user can:
    - Generate a password
    - Save website credentials

4. When saving:
    - The system checks if fields are empty
    - Password strength is analyzed
    - Weak passwords show warning
    - Strong passwords are saved directly
5. Credentials are stored in a text file with timestamp.

---

5. Components of the System

5.1 Password Generator

- Generates random passwords using:
    - Letters
    - Numbers
    - Special symbols
- Ensures the password is strong and unpredictable.

5.2 Password Validators

Password strength is checked using multiple rules:

- Minimum length
- At least one digit
- At least one special character
- At least one alphabet letter

Each rule checks **one condition only**, which makes the system easy to improve later.

5.3 Password Analyzer

- Uses all validators to test a password
- If any rule fails → password is **Weak**
- If all rules pass → password is **Strong**
- Also provides reasons why a password failed

5.4 Storage Manager

- Saves credentials into a file (data.txt)
- Each entry contains:
    - Date & time
    - Website
    - Email
    - Password

5.5 Controller

- Acts as the **bridge** between UI and logic
- Handles:
    - Password generation
    - Password validation
    - Saving credentials
- Keeps UI clean and simple

5.6 User Interface (GUI)

- Built using **Tkinter**
- Provides fields for:
    - Website
    - Email
    - Password
- Buttons:
    - Generate Password
    - Save
- Displays warnings and messages using dialog boxes

---

6. Error Handling

- Shows error if any field is empty
- Warns user if password is weak
- Allows user to decide whether to save weak password
- Prevents crashes using safe checks

7. Technologies Used

- Python
- Tkinter (GUI)
- OOP concepts
- Regular Expressions (password validation)
- File handling

---

8. OOP Concepts Used

- **Abstraction**: Abstract classes for validators, storage, generator
- **Inheritance**: Specific validators extend base validator
- **Encapsulation**: Each class handles its own task
- **Polymorphism**: Different validators follow same interface

---

9. Advantages of the System

- Easy to understand and use
- Strong password enforcement
- Modular and extensible design
- Safe handling of credentials
- Follows clean coding practices

---

10. Conclusion

This Advanced Password Manager demonstrates how **OOP principles** can be applied to build a real-world application.

The project focuses on **security, clarity, and maintainability**, making it suitable for academic and practical use.
