INTERNSHIP PROJECT PHASE REPORT

TITLE: Password Strength Analyzer with Custom Wordlist Generator

INTRODUCTION:

In the digital age, strong passwords are essential for safeguarding personal and sensitive information.
This project, **Password Strength Analyzer with Custom Wordlist Generator**, helps users evaluate password strength.
It uses the **zxcvbn** library to estimate how secure a password is and how long it would take to crack it. The tool provides instant feedback and suggestions for improvement.
Users can also generate custom wordlists using personal information like name, DOB, and pet name. This feature helps demonstrate how attackers might guess weak passwords based on user data. A stylish and user-friendly GUI is created using **ttkbootstrap** for better usability.

ABSTRACT:

This project presents a **Password Strength Analyzer with Custom Wordlist Generator**, designed to assess and improve password security.
By leveraging the **zxcvbn** library, the tool evaluates password strength using real-world data and cracking patterns.
It provides users with a visual score, estimated crack time, and actionable suggestions for enhancing their passwords.
The additional wordlist generator uses personal information to simulate targeted attacks for educational purposes.
A user-friendly interface built with **ttkbootstrap** ensures accessibility for both technical and non-technical users.
The project aims to raise awareness about common password vulnerabilities.
It serves as a valuable tool for both individuals and cybersecurity learners to promote better password practices.

TOOL USED:

- **Python** – The core programming language used for developing backend logic and GUI functionality.
- **zxcvbn** – A password strength estimation library developed by Dropbox, used to evaluate the security of passwords.
- **Tkinter** – Python's standard GUI library, used to create desktop-style interfaces for user interaction.
- **ttkbootstrap** – A modern theme extension of Tkinter that provides attractive, responsive widgets for enhanced user experience.
- **argparse** – Used in CLI-based versions (optional) to handle command-line arguments for custom wordlist generation.
- **OS & File I/O** – For saving analysis results and exporting wordlists to local directories.
- **Text Editor / IDE (VS Code, PyCharm, etc.)** – Used for writing and organizing the project code.

STEPS INVOLVED IN BUILDING THE PROJECT:

- **Requirement Analysis**

Identified the need for a user-friendly tool to evaluate password strength and generate custom wordlists based on user data.

- **Technology Selection**

Chose Python for backend logic, `zxcvbn` for strength evaluation, and `ttkbootstrap` with `Tkinter` for GUI design.

- **Password Analyzer Development**

Created a Python module that uses `zxcvbn` to analyze password complexity, estimate crack time, and provide feedback.

- **Wordlist Generator Module**

Developed a function to generate potential password wordlists using personal details like name, date of birth, and pet name.

- **GUI Design with ttkbootstrap**

Built an interactive GUI using `ttkbootstrap` to allow users to input passwords and receive real-time analysis with feedback and suggestions.

- **Testing and Debugging**

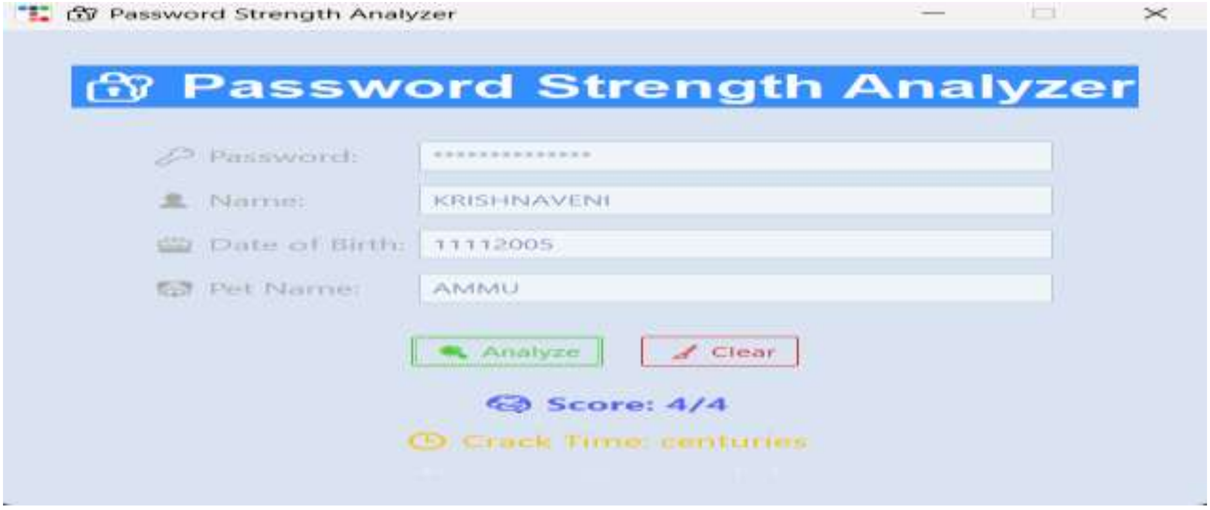
Verified all components (password analysis, wordlist generation, and file export) work correctly across multiple inputs and scenarios.

- **Packaging and Output Handling**

Ensured results are saved to appropriate output files (e.g., `analysis_result.txt`, `custom_wordlist.txt`) for review and further use.

CONCLUSION:

The Password Strength Analyzer helps users create strong passwords by analyzing their strength and giving feedback. It includes a wordlist generator based on user input for additional security auditing. The tool combines functionality and user-friendliness through a simple, effective GUI.



Password Strength Analyzer

Password: ***************

Name: KRISHNAVENI

Date of Birth: 11112005

Pet Name: AMMU

Score: 4/4

Crack Time: centuries

GITHUB REPOSITOY LINK: https://github.com/kveni1105/Password-_analyzer