Bozza di progetto
Password Strength Meter Engine

Team: *Belviso-Didonna-Vegliante*

1 Introduzione e obiettivi

In questo progetto vogliamo sviluppare un **Password Strength Meter Engine** che sia utilizzabile sia da utenti finali (ad esempio studenti o docenti che compilano un form di accesso) sia in contesti di valutazione sperimentale.

L'idea è di andare oltre i controlli tradizionali basati solo su lunghezza minima e presenza di certe categorie di caratteri, e di avvicinarci il più possibile alla nozione di *robustezza effettiva* della password. In particolare, il sistema dovrà:

- valutare in tempo reale la forza di una password mentre l'utente la digita;
- fornire suggerimenti concreti per renderla più robusta;
- esporre un'API che permetta ad altri sistemi di usare il nostro engine;
- includere un modulo per eseguire test su dataset di password e confrontare il nostro meter con uno strumento di riferimento (*baseline*);
- produrre risultati e statistiche che potremo discutere nella relazione finale.

Come riferimento teorico utilizzeremo i paper consigliati:

- Ur et al., *"How Does Your Password Measure Up? The Effect of Strength Meters on Password Creation"*, USENIX Security 2012;
- Wang et al., *"No Single Silver Bullet: Measuring the Accuracy of Password Strength Meters"*, USENIX Security 2023.

Questi lavori ci serviranno sia come supporto alla progettazione dell'algoritmo di valutazione, sia per definire come costruire il confronto sperimentale con un Baseline.

2 Panoramica del sistema e attori

Modelliamo il sistema come **Password Strength Meter Engine**, che interagisce con quattro attori principali:

- **Utente finale**: compila un form di accesso o registrazione, inserendo nome utente, email e password. Durante l'inserimento della password riceve in tempo reale indicazioni sulla forza e sui possibili miglioramenti.
- **Amministratore / Ricercatore**: utilizza il modulo sperimentale per caricare dataset di password, eseguire test con il nostro PSM e con il baseline, visualizzare statistiche ed esportare i risultati.
- **Sistema esterno**: applicazione di terze parti che invoca il nostro engine tramite API per ottenere una valutazione della password in fase di registrazione o cambio credenziali.
- **Baseline / Strumento esterno**: meter di riferimento (libreria o servizio) che usiamo per avere una seconda valutazione delle password nei test comparativi.

Nel diagramma dei casi d'uso allegato abbiamo cercato di riassumere lo scenario completo del sistema. Di seguito descriviamo i casi d'uso principali, organizzati per attore.

2.1 Casi d'uso lato Utente finale

Per l'utente finale consideriamo tre casi d'uso principali:

- **Compilare credenziali di accesso (nome utente, email, password)**: l'utente inserisce i propri dati nel form. Questo caso d'uso rappresenta il flusso "normale" di una schermata di login/registrazione.
- **Valutare la forza della password**: ogni volta che la password viene modificata, il sistema calcola un punteggio di forza e assegna un livello (ad esempio: debole, media, forte).
- **Generare suggerimenti di miglioramento**: a partire dall'analisi della password, il sistema produce messaggi testuali che spiegano come renderla più robusta (es. aggiungere caratteri, evitare pattern banali, evitare dati personali).

Nel nostro modello di casi d'uso, **"Compilare credenziali di accesso" include "Valutare la forza della password" e "Generare suggerimenti di miglioramento"**. In questo modo, durante la compilazione del form l'utente ha subito un feedback immediato sull'efficacia e sicurezza della password scelta.

2.2 Casi d'uso lato Amministratore / Ricercatore

Per l'amministratore o ricercatore definiamo i seguenti casi d'uso:

- **Caricare un dataset di password**: importare un dataset di password di prova (ad esempio in formato CSV), che verrà poi usato per gli esperimenti.
- **Eseguire test del PSM su un dataset**: applicare il nostro Password Strength Meter a tutte le password del dataset, salvando per ciascuna punteggio, livello di forza e altre informazioni utili.
- **Eseguire un test comparativo con il baseline**: eseguire in parallelo il test con il nostro PSM e con il baseline sullo stesso dataset, in modo da poter confrontare le due valutazioni.
- **Visualizzare risultati e statistiche**: consultare i risultati degli esperimenti, ad esempio distribuzioni dei punteggi, percentuali di password classificate come deboli/forti, differenze tra PSM e baseline.
- **Esportare risultati degli esperimenti**: salvare i risultati (ad esempio in CSV/JSON o tabelle) per poterli analizzare ulteriormente e inserirli nella relazione.

Il caso d'uso **"Eseguire test del PSM su un dataset" include "Valutare la forza della password"**, in quanto l'engine viene invocato su ciascuna password del dataset.

Il caso d'uso **"Eseguire un test comparativo con il baseline" include sia "Eseguire test del PSM su un dataset" sia "Ottenere valutazione del baseline"**. In pratica, quando lanciamo un test comparativo eseguiamo automaticamente sia il test con il nostro PSM sia il test con lo strumento di riferimento.

Per **"Visualizzare risultati e statistiche"** e **"Esportare risultati degli esperimenti"** preferiamo non modellare ulteriori `include` nel diagramma: consideriamo questi casi d'uso come operazioni che partono dal presupposto che almeno un esperimento sia già stato eseguito (test semplice o comparativo). Le eventuali condizioni (ad esempio "se sono disponibili anche i risultati del baseline, mostra anche il confronto") verranno descritte nel testo della specifica.

2.3 Casi d'uso lato Sistema esterno e Baseline

Per gli attori tecnici abbiamo:

- **Richiedere valutazione password via API** (Sistema esterno): un'applicazione invia una password ad un endpoint esposto dal nostro engine e riceve in risposta il punteggio di forza (ed eventualmente i suggerimenti di miglioramento). Questo caso d'uso include la logica di **"Valutare la forza della password"** e, se richiesto, anche quella di **"Generare suggerimenti di miglioramento"**.
- **Ottenere valutazione del baseline** (Baseline): il nostro sistema invoca il meter di riferimento per ottenere la valutazione della stessa password (o dello stesso dataset) utilizzata nei test comparativi. Questo caso d'uso è incluso da "Eseguire un test comparativo con il baseline".

3 Architettura logica prevista

A partire da questi casi d'uso abbiamo individuato una prima decomposizione logica del sistema in moduli:

- **PasswordStrengthEngine**: componente centrale che riceve una password e restituisce punteggio di forza, livello e informazioni sui pattern rilevati.
- **RuleSet / ScoringModel**: insieme di regole, parametri e soglie utilizzate dall'engine per il calcolo del punteggio.
- **FeedbackGenerator**: prende in input il risultato dell'engine e produce i messaggi di feedback e i suggerimenti di miglioramento da mostrare all'utente.
- **UI Web / UIController**: gestisce il form di accesso/registrazione, l'aggiornamento in tempo reale della barra di forza e la visualizzazione dei suggerimenti.
- **API Module**: espone un'interfaccia per il caso d'uso "Richiedere valutazione password via API".
- **DatasetManager**: si occupa del caricamento dei dataset di password e della validazione del formato.
- **EvaluationModule / ExperimentRunner**: esegue i test del PSM e i test comparativi con il baseline sui dataset caricati, coordinando le chiamate all'engine e al baseline.
- **BaselineAdapter**: incapsula le chiamate allo strumento baseline e normalizza il formato della sua risposta in modo da poterla confrontare con l'output del nostro PSM.
- **ResultsAnalyzer / ReportGenerator**: elabora i risultati degli esperimenti, calcola statistiche e produce report/tabelle da utilizzare nella relazione e nella valutazione finale del progetto.