**ASSIGNMENT#3**

**OBEJCT-ORIENTED PROGRAMMING PROJECT**

-***Hangman Game***

## 

**Prepared By:** Rameen Asif

**Registration No:** FA23-BSE-A-157

# **Abstract**

Initially, it features a login window where users input their username and password, which are then checked by reading from a file. In case of incorrect username or password alert window would be generated. Upon successful authentication, the interface transitions to a menu scene where users can choose from three categories: movies, countries, or animals. Each category has a predefined list of words, and one is randomly selected as the word to be guessed. This chosen word is initially displayed as underscores. In the game scene, users can make guesses using a virtual keyboard’s buttons. Correct guesses increase the user's score by 1, while incorrect ones reveal parts of the hangman figure sequentially. If the word is fully guessed before the hangman figure is completed, a "You win" message appears otherwise, if the full figure is displayed before guessing correctly, a "You lose" message appears. Users will have options to restart or play again, and their scores are persistently tracked through file handling. User will have play again and restart options by their respective buttons. Overall, the project encompasses login authentication, category selection, word selection, gameplay mechanics, score tracking, and user feedback within a unified GUI Hangman game.

# **Classes**

This project will consist of 6 classes namely:

* Movie: Word Category Selection Class
* Animal: Word Category Selection Class
* Country: Word Category Selection Class
* Game Scene: Handles GUI and main game play scene
* File Handling: Stores player’s score to file and read username and password saved in file
* Menu: Handles event handling of Animal, Movie and Country buttons in the selection menu scene.

## Interface

There will be one interface called ‘Word’ which will act as blueprint for Animal, Country and Movie classes to implement the generateRandomWord() method and adds predefined list of words in predefinedWords() method, allowing each class to specify their own list of words.