Riddle Me This

A guessing game

This is the milestone project that I have to build for the “Practical Python” stream, part of the New LMS in the “Full Stack Web Development Course” offered by Code Institute.

What would I like to achieve?

The first decision I felt I have to make is wether to use a text-based or a picture-based riddle. I did a google search and I found some material that I can use as a picture-based riddle game. I also thought that a picture-based would be more attractive to a wide range of users of different ages. I decided to go down this route.

Then I sketched a few layouts (pencil and paper) to get a general feel of how I would get the game to flow and an idea of layout of various parts. I sketched the logic of the game using a flow diagram.

The game will randomly select 10 riddles from a pool of riddles. The user will be presented with the first riddle. The user has to write the answer for the riddle and submit the answer by pressing a button. An option to pass is also provided.

A submitted answer will be checked against the stored answer and if correct a second riddle is presented. If the answer is wrong or if the user chooses to pass, a second attempt at answering will be offered. This time the user will get a feel of how many words are expected in the answer. Similar submit and pass options are offered. If the answer is wrong or the user passes, a third attempt is offered. This time the user will be shown how many words should be in the answer and how many letters are in each word. After this final attempt the user will be directed to the next riddle unless it is the final one, in which case the game will end and the user will be directed to the user page.

When a new user arrives at this app, there will be the option to register. I am going to keep registration simple and without the need to pass on any personal information. Registration only requires the user to create a unique username. When the user selects a username and sends it to the back end through a button called “Check Username Availability”, the username will be checked against a stored list (case sensitive – have not decided if I will accept anything beyond letters). If it is available, a text to this effect will be displayed and the button “Register” will be activated. The user will click this button and registration is complete.

The reason for selecting a username only approach is to avoid storing personal data like email in order to avoid legalities. I think that for the purpose of this project this will suffice.

Once a user registered they will be taken directly to their respective user page. If a user revisits the app they will login.

The game itself is very simple. A flow diagram has been produced which shows the outline of the procedures.

First Attempt: The user will be presented with an image and a text area with no further clues. The user will try to guess and write the answer in the text area. Then this is submitted to the server. If the answer is correct, 10 points will be gained and the next riddle is presented. If the answer is wrong or the user gives up and passes:

Second Attempt: The user will be presented with the same image, the text area will be replaced

with text fields, one for each word expected in the answer. The previous wrong answer will be shown. If a correct guess is made, 6 points will be awarded and the next riddle is presented. In case the answer is wrong or the user passes:

Third Attempt: The user will see the image, the text fields and a number showing the number of letters expected in each word. These clues are meant to guide and help the user to guess the answer. 3 point will be awarded if a correct guess is made. If the user makes a wrong answer or passes, no points will be awarded and the next riddle will be presented.

For each attempt, if the riddle is the last riddle of the game, then a game over splash screen (or something similar) will be presented. The User name and the points gained will be shown. After this (dismiss page) the user will be taken to the user page. Here the new game would have been added to the list of games played (most recent at the top) and the user can choose to play another game by using the “Start New Game” button.

Data stores

I have not decided yet the format of the storage within the game but keeping the data externally I will probably use a text file storing the data in json format.

When I load the data in the game I will need to decide if sorting is important or not. Another possible consideration will be uniqueness of data.

Pool of Riddles

Current Game Riddles

User names

Users games

Hall of Fame per game

Overall Hall of Fame

Extra

1. I would like to implement a way for the user to **navigate away** from a game which is being played. The game will be played in a sort of Pause. Then the user can navigate back to the game to finish it. Otherwise the user will be given the opportunity to abandon the game completely. In this case I need to decide if that game will still be added to the user's record of played games or trashed completely.
2. I would like to add a **timer** for each attempt to make it a bit more competitive. A set time is given for each attempt. Failing to submit a correct answer within the time allowed will be taken as a wrong answer.
3. I would like to add **sound** to the game. This will be in two forms; background music and functional sounds. Controls will allow to turn off both components. The functional sounds can accompany button clicks such as “Start New Game”, “Submit Answer”, “Pass”.
   There can also be ticking sounds for the timer, louder and more intense for the last 3 or 5 seconds. Background music can get more intense when moving to the second and third attempt.

Appendix 1

Technologies Used

- HTML
- CSS
- Python
- Template Language
- Bootstrap

Appendix 2

How will I address the following:

Usability and Real-world application

- Project Purpose
- UX design
- Suitability for purpose
- Navigation
- Ease of use
- Information Architecture
- Defensive Design

Layout and Visual Impact

- Responsive Design
- Image Presentation
- Colour scheme and typography

Code Quality

- Appropriate use of HTML
- Appropriate use of CSS
- Appropriate use of Python
- Appropriate use of Template Language

Software Development Practices

- Directory Structure and File Naming
- Version Control
- Testing Implementation
- Testing write-up
- Readme file
- Comments
- Data Store integration
- Deployment implementation
- Deployment Write-up