**Project Proposal: Magic 8 Ball Game Application**

**Project Overview**

The Magic 8 Ball Game Application is an interactive program that simulates the classic Magic 8 Ball toy. Users can input their questions, and the application will respond with a random answer, just like the traditional toy. This application aims to provide a fun and engaging user experience with a simple interface and accurate input validation.

**Objectives**

* Develop a console-based Magic 8 Ball game.
* Implement input validation to ensure only alphabetic characters and spaces are accepted.
* Provide a diverse range of responses to mimic the traditional Magic 8 Ball experience.
* Ensure the application is user-friendly and accessible to a wide audience.

**Target Audience**

The target audience includes individuals of all ages who enjoy casual entertainment and novelty applications. This application is suitable for users who seek a simple and fun way to pass the time and get playful answers to their questions.

**Features**

1. **Interactive Console Interface**:
   * Users can type questions directly into the console.
   * The application will provide responses in a conversational format.
2. **Input Validation**:
   * Ensures that the user inputs only alphabetic characters and spaces.
   * Provides error messages for invalid inputs to guide the user.
3. **Random Response Generation**:
   * Uses a random number generator to select a response from a predefined list of 20 possible answers.

**Technical Requirements**

* **Programming Language**: Python
* **Libraries**:
  + random for generating random responses.
  + re for input validation.

**System Components**

1. **Console Interface**:
   * **Input Field**: Users type their questions directly into the console.
   * **Output**: The application prints the Magic 8 Ball’s response in the console.
2. **Backend Logic**:
   * **Input Validation**: Uses regular expressions to ensure only valid questions are accepted.
   * **Random Response Generator**: Selects a response randomly from the predefined list.

**Development Plan**

1. **Initial Setup**:
   * Set up the development environment.
   * Create a basic structure for the application.
2. **Backend Development**:
   * Implement the input validation logic.
   * Create the random response generator.
3. **Integration**:
   * Integrate the input validation and random response generation with the console interface.
   * Ensure smooth interaction between the components.
4. **Testing and Validation**:
   * Perform thorough testing to ensure all features work as expected.
   * Validate the input and ensure accurate responses.
5. **Deployment**:
   * Package the application for distribution.
   * Provide documentation and user guides.

**Project Timeline**

* **Week 1**: Initial setup and backend development.
* **Week 2**: Integration of components and testing.
* **Week 3**: Final testing, validation, and deployment.

**Conclusion**

The Magic 8 Ball Game Application is a straightforward yet entertaining project that will provide users with a modern twist on a classic toy. By leveraging Python and its libraries, we can create a robust and enjoyable user experience. The project is expected to be completed within three weeks, ensuring timely delivery and deployment.

**Code Explanation**

1. **Importing Libraries**:

import random

import re

* **random**: This library is used for generating random numbers, which will be used to select a random response.
* **re**: This library provides support for regular expressions, which will be used for validating user input to ensure it contains only alphabetic characters and spaces.

1. **Main Loop**:

while True:

* This is an infinite loop that will keep the program running until the user decides to quit.

1. **User Input**:

* question = input("Ask a question: (press enter to quit) ")
* This line prompts the user to type a question. The user's input is stored in the variable question.

1. **Exit Condition**:

if question == "":

print("Goodbye")

break

* This condition checks if the user pressed enter without typing anything.
* If true, it prints "Goodbye" and breaks out of the loop, ending the program.

1. **Input Validation**:

if not re.match(r"^[A-Za-z\s]+$", question):

print("Please enter a valid question containing only letters and spaces.")

continue

* **re.match(r"^[A-Za-z\s]+$", question)**: This regular expression checks if the input contains only alphabetic characters (both uppercase and lowercase) and spaces.
* If the input does not match this pattern, it prints an error message and continues to the next iteration of the loop, prompting the user to enter a valid question again.

1. **Random Response Generation**:

answer = random.randrange(1, 20)

* **random.randrange(1, 20)**: This generates a random number between 1 and 20 (inclusive). This number is used to select a random response from the predefined list of answers.

1. **Display the Magic 8 Ball's Response**:

print("Magic 8 Ball says: ")

if answer == 1:

print("Yes, Definitely")

elif answer == 2:

print("Without a Doubt")

elif answer == 3:

print("It is certain")

elif answer == 4:

print("It is decidedly so")

elif answer == 5:

print("You may rely on it")

elif answer == 6:

print("Most Likely")

elif answer == 7:

print("Outlook seems good")

elif answer == 8:

print("Signs point to Yes!")

elif answer == 9:

print("Without a Doubt")

elif answer == 10:

print("You may rely on it")

elif answer == 11:

print("Please try again")

elif answer == 12:

print("Ask again later")

elif answer == 13:

print("Better not tell you right now")

elif answer == 14:

print("Cannot Predict Right now")

elif answer == 15:

print("Concentrate and ask again")

elif answer == 16:

print("Don't Count on it")

elif answer == 17:

print("My reply is No")

elif answer == 18:

print("My sources say no")

elif answer == 19:

print("Outlook not so good")

elif answer == 20:

print("Very Doubtful")

* This section prints "Magic 8 Ball says: " followed by the response corresponding to the randomly generated number.
* Each elif statement checks if the answer matches a specific number and prints the associated response.

**Pseudocode**

BEGIN

IMPORT random module

IMPORT re module

PRINT "Welcome to the Magic 8-Ball. Where we answer all your questions"

WHILE True DO

PRINT "Ask a question: (press enter to quit)"

READ question *from* user input

IF question *is* an empty string THEN

PRINT "Goodbye, Thanks for stopping by!"

BREAK *from* loop

END IF

IF question does *not* match the regular expression *for* only alphabetic characters *and* spaces THEN

PRINT "Please enter a valid question containing only letters and spaces."

CONTINUE to next iteration

END IF

SET answer to a random integer between 1 *and* 20

PRINT "Magic 8 Ball says:"

IF answer *is* 1 THEN

PRINT "Yes, Definitely"

ELSE IF answer *is* 2 THEN

PRINT "Without a Doubt"

ELSE IF answer *is* 3 THEN

PRINT "It is certain"

ELSE IF answer *is* 4 THEN

PRINT "It is decidedly"

ELSE IF answer *is* 5 THEN

PRINT "You may rely on it"

ELSE IF answer *is* 6 THEN

PRINT "Most Likely"

ELSE IF answer *is* 7 THEN

PRINT "Outlook seems good"

ELSE IF answer *is* 8 THEN

PRINT "Signs point to Yes!"

ELSE IF answer *is* 9 THEN

PRINT "Without a Doubt"

ELSE IF answer *is* 10 THEN

PRINT "You may rely on it"

ELSE IF answer *is* 11 THEN

PRINT "Please try again"

ELSE IF answer *is* 12 THEN

PRINT "Ask again later"

ELSE IF answer *is* 13 THEN

PRINT "Better not tell you right now"

ELSE IF answer *is* 14 THEN

PRINT "Cannot Predict Right now"

ELSE IF answer *is* 15 THEN

PRINT "Concentrate and ask again"

ELSE IF answer *is* 16 THEN

PRINT "Don't Count on it"

ELSE IF answer *is* 17 THEN

PRINT "My reply is No"

ELSE IF answer *is* 18 THEN

PRINT "My sources say no"

ELSE IF answer *is* 19 THEN

PRINT "Outlook not so good"

ELSE IF answer *is* 20 THEN

PRINT "Very Doubtful"

END IF

END WHILE

END