**Part 1: Initial Project Ideas**

**1. Expert System for Diagnostics**

* **What the system does:** Helps diagnose problems, such as identifying common illnesses based on symptoms provided by the user.
* **How it works using rules:** Uses a set of predefined rules to ask follow-up questions and reach a conclusion. For example, "if fever and cough, suggest cold; if fever and rash, suggest measles." The system evaluates inputs against a decision tree or logical conditions to provide diagnoses.

**2. Simple Game AI**

* **What the system does:** Creates an AI that plays a game like Tic-Tac-Toe, making strategic moves based on hardcoded rules.
* **How it works using rules:** Uses rules like "if two in a row, block the third" or "if the center square is open, take it." The AI evaluates the game board state and applies these rules to decide its actions.

**3. Text-Based Quiz System**

* **What the system does:** Provides a quiz where users answer questions, get feedback, and track their score. It adapts the difficulty or hints based on performance.
* **How it works using rules:** Uses a series of if-else conditions or decision trees to evaluate answers, provide feedback, and determine the next question. Rules like "if the answer is incorrect, give a hint" or "if score > X, increase difficulty" guide the system's behavior.

**Chosen Idea: Text-Based Quiz System**

**Justification:** I chose the text-based quiz system because it has real-world applications and aligns perfectly with my work on the AuraBloom mental health application. This system could be used for interactive quizzes or exercises to engage users and provide meaningful feedback.

**Part 2: Rules/Logic for the Chosen System**

The **Text-Based Quiz** system will follow these rules:

**Objective:**

To create a self-assessment quiz based on the MMSE format that evaluates cognitive abilities such as orientation, memory, attention, recall, and language.

**Rules and Logic:**

Here’s a breakdown of the rules and how the system will function:

**Rules Organized by Section**

**1. Orientation**

* **Purpose:** Test awareness of time, place, and personal details.
* **Questions and Rules:**
  1. **Question:** "What day of the week is it?"
     + **IF** the user’s answer matches the correct day (e.g., "Monday"), **THEN** display "Correct! You're aware of the day." and increment the score.
     + **ELSE** display "That's okay. Awareness of time can vary during the day."
  2. **Question:** "What city are you currently in?"
     + **IF** the user’s answer matches the correct city (e.g., "New York"), **THEN** display positive feedback and increment the score.
     + **ELSE** provide supportive feedback.

**2. Registration (Memory)**

* **Purpose:** Test the ability to remember and repeat information.
* **Questions and Rules:**
  1. **Task:** Ask the user to remember three unrelated words (e.g., "Apple, Car, Book").
     + **Rule:** Store the words temporarily for use in the Recall section.
  2. **Question:** "Please repeat these three words: Apple, Car, Book."
     + **IF** the user repeats all words correctly, **THEN** display "Excellent memory!" and increment the score.
     + **ELSE** display "Good try! We'll revisit this later."

**3. Attention and Calculation**

* **Purpose:** Test focus and problem-solving skills.
* **Questions and Rules:**
  1. **Task:** "Start with 100 and subtract 7 repeatedly (e.g., 100, 93, 86...)."
     + **IF** the user provides a correct sequence, **THEN** increment the score and display "Good job! You're staying focused."
     + **ELSE** display "Math under pressure can be tricky, but keep practicing!"
  2. **Task:** "Spell the word WORLD backward."
     + **IF** the user spells it correctly (e.g., "DLROW"), **THEN** display "Great focus and problem-solving!" and increment the score.
     + **ELSE** provide supportive feedback.

**4. Recall**

* **Purpose:** Test short-term memory.
* **Question and Rules:**
  1. **Question:** "What were the three words I asked you to remember earlier?"
     + **IF** the user recalls all three words (e.g., "Apple, Car, Book"), **THEN** display "Excellent recall!" and increment the score.
     + **IF** the user recalls 1-2 words, **THEN** display "Good effort! Recall can vary."
     + **ELSE** display "No worries! Memory is something we can strengthen."

**5. Language**

* **Purpose:** Test reading, writing, and comprehension.
* **Questions and Rules:**
  1. **Question:** "What is this object called?" (Show or describe an object like a pen or watch).
     + **IF** the user correctly identifies the object, **THEN** display "Correct!" and increment the score.
     + **ELSE** provide neutral feedback like: "That's okay. Let's move on."
  2. **Task:** "Read and follow this instruction: 'Close your eyes.'"
     + **IF** the user responds correctly, **THEN** display "Good comprehension!" and increment the score.
     + **ELSE** display "That's alright. Let's continue."

**Scoring and Feedback**

1. **Total Scoring System**: Assign 1 point per correct answer. Maximum score = 30.
2. **Feedback Based on Score**:
   * **25-30:** "Great cognitive performance!"
   * **20-24:** "Good job! Some areas might need attention."
   * **< 20:** "Consider reaching out to a professional for additional support."
3. **Replay Option**:
   * **IF** the user wants to retry, **THEN** reset the score and restart the quiz.
   * **ELSE** display a supportive farewell message.

**Pseudocode**

START QUIZ

DISPLAY welcome\_message and disclaimer

# Orientation Section

DISPLAY "What day of the week is it?"

GET user\_input

IF user\_input == correct\_answer:

INCREMENT score

DISPLAY "Correct!"

ELSE:

DISPLAY "That's okay!"

DISPLAY "What city are you in?"

GET user\_input

IF user\_input == correct\_answer:

INCREMENT score

DISPLAY "Correct!"

ELSE:

DISPLAY "That's okay!"

# Registration Section

DISPLAY "Repeat these three words: Apple, Car, Book."

STORE words

GET user\_input

IF user\_input matches all words:

INCREMENT score

DISPLAY "Great job!"

ELSE:

DISPLAY "Good try!"

# Attention and Calculation Section

DISPLAY "Start with 100 and subtract 7 repeatedly."

WHILE correct\_sequence:

GET user\_input

IF user\_input == next\_correct\_number:

INCREMENT score

ELSE:

DISPLAY "Good effort!"

DISPLAY "Spell 'WORLD' backward."

GET user\_input

IF user\_input == "DLROW":

INCREMENT score

DISPLAY "Well done!"

ELSE:

DISPLAY "That's okay!"

# Recall Section

DISPLAY "What were the three words I asked you to remember earlier?"

GET user\_input

IF user\_input matches stored words:

INCREMENT score

DISPLAY "Excellent recall!"

ELSE:

DISPLAY "Good try!"

# Language Section

DISPLAY "Read and follow this instruction: 'Close your eyes.'"

GET user\_input

IF user\_follows\_instruction:

INCREMENT score

DISPLAY "Well done!"

ELSE:

DISPLAY "That's okay!"

# Final Score and Feedback

CALCULATE score

IF score >= 25:

DISPLAY "Great cognitive performance!"

ELSE IF score >= 20:

DISPLAY "Good job! Some areas might need attention."

ELSE:

DISPLAY "Consider reaching out to a professional."

ASK "Do you want to take the quiz again?"

IF yes:

RESET score

RESTART quiz

ELSE:

DISPLAY "Thank you for taking the quiz!"

END

**Part 3: Rules/Logic for the Chosen System**

Here’s an example of **Part 3: Rules/Logic for the Chosen System** tailored to your **self-assessment MMSE-style quiz**:

**Part 3: Rules/Logic for the Self-Assessment Quiz**

**Sample Input and Output**

**Welcome and Disclaimer**  
**System Output:**  
"Welcome to the Self-Assessment Quiz! This quiz is inspired by the MMSE and is for educational purposes only. It is not a diagnostic tool. Let's begin!"

**Section: Orientation**  
**System Output:**  
"What day of the week is it?"  
**User Input:** "Monday"  
**System Output:** "Correct! You're aware of the day."

**Section: Registration (Memory)**  
**System Output:**  
"I'm going to say three words. Please remember them: Apple, Car, Book."  
**System Output:**  
"Now, repeat these three words."  
**User Input:** "Apple, Car, Book"  
**System Output:** "Excellent memory! We'll revisit these words later."

**Section: Attention and Calculation**  
**System Output:**  
"Start with 100 and subtract 7 repeatedly (e.g., 100, 93, 86...). Enter your numbers one at a time."  
**User Input:** "100"  
**System Output:** "Great! Next number?"  
**User Input:** "93"  
**System Output:** "Correct! Next?"  
**User Input:** "85"  
**System Output:** "Incorrect. The correct number is 86. Let's move on."

**Section: Recall (Memory)**  
**System Output:**  
"Now, what were the three words I asked you to remember earlier?"  
**User Input:** "Apple, Car, Book"  
**System Output:** "Excellent recall! You remembered all three words."

**Section: Language**  
**System Output:**  
"Spell the word 'WORLD' backward."  
**User Input:** "DLROW"  
**System Output:** "Correct! Great focus and problem-solving."

**System Output:**  
"Read this sentence and follow the instructions: 'Close your eyes.'"  
**User Input:** *Closes eyes*  
**System Output:** "Good job! Your comprehension is strong."

**End of Quiz and Feedback**  
**System Output:**  
"Your final score is: 27/30."  
"Great cognitive performance! Keep up the good work. If you'd like to try again or explore other assessments, type 'RETRY' or 'EXIT.'"

**Part 4: Reflection**

**Project Overview:**

This project involved designing a rule-based self-assessment system based off the Mini-Mental State Examination (MMSE). The MMSE system evaluates cognitive abilities through sections of questions related to orientation, memory, attention, and language. I followed this system to create a rule-based text quiz. Each section follows rules structured using if-then-else logic to assess user inputs and provide immediate feedback. Scores are then tallied to offer an overall assessment and provide feedback based on that score.

**Challenges:**

* 1. **Balancing Complexity with Clarity:**
     + Designing questions that test cognitive abilities while keeping them intuitive for users was a bit of a challenge.
  2. **Handling User Inputs:**
     + Allowing flexibility in the users’ input. I wanted to make sure that things like capitalization, for example, did not skew the results.