**Part1: Prompting AI for Project Ideas**

To begin the assignment, I prompted an AI assistant (ChatGPT) with the following request:

"I am working on a Python project to create a simple AI system based on rules or heuristics. This is for an assignment where I need to understand how AI worked before machine learning. Could you suggest some project ideas for a rule-based AI system? Please include examples like chatbots, recommendation systems, or diagnostic tools."

Based on this prompt, the AI suggested the following three project ideas:

1. **Mental Health Support Chatbot:**

* **What it does:** This chatbot provides supportive messages based on user input regarding their emotional state. It responds with encouragement, coping strategies, or self-care suggestions when the user expresses feelings like stress, sadness, or anxiety.
* **How it works using rules:** The chatbot detects specific keywords (e.g., "stressed," "sad," "anxious") and matches them to predefined responses. For example:
* IF user says "I feel stressed" → THEN respond with "I'm sorry to hear that. Have you tried deep breathing exercises?"
* IF user says "I feel lonely" → THEN respond with "Loneliness can be tough. Reaching out to a friend or journaling may help."

1. **Daily Affirmation Generator**

* **What it does:** This system generates daily affirmations based on user-selected themes such as confidence, gratitude, or motivation.
* **How it works using rules:** The system predefines affirmations for different categories. When a user selects a theme, it retrieves an appropriate affirmation. For example:
* IF user selects "confidence" → THEN respond with "I am strong, capable, and confident."
* IF user selects "gratitude" → THEN respond with "I appreciate the good things in my life."

1. **Astrology Personality Predictor**

* **What it does:** This system provides personality insights based on the user's zodiac sign.
* **How it works using rules:** The system matches each zodiac sign with a predefined personality description. For example:
* IF user enters "Leo" → THEN respond with "You are bold, charismatic, and natural leaders."
* IF user enters "Pisces" → THEN respond with "You are empathetic, creative, and deeply intuitive."

**Chosen Project: Mental Health Support Chatbot**

I decided to go with the Mental Health Support Chatbot because it offers a simple yet effective way to provide emotional support. Mental health is an important topic, and a chatbot with predefined supportive responses could be a small but meaningful way to help those in need. This project is both practical and impactful, allowing me to explore rule-based AI while creating something useful.

**Part 2: Designing Your Rule-Based System**

Now that I have chosen my project, I need to define the rules and logic that the chatbot will use to provide support. This chatbot will function using keyword-based rules, where specific words or phrases trigger predefined responses. Below is a structured rule set:

**Rule Set for the Mental Health Support Chatbot:**

Stress Management:

IF user input contains "stressed," "overwhelmed," or "burned out" → THEN respond with:

"I'm sorry you're feeling this way. Have you tried deep breathing exercises?"

"Stress can be overwhelming. Taking a short break or a walk might help."

Sadness and Emotional Support:

IF user input contains "sad," "down," or "depressed" → THEN respond with:

"I'm here for you. Talking to a friend or journaling might help."

"It's okay to feel sad. Sometimes, expressing your feelings through art or music can be helpful."

Anxiety and Grounding Techniques:

IF user input contains "anxious," "worried," or "nervous" → THEN respond with:

"Anxiety can be tough. Try grounding techniques like naming five things you can see."

"You’re not alone. Deep breathing and mindfulness exercises might help."

Loneliness and Social Connection:

IF user input contains "lonely" or "alone" → THEN respond with:

"Loneliness can be tough. Connecting with someone you trust might help."

"Maybe a call with a friend or joining an online community could help you feel more connected."

Lack of Motivation:

IF user input contains "unmotivated," "lazy," or "no energy" → THEN respond with:

"It’s okay to feel unmotivated sometimes. Starting with a small step can help build momentum."

"Try setting a tiny goal for today. Even a small win can give you a boost."

Sleep Issues:

IF user input contains "can't sleep," "insomnia," or "trouble sleeping" → THEN respond with:

"A calming bedtime routine, like reading or meditation, could help."

"Have you tried limiting screen time before bed? That can make a big difference."

Default Response (Fallback Mechanism):

IF the user input does not match any predefined keywords → THEN respond with:

"I'm here to listen. Can you tell me more about how you're feeling?"

"I might not have all the answers, but I’m happy to chat if it helps."

"That sounds difficult. Would you like some general coping strategies?"

"Sometimes just talking about things can make a difference. Want to tell me more?"

"You're not alone. Talking about it might help."

**Part 3: Coding Your Rule-Based System**

I developed the Mental Health Support Chatbot using Python, implementing a rule-based system that recognizes specific emotional keywords (e.g., "stressed," "anxious," "sad") and responds with predefined supportive messages. To enhance engagement, I incorporated varied responses and follow-up questions, making the conversation feel more natural and human-like. The chatbot was thoroughly tested with different inputs to ensure it accurately detects emotional triggers and provides meaningful support. The screenshots below showcase how the chatbot responds to users experiencing different emotions.

In this interaction, the user expresses feeling "kind of sad." The chatbot detects the keyword "sad" and responds with an encouraging message suggesting journaling or talking to a friend for support.

A screen shot of a computer

AI-generated content may be incorrect.

The user states "I'm feeling super anxious." The chatbot recognizes the word "anxious" and provides a relevant coping strategy, suggesting a grounding technique to help manage anxiety.

A screen shot of a computer

AI-generated content may be incorrect.

The user shares "I'm so overwhelmed right now." Since "overwhelmed" is not explicitly listed as a keyword in the chatbot’s rules, it defaults to a general supportive response, reassuring the user that they are not alone and encouraging conversation.

A screen shot of a computer screen

AI-generated content may be incorrect.

**Part 4: Reflection**

The Mental Health Support Chatbot was built using Python and follows a rule-based approach, where it detects specific emotional keywords and provides predefined supportive responses. It was designed to offer encouragement and coping strategies for users experiencing emotions like stress, anxiety, or sadness. To enhance user engagement, I included varied responses and follow-up questions that make conversations feel more natural and interactive.

One of the key challenges I faced was ensuring that the chatbot recognized variations in phrasing (e.g., "I'm feeling a little sad" vs. "I'm sad"). Initially, the chatbot only responded when an exact keyword was found, so I had to refine its logic to recognize similar phrases. Additionally, I wanted the chatbot to feel less robotic, so I incorporated randomized responses to add variety and warmth to interactions.

Overall, this project deepened my understanding of early AI systems and how pre-machine learning chatbots functioned. It highlighted the limitations of rule-based AI but also demonstrated how simple logic can still create meaningful user interactions. If I were to expand on this, I would explore adding sentiment analysis or a broader set of conversational patterns to improve accuracy and engagement.