Task: Multi-Agentic Real Estate Chatbot (Image + Text Based)

Reply to:

- saksham@data-hat.com
- hiring@data-hat.com

PLEASE PUT YOUR NAME and email AT THE TOP OF THE SUBMISSION

Project Title: Multi-Agent Real Estate Assistant Chatbot (Text + Image Enabled)

Objective: Design a **multi-agentic chatbot system** capable of solving real estate-related issues with a specialization split into two distinct virtual agents:

Agent 1: Issue Detection & Troubleshooting Agent (Image + Text)

Responsibilities:

- Accepts user-uploaded images of properties, along with optional textual context.
- Detects visible issues in the property (e.g., water damage, mold, cracks, poor lighting, broken fixtures).
- Provides troubleshooting suggestions, such as:
    - "You might need to contact a plumber."
    - "This looks like paint peeling due to moisture—consider using the anti-damp coating."
- Can ask clarifying follow-up questions to diagnose better.

Example Interaction:

User: "What's wrong with this wall?" (User uploads image)

Agent 1: "It appears there is mould growth near the ceiling. This might be due to high humidity or a leak. I recommend checking for water seepage and using a dehumidifier."

Agent 2: Tenancy FAQ Agent (Text-based)

Responsibilities:

- Handles frequently asked questions related to tenancy laws, agreements, landlord/tenant responsibilities, and rental processes.
- Capable of giving location-specific guidance if the user's city or country is provided.
- Can answer common questions like:
    - "How much notice do I need to give before vacating?"
    - "Can my landlord increase rent midway through the contract?"
    - "What to do if the landlord is not returning the deposit?"

Example Interaction:

User: "Can my landlord evict me without notice?"

Agent 2: "In most jurisdictions, landlords must give written notice before eviction, unless it's an emergency like non-payment or illegal activity.

Please let me know your city or region for more accurate info."

Multi-Agent Architecture (Design Suggestion)

- Agent Router/Manager: Automatically identifies which agent should respond based on:
    - Text classification (FAQ vs issue description)
    - Presence of an image
- Fallback: If unclear, the chatbot should ask a clarifying question to route to the right

Agent.

Tools & Platforms Allowed

You may use any combination of the following:

- AI platforms like OpenAI Assistants, GPTs, Langchain, or HuggingFace
- Image analysis APIs like Google Vision API, Azure Cognitive Services, or CLIP / BLIP models
- Optionally host the agents on a shared UI or landing page with switchable modes

Deliverables

1. 1. Working chatbot demo (video, screenshots, or live link) and private github repository link (access given to saksham@data-hat.com).
2. 2. Short README or documentation (1-2 pages) describing:

a. Which tools/tech were used
b. The logic behind agent switching
c. How image-based issue detection works
d. Use case examples covered