

Task 10 Hostel Information Chatbot

Description:

Task 10 is a question-answer (QnA) chatbot that responds with basic details regarding a hostel, including types of rooms, fees, facilities, rules, place, and contact information.

File: chatbot.py

Method :get_response function

Parameters:

1. **user_input** (str):
 - The message or query provided by the user. It is expected to be a string.
 - The function converts the input to lowercase for case-insensitive comparison with predefined keywords.
2. **last_topic** (str, optional):
 - The last identified topic from a prior interaction, which is optionally passed to maintain context. If no topic is found for the current user input, this topic is used as a fallback.

Returns:

- **response** (str):
 - A string response corresponding to the identified topic. If no topic is identified, a default message is returned.
- **topic** (str or None):
 - The topic related to the response (e.g., "rooms", "fees", etc.). If no topic is found, None is returned.

Logic:

1. **Keyword Matching:**
 - The function first converts the `user_input` to lowercase to make the comparison case-insensitive.
 - It then checks if any keyword in the predefined `keywords` dictionary matches any word in the `user_input`.
 - If a match is found, the associated topic (e.g., "rooms", "fees") is assigned to the `topic` variable.
2. **Fallback to Last Topic:**
 - If no topic is found and a `last_topic` is provided, it defaults to using the `last_topic` value.
3. **Response Retrieval:**
 - If a topic is identified, the function retrieves the corresponding response from the `responses` dictionary.

- If no response is found for the identified topic, a default response indicating "the information for this topic is not available" is returned.
- 4. **No Topic Identified:**
 - If no topic is identified in the user input and no `last_topic` is available, the function returns a default message prompting the user to ask about available topics such as rooms, fees, facilities, rules, location, or contact info.

Error Handling:

- The function uses `responses.get(topic, "Sorry, the information for this topic is not available.")` to ensure that if a topic exists but has no response associated with it, a default error message is returned.
- If no topic is identified, the user is given instructions to ask about available topics (rooms, fees, facilities, etc.).

File: app.py

Functionality:

Creates a Flask app with two routes:

`/` → serves the chatbot UI (`index.html`).

`/chat` → processes user input and returns the bot's response in JSON form.

Uses `get_response()` from `chatbot.py` to compute replies.

Template: templates/index.html

A basic HTML page with:

An input field for user messages. A chat log to output conversation. JavaScript to manage user interaction using AJAX.

Output:

AI Chatbot

rooms

We offer single, double, and triple rooms with attached bathrooms, study tables, and Wi-Fi.

internet

We offer single, double, and triple rooms with attached bathrooms, study tables, and Wi-Fi.

rules

Rules: No smoking or loud music. Visitors allowed 10 AM - 6 PM. Entry after 10 PM requires approval.

fees

Fees: Single - Rs. 8000/month, Double - Rs. 6500/month, Triple - Rs. 5000/month. Security: Rs. 5000.

location

We're located at Sector 12, near XYZ University, Sher Shah Town.

Ask me anything...

Send