

Chatbot Deployment with IBM Cloud Watson Assistant

Team Members:

1. Akshaya.E – 311521106006
2. Harini.B– 311521106033
3. Mageswari.M– 311521106057
4. Jayashree Priya.L– 311521106043

Problem Statement: Create a helpful virtual guide using IBM Cloud Watson Assistant. Customize the chatbot to assist users on popular messaging platforms like Facebook Messenger and Slack. Provide useful information, answer FAQs, and offer a friendly conversational experience. Empower users with quick access to information and create meaningful connections through your virtual guide!

Project Steps

Phase 1: Problem Definition and Design Thinking

Problem Definition: The project involves creating a chatbot using IBM Cloud Watson Assistant. The goal is to develop a virtual guide that assists users on messaging platforms like Facebook Messenger and Slack. The chatbot should provide helpful information, answer frequently asked questions (FAQs), and offer a friendly conversational experience. The project includes designing the chatbot's persona, configuring responses, integrating with messaging platforms, and ensuring a seamless user experience.

Design Thinking:

1. **Persona Design:** Define the chatbot's persona, including its name, tone, and style of communication.
2. **User Scenarios:** Identify common user scenarios and FAQs that the chatbot should be able to address.
3. **Conversation Flow:** Design the conversation flow, outlining how the chatbot responds to user queries and prompts.
4. **Response Configuration:** Configure the chatbot's responses using Watson Assistant's intents, entities, and dialog nodes.
5. **Platform Integration:** Integrate the chatbot with popular messaging platforms like Facebook Messenger and Slack.
6. **User Experience:** Ensure a seamless and user-friendly experience, with clear prompts and informative responses.

Phase 2: Innovation

Consider implementing advanced features such as natural language understanding (NLU) for more accurate user intent recognition.

Step 1: Create a New Chatbot

GO BACK LEARN MORE

Create New Chatbot

Progress: Add Bot Knowledge | Basic Bot Setup | Create Your Bot

Choose your data source

☒ Website ☐ SiteMap ☐ GitBook ☐ PDF ☐ Text ☐ FAQs ☐ Catalog

Transform Your Website into Interactive Dialogues

Type your website URL here. For example, <https://your-website.com> BEGIN CRAWLING

Included Resources: 0 Link, 0 PDF NEXT

Step 2: Name your Chatbot and frame a URL

GO BACK LEARN MORE

Create New Chatbot

Progress: Add Bot Knowledge | **Basic Bot Setup** | Create Your Bot

Choose your data source

☒ Website ☐ SiteMap ☐ GitBook ☐ PDF ☐ Text ☐ FAQs ☐ Catalog

Transform Your Website into Interactive Dialogues

BEGIN CRAWLING

Included Resources: 0 Link, 0 PDF NEXT

Step 3: Setup your Bot

The screenshot shows the 'Create New Chatbot' interface in a web browser. The browser tabs include '(1) WhatsApp' and 'Onboarding'. The address bar shows 'https://orimon.ai/app/onboarding'. The page has a purple header with a logo and version '4.3.3', and navigation icons. The main content area is titled 'Create New Chatbot' and features a progress bar with three steps: 'Add Bot Knowledge', 'Basic Bot Setup' (which is the current step), and 'Create Your Bot'. Below the progress bar, there are three input fields: '*Bot Name' with the value 'leo', '*Company Name' with the value 'tourism', and 'Language' with a dropdown menu set to 'English'. At the bottom left, it says 'Included Resources: 1 Link, 0 PDF'. A green 'CREATE' button is at the bottom right. The Windows taskbar at the bottom shows the search bar, task view, and various application icons, along with system information like '29°C Partly cloudy' and the date '11-10-2023'.

1 (1) WhatsApp x Onboarding x +
https://orimon.ai/app/onboarding

Create New Chatbot

Add Bot Knowledge Basic Bot Setup Create Your Bot

*Bot Name
leo

*Company Name
tourism

Language
English

Included Resources: 1 Link, 0 PDF

CREATE

Type here to search 29°C Partly cloudy Meet Now ENG 20:52 11-10-2023

Step 4: Test your AI and set it.

The screenshot shows the 'Test Bot' interface in a web browser. The browser tabs include '(2) WhatsApp' and 'Test Bot'. The address bar shows 'https://orimon.ai/app/tenant/0a0d90ce-c763-4388-bce1-1be51d650803/test-bot'. The page has a purple header with a logo and version '4.3.3', and navigation icons. The main content area is titled 'Test Bot' and features a sidebar on the left with options: 'Spread the bot' (with 'Invite your team' and 'Share bot URL'), and 'Launch your bot' (with 'Add on Website'). The main area shows a chat window for 'leo online'. The chat window has a purple header with the bot's name and a close button. The chat history shows a message: 'what is marina beach'. The response from the bot is: 'Marina Beach is a popular tourist attraction in Chennai, India. It is a natural urban beach along the Bay of Bengal and stretches for 13 kilometers. It is known for its scenic views, street food, and various activities such as horseback riding and kite flying.' Below the chat window, there are buttons for 'COMPANY SERVICES' and 'HOW DOES YOUR PRODUCT COMPARE TO SIMILAR PRODUCTS IN THE MARKET?'. At the bottom, there is a text input field labeled 'Enter your Query...'. The Windows taskbar at the bottom shows the search bar, task view, and various application icons, along with system information like '29°C Partly cloudy' and the date '11 October 2023 Wednesday'.

2 (2) WhatsApp x Test Bot x +
https://orimon.ai/app/tenant/0a0d90ce-c763-4388-bce1-1be51d650803/test-bot

Enjoying Orimon AI? ★ Upgrade from your Always Free Plan to unlock more messages, chatbots, and exclusive features! Dive deeper into the world of chatbots with us. SUBSCRIBE

4.3.3

Spread the bot

Invite your team

Share bot URL

https://bot.orimon.ai/deploy/index.html?tenantId=0a0d90ce-c763-4388-bce1-1be51d650803&testBot=true&defaultOpen=true

You can share this link to your team for testing purposes

Launch your bot

Add on Website

leo online

what is marina beach

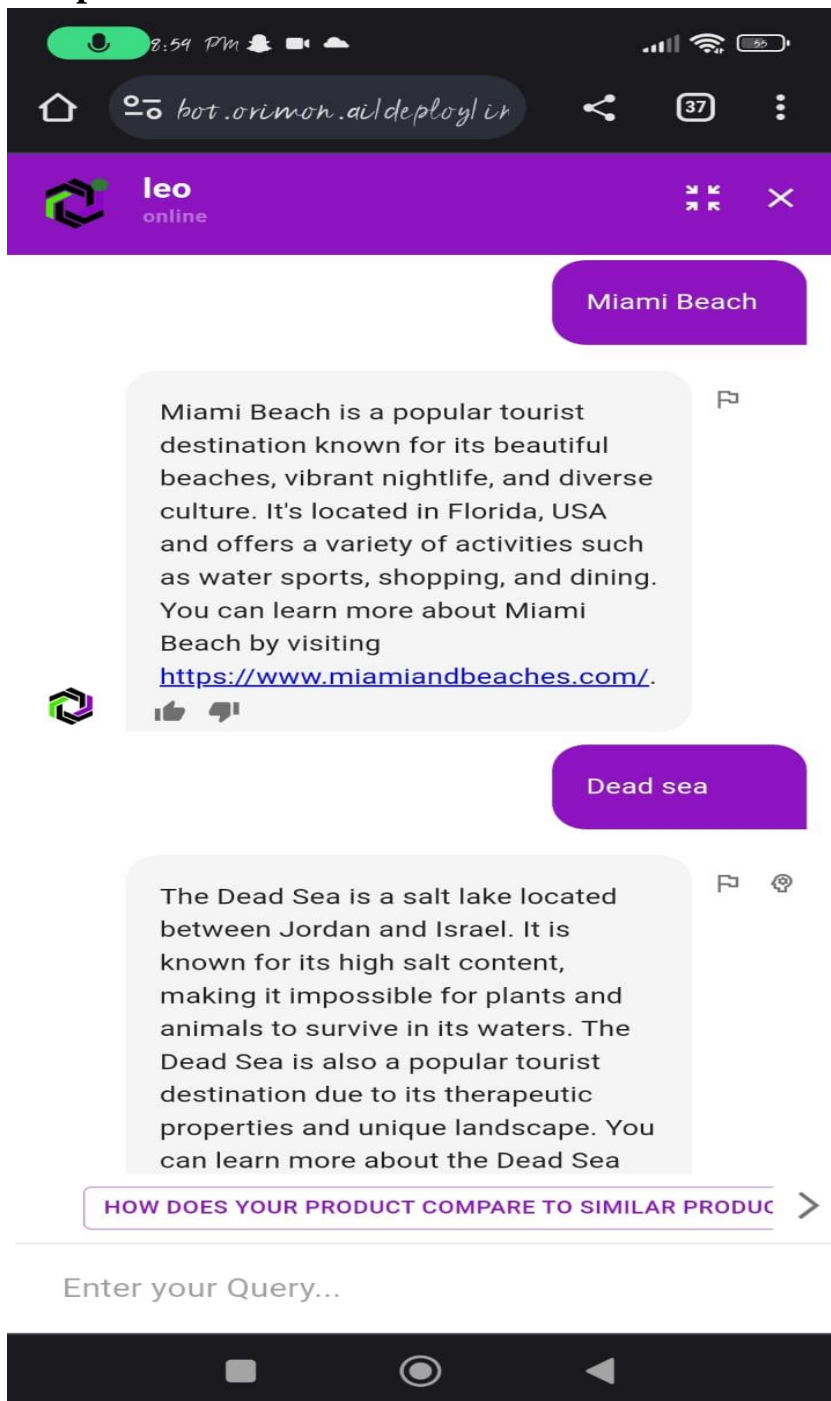
Marina Beach is a popular tourist attraction in Chennai, India. It is a natural urban beach along the Bay of Bengal and stretches for 13 kilometers. It is known for its scenic views, street food, and various activities such as horseback riding and kite flying.

COMPANY SERVICES HOW DOES YOUR PRODUCT COMPARE TO SIMILAR PRODUCTS IN THE MARKET?

Enter your Query...

11 October 2023 Wednesday 21:00 29°C Partly cloudy 11-10-2023

Sample user:



AI link/URL:

<https://bot.orimon.ai/deploy/index.html?tenantId=0a0d90ce-c763-4388-bce1-1be51d650803&testBot=true&defaultOpen=true>

Source Code:

```
import json
from ibm_watson import AssistantV2
from ibm_cloud_sdk_core.authenticators import IAMAuthenticator

# Set up your Watson Assistant credentials
apikey = "YOUR_API_KEY"
url = "YOUR_ASSISTANT_URL"
assistant_id = "YOUR_ASSISTANT_ID"

# Authenticate with Watson Assistant
authenticator = IAMAuthenticator(apikey)
assistant = AssistantV2(
    version='2021-06-14',
    authenticator=authenticator
)
assistant.set_service_url(url)

# Create a session
response = assistant.create_session(assistant_id=assistant_id)
session_id = response.get_result()['session_id']

# Define a function to send a message to Watson Assistant
def send_message(message):
    response = assistant.message(
        assistant_id=assistant_id,
        session_id=session_id,
        input={
            'message_type': 'text',
            'text': message
        }
    )
    return response.get_result()

# Interaction with the chatbot
user_input = "Find hotels in Paris"
response = send_message(user_input)

# Extract and display Watson Assistant's response
```

```

bot_response = response['output']['generic'][0]['text']
print("Chatbot: " + bot_response)

# Remember to end the session when you're done
assistant.delete_session(assistant_id=assistant_id, session_id=session_id)
from flask import Flask, render_template

from watson_developer_cloud import AssistantV2

app = Flask(__name__)

assistant = AssistantV2(
    iam_apikey='YOUR_API_KEY',
    url='https://{region}.watsonconversation.com/assistant/api'
)

deployment_id = assistant.deploy_chatbot(
    assistant_id='YOUR_ASSISTANT_ID'
)['deployment_id']

@app.route('/chatbot')
def chatbot():
    embed_code = assistant.generate_chatbot_embed_code(
        deployment_id=deployment_id
    )
    return render_template('chatbot.html', embed_code=embed_code)

if __name__ == '__main__':
    app.run(debug=True)

```

Briefing:

To deploy this chatbot, you will need to:

1. Create a Watson Assistant service instance on IBM Cloud.
2. Obtain a Watson Assistant API key.
3. Update the `watson_assistant_url` and `watson_assistant_api_key` variables in the source code with your own values.
4. Save the source code as a Python file (e.g. `tourism.py`).

5. Run the following command to deploy the chatbot:
6. `python tourism_chatbot.py`
7. This will start a conversation with the chatbot. You can type in messages and the chatbot will respond.

Here are some additional tips for deploying a chatbot with IBM Cloud Watson Assistant:

- You can use the Watson Assistant web UI to train your chatbot on your data.
- You can use Watson Assistant integrations to connect your chatbot to other services, such as Slack or Facebook Messenger.
- You can monitor the performance of your chatbot using the Watson Assistant analytics dashboard.

Conclusion:

To conclude, deploying a tourism chatbot with IBM Cloud Watson Assistant is a relatively straightforward process. By following the steps outlined above, you can create a chatbot that can answer your users' questions about places and provide them with information about different places to visit and explore.