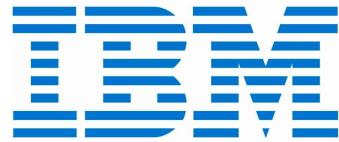




COURSE RECOMMENDATION BOT



VIT[®]
BHOPAL



Name: Anushree Chatterjee

Roll Number: 23BCE11160

College Name: VIT BHOPAL

Branch: CSE CORE

Date of Submission: [07/07/2025]

Table of Contents

S.No.	Section Title
1	Introduction
2	Objective
3	Tools & Technologies Used
4	Methodology / Working
5	Code Snippets
6	Output Screenshots
7	GitHub Link
8	Challenges & Solutions
9	Conclusion
10	References

1. Introduction

The **Course Recommendation Bot** is an interactive virtual assistant designed to help users discover learning paths, courses, and certifications tailored to their interests and backgrounds. By combining IBM Watson Assistant's conversational capabilities with IBM watsonx.ai's powerful generative models, the bot provides personalized recommendations and guidance in real time. This solution aims to simplify course discovery and empower users to upskill efficiently in areas relevant to their goals.

2. Objective

Provide personalized course and certification recommendations based on user interests and background, enable intuitive natural conversations through Watson Assistant integrated with watsonx.ai's generative capabilities, and enhance learner engagement by simplifying the discovery of relevant learning paths.

3. Tools & Technologies Used

- **IBM Watson Assistant** – For building and managing the conversational flow, handling intents, entities, and dialog nodes.
- **IBM watsonx.ai** – For generative AI capabilities, enabling the assistant to craft dynamic, personalized course recommendations.
- **watsonx.ai Prompt Lab** – For prototyping and refining prompts before integrating them into the Assistant.
- **Watson Assistant Extensions** – For seamless integration with watsonx.ai, allowing the assistant to call generative models directly within conversation steps.
- **IBM Cloud** – As the deployment platform for both Watson Assistant and watsonx.ai services.
- **JSON & REST APIs** – For handling communication between the Assistant and watsonx.ai endpoints (internally managed through Extensions).
- **Web-based Chat Interface** – The assistant can be embedded into websites, portals, or learning platforms for user interaction.

4. Methodology / Working

1 User Interaction and Query Collection

Users engage with the Course Recommendation Bot through a chat interface embedded on a website or platform. They can type questions like "Which certifications should I take for cloud computing?" or "Suggest beginner courses in data science." The chatbot's conversational design allows for free-form, natural language queries, encouraging open-ended interactions.

2 Intent Recognition and Context Management

IBM Watson Assistant analyzes the user's message, identifies their intent (e.g., looking for a course, exploring career options, or learning about certifications), and maintains context across multiple turns in the conversation. Context variables are set to capture details like user interests, current skill level, or previous topics discussed, enabling personalized dialog continuity.

3 Integration with watsonx.ai for Dynamic Responses

When a recommendation or detailed response is needed, Watson Assistant invokes a watsonx.ai extension. This extension sends the user's question and captured context to a generative AI model deployed in watsonx.ai (e.g., granite-3-3-8b-instruct). The model processes the input and generates context-aware suggestions tailored to the user's background and expressed interests.

4 Processing and Response Handling

The Assistant receives the watsonx.ai response, extracts the generated text from the returned JSON, and maps it into a user-friendly reply. The Assistant's dialog design includes conditional logic to check for successful responses and handle errors gracefully, ensuring a smooth user experience even if the AI service encounters issues.

5 Interactive Follow-up and Conversation Flow

After delivering recommendations, the Assistant engages the user by offering options to ask more questions, narrow recommendations, or explore related topics. The bot continues to collect additional information if needed, refines suggestions, and provides ongoing support, maintaining an interactive, helpful dialog.

[Start]

|

[User enters query in chatbot]

|

[Watson Assistant identifies intent + context]

|

[Send query + context to watsonx.ai model (via Extension)]

|

[watsonx.ai model generates personalized recommendations]

|

[Assistant receives and processes response]

|

[Chatbot displays recommendations to user]

|

[Does user want more info?]--
Yes-->[Return to "User enters query"]

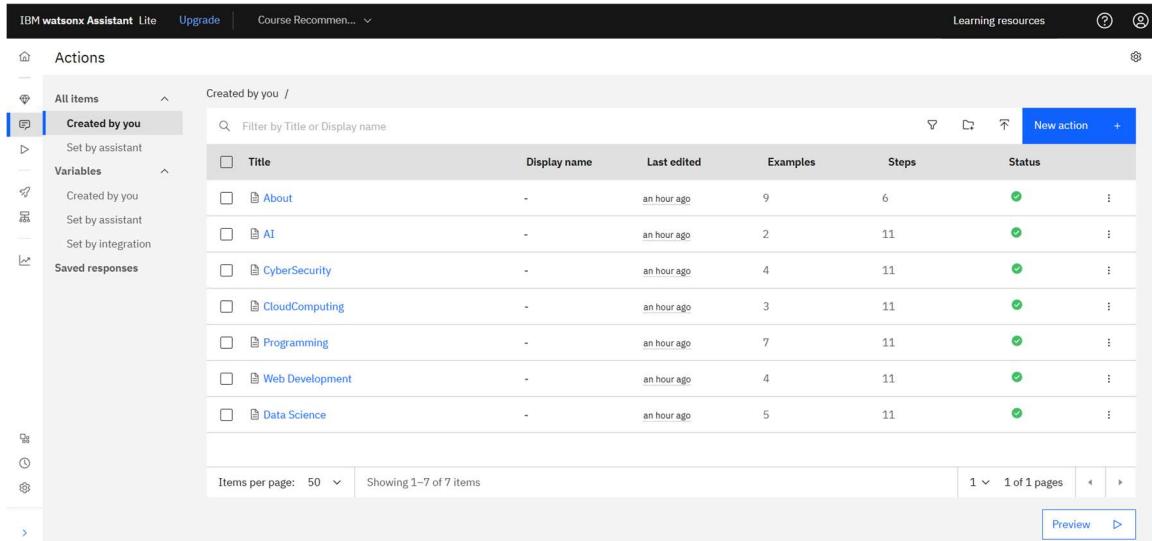
|

No

|

[End]

5. Code / How it works



The screenshot shows the IBM Watson Assistant Lite interface. The left sidebar has a tree view with 'Actions' selected, expanded to show 'All items', 'Created by you' (which is selected and highlighted in blue), 'Set by assistant', 'Variables', and 'Saved responses'. Under 'Created by you', there are sub-folders for 'Set by assistant', 'Set by integration', and 'Saved responses'. The main content area displays a table titled 'Created by you /' with the following data:

Title	Display name	Last edited	Examples	Steps	Status	More
About	-	an hour ago	9	6	Green checkmark	⋮
AI	-	an hour ago	2	11	Green checkmark	⋮
CyberSecurity	-	an hour ago	4	11	Green checkmark	⋮
CloudComputing	-	an hour ago	3	11	Green checkmark	⋮
Programming	-	an hour ago	7	11	Green checkmark	⋮
Web Development	-	an hour ago	4	11	Green checkmark	⋮
Data Science	-	an hour ago	5	11	Green checkmark	⋮

At the bottom, there are buttons for 'Items per page: 50', 'Showing 1–7 of 7 items', '1 of 1 pages', and 'Preview'.

Figure: Configured Actions in IBM Watson Assistant for Domain-Specific Recommendations

- Displays the Actions workspace where conversation flows are organized by domain, including AI, Cybersecurity, Cloud Computing, Programming, Web Development, and Data Science.
- Each action contains examples and multi-step dialog logic to handle queries related to its specific topic effectively.
- Actions are designed to recognize diverse user intents, capture context, and seamlessly integrate with watsonx.ai for generating dynamic, personalized course recommendations.
- This modular approach enables the chatbot to provide tailored guidance across multiple domains while maintaining a coherent and user-friendly conversation experience.

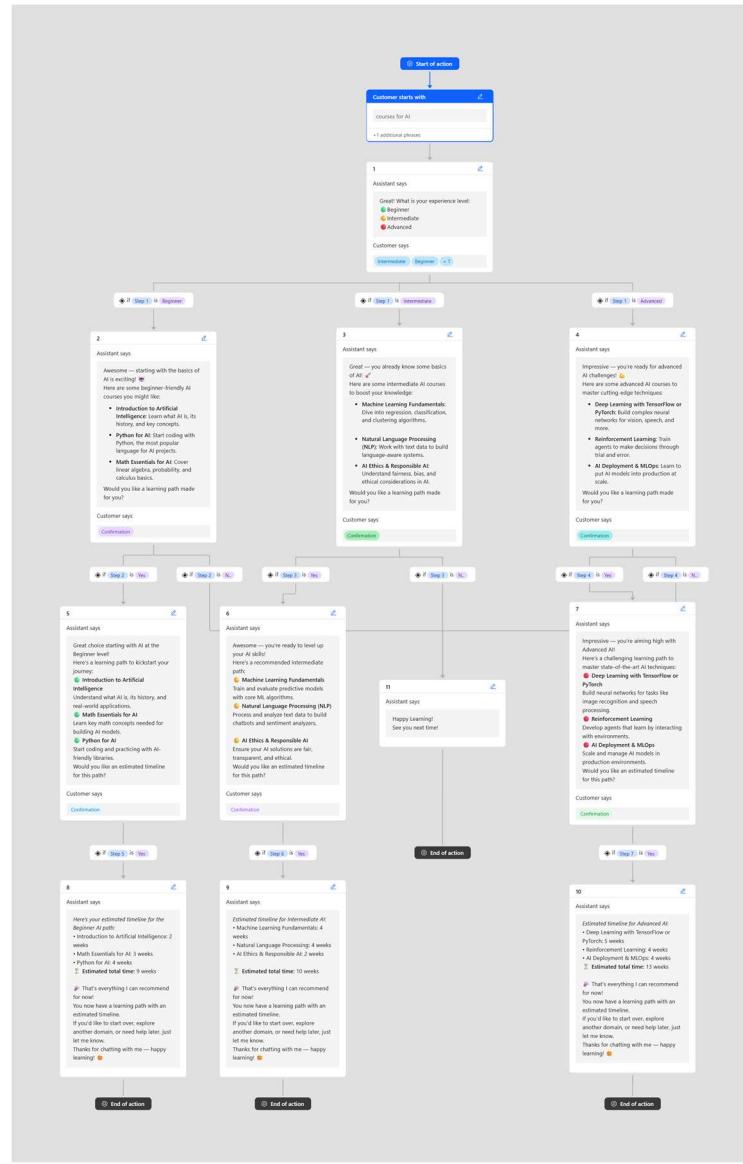


Figure: Conversation Flow for AI Course Recommendations in the Chatbot

- Visualizes the step-by-step dialog structure for recommending AI courses based on the user's experience level: Beginner, Intermediate, or Advanced.
- Starts by asking the user for their AI knowledge level, then branches into personalized learning paths with tailored course suggestions for each level.
- Each branch provides detailed course recommendations, an estimated timeline, and follow-up options to confirm the learning path or end the conversation.
- Highlights the chatbot's ability to dynamically adapt responses based on user input, ensuring an engaging and customized experience aligned with user needs.



The top screenshot shows the 'Assets' section of the IBM WatsonX project. It displays a list of assets under 'All assets', with one asset named 'Courses' selected. The bottom screenshot shows the detailed setup of the 'Courses' prompt session. It includes sections for 'Instruction (optional)', 'Examples (optional)', and a table showing input prompts and their corresponding AI-generated outputs for topics such as Artificial Intelligence, Cybersecurity, Cloud Computing, Web Development, and Data Science.

Figure: watsonx.ai Project Assets and Prompt Session for Technology Domain Explanations

- The first image shows the Assets section of the IBM watsonx.ai project, highlighting a prompt session asset named Courses, which is used to power the chatbot's responses for various technology topics.
- The second image displays the detailed setup of the prompt session, including instructions that guide the model to act as a helpful assistant providing clear, concise explanations of technology domains such as AI, Cybersecurity, Cloud Computing, Web Development, and Data Science.
- Example input-output pairs demonstrate how the model responds with informative overviews and key points when users ask about different technology topics, ensuring accurate, context-aware answers during conversations.

The figure consists of two screenshots of the IBM Watson Assistant interface. The top screenshot shows the 'Extensions' page, where users can add features or content to their assistant using custom or pre-built extensions. It lists three available integrations: 'Search' (represented by a magnifying glass icon), 'Segment' (represented by a bar chart icon), and 'watsonx' (represented by a person icon). The 'watsonx' integration is selected, showing its details: 'Get a better understanding of your users' end-to-end journeys by combining your assistant's data with other sources.' Below this are 'Learn more' and 'Open' buttons. The bottom screenshot shows the 'Integration setup' for the 'watsonx' extension. It displays a configuration dialog with several 'Set' and 'To' fields. The 'Set' fields include 'Tr input' (To: 'Tr prompt_data .append(1. What topic)'), 'Tr model_id' (To: '@Model_Id'), 'Tr project_id' (To: '@Project Id'), and 'Tr version' (To: '@Version'). There are also sections for 'Operation' (set to 'Generation') and 'Privacy' (with an unchecked checkbox for 'Protect data returned from this extension'). The background shows the main Watson Assistant interface with a conversation history and a 'New step +' button.

Figure: Extensions in IBM Watson Assistant and Integration Setup with watsonx.ai

- The screenshot shows the Extensions page in Watson Assistant, where available integrations like Search, Segment, and watsonx can be added to enhance assistant capabilities.
- By configuring the watsonx extension here, the assistant can call powerful generative AI models to provide personalized, dynamic responses.
- Within the about action, the extension is set up by specifying model parameters, input variables, and response handling, enabling seamless interaction between Watson Assistant's dialog system and watsonx.ai's language model for real-time course recommendations.

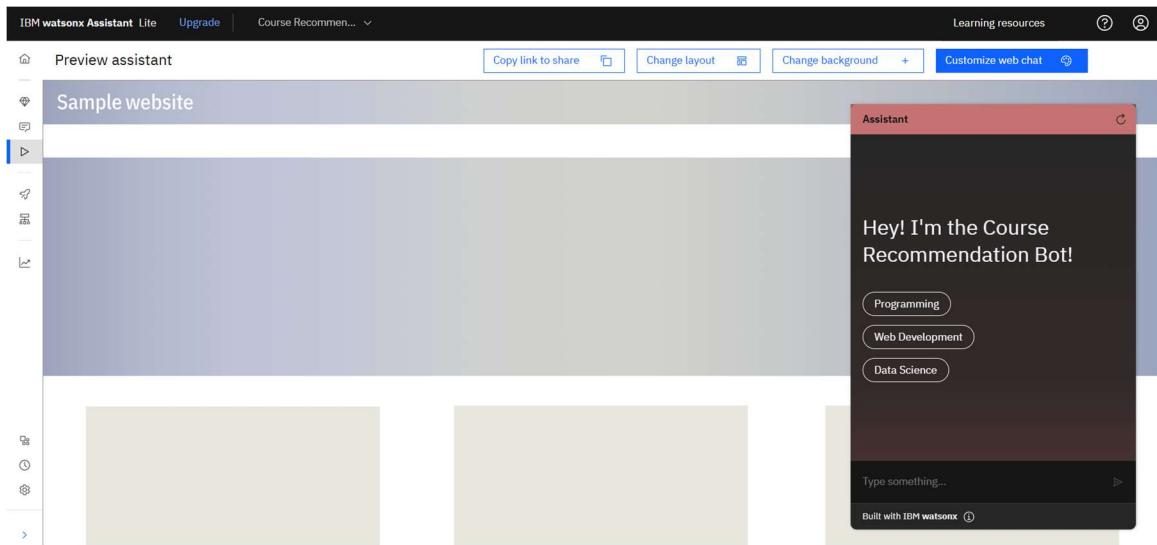


Figure: Preview of the assistant on a sample website



```
<script>
window.watsonAssistantChatOptions = {
  integrationID: "68e913cd-0dbb-4baa-bee3-f2e3995493db9", // The ID of this integration.
  region: "us-south", // The region your integration is hosted in.
  serviceInstanceId: "ddabes57-8260-46f2-b53b-fd065bcb14df", // The ID of your service instance.
  onLoad: async (instance) => { await instance.render(); }
};
setTimeout(function(){
  const t=document.createElement('script');
  t.src="https://web-chat.global.assistant.watson.appdomain.cloud/versions/" + (window.watsonAssistantChatOptions.clientVersion || 'latest') + "/WatsonAssistantChatEntry.js";
  document.head.appendChild(t);
});
</script>
```

Figure: Embedding Watson Assistant Chatbot in a Custom Webpage

- This implementation shows how the Watson Assistant chatbot is integrated into a custom HTML webpage using the provided JavaScript snippet.
- The script uses the integration ID, region, and service instance ID to connect the web page to your configured Watson Assistant instance hosted on IBM Cloud.
- Upon loading, the chatbot appears as a floating chat widget on the webpage, allowing users to interact with the AI-powered assistant for personalized course recommendations.
- This webpage serves as a simple front-end interface where users can explore learning suggestions based on their interests and background — making the solution more interactive and publicly accessible.

4. Sample Input and Output

The screenshot shows a web browser window with a pink header bar containing the text "Our Courses" and "Explore our wide range of learning opportunities.". Below the header is a navigation bar with links for "Home", "Courses", "About Us", and "Contact". The main content area is titled "Available Courses" and lists two courses:

- Web Development Fundamentals**: Learn the basics of HTML, CSS, and JavaScript to build responsive web pages. Duration: 8 Weeks | Level: Beginner. Includes a "Learn More" button.
- Python for Data Science**: Master Python programming for data analysis, visualization, and machine learning. Duration: 12 Weeks | Level: Intermediate. Includes a "Learn More" button.

A red circular icon with a white arrow is visible on the right side of the page. A small red speech bubble is also present near the bottom right.



The screenshot shows a web browser window with a pink header bar containing the text "Our Courses" and "Explore our wide range of learning opportunities.". Below the header is a navigation bar with links for "Home", "Courses", "About Us", and "Contact". The main content area is titled "Available Courses" and lists two courses:

- Web Development Fundamentals**: Learn the basics of HTML, CSS, and JavaScript to build responsive web pages. Duration: 8 Weeks | Level: Beginner. Includes a "Learn More" button.
- Python for Data Science**: Master Python programming for data analysis, visualization, and machine learning. Duration: 12 Weeks | Level: Intermediate. Includes a "Learn More" button.

To the right of the course list, there is a dark-colored AI recommendation bot window. The bot has a "Assistant" title at the top. Inside, it says "Hey! I'm the Course Recommendation Bot!" and lists three categories: "Programming", "Web Development", and "Data Science". At the bottom of the bot window is a text input field with the placeholder "Type something...".

Assistant

tell me about courses in cybersecurity

10:26 PM
Great! What is your experience level:
● Beginner
● Intermediate
● Advanced

Beginner Intermediate
Advanced

You 10:26 PM
Beginner

10:26 PM
Awesome — starting with the basics of Cybersecurity is a smart move! 🔒
Here are some beginner-friendly cybersecurity courses you might like:

Type something... ➤

Built with IBM watsonx ⓘ

Assistant

Beginner

10:26 PM
Awesome — starting with the basics of Cybersecurity is a smart move! 🔒
Here are some beginner-friendly cybersecurity courses you might like:

- **Introduction to Cybersecurity:** Learn what cybersecurity is, why it matters, and key concepts.
- **Computer Networks Fundamentals:** Understand how networks work and common vulnerabilities.
- **Security Basics & Best Practices:** Learn how to stay safe online and secure devices.

Would you like a learning path made for you?

Yes No

Type something... ➤

Built with IBM watsonx ⓘ

Assistant

Yes

10:30 PM
Great choice starting with Cybersecurity at the Beginner level!
Here's a learning path to kickstart your journey:
● Introduction to Cybersecurity
Explore the basics of security concepts, threats, and defense mechanisms.
● Computer Networks Fundamentals
Learn how data travels across networks and how attackers exploit weaknesses.
● Security Basics & Best Practices
Get practical tips to protect your data and devices.
Would you like an estimated timeline for this path?

Yes No

Type something... ➤

Built with IBM watsonx ⓘ

Assistant

Now attackers exploit weaknesses.

● Security Basics & Best Practices
Get practical tips to protect your data and devices.
Would you like an estimated timeline for this path?

Yes No

You 10:30 PM
No

10:30 PM
Happy Learning!
See you next time!

Do you want to continue with the previous topic: Web Development?

Yes No

Type something... ➤

Built with IBM watsonx ⓘ

5. GitHub Link

<https://github.com/AnushreeChatterjee2005/Course-Recommendation-Bot>

6. Challenges and Solutions

1. Integration Errors Between Watson Assistant and watsonx.ai

- **Problem:** Initial setup of the watsonx.ai extension showed "Extension error" due to missing or mismatched configuration (e.g., incorrect project ID, model ID, or endpoint).
- **Solution:** Verified and corrected the project_id, model_id, and ensured the watsonx prompt session was correctly added as an asset. Confirmed that the Assistant extension was calling the correct endpoint with valid input and parameters.

2. Structuring Dynamic Conversations Based on User Input

- **Problem:** Creating flexible conversation flows that adapt based on user experience level (Beginner, Intermediate, Advanced) was complex.
- **Solution:** Used **conditional branching** in Watson Assistant Actions and designed clear flowcharts to plan each possible path. Leveraged variable setting and reusability of actions to manage complexity.

3. Ensuring Model Responds with Relevant Course Information

- **Problem:** The model initially responded with vague or off-topic information.
- **Solution:** Designed **structured prompts** in watsonx.ai with clear instructions and example inputs/outputs for each domain. Added guiding context like: "Explain clearly and concisely. Focus on recommended courses."

4. Limited Control with Lite Plan

- **Problem:** Some advanced features like the Search extension and more usage capacity were restricted in the Lite version.
- **Solution:** Optimized the chatbot for minimal calls and reused responses where possible. Considered upgrading to Plus plan for production-level deployment.

7. Conclusion

The Course Recommendation Bot effectively combines the conversational capabilities of **IBM Watson Assistant** with the powerful generative intelligence of **watsonx.ai** to deliver personalized learning path suggestions based on user interests and experience levels. By leveraging structured prompts, dynamic conversation flows, and seamless backend integration, the solution provides accurate, engaging, and informative responses to learners. This project demonstrates how AI-powered chatbots can enhance educational guidance, offering a scalable and interactive alternative to traditional recommendation systems.