|  |  |
| --- | --- |
| Date | 26 October 2023 |
| Team id | NM2023TMID5559 |
| Team Name | .P roj\_227254\_Team\_2 |
| Project Name | Chatbot Deployment with IBM  Cloud Watson Assistant |
| Phase | Four(4) |
| Name | 1. Yuvashree |

**Facebook Messenger Integration:**

**Create a Facebook App:**

Go to the Facebook Developers portal, create an app,

and set up a Messenger product for it.

**Generate Page Access Token:**

Link your app to a Facebook Page and generate a Page Access Token.

**Set Up Webhooks:**

Configure webhooks to receive messages and events from Messenger. You'll need a publicly accessible endpoint (e.g., a web server) to handle incoming requests.

**Verify Callbacks:**

Ensure that your server can verify the authenticity of incoming requests using the app secret.

**Handle Messages:**

Implement logic to process incoming messages, interpret user requests, and generate responses.

**Slack Integration:**

**Create a Slack App:**

Visit the Slack API site and create a new app. Configure it with the necessary permissions and settings.

**Install Your App:**

Install your app to your Slack workspace to obtain an OAuth access token.

**Set Up Event Subscriptions:**

Enable Event Subscriptions for your app and configure it

to receive events from Slack.

Implement the OAuth 2.0 flow to obtain access tokens to interact with Slack's API on behalf of users.

**Handle Events:**

Write code to handle events and messages sent to your app in Slack, interpreting user inputs and generating appropriate responses.

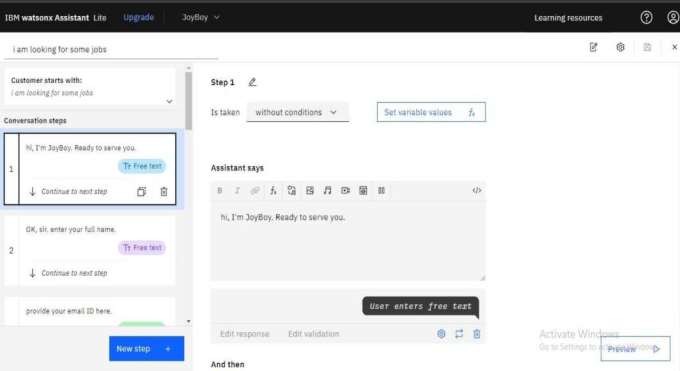
**Interactive Components:**

If you want interactive elements (buttons, menus), you'll need to handle interactions using Slack's interactive components.

In my phase 3 project a basic chatbot has been created using IBM Cloud Watson Assistant. And it has only basic commands which have been integrated into it.

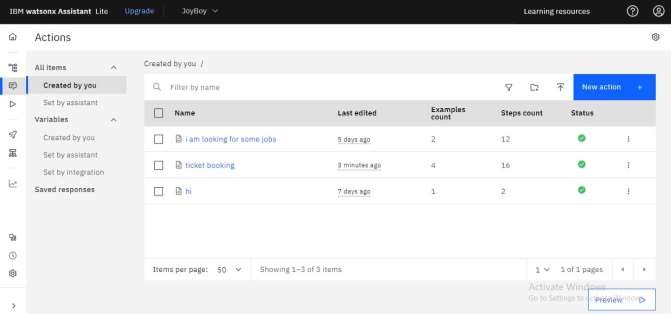
But now in my Phase 4 project the chatbot has been developed a step ahead and integrated some Intents and Entities to it. At the first stage of the chatbot project (i.e phase 3), the command which have been integrated in the command and response page will be the only output from the chatbot but now with the help of the intents and entity the chatbot can understand the user input with the words that is given by the user and it will give the desired response.

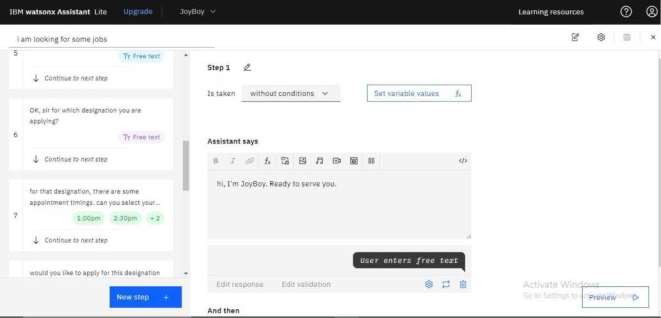
Let's see it.



At first we were only able to make the chatbot function by actions. Which is very complicated since the user should give the whole user input then only the bot can understand the command.

The action functions and the user input and the bot responses both should be integrated.

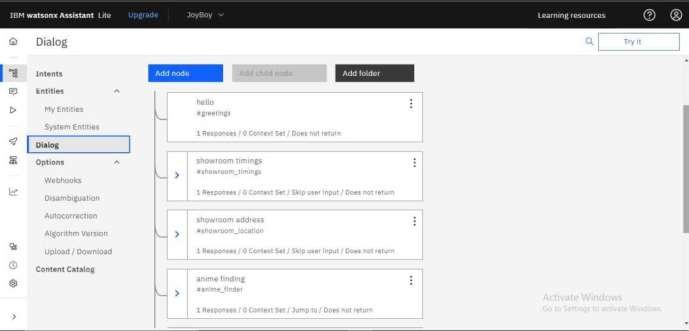




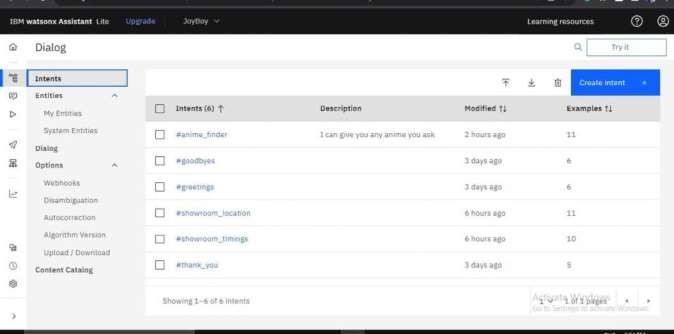
©

And from phase 4, the chatbot can work faster by understanding the user input with the required words alone. With the Intents and Entities the chatbot do require a full sentence instead the chatbot requires only the required word which is already integrated in the entities and intent.

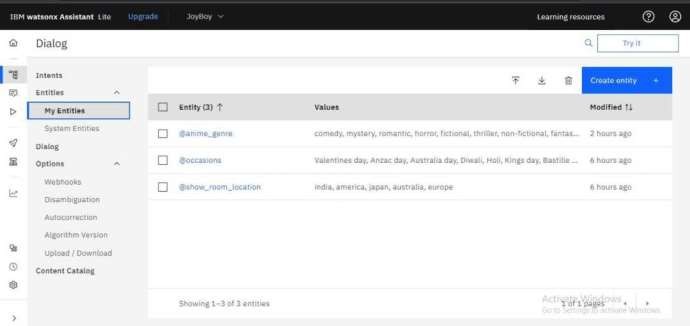
And from the dialog every intent and entity has been integrated with the help of nodes. The work of the node is to integrate the Intent and the Entity.



# The Dialog page containing nodes



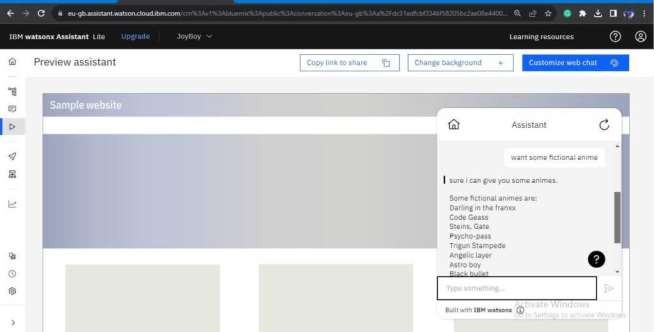
**The Intent page containing Intent:**



# The Entity page containing Entities



**The preview page of the Chatbot:**



With these images we can see the chatbot responses which are obtained from the Intent and Entity. Therefore the chatbot has been developed ahead.

**Chatbot link:**

https://web- chat.global.assistant.watson.appdomain.cloud/previe w.html?backgroundImageURL= https%3A%2F%2Feu- gb.assistant.watson.cloud.ibm.com%2Fpublic%2Fim ages%2Fupx-be804 cb6-e8e8-4d11 -95a9- f0280aad783e%3A%3A5a767323-aa5f-436a-a3ec- cb9ba57d9c69&integra tionID=0e28d050-204c-4473- 8664-6b9011cd9660&region=eu- gb&serviceInstanceID=be804cb6

-e8e8-4d11-95a9-f0280aad783e