

SPS-6414-Movie-ticketing-bot

Movie ticketing bot

hydrabada_Movie_Booking_Bot

This sample chat bot displays theaters list in hydrabada and allows users to book the tickets online..

Simple-Movie-Booking-Chat-Bot-using-Slots

This repository is aimed at helping anyone get started with understanding how to use Watson Assistant service to create a simple chat bot that will book seats to a movie for you. This will be done using a feature called Slots.

Watson Assistant

The [Watson Assistant](<https://www.ibm.com/watson/ai-assistant/>) service available as a Platform as a Service (PaaS) on IBM Cloud provides a AI tooling that can easily allow creating conversational solutions that fits one's business needs.

Basic Concepts & Terminologies

Workspace

A workspace in Watson Assistant is a container for all artifacts that define the behaviour of your system, i.e. Chatbot.

Utterance

An utterance or user example is an input that a user provides when prompted, including questions and statements.

Intent

An intent is the purpose expressed by user input, which usually acts as a label for a group of utterances.

For instance, if "Where can I find the gym?" is the question provided by a user, the Watson Assistant service understands that the user's intent is to ask about the location of something (in this case, the gym, which is called the entity).

Entity

An entity is usually a classification of objects aimed to help alert the response to an intent.

Using the same example of the user asking "Where can I find the gym?", the Watson Assistant service understands that the entity being asked about is the gym. The entity could have been something else like the restaurant, to which the Watson Assistant service would have provided a different response, despite the intent being the same.

Context

Context is information gathered from an external source to customize responses.

Response

A response is what the Conversation service returns to the user's utterances based on the detected intent, and entity can be in the form of text or an action like displaying a map.

Dialog

A dialog defines the conversational flow, which is simply a logical flow that determines responses based on a met condition. The dialog flows in a top-to-bottom, left-to-right fashion.

Dialog Node

A dialog node is a single interaction in a conversation that is triggered when a condition is met and provides a response back to the user.

Slots

Slots are considered the easiest way to gather information from users, allowing what usually takes several dialog nodes to be consolidated into a single node.

Process

Sign up on IBM Cloud

An IBM Cloud account - A lite account, which is a free of charge account that doesn't expire, can be created through going to [IBM Cloud](<http://ibm.biz/childrenday>).

Create a Watson Assistant service

1. Select **Catalog** found at the top right of the page.
2. Click on **Watson** from the menu on the left, which you can find under **Platform** services.
3. Select **WWatson Assistant (formerly Conversation)**.

4. Enter the **Service name** or keep the default value and make sure to select your desired **region/location**, **organization**, and **space**.
5. Select **Lite** for the **Plan**, which you can find under **Pricing Plans** and is already selected. Please note you are only allowed one instance of a Lite plan per service.
6. Click on **Create**.

7. You will be taken to the main page of the service. Click on **Launch tool**.

Create a Workspace

1. Scroll down and click on **Create a Workspace** found under **Get started now**
2. Click on **Create** found in the dotted box that is titled **Create a new workspace**.
3. Give your workspace a **Name** and **Description** (optional) based on the purpose of the conversational solution (We will call the workspace *Simple Movie Booking Bot*).
4. Click **Create**. This will open the workspace, where you will define the **Intents**, **Entities** and **Dialog**.

Define Intents

For the intents, we can define 3 main intents that the user will have: greeting the bot (#greeting), thank the bot for accomplishing the task (#thankyou), and booking movie ticket(s) (#book_tickets).

1. Under the tab **Intents**, click on **Add intent**.
2. Enter the **Intent name** and **Description** (optional) and click on **Create intent**.
3. Under **Add user examples**, add the utterances that are expected to be mentioned based on the intent and click on **Add example**. Make sure to add atleast 5 user examples per intent to proper allow the Natural Language Classifier that is embedded in the Watson Assistant service.

Details about the intent #greeting

Details about the intent #thankyou

Details about the intent #book_tickets

Define Entities

For the entities, we can define 3 main entities that the user might use: terms related to the category of movies (@movie), examples of movie names (@movie-name), and example of cinema names found in hydrabada. For each entity, we will be defining a value and the different synonyms that a person might use by which he/she means the same value.

1. Under the tab **Entities**, click on **Add entity** that will be under the tab **My entities** (these are the user defined entities).
2. Enter the **Entity name** and click on **Create entity**.
3. Add the **Value name** and its corresponding **Synonyms** and click on **Add value**. Here, there are restrictions on the number of values and synonyms that can be added.

Details about the entity @movie

Details about the entity @movie-name

Details about the entity

4. Under **System entities**, which are pre-defined entities that can be directly used, enable **sys-date**, **sys-time**, and **sys-number**. These will be used in the dialog to get the date

and time of when the user wishes to see the movie and the number of seat he/she wishes to reserve.

Define Dialog Flow

1. Click on the tab **Dialog**, where you will see to pre-defined nodes: **Welcome** and **Anything else**.

The **Welcome** has a special condition called **welcome** that is triggered when a conversation is started by the system. The **Anything else** node has a special condition called **anything_else** that is triggered when the user input does not match any of the conditions in previous nodes.

2. Click on the **Welcome** node and modify the responses that will determine how the bot first greets the user, as seen in the diagram below.

3. Also, click on **Set to random**, so that the response change everytime the user tries out the bot.

4. Click on **Add node** to add a node under the **Welcome** node.

5. Call the node **Greetings** and set the condition under **If bot recognizes** to **#greeting**. This means that, after the welcoming message, if the bot detects that the user is greeting it, it will respond with whatever is added under **Then respond with:** and **wait for user input**, which is indicated under **And finally**.

6. Create a new node and call it **Thank You**, which is triggered when **#thankyou** is detected. Complete the rest of the details as follows.

7. Click on the **Greetings** node and click on **Add node**.

8. Call the node **Movie Booking Details**. As a name suggests, this is where we will be gathering details about the booking, which will be done through slots.

9. Beside the name of the node we just defined, there is a **Customize** button. Click on it, enable **Slots** and click on **Apply**.

10. Set the triggering condition under ****If bot recognizes:**** to ***#book_tickets*** and fill the conditions to be checked afterwards under ****Then check for:**** as follows:

1) Check for: @movie-name

Save it as: \$moviename

If not present, ask: There are only 2 movies available (Spider-Man: Homecoming and Star Wars:The Last Jedi). Please enter the name of the movie you would like to watch

2) Check for: @hydrabada-cinema

Save it as: \$movielocation

If not present, ask: Please enter the location of your preference in hydrabada

3) Check for: @sys-date

Save it as: \$moviedate

If not present, ask: Please enter preferred date of booking

4) Check for: @sys-time

Save it as: \$movietime

If not present, ask: Please enter preferred time of booking

5) Check for: @sys-number

Save it as: \$seats

If not present, ask: For how many people should I book?

All of this information can be gathered from one user utterance, if provided by the user. Otherwise, the user will be asked about anything that is missing by presenting what is under ****If not present, ask****. The information provided is saved in context variables, which are defined under ****Save it as****, that can be used at any point throughout the conversation.

11. Provide the user with a summary of the information he/she provided through the response under ****Then respond with**** (as seen below).

Try It Out

Next is trying out the flow defined.

1. Click on the **Try it** button found at the top right of the page.
2. Start conversing with the chatbot (an example is given below).

We will notice that the intent for when we mentioned the location and the number of seats is **Irrelevant**. That is because we only mentioned entities, which were labelled correctly. We can adjust the intent and change it from **Irrelevant** to **#book_tickets** based on which the bot will be retrained.

Importing a workspace

Alternatively, instead of going step-by-step, we can import the entire workspace containing all defined intents, entities and the dialog flow.

1. In the main page of the Workspaces, click on the arrow found beside **Create** that is next to **Workspaces** title.
2. Select the file called ***Hydrabada_Movie_Booking_Bot.json***, which can be found in this repository.