MINI PROJECT

ON

“Restaurant Chatbot Web Application using NLP"

Submitted for the degree of

B.E. (Computer Engineering)

BY

Rahul P. Bhere	Roll No: - 02
Dhiraj R. Manje	Roll No: - 21
Rahul K. Pawade	Roll No: - 31

UNDER THE GUIDANCE OF

Prof. Gayatri Naik.

DEPARTMENT OF COMPUTER ENGINEERING

SHIVAJIRAO S. JONDHLE COLLEGE OF ENGINEERING AND TECHNOLOGY,

ASANGAON 421601.

UNIVERSITY OF MUMBAI

2020-21

Table of Contents

Chapter No.	Chapter	Page No.
	Abstract	3
1.	Introduction	4
2.	Aim And Objectives	6
3.	Literature Survey	7
4.	Existing System	8
5.	Problem Statement	9
6.	Scope	9
7.	Proposed System	10
8.	System Architecture	10
9.	Algorithm	10
10.	Source Code	13
11.	Screenshots	18
12.	Hardware & Software Details	26
13.	Applications	26
14.	Conclusion	26

ABSTRACT

Natural language Processing & Artificial intelligence is on its way to a faster human lifestyle in an efficient way. Virtual bots have become handy nowadays by supporting various domains for smooth lives.Chatbots are helping many sectors by providing services in the form of information agents by effectively accomplishing tasks.

In this project, we develop an end to end mobile application capable of managing/placing orders, displaying recommendations, showing the menu, prompting the best deals or collecting the customer feedback using the IBM Watson Assistant. The customer and the order details are stored in the Cloudant DB. Alert is sent when the order is confirmed using the Cloud Messaging. Speech to Text and Text to Speech Services must be used to take the speech input and give out the speech output.

Features

- Using chatbot we can manage users reservations and orders
- We can give food recommendations and display the menu to the users
- We can Promote best deals and offers on that day
- We will store the customer's details and orders in the database
- Chat will send a notification to customers if the order is confirmed
- The chatbot is also useful in Follow up on customer feedback

Services Used for the project development are:

1. IBM CLOUD/IBM CLOUDANT/Watson Assistant /Intent,Entity using NLP
2. Node-Red APP
3. FAST2SMS service

1. INTRODUCTION

1.1 Restaurant Chatbot Application Using NLP

In this project, we develop an end to end mobile application capable of managing/placing orders, displaying recommendations, showing the menu, prompting the best deals or collecting the customer feedback using the IBM Watson Assistant. The customer and the order details are stored in the Cloudant DB. Alert is sent when the order is confirmed using the Cloud Messaging. Speech to Text and Text to Speech Services must be used to take the speech input and give out the speech output.

NLU (NATURAL LANGUAGE UNDERSTANDING)

The NLP process is a core part of the chatbot architecture and process, since it is the foundation for translating the natural human language to structured data.

In order for the chatbot to understand the user's message, it needs to somehow convert the unstructured human language to structured data that computers can interpret. When a user sends a message to the chatbot, it needs to use algorithms to get meaning and context from every sentence to collect data from them. This process is called natural language understanding (NLU), and it's a subset of natural language processing. It consists of interpreting the user's message by extracting important and relevant details from it.

A way to extract the essential parts of a sentence is to differentiate between the entities and the intent. An intent of a sentence is the goal of the statement.

The chatbot basically needs to recognize the entities and intents of the user's messages. In order to do that, we need to build an NLP model for every entity for an intent. For example, we can build an NLP intent model for the chatbot to recognize when a user wants to know the opening hours of a place. We can build an NLP entity model for the chatbot to recognize locations and directions. We can then use these NLP models for the chatbot to offer the opening hours of any place, based on the user's location.

Intents: It is basically the action chatbot should perform when the user say something. For instance, intent can trigger same thing if user types "I want to order a red pair of shoes", "Do you have red shoes? I want to order them" or "Show me some red pair of shoes", all of these user's

text show trigger single command giving users options for Red pair of shoes. Context: When a NLU algorithm analyses a sentence, it does not have the history of the user conversation. It means that if it receives the answer to a question it has just asked, it will not remember the question. For differentiating the phases during the chat conversation, it is state should be stored. It can either be flags like "Ordering Pizza" or parameters like "Restaurant: 'Dominos'". With context, you can easily relate intents with no need to know what the previous question was.

Entities: Entity basically represents a concept in your Chatbot. It might be a payment system in your Ecommerce Chatbot.

Context: When a NLU algorithm analyses a sentence, it does not have the history of the user conversation. It means that if it receives the answer to a question it has just asked, it will not remember the question. For differentiating the phases during the chat conversation, it is state should be stored. It can either be flags like "Ordering Pizza" or parameters like "Restaurant: 'Dominos'". With context, you can easily relate intents with no need to know what the previous question was.

2.AIM AND OBJECTIVE

2.1 Aim

Now day's automation systems are everywhere whether its home, office or any big industry, all are equipped with automation systems. Restaurants/Hotels are also adopting recent automation trends and are installing robots to deliver food and tablets for taking orders. Using these digital menu cards like tablets, customers can easily select the items. This information will be sent to the kitchen of the Restaurant and also displayed on the display.

2.2 Objective

- In this project, a chatbot is developed using Watson assistant. This Chatbot provides following capabilities
- Generates the menu list available in restaurant. .
- Different offer show which available in restaurant..
- Different specials menu show
- After selecting the menu. • Payment mode also available i.e UPI, COD, or Card
- When a link is down, it should be highlighted. This gives an idea of how the routing goes when a link is down.
- The statistics for the bot and the link are to be calculated for every 500 m sec.
- A provision for redrawing the intent and entity is to be provided.

3. LITERATURE SURVEY

Method 1

Paper Name : Design and Implementation of A restaurant Chatbot

Authors Name : Peng Zhang, Raimo Kantola, Zhansong Ma

Explanation:

A bot is a program that automatically completes an action based on specific triggers and algorithms. A chatbot is a computer program that is designed to simulate human conversation. Users communicate with these tools using a chat interface or via voice, just like they would converse with another person. Chatbots interpret the words given to them by a person and provide a pre-set answer. Chatbots, like regular applications, have application layers, databases, conversational user interfaces (CUIs) and APIs. There are 3 common kinds of a chatbot available today: Rule-based chatbots: The most straightforward option, these bots simply provide a pre-defined answer to very specific questions. These bots are great for things like qualifying leads or offering customers an interactive FAQ experience Intelligent chatbots: These intelligent bots use machine learning or “ML” to learn from the user’s requests and information. Intellectual bots are trained to understand specific words and phrases that trigger a reply. They teach themselves over time to understand more questions and deliver better answers AI-powered chatbots: These bots combine the benefits of rule-based bots with the power of intellectually independent programs to solve user problems. They can remember the context of conversations and understand user preferences. These bots use a combination of natural language processing, machine learning, and AI to understand customers. Natural language processing helps the interactions between humans and computers to feel more natural.

4.EXISTING SYSTEM

Artificial intelligence is on its way to a faster human lifestyle in an efficient way. Virtual bots have become handy nowadays by supporting various domains for smooth lives. Chatbots are helping many sectors by providing services in the form of information agents by effectively accomplishing tasks.

There are several limitations in existing system are as given below:

- Credibility and Validation
  - Perfect simulation of reality is impossible
  - Requirements for trusting models Integration of all components Correct behavior
  - Solution approach: Validation
    - Comparing simulation models to testbed implementation
    - Different scenarios
  - Deficiencies: No absolute guarantee, no perfect testbeds, elaborate
- Scalability Limits
  - Finite computational power vs unlimited simulation network
  - Internet-like environment hard to debug

5.PROBLEM STATEMENT

The final crucial methodology for chatbots is to use artificial neural networks. These are solutions that give the bots a way to calculate the response to a question using weighted connections and context in data. With artificial neural networks, each sentence provided to a bot is broken down into different words, and each word is used as an input for the neural network. Over time, the neural network becomes stronger and more advanced, helping the bot to create a more accurate set of responses to common queries.

6.SCOPE

This Routing Simulator model can be made a more realistic one by considering the effects of most of the System parameters. Even though the mathematical model established for efficiency of Subnet yields acceptable results, it believed that an improved model can be generated. This has the potential to be used as one of the tools for experimentation on design and analysis of Subnets.

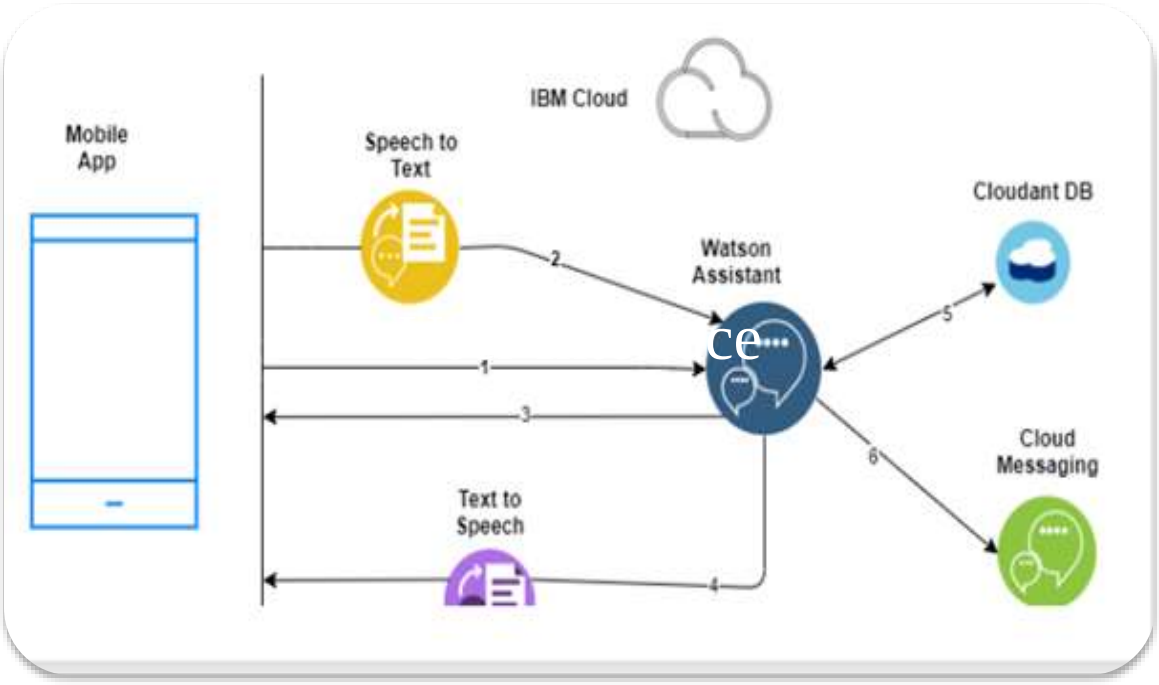
The smaller networks can be analyzed and the results can be employed in larger networks to make routing efficient and economic. As the Simulator has provision for the crashing of routers, it gives an idea of which path is followed when a crash occurs. It can be employed in real networks to increase the performance of routers and links. As it not feasible in real networks to test algorithms and then implement a best one, Routing Simulator can be helpful. Hence it is useful for people who provide networking services and those who design networks.

7. PROPOSED SYSTEM

Develop an end to end mobile application capable of managing/placing orders, displaying recommendations, showing the menu, prompting the best deals or collecting the customer feedback using the IBM Watson Assistant. The customer and the order details are stored in the Cloudant DB. Alert is sent when the order is confirmed using the Cloud Messaging. Speech to Text and Text to Speech Services must be used to take the speech input and give out the speech output.

Features Using chatbot we can manage users reservations and orders We can give food recommendations and display the menu to the users We can Promote best deals and offers on that day We will store the customer's details and orders in the database Chat will send a notification to customers if the order is confirmed The chatbot is also useful in Follow up on customer feedback 3 THEORITICAL ANALYSIS 3.1 Block diagram Figure 1. Block Diagram 3.2 Hardware / Software designing Figure 2. Flow diagram of Node Red Application.

7.1 SYSTEM ARCHITECTURE



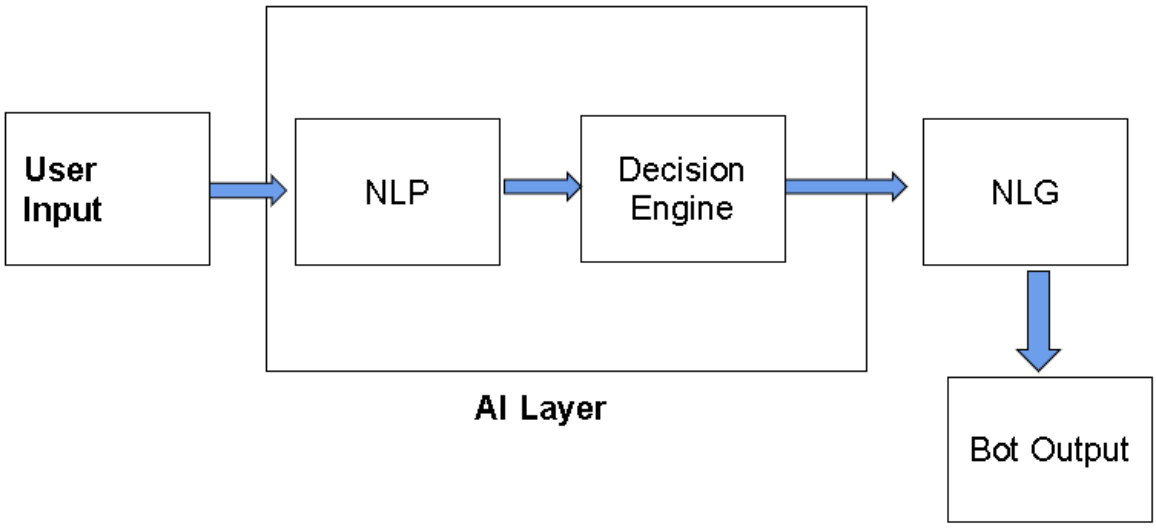


Fig. *7.1 System Architecture for Restaurant chatbot*

7.2 ALGORITHM

Calculation of Efficiency of Chatbot:

1. Define the purpose.

This is the most important thing to determine for your chatbot. Why are you creating a bot in the first place? If you are taking an existing interaction or process and automating it, what is that current experience like, and how could a bot help the customer or improve the process?

2. Define the goal.

Now that you understand the why, what will this bot do? What is the valuable outcome that a user will gain by interacting with it? It is essential to define your chatbot strategy before the writing begins, so you know exactly what the bot will do and why that goal is important.

3. Outline the steps.

With that end goal, work backward to determine all of the steps necessary to reach that goal. You can likely gain the information from the business website, sales materials, interviews with customers or sales agents, etc.

4. Define the audience and personality.

In order to design an experience that converts, it's crucial to know what the user wants and what their sentiment will be during the interaction. To connect with the audience, you have to know them! How do they buy? What are their challenges? How familiar are they with your topics? These are all questions you will need to answer.

5. Map the flows.

Using the node red app, the flow was created with nodes to integrate with Watson assistant , text to speech and speech to text service and the cloudant DB

6. Write the key flows (Hello, Main, Outcome).

With these conversations design best practices in mind, it’s time to write! Begin with the most necessary parts of your chatbot conversation, and write the beginning-to-end "ideal" experience.

This will also help you discover offshoot flows you need to add. I've created an online template for writing all of the dialogue for your chatbot, including these flows, but you may want to write yours in another internal tool or document, depending on your specific needs.

7. Write the secondary flows (other answers to main questions, about, contact, etc.).

When writing the main flows (and in your flow map), you will notice there are many points where flow needs to extend to offer another path, if a user selects a different answer. After you create the main flows, go back and fill in anything that is currently a dead end and make sure all of your flows connect.

8. Create a demo/prototype.

Turn your 2D writing into a working, 3D conversation. Creating a video mockup or working example of a chatbot is the best way to illustrate the experience and demonstrate it for your team.

9. Edit & test the build.

After using your prototype and sharing it with your team, you've likely uncovered messages that are too long, along with dead ends or pathways that don't make sense. Once you have completed these steps, you are ready to build and release a chatbot

8.DESIGN DETAILS

Sorce Code (JSON):

{ "intents": [{ "intent": "enquiry", "examples": [{ "text": "Can you please provide menu list?" }, { "text": "I want to book into the menu" }, { "text": "What are the Offers available?" }, { "text": "What are the special item today?" }], "description": "" }, { "intent": "order_place", "examples": [{ "text": "Can u please take the order" }, { "text": "Can you please take my order" }, { "text": "I want to place order" }], "description": "" }], "entities": [{ "entity": "email", "values": [{ "type": "patterns", "value": "email", "patterns": ["[a-z0-9._%+-]+@[a-z0-9.-]+\.[a-z]{2,}$"] }], "fuzzy_match": true }, { "entity": "enquiry", "values": [{ "type": "synonyms", "value": "deal", "synonyms": ["deals"] }, { "type": "synonyms", "value": "item", "synonyms": ["items"] }, { "type": "synonyms", "value": "menu", "synonyms": [] }, { "type": "synonyms", "value": "offers", "synonyms": [] }, { "type": "synonyms", "value": "specials", "synonyms": ["item", "items"] }], "fuzzy_match": true }, { "entity": "items", "values": [{ "type": "synonyms", "value": "burger", "synonyms": [] }, { "type": "synonyms", "value": "Non veg Sandwich", "synonyms": [] }, { "type": "synonyms", "value": "Paneer Masala", "synonyms": [] }, { "type": "synonyms", "value": "Pizza", "synonyms": [] }, { "type": "synonyms", "value": "Veg Pasta", "synonyms": [] }, { "type": "synonyms", "value": "Veg Sandwich", "synonyms": [] }, { "type": "synonyms", "value": "Zingy Parcel", "synonyms": [] }], "fuzzy_match": true }, { "entity": "order_place", "values": [{ "type": "synonyms", "value": "order", "synonyms": ["allowing", "allows", "attempt", "attempting", "enabling", "ordered", "orders", "permitting", "wishing"] }], "fuzzy_match": true }, { "entity": "payment", "values": [{ "type": "synonyms", "value": "Card", "synonyms": [] }, { "type": "synonyms", "value": "COD", "synonyms": [] }, { "type": "synonyms", "value": "UPI", "synonyms": [] }], "fuzzy_match": true }, { "entity": "specials", "values": [{ "type": "synonyms", "value": "Burger", "synonyms": [] }, { "type": "synonyms", "value": "non veg sandwich", "synonyms": [] }, { "type": "synonyms", "value": "Paneer Masala", "synonyms": [] }, { "type": "synonyms", "value": "Pasta", "synonyms": ["mozzarella", "pastas"] }, { "type": "synonyms", "value": "Pizza", "synonyms": [] }, { "type": "synonyms", "value": "veg pasta", "synonyms": [] }, { "type": "synonyms", "value": "Veg Sandwich", "synonyms": [] }, { "type": "synonyms", "value": "Zingy Parcel", "synonyms": [] }], "fuzzy_match": true }, { "entity": "sys-number", "values": [], "fuzzy_match": true }], "metadata": { "api_version": { "major_version": "v2", "minor_version": "2018-11-08" } }, "dialog_nodes": [{ "type": "standard", "title": "Anything else", "output": { "generic": [{ "values": [{ "text": "I didn't understand. You can try rephrasing." }, { "text": "Can you reword your statement? I'm not understanding." }, { "text": "I didn't get your meaning." }], "response_type": "text", "selection_policy": "sequential" }] }, "conditions": "anything_else", "dialog_node": "Anything else", "previous_sibling": "node_7_1603467781495", "disambiguation_opt_out": true }, { "type": "event_handler", "parent": "node_7_1603467781495", "event_name": "focus", "dialog_node": "handler_1_1603468020792", "previous_sibling": "node_10_1603470359417" }, { "type": "event_handler", "output": { "text": { "values": ["Can you please provide me the item name that you want "], "selection_policy": "sequential" } }, "parent": "slot_4_1603468022275", "event_name": "focus", "dialog_node": "handler_3_1603468022288", "previous_sibling": "handler_8_1603468022288" }, { "type": "event_handler", "output": { "text": { "values": ["How many do you want ?"], "selection_policy": "sequential" } }, "parent":

"slot_3_1603468740585", "event_name": "focus", "dialog_node": "handler_6_1603468740586", "previous_sibling": "handler_7_1603468740586" }, { "type": "event_handler", "output": {}, "parent": "slot_3_1603468740585", "context": { "number": "@sys-number" }, "conditions": "@sys-number", "event_name": "input", "dialog_node": "handler_7_1603468740586" }, { "type": "event_handler", "output": {}, "parent": "slot_10_1603468774641", "context": { "payment": "@payment" }, "conditions": "@payment", "event_name": "input", "dialog_node": "handler_7_1603468774642" }, { "type": "event_handler", "output": {}, "parent": "slot_4_1603468022275", "context": { "items": "@items" }, "conditions": "@items", "event_name": "input", "dialog_node": "handler_8_1603468022288" }, { "type": "event_handler", "output": { "text": { "values": ["We can accept COD,UPI and Card what is your mode of payment?"], "selection_policy": "sequential" } }, "parent": "slot_10_1603468774641", "event_name": "focus", "dialog_node": "handler_9_1603468774642", "previous_sibling": "handler_7_1603468774642" }, { "type": "standard", "title": "Delete Context", "output": { "deleted": "<?context.remove('email')?", "generic": [{ "values": [], "response_type": "text", "selection_policy": "sequential" }] }, "parent": "node_7_1603467781495", "conditions": "true", "dialog_node": "node_10_1603470359417" }, { "type": "standard", "title": "email", "output": { "generic": [{ "values": [{ "text": "Hello , Thank You for giving your mail id $email is it for further references. " }], "response_type": "text", "selection_policy": "sequential" }] }, "parent": "Welcome", "context": { "email": "@email.literal" }, "conditions": "@email", "dialog_node": "node_10_1603472772990" }, { "type": "standard", "title": "enquiry", "metadata": { "_customization": { "mcr": true } }, "conditions": "#enquiry || @enquiry", "digress_in": "returns", "dialog_node": "node_2_1603463340455", "digress_out": "allow_all_never_return", "previous_sibling": "Welcome" }, { "type": "frame", "title": "order_palce", "output": { "generic": [{ "values": [{ "text": "Thank You ,Your order of $number of $items is placed ,please pay using $payment . Your receipt is sent to your $email ." }], "response_type": "text", "selection_policy": "sequential" }] }, "next_step": { "behavior": "skip_user_input" }, "conditions": "#order_place && @order_place", "digress_in": "does_not_return", "dialog_node": "node_7_1603467781495", "digress_out": "allow_all", "previous_sibling": "node_2_1603463340455", "digress_out_slots": "allow_returning" }, { "type": "standard", "title": "specials", "parent": "node_2_1603463340455", "metadata": { "_customization": { "mcr": true } }, "conditions": "@specials", "dialog_node": "node_9_1603466848292" }, { "type": "response_condition", "output": { "generic": [{ "values": [{ "text": "Is cost is 150 per plate" }], "response_type": "text", "selection_policy": "sequential" }] }, "parent": "node_9_1603466848292", "conditions": "@specials:(Burger)", "dialog_node": "response_10_1603467151631", "previous_sibling": "response_5_1603467123870" }, { "type": "response_condition", "output": { "generic": [{ "values": [{ "text": "Is cost is 200 per plate" }], "response_type": "text", "selection_policy": "sequential" }] }, "parent": "node_9_1603466848292", "conditions": "@specials:(Paneer Masala)", "dialog_node": "response_1_1603467042239", "previous_sibling": "response_6_1603466955069" }, { "type": "response_condition", "output": { "generic": [{ "values": [{ "text": "Is cost is 150 per plate" }], "response_type": "text", "selection_policy": "sequential" }] }, "parent": "node_9_1603466848292", "conditions": "@specials:(Veg Sandwich)", "dialog_node": "response_4_1603467062378", "previous_sibling": "response_1_1603467042239" }, { "type": "response_condition", "output": { "generic": [{ "values": [{ "text": "Is cost is 50 per plate" }], "response_type": "text", "selection_policy": "sequential" }] }, "parent": "node_9_1603466848292", "conditions": "@specials:(Zingy Parcel)", "dialog_node": "response_5_1603467123870",

"previous_sibling": "response_7_1603467084899" }, { "type": "response_condition", "output": { "generic": [{ "values": [{ "text": "Is cost is 100 per plate" }], "response_type": "text", "selection_policy": "sequential" }] }, "parent": "node_9_1603466848292", "conditions": "@specials:(veg pasta)", "dialog_node": "response_6_1603466955069" }, { "type": "response_condition", "output": { "generic": [{ "title": "Special Items", "options": [{ "label": "Pizza", "value": { "input": { "text": "Pizza" } } }, { "label": "Paneer Masala", "value": { "input": { "text": "Paneer Masala" } } }, { "label": "Veg Pasta", "value": { "input": { "text": "Veg Pasta" } } }, { "label": "Veg Sandwich ", "value": { "input": { "text": "Veg Sandwich " } } }, { "label": "Burger", "value": { "input": { "text": "Burger" } } }, { "label": "Zingy Parcel", "value": { "input": { "text": "Zingy Parcel" } } }, { "label": "Non veg Sandwich", "value": { "input": { "text": "Non veg Sandwich" } } }], "response_type": "option" }] }, "parent": "node_2_1603463340455", "conditions": "@enquiry:specials", "dialog_node": "response_7_1603464897321", "previous_sibling": "response_8_1603464584961" }, { "type": "response_condition", "output": { "generic": [{ "values": [{ "text": "Is cost is 300 per plate" }], "response_type": "text", "selection_policy": "sequential" }] }, "parent": "node_9_1603466848292", "conditions": "@specials:(non veg sandwich)", "dialog_node": "response_7_1603467084899", "previous_sibling": "response_4_1603467062378" }, { "type": "response_condition", "output": { "generic": [{ "values": [{ "text": "Is cost is 350 per plate" }], "response_type": "text", "selection_policy": "sequential" }] }, "parent": "node_9_1603466848292", "conditions": "@specials:(Pizza)", "dialog_node": "response_7_1603467194995", "previous_sibling": "response_10_1603467151631" }, { "type": "response_condition", "output": { "generic": [{ "values": [{ "text": "We are having 20% off on vegetarian " }], "response_type": "text", "selection_policy": "sequential" }, { "values": [{ "text": "We have 10% off on all non veg menu for women day special discount." }], "response_type": "text", "selection_policy": "sequential" }, { "values": [{ "text": "We are having 10% discount on fast food." }], "response_type": "text", "selection_policy": "sequential" }] }, "parent": "node_2_1603463340455", "conditions": "@enquiry:offers", "dialog_node": "response_8_1603464584961", "previous_sibling": "response_9_1603463590525" }, { "type": "response_condition", "output": { "generic": [{ "title": "Menu", "source": "https://im1.dineout.co.in/images/uploads/restaurant/sharpen/4/j/o/m41327-15239489375ad59d896b40e.jpg", "response_type": "image" }] }, "parent": "node_2_1603463340455", "conditions": "@enquiry:menu", "dialog_node": "response_9_1603463590525", "previous_sibling": "node_9_1603466848292" }, { "type": "slot", "parent": "node_7_1603467781495", "variable": "$payment", "dialog_node": "slot_10_1603468774641", "previous_sibling": "slot_3_1603468740585" }, { "type": "slot", "parent": "node_7_1603467781495", "variable": "$number", "dialog_node": "slot_3_1603468740585", "previous_sibling": "slot_4_1603468022275" }, { "type": "slot", "parent": "node_7_1603467781495", "variable": "$items", "dialog_node": "slot_4_1603468022275", "previous_sibling": "handler_1_1603468020792" }, { "type": "standard", "title": "Welcome", "output": { "generic": [{ "values": [{ "text": "Hello. I am Restaurant Bot , How can I help you? Can you please provide your name and mail id" }], "response_type": "text", "selection_policy": "sequential" }] }, "conditions": "welcome", "dialog_node": "Welcome" }], "counterexamples": [], "system_settings": { "off_topic": { "enabled": true }, "disambiguation": { "prompt": "Did you mean:", "enabled": true, "randomize": true, "max_suggestions": 5, "suggestion_text_policy": "title", "none_of_the_above_prompt": "None of the above" }, "system_entities": { "enabled": true },

"human_agent_assist": { "prompt": "Did you mean:" }, "spelling_auto_correct": true }, "learning_opt_out": false, "name": "Disha_Bot", "language": "en", "description": "" }

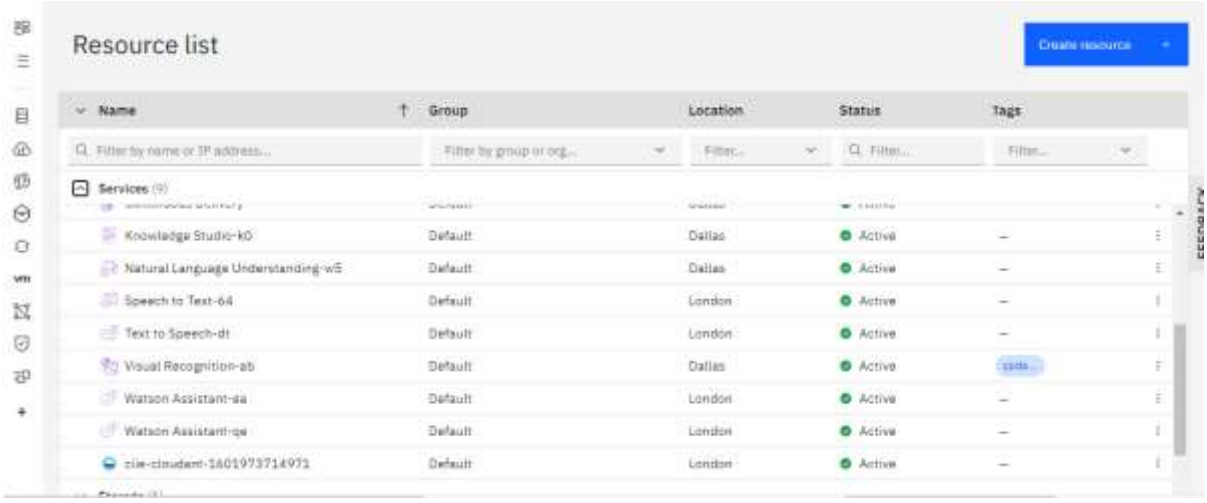
NodeRed Json File_ Original

[{"id":"ad1dc02c.ce618","type":"watson-conversation-v1","z":"33638b7b.8328f4","name":"Resto Bot","workspaceid":"9e85b3f5-01b5-40dc-a7c4-e2191ad97991","multiuser":false,"context":true,"empty-payload":false,"service-endpoint":"https://api.eu-gb.assistant.watson.cloud.ibm.com/instances/2b7174ab-f20b-4f7b-ad9a-a03810ba74b0","timeout":"","optout-learning":false,"x":400,"y":100,"wires":[["1b4772a7.eca23d"]]}]

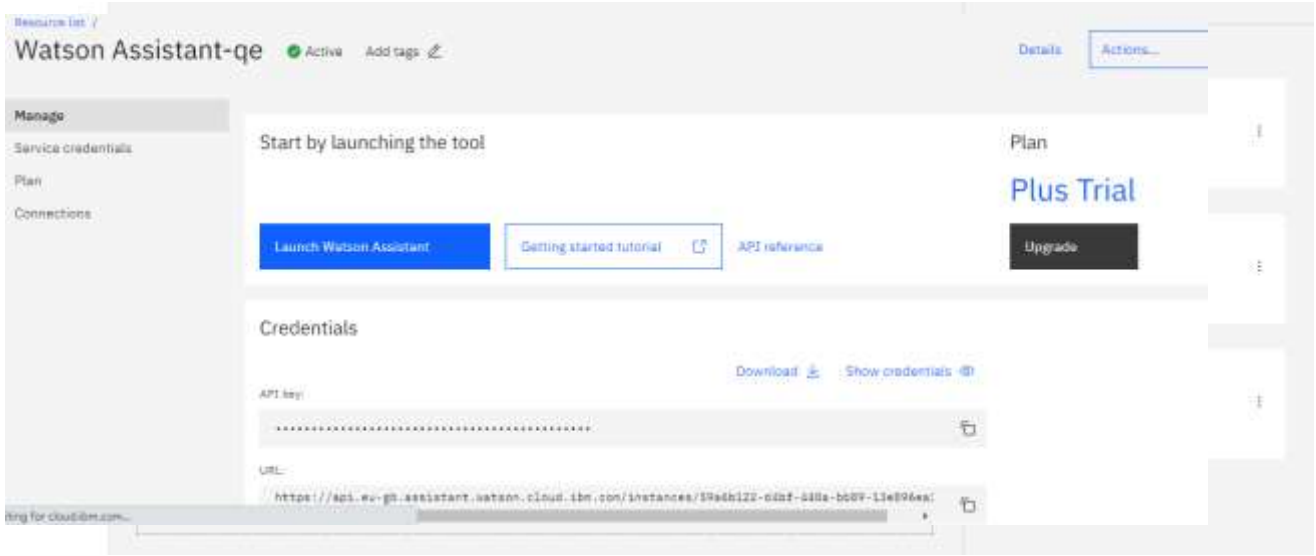
9. Screenshot

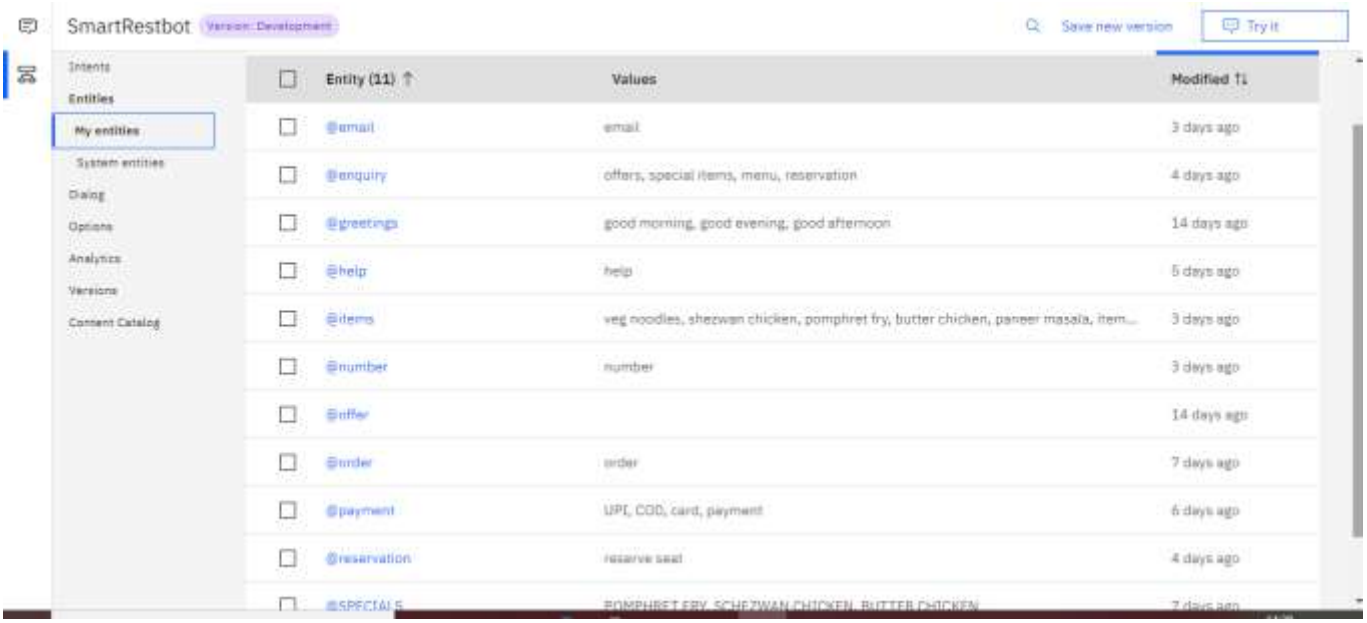
The output obtained in each stages are shown in figures below.

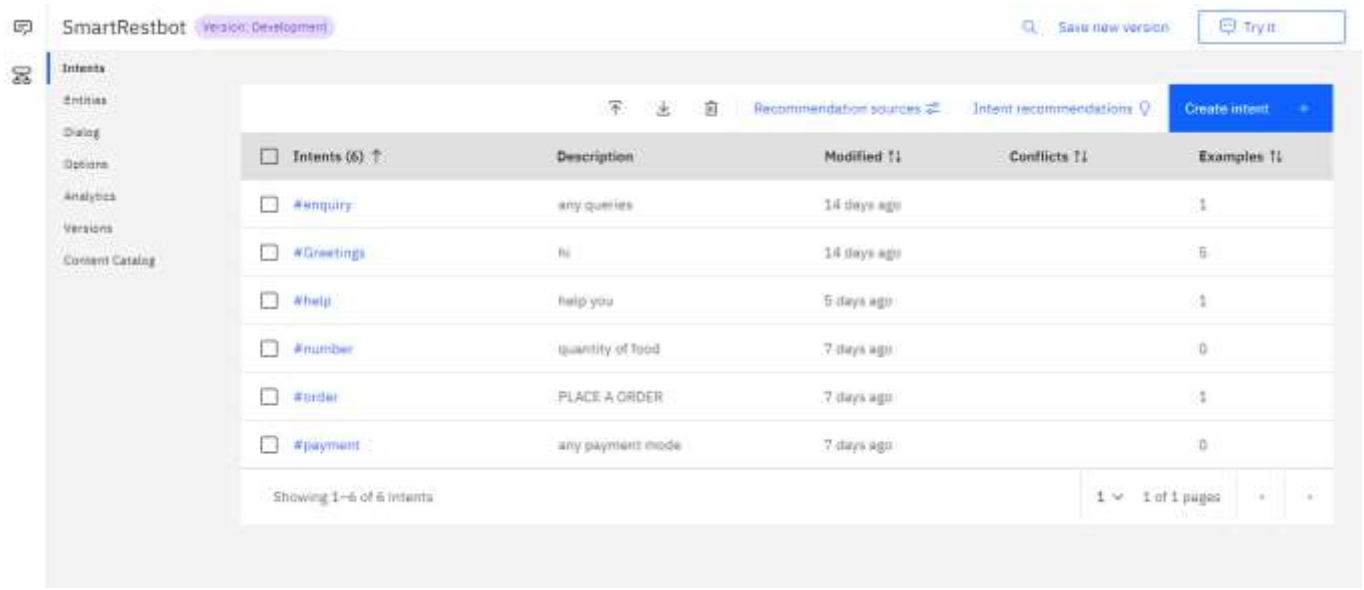
1. **SET UP IBM CLOUD ACCOUNT,CLOUDANT DB & ACTIVATE NECESSARY SERVICES**

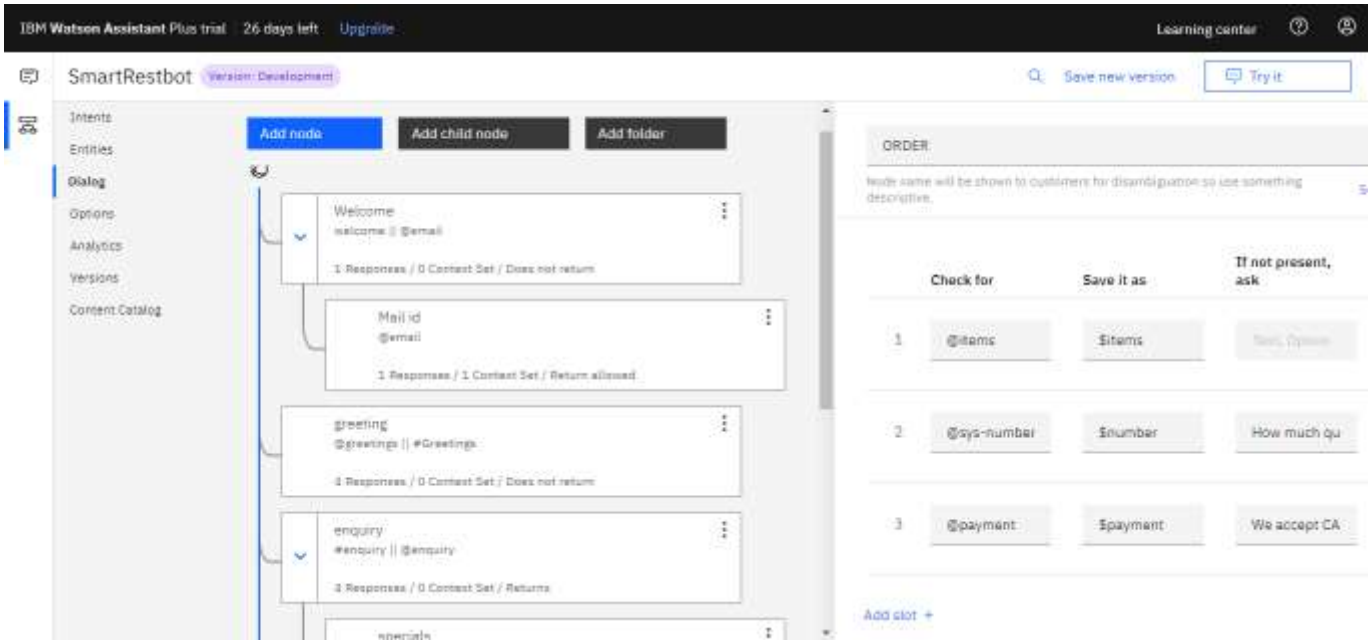


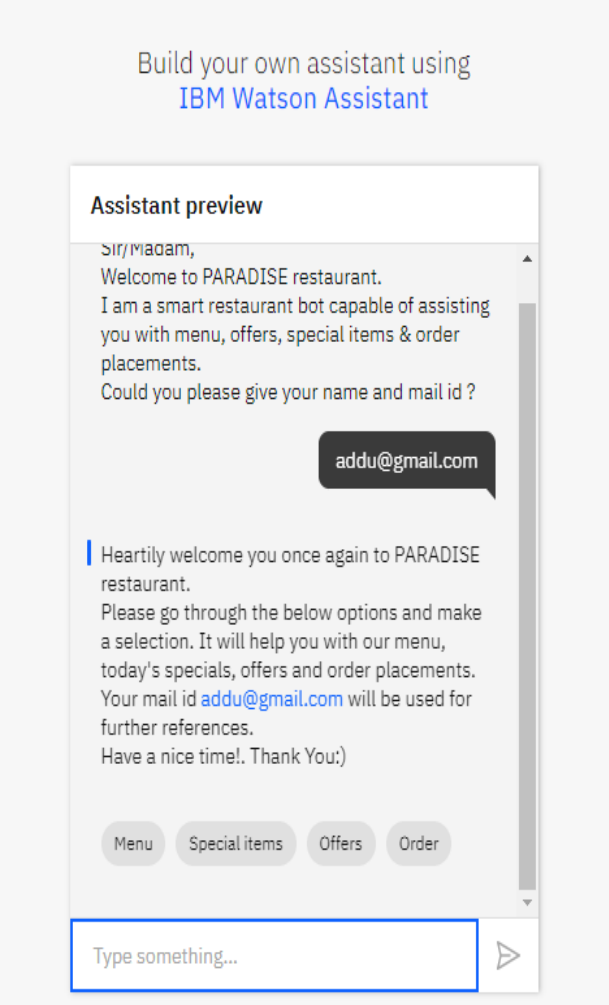
2. **BUILD A CHAT BOT USING WATSON ASSISTANT**

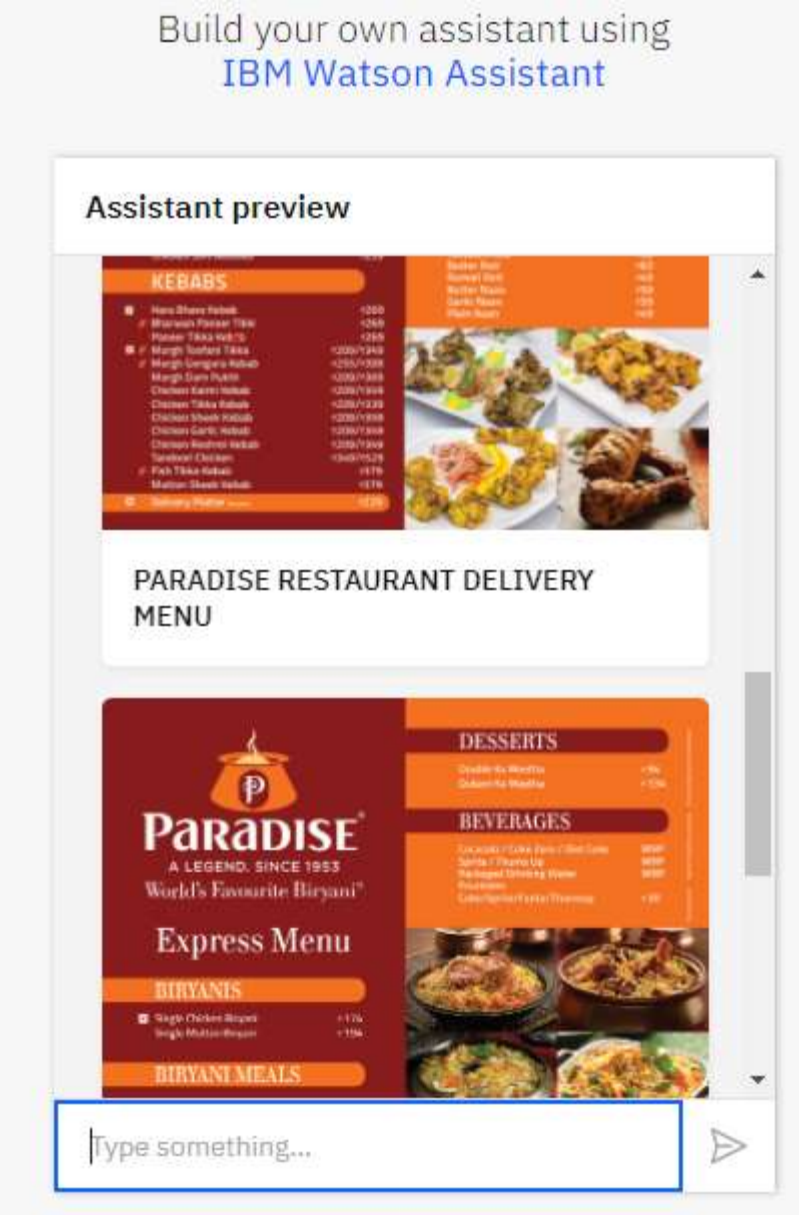


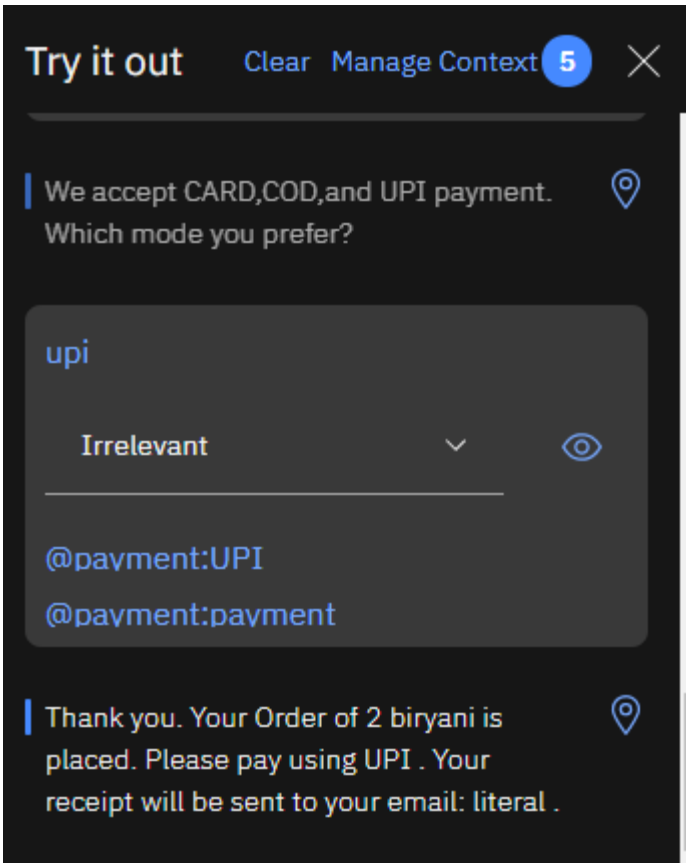

SmartRestbot
Version: Development
Save new version
Try it
Intents
Entities
My entities
System entities
Dialog
Options
Analytics
Versions
Content Catalog
Entity (11)
Values
Modified
@email
email
3 days ago
@enquiry
offers, special items, menu, reservation
4 days ago
@greetings
good morning, good evening, good afternoon
14 days ago
@help
help
5 days ago
@items
veg noodles, shezwan chicken, pomphret fry, butter chicken, paneer masala, item...
3 days ago
@number
number
3 days ago
@offer
14 days ago
@order
order
7 days ago
@payment
UPI, COD, card, payment
6 days ago
@reservation
reserve seat
4 days ago


SmartRestbot
Version: Development
Save new version
Try it
Intents
Entities
Dialog
Options
Analytics
Versions
Content Catalog
Recommendation sources
Intent recommendations
Create intent
Intents (6)
Description
Modified
Conflicts
Examples
#enquiry
any queries
14 days ago
1
#Greetings
hi
14 days ago
6
#help
help you
5 days ago
1
#number
quantity of food
7 days ago
0
#order
PLACE A ORDER
7 days ago
1
#payment
any payment mode
7 days ago
0
Showing 1–6 of 6 intents
1 of 1 pages

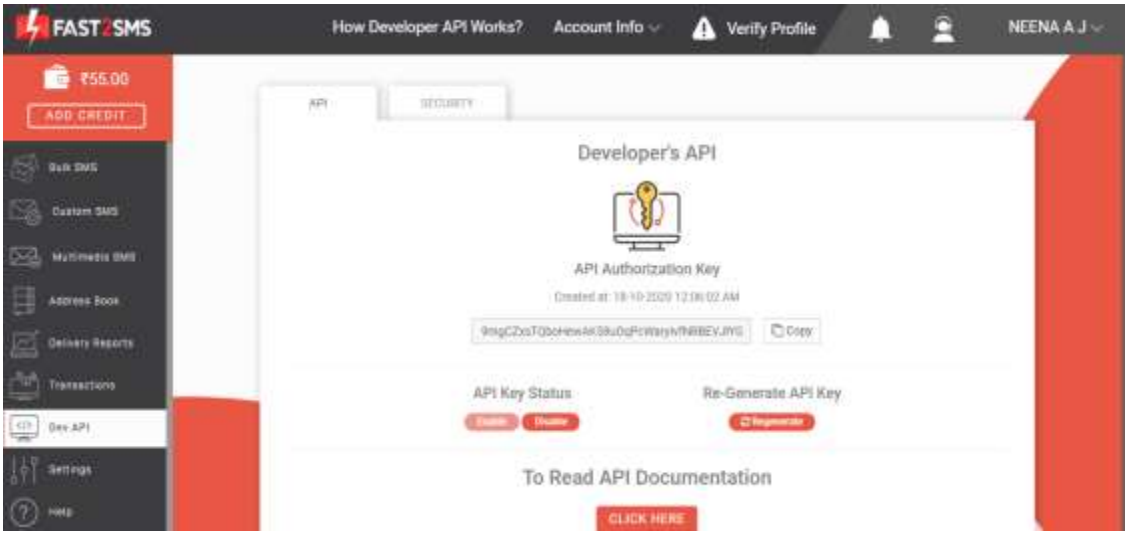

IBM Watson Assistant Plus trial 26 days left Upgrade
Learning center
SmartRestbot
Save new version
Try it
Intents
Entities
Dialog
Options
Analytics
Versions
Content Catalog
Add node
Add child node
Add folder
Welcome
Mail id
@email
greeting
@greetings || #Greetings
enquiry
#enquiry || @enquiry
ORDER
Check for
Save it as
If not present, ask
1
@items
$items
2
@sys-number
$number
How much qu
3
@payment
$payment
We accept CA
Add slot +


Build your own assistant using
IBM Watson Assistant
Assistant preview
Sir/Madam,
Welcome to PARADISE restaurant.
I am a smart restaurant bot capable of assisting you with menu, offers, special items & order placements.
Could you please give your name and mail id ?
addu@gmail.com
Heartily welcome you once again to PARADISE restaurant.
Please go through the below options and make a selection. It will help you with our menu, today's specials, offers and order placements.
Your mail id addu@gmail.com will be used for further references.
Have a nice time!. Thank You:)
Menu
Special items
Offers
Order
Type something...

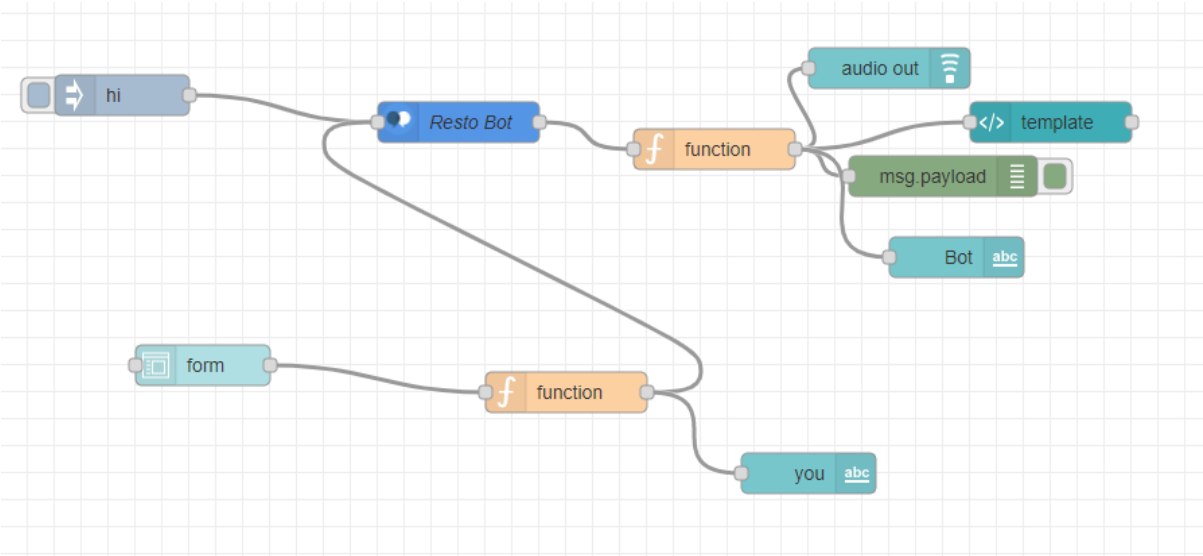

Build your own assistant using
IBM Watson Assistant
Assistant preview
KEBABS
PARADISE RESTAURANT DELIVERY MENU
PARADISE
A LEGEND. SINCE 1953
World's Favourite Biryani
Express Menu
BIRYANIS
BIRYANI MEALS
DESSERTS
BEVERAGES
Type something...



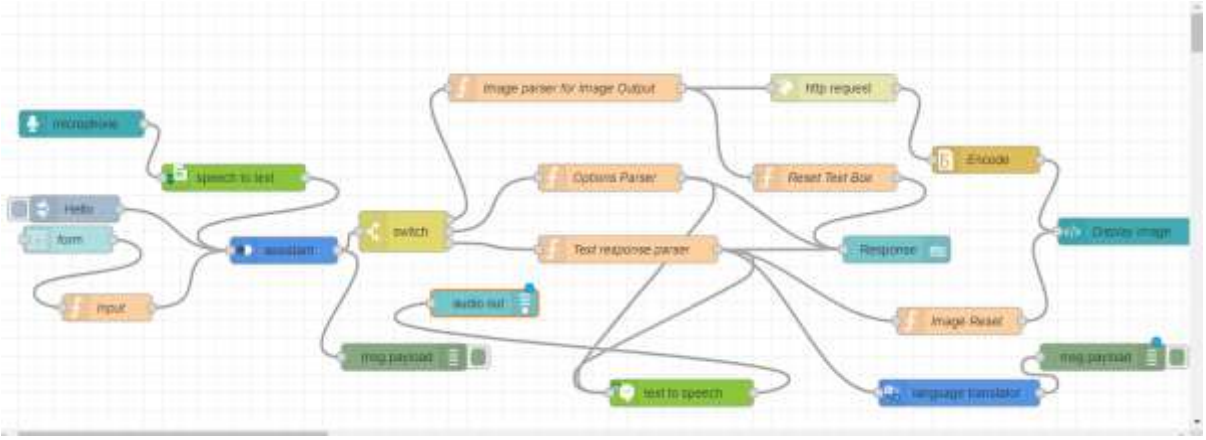
3. CREATE A FAST2SMS SERVICE ACCOUNT



4. BUILD NODE RED FLOW AND INTEGRATE WITH WATSON ASSISTANT



Flow 1: NODE RED FLOW WITHOUT SPEECH TO TEXT INTEGRATION



FLOW 2: NODE RED FLOW WITH MICROPHONE AND STT

5. FINAL OUTPUT BASED ON NODE RED FLOW ABOVE

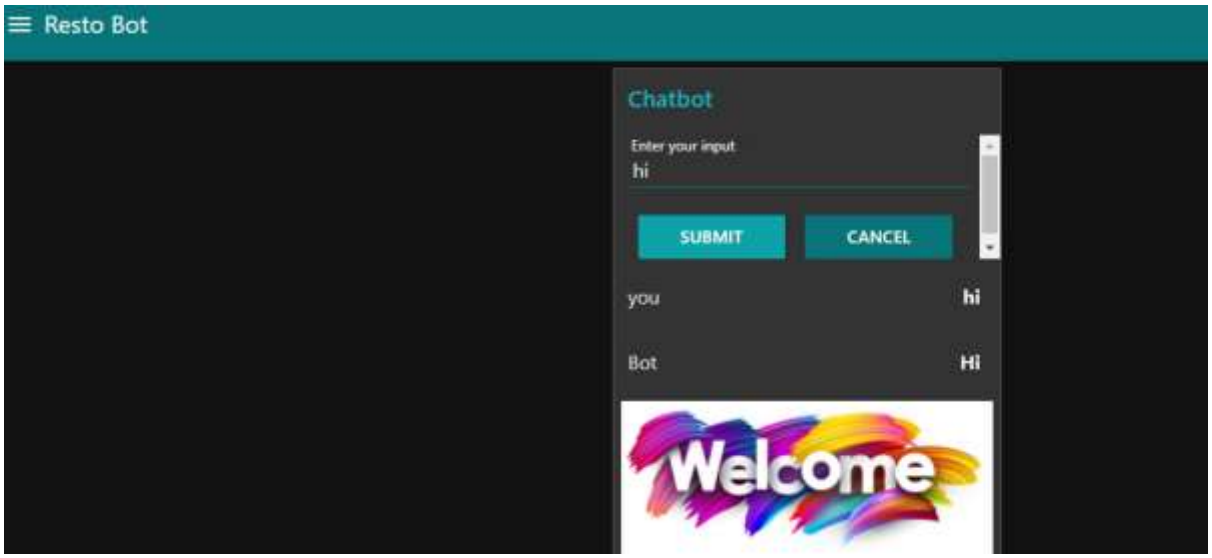
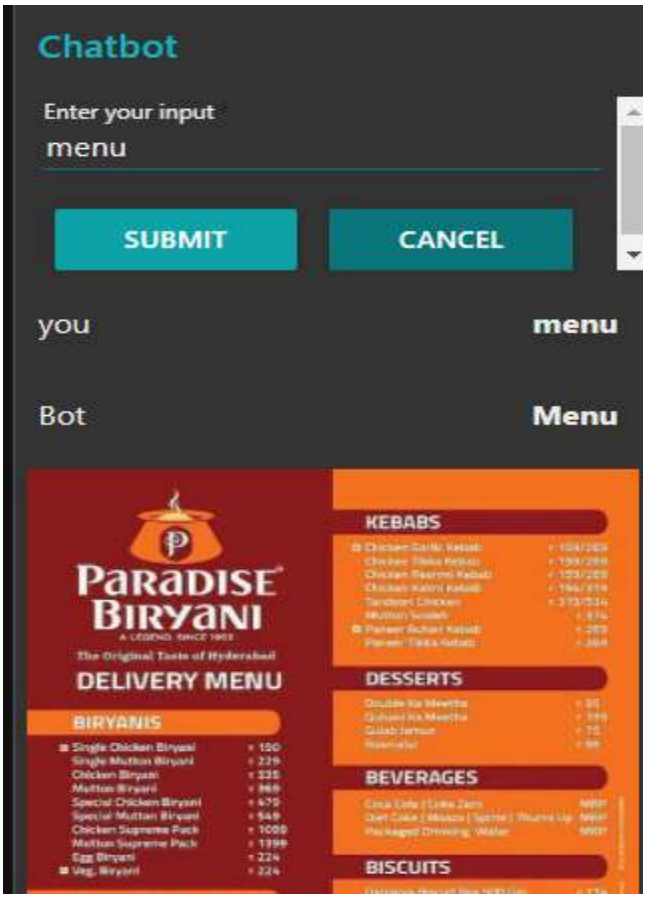


Figure 1: OUTPUT OF FLOW 1 UI WITHOUT STT



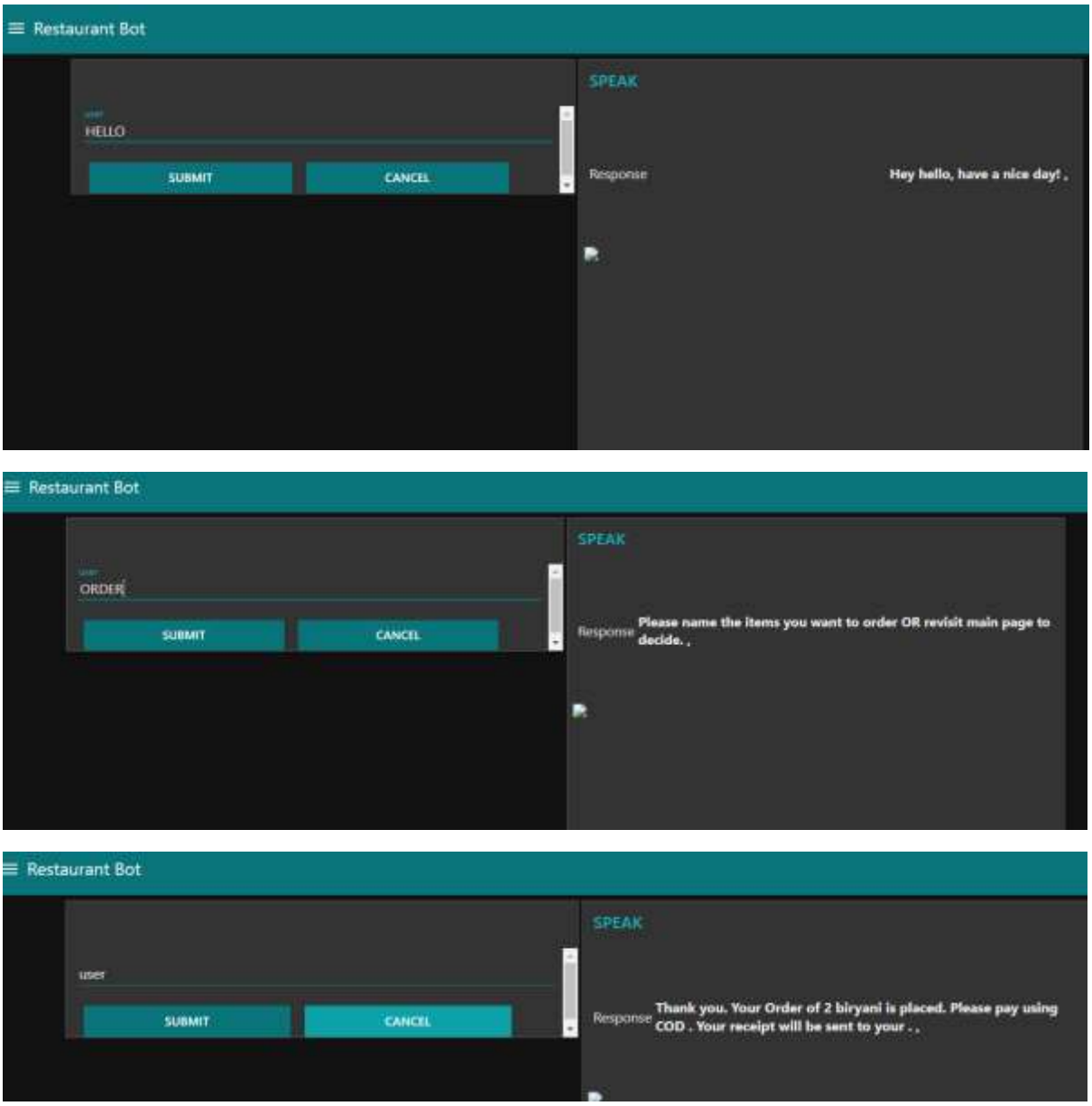


Figure 2: OUTPUT OF FLOW 2 WITH STT

10. Details Hardware and Software

10.1 Hardware Requirements

- Processor : INTEL DUAL CORE
- Ram : 1GB RAM minimum
- Resolution : 800*600 resolution; 16 bit color (1024*768, 24 bit)

10.2 Software Requirements

- Application : IBM CLOUD
- Language: : JSON
- Operating Systems : Windows 2000,XP,7,10

11. ADVANTAGES

1. Easy and quick to do the Restaurant Related Activity.
2. We can give order
3. We can check special /Offer related to menu.
4. We can access more information

12. CONCLUSION

As a conclusion, this software of project is successfully functioned as the objectives of the project. This project helps to solve the problem faced by the restaurant entrepreneur in the attempt to organize the restaurant more efficiently skilled. It can also be used to reduce the lateness and the error caused on ordering foods by the customers by waiters. By using this system, the complaints about the services are eliminated