Intelligent Customer Help Desk with Smart Document Understanding - SB5064

Category:Artificial Intelligence

Application ID: SPS_APL_20200000639

Project ID:SPS_PRO_99

Internship at SmartInternz

Rajviba Gohil

rajvibagohil2411@gmail.com

PROJECT PLANNING

1	**Project Summary**	The project is on building an Intelligent Customer Help Desk chatbot, which can answer simple questions such as store locations and hours, directions, and even make appointments. When a question falls outside of the scope of the pre-determined question set, this intelligent customer care chatbot will pass the question onto Watson Discovery Service, which has been pre-loaded with the device's owner's manual.Also the projectuses the Smart Document Understanding feature of Watson Discovery to train it on what text in the owner's manual is important and what is not.
2	**Project requirements**	The project requires basic knowledge of python programming, IBM cloud and its services.

3	**Functional requirements**	- Watson Assistant - Watson Discovery - Node-red - Cloud functions
4	**Technical requirements**	Basic idea and knowledge of AI /ML
5	**Project Deliverables**	A fully functional chatbot that can handle basic customer requests and when the question falls out of scope of the chatbot it uses Smart document understanding skill to return the most relevant answer.

6	**Project team**	Individual Project
7	**Project Schedule**	30th April to 31st May

INDEX

1. INTRODUCTION

1.1 OVERVIEW

A chatbot is an artificial intelligence (AI) software that can simulate a conversation (or a chat) with a user in natural language through messaging applications, websites, and mobile apps or through the telephone

1.2 PURPOSE

A chatbot is often described as one of the most advanced and promising expressions of interaction between humans and machines. However, from a technological point of view, a chatbot only represents the natural evolution of a Question Answering system leveraging Natural Language Processing (NLP). Formulating responses to questions in natural language is one of the most typical Examples of Natural Language Processing applied in various enterprises' end-use applications

2 LITERATURE SURVEY

2.1 Existing Problem

The typical customer care chatbot can answer simple questions, such as store locations and hours, directions, and maybe even making appointments. When a question falls outside of the scope of the pre-determined question set, the option is typically to tell the customer the question isn't valid or offer to speak to a real person. But what if a real person is somehow not able to communicate? This is where it creates a barrier in communication.

2.2 Proposed Solution

In this project, there will be another option. If the customer question is about the operation of a device, the application shall pass the question onto Watson Discovery Service, which has been pre-loaded with the device's owner's manual. So now, instead of "Would you like to speak to a customer representative?" we can return relevant sections of the owner's manual to help solve our customers' problems.

> To take it a step further, the project shall use the Smart Document Understanding feature of Watson Discovery to train it on what text in the owner's manual is important and what is not. This will improve the answers returned from the queries.

3 THEORITICAL ANALYSIS

3.1 Block diagram

3.2 HARDWARE/SOFTWARE DESIGINING

1. Create IBM Cloud Services

2. Configure Watson Discovery

3. Create IBM Cloud Functions action

4. Configure Watson Assistant

5. Build Node-RED Flow to Integrate All Services

6. Configure the nodes and Build A Web Dashboard in Node-RED

7. Deploy and Run the application

4 EXPERIMENTAL INVESTIGATIONS

1.Create IBM Cloud Services:

1.Firstly go to https://cloud.ibm.com

2. Create IBM cloud account or login if you already have one

3.IBM dashboard

For this project, we need to Create the following services:

1. Watson Discovery

2. Watson Assistant

1.To create a Watson Discovery Service, search for Discovery in the search box and click on discovery service.

Next,create necessary discovery services using lite plan(Select London as your region) .After successful configuration, an discovery instance will be created.

(If you are on Lite Plan, you can have only one instance per service)

2. To create a Watson Assistant Service, search for Assistant in the search box and click on watson service.

Next,create necessary watson services using lite plan(Select London as your region) .After successful configuration, an assistant instance will be created.

(If you are on Lite Plan, you can have only one instance per service)

To check whether you have correctly configured the services, go back to the IBM Dashboard and click on View All from the Resource Summary Tab.

All of your existing Resource list will be shown here, click on Services to unveil the list of services you have.

Here we can find that the status of Watson Discovery and Watson Assistant as Active which means we have configured the services correctly.

2.Configure Watson Discovery

From resource list select watson discovery and click on “Launch on watson discovery”

Now in the screen that appears,click on “Upload your own data” and give it a unique name.

A tab like this will appear:

Now select a file that you want to use(Lite plans: PDF, Word, PowerPoint, Excel, JSON, HTML)

SDU trains Watson Discovery to extract custom fields in your documents. Customizing how your documents are indexed into Discovery will improve the answers returned from queries.With SDU, you annotate fields within your documents to train custom conversion models. As you annotate, Watson is learning and will start predicting annotations. SDU models can also be exported and used on other collections. Current document type support for SDU is based on your plan

After you have selected a document to be uploaded,you will see that now discovery is processing your data and extracting information from it.

Now click on “Configure your data” button in the top right corner.

For this specific owner's manual, at a minimum, it is suggested to mark the following:

- The main title page as title

- The table of contents (shown in the first few pages) as table_of_contents

- All headers and sub-headers (typed in light green text) as a subtitle

- All page numbers as footers

• All warranty and licensing information (located in the last few pages) as a footer • All other text should be marked as text.

Now use manage fields to choose which fields to use to enrich data and split the document accordingly.

After successful configuration you will see a tab like this.

Now in the parameters tab,provide with correct credentials.

Now invoke your cloud function and you will see the function running.

Now click on endpoints in the same tab containing parameters.

Configure as shown above.

4. Configure Watson Assistant

Now go back to dashboard and open watson assistant.Click on launch Watson Assistant.

Go to skills tab and create your own skill.

Here you will create

1.Intents (user post questions)

2.Enteties(keywords from user post questions)

3.Dialog (flow for the chatbot)

Create necessary services for the skill as required.

For,the product information dialog we will use webhooks to connect cloudfunctions to our assistant.

Refer ibm cloud documentation for detailed information.

Try the assistant from the “Try it.”

5. Build Node-RED Flow to Integrate All Services

Now,search node red app in the catlog.

Create necessary services.(Follow this tutorial :https://developer.ibm.com/tutorials/how-to-create-a-node-red-starter-application/)

Click on visit app url.

Click on go to Node-red flow editor.

And you will see the node-red editor.

Create the flow for your chatbot

6. Configure the nodes and Build A Web Dashboard in Node-RED

.

7. Deploy and Run the application

Now go to dashboard and the web ui is created after successfull configuration of all services.

7.ADVANTAGES & DISADVANTAGES

Advantages:

- Faster Customer Service
- Increased Customer Satisfaction
- Lower Labour Costs
- Variety of Uses
- Data collection
- 24-7 availability
- Multiple Customer Handling

Disadvantages:

- Limited Responses for Customers
- Customers Could Become Frustrated
- Maintenance
- They aren’t human
- Time-Consuming

8.APPLICATIONS

A Product or Software Company Customer Help Desk

9.CONCLUSION An Intelligent Customer Helpdesk with Smart Document Understanding is made using various IBM Services like IBM Watson Discovery, IBM Watson and IBM Cloud Function.

10.FUTURE SCOPE

A More Human Friendly Chatbot, or a personalized Chatbot is to be expected.

11.BIBILOGRAPHY

A.Source Code

Cloud function:

```
/**

*

* @param {object} params

* @param {string} params.iam_apikey

* @param {string} params.url

* @param {string} params.username

* @param {string} params.password

* @param {string} params.environment_id

* @param {string} params.collection_id

* @param {string} params.configuration_id

* @param {string} params.input

*

* @return {object}
```

```
*
*/
const assert = require('assert');
const DiscoveryV1 = require('watson-developer-cloud/discovery/v1');
/**
*
* main() will be run when you invoke this action
*
* @param Cloud Functions actions accept a single parameter, which must be a JSON object.
*
* @return The output of this action, which must be a JSON object.
*
*/
function main(params) {
return new Promise(function (resolve, reject) {
let discovery;
if (params.iam_apikey){
discovery = new DiscoveryV1({
'iam_apikey': params.iam_apikey,
'url': params.url,

'version': '2019-03-25'

});

}

else {

discovery = new DiscoveryV1({

'username': params.username,

'password': params.password,

'url': params.url,

'version': '2019-03-25'

});

}

discovery.query({

'environment_id': params.environment_id,

'collection_id': params.collection_id,

'natural_language_query': params.input,

'passages': true,

'count': 3,

'passages_count': 3

}, function(err, data) {

if (err) {
```

```
return reject(err);
}
return resolve(data);
});
});
}
```

Watson Assistant:

```
{
"intents": [
{
"intent": "Help",
"examples": [
{
"text": "help me decide"
},
{
"text": "i need assistance"
},
{
"text": "help me"
},
{
"text": "can you help"
},
{
"text": "can you assist me"
},
{
"text": "help"
}
],
"description": "Need Assistance"
},
{
"intent": "Customer_Assistant_appointment",
"examples": [
{
"text": "I want to talk in person with someone about my case"
},
{
"text": "I would like to make an appointment to visit the nearest store to my location."
```

```
},
{
"text": "Make an appointment"
},
{
"text": "Set up an appt"
},
{
"text": "Store appointment"
},
{
"text": "What time can I meet the staff?"
},
{
"text": "How do i meet you in person"
},
{
"text": "can i book for tonight"
},
{
"text": "are you available on tuesday"
```

```
},
{
"text": "Can I book an in person session"
},
{
"text": "can i make an appointment"
},
{
"text": "Could I speak to someone in the store next tuesday?"
},
{
"text": "I prefer a face to face visit"
}
],
"description": "Schedule or make appointment with the store"
},
{
"intent": "Product_information",
"examples": [
{
}
"text": "how to increase screen brightness"
},
{
"text": "How to use the heater"
},
{
"text": "how to turn on the heater"
},
{
"text": "what is thermostat"
},
{
"text": "give me product information"
},
{
"text": "heater turn on"
},
{
"text": "how to adjust screen brightness"
}
],
"description": "redirects "
},
{
"intent": "Goodbye",
"examples": [
{
"text": "Time to leave"
},
{
"text": "See you"
},
{
"text": "Bye"
},
{
"text": "Goodbye"
},
{
"text": "see ya"
}
],
"description": "Bye"
},
{
"intent": "Thanks",
"examples": [
{
"text": "thanks"
},
{
"text": "that's nice of you"
},
{
"text": "i appreciate it"
},
{
"text": "thx"
},
{
"text": "many thanks"
},
{
},
"text": "much appreciated"
},
{
"text": "thank you"
},
{
"text": "thank you very much"
}
],
"description": "Thanks-giving"
},
{
"intent": "Customer_Assistant_hours",
"examples": [
{
"text": "are you open on holidays?"
},
{
"text": "open on new year?"
},
{
```

```
"text": "are stores open on sunday?"
},
{
"text": "Are the stores open early?"
},
{
"text": "Are you closed new Year's eve?"
},
{
"text": "Are you closing early today"
},
{
"text": "How long is the store open for?"
},
{
"text": "Can you tell me how late the stores are open till?"
},
{
"text": "Store closed today?"
},
{
"text": "Store open today?"
},
{
"text": "Can I come to store today?"
},
{
"text": "Can you tell me store hours?"
},
{
"text": "store open"
}
],
"description": "Shows store's hours for opening and
closing."
},
{
"intent": "Customer_Assistant_Greetings",
"examples": [
{
"text": "Hi"
```

```
},
{
"text": "Hello"
},
{
"text": "Good morning"
},
{
"text": "Good day"
},
{
"text": "Hey"
},
{
"text": "Greetings"
},
{
"text": "How are you today?"
},
{
"text": "How is it going?"
```

```
},
{
"text": "What's up?"
},
{
"text": "Who are you?"
},
{
"text": "You there"
},
{
"text": "Yo"
},
{
"text": "Heya"
},
{
"text": "gm"
},
{
"text": "Hi there"
```

```
},
{
"text": "Take me back"
}
],
"description": "General Greetings"
},
{
"intent": "Customer_Assistant_location",
"examples": [
{
"text": "How can i come to your store?"
},
{
"text": "what is your address"
},
{
"text": "find store"
},
{
"text": "How to get to your place?"
```

```
},
{
"text": "please suggest route"
},
{
"text": "store directions"
},
{
"text": "where are you?"
},
{
"text": "Where are you located?"
},
{
"text": "How can i find or reach your store?"
},
{
"text": "How to locate store?"
}
],
"description": "Location of the store"
```

```
}
],
"entities": [
{
"entity": "phone",
"values": [
{
"type": "patterns",
"value": "Indian phone pattern",
"patterns": [
"^((\\+){1}91){1}[1-9]{1}[0-9]{9}$"
]
}
],
"fuzzy_match": true
},
{
"entity": "sys-time",
"values": [],
"fuzzy_match": true
},
{
"entity": "landmark",
"values": [
{
"type": "synonyms",
"value": "grand central",
"synonyms": []
},
{
"type": "synonyms",
"value": "times square",
"synonyms": [
"time sqaure place"
]
},
{
"type": "synonyms",
"value": "london eye",
"synonyms": []
},
{
```

```
"type": "synonyms",
"value": "empire state building",
"synonyms": [
"empire state"
]
}
],
"fuzzy_match": true
},
{
"entity": "holidays",
"values": [
{
"type": "synonyms",
"value": "christmas",
"synonyms": [
"xmaz ",
"eve",
"xmas",
"thanksgiving"
]
```

```
},
{
"type": "synonyms",
"value": "new year",
"synonyms": [
"diwali",
"new year eve"
]
},
{
"type": "synonyms",
"value": "independence day",
"synonyms": []
},
{
"type": "synonyms",
"value": "republic day",
"synonyms": []
}
],
"fuzzy_match": true
```

```
},
{
"entity": "reply",
"values": [
{
"type": "synonyms",
"value": "no",
"synonyms": [
"definitely not",
"don't think so",
"i think not",
"dont think so",
"not at this time",
"not now"
]
},
{
"type": "synonyms",
"value": "yes",
"synonyms": [
"definitely",
"go for it",
"let's do it",
"ok",
"sure",
"please",
"yes",
"yeah",
"yep"
]
}
],
"fuzzy_match": true
},
{
"entity": "Greetings",
"values": [
{
"type": "synonyms",
"value": "Good evening",
"synonyms": [
"evening",
"eve",
"dawn"
]
},
{
"type": "synonyms",
"value": "Good morning",
"synonyms": [
"morning",
"gm",
"mrng",
"mornings"
]
},
{
"type": "synonyms",
"value": "Good afternoon",
"synonyms": [
"afternoon",
"noon"
]
```

```
}
],
"fuzzy_match": true
},
{
"entity": "sys-date",
"values": [],
"fuzzy_match": true
},
{
"entity": "sys-number",
"values": [],
"fuzzy_match": true
}
],
"metadata": {
"api_version": {
"major_version": "v2",
"minor_version": "2018-11-08"
}
},
"webhooks": [
{
"url": "https://eu-gb.functions.cloud.ibm.com/api/v1/web/201801446%40daiict.ac.in_dev/default/Chatbot_function.json",
"name": "main_webhook",
"headers": []
}
],
"dialog_nodes": [
{
"type": "response_condition",
"output": {
"text": {
"values": [
"Good evening,Hope you had a great day."
],
"selection_policy": "sequential"
}
},
"parent": "node_5_1589661264481",
},
"conditions": "@Greetings:(Good evening)",
"dialog_node": "response_7_1589662306907",
"previous_sibling": "response_5_1589662265628"
},
{
"type": "response_condition",
"output": {
"text": {
"values": [
"Good afternoon,Hope you are having good day.How may i help you? "
],
"selection_policy": "sequential"
}
},
"parent": "node_5_1589661264481",
"conditions": "@Greetings:(Good afternoon)",
"dialog_node": "response_5_1589662265628",
"previous_sibling": "response_9_1589662211352"
},
{
```

```
"type": "response_condition",
"output": {
"text": {
"values": [
"Hello ,Hope you are having a nice day.How may i help you?"
],
"selection_policy": "sequential"
}
},
"parent": "node_5_1589661264481",
"conditions": "#Customer_Assistant_Greetings",
"dialog_node": "response_6_1589662348448",
"previous_sibling": "response_7_1589662306907"
},
{
"type": "response_condition",
"output": {
"text": {
"values": [
"Good morning,Hope you are having a great morning.How may i help you?"
],
"selection_policy": "sequential"
}
},
"parent": "node_5_1589661264481",
"conditions": "@Greetings:(Good morning)",
"dialog_node": "response_9_1589662211352"
},
{
"type": "response_condition",
"output": {
"text": {
"values": [
"try again.I didnt get it"
],
"selection_policy": "sequential"
}
},
"parent": "node_9_1590437082562",
"conditions": "anything_else",
"dialog_node": "response_1_1590437156819",
"previous_sibling": "response_9_1590437151924"
},
{
"type": "response_condition",
"output": {
"generic": [
{
"values": [
{
"text": "$webhook_result_1"
}
],
"response_type": "text",
"selection_policy": "sequential"
}
]
},
"parent": "node_9_1590437082562",
"conditions": "$webhook_result_1",
"dialog_node": "response_9_1590437151924"
},
{
"type": "event_handler",
"output": {
"generic": [
{
"values": [
{
"text": "Sorry.Let's try again"
}
],
"response_type": "text",
"selection_policy": "sequential"
}
]
},
"parent": "slot_6_1589776478754",
"context": {
"date": null,
"time": null,
"confirm": null
},
"conditions": "@reply:no",
"event_name": "filled",
"dialog_node": "handler_9_1589778393058",
"previous_sibling": "handler_5_1589778380097"
},
{
"type": "event_handler",
"output": {
"text": {
"values": [
"Perfect!"
],
"selection_policy": "sequential"
}
},
"parent": "slot_6_1589776478754",
"conditions": "@reply:yes",
"event_name": "filled",
"dialog_node": "handler_5_1589778380097",
"previous_sibling": "handler_8_1589776478757"
},
{
"type": "event_handler",
"output": {
"text": {
"values": [
"Let me confirm: You want an appointment for <?
$date.reformatDateTime(\"EEEEE\") ?> at <?
$time.reformatDateTime(\"h a\") ?>. Is this
correct?"
],
"selection_policy": "sequential"
}
},
"parent": "slot_6_1589776478754",
"event_name": "focus",
"dialog_node": "handler_8_1589776478757",
"previous_sibling": "handler_7_1589776478757"
},
{
"type": "event_handler",
"output": {},
"parent": "slot_6_1589776478754",
"context": {
"reply": "@reply && slot_in_focus"
},
"conditions": "@reply && slot_in_focus",
"event_name": "input",
"dialog_node": "handler_7_1589776478757"
},
{
"type": "response_condition",
"output": {
"generic": [
{
"values": [
{
"text": "To get to our business from the Empire State Building, walk to Herald Square and take the N train to Union Square"
}
],
"response_type": "text",
"selection_policy": "sequential"
}
```

```
]
},
"parent": "node_6_1589771493060",
"conditions": "@landmark:(empire state building)",
"dialog_node": "response_1_1589771523116",
"previous_sibling": "node_3_1589772173647"
},
{
"type": "response_condition",
"output": {
"generic": [
{
"values": [
{
"text": "To get to our business from the Empire
State Building, walk to Herald Square and take the
N train to Union Square."
},
{
"text": "To get to our business from Times Square,
take the N train downtown to Union Square"
},
{
"text": "To get to our business from Grand Central,
take the 4,5 or 6 train downtown to Union Square."
},
{
"text": "To get to our business from london eye, take
the 2 or 6 train downtown to bridge."
}
],
"response_type": "text",
"selection_policy": "multiline"
}
]
},
"parent": "node_6_1589771493060",
"conditions": "#Customer_Assistant_location",
"dialog_node": "response_1_1590525503572",
"previous_sibling": "response_3_1589771606300"
},
{
"type": "response_condition",
"output": {
```

```
"text": {
"values": [
"To get to our business from london eye, take the 2 or 6 train downtown to bridge."
],
"selection_policy": "sequential"
}
},
"parent": "node_6_1589771493060",
"conditions": "@landmark:(london eye)",
"dialog_node": "response_3_1589771606300",
"previous_sibling": "response_7_1589771586226"
},
{
"type": "response_condition",
"output": {
"text": {
"values": [
"To get to our business from Times Square, take the N train downtown to Union Square"
],
"selection_policy": "sequential"
}
},
"parent": "node_6_1589771493060",
"conditions": "@landmark:(times square)",
"dialog_node": "response_8_1589771563213",
"previous_sibling": "response_1_1589771523116"
},
{
"type": "response_condition",
"output": {
"text": {
"values": [
"To get to our business from Grand Central, take the 4,5 or 6 train downtown to Union Square."
],
"selection_policy": "sequential"
}
},
"parent": "node_6_1589771493060",
"conditions": "@landmark:(grand central)",
"dialog_node": "response_7_1589771586226",
"previous_sibling": "response_8_1589771563213"
},
{
"type": "standard",
"title": "#provide_location",
"output": {
"generic": [
{
"values": [
{
"text": "We're located by Union Square on the corner of 13th and Broadway"
}
],
"response_type": "text",
"selection_policy": "sequential"
}
]
},
"parent": "node_6_1589771493060",
"conditions": "true",
"dialog_node": "node_3_1589772173647"
},
{
"type": "response_condition",
"output": {
"text": {
"values": [
"Our hours are Monday to Friday 10am to 8pm and Saturday 11am to 6pm."
],
"selection_policy": "sequential"
}
},
"parent": "node_5_1589684905181",
"conditions": "true",
"dialog_node": "response_3_1589760146099",
"previous_sibling": "response_6_1589760114598"
},
{
"type": "response_condition",
"output": {
```

```
"text": {

"values": [

"We are open on @holidays as per regular hours"

],

"selection_policy": "sequential"

}

},

"parent": "node_5_1589684905181",

"conditions": "@holidays",

"dialog_node": "response_8_1589760694177",

"previous_sibling": "response_3_1589760146099"

},

{

"type": "response_condition",

"output": {

"text": {

"values": [

"We are open on <?
@sys-date.reformatDateTime(\"EEEEE\") ?> from
10am until 8pm"

],

"selection_policy": "sequential"
```

```
}

},

"parent": "node_5_1589684905181",

"conditions":
"@sys-date.reformatDateTime(\"EEEEE\") ==
\"Monday\" ||
@sys-date.reformatDateTime(\"EEEEE\") ==
\"Tuesday\" ||
@sys-date.reformatDateTime(\"EEEEE\") ==
\"Wednesday\" ||
@sys-date.reformatDateTime(\"EEEEE\") ==
\"Thursday\" ||
@sys-date.reformatDateTime(\"EEEEE\") ==
\"Friday\"",

"dialog_node": "response_10_1589760058890",

"previous_sibling": "response_1_1589760010823"

},

{

"type": "response_condition",

"output": {

"text": {

"values": [

"Our hours on <?
@sys-date.reformatDateTime(\"EEEEE\") ?> are
11am to 6pm."
```

```
],
"selection_policy": "sequential"
}
},
"parent": "node_5_1589684905181",
"conditions":
"@sys-date.reformatDateTime(\"EEEEE\") ==
\"Saturday\" ||
@sys-date.reformatDateTime(\"EEEEE\") ==
\"Sunday\"",
"dialog_node": "response_6_1589760114598",
"previous_sibling": "response_10_1589760058890"
},
{
"type": "response_condition",
"output": {
"generic": [
{
"values": [
{
"text": "We are closed on @holidays"
}
],
"response_type": "text",
"selection_policy": "sequential"
}
]
},
"parent": "node_5_1589684905181",
"conditions": "@holidays:christmas || @holidays:(new year)",
"dialog_node": "response_1_1589760010823"
},
{
"type": "event_handler",
"output": {
"text": {
"values": [
"Thanks"
],
"selection_policy": "sequential"
}
},
"parent": "slot_6_1589776517570",
"conditions": "true",
"event_name": "filled",
"dialog_node": "handler_7_1589778632634",
"previous_sibling": "handler_9_1589776517572"
},
{
"type": "event_handler",
"output": {
"text": {
"values": [
"I'll just need a phone to hold your reservation"
],
"selection_policy": "sequential"
}
},
"parent": "slot_6_1589776517570",
"event_name": "focus",
"dialog_node": "handler_9_1589776517572",
"previous_sibling": "handler_3_1589776517572"
},
{
"type": "event_handler",
"output": {},
"parent": "slot_6_1589776517570",
"context": {
"phone": "@phone"
},
"conditions": "@phone",
"event_name": "input",
"dialog_node": "handler_3_1589776517572"
},
{
"type": "response_condition",
"output": {
"text": {
"values": [
"Let me check availability... and also you can visit us anytime on our store hours"
],
"selection_policy": "sequential"
}
},
"parent": "node_10_1589776099437",
"disabled": true,
"conditions": "true",
"dialog_node": "response_5_1589778753378",
"previous_sibling": "handler_9_1590525786951"
},
{
"type": "slot",
"output": {},
"parent": "node_10_1589776099437",
"variable": "$date",
"dialog_node": "slot_6_1589776288149",
"previous_sibling": "response_1_1590525732653"
},
{
"type": "slot",
"output": {},
"parent": "node_10_1589776099437",
"variable": "$phone",
"dialog_node": "slot_6_1589776517570",
"previous_sibling": "slot_6_1589776478754"
},
{
"type": "response_condition",
"output": {
"text": {
"values": [],
"selection_policy": "sequential"
}
},
"parent": "node_10_1589776099437",
"disabled": true,
"dialog_node": "response_1_1590525732653",
"previous_sibling": "response_5_1589778753378"
},
{
"type": "slot",
"output": {},
"parent": "node_10_1589776099437",
"variable": "$reply",
"dialog_node": "slot_6_1589776478754",
"previous_sibling": "slot_1_1589776339852"
},
{
"type": "slot",
"output": {},
"parent": "node_10_1589776099437",
"variable": "$time",
"dialog_node": "slot_1_1589776339852",
"previous_sibling": "slot_6_1589776288149"
},
{
"type": "event_handler",
"parent": "node_10_1589776099437",
"event_name": "generic",
"dialog_node": "handler_9_1590525786951"
},
{
"type": "event_handler",
"output": {
"text": {
"values": [
"What time on <?
$date.reformatDateTime(\"EEEEE\") ?> do you
want to come in?"

],

"selection_policy": "sequential"

}

},

"parent": "slot_1_1589776339852",

"event_name": "focus",

"dialog_node": "handler_1_1589776339856",

"previous_sibling": "handler_3_1589776339856"

},

{

"type": "event_handler",

"output": {

"text": {

"values": [

"We only accept appointments between 11am and
5pm"

],

"selection_policy": "sequential"

}
```

```
},
"parent": "slot_1_1589776339852",
"next_step": {
"behavior": "reprompt"
},
"conditions": "$time.after('17:30:30') ||
$time.before('10:59:59')",
"event_name": "filled",
"dialog_node": "handler_9_1589778592381",
"previous_sibling": "handler_1_1589776339856"
},
{
"type": "event_handler",
"output": {},
"parent": "slot_1_1589776339852",
"context": {
"time": "@sys-time"
},
"conditions": "@sys-time",
"event_name": "input",
"dialog_node": "handler_3_1589776339856"
```

```
},
{
"type": "event_handler",
"output": {
"text": {
"values": [
"What day would you like to come?"
],
"selection_policy": "sequential"
}
},
"parent": "slot_6_1589776288149",
"event_name": "focus",
"dialog_node": "handler_8_1589776288164",
"previous_sibling": "handler_2_1589776288164"
},
{
"type": "event_handler",
"output": {
"text": {
"values": [
"Looks like you're trying to make a reservation in
the past. Try again."
],
"selection_policy": "sequential"
}
},
"parent": "slot_6_1589776288149",
"conditions": "$date.before(now())",
"event_name": "filled",
"dialog_node": "handler_2_1589778548041",
"previous_sibling": "handler_8_1589776288164"
},
{
"type": "event_handler",
"output": {},
"parent": "slot_6_1589776288149",
"context": {
"date": "@sys-date"
},
"conditions": "@sys-date",
"event_name": "input",
"dialog_node": "handler_2_1589776288164"
},
{
"type": "standard",
"title": "Store_hours",
"metadata": {
"_customization": {
"mcr": true
}
},
"next_step": {
"behavior": "jump_to",
"selector": "condition",
"dialog_node": "node_3_1589772173647"
},
"conditions": "#Customer_Assistant_hours || @holidays",
"dialog_node": "node_5_1589684905181",
"previous_sibling": "node_5_1589661264481"
},
{
```

```
"type": "standard",
"title": "Product information",
"actions": [
{
"name": "main_webhook",
"type": "webhook",
"parameters": {
"input": "<?input.text?>"
},
"result_variable": "webhook_result_1"
}
],
"metadata": {
"_customization": {
"mcr": true
}
},
"conditions": "#Product_information",
"dialog_node": "node_9_1590437082562",
"previous_sibling": "node_5_1589777521523"
},
{
"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": "random"
}
```

```
]
},
"conditions": "anything_else",
"dialog_node": "Anything else",
"previous_sibling": "node_9_1590437082562",
"disambiguation_opt_out": true
},
{
"type": "standard",
"title": "Thanks",
"output": {
"generic": [
{
"values": [
{
"text": "You're welcome. Just let me know if you need anything else"
},
{
"text": "No problem. Just let me know if you need anything else"
},
{
"text": "My pleasure. Just let me know if you need anything else"
}
],
"response_type": "text",
"selection_policy": "sequential"
}
]
},
"conditions": "#Thanks",
"dialog_node": "node_5_1589777521523",
"previous_sibling": "node_9_1589777100512"
},
{
"type": "standard",
"title": "Make_appointment",
"output": {
"text": {
"values": [
" you can visit us anytime during our store hours"
```

```
],
"selection_policy": "sequential"
}
},
"metadata": {
"_customization": {
"mcr": false
}
},
"conditions": "#Customer_Assistant_appointment",
"dialog_node": "node_10_1589776099437",
"previous_sibling": "node_6_1589771493060"
},
{
"type": "standard",
"title": "Greetings",
"metadata": {
"_customization": {
"mcr": true
}
},
"conditions": "#Customer_Assistant_Greetings || @Greetings",
"digress_in": "does_not_return",
"dialog_node": "node_5_1589661264481",
"previous_sibling": "Welcome"
},
{
"type": "standard",
"title": "Store_location",
"metadata": {
"_customization": {
"mcr": true
}
},
"next_step": {
"behavior": "skip_user_input"
},
"conditions": "#Customer_Assistant_location || @landmark",
"dialog_node": "node_6_1589771493060",
"previous_sibling": "node_5_1589684905181"
},
{
"type": "standard",
"title": "Help",
"output": {
"generic": [
{
"values": [
{
"text": "I can tell you about our store locations and opening hours, or help you set up an appointment."
},
{
"text": "Yeah sure please feel free to ask"
}
],
"response_type": "text",
"selection_policy": "sequential"
}
]
},
"conditions": "#Help",
"dialog_node": "node_6_1589777887891",
"previous_sibling": "node_10_1589776099437"
},
{
"type": "standard",
"title": "Goodbye",
"output": {
"generic": [
{
"values": [
{
"text": "See ya"
},
{
"text": "Bye.Have a nice day"
},
{
"text": "Bye,come soon"
},
{
"text": "Long time"
```

```
}
],
"response_type": "text",
"selection_policy": "random"
}
]
},
"conditions": "#Goodbye",
"dialog_node": "node_9_1589777100512",
"previous_sibling": "node_6_1589777887891"
},
{
"type": "standard",
"title": "Welcome",
"output": {
"generic": [
{
"values": [
{
"text": "Hello, Welcome to Customer Assistant
chatbot. I can provide you with information about
```

```
our store's locations and hours, directions, and even
make appointments. \nHow may I help you?"
}
],
"response_type": "text",
"selection_policy": "random"
}
]
},
"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": "Customer Care",

"language": "en",

"description": "Customer Helpdesk using Watson Assistant"

}
```

B. Reference

1.https://developer.ibm.com/patterns/enhance-customer-help-desk-with-smart-documentunderstanding/

2.https://github.com/IBM/watson-discovery-sdu-with-assistant

3.https://www.youtube.com/watch?v=-yniuX-Poyw&feature=youtu.be

4.https://developer.ibm.com/tutorials/how-to-create-a-node-red-starter-application/

YOUTUBE LINK:

https://youtu.be/8UV_ciSJgjQ

GITHUB LINK:https://github.com/SmartPracticeschool/llSPS-INT-334-Intelligent-Customer-Help-Desk-with-Smart-Document-Understanding