Creating chatbot with Google Dialogflow

1. **Chatbot Fundamentals**
   1. **What is a chatbot?**

**A chatbot is a software application used to conduct an on-line chat conversation via text or text-to-speech, in lieu of providing direct contact with a live human agent.Today, most chatbots are accessed on-line via website popups, or through virtual assistants such as Google Assistant, Amazon Alexa, or messaging apps such as Facebook Messenger or WeChat.**



Image Courtesy: <https://expertsystem.com/chatbot/>

* 1. **Uses of chatbots:**

**Here are some of the use case of chatbot:-**

1. **Chatbots Answer Questions And Inquiries**
2. **Book Tickets To Events/Shows With Chatbots**
3. **Use Chatbots To Find Products, Check Inventory and Recommend Items**
4. **Chatbots To Build Remarkable Customer Experience**
5. **Chatbots Can Confirm Orders And Track Shipping**
6. **Chatbots Can Do Quizzes, Promotions, And Contests With Customers**
7. **Chatbots Become Personal Shopping Assistants**
   1. **Advantages of Chatbots**
8. **1- 24 hour availability**
9. **Keeping Up with the Trends: Being Present on Messaging Platforms**
10. **Improved Customer Service**
11. **Increased Customer Engagement**
12. **Cost Savings**
13. **Management of multiple clients**
14. **Gaining a deeper understanding of customers**
15. **Google Dialogflow**
    1. **Introduction:**

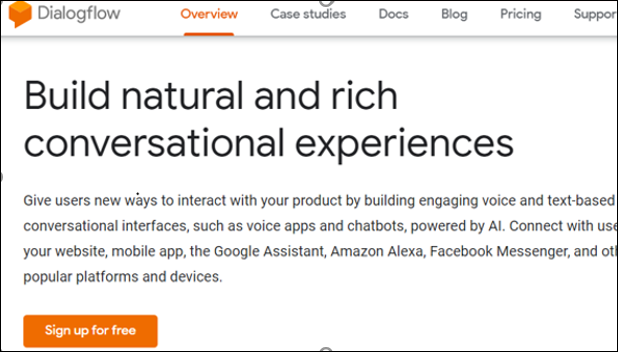
**Dialogflow is an end-to-end, build-once deploy-everywhere development suite for creating conversational interfaces for websites, mobile applications, popular messaging platforms, and IoT devices. You can use it to build interfaces (such as chatbots and conversational IVR) that enable natural and rich interactions between your users and your business**

**Dialogflow can analyze multiple types of input from your customers, including text or audio inputs (like from a phone or voice recording). It can also respond to your customers in a couple of ways, either through text or with synthetic speech.**

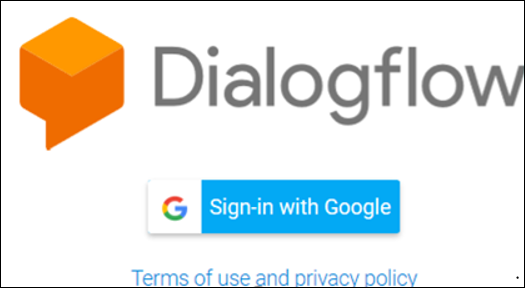
* 1. **Signup For Dialogflow:**

**You need to have a google account to signup for Dialogflow.**

1. **Go to** <https://dialogflow.com/> **and click on ‘Sign Up for Free’ button.**



1. **Click ‘sign-in with Google’.**



1. **Select your google account and once you are redirected, click on ‘Go To Console’ on the upper right corner of the screen.**
   1. **Dialogflow Console:**

**Dialogflow provides a web user interface called the *Dialogflow Console*** ([open console](https://dialogflow.cloud.google.com/)). **The console is used to create, build, manage, fine-tune, and test your agents. With it, you can**:

* **Create**[**agents**](https://cloud.google.com/dialogflow/docs/agents-overview)**that define the conversational experienceCreate Intents**
* **Create**[**intents**](https://cloud.google.com/dialogflow/docs/intents-overview)**that map user input to responses**
* **Create** [**entities**](https://cloud.google.com/dialogflow/docs/entities-overview) **to extract useful data from user input**
* **Control conversation paths with** **[contexts](https://cloud.google.com/dialogflow/docs/contexts-overview)**
* [**Integrate**](https://cloud.google.com/dialogflow/docs/integrations) **with other conversational platforms**
* **Implement** **[fulfillment](https://cloud.google.com/dialogflow/docs/fulfillment-overview)** **to connect your service when using integrations**
* [**Analyze**](https://cloud.google.com/dialogflow/docs/analytics) **agent performance**

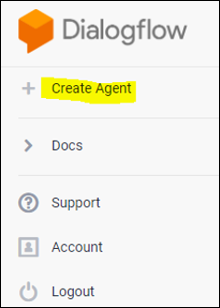
1. **Building a chatbot using google Dialogflow**
   1. **The problem statement:**

**To build a chatbot which can answer all the queries of a customer and whenever a customer does an enquiry, it automatically send the customer the course details. Also an email is sent to the support team to assist the customer futher with their queries.**

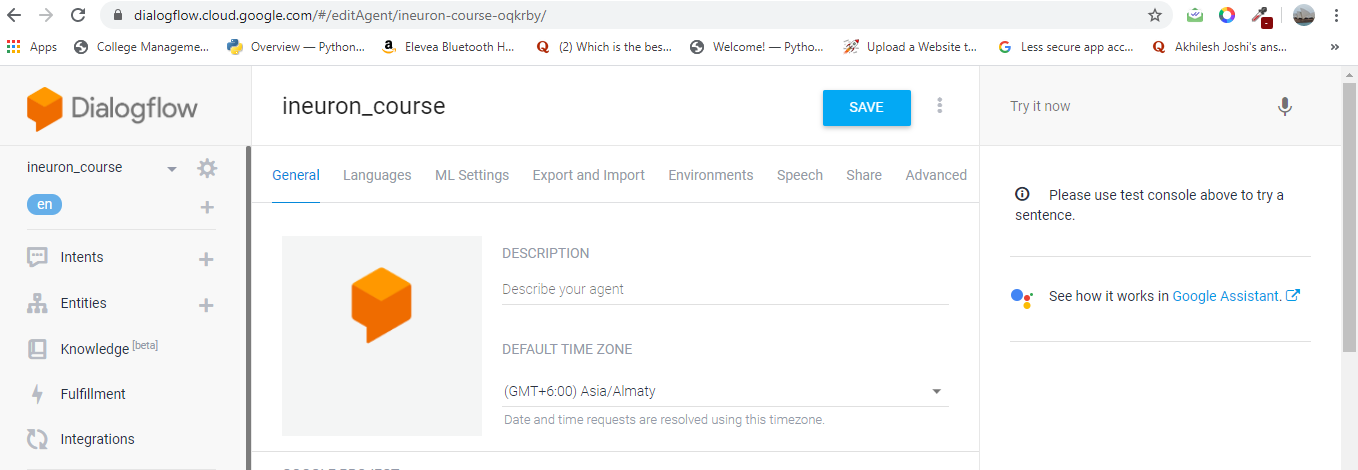
* 1. **Agent:**

**A Dialogflow *agent* is a virtual agent that handles conversations with your end-users. It is a natural language understanding module that understands the nuances of human language. A Dialogflow agent is similar to a human call center agent. You train them both to handle expected conversation scenarios, and your training does not need to be overly explicit.**

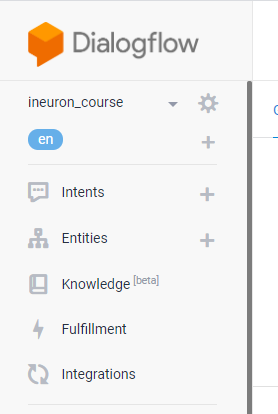
* + 1. **Creating an Agent:**
  1. **Click on *Create Agent* from the left menu.**



* 1. **Provide the name of the agent and click on the *SAVE* button to create the Agent.**



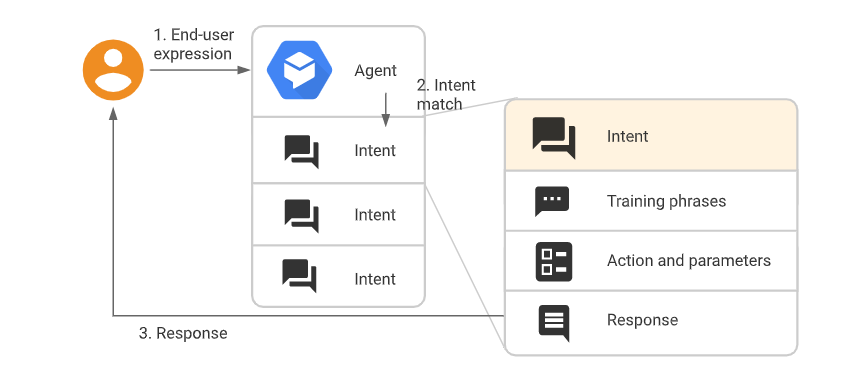
* 1. **After saving, the agent is shown in the left hand side of your console. You can click the gear icon to edit the agent settings.**



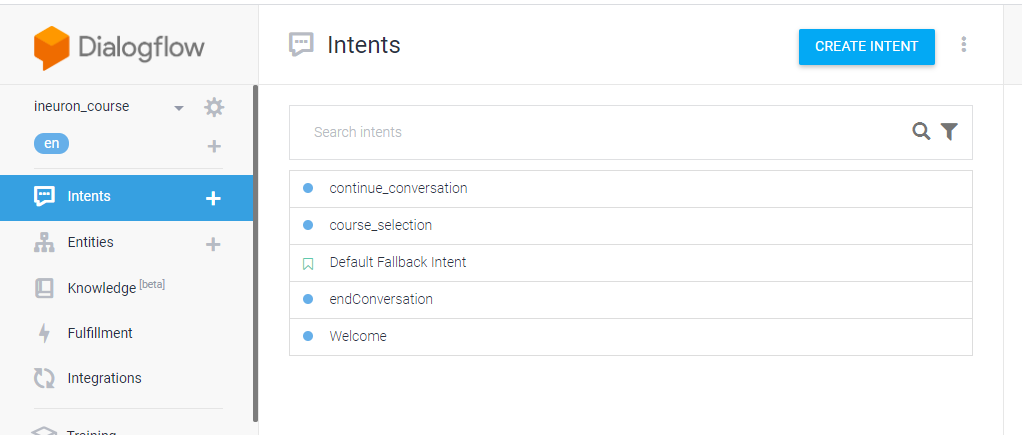
* 1. **Intent:**

**An *intent* categorizes an end-user's intention for one conversation turn. For each agent, you define many intents, where your combined intents can handle a complete conversation. When an end-user writes or says something, referred to as an *end-user expression*, Dialogflow matches the end-user expression to the best intent in your agent. Matching an intent is also known as *intent classification*.**

**The following diagram shows the basic flow for intent matching and responding to the end-user:**

****

* + 1. **Creating an Intent:**
    2. **Click the + add button next to**Intents**in the left sidebar menu. Enter a name for your intent. Your intent name should represent the end-user expressions it recognizes and click** Save**.**



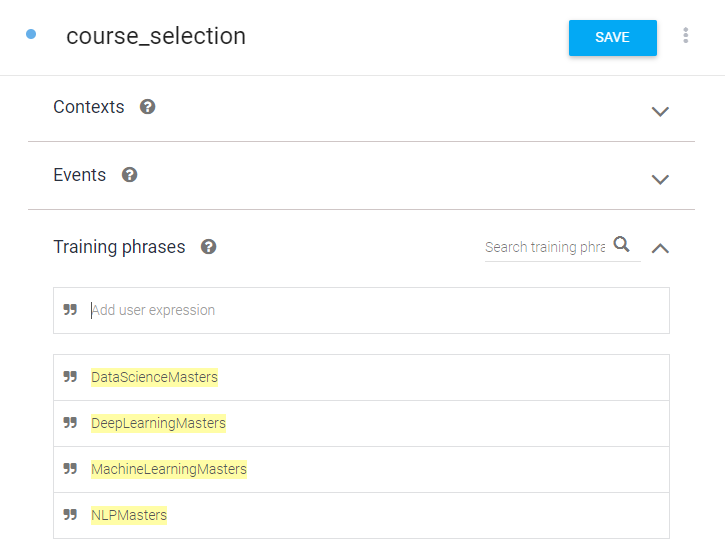
* + - 1. **Training Phrases:**

***Training phrases* are example phrases for what end-users might type or say, referred to as *end-user expressions*. For each intent, you create many training phrases. When an end-user expression resembles one of these phrases, Dialogflow matches the intent.**

**For example, the training phrase "I want pizza" trains your agent to recognize end-user expressions that are similar to that phrase, like "Get a pizza" or "Order pizza".**

**Adding the training Phrases:**

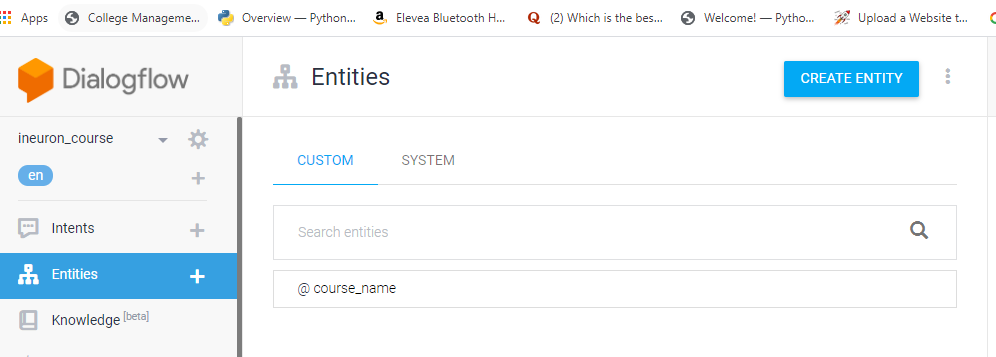
1. **Click the text field that shows "Add user expression".**
2. **Type your training phrases and press the Enter key after each.**



* + - 1. **Extracting the Entities:**

**When an intent is matched at runtime, Dialogflow provides the extracted values from the end-user expression as *parameters*. Each parameter has a type, called the**[**entity type**](https://cloud.google.com/dialogflow/docs/entities-overview)**, which dictates exactly how the data is extracted. Unlike raw end-user input, parameters are structured data that can easily be used to perform some logic or generate responses.**

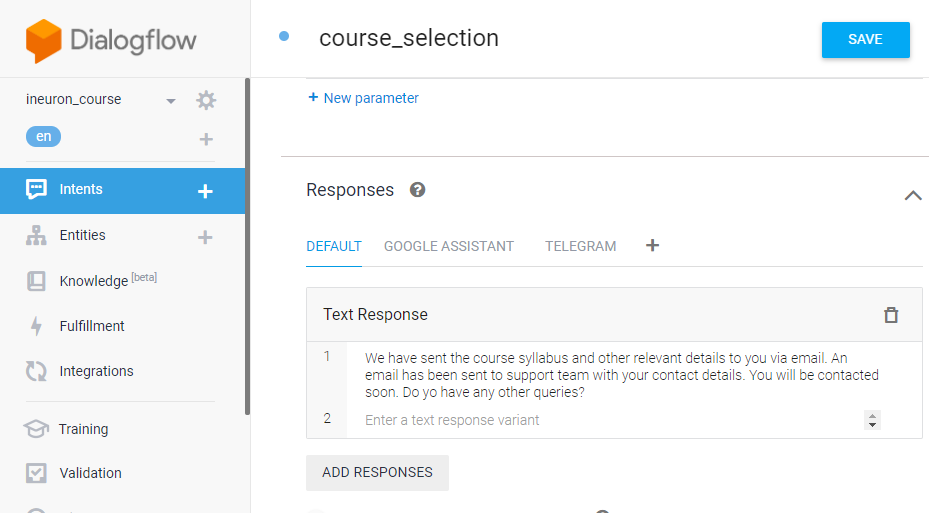
**Each intent**[**parameter**](https://cloud.google.com/dialogflow/docs/intents-actions-parameters#params)**has a type, called the *entity type*, which dictates exactly how data from an end-user expression is extracted.**



**Dialogflow provides predefined**[**system entities**](https://cloud.google.com/dialogflow/docs/entities-system)**that can match many common types of data. For example, there are system entities for matching dates, times, colors, email addresses, and so on. You can also create your own**[**custom entities**](https://cloud.google.com/dialogflow/docs/entities-custom)**for matching custom data.**

* + - 1. **Specifying custom Responses:**

**Intents have a built-in response handler that can return responses after the intent is matched. This feature only supports static responses, though you can use**[**parameter references**](https://cloud.google.com/dialogflow/docs/intents-actions-parameters#values)**in these responses to make them somewhat dynamic. This is helpful for recapping information provided by the end-user. For example, your intent response could look like: "Okay, I booked a room for you on $date".**

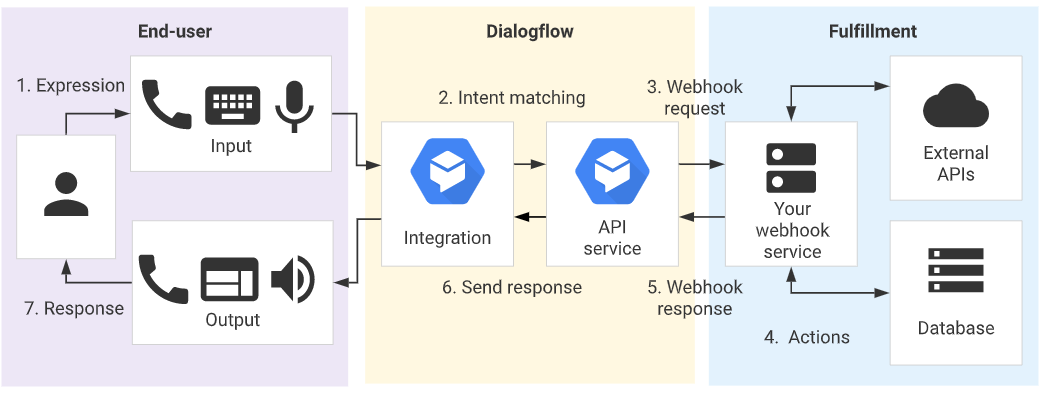


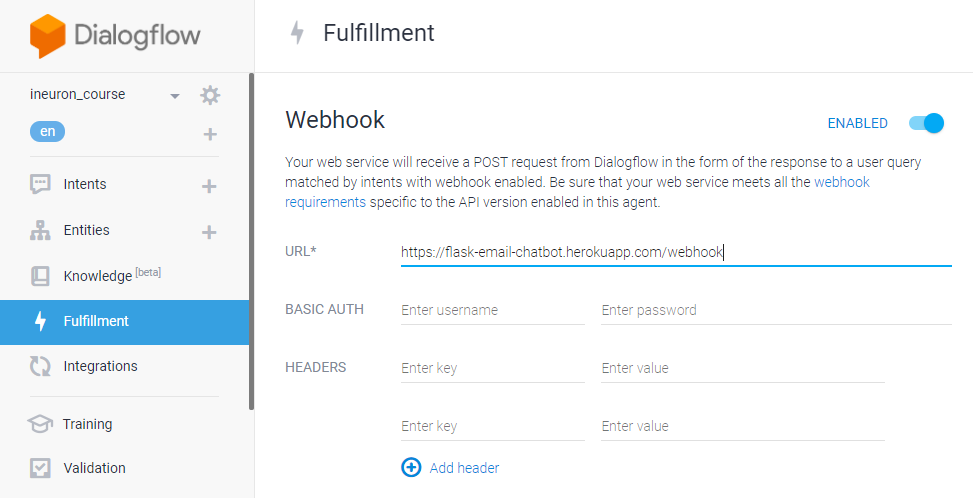
* 1. **Fulfillment:**

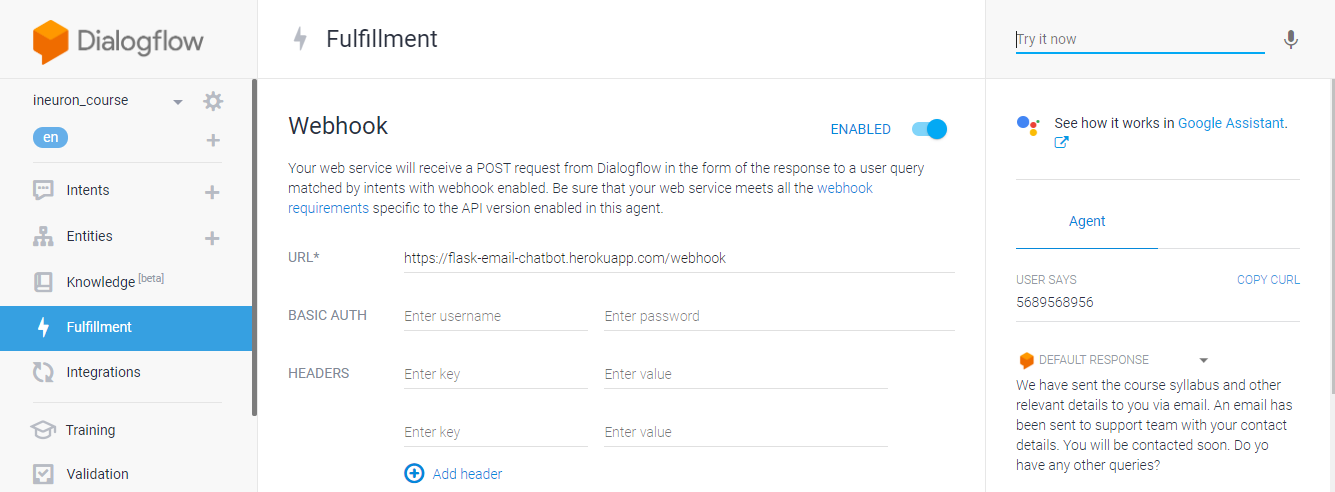
**By default, your agent responds to a matched intent with a static response. If you're using one of the**[**integration**](https://cloud.google.com/dialogflow/docs/integrations)**options, you can provide a more dynamic response by using *fulfillment*. When you enable fulfillment for an intent, Dialogflow responds to that intent by calling a service that you define. For example, if an end-user wants to schedule a haircut on Friday, your service can check your database and respond to the end-user with availability information for Friday.**

**Each**[**intent**](https://cloud.google.com/dialogflow/docs/intents-overview)**has a setting to enable fulfillment. If an intent requires some action by your system or a dynamic response, you should enable fulfillment for the intent. If an intent without fulfillment enabled is matched, Dialogflow uses the static response you defined for the intent.**

**When an intent with fulfillment enabled is matched, Dialogflow sends a request to your *webhook* service with information about the matched intent. Your system can perform any required actions and respond to Dialogflow with information for how to proceed. The following diagram shows the processing flow for fulfillment.**

****

1. **The end-user types or speaks an expression.**
2. **Dialogflow matches the end-user expression to an intent and extracts parameters.**
3. **Dialogflow sends a [webhook request](https://cloud.google.com/dialogflow/docs/fulfillment-webhook" \l "webhook_request) message to your webhook service. This message contains information about the matched intent, the action, the parameters, and the response defined for the intent.**
4. **Your service performs actions as needed, like database queries or external API calls.**
5. **Your service sends a**[**webhook response**](https://cloud.google.com/dialogflow/docs/fulfillment-webhook#webhook_response)**message to Dialogflow. This message contains the response that should be sent to the end-user.**
6. **Dialogflow sends the response to the end-user.**
7. **The end-user sees or hears the response.**
   * 1. **Functionalities achived in fulfilment:**
   1. **Sending an email to the customer with the syllabus and all the course details based on the course selected.**
   2. **Sending an email to the Support team to further contact the customer for further clarification.**
      1. **Webhook for fulfilment:**
      2. **Select**Fulfillment**in the left sidebar menu.**
      3. **Toggle the**Webhook**field to**Enabled**.**
      4. **Provide the details for your webhook service in the form. If your webhook doesn't require authentication, leave the authentication fields blank.**
      5. **Click**Save**at the bottom of the page.**
      6. **Fulfilment Request and Response:**

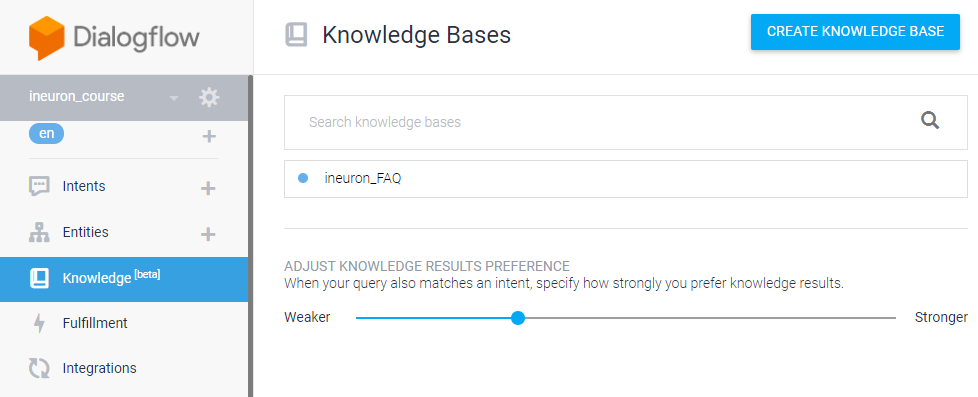


* 1. **Knowledge base:**

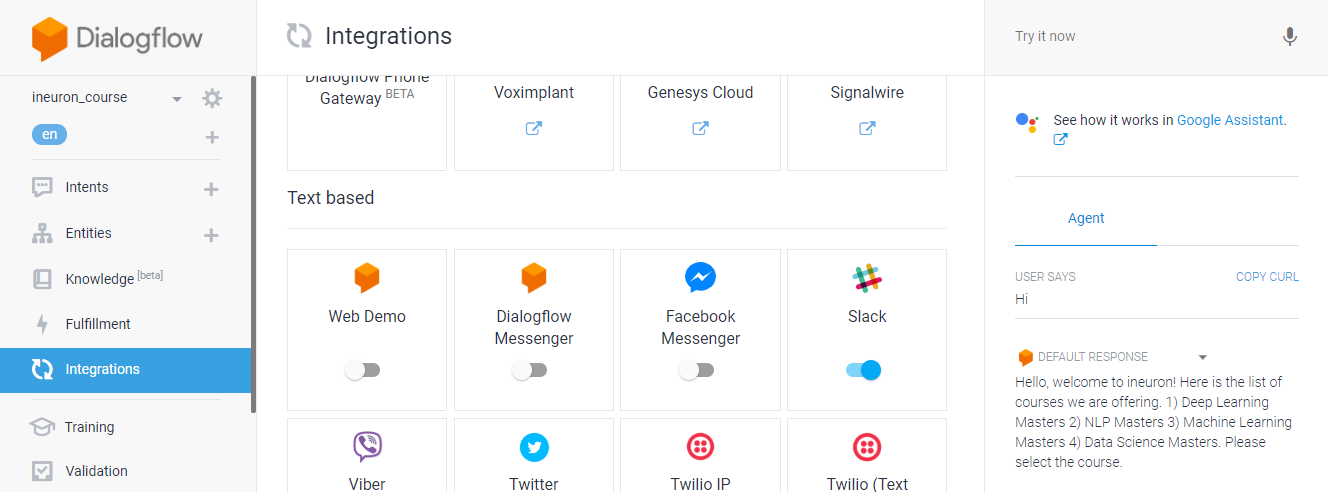
***Knowledge connectors* complement defined intents. They parse *knowledge documents* (for example, FAQs or articles) to find automated responses. To configure them, you define one or more**[**knowledge bases**](https://cloud.google.com/dialogflow/docs/knowledge-bases)**, which are collections of knowledge documents.**

**For this project, we have created a knowledge document which contains all the FAQs of the sudents and we have integrated the same with the bot.**

* + 1. **Creating a Knowledge base**

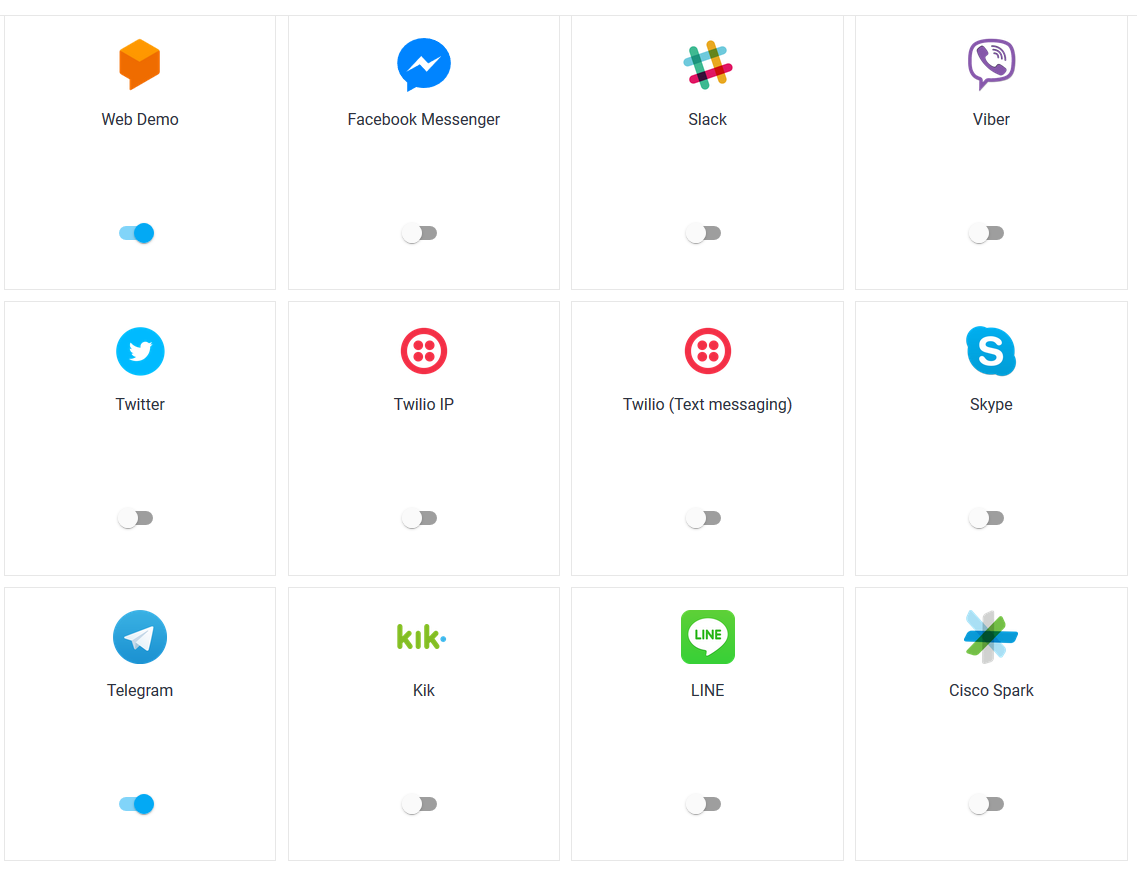
1. **To create a knowledge base, click on ‘Knowledge’ on the left hand side of your console and then click ‘CREATE KNOWLEDGE BASE’.**
2. **Give the name of your knowledge base and click ‘Save’.**
3. **Once created, you’ll see a message stating that no knowledge document has been created yet. Click ‘create the first one’ to start creating your document.**
4. **Select FAQ and text/CSV and the upload the FAQ CSV file created and then click Create.**
5. **Scroll down to the**Responses**section and add responses as desired:**
   * + - **When defining the first response for the intent, use $Knowledge.Question[1] and $Knowledge.Answer[1] where you want the question and answer to be supplied.**
       - **The index for $Knowledge.Question and $Knowledge.Answer starts at 1, so increase this index when adding more responses.**
6. **Click SAVE once you are done editing.**
   1. **Test the Agent:**

**You can now test the agent from the ‘test it now’ section.**



* 1. **Integration:**

**Dialogflow integrates with many popular conversation platforms like Google Assistant, Slack, and Facebook Messenger. If you want to build an agent for one of these platforms, you should use one of the many *integrations* options. Direct end-user interactions are handled for you, so you can focus on building your agent.**

****

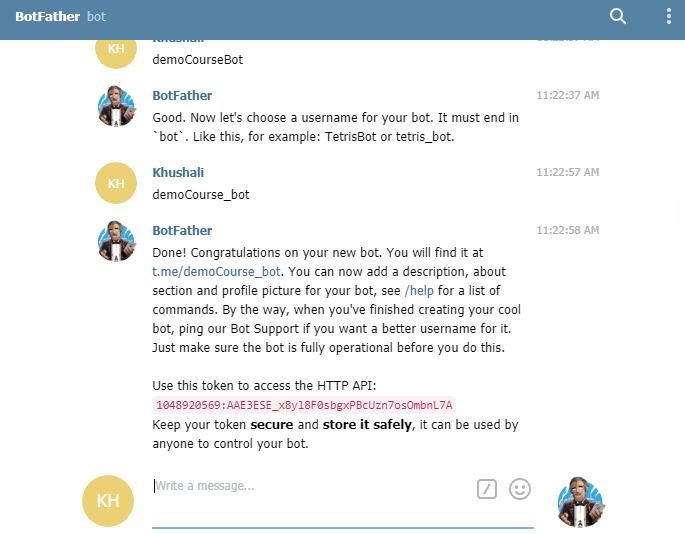
* + 1. **Integration with Telegram:**

**Dialogflow Telegram Integration allows you to easily create Telegram bots with natural language understanding based on the Dialogflow technology.**

**In order to set up the Telegram integration for your agent, you'll need the following:**

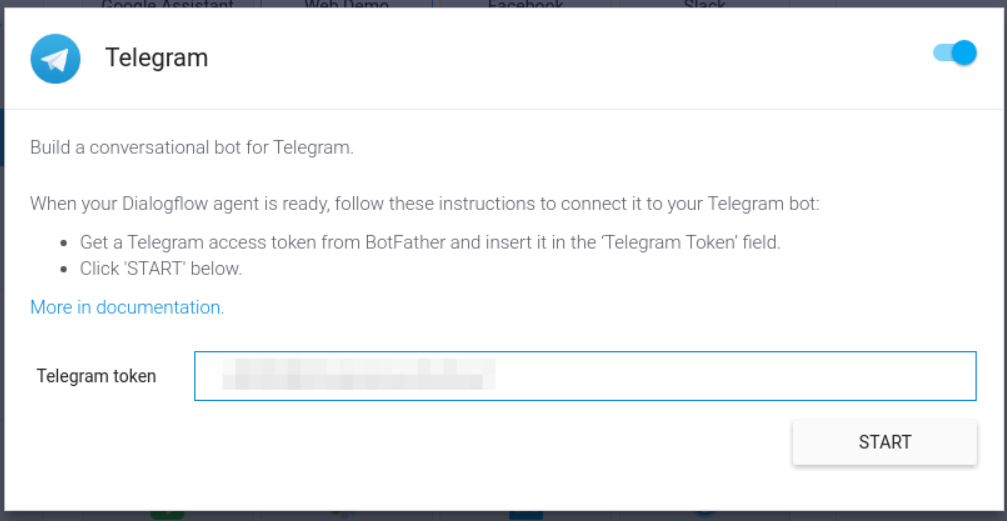
1. [Telegram account](https://web.telegram.org/#/login).

### **3.7.1.2 Creating a Bot in Telegram**

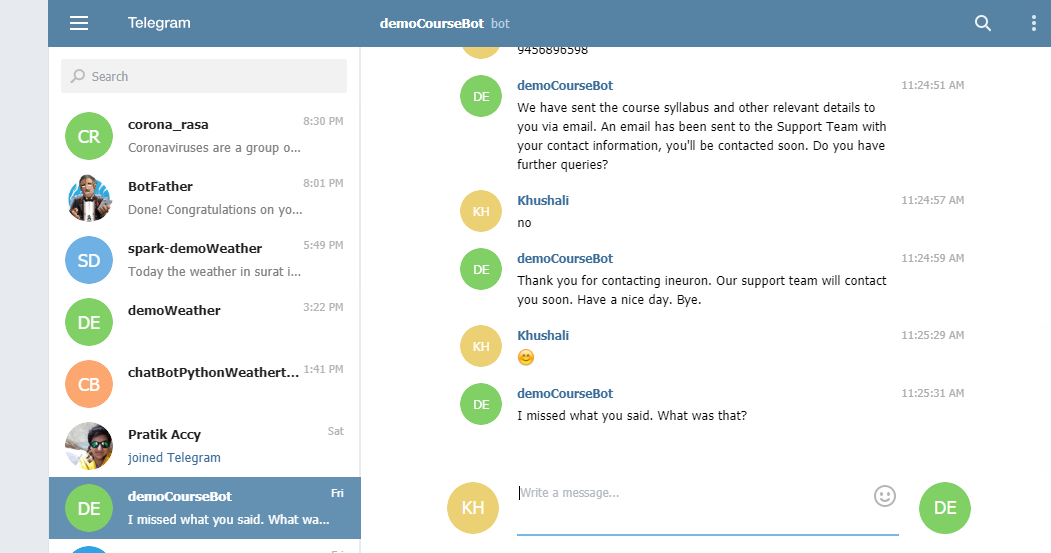
1. **Login to Telegram and go to** <https://telegram.me/botfather>
2. **Click the**Start**button in the web interface or type /start**
3. **Click on or type**/newbot**and enter a name**
4. **Enter a username for the bot, ending in "bot" (e.g. garthsweatherbot)**
5. **Copy the generated access token**

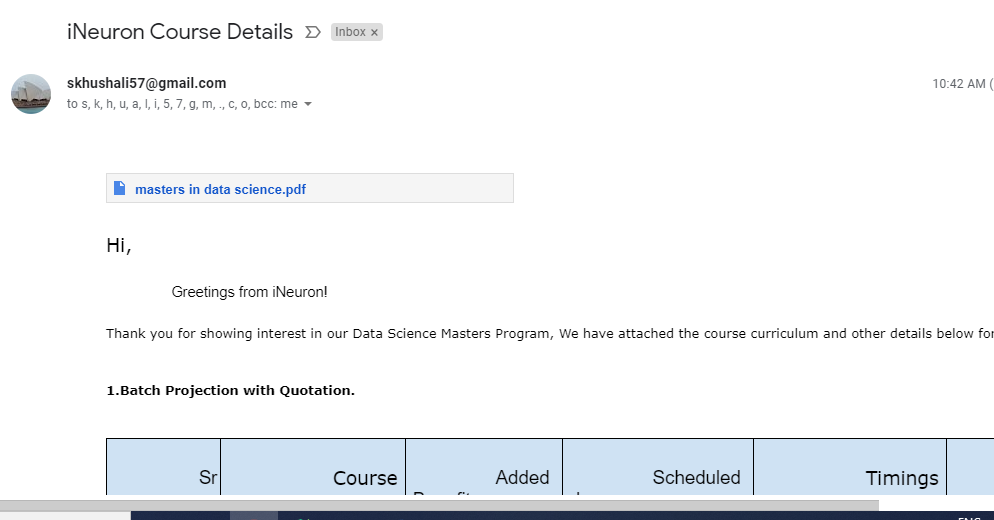
## **3.7.1.3 Setting Up Dialogflow**

1. **In Dialogflow, go to Integrations in the left hand menu**
2. **Click on the Telegram tile**
3. **Paste the Access Token into the related field**
4. **Click the Start button**

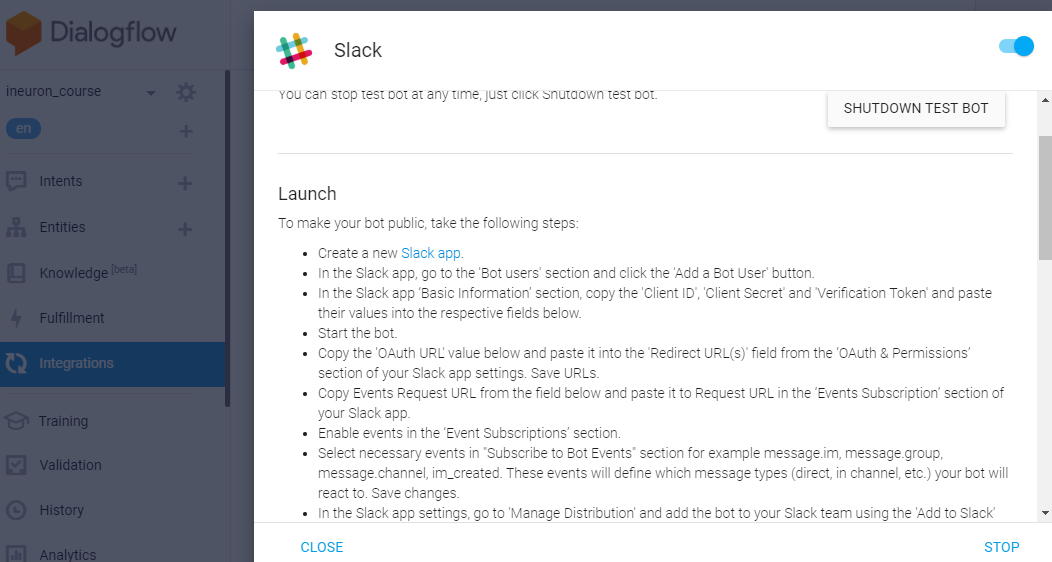
****

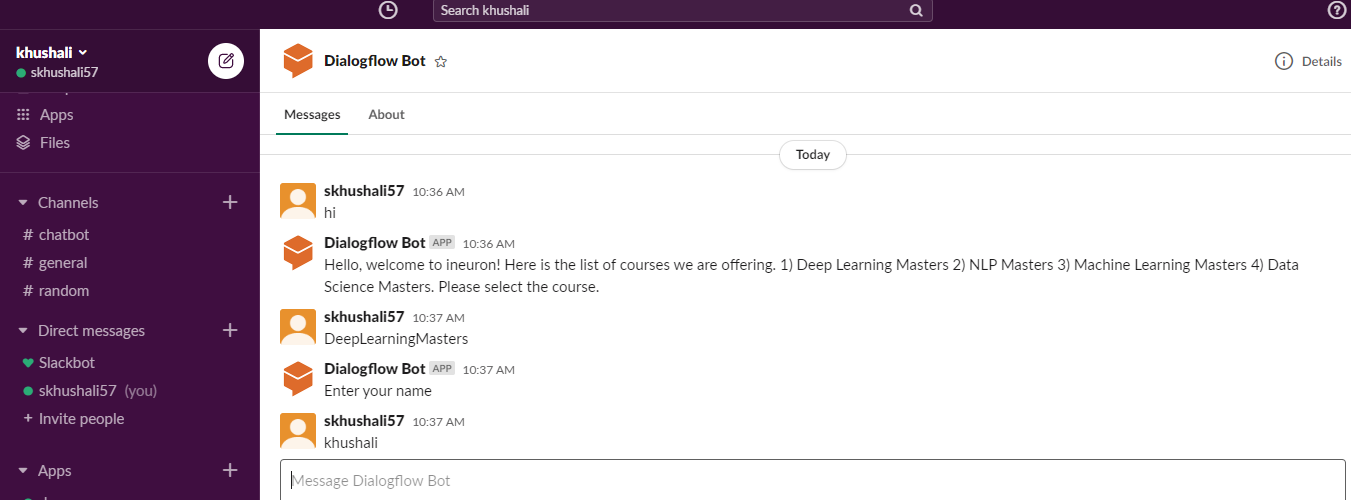
**Now you can search your bot in telegram and you can chat with it.**

**Telegram id:- demoCourseBot**



* + 1. **Integration with Slack:**

**Follow below mentioned steps for sloack integration:-**

Slack working chat-bot:-

