



Build your First Facebook Messenger Bot with IBM Watson Assistant

AP Cloud 2018 Workshop

AP Cloud Team
Miracle Software Systems, Inc.

Build Your First Facebook Messenger Bot with IBM Watson Assistant

Introduction

This document contains a step-by-step process of creating a Chatbot with IBM Watson Assistant NLP and will teach you how to create a chatbot with Watson Assistant and how to integrate with Facebook Messenger using Node JS

This guide was prepared by [Miracle's Innovation Labs](#)

Pre-Requisites

All attendees must have their workstation (with Internet) to participate in the lab (Both PC and MAC are compatible). The following pre-requisites will help you to make the Hands-on Lab experience easier.

- Active email ID for registering with IBM Bluemix to get access for Watson Assistant
- Download and install Node JS and ngrok
- Access for Facebook and Facebook for Developers
- Test Editor such as Sublime Text (or) Notepad++

Technology Involved

- IBM Bluemix (PaaS)
- NLP - IBM Watson Assistant
- Server Side - Node JS

Labs Steps

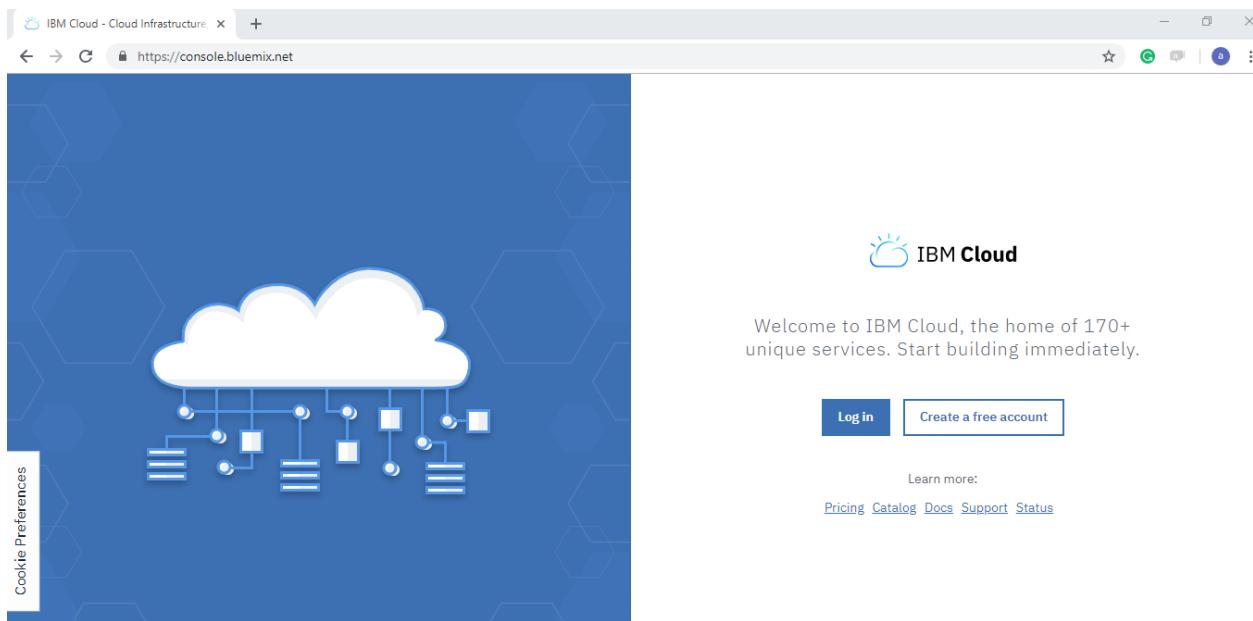
So, let us get started with the bot!

The following steps will outline how you can create a Miracle Bot using Watson Assistant and integrate it with Facebook Messenger. Users will be able to directly message your bot through Facebook Messenger to know about Miracle Software Systems and AP Cloud.

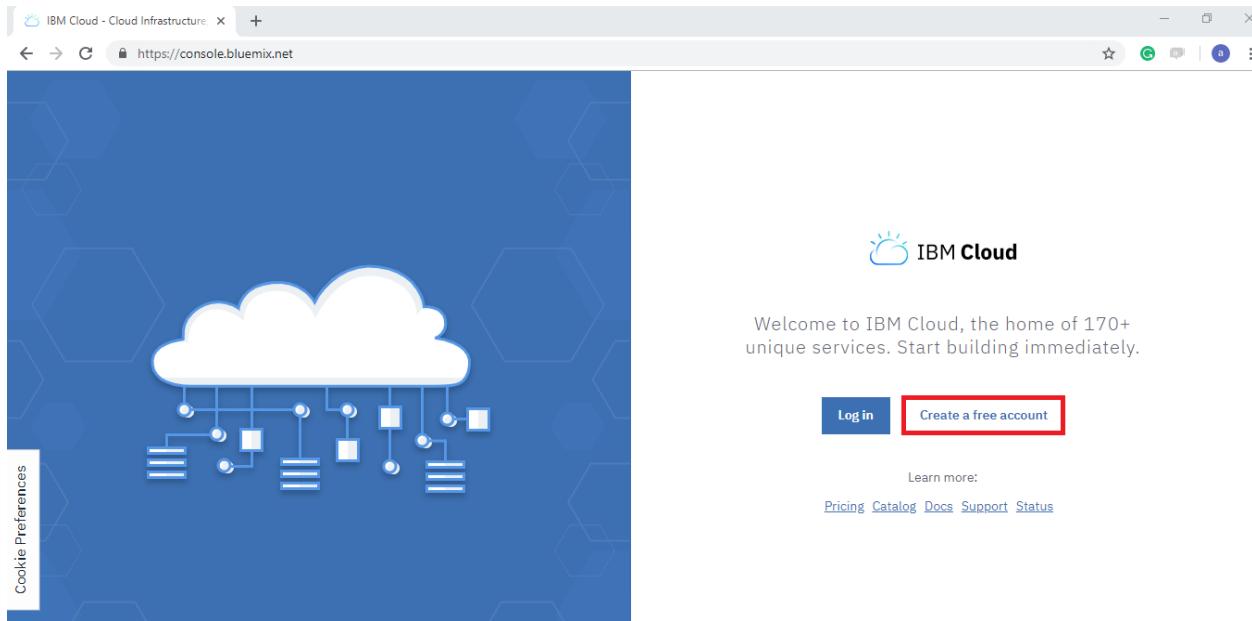
Step #1 | Create Watson Assistant Service in IBM Bluemix

The first step will be to make sure that we have access to the IBM Bluemix console with either the free trial option (or) the paid subscription option.

Login to Bluemix at <http://bluemix.net> (or) Register today at, <https://console.bluemix.net/registration>

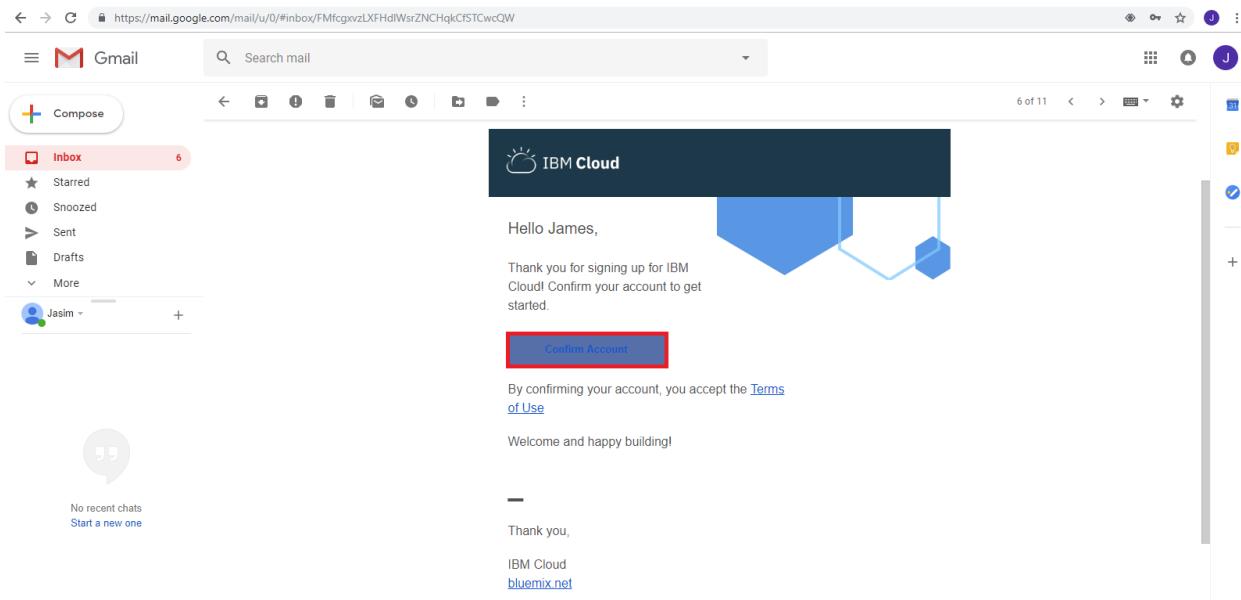


If you are a new user, click on **Create a free account**.



Now, you need to provide the details to the fields that are marked as required and click on **Create Account**.

After creating account, confirmation mail will be sent to the registered mail id.



Click on **Confirm Account** and then Login to your Bluemix account.

The screenshot shows the 'Log in to IBM' page. It features a header with the 'IBM' logo. Below the header, there are two sections: 'Don't have an IBMid?' with a 'Create an account' link, and 'Log in with your company credentials (SSO)' with a 'Contact the IBM Help Desk' link. The main login form has fields for 'IBMid' (containing 'healthcarebot12345@gmail.com'), 'Forgot IBMid?', 'Password' (containing '*****'), 'Forgot password?', and a 'Remember me' checkbox. To the right of the password field is a 'Log in' button with a red border. At the bottom of the page, there is a footer with links: 'Contact', 'Privacy', 'Terms of use', and 'Accessibility'.

You can see the dashboard as shown below. Now, click on **Catalog**.

https://console.bluemix.net/dashboard/apps/

IBM Cloud Catalog Docs Support Manage Search for resource... James smith's Account

Dashboard

RESOURCE GROUP CLOUD FOUNDRY ORG CLOUD FOUNDRY SPACE LOCATION CATEGORY Filter by resource name... Create resource

All Resources All Organizations All Spaces All Locations All Categories

Fast-track your app development

Want to go from prototype to production with the click of a button? Hit the ground running with one of our IBM Cloud starter kits. You can also check out our solution tutorials that provide real-world cloud solution examples.

Build with Watson

Starter Kit · Lite Services · IBM Accelerate your AI development to build smarter solutions.

Create enterprise-level web apps

Starter Kit · Lite Services · IBM Develop and deploy cloud-native apps within minutes.

Learn to build, deploy, and scale

Tutorials · Lite Services · IBM Use IBM Cloud to implement common patterns based on best practices and proven technologies.

[View Watson starter kits](#) [View App Service starter kits](#) [View Solution tutorials](#)

FEEDBACK

In Catalog, under the **AI** category you can see all the Watson Services where you can find **Watson Assistant**. Click on **Watson Assistant** tile for training the chatbot.

https://console.bluemix.net/catalog/?search=label:lite&category=ai

IBM Cloud Catalog Docs Support Manage Search for resource... James smith's Account

Catalog

label:lite Filter

All Categories (52) AI

- Compute (10)
- Containers (1)
- Networking
- Storage (1)
- AI (14)**
- Analytics (4)
- Databases (3)
- Developer Tools (7)
- Integration (2)
- Internet of Things (1)
- Security and Identity (3)
- Starter Kits (2)
- Web and Mobile (2)
- Web and Application (2)

Watson Assistant (formerly Conversation) Lite · IBM

Watson Assistant is a platform that allows developers and non-technical users to collaborate on building conversational AI...

AI OpenScale Lite · IBM

IBM AI OpenScale is an enterprise-grade environment for AI infused applications that provides enterprises with visibility into how AI...

Discovery Lite · IBM

Add a cognitive search and content analytics engine to applications.

Knowledge Catalog Lite · IBM

Discover, catalog, and securely share enterprise data.

Knowledge Studio Lite · IBM

Teach Watson the language of your domain.

Language Translator Lite · IBM

Translate text, documents, and websites from one language to another. Create industry or region-specific translations via the service's...

Machine Learning Lite · IBM

Natural Language Understanding Lite · IBM

Personality Insights Lite · IBM

FEEDBACK

Now, type a unique name for the service instance in the **Service name** field. For example, type **my-conversation-demo**. Leave the default values for the other options and click on **Create**.

<https://console.bluemix.net/catalog/services/watson-assistant-formerly-conversation>

IBM Cloud Catalog Docs Support Manage Search for resource... James smith's Account

View all Watson Assistant (formerly Conversation) Lite • IBM

Watson Assistant a platform that allows developers and non-technical users to collaborate on building conversational AI-powered assistants. Its powerful visual dialog editor, industry leading natural language processing capabilities, flexible API, and improvement and recommendation features allow for the rapid creation and management of anything from simple chatbots to complex enterprise grade solutions for customer service and more.

Service name: Watson Assistant Demo

Choose a region/location to deploy in: Sydney Select a resource group: Default

Pricing Plans Monthly prices shown are for country or region: India

PLAN	FEATURES	PRICING
Lite	10,000 Messages/Month AI-Based Intent and Entity Recognition	Free

View Docs Terms AUTHOR IBM PUBLISHED 11/07/2018 Need Help? Contact IBM Cloud Support Estimate Monthly Cost Cost Calculator Create

Note - Creation of service may take up to a minute or two.

You will land on the **Manage** page of the service dashboard. Click on **Show Credentials** to make use of API key.

<https://console.bluemix.net/services/conversation/crn%3Av1%3Abluemix%3Apublic%3Aconversation%3Aau-syd%3Aa%2F8e3637fad2b948fc8198185ab45bdad7%3...>

IBM Cloud Catalog Docs Support Manage Search for resource... James smith's Account

Watson Services / Assistant / Watson Assistant - Demo Resource Group: Default Location: Sydney

Get started by launching the tool. Plan: Lite Upgrade now and receive \$200 in credit

Launch tool Getting started tutorial API reference

Credentials Show Credentials

API Key: Url: https://gateway-syd.watsonplatform.net/assistant/api

Copy the API key and navigate to **app.js** file which is in your local drive and paste the API key in the place of <Your-Watson-Assistant-APIKey> at line #20. Click on launch tool for landing into Skills page.

Watson Services / Assistant /

Watson Assistant - Demo

Resource Group: Default Location: Sydney

Get started by launching the tool.

Plan: Lite [Upgrade now and receive \\$200 in credit](#)

[Launch tool](#) [Getting started tutorial](#) [API reference](#)

Credentials [Show Credentials](#)

API Key: [REDACTED]

Url: <https://gateway-syd.watsonplatform.net/assistant/api>

Step #2 | Create Skills in Assistant Service in IBM Bluemix

Your first step in the Watson Assistant tool is to create a **Skill**.

From the home page of the Watson Assistant tool, click on the **Skills** tab or click on the **Create a Skill** button to build your chatbot as shown below,

IBM Watson Assistant

Home Skills Assistants Instance Watson Assistant - Demo Change

IBM Watson Assistant

Build cognitive virtual assistants that help your customers when they need it most. See what's new [↗](#)

Three easy steps

Follow these steps to create an assistant.

- 1 Build a skill

A skill contains the training data and machine learning logic that enable your chat bot to understand and help customers. Add training data built by IBM or use graphical tools that help you build your own.

[Learn more ↗](#)
- 2 Deploy with an assistant

An assistant orchestrates the flow of information between a skill and your customers. Add your skill to an assistant, and then deploy the assistant to the media channel of your choice.

[Learn more ↗](#)
- 3 Analyze and improve

Use metrics collected from conversations with real customers to iteratively improve your skill.

[Learn more ↗](#)

Get Started Now

[Create a Skill](#)

Then you will be navigated to skills tab, where you can create a new skill. Click on **Create new** button as shown below,

The screenshot shows the 'Skills' section of the IBM Watson Assistant interface. At the top, there's a navigation bar with links for 'Home', 'Skills', and 'Assistants'. On the right, it says 'Instance Watson Assistant - Demo' and 'Change'. Below the navigation, the word 'Skills' is prominently displayed. A sub-section titled 'Time to create a Skill' is shown, with a note explaining that a skill is a container for AI that enables an assistant to help customers. A blue 'Create new' button is highlighted with a red box.

Once you click on **Create new** button in the skills tab, you will need to provide a name and description to your skill. After entering all details, click on **Create** button.

The screenshot shows the 'Add Dialog Skill' creation form. At the top, there are three tabs: 'Create skill' (highlighted with a red box), 'Use sample skill', and 'Import skill'. The 'Name' field contains 'AP Cloud Demo Bot'. The 'Description (optional)' field contains 'Here is the sample dialog-model for the AP Cloud Demo Bot'. The 'Language' dropdown is set to 'English (US)'. A blue 'Create' button is highlighted with a red box at the bottom right.

If you want to import the existing skill click on the **Import Skill** tab.

The screenshot shows the 'Add Dialog Skill' interface. At the top, there are three tabs: 'Create skill', 'Use sample skill', and 'Import skill', with 'Import skill' being the active tab and highlighted with a red box. Below the tabs, a note says 'Select the JSON file for the dialog skill with the data you want to import and choose the artifacts to import to the new skill.' A 'Choose JSON File' button is present. Underneath, two radio buttons are shown: 'Everything (Intents, Entities, and Dialog)' (selected) and 'Intents and Entities'. At the bottom right is a blue 'Import' button.

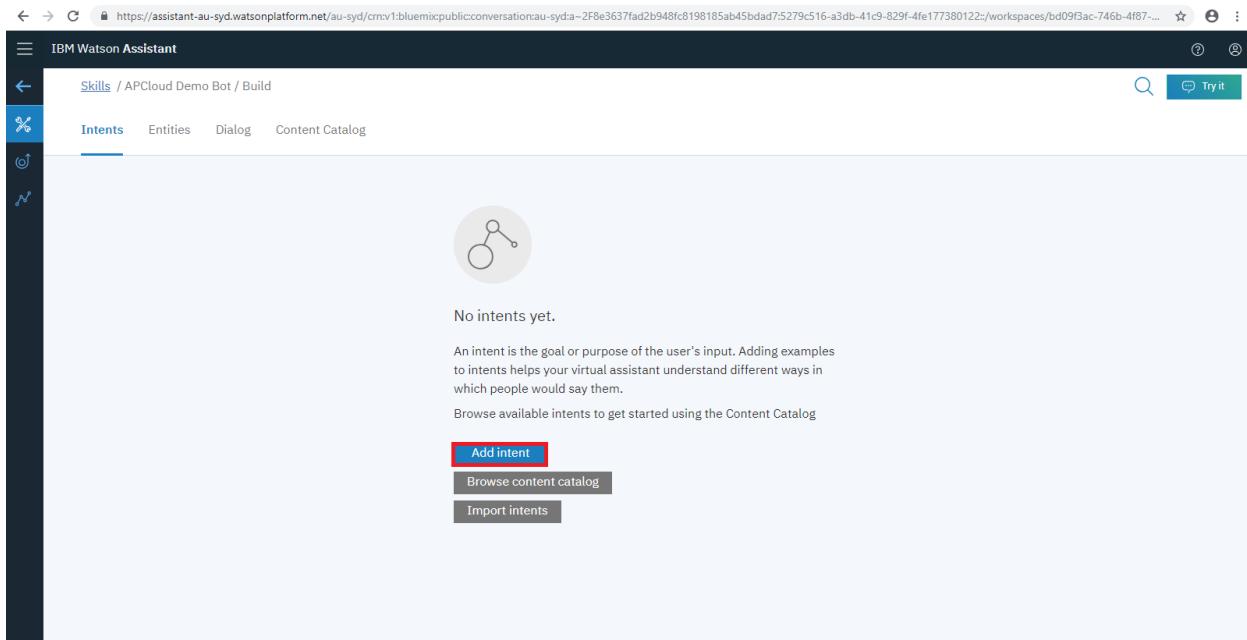
Now choose any of the existing Watson Assistant Skill which is in **JSON** format to upload and click on **Import** as shown below,

This screenshot is identical to the one above, showing the 'Add Dialog Skill' interface with the 'Import skill' tab selected. The 'Import' button at the bottom right is now highlighted with a red box.

Step #3 | Training the Skill using Intents, Entities and Dialogs

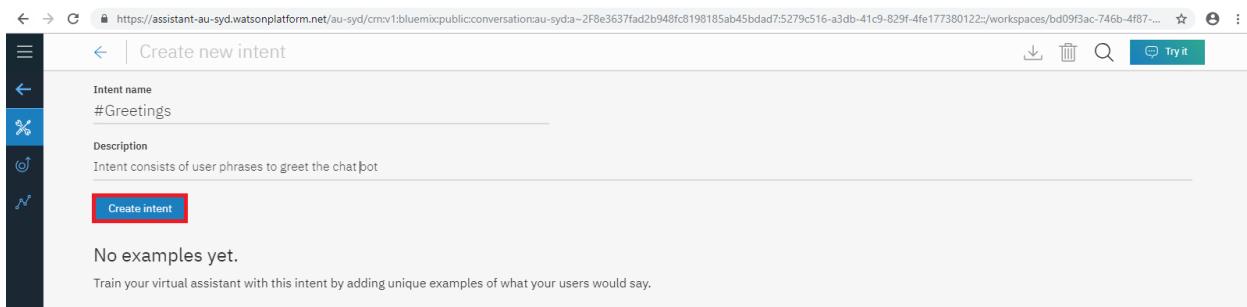
a) Intents

Now, you will land on the Intents page of your new skill. Intents are purpose or goals expressed in a user's input. Click on **Add Intents** to define new Intents for your chatbot to get trained or else browse pre-defined Intents from **Content Catalog** or if you have any existing intents import that **JSON** file.



The screenshot shows the IBM Watson Assistant interface. At the top, there's a navigation bar with back, forward, and search icons, followed by the URL: <https://assistant-au-syd.watsonplatform.net/au-syd/cmv1:bluemixpublicconversation:au-syda-2f8e3637fad2b948fc8198185ab45bdad7:5279c516-a3db-41c9-829f-4fe177380122:/workspaces/bd09f3ac-746b-4f87...>. Below the URL is a search bar and a "Try it" button. The main area has a sidebar with icons for skills, entities, dialog, and content catalog. The "Intents" tab is selected. In the center, there's a circular icon with a person and a speech bubble. Below it, the text "No intents yet." is displayed. A descriptive message follows: "An intent is the goal or purpose of the user's input. Adding examples to intents helps your virtual assistant understand different ways in which people would say them." It also says "Browse available intents to get started using the Content Catalog". At the bottom are three buttons: "Add intent" (highlighted with a red border), "Browse content catalog", and "Import intents".

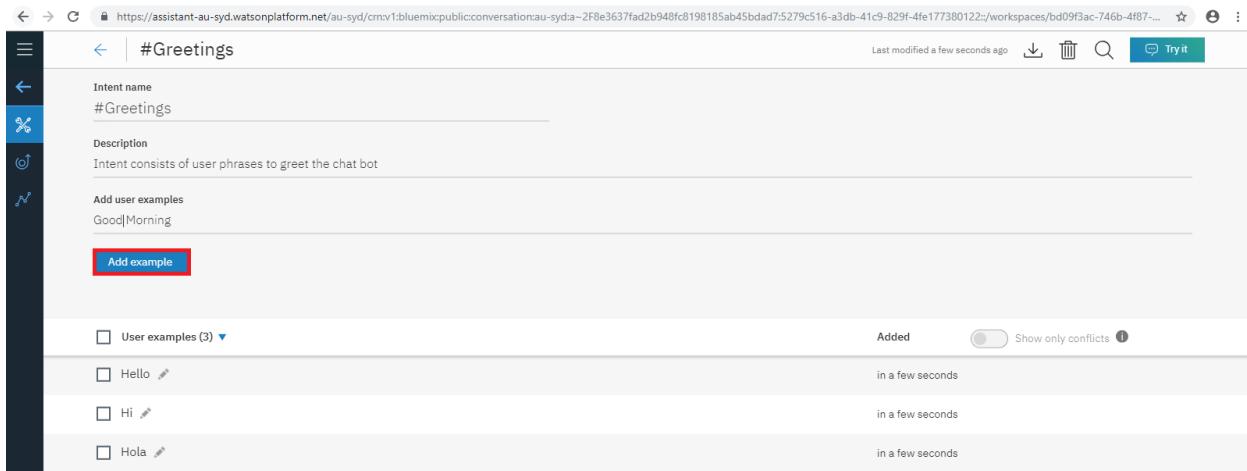
After clicking on **Add intent** button, you need to provide the Intent name, Description, and then click on **Create intent**.



The screenshot shows the "Create new intent" form. At the top, there's a back arrow and a search bar. The main area has two input fields: "Intent name" containing "#Greetings" and "Description" containing "Intent consists of user phrases to greet the chatbot". Below these is a "Create intent" button. At the bottom, the text "No examples yet." is shown, followed by the instruction "Train your virtual assistant with this intent by adding unique examples of what your users would say."

Now, need to add user examples for that intent as shown below,

Note - After each example, you can click on **Add example** button or you can hit the enter button.



Intent name
#Greetings

Description
Intent consists of user phrases to greet the chat bot

Add user examples
GoodMorning

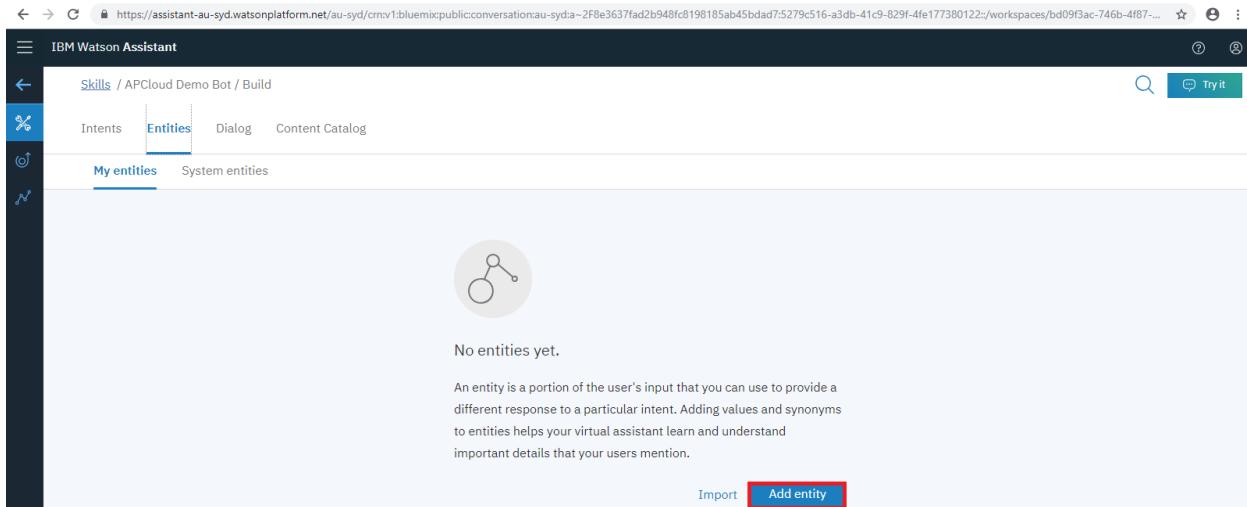
Add example

User examples (3) ▾

	Added	Show only conflicts ⓘ
Hello ↗	in a few seconds	
Hi ↗	in a few seconds	
Hola ↗	in a few seconds	

b) Entities

An entity represents information in the user input that is relevant to the user's purpose. To create an entity, go to **Entities** tab, under **My entities** tab click on **Add entity**.



Skills / APCloud Demo Bot / Build

Entities

Intents Dialog Content Catalog

My entities System entities

No entities yet.

An entity is a portion of the user's input that you can use to provide a different response to a particular intent. Adding values and synonyms to entities helps your virtual assistant learn and understand important details that your users mention.

Import Add entity

In the Entity name field, provide a name for the entity and then click on **Create entity**.

The screenshot shows the 'Create new entity' page in the Watson Assistant interface. The URL is https://assistant-au-syd.watsonplatform.net/au-syd/cmv1:bluemixpublic:conversation:au-syd:a-2F8e3637fad2b948fc8198185ab45bdad7:5279c516-a3db-41c9-829f-4fe177380122:/workspaces/bd09f3ac-746b-4f87-... . The 'Entity name' field contains '@Company'. A red box highlights the 'Create entity' button. On the right, there is a 'Fuzzy Matching' toggle switch set to 'On'. Below the entity name, tabs for 'Dictionary' and 'Annotation BETA' are visible. A message says 'No values yet.' followed by a note: 'Once you've named your entity, begin by adding values, synonyms, and patterns to entities to help your virtual assistant learn and understand important details that your users mention.'

After entity creation, you need to enter a Value name for the entity, click on **Add value** and also you can add any synonyms which are based on contextual similarity for specific values as shown below,

The screenshot shows the entity details page for '@Company'. The URL is https://assistant-au-syd.watsonplatform.net/au-syd/cmv1:bluemixpublic:conversation:au-syd:a-2F8e3637fad2b948fc8198185ab45bdad7:0b8fc765-0a3a-4999-bdd9... . The entity name is '@Company'. The 'Value name' field contains 'Miracle Software System'. Under the 'Synonyms' section, 'Miracle' is listed with a red box around the 'Add synonym...' button. A red box also highlights the 'Add value' button. Below the entity name, tabs for 'Dictionary' and 'Annotation BETA' are visible. A message says 'No values yet.' followed by a note: 'Once you've named your entity, begin by adding values, synonyms, and patterns to entities to help your virtual assistant learn and understand important details that your users mention.'

You can also use predefined entities called System Entities. Click on **System entities** tab to select from a list of common entities that are provided by IBM.

The screenshot shows the IBM Watson Assistant interface. The top navigation bar includes back, forward, and search icons, followed by the URL: <https://assistant-au-syd.watsonplatform.net/au-syd/cmv1/bluemixpublicconversation:au-syda-2F8e3637fad2b948fc8198185ab45bdad7:5279c516-a3db-41c9-829f-4fe177380122:/workspaces/bd09f3ac-746b-4f87-...>. Below the URL is a toolbar with a question mark icon, a gear icon, and a three-dot menu icon.

The main content area has a header "IBM Watson Assistant" and a breadcrumb trail "Skills / APCloud Demo Bot / Build". On the left is a sidebar with icons for skills, intents, entities, dialog, and content catalog. The "Entities" tab is currently selected, and its sub-tab "System entities" is highlighted with a red box.

The main panel displays a list of system entities:

Name (?)	Description	Status
@sys-currency	Extracts currency values from user examples including the amount and the unit. (20 cents)	<input type="checkbox"/> Off
@sys-date	Extracts date mentions (Friday)	<input type="checkbox"/> Off
@sys-location BETA	The @sys-location system entity extracts place names (country, state/province, city, town, etc.) from the user's input. (Boston)	<input type="checkbox"/> Off
@sys-number	Extracts numbers mentioned from user examples as digits or written as numbers. (21)	<input type="checkbox"/> Off
@sys-percentage	Extracts amounts from user examples including the number and the % sign. (15%)	<input type="checkbox"/> Off
@sys-person BETA	The @sys-person system entity extracts names from the user's input. (Anna)	<input type="checkbox"/> Off
@sys-time	Extracts time mentions (at 10)	<input type="checkbox"/> Off

c) Dialog

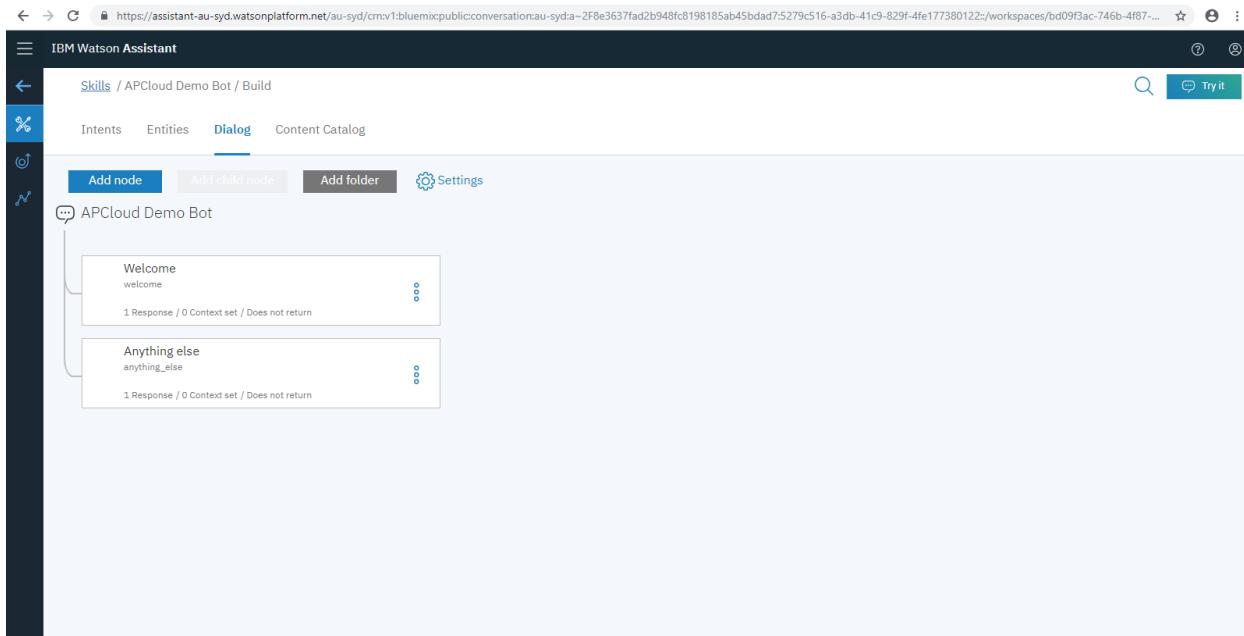
Dialog uses the intents and entities together that are identified in the user's input and respond to the user based on the matched intent. To create your Dialog nodes, click on **Create** button.

The screenshot shows the IBM Watson Assistant interface. The top navigation bar includes back, forward, and search icons, followed by the URL: <https://assistant-au-syd.watsonplatform.net/au-syd/cmv1/bluemixpublicconversation:au-syda-2F8e3637fad2b948fc8198185ab45bdad7:5279c516-a3db-41c9-829f-4fe177380122:/workspaces/bd09f3ac-746b-4f87-...>. Below the URL is a toolbar with a question mark icon, a gear icon, and a three-dot menu icon.

The main content area has a header "IBM Watson Assistant" and a breadcrumb trail "Skills / APCloud Demo Bot / Build". On the left is a sidebar with icons for skills, intents, entities, dialog, and content catalog. The "Dialog" tab is currently selected.

The main panel displays a message bubble icon with a question mark and a plus sign, indicating no dialogs yet. Below it is a brief description: "A dialog uses intents, entities, and context from your application to define a response to each user's input. Creating a dialog defines how your virtual assistant will respond to what your users are saying." At the bottom is a blue "Create" button with a white plus sign.

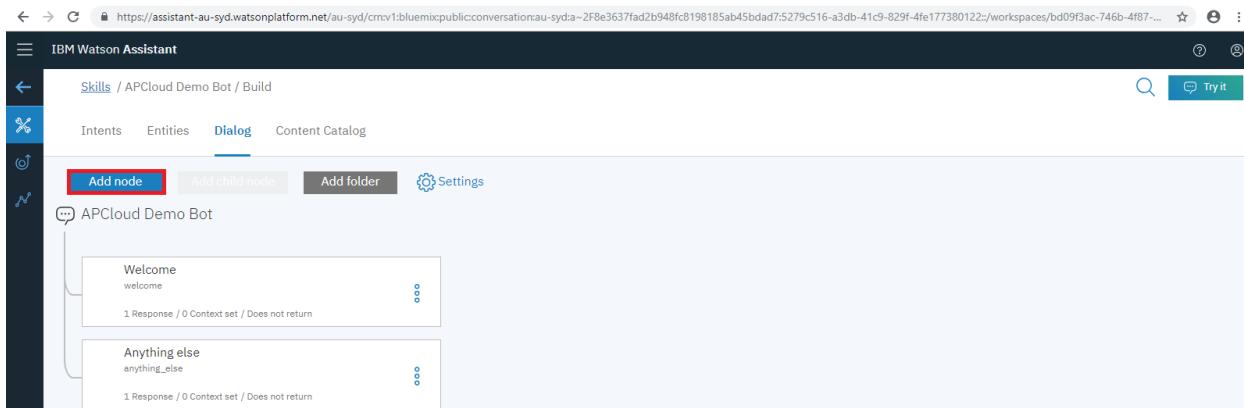
Once you click on **Create** button under the **Dialog** tab, you will get two nodes initially,



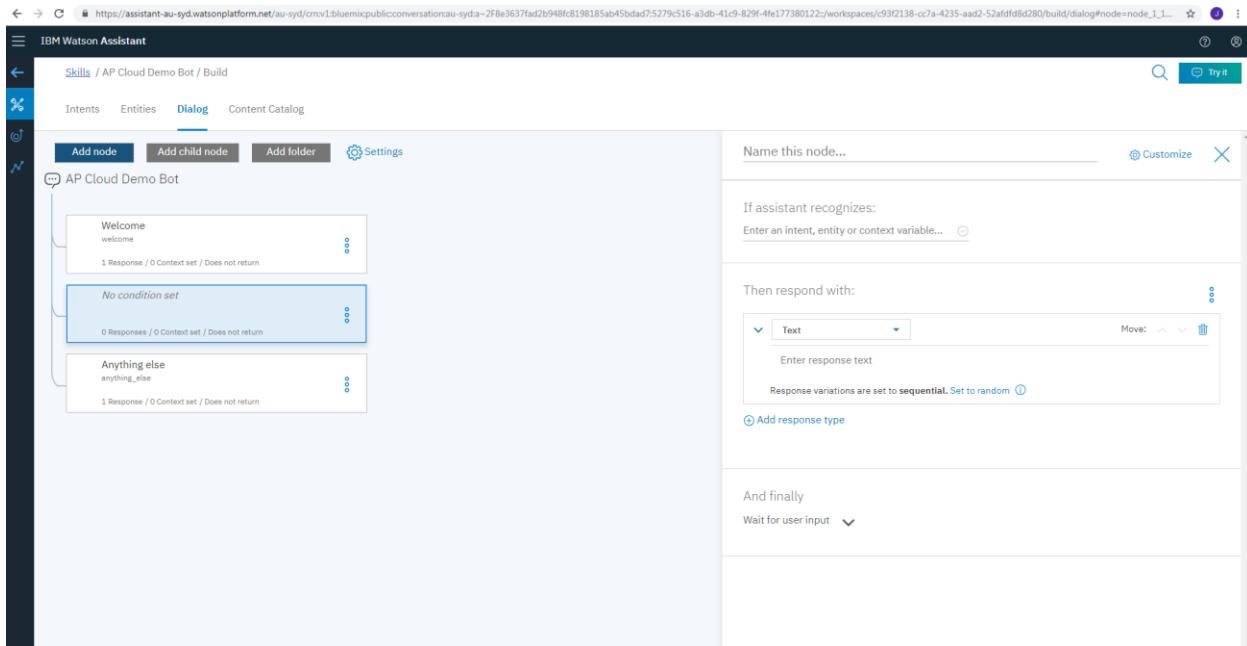
Welcome - This is the first node that contains the initial greetings when users engage with the service.

Anything else - This is the final node that contains phrases which are used to respond to the user when input is not recognized.

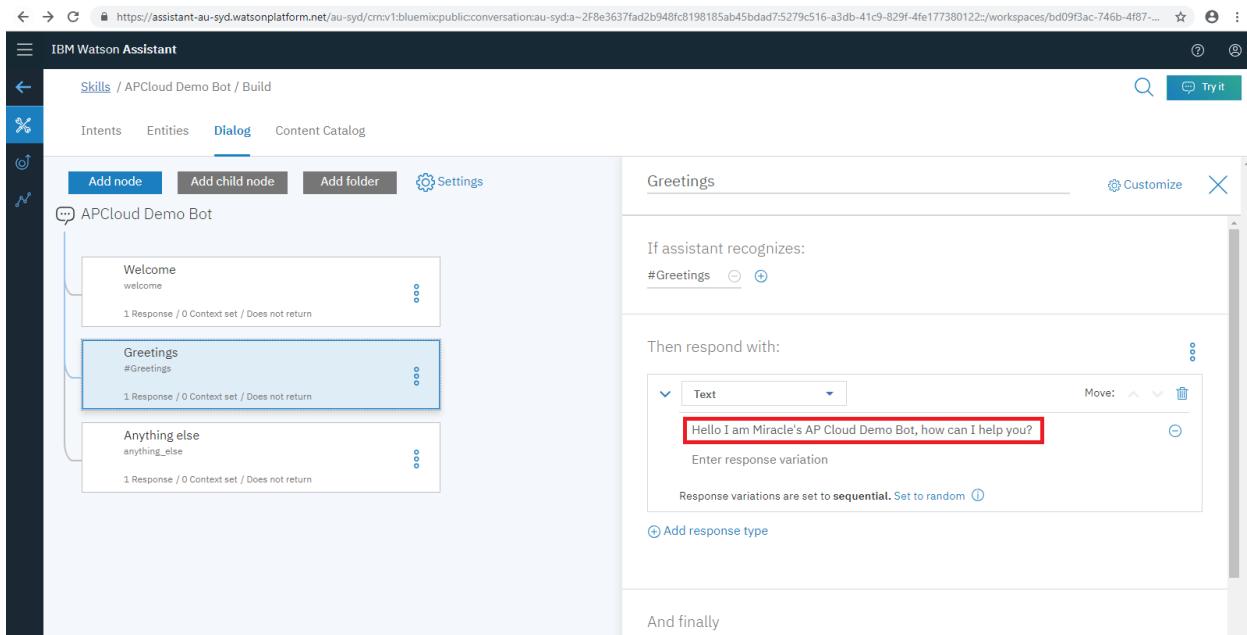
If you want to add more nodes to your dialog tree, click on **Add Node** button on the top.



Then you will have one new node added into your dialog tree as shown below,

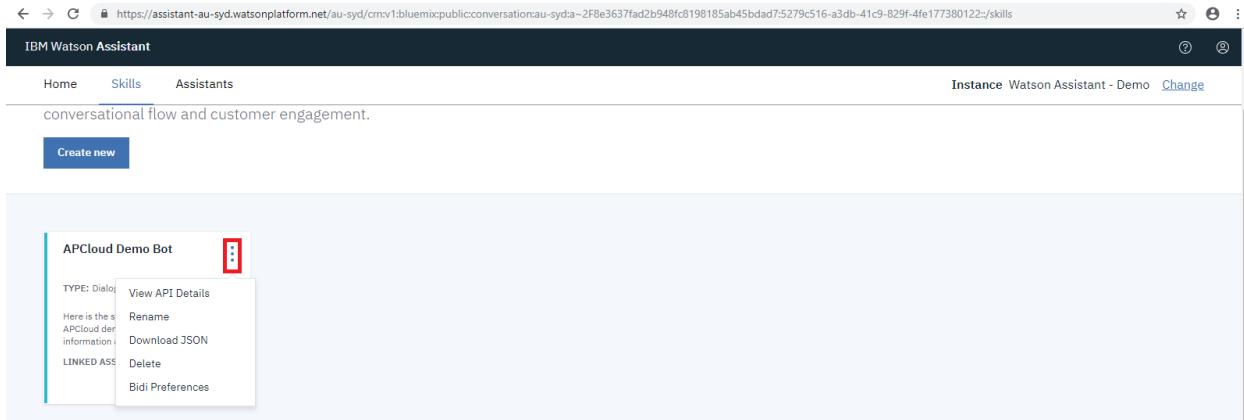


Add the text or multimedia elements that you want the service to display to the user as a response. For that, click on the node and provide the response to the user's request.



If you want to test your skill in the Watson Assistant tool itself, click on **Try it** as shown below and then test your chatbot.

Your first chatbot is successfully created. To access that skill through Node JS, you need to provide the Workspace ID in **app.js** file for that click on three dots (⋮), and the below options are displayed.



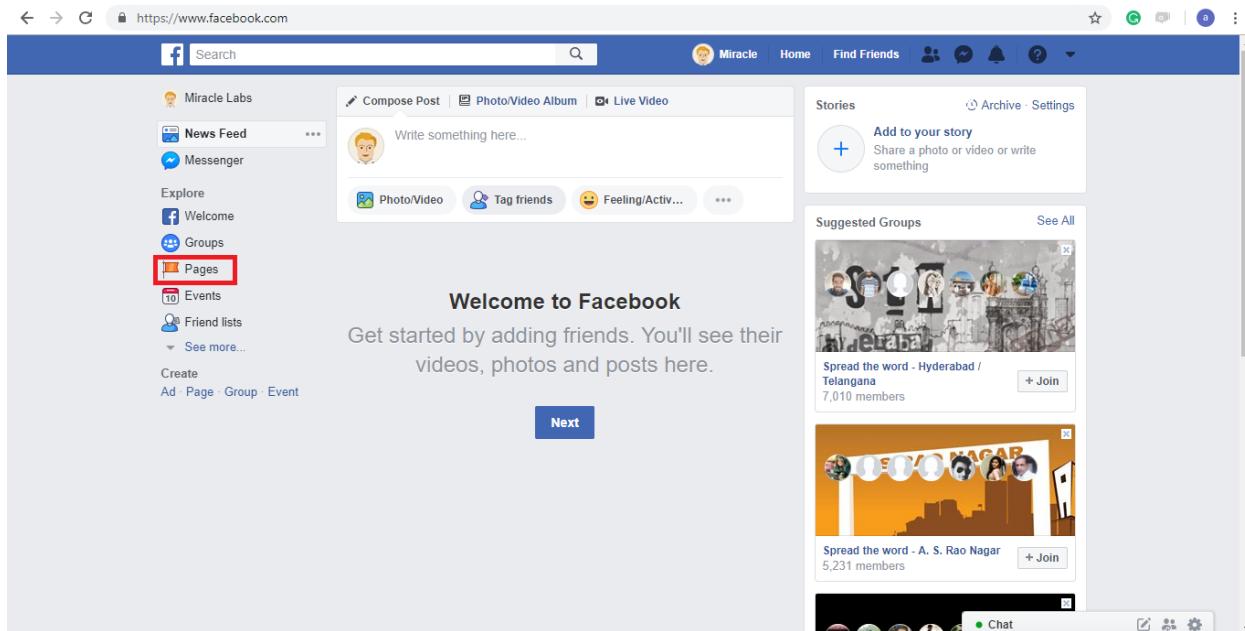
Choose **View API Details** option and click on it. You will get the all API Details.



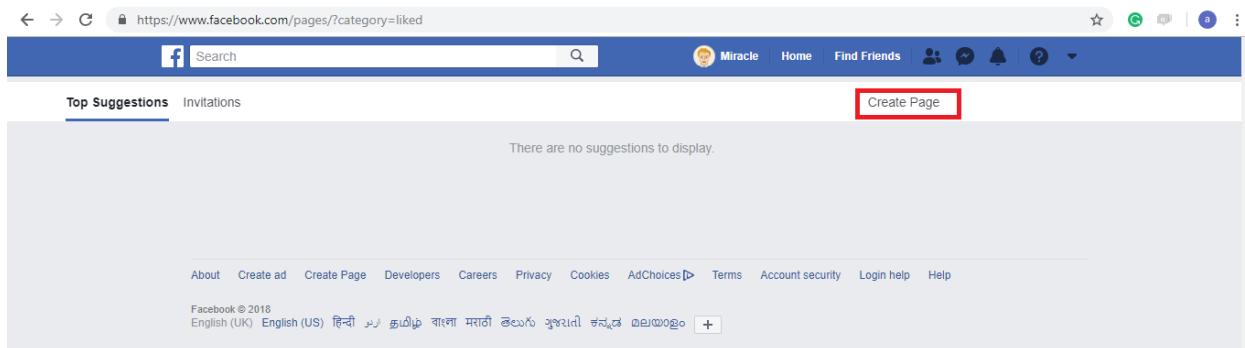
From those API details, you will get the Workspace ID. Copy that Workspace ID and navigate to **app.js** file, paste the Workspace ID in the place of <**Your-Watson-Assistant-WorkspaceID**> at line #42.

Step #4 | Create Facebook Page to Access Chatbot

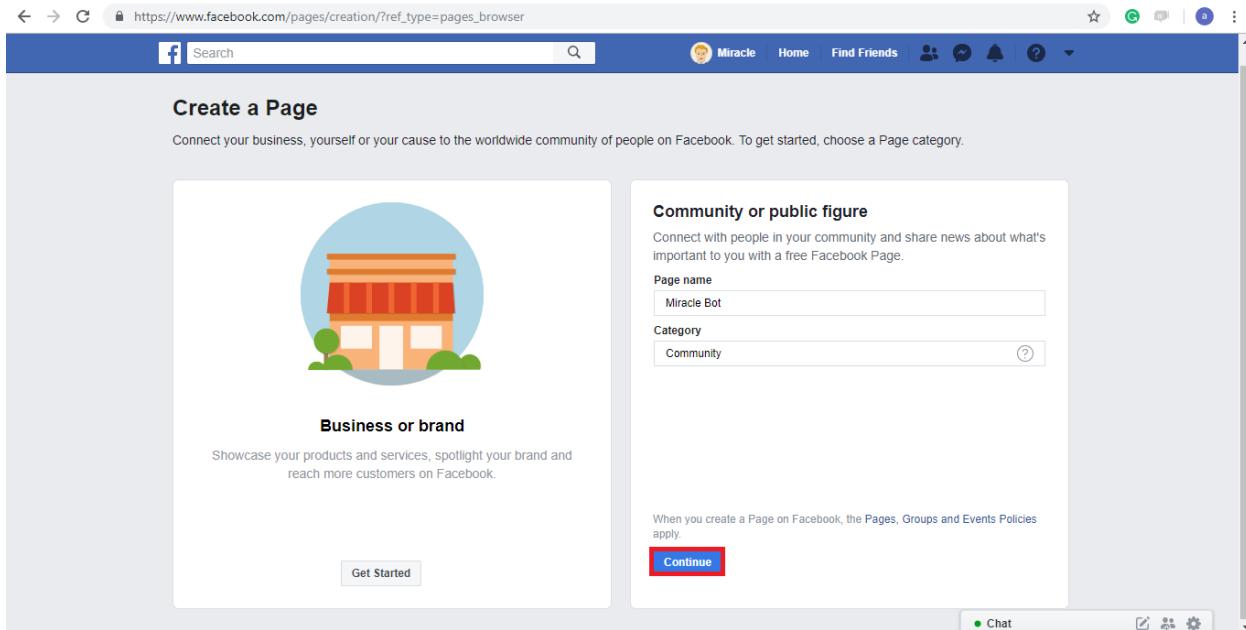
The first step will be to create a Facebook account and Login by providing a valid Email or Phone and Password at, <https://www.facebook.com/>. Once you login into Facebook, you need to create a new page by opening the below link in a new tab, <https://www.facebook.com/pages/create/> or else click on **Pages** option.



If you are creating a page for very first time, you will not find any existing pages. To create a page, click on **Create Page** button as below,



Now, provide a **Page name** and select **Category** of that page and click on **Continue** button. Note that page name should be unique.



The screenshot shows the Facebook 'Create a Page' interface. On the left, there's a section for 'Business or brand' featuring an icon of a storefront. On the right, there's a section for 'Community or public figure' with a 'Page name' field containing 'Miracle Bot' and a 'Category' field set to 'Community'. A 'Continue' button is located at the bottom right of this panel.

After clicking on **Continue** button, your page will be created and looks like below,

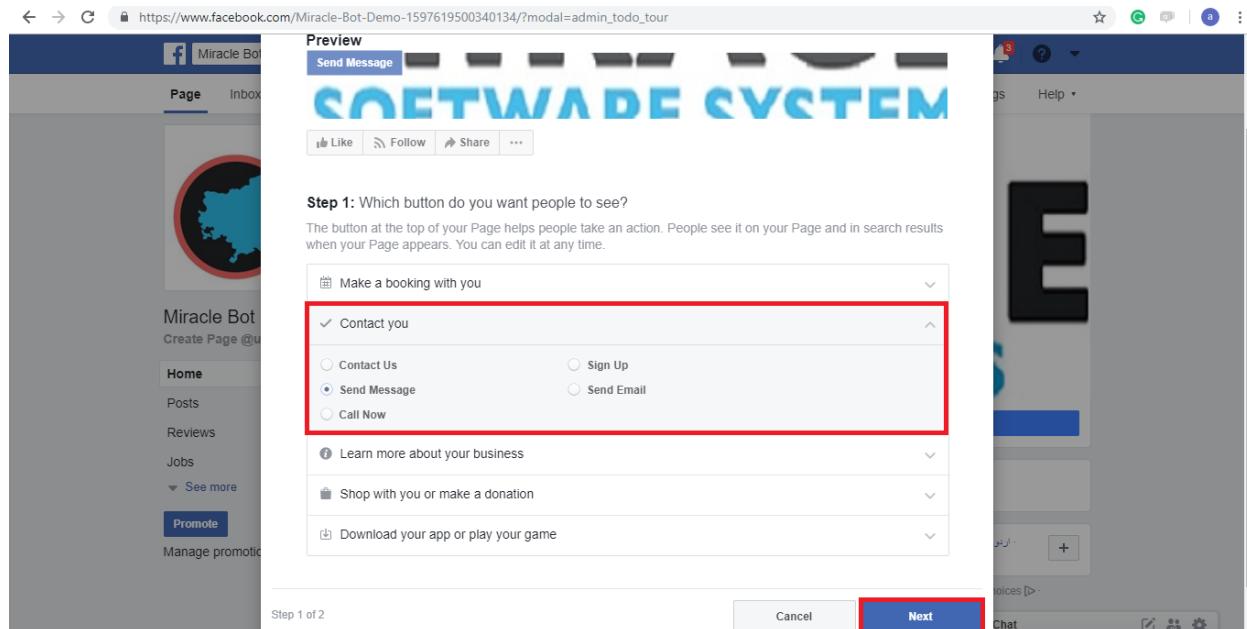


The screenshot shows the newly created Facebook page for 'Miracle Bot Demo'. The page header includes the page name and a profile picture. The main content area features the text 'MIRACLE SOFTWARE SYSTEMS' in large, bold letters. A sidebar on the left provides navigation options like Home, Posts, Reviews, Jobs, and Promote. A message box at the bottom left offers tips for sharing the page. A prominent blue '+Add a Button' button is located in the center of the page.

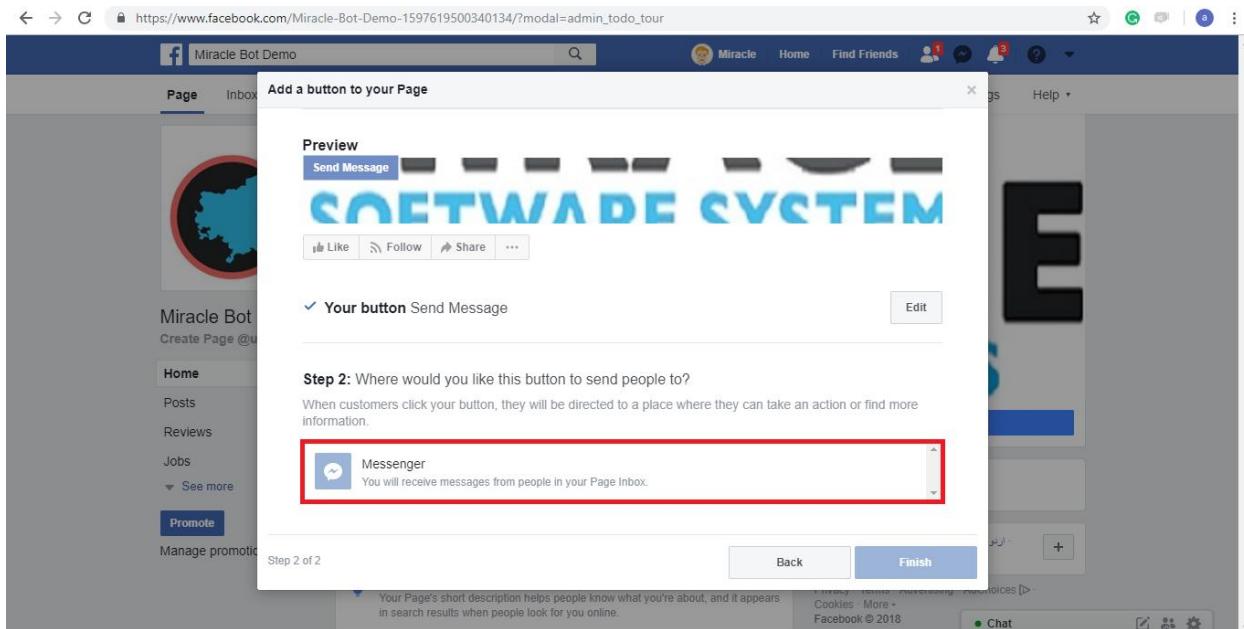
Once you are done with the page creation, click on **+Add a Button** to get send message button and start the conversation.



Now, you will get the options to select the button category. Please select the **Contact Us** option and choose **Send Message** radio button as shown below,

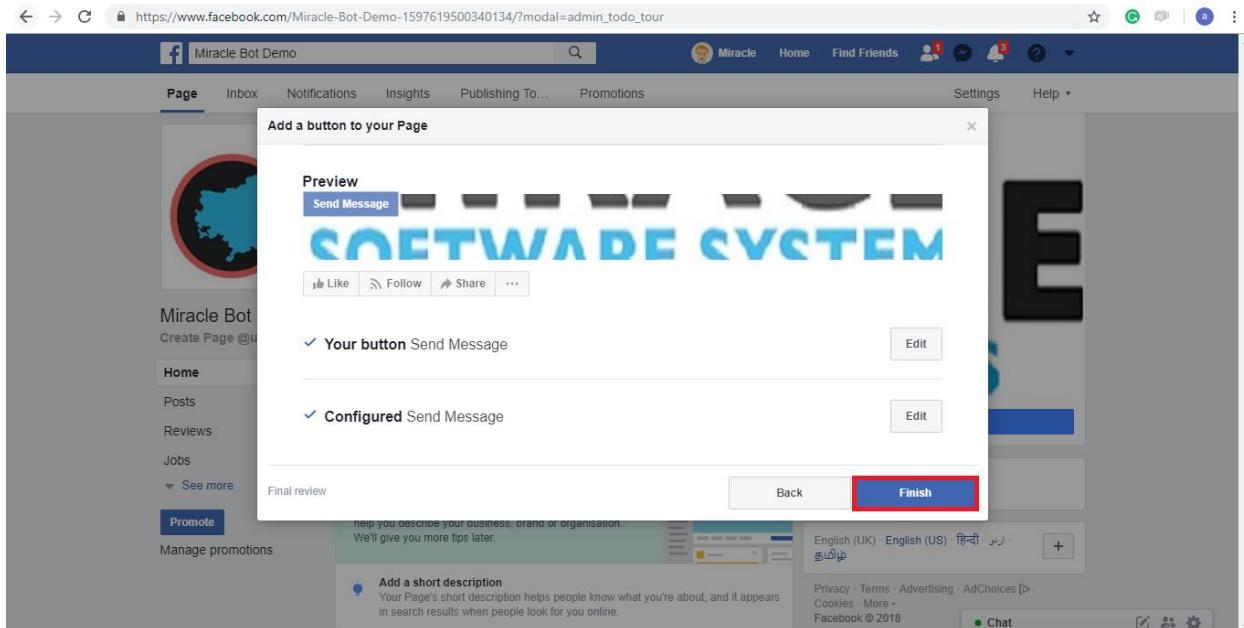


After clicking on next button, it will ask for another option as, **Where would you like this button to send people to?** Place your cursor over **Messenger** and click on it.



Once you complete all the above steps, click on **Finish** button to get the **Send Message** in your page.

Note - If you want to edit any options that you selected previously, click on edit button which is present above the finish button.



Finally you will see the **Send Message** button on your page.



If you want to test the bot that you had created, place the cursor on **Send Message** button, you will get a list of options as shown below,



To send messages to your chatbot, click on **Test button** from the options as shown below,

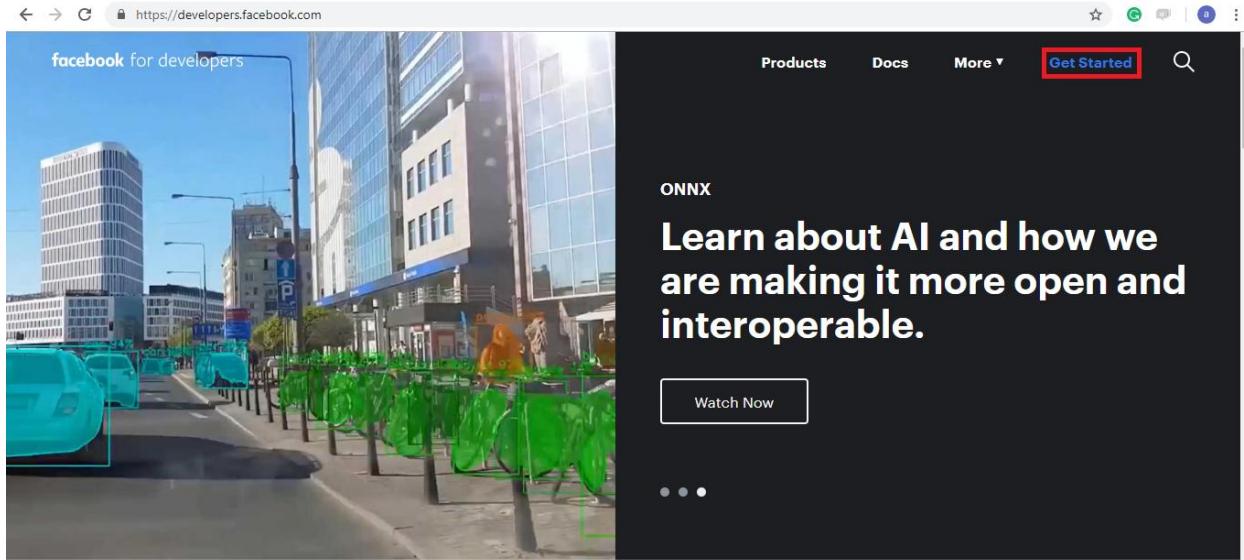


A chat box is popped up to start conversation with your bot.



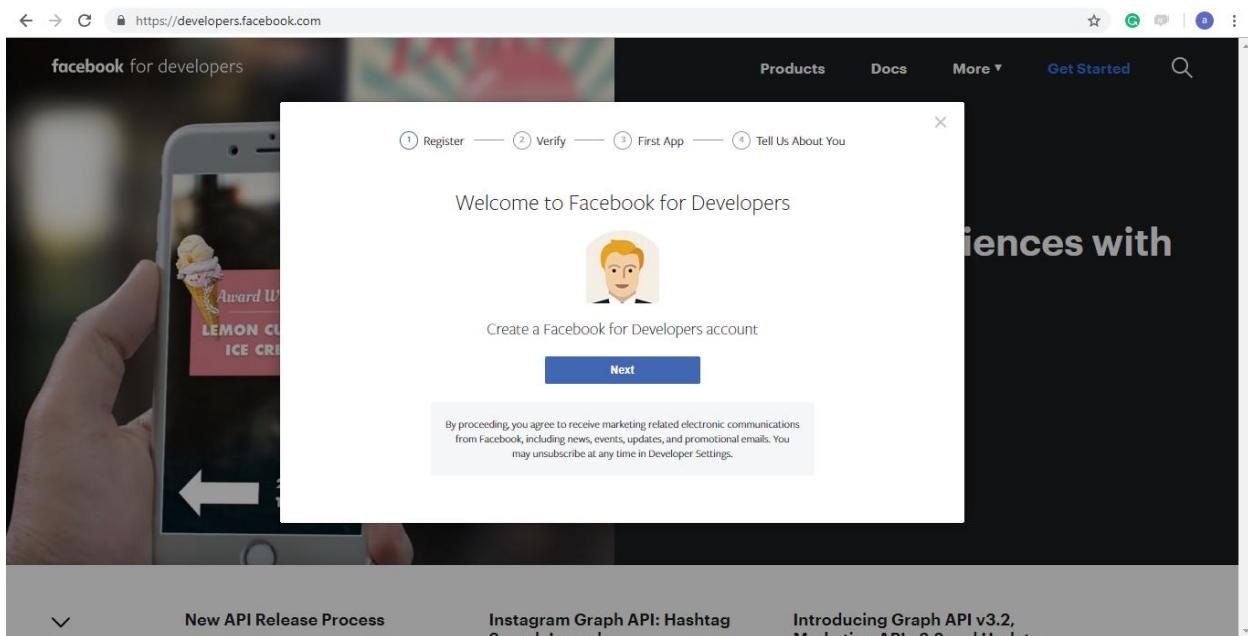
Step #5 | Accessing Facebook for Developers

Login to **Facebook for Developers** page (<http://developers.facebook.com/>), inorder to create apps for generating page access token. Click on **Get Started**.

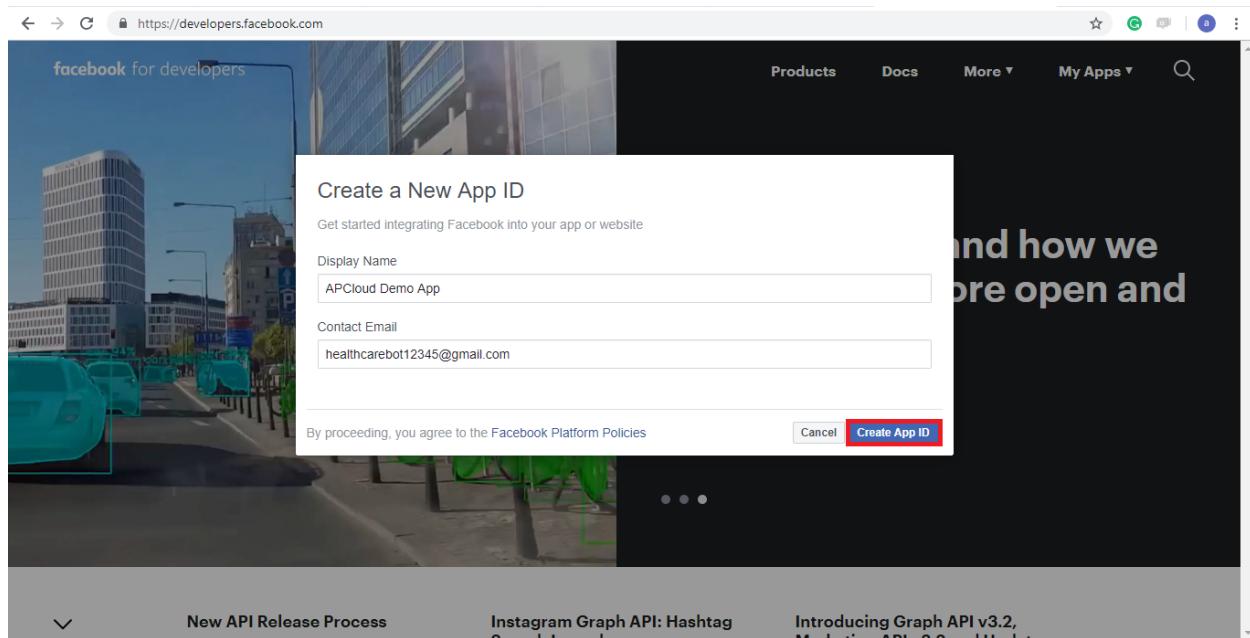


Complete all 4 steps - **Register**, **Verify**, **First App** and **Tell Us About You**. To complete the above steps, authenticate yourself by Email or Phone and provide **<Your App Name>**.

Example - "**AP Cloud Demo Bot**" as the App name.



By default, it gives one app name. Specify a display name for your app and then click on **Create App ID**.



Note - If you have already created the app and you want to use, select it from the **My Apps** menu.

The screenshot shows the Facebook Developers dashboard at <https://developers.facebook.com/apps/2238084399803814/add/>. The top navigation bar includes links for Docs, Tools, Support, and My Apps, along with a search bar. The 'My Apps' section is visible, showing a single app entry: 'APCloud Demo App'. The main content area displays various Facebook products in a grid format. The 'Messenger' product is highlighted with a green circle at the bottom. Each product card includes a 'Read Docs' and a 'Set Up' button.

Select Messenger option and click on **Set Up** to customize the way to interact with people on Messenger through your app

The screenshot shows the Facebook Developers App Dashboard for the 'APCloud Demo App'. On the left, there's a sidebar with 'Dashboard', 'Settings', 'Roles', 'Alerts', and 'App Review'. Under 'PRODUCTS', 'Messenger' is selected, indicated by a red box around its 'Set Up' button. The main area displays six tiles: 'Account Kit', 'Facebook Login', 'Audience Network', 'Analytics' (selected), 'Messenger' (with its 'Set Up' button highlighted in red), and 'Webhooks'. Each tile has a 'Read Docs' button.

Once you click on the **Set Up** option, you will get the Messenger settings. In left side Panel under **Products**, go to Messenger tile and select **Settings**

The screenshot shows the 'Messenger Platform' settings page for the 'APCloud Demo App'. The left sidebar shows 'Dashboard', 'Settings' (selected and highlighted with a red box), 'Roles', 'Alerts', 'App Review', and 'Messenger' with its 'Settings' option also highlighted with a red box. The main content area contains sections for 'Welcome to the Messenger Platform', 'Get started', and 'Token Generation'. The 'Get started' section includes links to 'Quick Start' and 'Complete Documentation'.

Scroll down to **Token Generation**, click on **Select a Page** under **Page** and choose the Facebook page you want to use for your app.

The screenshot shows the Facebook Developers website at <https://developers.facebook.com/apps/2238084399803814/messenger/settings/>. The left sidebar is for the 'APCloud Demo App' under the 'Messenger' category. The main content area is titled 'Token Generation'. It includes a note about page tokens being required for API usage and mentions the 'Quick Start' and 'Complete Documentation'. Below this, there's a 'Page' dropdown set to 'Select a Page' with an option for 'Miracle Bot'. A note says 'You must select a Page to generate an access token.' There's also a 'Create a new page' link. At the bottom, there's a 'Webhooks' section with a 'Setup Webhooks' button.

Now, copy this **Page Access Token** and paste in the **app.js** file at line #13 in the place of <Your-FB-Page-Access-Token>

This screenshot is identical to the one above, but it shows the 'Page Access Token' field filled with a long, redacted string of characters, indicating that the token has been copied from the page settings screen.

Step #6 | Integration with Facebook Messenger using Node JS App

Go to **app.js** file, at line #28 provide any name in the place of <**Verify-Token**> and copy this for later use.

Navigate to the workspace folder where the code exists, and open command prompt Run **node app.js**

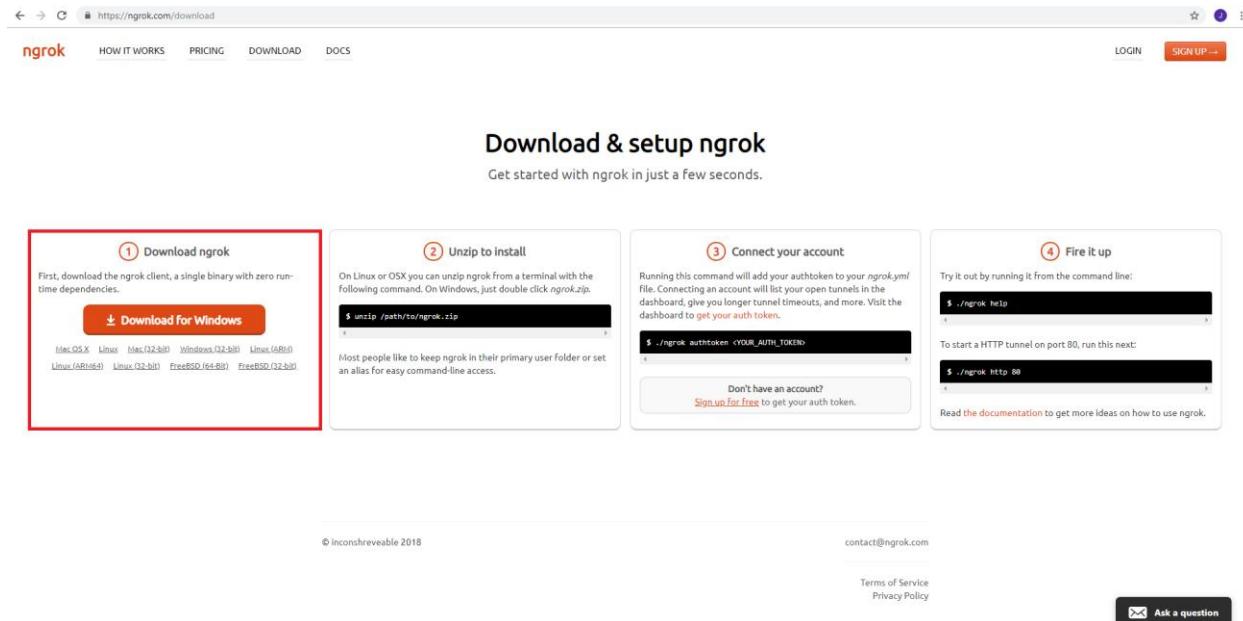


```
C:\Windows\System32\cmd.exe - node app.js
C:\Users\avennela\Desktop\APcloud lab\ node app.js
running on 8000
```

Application is **running on 8000** port. Remember this for later use.

Open the browser and download **ngrok** by using the below link, <https://ngrok.com/download> (ngrok helps your app to get secure URL to your localhost server).

You need to install the ngrok based on the system requirement as provided below,



The screenshot shows the ngrok download page. The main heading is "Download & setup ngrok". Below it, a sub-headline says "Get started with ngrok in just a few seconds." The page is divided into four numbered steps:

- ① Download ngrok**: First, download the ngrok client, a single binary with zero runtime dependencies.
[Download for Windows](#)
Available for Mac OS X, Linux, Mac (32-bit), Windows (32-bit), Linux (32-bit), Linux (64-bit), Linux (ARM64), Linux (32-bit), FreeBSD (64-bit), FreeBSD (32-bit).
- ② Unzip to install**: On Linux or OSX you can unzip ngrok from a terminal with the following command. On Windows, just double click `ngrok.zip`.
`$ unzip /path/to/ngrok.zip`
Most people like to keep ngrok in their primary user Folder or set an alias for easy command-line access.
- ③ Connect your account**: Running this command will add your auth token to your `ngrok.yml` file. Connecting an account will list your open tunnels in the dashboard, give you longer tunnel timeouts, and more. Visit the dashboard to [get](#) your auth token.
`$./ngrok auth tokens <YOUR_AUTH_TOKEN>`
Don't have an account? [Sign up for free](#) to get your auth token.
- ④ Fire it up**: Try it out by running it from the command line:
`$./ngrok http 80`
To start a HTTP tunnel on port 80, run this next:
`$./ngrok http 80`
Read the documentation to get more ideas on how to use ngrok.

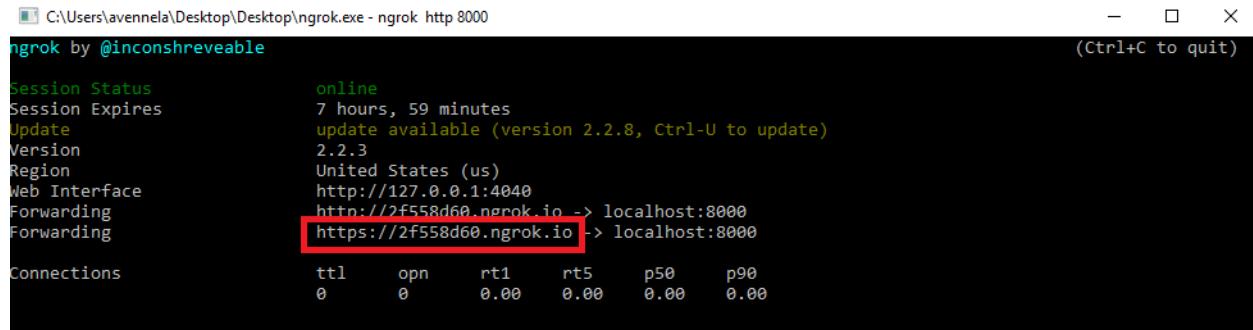
After downloading the ngrok, extract the downloaded folder and run the **ngrok.exe** file. Now, enter the following command and click Enter button.

ngrok http <Your-Application-Port>

Example: ngrok http 8000

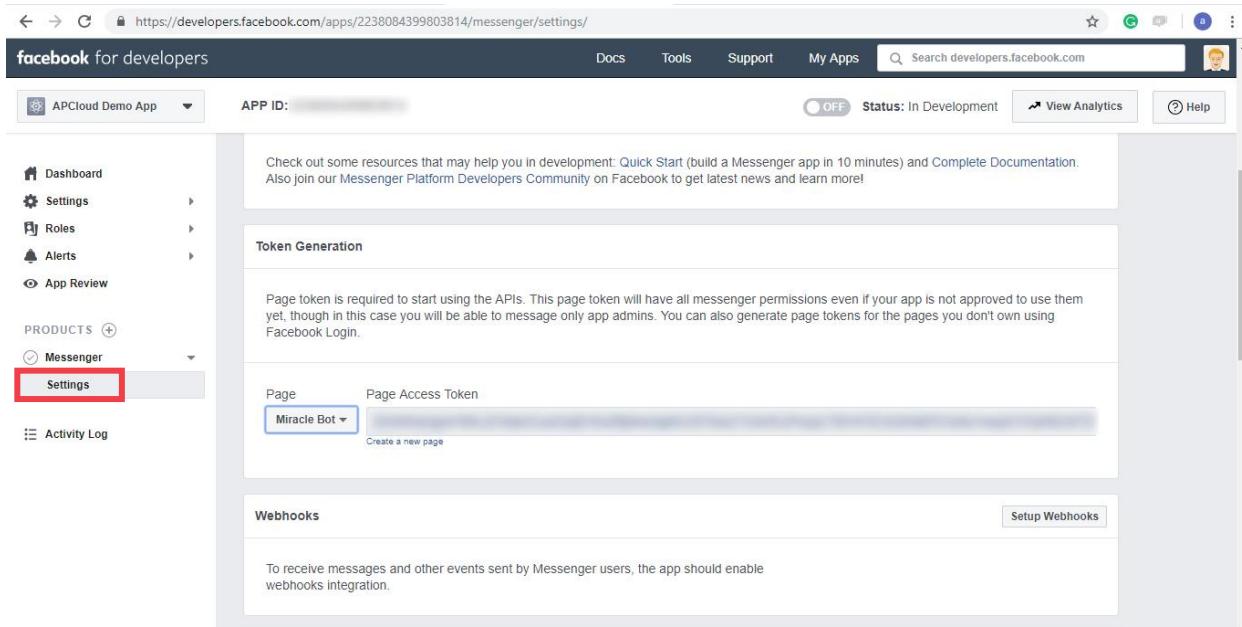
Once you run the application successfully with the above command, you will get the URL as **https://<random code>.ngrok.io>**. Copy this for later use.

Note - You should not close this window until you stop running your Node JS application



```
C:\Users\avennela\Desktop\Desktop\ngrok.exe - ngrok http 8000
ngrok by @inconshreveable
Session Status          online
Session Expires        7 hours, 59 minutes
Update                 update available (version 2.2.8, Ctrl-U to update)
Version                2.2.3
Region                 United States (us)
Web Interface          http://127.0.0.1:4040
Forwarding             http://2f558d60.ngrok.io -> localhost:8000
Forwarding             https://2f558d60.ngrok.io -> localhost:8000
Connections            ttl     opn      rt1     rt5     p50     p90
                         0       0       0.00   0.00   0.00   0.00
(Ctrl+C to quit)
```

Now, navigate to your Facebook Developer Page, in your app and go to the Messenger settings and scroll to the **Webhooks** section. Click on **Setup Webhooks**.



facebook for developers

APP ID: [REDACTED]

Status: In Development

Token Generation

Page token is required to start using the APIs. This page token will have all messenger permissions even if your app is not approved to use them yet, though in this case you will be able to message only app admins. You can also generate page tokens for the pages you don't own using Facebook Login.

Page Access Token

Miracle Bot

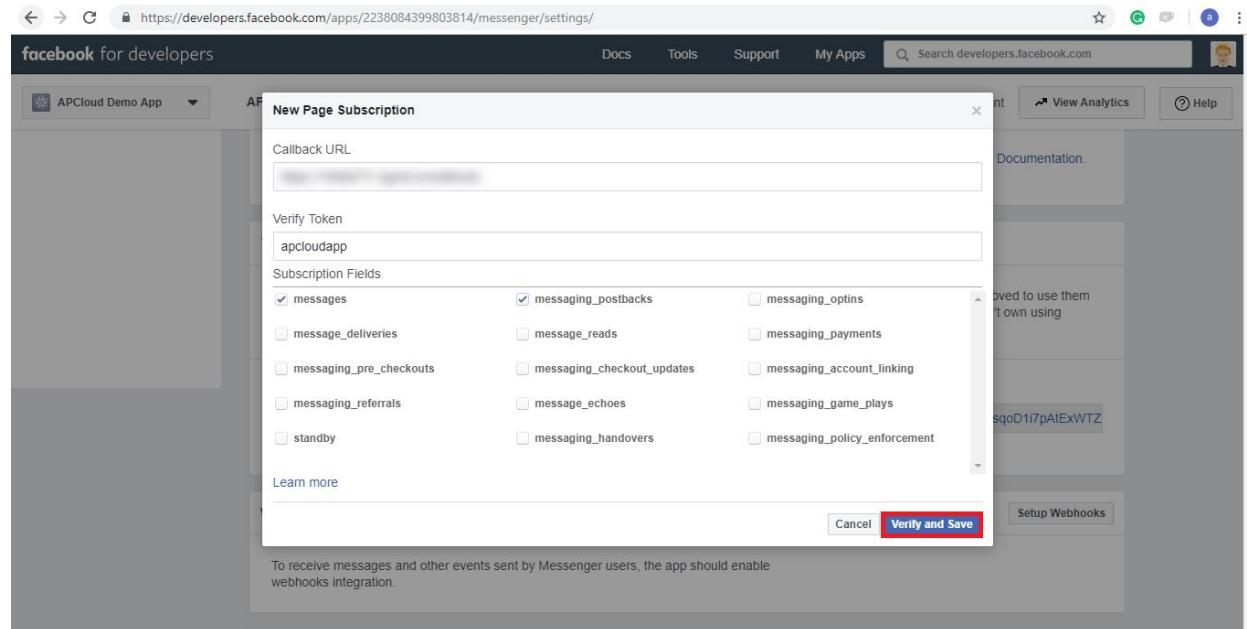
Webhooks

To receive messages and other events sent by Messenger users, the app should enable webhooks integration.

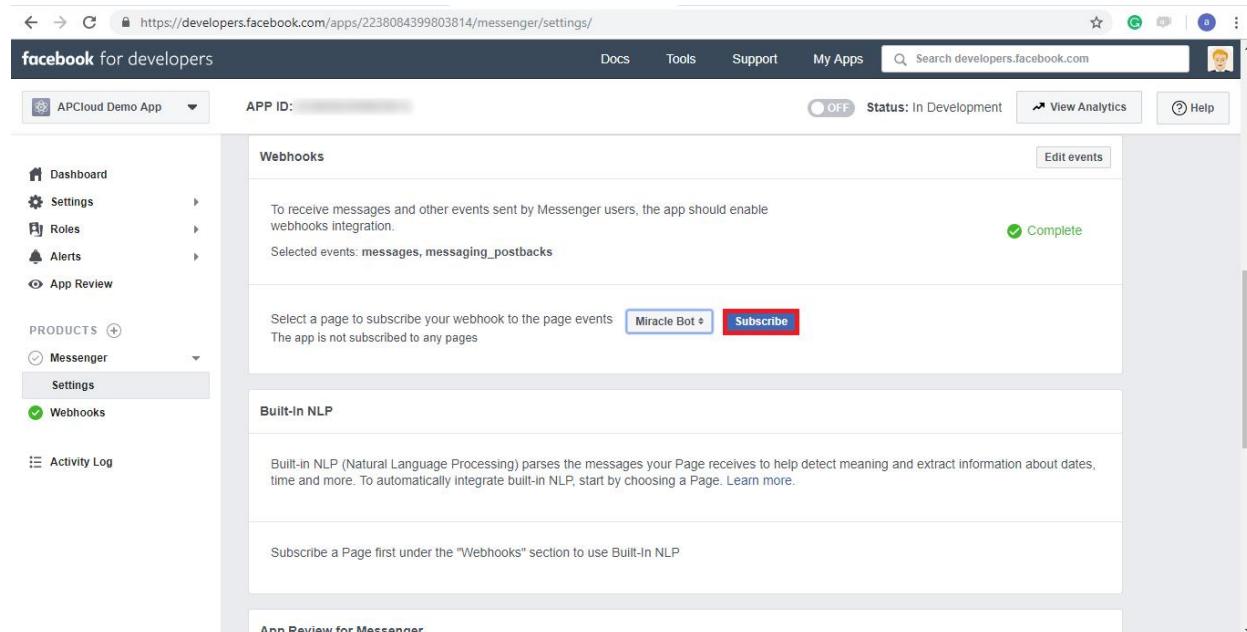
Setup Webhooks

You will be navigated to **New Page Subscription** window, paste the ngrok URL into the **Callback URL** field. In the **Verify Token** field, specify the same Facebook verify

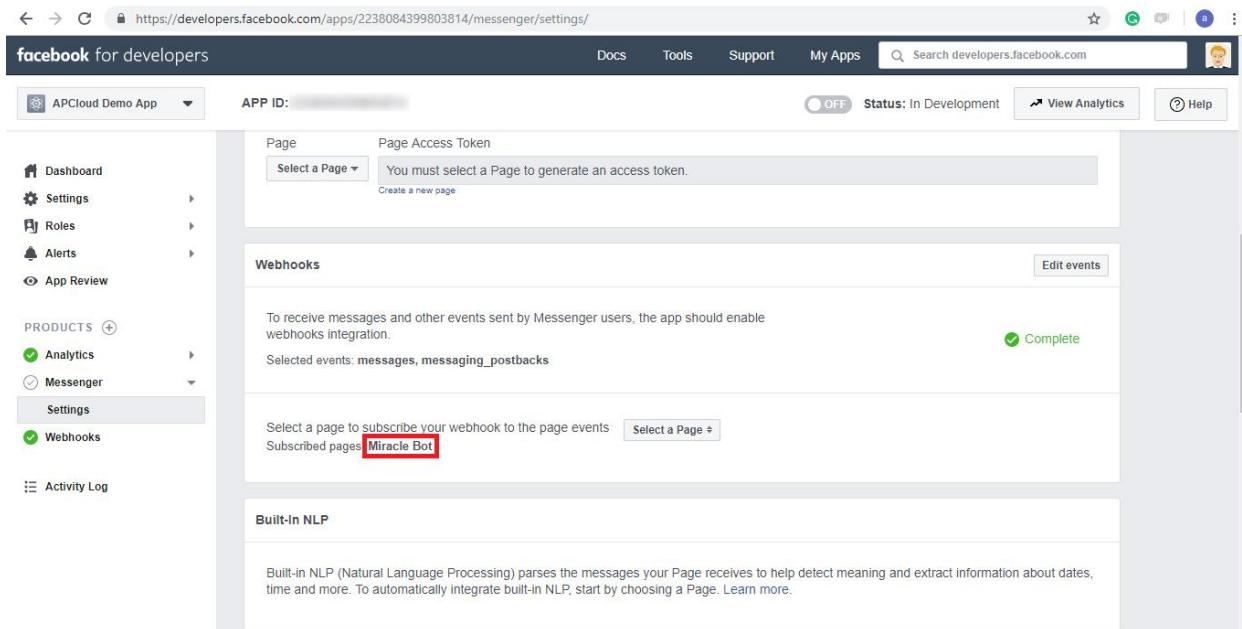
token that you created earlier in `app.js`. Under **Subscription Fields**, select **messages** and **messaging_postbacks**. Then click on **Verify and Save**.



Once the verification is completed, go back to **Webhooks** section in the Messenger settings and click **Select a Page**. Select the same page you selected during token generation, and then click on **Subscribe**.



Now, you will get the **Subscribed pages**: <Name of the page you selected> in the **Webhooks** section as shown below,



The screenshot shows the Facebook Developers website at <https://developers.facebook.com/apps/2238084399803814/messenger/settings/>. The left sidebar is for the 'APCloud Demo App' under the 'Messenger' category. The main area shows the 'Webhooks' configuration. It has a 'Page Access Token' section where 'Select a Page' is dropdowned to 'Miracle Bot'. Below it is a 'Webhooks' section with a note about enabling integration for messages and messaging_postbacks, marked as 'Complete'. A 'Subscribed pages' section lists 'Miracle Bot'.

Step #7 | Testing Miracle Bot in Facebook Messenger

Open your Facebook Messenger, search for your Facebook bot (or the Facebook page you had subscribed to) and talk to it!



The screenshot shows the Facebook Messenger interface at https://www.facebook.com/Miracle-Bot-273867316599278/?modal=admin_todo_tour. The top navigation bar shows 'Miracle Bot'. The main area shows the 'Miracle Bot' page with its profile picture and name. A message from the bot says: 'Hello I am AP Cloud Demo Bot, how can I help you?'. The message input field is visible at the bottom.



Hurrah!! With this lab you were able to create your first chatbot with **Watson Assistant** using **Facebook Messenger**

For any questions regarding the lab please feel free to reach out to innovation@miraclesoft.com. We hope you enjoyed creating bots with us 😊