Step 1. First of all, Create account on GCP (Google Cloud Platform).

Step 2. After creating account we have to creat a project. After creating project, You will see a interface like this.

## A screenshot of a computer AI-generated content may be incorrect.

## ✅ Types of Chatbots We Can Build on GCP

Google Cloud Platform (GCP) enables the development of various types of chatbots, each tailored for specific use cases and platforms. Below are the primary chatbot types you can create:

### 1. **WhatsApp / Telegram / Web Chatbot**

* **Tools Used**: Dialogflow + Twilio / Telegram API / Webhook
* **Functionality**: A chatbot integrated with messaging platforms or websites
* **Use Cases**:
  + WhatsApp-based customer service
  + Telegram bots for reminders or alerts
  + Website-integrated support bots

### 2. **Multilingual Chatbot**

* **Tools Used**: Dialogflow CX + Cloud Translation API
* **Functionality**: A chatbot capable of understanding and responding in multiple languages
* **Use Cases**:
  + Global customer support
  + Language-agnostic service desks
  + Tourism or international business assistants

### 3. **AI-Powered Chatbot (Generative AI + RAG)**

* **Tools Used**: Vertex AI (PaLM/Gemma) + LangChain + Dialogflow
* **Functionality**: An intelligent, ChatGPT-style chatbot that supports reasoning and retrieval-augmented generation (RAG)
* **Use Cases**:
  + Legal or financial assistants
  + Document-based Q&A bots
  + Code explanation and technical support chatbots

### 4. **Data-Driven Chatbot**

* **Tools Used**: Dialogflow + BigQuery + Cloud Functions
* **Functionality**: A chatbot that fetches live data from a database or BigQuery based on user input
* **Use Cases**:
  + Sales and business intelligence dashboards
  + Analytics and reporting bots
  + Inventory and performance monitoring assistants
* **So now We are focusing on Data-driven chatbot –**

**For creating this chatbot we need to understand the "Architecture Skeleton for Building a Data-Driven Chatbot using Dialogflow CX"**

**A yellow background with text and icons

AI-generated content may be incorrect.**

**Step1. For creating Data-driven chatbot first of all we will enable Dialogue Flow CX API-**

To enable the Dialogflow CX API in Google Cloud, follow these steps:

1. **Open the Google Cloud Console**:  
   Go to the [Google Cloud Console](https://console.cloud.google.com/).
2. **Select or Create a Project**:  
   In the top navigation bar, click on the project dropdown and either choose an existing project or create a new one.
3. **Navigate to the API Library**:  
   On the left-hand menu, go to **APIs & Services** and then select **Library**.
4. **Search for Dialogflow CX API**:  
   In the search bar, type **Dialogflow CX API**.
5. **Enable the API**:  
   Select **Dialogflow CX API** from the search results, then click **Enable**.

**Step 2. Open Dialog flow CX**

**This is the link for opening dialogue flow console –**

**https://dialogflow.cloud.google.com/cx/** **projects**

**A screenshot of a computer

AI-generated content may be incorrect.**

Now In the dialogue flow CX There is option for creating Agent.

What is agent - An agent in Dialog flow CX is the main part of the chatbot that manages conversations. It contains all the logic like intents, entities, flows, and responses.

Step 3 – Create a Agent -

When you click on the Create agent you will see 2 options-

A screenshot of a chat

AI-generated content may be incorrect.

**So choose here Build Your Own**after clicking on build your own you will see a page like this

A screenshot of a computer

AI-generated content may be incorrect.after filling these details you will see your agent name under display name.A screenshot of a computer

AI-generated content may be incorrect.

After that we will click on agent which we can show here as EmployeeDataBot

Then we will see a interface like this

A screenshot of a computer

AI-generated content may be incorrect.

Step 4. Create a intent

1. **Go to Intents**:  
   On the left menu, click on **"Manage"**, then click on **"Intents"**.
2. **Create a New Intent**:  
   Click on the **"Create Intent"** button.

A screenshot of a computer

AI-generated content may be incorrect.

1. **Name the Intent** where is written Display name  
   Give your intent a name, for example: Getaverage salary
2. **Add Training Phrases**:  
   In the **Training Phrases** section, type examples of how users might say something to trigger this intent.  
   Example:

* What’s the average salary?”
* “What is the average salary of employees?”
* “Tell me about the average salary in Sales.”
* “What’s the average pay?”
* “How much is the average salary for HR?”

1. **Save the Intent**:  
   After adding phrases and responses, click **"Save"**.

After creating intent we will click on Entity type

Step 5 – Create a entity

When you will click on entity type there will be 2 methods for creating entity

1. First is custom and (II) second is system

A screenshot of a computer

AI-generated content may be incorrect.

So there is details for custom and system

**✅ Custom vs System Entity in Dialogflow CX**

| **Option** | **Use it when...** | **Example** |
| --- | --- | --- |
| **Custom Entity** | You want to define your own values | Departments like HR, Finance, IT |
| **System Entity** | Dialogflow already has a built-in entity | Dates, numbers, email, phone, currency, etc. (@sys.date, @sys.number) |

So we will choose here Custom Entity

After that we will click on + Create then we will see a interface like this

A screenshot of a computer

AI-generated content may be incorrect.

**Step 5 - Create a New Entity**:

* Click on the **+ Create** button (or a plus icon) to add a new entity.
* **Entity Name**: Enter Department.
* **Entity Type**: Ensure it’s set to Custom Entity (not a system entity).

**Add Entity Values**:

* In the **Entity Entries** section, add possible department names that users might mention.
* Add these values (one per line):
  + Sales
  + Engineering
  + HR
  + Marketing
  + Finance
* For each value, you can optionally add synonyms if needed (e.g., for HR, a synonym could be Human Resources), but for now, just the values are fine.
* Example:
  + Sales (no synonyms)
  + Engineering (no synonyms)
  + HR (synonym: Human Resources)
* Save the entity by clicking the **Save** button in the Entities section.

Step 6 - **Enable Entity Extraction for the Intent**:

* Go back to the **Intents** section.
* Open the GetAverageSalary intent.
* In the **Parameters** section, add a parameter:
  + **Parameter Name**: department
  + **Entity Type**: Select @Department (the entity you just created).
  + **Is Required**: Leave unchecked (since the department is optional—users might ask for the overall average salary).

A screenshot of a computer

AI-generated content may be incorrect.

* In the **Training Phrases**, annotate the department if needed:
  + For the phrase “What’s the average salary in Sales?”, highlight “Sales” and tag it as @Department.
  + For the phrase “What’s the average salary for HR?”, highlight “HR” and tag it as @Department.
  + For phrases like “What’s the average salary?” (no department), leave them as is.

**Save Everything**:

* Save the intent by clicking the **Save** button in the Intents section.

Step 7 - Create a Flow to Handle the GetAverageSalary Intent

**Create a New Flow**:

* We will go in the build option near manage
* Click on the **+ Create** button (or a plus icon) to add a new flow.
* **Flow Name**: Enter GetAverageSalaryFlow.
* Click **Create** or **OK** to create the flow.
* The new flow will open in the visual flow editor, with a default **Start** page.

A screenshot of a computer

AI-generated content may be incorrect.

**Route the GetAverageSalary Intent to a New Page**:

* In the GetAverageSalaryFlow, click on the **Start** page to open its settings (right side of the canvas).
* In the **Routes** section:
  + Click **+ Add Route**.
  + **Intent**: Select GetAverageSalary from the dropdown (the intent you created )

A screenshot of a computer

AI-generated content may be incorrect.

* + **Transition**: Create a new page for this intent:
    - Click **Create New Page**.
    - **Page Name**: Enter GetSalaryPage.
    - Click **Create** or **OK**.
  + This sets up the flow so that when the GetAverageSalary intent is detected, it transitions to the GetSalaryPage.

A screenshot of a computer

AI-generated content may be incorrect.

**Configure the GetSalaryPage**:

* Click on the GetSalaryPage in the flow editor to open its settings (right side).
* **Parameters Section**:
  + The department parameter (from the GetAverageSalary intent) should already be listed since it’s tied to the intent.
  + If not, add it manually:
    - Click **+ Add Parameter**.
    - **Parameter Name**: department
    - **Entity Type**: @Department (the entity you created in Step 3).
    - **Is Required**: Uncheck this box (to make the parameter optional, as users might ask for the overall average salary without specifying a department).

A screenshot of a computer

AI-generated content may be incorrect.

* **Fulfillment Section**:
  + Enable the webhook to handle the salary calculation:
    - Toggle **Enable Webhook** to ON (or check a box like “Use Webhook”).
    - **Webhook**: We need to create a webhook resource first (next sub-step).
    - **Webhook Tag**: Enter get-average-salary (this tag helps identify the webhook request in your backend logic).

**A yellow background with text and icons

AI-generated content may be incorrect.**