CHATBOT PROJECT USING AZURE LUIS

Contents

[Introduction 2](#_Toc38587611)

[Background 2](#_Toc38587612)

[Key Objectives 2](#_Toc38587613)

[Chatbot Requirements 2](#_Toc38587614)

[Application Architecture 3](#_Toc38587615)

[Overview 3](#_Toc38587616)

[Cloud Platform (Azure Portal) 3](#_Toc38587617)

[Azure Services 4](#_Toc38587618)

[Functional Requirements and Validation 5](#_Toc38587619)

[Web Chat 5](#_Toc38587620)

[Microsoft Teams Chat 6](#_Toc38587621)

[Telegram Chat 7](#_Toc38587622)

[Email Notification 7](#_Toc38587623)

[Data Storage 8](#_Toc38587624)

[Bot Card 8](#_Toc38587625)

[Technical Requirements 9](#_Toc38587626)

[Azure Subscription 9](#_Toc38587627)

[Resource Group. 9](#_Toc38587628)

[Web App Service 9](#_Toc38587629)

[Bot Channel Registration 10](#_Toc38587630)

[LUIS App Registration 10](#_Toc38587631)

[Azure CosmosDB Creation 10](#_Toc38587632)

[Source Code 10](#_Toc38587633)

**CHATBOT PROJECT USING AZUIRE LUIS**

# Introduction

The purpose of this document is to detail out the functionality of the Chatbot Project using Azure LUIS the assumptions to ensure that the Project team and the various Stakeholders have a common understanding to ensure the success of the project.

# Background

**Ineuron AI** team conducted Chatbot project challenge for its students to motivate the potentials to deliver a project end to end.

# Key Objectives

Build a covid-19 Chatbot, which will be able to tell us a number of cases available in any of the locality by providing Pin-code, city name, country etc., as input to the Chatbot and it will give all the instruction and preventive measures.

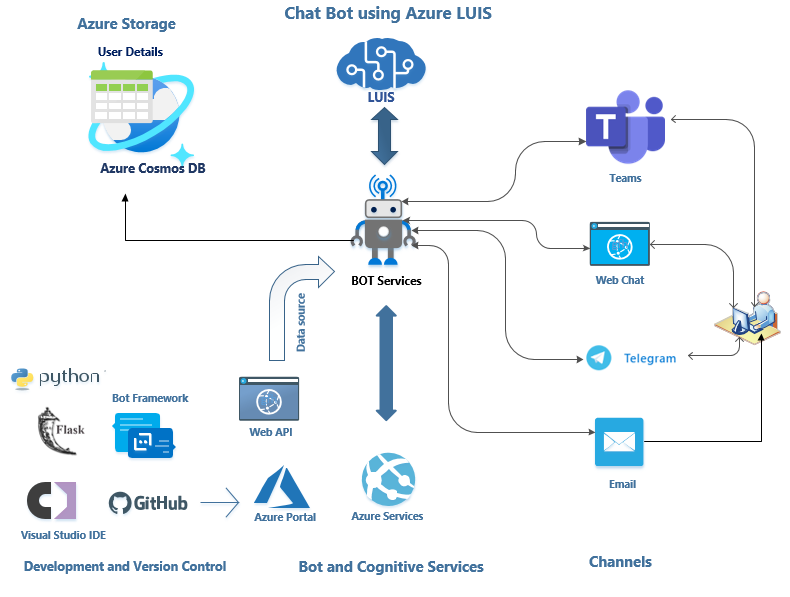
## Chatbot Requirements

* Chatbot should welcome the user
* Request the user’s name, email ID, mobile number and city name. Store the details in any cloud database
* Validations to be done while collecting the data
* Chatbot should give show details of Total active, Recovered and Cases for the city or country provided by the user.
* Send email to the user about the preventive measures to be taken for Covidh-19 Corona Virus
* Bot should provide the option to show map wise COVID cases in case a user is asking for a worldwide case of covid-19

# Application Architecture

## Overview

Application Architecture is to detail about how the application works, technologies used and how all the components communicate with each other with a pictorial representation.



### Cloud Platform (Azure Portal)

Chatbot project is built and deployed in Microsoft azure cloud platform with an Azure subscription (free options).

### Azure Services

Microsoft cloud platform provides multiple services for different kind of projects based on the usage. For this Chatbot project we have used below services

#### Language Understanding (LUIS)

A machine learning-based service to build natural language into apps, bots, and IoT devices. It is designed to identify valuable information in conversations, LUIS interprets user goals (intents) and distills valuable information from sentences (entities), for a high quality, nuanced language model. LUIS integrates seamlessly with the Azure Bot Service, making it easy to create a sophisticated bot.

#### Azure Web App

This service is used create and deploy web applications in different platforms like Windows and Linus. Continuous deployment can be done using GitHub, DevOps etc. For this project we have used **GitHub** to deploy the website in Azure Web App using **KUDU** which is used as Service Control manager for Azure web sites.

**Development Platform and Tools**

Website is developed using Flask web framework and Python as programming language. Microsoft Bot framework used as the primary component to integrate with LUIS and retrieve and record user conversations. Visual Studio Community edition is used as the IDE for development.

#### Azure Cosmos DB

Data provided by user through Chatbot is stored in Azure Cosmos DB. Azure Cosmos DB is Microsoft's globally distributed, multi-model database service which provides the options to store data using SQL, MongoDB, Cassandra, Tables. For this project we have used DB Table storage. It provides the simplest and fastest way to retrieve and store data.

**Data Source for Web App**

Live website which provided the data related Corona virus across the world is used as the data source Chatbot service integrating with LUIS.

**Website Link 🡪** <https://www.trackcorona.live>

#### Bot Channels Registration

Azure Bot Channels registration services is used to integrate the Web App service with different communication channels to provide requested by user. A Channel is the communication between the bot and communication Apps. For this project we have used Azure Web App service as the communication platform. Channels used for this project are

1. Web Chat
2. Microsoft Teams
3. Telegram

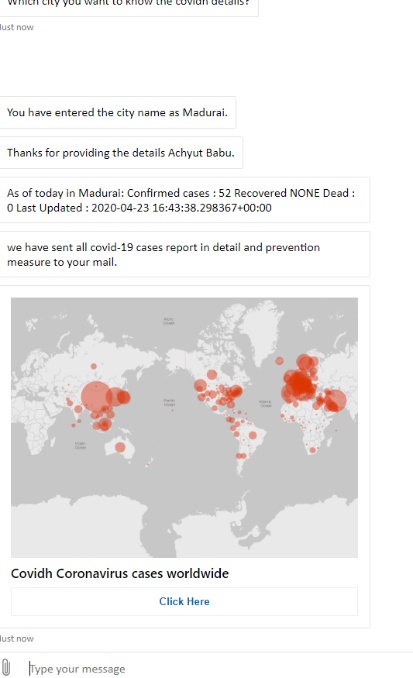
# Functional Requirements and Validation

## Web Chat

Separate link for the user to interact with the bot any time to get the information about coivdh corona virus cases for the city or country. Also a prebuilt iframe for any to paste in the website and us it as Chatbot. **Below is the link highlighted and iframe to do Web Chat**

**IFrame for this project**

**<iframe src='https://webchat.botframework.com/embed/covidhbotchannel?s=vPdkVvBJxS8.yCHhKpUpIpiDGBYHIJcJylS3miBTBcoYwFqzPL2fTqk' style='min-width: 400px; width: 100%; min-height: 500px;'></iframe>**



**Details of covidh corona virus cases (sample)**

As of today in Chennai: Confirmed cases: 400 Recovered NONE Dead: 7 Last Updated: 2020-04-23 17:03:36.006021+00:00

## Microsoft Teams Chat

Separate link and iframe for the team to chat with the bot get details about corona virus cases.

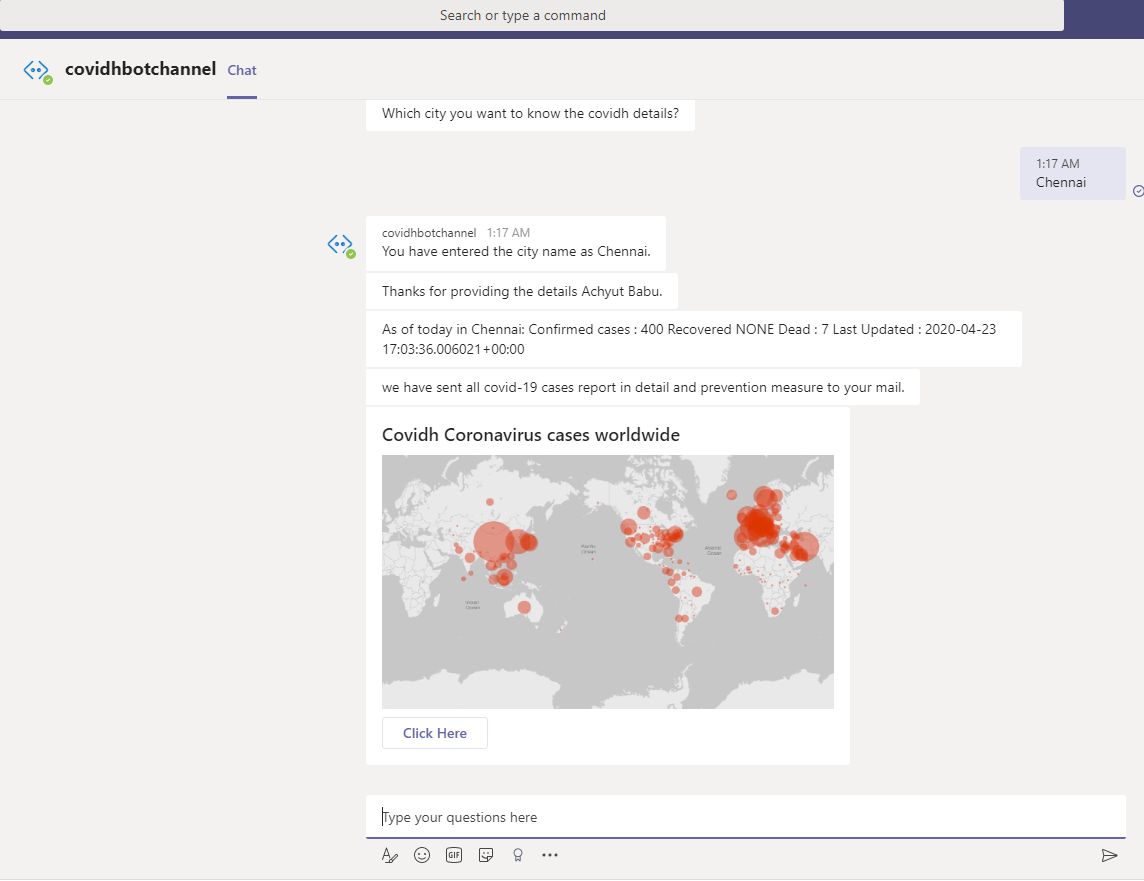
Note: Teams can also be used from mobile and also it has got client version.

**Below is the link highlighted and iframe to do Web Chat**

**Iframe**

**<a href='https://teams.microsoft.com/l/chat/0/0?users=28:90258c37-4e41-4894-a48f-0bd342d4bc1c'><img src='https://dev.botframework.com/Client/Images/Add-To-MSTeams-Buttons.png'/></a>**

**Teams Chat**



Note: Both Web Chat and Microsoft Teams can be directly configured from the Bot Channel registration section as show in the picture below. **For telegram separate token needs to be registered using the Botfather app available in Telegram**

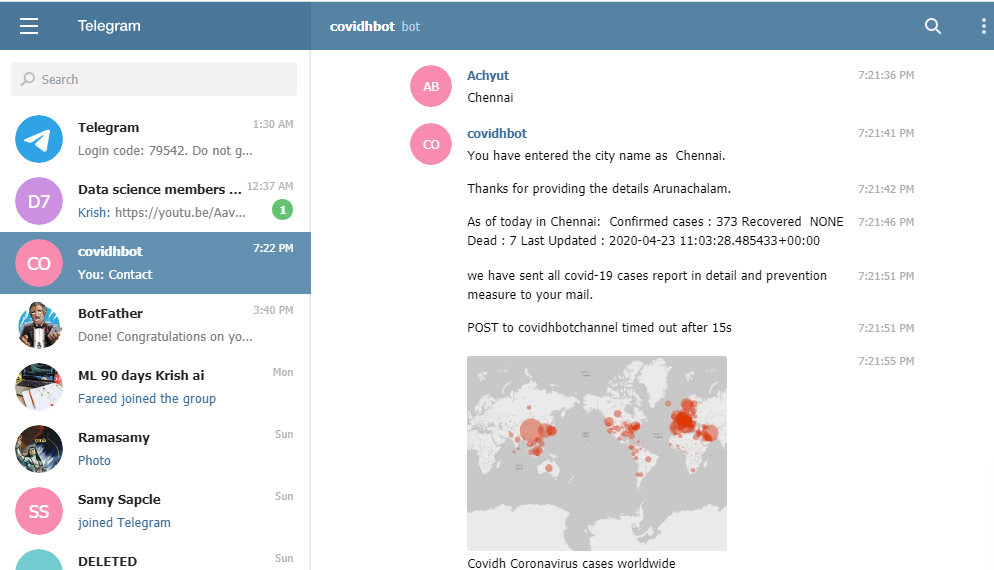
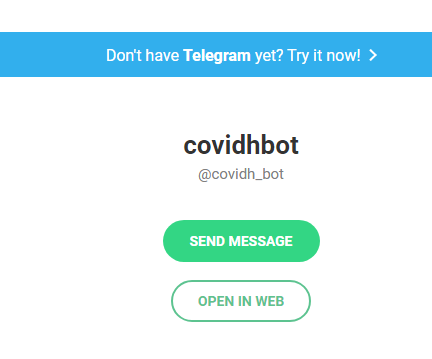
## Telegram Chat

Separate link and iframe for the team to chat with the bot get details about corona virus cases.

Note: Telegram can also be used from mobile and also it has got client version.

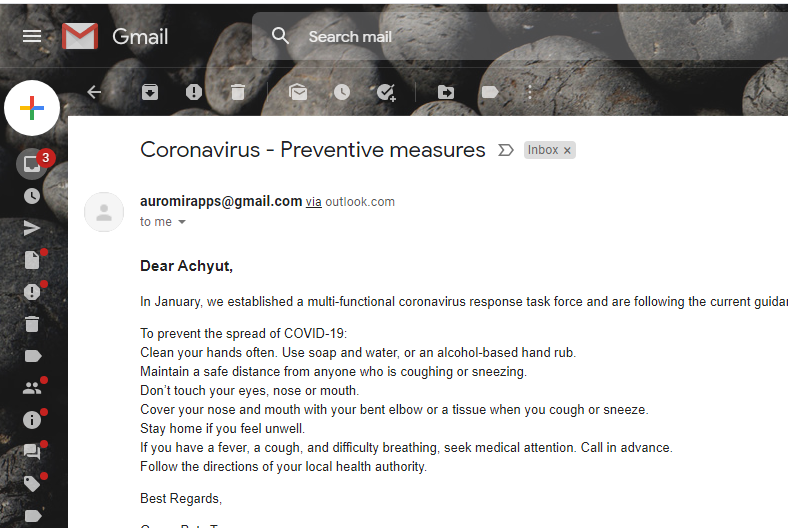
**IFrame**

**<a href='https://telegram.me/covidh\_bot'>@covidh\_bot</a>**



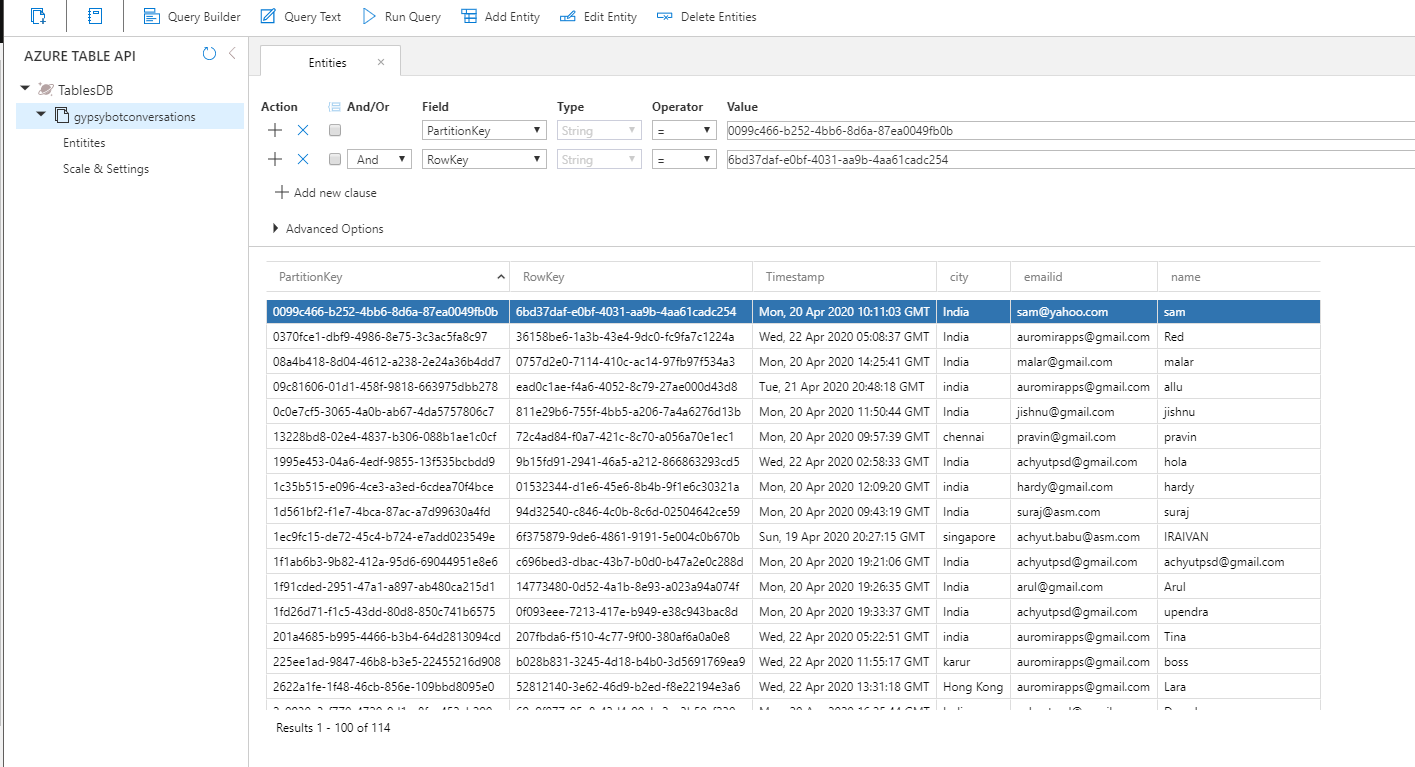
## Email Notification

Separate email should be sent about the preventive measures to be followed to avoid corona virus



## Data Storage

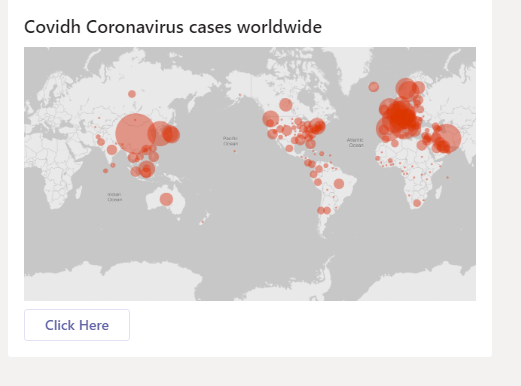
Data is stored in CosmosDB TableDB.



## Bot Card

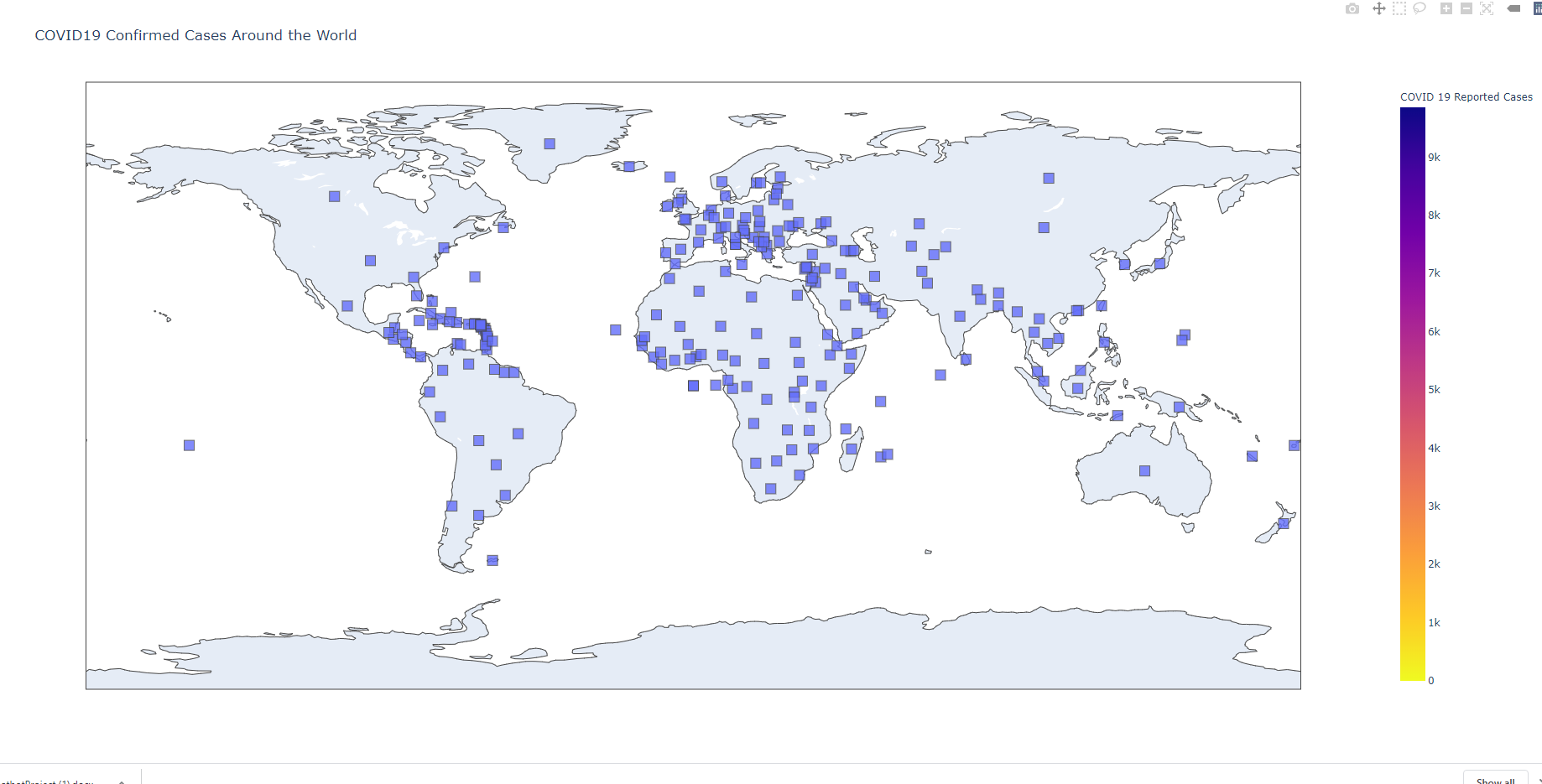
After sending email notification, Bot card needs to be sent for the user to click and see the world wide map related to corona virus. For this project we have used Microsoft

Link 🡪 <https://bing.com/covid/local/singapore?form=msntrk>



## World Coronavirus Map

In this project we have also built the worldwide corona map using the Web API but not able to embed it with bot due to time challenges. Below is the output.



# Technical Requirements

This section is to detail about the technical requirements need to build and deploy this project.

## Azure Subscription

Create a new account in Azure portal and register for free subscription.

## Resource Group.

Create new resource group and use the same for all the below services.

## Web App Service

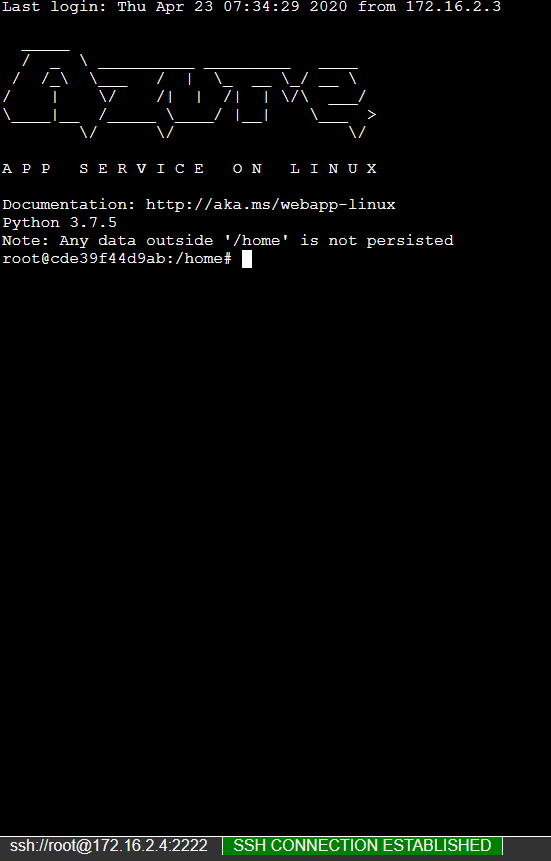
Create Web App Service with Runtime stack as Python 3.7. Free service plan is available for selected region. Note: Ensure to choose the closest region based on the residing county.

**How to check web app health.**

Web app service health can be checked using the secured console provided when the web app service is created. SSH connection button should be in green after deploying the website. It will show as orange or red for 2 reasons.

1. Region chosen is too far from the residing country and the service timeout due to slowness
2. Web is not deployed correctly

<https://covidhwebsite.scm.azurewebsites.net/webssh/host>



## Bot Channel Registration

Create new Bot Channel registration service using the azure account. Free tiers are available.

## LUIS App Registration

Register a new LUIS app using the azure account by visiting the site [**https://www.luis.ai**](https://www.luis.ai)

Create necessary intents, utterances to train and publish the service.

## Azure CosmosDB Creation

Create Azure CosmsDB service For this project Table API is used for data storage. It is not free.

# Source Code

Source code is available in my GitHub. Website is developed using Flask Web framework.

Link 🡪 <https://github.com/auromirapps/CovidhBot>

