# **IGNITE SOLUTIONS TASK-2**

Q.1. Hello World API

The attached file, hello\_world\_api.yml, provides the specification of a simple Hello World API that supports a single REST

GET request. You can open it in https://editor.swagger.io/ to read it clearly. Explore the page, clicking all options, to

understand all requirements before proceeding.

(a) Your task is to implement this API on your local machine i.e. you should be able to run the API on http://localhost:5000

As a test, if you run http://localhost:5000/hello?language=English on a browser on your machine, you should see a

response “Hello world”. SimIlarly, http://localhost:5000/hello?language=French should return “Bonjour le

monde” and http://localhost:5000/hello?language=Hindi should return “Namastey sansar”.

As a deliverable, please provide the code and the exact steps required to deploy the API implementation. You can

feel free to consult the internet but please make sure that the code is your own.

(b) Create a webpage to use the API you’ve defined in part (a) to allow the user to choose a language, call the API with

the language as a parameter and display the resultant hello world message on the webpage. Needless to say that

this webpage will work only on your machine.

(c) For additional credit make your working service available in a Cloud Provider of your choice e.g. AWS, Heroku,

GAE and send us the link.

## **Technology**

### **Front End**

* HTML5
* CSS3

### **Back End**

* Python 3
* Flask

## **Instructions to run the apI:**

## **1.Create a file .env in the following format**

SESSION\_TYPE=memcached

SECRET\_KEY=c0o1v2i3d4jgh798ujh98

FLASK\_ENV=development

UPLOAD\_FOLDER=static/uploads

### **2. Install the Requirements**

pip3 install -r requirements.txt

### **3. Run the Flask app**

flask run

## **STEPS to deploy the API on Heroku:**

**STEP 1 :** Create a virtual environment with pipenv and install **Flask** and **Gunicorn** .

pipenv install flask gunicorn

**STEP 2 :** Create a **“Procfile”** and write the following code.

web: gunicorn app:app

**STEP 3 :** Create **“runtime.txt”** and write the following code.

Python-3.7.2

**STEP 4 :** Run the vitual environment.

pipenv shell

**STEP 5 :**Initialize an empty repo, add the files in the repo and commit the changes.

$ git init

$ git add .

**STEP 6 :** Login to heroku CLI using

heroku login

Now, give the name of your web app, in my case i have given ignitesol

$ heroku create ignitesol-app

**STEP 7 :** Push your code from local to the heroku remote.

$ git push heroku master

## **CLICK ON THE LINK BELOW TO USE THE API HOSTED ON HEROKU:**

<https://ignitesol-app.herokuapp.com>

**Output after running the api:**







