**PROJECT: SMS FLASK APPLICATION PROJECT**

**PROGRAMMING LANGUAGE: PYTHON**

**ASSIGNED STUDENTS: GROUP**

There exists third-party services that allow you to send free sms from your device to anyone on your contact list. One of such third-party service is **"SMS CHEF"**. Your android or Iphone devices are added on your SMS CHEF account and they serve as the **"GATEWAY"** to send an SMS.   
**SMS CHEF** has an **API** that allows applications to make api requests containing the sms information a user wants to send. With the API, you could send an sms to a valid contact.

This project requires you build a Flask application that has **4** distinct parts/functions: You have been given a JSON file ***(contacts.json)*** which contains only a single contact. **(Mr. Wisdom)**. In the first part your application, display a front-end form which allows you or any user to enter a contact's name and the contact's phone number. Once the form is submitted, and submit button is clicked, this new contact's information should be added to your contacts.json file which the user does not see but exists in our back-end. As for the second part of your application, display an HTML form that ONLY has 3 fields - Name field***(text input)***, Message field***(Textarea)*** and Send SMS***(Button)***. In the third part of your application, user must be able to delete a contact by initiating it from the front-end. Contact to be deleted must exist in your **contacts.json** file which is on the backend. Here, display an HTML form that ONLY has 2 fields - Name field**(text input)** and Delete Contact***(Button)***.

**WHAT YOUR PROGRAM MUST ACCOMPLISH**

1. A valid account with **SMS CHEF** with your account details must be created with your device added to this account to serve as an **SMS GATEWAY**

2. You or any user must be able to add new contacts to the contacts.json file - Name of contact, Phone number of contact.

3. You must generate an **API** that displays all the contacts in the **contacts.json** file. Whenever new contact info has been added, the API must be updated and rendered (displayed accordingly) with this new information i.e. If for instance 2 new contacts are added from the front-end **(NAME, PHONENUMBER)**, the API endpoint should now display a total of 3 contacts with their contact information.

4. When an SMS is to be sent, ANY USER (OR YOU) is expected to fill out the second form **(NAME, MESSAGE, SEND SMS)** and click the SEND SMS button which should then the SMS. Your program should retrieve the contact number of that NAME from the backend of your application by itself, package the message to be sent and send the sms message to that specific contact using **SMS CHEF'S API** through your account/device.

5. If however, the NAME filled on the front-end HTML form does not exist in our contacts (i.e. contacts.json file), flash the following message to the user **"SMS REQUEST FAILED! CONTACT DOES NOT EXIST"**.

6. All sms should be send through your account/device which has already been added on SMS CHEF.

7. When a contact is to be deleted, USER is to fill out the third form **(NAME, DELETE CONTACT)** and click the DELETE CONTACT button(on the front-end) which should then delete that contact from our contacts.json file(in the backend). If contact does not exist and the delete attempt failed, display a flash message to the user.

8. Whenever the user makes a failed request such as failed sms, or failed delete attempt, redirect the user to the same front-end page they made the request from and display a flash message on that page.

9. When a contact is deleted from contacts.json file, it should no longer appear in the API which displays all our contacts.

10. All contacts passed into our **contacts.json** file must be in **Title Case** **(e.g. John Smith)**

11. User should be able to search for a contact and the contact details returned as JSON.

12. Generate a text file of all sms messages and recipients. Use this file to generate a csv file.

**GOOD LUCK!**