**Steps and structure for Flask application development**

Prerequisite: Python installed (version > 3.0 preferred).

* To check which python version is installed, open terminal and type “python - -version”

Step 1: Creating Virtual Environment (Windows):

* Python virtual environment provides the ability to isolate the python development projects from other python environments in the system. As a result, you can organize your packages better in your project and know exactly the necessary packages you need to run your code. Also, it avoids the flooding of your main python package directory with unnecessary python packages.
* Now for creating virtual environment: Create a new folder for the project. Example “capstone-project”.
* Open terminal and browse to the created folder using “cd” command.
* Now create a virtual environment for the project using the command “py -m venv *foldername*”. Example: “py -m venv deploy-calanus-detection-and-measurement-env”.
* Create another sub folder within “capstone-project” for project deployment. Example, a folder with name “deploy-calanus-detection-and-measurement-project”. This is where all our flask app files and folders will be kept.
* Now to activate the virtual environment. Open terminal and browse to virtual environment folder using “cd” command. Then after being in the virtual environment folder, type the command “Scripts\activate” to activate the virtual environment. Similarly, use “Scripts\deactivate” to deactivate the virtual environment.

Step 2: Installing Flask:

* After activating the virtual environment via terminal, type the command “pip install flask”. To check if “Flask” has been installed, type “pip list”.

Step 3: Opening project from IDE

* Open any IDE of your preference (Example VS Code).
* From IDE, click “Open Folder” and browse to project deployment folder. Example “deploy-calanus-detection-and-measurement-project”. Alternatively, drag and drop folder to IDE (action available in VS code).

Step 4: Structure of our flask application

* Contains Two folders: static and templates and python file app.py. Be careful that the folder names “static” and “templates” should be exactly that. By default, Flask looks for the relevant files in these folders using built-in functions.
* Folder “static” contains assets such as files, JavaScript files, and images used by the “templates” folder.
* Folder “templates” contains all HTML files.
* Python file “app.py” is the root of Flask application where all the routes and functions to perform for each action are defined.

Step 5: Running flask application:

* Ensure virtual environment is activated via terminal.
* After that, browse to project deployment folder (“deploy-calanus-detection-and-measurement-project”).
* Run the app.py file via terminal with the command “python app.py”
* Now copy the URL from the terminal to the browser.

