**SI 507**

**Homework 11: Flask**

**Homework Objectives**

- Understand how virtual environments work in python, and know why they are used

- Be able to install Flask and use Flask to create basic web applications

- Be able to use templates to integrate python and HTML code to create dynamic web pages

**Helpful Resources**

Readings

* Flask Web Development, [Chapter 1: Installation](https://umich.instructure.com/courses/196493/files/7519744/download?verifier=YgReo9hmJktqtqYTNef7MaeLsVHXZ4TVqD0fdPUu&wrap=1)
* Flask Web Development, [Chapter 2: Basic Application Structure](https://umich.instructure.com/courses/196493/files/7519745/download?verifier=1OrVZ2Z2ooBFz3W3KT9tdzOqxwsWLoe8Y7vx8OBo&wrap=1)
* Flask Web Development, [Chapter 3: Templates](https://umich.instructure.com/courses/196493/files/7519746/download?verifier=UF3Bffm4sFlhnGR16R4oUQxquuCprnw8DeTwvIud&wrap=1)

Other

* newsapi.org API documentation: <https://newsapi.org/docs>

For this assignment, you will be creating a basic Flask application that greets a user and tells them the top technology news stories for that day.

**Part 1 - Setting Up Flask [6 points]**

1. Create a directory for your homework assignment.
2. Inside that directory, set up a virtual environment called “flask\_ve” and, after activating it, install flask and requests modules.
3. Create the “requirements.txt” file
4. Add the “top\_headlines.py” file. You can download the starter version of the file [here](https://www.dropbox.com/s/qazv1578pu1yelr/top_headlines.py?dl=0). You will insert your code into it in part 2.
5. Create the “templates” directory.

Take a screenshot of your homework directory in Finder/Explorer. Upload it to Canvas as part of your submission. We will be looking to make sure that all the files are in the correct location.

**Part 2 – Greeting the user and showing current technology news [14 points]**

In this part, you will create a web app that greets the user by name and shows her/him the top five current technology news headlines. This content will be displayed by going to the following web page: http://127.0.0.1:5000/newsfor/<user\_name>. So, if, for example, one were to go to <http://127.0.0.1:5000/newsfor/hannah> one would see something like the following web page (although you can make it prettier, if you’d like, using CSS [see this week’s lecture slides for details of how]):

A screenshot of a social media post

Description automatically generated

***Print user’s name [4 points]***

* Using Flask and a template, write a function bound to the /newsfor/<name> route that displays the “Hello, <name>!” at the top of the page for any name that follows /newsfor/ in the URL.
* Make sure the name is capitalized when it’s displayed on the page, even if the user types the URL in the browser in all lowercase. So, if I were to go to localhost:5000/newsfor/pedja it should still say “Hello, Pedja!” at the top of the page.
* To make this work, you will need to write an appropriate template file.

***Query newsapi.org API for top 5 technology headlines [10 points]***

In your top\_headlines.py file, write code to retrieve the top 5 current technology headlines from the newsapi.org and display them for the user:

* To do this, you will need to access the newsapi.org API like we did in class earlier this term.
* In a file called secret.py, put your API key and name it like this:
  + API\_KEY = “<your api key>”
* Format the top 5 articles with the corresponding titles linked to their URLs as an unordered (number) list.

---------------------------------------------------------------------------------------------------------------------

**Extra Credit 1 - Make your Flask Application Interactive [2 points]**

Update your top\_headlines.py file to create another route /news/<topic> that will display top five current headlines for a topic of the user’s choice (e.g., science, sports, etc.). So, the user could go to <http://localhost:5000/news/impeachment> and the system would give back a page with the title “Top news about impeachment” and then list top five headlines about impeachment. Note that you should be able to use the exact same template file for this part as you did for part 2 of the homework.

**What to turn in:**

1. Upload your top\_headlines.py file to Canvas
2. Upload the screenshot of your working directory (for part 1)
3. You do not need to upload your secret.py file. As long as you named your api key API\_KEY, we should be able to run your code.