­­­­**Flask Application:**

For ease of grading I will put the file and line numbers below the required elements:

**Use at least 3 different heading styles (e.g. <h1>, <h2>, <h3>):**

Index.html <h1> @ 6; <h4> @ 16

Ncl.html <h2> @ 18

**Paragraph (<p>)**

index.html <p> @ 26

**Comments <!-- -->)**

Throughout all html documents

**Ordered list**

Ncl.html body

**Unordered list**

Ncl.html body

**At least 3 Links to other External Web Sites**

main.html <a> @ 50

clocks.html <a> @ 70; <a> @ 61

**Display the Date and Time on a Web page (Hint: Just use the Python datetime functions)**

Time.html

**Routes**

Three routes in the routes.py file; index, time, ncl. Each navigable in the navbar, or through links on the index page.

**Description:**

This app uses the Flask framework to create a three page site. The pages include the index, a time page, and a page on the NCL competition. In addition they all extend the main.html template which outlines the base for all of the pages. The base includes a navbar with links to each page, and a contact dropdown with a link to my Github account and an email link. The pages should be mobile responsive based on the use of bootstrap and the viewport attributes. For testing I will navigate to each page from the navbar, and test each link. To test the programming logic I will use the formula in routes.py as it is written, and then I will reverse it to make the date positive instead of negative. This will allow me to test the conditional logic in the ncl.html page and the formula in the routes.py file.

**Pages Test Set:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Case** | **Input** | **Expected Output** | **Actual Output** | **Passed** |
| 1 | localhost:5000 | index page with three columns of cards and expanded navbar | index page with three columns of cards | Y |
| 2 | localhost:5000; half page window | index page with stacked cards and minimized navbar | index page with stacked cards and minimized navbar | y |
| 3 | localhost:5000/time (link clicked from navbar or from index) | time page with three column cards and expanded navbar | time page with thee column cards and expanded navbar | Y |
| 4 | localhost:5000/time; half page window | time page with stacked cards and minimized navbar | time page with stacked cards and minimized navbar | Y |
| 5 | localhost:5000/ncl (link clicked from navbar or from index) | ncl page with three column cards and expanded navbar | ncl page with three column cards and expanded navbar | Y |
| 6 | localhost:5000/ncl; half page window | ncl page with stacked cards and minimized navbar | ncl page with stacked cards and minimized navbar | Y |
| 7 | localhost:5000/foo | 404 error | 404 error | Y |
| 8 | localhost:5000/ncl (with adjusted time formulat) | display secondary message on ncl page | display secondary message on ncl page | Y |
| 9 | index.html card 3 link to github | open github | open github | Y |
| 10 | time.html link to ntp.org | open ntp.org | open ntp.org | Y |
| 11 | time.html link to time.gov | open time.gov | open time.gov | Y |

Case 1:

Graphical user interface, website

Description automatically generated

Case 2:

Text

Description automatically generated

Case 3:

Graphical user interface, text, application

Description automatically generated

Case 4:

Text, application, chat or text message

Description automatically generated

Case 5:

A screenshot of a computer

Description automatically generated

Case 6:

Text

Description automatically generated

Case 7:

Graphical user interface, text, application

Description automatically generated

Case 8:

A screenshot of a computer

Description automatically generated

Case 9:

Graphical user interface, website

Description automatically generated

Case 10:

Graphical user interface, text

Description automatically generated

Case 11:

Map

Description automatically generated

**Pylint:**

Linting this time was interesting as the only python file I have is the routes.py file. In the beginning I had errors for the flask import which brought my code down significantly as it is so short. After I discovered the –disable flag, I was able to get to a ten with no problems. Below is the final lint with a score of 10.

Text

Description automatically generated

One thing I did notice is that on my desktop I had to use the –disable flag but on my laptop I did not. I am not sure why that is, but perhaps it has something to do with the way that I have my default directory structure for python setup. When I get home, I will go through and see if there is a difference between the way that python is installed on my laptop vs my desktop.