

COMPSCI 326 - Web Programming
Final Project #7 - Final Project HTML Front End
Due March 22, 2021, 11 pm EDT

<https://docs.google.com/document/d/1bgBEQErgFhIr-mpqu4WL1HWcCqrNfzDWWmg7LneGohc/edit?usp=sharing>

Project Description

The HTML front end should be a simple translation of the website wireframe to HTML and CSS source code. Mockup text and images should be filled in and the final appearance of the website determined. However, the javascript functionality of the website will not be completed at this time.

Where dynamically generated information would populate the website, filler data should be incorporated. For example, the restaurant website Service Listing Page included information about the status of customer orders. The HTML Front End does not have facilities for retrieving information from a database of customer orders. Instead, the HTML Front End simply hard-codes customer order information in a format that will facilitate incorporation of the back-end at a later date when database and javascript are completed.

Your HTML Front End should be hosted on a new Heroku site designed for the purpose of hosting your final project. Detailed instructions will be provided later in this document. Place the new web pages in the *public* directory, commit them to the github repository, and push the changes to the Heroku site. These pages will then be statically served by the web server and you will provide URLs to each of the pages as part of the turn-in.

Turn-In Instructions

Please fill out the yellow sections of the following table with the group member responsible and the web page URL for each of the pages in your web site. Also include the URL of the git repository (please make sure to include pklemperer@umass.edu as a collaborator in your repository). After filling out the table, please export this document as a PDF and upload it to gradescope.

Group Member	Web Page URL
Git Repository URL	

Part 1.

The instructions in part 1 should be completed by only one group member. The other group members should wait until that group member is done and then start at part 2.

1. Log in to the github.com using your account information
2. Open <https://github.com/ftPeter/web-programming-final-project-base> in a web browser
3. Click the fork button and create a fork local to your own account
4. git clone your new fork of the project and cd into that directory
5. heroku create
6. git push heroku master
7. heroku open /
 - a. At this point your browser should have opened to the heroku hosted web programming base website.
8. Add your group members as Heroku “collaborators” according to the instructions here: <https://devcenter.heroku.com/articles/collaborating>

Part 2.

The instructions in part 2 should only be attempted after one member of the group completes part 1. All group members need to complete Part 2 and beyond.

1. Open a browser and get to the [heroku dashboard view](#).
 - a. Click on the machine name shared with you by the first team member.
 - b. Click the “Deploy” tab
 - c. Scroll down to “Deploy using Heroku Git”
 - d. Follow the instructions for “cloning the repository” and “deploy some changes.”
2. You should now have a copy of your group’s website and the ability to push changes to the live website! Note that from now, all members should use this Heroku repository, not the fork one on Github.

Part 3.

The instructions in part 3 should be completed by each member of the group.

3. Write a new html file and save it to the directory
your-web-programming-final-project-base/views/pages with the filename yourname.ejs.

Example short html file (peter.ejs):

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
</head>
```

```
<body>
```

```
<h2>Test EJS File for Peter</h2>
```

```
</body>
```

```
</html>
```

Please replace the header “Peter” with your name.

2. Open the file index.js in the project folder and add the following lines before the line which says

```
.listen(PORT, () => console.log(`Listening on ${PORT}`))
```

```
.get('/yourname', (req, res) => res.render('pages/yourname'))
```

3. Commit the changes to the repository, push them to the heroku website, and open the new web page in a browser:

```
git add views/pages/yourname.ejs index.js
git commit -m 'my first web page in the repository'
git push heroku master
heroku open yourname
```

4. Congratulations, you now have made your first website using node.js on Heroku. Now you can replace the test ejs file you created with an html version of your wireframe from the previous project assignment.

Example HTML-only front ends are included in this project in the public directory. You can access them in the browser by using the following heroku command:

```
heroku open menu.html
```

Feel free to experiment with the provided web pages. I have included the source code for the pizza ordering website for you to use as examples of some of the capabilities of node.js with EJS and Heroku. *The Pizza ordering website will not perform as expected until the database has been set up. We will set up the database in a later assignment.*

Grading Rubric

- HTML neatly indented and follows course style guidelines
- CSS styling applied to create a neat and legible appearance
 - Like information grouped into columns or rows
 - Appropriate colors for warnings or error messages
- All web pages from the wireframe are provided as HTML pages
- Changes from wireframe are incorporated into the HTML pages
- All group members contributed at least one HTML page each