Project Plan for Cooking Recipes Web App

Web App Description

The Cooking Recipes Web App is designed to store and display cooking recipes. The app will be built using Flask, Jinja2, and Pandas as the primary technologies. Users will be able to view a homepage introducing the website, browse and view recipes, upload new recipes, remove existing recipes, and search for recipes based on names or ingredients.

Pages Description:

Homepage: An introduction to the website.

View Recipe Page: Displays individual recipes with images, ingredients, preparation instructions, and serving instructions.

Add Recipe Page: A form to upload new recipes to the server.

Remove Recipe Page: A page to remove recipes items from the server.

Search Recipe Page: A page to search recipes based on names or ingredients, leading to the Recipe Display Page.

Template vs. Dynamic Content:

The base.html template will contain the header, footer, and navigation menu, which will be common across all pages.

The Homepage and Recipe Display Page will be generated dynamically, pulling data from a CSV file to display recipes.

The Upload Recipe Page, Remove Recipe Page, and Search Recipe Page will have static content and use forms for user interaction.

Navigation:

The navigation menu in the base.html template will provide links to all pages of the web app.

On the Homepage, users will find links to the View Recipes page, the Search Recipes page, the Add Recipes page, and the Remove Recipes page.

Rubric Level:

The web app aims to achieve a 60-100% in the rubric.

Technical Requirements

The recipes will be stored in a CSV file with the following format:

Image, Ingredients, Preparation Instructions, Serving Instructions

Image file, Comma-separated ingredients, Text containing preparation steps, Text containing serving info  
Ex:

lasagna.jpg, "Pasta, tomato sauce, cheese, mushrooms, ground beef", "Layer the tomato sauce, cheese, mushrooms, and ground beef between the pasta, bake for 30 minutes at 375F","Serves 3-4 people"

Optional Technologies:

No additional optional technologies are planned for this project as of the time of this plan submission.

Deployment Plans:

The web app will be deployed on a web server using a service such as https://www.pythonanywhere.com/. The server will run the Flask application, and the CSV file containing recipes will be hosted alongside the app.

Proposed File Structure:

The template files for the web app include the following:

HTML Files: base.html, index.html, recipes.html, add\_recipe.html, remove\_recipe.html, search.html, view\_recipe.html, 404.html

The static files for the web app include the following:

CSS Files: main.css, normalize.min.css

JavaScript Files: main.js, jquery-1.11.2.min.js, modernizr-2.8.3-respond-1.4.2.min.js

Images: All images for recipes stored in the "img" folder.

Python Files: application.py, forms.py

The github link for this project:  
https://github.com/TClarke19/CP1895-Project

The deployed website for this project is:  
https://tcpy.pythonanywhere.com/