Introduction

This assignment is the last of six assignments. In this assignment, you will implement a shopping cart experience for your customers.

Before you begin this assignment, you must finish your previous assignment. All objectives listed for this assignment are to be made "on top" of your previous assignment.

This assignment is worth 9% of your final grade.

Reminder about academic integrity

You must comply with Seneca College's Academic Integrity Policy. Although you may interact and collaborate with others, this assignment must be worked on individually and you must submit your own work.

You are responsible to ensure that your solution, or any part of it, is not duplicated by another student. If you choose to push your source code to a source control repository, such as GIT, ensure that you have made that repository private.

A suspected violation will be filed with the Academic Integrity Committee and may result in a grade of zero on this assignment or a failing grade in this course.

Technical Requirements

- All back-end functionality **must** be done using **Node.js** and **Express**.
- You will use the **body-parser** module to handle form submissions.
- You will use the **express-session** module to handle user session state information.
- You will use **bcrypt.js** to encrypt user passwords.
- Your views **must** be created with **Express-Handlebars**.
- You **can use** a front-end CSS framework such as Bootstrap, Bulma or Materialize CSS to make your website responsive and aesthetically pleasing.
- You are **not allowed** to use any Front-End JavaScript Frameworks. For example, you may not use React, Vue, or Angular.
- You **must** use MongoDB as your database engine.

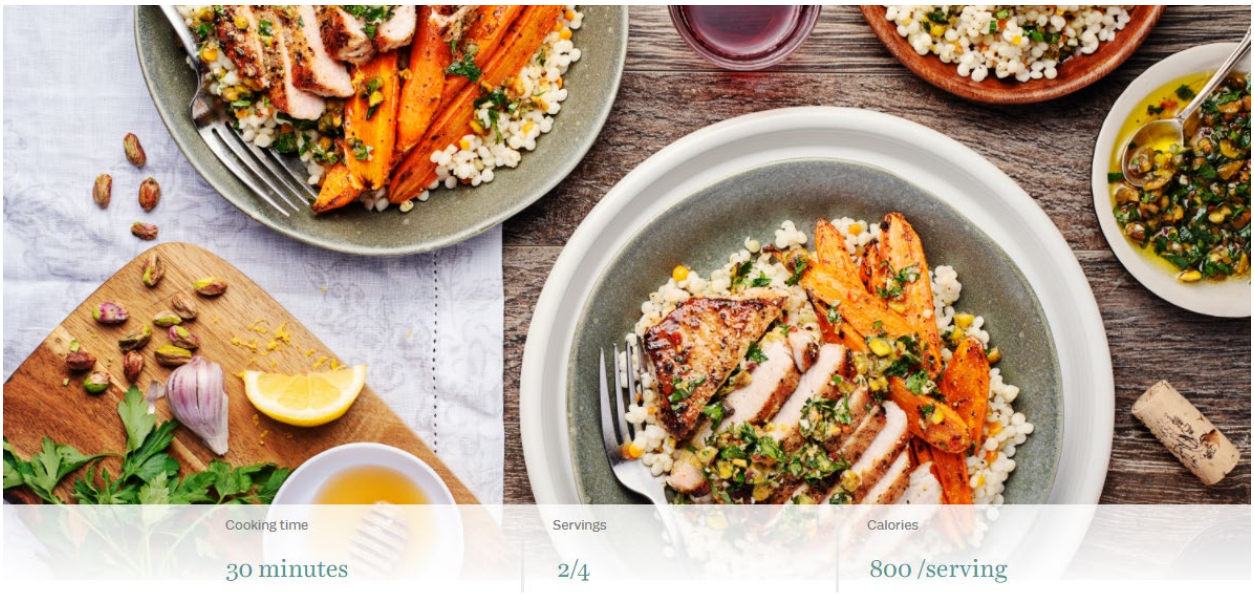
Objectives

Meal Kit Description Page

Logged in users must be able to "purchase" meal kits by adding selected meal kits to their shopping cart.

From the "on the menu" page, when a user clicks on a particular meal kit, they should be navigated to the Meal Kit Description page (of the clicked kit). From this description page, users can add the meal package to their shopping cart.

A meal kit description page may look like the following screenshot:



The meal kit description page of any meal package should include, at minimum, the following:

- Meal kit image
- Title
- Description
- Price
- Cooking time
- Number of servings
- Calories per serving
- Add to order button

When the user clicks the "Add to Order" button, the kit will be added to the user's shopping cart.

Shopping Cart Module

The customer's dashboard should have a link to the logged-in user's shopping cart. This page must display all the meal kits added to the shopping cart as well as a quantity for each. The page should also include a "Place your order" button and the order total.

When the "Place your order" button is pressed, your web application will send an email to the logged-in user's email, showing all the packages purchased along with the quantity for each. The email must include the customer's order total but do not worry about taxes.

After the email is sent, clean *(zero-out)* the shopping cart.

GitHub

You will continue to push your web application to a remote GitHub repository in your own account. **Do not forget to set your remote repository to private.** Add your professor as a collaborator so he/she can view your web application.

A realistic view of your progress must be showed by looking at your commits.

Heroku

You are required to deploy the working web application to Heroku. See the "Heroku Guide" on the web322 website for help on this topic. Do not forget to provide the URL to your professor.

Rubric

Criteria	**Not Implemented (0)** Little or no work done. Unacceptable attempt.	**Partially Implemented (3)** Work is minimally acceptable but is incomplete or needs significant modification.	**Fully Implemented (5)** Work is complete and done perfectly.
Meal Kit Details • Clicking a meal kit will show the details of the meal kit with all necessary fields. There is an "Add to Order" button.			
Ordering Module • Only logged in users can add meal kits to their shopping cart. • User's can view the purchased meal kits and the quantity in their shopping cart. The shopping cart includes the total amount (taxes are not necessary). • When the user clicks the "Place Order" button, the application must "clear" the shopping cart and send an email with the entire order information.			
Cloud Services • Website is deployed to Heroku and works correctly.			

Criteria	**Not Implemented (0)** Little or no work done. Unacceptable attempt.	**Partially Implemented (5)** Work is minimally acceptable but is incomplete or needs significant modification.	**Fully Implemented (10)** Work is complete and done perfectly.
Look and Feel • Overall site looks polished on all devices			

Total: 35 Marks

Note: Half marks may be awarded.

Submitting your work

Make sure you submit your assignment before the due date and time. It will take a few minutes to package up your project so make sure you give yourself a bit of time to submit the assignment.

1. Locate the folder that holds your solution files.

2. Compress the copied folder into a zip file. **You must use ZIP compression, do not use 7z or other compression algorithms or your assignment will not be marked.**

3. Login to My.Seneca.

4. Open the **Web Programming Tools and Frameworks** course area and click the **Project** link on the left-side navigator. Follow the link for this assignment.

5. Submit/upload your zip file. The page will accept three submissions so you may re-upload the project if you need to make changes. Make sure you make all your changes before the due date. Only the latest submission will be marked.