

Introduction

In this milestone, you will create prototypes for additional pages used in your web application. Prototypes show the visual elements of the page, but the page itself is not functional other than to link to the other page prototypes you have created.

Prototypes for the following pages are part of this milestone:

- Shopping cart page showing all the items the shopper has put into their shopping cart.
- Prototypes for pages that will eventually only be viewable to the admins of the site.

Specifications

Shopping cart page

This page lists all the items the user has added to their shopping cart. For now, only show a maximum of three items in the Cart. One of the items should have a quantity of 2 or more. The shopping cart page should have:

- A list of the items in the cart. Each item in the cart should have:
 - A small picture
 - The name
 - The price
 - A field to enter/edit the quantity.
 - A button to remove the item from the cart.
- The total per item type. So, if I am buying two pillows that cost \$4, then the total for this item is \$8.
- The shopping cart total:
 - The cost of all the items added together.
 - Plus 6.75% tax (just taxed on the items, not the fees below)
 - Plus, any delivery fees if applicable.
 - Plus, a service fee if all your items are downloadable.
- A checkout button.

As part of this milestone, update the shopping cart link in the navigation menu for all pages you have made so far to link to this prototype. Name the html file for this page *cart.html*.

Admin Pages

These are not intended for the general public view. Because of this, the design of the admin pages does not necessarily need to fully match the design of the main site. This milestone is requiring three pages be created to manage products in the site. The first is a bulk upload page that will allow the admin to upload content for a product onto the site. The second is a product listing page with admin actions, and the third is a page that allows the admin to edit all the features of a product including the image.

Admin Bulk Upload page.

Using this page, an admin user can select a file with bulk product data in JSON format (or more such as csv or text) and upload it into the app database. This page should have a form field to select a file to upload. This form should also have a submit button to allow them to upload the file. Include on the page an example file format so they know what format the data should be in and instructions for the type of files that you accept. Name the html file for this page *admin-upload.html*.

Admin Product Listing page.

This is a page that gives a quick glance of all listed products. For each product, include admin actions for the product such as delete, archive, edit. The list of products must have the ability to be filtered by searching for a specific product or category. The page should also allow them to add/create a new product that does not exist. A product should have the following attributes, in addition to other attributes unique to your product:

- Product id (not editable)
- Name
- Description
- Category id (or name)
- Image path
- Price

If the option to edit is selected, the product edit page is outlined below. Name the html file for this page *admin-products.html*.

Admin Product edit page.

A product edit page is needed to allow admins to modify details of a product. All the fields (except the product id) should be editable on this page. Product image should just be a field, we will assume that the image is already on the server. Name the html file for this page *product-edit.html*.

Admin Navigation.

Since this is an admin view, the above pages should be thematically similar. Additionally, the admin navigation does not need to match the public pages, but it should still have some things in common like the logo (or a similar version of the logo). The navigation should have:

- A link to bulk upload.
- A link to the admin product listing page.
- A link that will sign the admin out.

Technical Requirements

- All CSS rules should reside in a single file named *admin.css* and be applied to HTML elements via appropriate class or id attributes.
- Follow the class coding guidelines.
- Any other resources needed by your pages (i.e. image files) should reside in the sub-directory "images". Make sure your relative links to the files work as desired.
- You can use "href=#" for links to resources you have not created yet, however, all the pages you have prototyped so far should be reachable by navigation as opposed to opening individual files.

Submission: Add your work the project subfolder in GitHub pages. Submit a link to the deployed pages at github.io. When you click on the link for your deployed pages, ensure it is possible to navigate through all the other pages by clicking on hyperlinks.