**Ask for the Home Page**

Request components:

* Method: GET
* URL: /
* Headers: none
* Body: none

Response components:

* Status Code: 200
* Headers:
  + Content-Type: text/html
* Body: HTML page with navigation links to other pages

In your browser, navigate to [http://localhost:5000](http://localhost:5000/). Congratulations! You have performed a GET request to / showing the home page of our e-commerce website.

**Ask for a page that doesn't exist**

Request components:

* Method: GET
* URL: /isnd
* Headers:
  + Host: site.com
* Body: none

Response components:

* Status code: 404
* Headers:
  + Content-Type: text/html
* Body: HTML page saying 404 NOT FOUND

**Ask for the products list page**

Request components:

* Method: GET
* URL: /products
* Headers:
  + Host: site.com
* Body: none

Response components:

* Status code: 200
* Headers:
  + Content-Type: text/html
* Body: HTML page showing product list page

**Ask for the product detail page**

Here's an example product on the server:

| **DETAIL** | **VALUE** |
| --- | --- |
| productId | 1 |
| name | "Facial Cleansing Brush" |
| description | "Reaches deep pores to cleanse oil, dirt, and blackheads." |
| price | 23.99 |
| categories | "beauty", "electronics" |

Request components:

* Method: GET
* URL: /products/1
* Headers:
  + Host: site.com
* Body: none

Response components:

* Status code: 200
* Headers:
  + Content-Type: text/html
* Body: HTML page showing a one product

**Ask for the create new product page**

Request components:

* Method: GET
* URL: /products/new
* Headers:
  + Host: site.com
* Body: none

Response components:

* Status code: 200
* Headers:
  + Content-Type: text/html
* Body: HTML page to create new product

**Submit a new product**

Remember, for a traditional HTML web server, whenever a successful POST request is sent to the server, the server should respond with a redirection to a page.

After successful submission, user should be looking at the product detail page.

Here are the categories on the server:

| **TAG** | **NAME** |
| --- | --- |
| grocery | Grocery |
| electronics | Electronics |
| beauty | Beauty |
| toys-games | Toys and Games |
| health | Health |
| furniture | Furniture |
| clothing | Clothing |

Request components:

* Method: POST
* URL: /products
* Headers:
  + Host: site.com
  + Content-Type: x-www-form-urlencoded
* Body: The new product

Response components:

* Status code: 302 /products then 200 /products/productId
* Headers:
  + Content-Type: text/html
* Body: redirect to product detail page

**Ask for the edit product page**

Request components:

* Method: GET
* URL: /products/productId/edit
* Headers:
  + Host: site.com
* Body: none

Response components:

* Status code: 200
* Headers:
  + Content-Type: text/html
* Body: HTML page for editing a product

**Submit an edit for an existing product**

After successful submission, user should be looking at the product detail page.

Request components:

* Method: POST
* URL: /products/productId
* Headers:
  + Host: site.com
  + Content-Type: x-www-form-urlencoded
* Body: The edited product

Response components:

* Status code: 302 /products/ productId then 200 /products/ productId
* Headers:
  + Content-Type: text/html
* Body: HTML page for product detail

**Submit a delete for an existing product**

After successful submission, user should be looking at the products list page.

Request components:

* Method: POST
* URL: /products
* Headers:
  + Host: site.com
  + Content-Type: x-www-form-urlencoded
* Body: The product to be deleted

Response components:

* Status code: 302 /products/5/delete THEN 200 /products
* Headers:
  + Content-Type: text/html
* Body: HTML page for product list

**Submit a new review for a product**

After successful submission, user should be looking at the product detail page.

Here's an example review on the server:

| **DETAIL** | **VALUE** |
| --- | --- |
| reviewId | 1 |
| comment | "I love this product!" |
| starRating | 5 |
| productId | 1 |

Request components:

* Method: POST
* URL: /products/productId/reviews
* Headers:
  + Host: site.com
  + Content-Type: x-www-form-urlencoded
* Body: The review for the chosen product

Response components:

* Status code: 302 /products/productId/reviews THEN 200 /products/productId
* Headers:
  + Content-Type: text/html
* Body: HTML page for product detail

**Ask for the edit review page for a product**

Request components:

* Method: GET
* URL: /reviews/reviewId/edit
* Headers:
  + Host: site.com
* Body: none

Response components:

* Status code: 200
* Headers:
  + Content-Type: text/html
* Body: HTML page for editing a review

**Submit an edit for an existing review**

After successful submission, user should be looking at the product detail page.

Request components:

* Method: POST
* URL: /reviews/reviewId
* Headers:
  + Host: site.com
  + Content-Type: x-www-form-urlencoded
* Body: The edited review of an existing one

Response components:

* Status code: 302 /reviews/reviewId THEN 200 /products/productId
* Headers:
  + Content-Type: text/html
* Body: HTML page for product detail

**Submit a delete for an existing review**

After successful submission, user should be looking at the product detail page.

Request components:

* Method: POST
* URL: /reviews/reviewId/delete
* Headers:
  + Host: site.com
  + Content-Type: x-www-form-urlencoded
* Body: The review to be deleted

Response components:

* Status code: 302 /reviews/reviewId/delete THEN 200 /products/productId
* Headers:
  + Content-Type: text/html
* Body: HTML page for product detail

**Ask for all the products in a particular category by tag of the category**

Request components:

* Method: GET
* URL: /categories/categoryTag/products
* Headers:
  + Host: site.com
* Body: none

Response components:

* Status code: 200
* Headers:
  + Content-Type: text/html
* Body: The products under that category

**Ask for the best-selling product**

Look for clues in the HTML pages from the prior responses for what the route should be.

Request components:

* Method: GET
* URL: products/best-selling
* Headers:
  + Host: site.com
* Body: none

Response components:

* Status code: 200
* Headers:
  + Content-Type: text/html
* Body: HTML page for the best-selling products