Assignment - Product Purchasing Web Application

Create a Web App consisting of four (4) customer facing views:

1. Product Listing
2. Product Detail
3. Product Purchase
4. Purchase Complete

1 - Product Listing

Displays a list of products along with pricing information for each product. Each product will link to a product detail page.

> The list of products is available and expected response can be seen in products_list.schema

Assume the API endpoint is

- https://test.com/products
- https://test.com/products/XXX

Where XXX is replaced with the appropriate product id

2 - Product Detail

Displays the details of the product along with a purchase option if the product has inventory.

> *The list of product details is available and expected response can be seen in products_detail.schema*

Assume the API endpoint is https://test.com/products/XXX/details

Where XXX is replaced with the appropriate product id

Functional requirements/notes: inventory_on_hand should be checked upon page load and the purchase option only displayed if the product is available (inventory_on_hand > 0).

The purchase option will take you to the Product Purchase page.

3 - Product Purchase

Provide a form to collect Customer Details and purchase the product by POSTING the correct information to the API. Save the customer details, purchased product details and response from the API to web app storage.

Submit a purchase request via API POST:

https://test.com/products/XXX/purchase

Where XXX is replaced with the appropriate product id

> *You can view the expected payload in the create_order schema.*

4 - Purchase Complete

This landing page will show the customer whether the purchase was successful or not.

Store the following data for each purchase:

- Confirmation Code (confirmation_code)
- Product Name
- Product Price
- Customer Name Customer Email Address Customer Phone Number

FAQS

What should the pages look like?

We would the pages to look presentable but we do not expect you to spend a lot of time on layout. Programmer styling is okay, it doesn't need to look pretty!

Assignment Delivery

Have your project in a folder with proper structure and instructions on how to run and interact with the site.

Bonus

Include tests and coverage