Secretlab
Tech Exercise
(Front End)

In order to avoid any misunderstandings for both parties in the future, you should proceed with this exercise only if you agree with all the statements below:

1. *You agree and recognise that Secretlab may already be independently working on the same project. You agree that there is a likelihood that Secretlab's employees or contractors may author, create or develop the same or similar design or content as covered by your submission now or in the future, and, therefore, you agree that any similarity between Secretlab's products and your submission is merely coincidental and cannot serve as circumstantial evidence of any misappropriation.*
2. *You will not submit to us any information in breach of any confidentiality, or any contractual obligations you have towards others.*
3. *You agree that no contractual or working relationship is being created between you and Secretlab solely by your participation in this exercise.*
4. *You agree that any materials provided for this exercise are to be kept strictly private and confidential.*

Secretlab Front End Task: 3 page Ecommerce Webstore

Test Conditions

1. You are only allowed to use the following packages
   a. NextJS
   b. Tailwind
   c. Typescript
   d. State management library (Redux, Zustand, React Context)
   e. Animation library (Framer Motion, three.js)
2. All the product data to be fetched from dummyjson.com product API

A good submission will have the following characteristics:

1. Clean and extendable code.
2. Handles a variety of screen solutions responsively.
3. Handles potential device and browser compatibility issues.
4. Well thought out visual and layout for the web page.
5. Unit testing coverage for automated testing of website functionality

Should you be visually or creatively inclined, you may express them within this test though css styling/animation, etc. Otherwise, please follow the test reference accordingly.
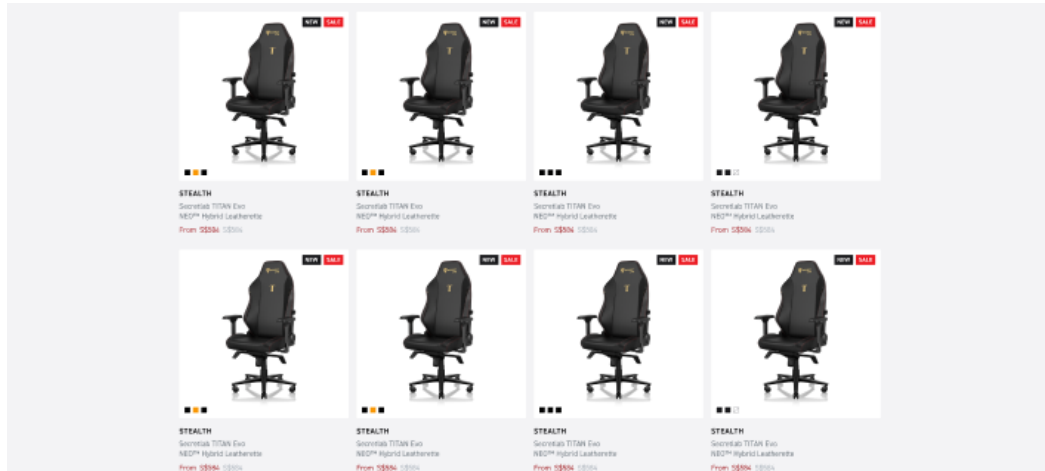

Instructions

Your task is to create a front-end only application that will allow users to purchase products from a Ecommerce website, it will consists mainly of 3 pages (see reference for details)

There is **no** need to integrate the checkout page to an actual payment gateway system

Codes are to be hosted on a public GIT repository (Bitbucket, Github, etc). Include a Readme file explaining your thought process and the features of the application including limitations that you had to workaround. Actual website must be hosted on a live server e.g Vercel with the URL provided to us
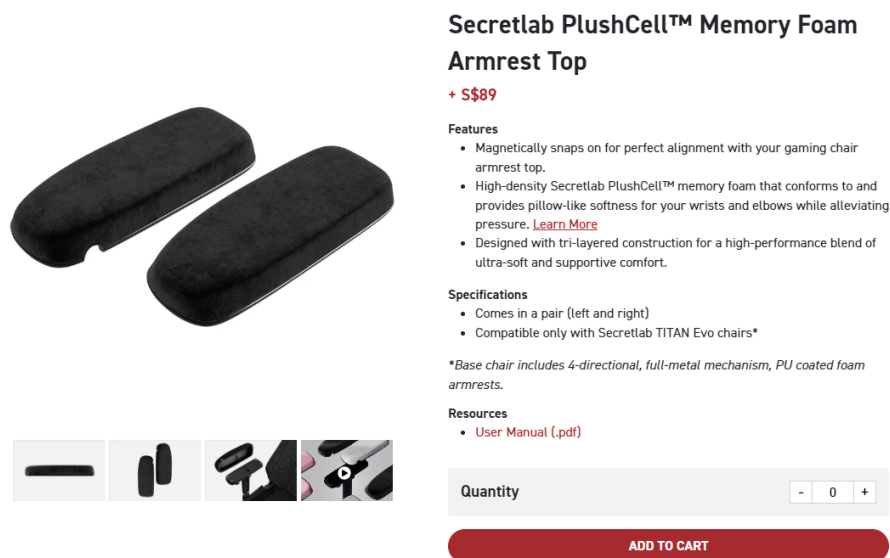
- **Catalog Page**

  - A List of products showcased in a grid format
  - A filter function that allows the customer to filter based on categories, rating and price range
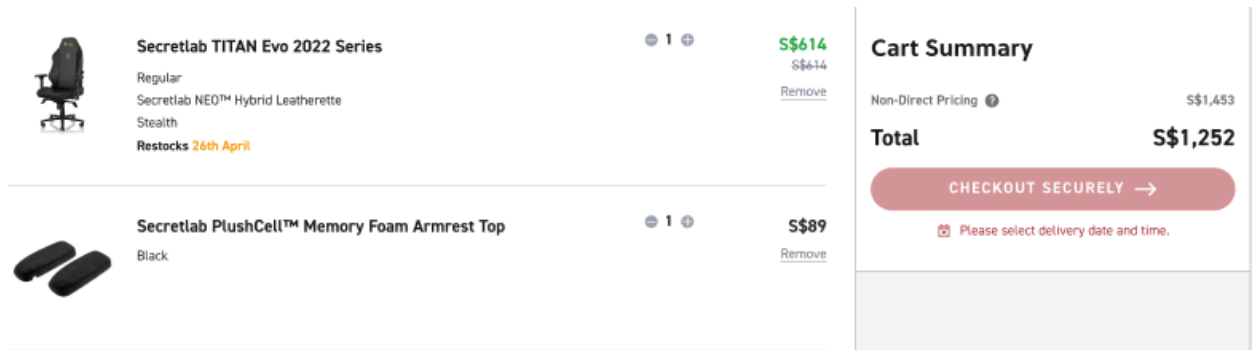
- **Product Detail Page**

  - A page for the customer to view the product in detail and to add the product to their cart
  - When the customer enter a product detail that has been added to the cart, the quantity should reflect what has been added to the cart

● **Cart Page**

○ To view the products that the customer added to their cart
○ To manage the cart (add or reduce the quantity, or remove the product from the cart entirely)
○ All cart update quantity and prices to update dynamically without needing for page refresh
○ Live calculation of all the prices (including the cart summary total) based on the quantity added/removed from the cart

Requirements

● A mini cart on the header that is available in all pages to show the live quantity of products that is added to the cart
● All data required to be fetched through proxy from you NextJS app API
● All pages to be responsive in desktop, tablet and mobile
● All elements to be componentized and to be made reusable, managed in an easy to navigate folder structure
● All components, layout and templates to be developed from scratch, not allowed to use any off the shelf templates or packages
● All components that has logic must have a test unit (Jest, React testing library)
● Push your codes to github and deploy your app to Vercel.
● Set up CI/CD that would automatically deployed your app to Vercel once you push your codes to your master/main branch
● Provide us both the github link and the Vercel domain for your app

Bonus points:

● Create a feature to compare the various product details on the same screen
● Use Zustand for the state management
● Show off your creative skills by adding as many animations as possible
● Integrate to a test payment gateway (Huge Plus)