



Tech task (Frontend dev role)

Task description

As an assignment, you need to create a PoC of the e-commerce solution. One of our clients wants to have an application where he could sell images and artworks. He gave us designs and we need to transfer his vision to real code.

Scoring

To meet requirements you need to accomplish between **3 and 7 features** (when 3 is a minimum and 7 is a maximum).

Tech Stack

We want to encourage you to use libraries and tools that you feel comfortable with. Our favorite stack is Next.JS + Typescript but you can propose your own solutions. The only thing that is mandatory is the usage of React.JS.

Database Design

You can create your own database of products or use our schema proposition. Create min 20 products. You could get images from <https://www.pexels.com/search/lifestyle/>

```
{
  "products": [
    {
      "name": "Red Bench",
      "category": "people",
      "price": 3.89,
      "currency": "USD",
      "image": {
        "src": "",
        "alt": ""
      },
      "bestseller": true,
      "featured": false,
      "details": null
    },
    {
      "name": "Egg Balloon",
      "category": "food",
      "price": 93.89,
      "currency": "USD",
      "image": "",
      "bestseller": false,
      "featured": false,
      "details": null
    },
    {
      "name": "Man",
      "category": "people",
      "price": 100,
      "currency": "USD",
      "image": {
        "src": "",
        "alt": ""
      },
      "bestseller": false,
      "featured": false,
      "details": null
    },
    {
      "name": "Architecture",
      "category": "landmarks",
      "price": 101,
      "currency": "USD",
      "dimensions": {
        "width": 1020,
        "height": 1020
      },
      "image": "",
      "bestseller": false,
      "featured": false,
    }
  ]
}
```

```

      "details": null
    },
    {
      "name": "Samurai King Restling",
      "category": "landmarks",
      "price": 101,
      "currency": "USD",
      "image": {
        "src": "",
        "alt": ""
      },
      "bestseller": false,
      "featured": true,
      "details": {
        "dimensions": {
          "width": 1020,
          "height": 1020
        },
        "size": 15000,
        "description": "So how did the classical Latin become so incoherent? According to McClintock, a 15th century typesetter likely",
        "recommendations": [
          {
            "src": "",
            "alt": ""
          },
          {
            "src": "",
            "alt": ""
          },
          {
            "src": "",
            "alt": ""
          }
        ]
      }
    }
  ]
}

```

Features:

1. Featured Product

One of the products should have a flag that it's a featured artwork. It should be displayed above the product list.

2. Product List

The product list should contain 6 artworks. After hovering over the image, you should display the "add to cart" bar. Remember that some products have the bestseller flag.

3. Add to Cart

After clicking "Add to cart" in the cart icon in the header should appear the badge with the counter of elements in the cart. Each 'Add to cart' action should open the cart dropdown with items. The dropdown should be also visible after clicking the basket icon in the header. Clicking in the "Clear" button should remove items from the cart and hide the dropdown.

4. Pagination

Products should be paginated. On one page should be 6 items. The pagination should show the current page. Hide the 'prev' arrow on the first page and hide the 'next' arrow on the last page

5. Sorting

Implement 2 types of sorting: alphabetically or by price. Use basic HTML select. Clicking on arrows should change the order to 'ascending' or 'descending'.

6. Filtering

Products should be filterable. You can filter products by multiple categories (multiple filters) and only one price range (single filter). Using filters should affect the pagination. Try to build filter options dynamically (don't hardcode them).

7. Web performance*

Optimise Your website using lighthouse reports. Try to achieve 80+ score in web core vitals. (mobile and desktop)

8. **API implementation***

Try to implement your own database solution. (You can use tools like Fauna, Hasura, Amplify, Firebase)

Designs:

<https://www.figma.com/file/wYrjKlxB3g80yge1kcahWx/Bejamas-Recruitment-task?node-id=7%3A376>

Sent the finished solution as a .zip file to hr@bejamas.io