KUTAISI INTERNATIONAL UNIVERSITY

THESIS IN SCRIPTING LANGUAGES PROJECT:

MARKETING PLATFORM

Student Name: Nino Bendianishvili

07/04/2024

Abstract

The thesis presents the development of a Marketing Platform website created for Scripting Languages, using HTML, CSS, Javascript & React. The project involved designing a multi-page, interactive website with various functionalities, including but not limited to a gallery of merchandising products, their filters, sorts, search features & user profile authentication. A set of criteria was used to make sure the project is in order with all the requirements & is user-friendly.

Project Goals

The primary goals of this project were to:

1. Create a website for a marketing platform.
2. Implement key functionalities using React (product galleries, search, sort, filter, user log-in & cart systems and contact form).
3. Ensure user-friendly interface.
4. Design a visually aesthetic interface.

Project Description

Introduction

The Marketing Platform project provides users with a website endorsing a musical band, prompting users to buy their merchandise products. Using web technologies (mainly React JS, CSS and HTML), the platform aims to enhance user experience with a clean functionality.

Project Criteria

The project was divided into several key tasks:

1. Develop a multi-page website.
2. Create a product catalog (with the page featuring sort menus & filters).
3. Create profile cards for products.
4. Create a contact form.
5. Create user-friendly design.
6. Create a functional navigation bar.
7. Create an authorization field.
8. Create a contact form.

Ways of Solutions

To meet the criteria, following solutions were implemented:

Multi-Page Website

- **Home Page:** Designed using React & CSS, featuring mouse-over effects, including a product catalog.
- **Navigation Bar:** A navigation bar with links to respective pages is implemented.
- **Profile Page:** A login & profile pages are created, with username & mail information.
- **Cart:** A cart is displayed in the sidebar, with the calculator responding to the product prices that the user adds to cart.

Product Catalog

- **Clickable Images:** Images link to the product pages, with mouseover effect making it visible which product is under selection.
- **On-sale:** Products that are currently on sale show specific red sign, also possible to filter only the items that are on-sale.

Fast-Accessible Features

- **Authorization:** Navigation bar leading to Login, Signup and Profile pages.
- **Language Switcher:** Navigation bar having a drop-down menu which allows users to switch between three languages.
- **Facebook Link:** As the site is informational, it includes the link in the navigation bar leading to the Facebook page.

Filter & Sort

- **Filtering Parameters:** Users can filter through the products that are on-sale or show all the products in the catalog. Also, they can feature according to the merch type.
- **Sort Parameters:** Users can sort the items according to their price or their name, ascending & descending.

Search Field

- **Search Box:** Users can search through the products of their choice with the relevant keyword.

Contact Form

- **Contact Information:** Includes links, newsletter subscription field, textboxes, check- and radio boxes & a submit button.

Cart

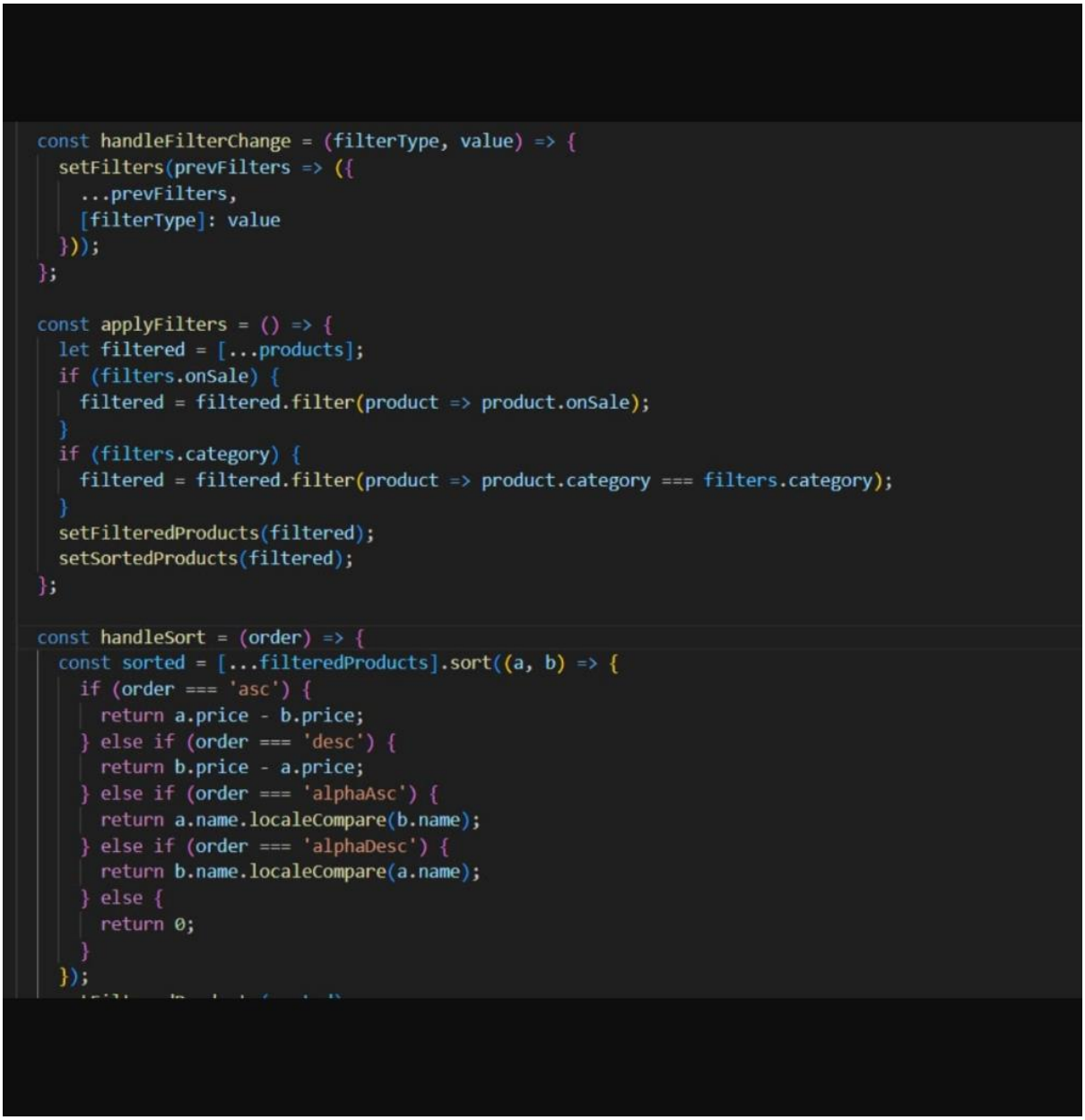
- **Add to Cart:** Users can add the products of their choice to their cart, displayed in the sidebar on the left.
- **Calculator:** The prices of the products in the cart are summed up dynamically.

Web Page Design

- **User Friendly Design:** Users can easily navigate between the pages with the design.
- **User Friendly Layout:** The items follow the distribution standards for the CSS objects.

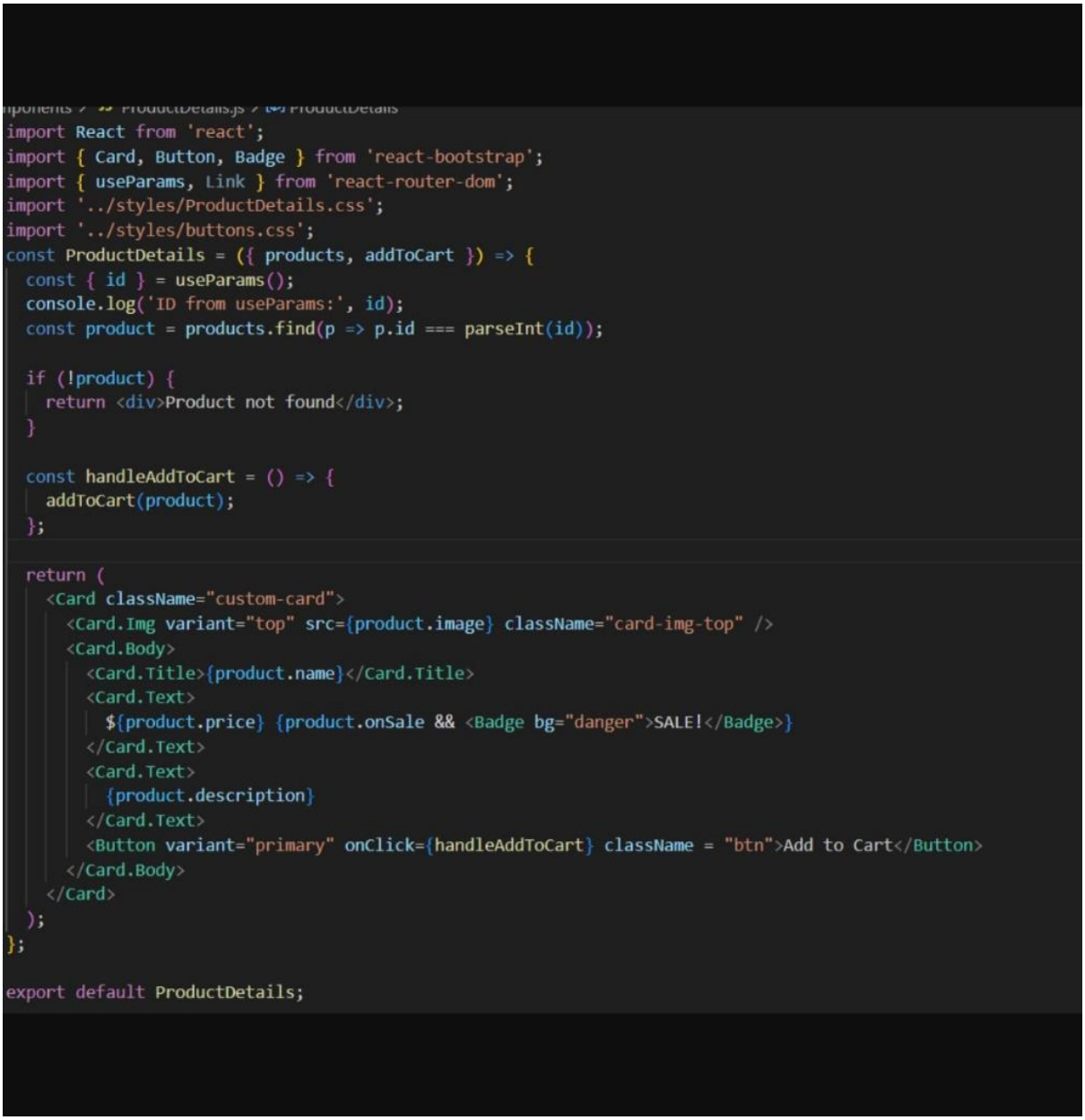
Code Fragments

1. Filter & Sort Code



```
const handleFilterChange = (filterType, value) => {
  setFilters(prevFilters => ({
    ...prevFilters,
    [filterType]: value
  }));
};

const applyFilters = () => {
  let filtered = [...products];
  if (filters.onSale) {
    filtered = filtered.filter(product => product.onSale);
  }
  if (filters.category) {
    filtered = filtered.filter(product => product.category === filters.category);
  }
  setFilteredProducts(filtered);
  setSortedProducts(filtered);
};

const handleSort = (order) => {
  const sorted = [...filteredProducts].sort((a, b) => {
    if (order === 'asc') {
      return a.price - b.price;
    } else if (order === 'desc') {
      return b.price - a.price;
    } else if (order === 'alphaAsc') {
      return a.name.localeCompare(b.name);
    } else if (order === 'alphaDesc') {
      return b.name.localeCompare(a.name);
    } else {
      return 0;
    }
  });
```

2. Product Details Code

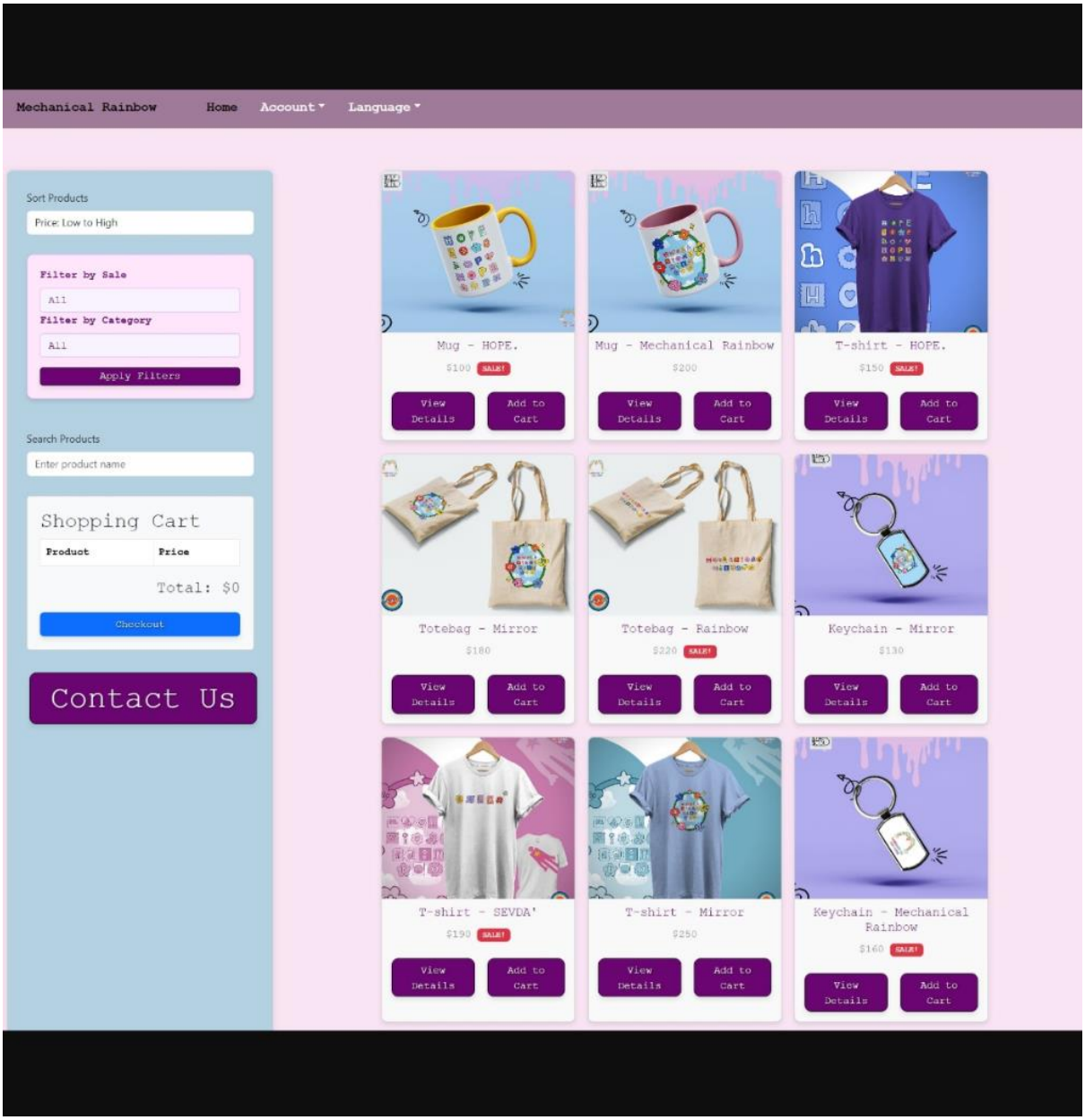


```jsx
import React from 'react';
import { Card, Button, Badge } from 'react-bootstrap';
import { useParams, Link } from 'react-router-dom';
import '../styles/ProductDetails.css';
import '../styles/buttons.css';
const ProductDetails = ({ products, addToCart }) => {
  const { id } = useParams();
  console.log('ID from useParams:', id);
  const product = products.find(p => p.id === parseInt(id));

  if (!product) {
    return <div>Product not found</div>;
  }

  const handleAddToCart = () => {
    addToCart(product);
  };

  return (
    <Card className="custom-card">
      <Card.Img variant="top" src={product.image} className="card-img-top" />
      <Card.Body>
        <Card.Title>{product.name}</Card.Title>
        <Card.Text>
          ${product.price} {product.onSale && <Badge bg="danger">SALE!</Badge>}
        </Card.Text>
        <Card.Text>
          {product.description}
        </Card.Text>
        <Button variant="primary" onClick={handleAddToCart} className = "btn">Add to Cart</Button>
      </Card.Body>
    </Card>
  );
};

export default ProductDetails;
```

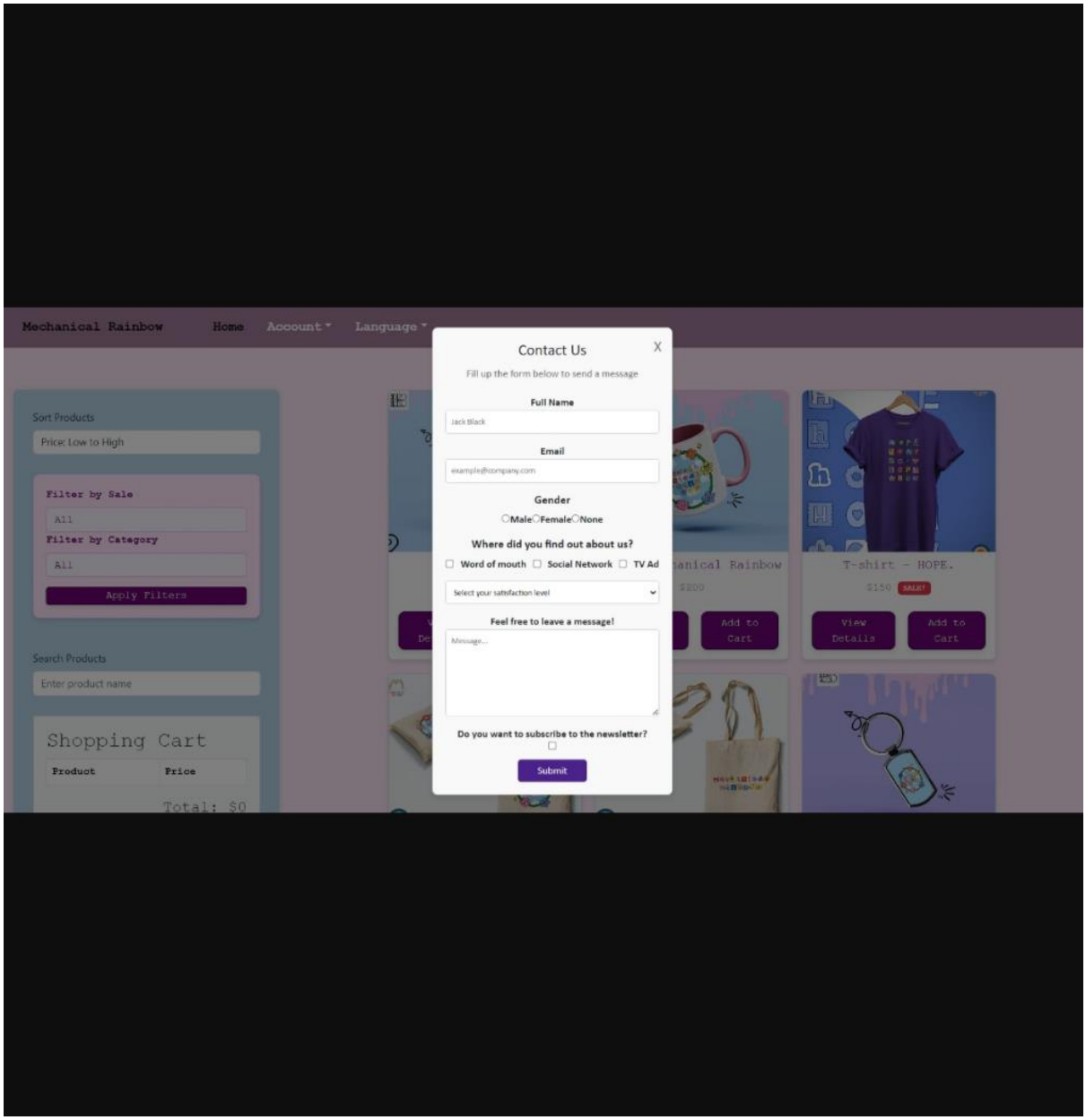
3. CSS Fragments from the profile, along with the component list from the folder

Screenshots from the Final Product

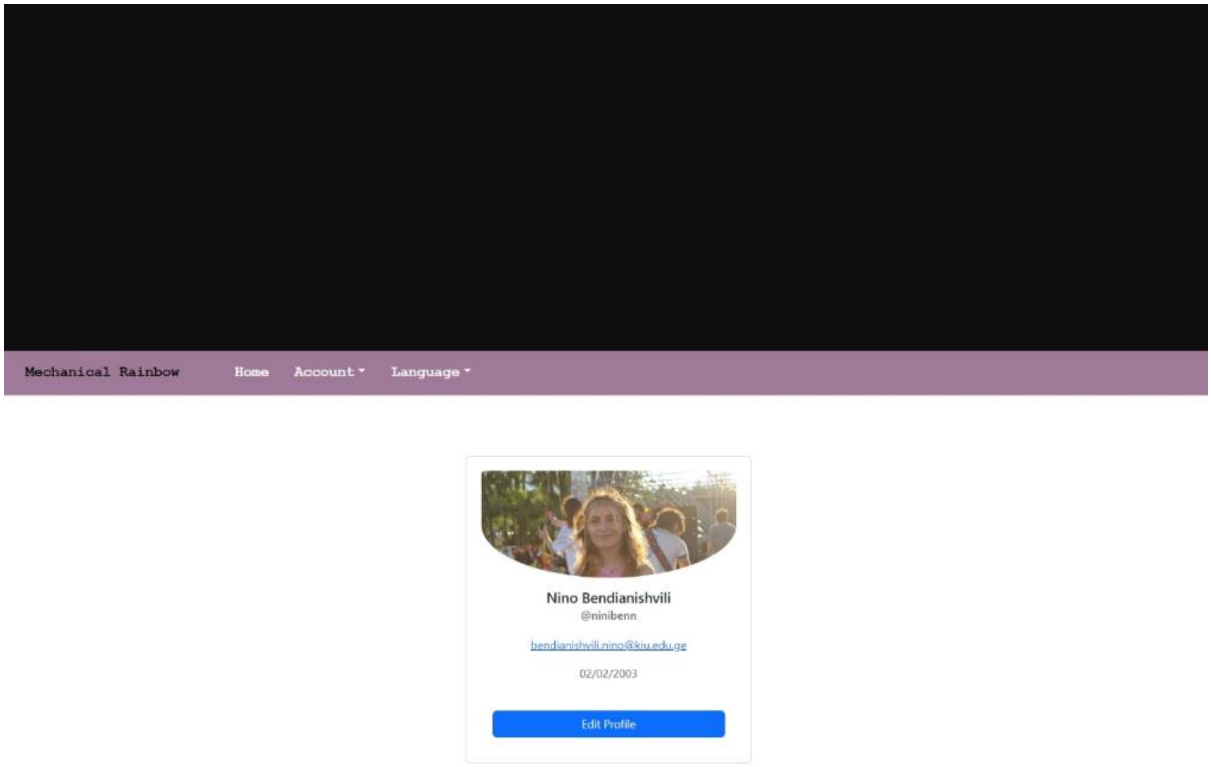
1. Home Page including the product catalog



2. Contact Form, including the text boxes, different selection types & a newsletter subscription



3. A Profile Card



Conclusions

The Marketing Platform created is an example of the effective use of the aforementioned languages to create a comprehensive, user-friendly platform. The project meets the evaluation criteria, provides valuable experience in front-end development and UI design, as well as integrating different functionalities. Future improvements could include adding different functionalities, including but not limited to payment, linking with social media pages, embed links & connecting to back-end of the website.

References

1. Fetching data using Axios: Simplest way to use axios to fetch data from an api in ReactJS | by Manish Mandal | How To React | Medium
2. Simplify CSS writing using cheatsheet and simulator: CSS Cheat Sheet - Interactive, not a PDF | HTMLCheatSheet.com
3. Simple Filtering: How to Filter an Array of Objects in React | bobbyhadz
4. Sorting Techniques: How to sort table data with React (freecodecamp.org)
5. For Design Inspirations: 19 Best Product Page Design Examples for Inspiration in 2024 - Shopify
6. Contact me Form in React: How to Create Contact Form with React | Mailtrap Blog