**Міністерство освіти і науки України**

**A blue and red logo

Description automatically generated**

**ВСЕУКРАЇНСЬКА ЦЕНТРАЛЬНА СПІЛКА СПОЖИВЧИХ ТОВАРИСТВ (УКРКООПСПІЛКА)**

Рівненський кооперативний економіко-правовий фаховий коледж

# Звіт

***З технологічної практики***

***студента Свинобоя Максима Олеговича***

групи ІТ-41

Період практики з «27» січня2025 року до «21» лютого 2025 року

Галузь знань ***12 Інформаційні технології***

Спеціальність ***121 Інженерія програмного забезпечення***

***Місце практики***

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

***Керівник практики від організації***

\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*(підпис) (прізвище, ім'я, по-батькові)*

***МП***

***Керівник практики від закладу освіти*** \_\_\_\_\_\_\_\_\_\_\_\_\_\_

***Рівне 2025***

# Зміст

Вступ

Календарно-тематичний план і щоденник практики

Основна частина

Висновки

Список використаних джерел

Додатки

# Вступ

Під час проходження виробничої практики я мав унікальну можливість зануритися у процес розробки сучасного веб-сервісу. Основна мета роботи полягала у створенні системи для порівняння характеристик побутової техніки, що базується на інтеграції з публічними API. З самого початку завдання вимагало не лише глибокого аналізу існуючих рішень на ринку, а й розробки власної архітектури, що забезпечить стабільну роботу програми при високих навантаженнях. Практична частина включала формування технічного завдання, створення UML-діаграм, розробку клієнтської та серверної частин з використанням React, Material-UI, Express, MongoDB, а також проведення комплексного тестування продукту. Такий підхід дозволив мені застосувати теоретичні знання на практиці та опанувати сучасні технології розробки програмного забезпечення.

# Календарно-тематичний план

# Щоденник практики

Загальна характеристика та аналіз існуючих сервісів порівняння характеристик товарів

Дата: 27.01.2025

Аналіз існуючих сервісів порівняння характеристик товарів.

1. Ознайомився з існуючими веб-сервісами для порівняння побутової техніки.
2. Провів аналіз функціональних можливостей конкурентів.
3. Вивчив переваги та недоліки різних рішень на ринку.

Відзначено:

1. Визначив основні функції, необхідні для нашого сервісу.
2. Сформував список вимог до функціональності веб-сервісу.

Дата: 28.01.2025

Вибір публічного API та обґрунтування вибору.

1. Дослідив доступні публічні API для отримання інформації про побутову техніку.
2. Порівняв такі API, як dummyjson.com, та інші.
3. Вибрав dummyjson.com API за його доступність.

Відзначено:

1. Обґрунтував вибір API для подальшої інтеграції.
2. Розпочав роботу з документацією обраного API.

Дата: 29.01.2025

Розробка архітектури веб-сервісу.

1. Створив UML діаграми для опису архітектури системи.
2. Визначив основні модулі: клієнтська частина, серверна частина, база даних.
3. Спроектував взаємодію між компонентами за допомогою діаграм діяльності.

Відзначено:

1. Завершив розробку архітектурних діаграм.
2. Підготував план реалізації модулів.

Дата: 30.01.2025 - 02.02.2025

Реалізація основних функцій веб-сервісу.

1. Налаштував середовище розробки на базі React та Material-UI.
2. Реалізував компоненти для відображення списку товарів.
3. Інтегрував публічний API для отримання даних про побутову техніку.
4. Реалізував функції пошуку та фільтрації товарів.

Відзначено:

1. Отримав основні дані з API та відобразив їх у вигляді карток товарів.
2. Реалізував базові функції пошуку та фільтрації.

Дата: 03.02.2025 - 05.02.2025

Розробка інтерфейсу користувача.

1. Створив макет інтерфейсу користувача з використанням Material-UI.
2. Реалізував компоненти фільтрації та сортування товарів.
3. Оптимізував інтерфейс для різних розмірів екрану.
4. Провів тестування зручності використання інтерфейсу.

Відзначено:

1. Інтерфейс користувача став інтуїтивно зрозумілим та зручним.
2. Забезпечено адаптивність для мобільних пристроїв.

Дата: 06.02.2025 - 10.02.2025

Тестування веб-сервісу.

1. Провів ручне тестування функціональності пошуку та фільтрації.
2. Написав автоматизовані модульні тести для основних компонентів.
3. Використав інструменти для перевірки продуктивності веб-сервісу.
4. Виявив та виправив кілька помилок у функціональності додавання товарів до порівняння.

Відзначено:

1. Підвищено надійність та стабільність веб-сервісу.
2. Забезпечено високу продуктивність при великих обсягах даних.

Дата: 11.02.2025 - 15.02.2025

Підготовка технічної документації.

1. Написав опис функціональних можливостей веб-сервісу.
2. Підготував інструкцію для користувача щодо використання сервісу.
3. Оформив UML діаграми та скріншоти інтерфейсу у документацію.
4. Перевірив документацію на відповідність вимогам та повноту інформації.

Відзначено:

1. Технічна документація готова та відповідає вимогам.
2. Документація включає всі необхідні розділи та матеріали.

# Основна частина звіту

У процесі роботи я спершу провів аналіз існуючих веб-сервісів, що надають можливості порівняння характеристик товарів, визначивши ключові функції та вимоги до майбутнього продукту. Після цього був здійснений вибір публічного API, де dummyjson.com став очевидним кандидатом завдяки своїй доступності та обширним даним про побутову техніку. Для опису архітектури системи я створив UML-діаграми, які ілюстрували взаємодію між клієнтською частиною, сервером та базою даних. Особливу увагу приділено проектуванню модульної структури, що дозволило ефективно організувати розробку та подальшу підтримку системи.

Розробка клієнтської частини була побудована на базі React із застосуванням бібліотеки Material-UI. Основні компоненти – ProductCard та CompareTable – відповідають за відображення інформації про товари, забезпечення функцій фільтрації, сортування, додавання товарів до списку порівняння, а також відображення найкращих значень за окремими характеристиками. Завдяки використанню таких технологій, як Axios для роботи з API та адаптивному дизайну, вдалося створити інтерфейс, який є як зручним, так і функціональним. Програмна логіка також передбачала збереження вибраних товарів у локальному сховищі, що сприяє покращенню користувацького досвіду.

Серверна частина реалізована з використанням Node.js та Express, що дозволило організувати RESTful API для роботи з даними про побутову техніку. Для зберігання інформації застосовано MongoDB з використанням бібліотеки Mongoose. Важливим етапом було створення функцій для отримання, створення, оновлення та видалення записів у базі даних, а також реалізація механізму заповнення бази демонстраційними даними (seed). У коді серверної частини використано принципи асинхронного програмування, що дозволило забезпечити високу продуктивність і стабільність роботи системи.

Крім того, особлива увага була приділена тестуванню веб-сервісу. Проведено як ручне тестування, так і написано автоматизовані модульні тести, що дозволило виявити та усунути можливі помилки на ранніх етапах розробки. Також було здійснено аналіз продуктивності, завдяки чому сервіс витримує великі обсяги даних без значних затримок у роботі.

Підготовка технічної документації завершувалася оформленням UML-діаграм, скріншотів інтерфейсу та детального опису реалізованих функцій. У звіті надано посилання на вихідний код як клієнтської, так і серверної частин, що підтверджує виконання поставлених завдань і демонструє повний життєвий цикл програмного забезпечення.

# Висновки

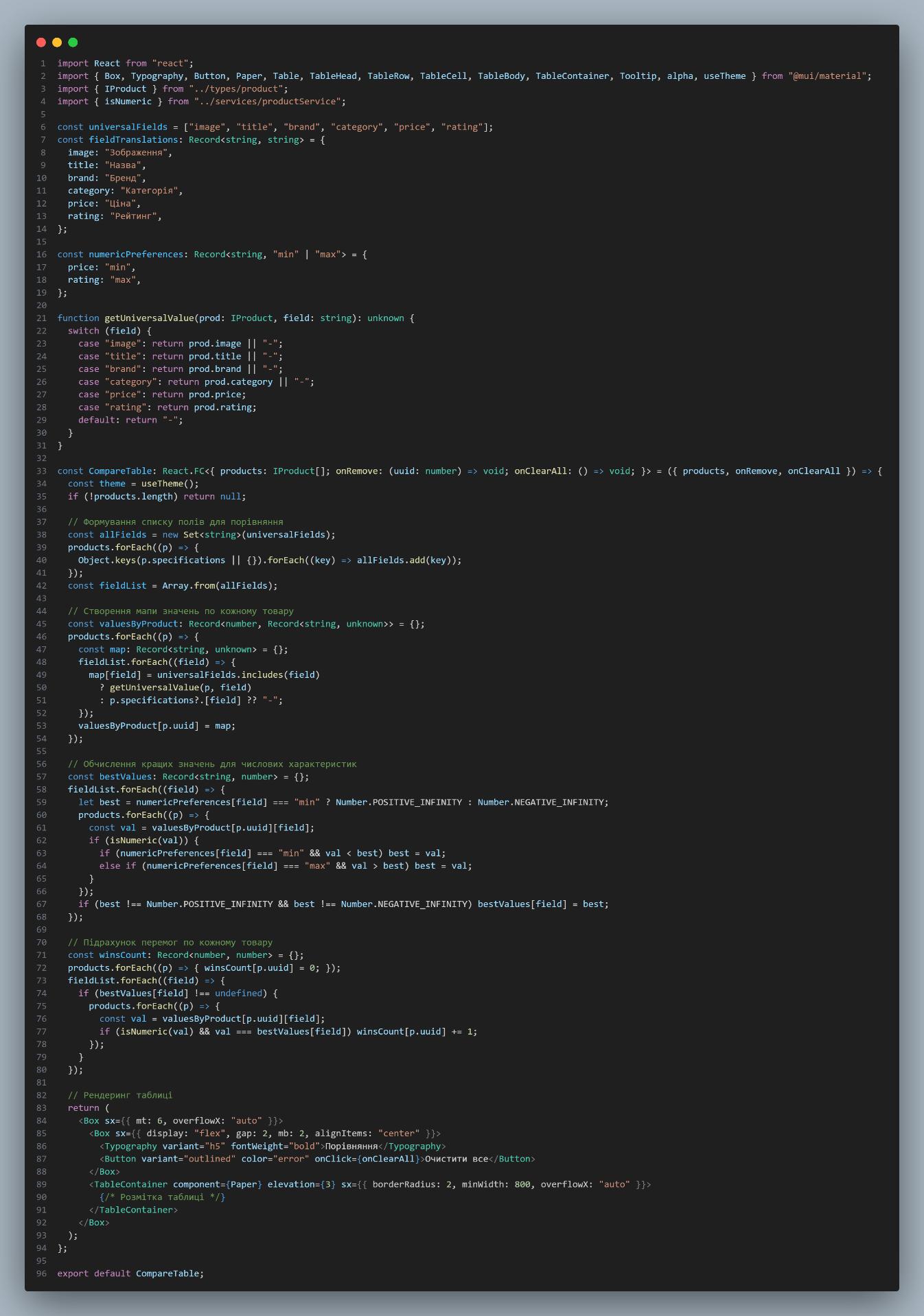
Підсумовуючи результати практики, я можу стверджувати, що отримані знання та практичний досвід значно розширили мої навички у сфері розробки веб-сервісів. Виконання поставленого завдання – створення сервісу порівняння характеристик побутової техніки – дозволило мені опанувати сучасні технології, такі як React, Material-UI, Node.js, Express та MongoDB, а також зрозуміти особливості інтеграції з публічними API. Ретельне тестування та підготовка технічної документації засвідчили важливість систематичного підходу до розробки програмного продукту.

На основі отриманого досвіду я пропоную подальший розвиток проекту шляхом розширення функціональності: додавання можливості залишення відгуків користувачів, інтеграції додаткових API для розширення бази даних товарів та оптимізації продуктивності для роботи з великими обсягами інформації. Отриманий практичний досвід безумовно стане основою для подальшої професійної діяльності у сфері розробки програмного забезпечення.

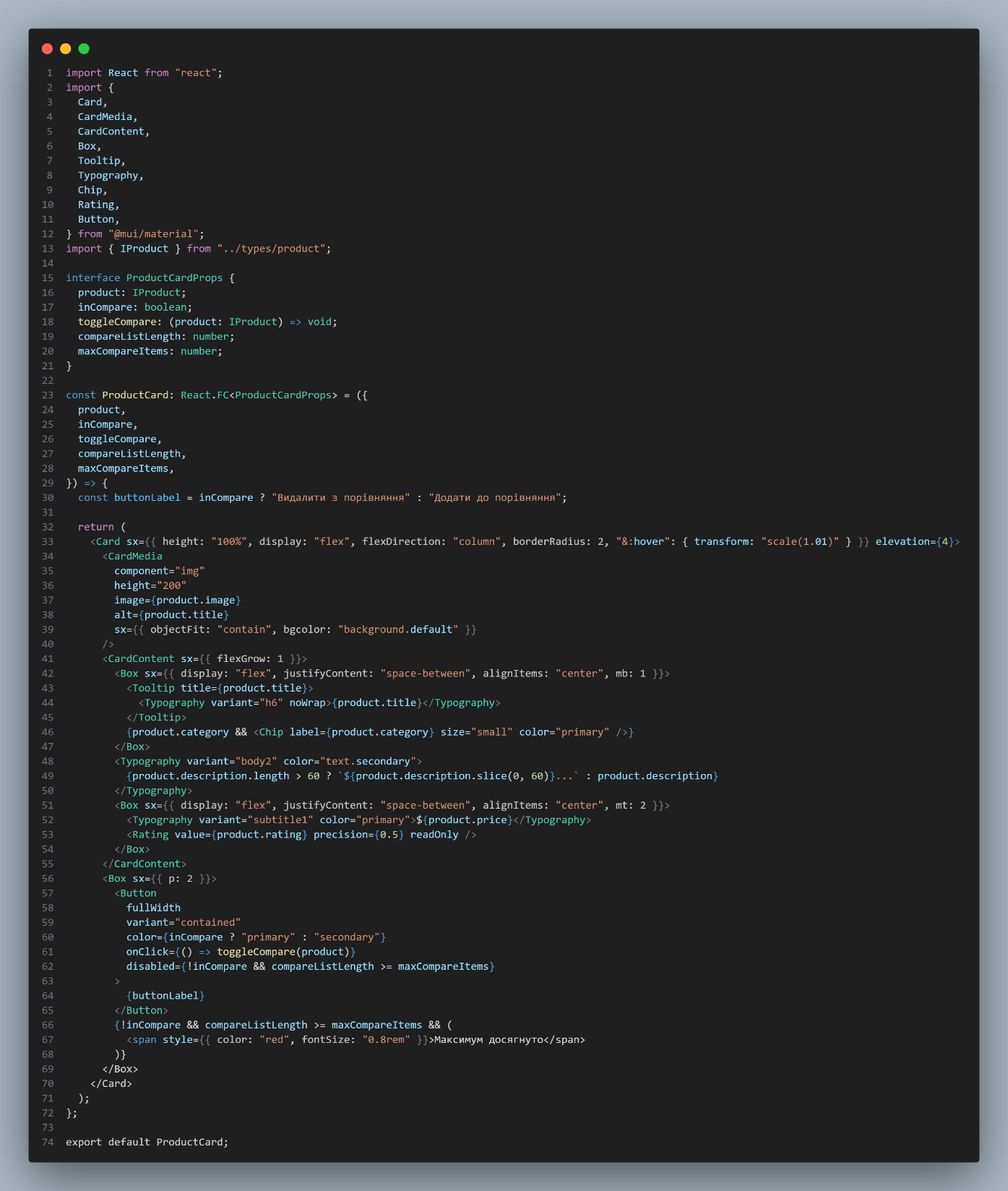
# Список використаних джерел

# Додатки

Додаток 1

 Цей компонент формується на основі списку товарів, які користувач додав для порівняння. Він визначає єдині властивості (як універсальні, так і специфічні) для всіх товарів, проводить порівняння числових характеристик (наприклад, ціна чи рейтинг) та виділяє кращі значення.

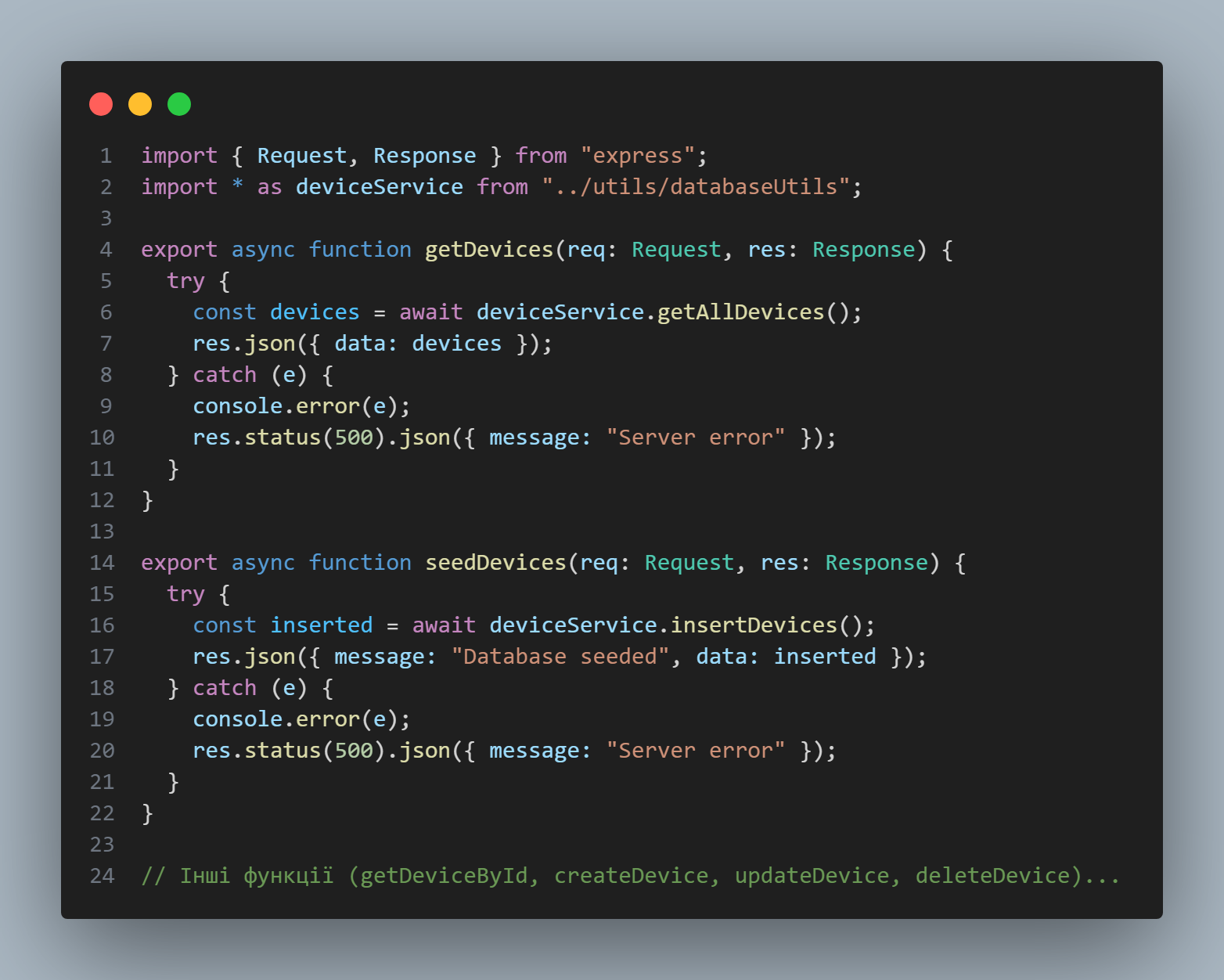
# Додаток 2

Компонент ProductCard відповідає за відображення інформації про окремий товар: зображення, назву, опис, категорію, ціну, рейтинг та кнопку для додавання або видалення товару зі списку порівняння.

# Додаток 3

Головний компонент клієнтської частини організовує роботу додатку, завантажує список товарів із API, організовує фільтрацію, а також управляє списком товарів для порівняння. Також реалізовано перемикання темного/світлого режиму.

# Додаток 3

Контролери обробляють запити клієнтів, забезпечуючи CRUD операції та seed бази даних. Роутинг організовано через Express, що дозволяє зв’язати URL запити із відповідними функціями.

# Додаток 4

Ця схема описує основні властивості товарів, додаткові поля, а також використовує віртуальні властивості та middleware для попередньої обробки даних.

Відгук