**Setup projektu**

Projekt jest zbudowany przy użyciu **React** i **Node.js z Express**.

**Frontend**

* React
* React DOM
* React Router DOM
* Axios

**Backend**

* Express.js
* SQLite
* JSON Web Token (JWT)
* Middleware autoryzacji użytkownika
* CORS
* Bcrypt

**Struktura**

Backend:

/server

/controllers

/authControllers #kontrola logowania

/productControllers #obsługa produktów

/reviewsControllers #pobieranie wszystkich ocen

/userControllers #obsługa użytkowników

/middleware

/middleware #obsługa adminów i autoryzacji

/routes # Metody współpracy z bazą danych(post,get,delete…)

/authRoutes

/orderRoutes

/productRoutes

/reviewsRoutes

/userRoutes

database.db # baza sqlite

db.js # inicjalizacja bazy

server.js # główny plik serwera obsługujący ścieżki

Frontend

/src

/App.tsx #główny komponent

/main.tsx #wywołuje główny komponent

/FakeApp.tsx #inna implementacja z react-router-dom

/components

/AdminReviews # usuwanie ocen przez admina

/Cart #koszyk

/ErrorBoundary # obsluga bledow

/History # historia zamowien

/Login # logowanie

/ProductDetails # opisy produktow

/ProductList # lista produktow

/Register # rejstracja

index.html # plik html obslugujacy strone

**Podział pracy**

**Backend: Maciej Wiśniewski**

**Frontend: Maciej Wiśniewski**

**Grafika: DeepSeek > Maciej Wiśniewski**

**Dokumentacja Postman:**

**POST** [**http://localhost:5000/api/users**](http://localhost:5000/api/users) **# rejstracja usera**

**POST** [**http://localhost:5000/api/auth/login**](http://localhost:5000/api/auth/login) **# logowanie admin/user**

**GET** [**http://localhost:5000/api/reviews/**](http://localhost:5000/api/reviews/) **# pobieranie wszystkich ocen**

**GET** [**http://localhost:5000/api/orders**](http://localhost:5000/api/orders) **# pobieranie wszystkich zamówień**

**POST** [**http://localhost:5000/api/orders**](http://localhost:5000/api/orders) **# dodawanie zamówienia**

**GET** [**http://localhost:5000/api/orders/5**](http://localhost:5000/api/orders/5) **# pobieranie zamówienia po id**

**GET** [**http://localhost:5000/api/products**](http://localhost:5000/api/products) **# pobieranie produktów**

**GET** [**http://localhost:5000/api/products/1**](http://localhost:5000/api/products/1) **# pobieranie produktów po id**

**POST** [**http://localhost:5000/api/products**](http://localhost:5000/api/products) **# dodawanie produktu**

**PUT** [**http://localhost:5000/api/products/6**](http://localhost:5000/api/products/6) **# edytowanie produktów po id**

**DELETE** [**http://localhost:5000/api/products/7**](http://localhost:5000/api/products/7) **# usuwanie produktów po id**

**GET** [**http://localhost:5000/api/products/1/reviews/**](http://localhost:5000/api/products/1/reviews/) **# pobieranie ocen po id**

**POST** [**http://localhost:5000/api/products/4/reviews**](http://localhost:5000/api/products/4/reviews) **# dodawanie ocen po id**

**DELETE** [**http://localhost:5000/api/products/reviews/22**](http://localhost:5000/api/products/reviews/22) **# usuwanie opinii po id**

**GET** [**http://localhost:5000/api/users/**](http://localhost:5000/api/users/) **# pobieranie userów**

**GET** [**http://localhost:5000/api/users/1**](http://localhost:5000/api/users/1) **# pobieranie userów po id**

**POST** [**http://localhost:5000/api/users/**](http://localhost:5000/api/users/) **# dodawanie userów**

**PUT** [**http://localhost:5000/api/users/20**](http://localhost:5000/api/users/20) **# edytowanie usersów**

**DELETE** [**http://localhost:5000/api/users/20**](http://localhost:5000/api/users/20) **# usuwanie userów**

**GET** [**http://localhost:5000/api/users/1/isAdmin**](http://localhost:5000/api/users/1/isAdmin) **# sprawdzanie czy uzytkownik jest adminem**