**Aplicație pentru găsirea cărților și crearea unui istoric cu cele favorite**



Nume: Nițescu Marisa

Grupa: 1118

Link video:

Link publicare: <https://pure-stream-96060.herokuapp.com/>

**Introducere**

Cloud Computing oferă foarte multe servicii și la curs și seminar am avut ocazia să aflăm mai multe despre acestea. În aplicația realizată, pe un model asemănător celei de la seminar, am avut ocazia să învățăm despre tehnologii precum React, Node, MySql, Google Cloud Platform dar și cum se face deploy-ul unei aplicații web, folosind Heroku.

**Descriere problemă**

Folosind Book API de la Google Cloud Platform se pot prelua cărți specificând titlul, autor sau domeniul reprezentativ. Din cărțile afișate, poți adăuga pe cele favorite, putând fi vizualizate în contul tău. De asemenea, acestea pot fi șterse.

**Descriere API**

**Books API** este un serviciu de la Google Cloud care permite găsirea cărților în funcție de query-ul dat.

În aplicația mea, am făcut un request prin metoda HTTP Get către acest endpoint găsit în documentația API-ului:

<https://www.googleapis.com/books/v1/volumes?q=search+terms>

În componența acestui endpoint a fost necesar să adaug API key-ul generat din platforma Google Cloud Services. Ca răspuns la cererea făcută primim un array de obiecte, fiecare obiect dintrea acestea conținând date despre cartea respectivă ca de exemplu: titlu, autor, data publicării, review, descriere, poza cărții și link către site-ul pe care poate fi găsită cartea.

**Flux de date**

În urma răspunsului dat de Books API, afișez cărțile în ecranul principal, fiecare având câte un buton de Add. La apăsarea butonului Add se generează un request de tip POST către backend, de unde cartea este adăugată în baza de date, în tabela Books, cu câmpurile aferente.

Ca răspuns, în caz de succes primim un mesaj conform căruia cartea a fost adăugată, iar în caz de eșec, primim un mesaj, acela că nu au fost specificați toți parametrii.

În Contul Meu prin intermediul unei cereri de tip GET, solicităm backendului să ne preia din baza de date cărțile adăugate. Afișăm cărțile preluate împreună cu un buton de Delete pentru fiecare. La apăsarea acestuia, fac o cerere de tip DELETE prin care șterg din baza de date cartea cu id-ul respectiv.

În frontend design-ul este făcut folosind Tailwind CSS. De asemenea, am folosit react hooks pentru definirea funcțiilor. Există două pagini, pagina principală în care căutăm cărțile dorite și le afișăm pe ecran(în număr de 20) și pagina Contul Meu în care sunt afișate cărțile adăugate.

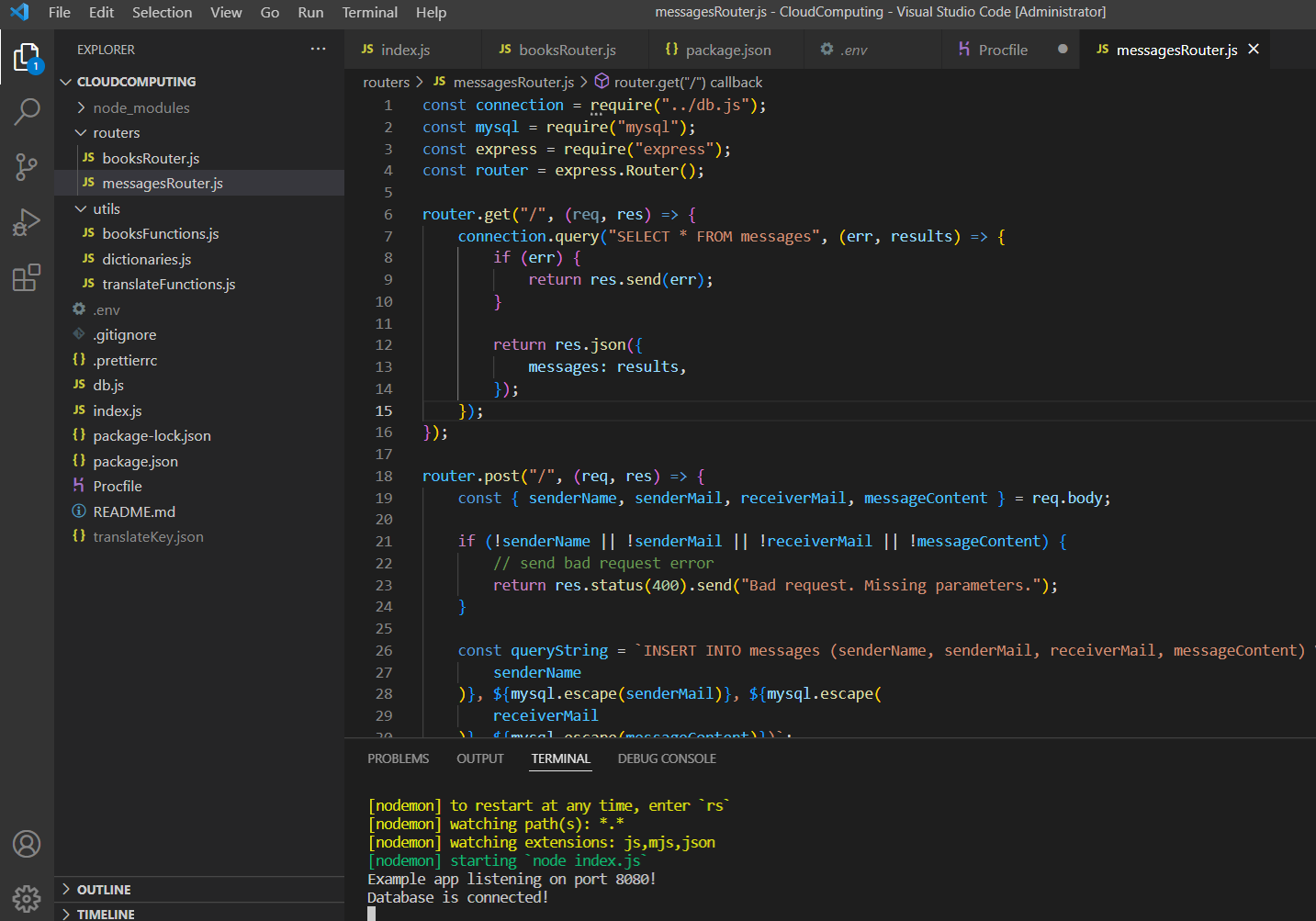
În pagina principală avem următoarele funcții: **searchBooks** prin care face request-ul către Books API; **addBooks** care se declanșează la apăsarea butonului Add prin care se adaugă cartea în baza de date.

În pagina Contul Meu folosesc hook-ul useEffect pentru preluarea cărților din baza de date și funcția deleteBook care șterge cartea selectată din baza de date.

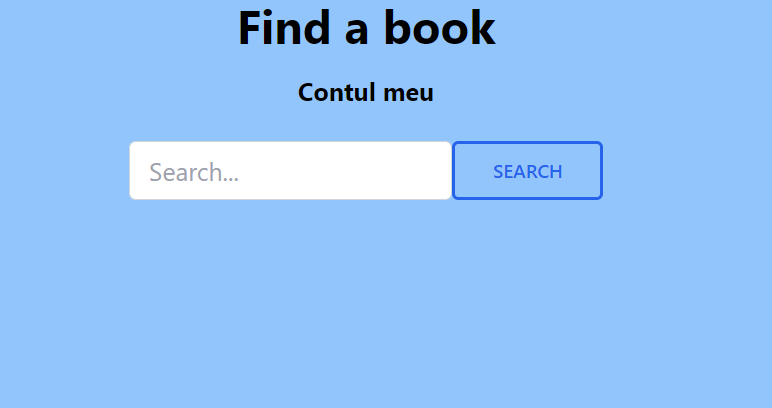
De asemenea, aplicația este Responsive, datele afișate putând fi vizualizate și de pe telefon.

**Capturi de ecran**

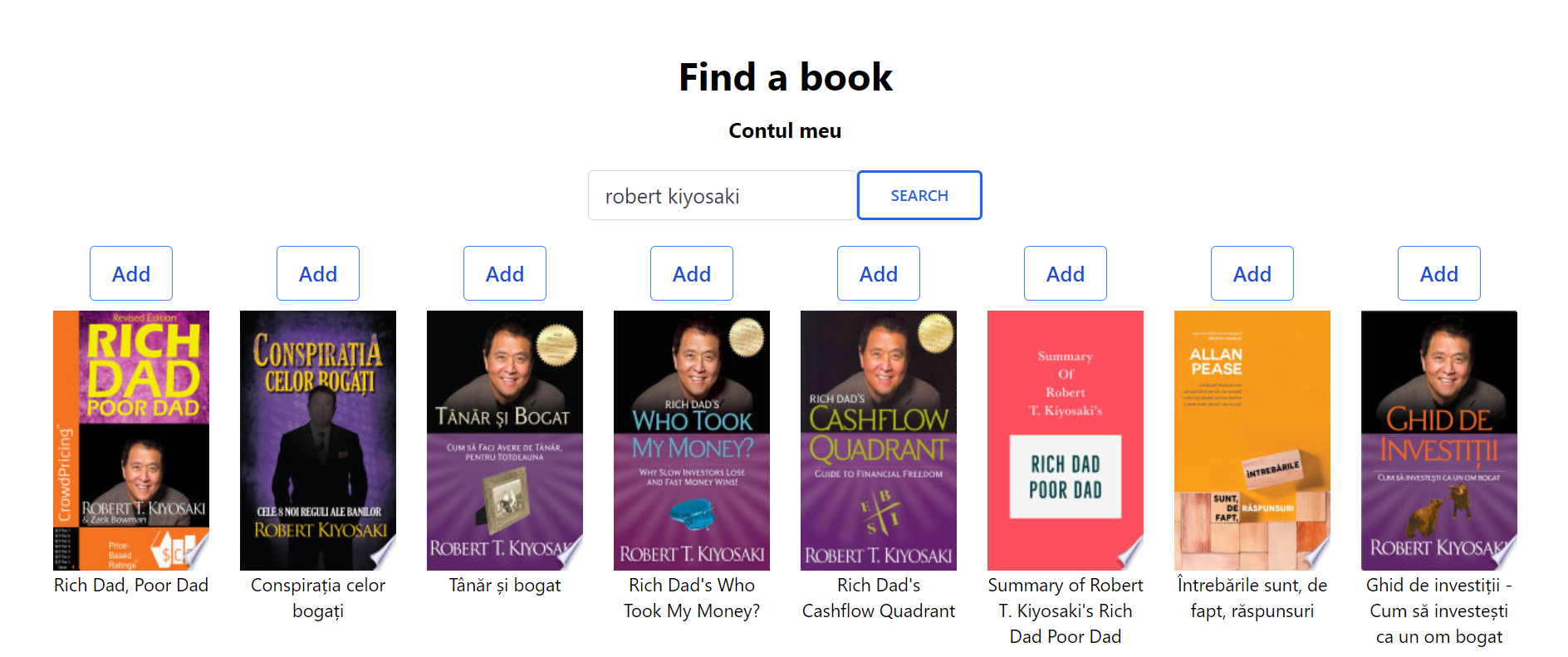
**Pornire Backend și conectare la Baza de date**

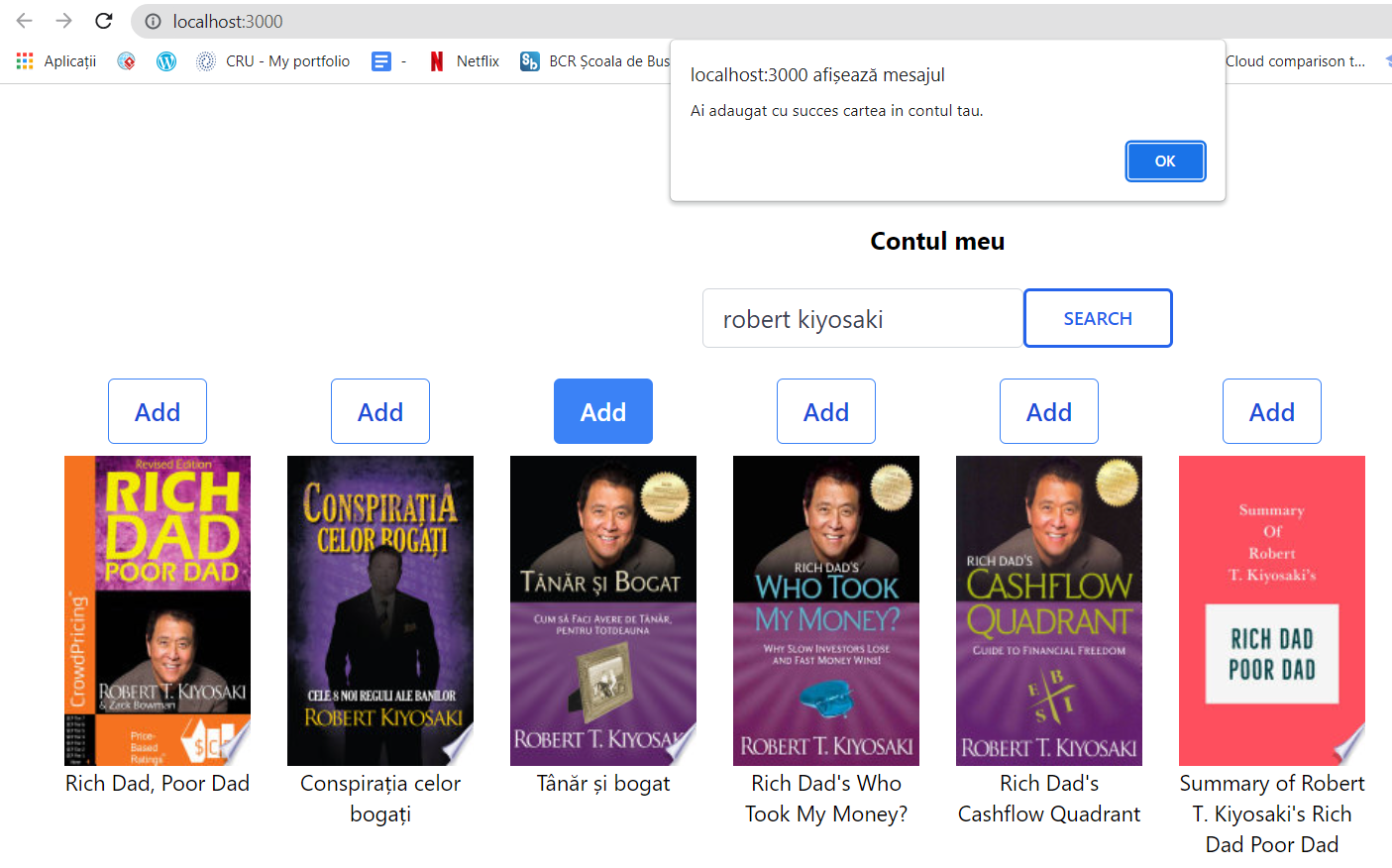
****

**Main Page:**

****

**Afișare cărți după query-ul făcut:**

****

****

**Afișare cărți adăugate în Contul Meu**

****

**Pentru deploy am folosit Heroku.**

**Referințe:**

Books API: <https://developers.google.com/books>

React Js: <https://reactjs.org/docs/getting-started.html>

Express with Node Js: <https://expressjs.com/en/guide/routing.html>