Job Simulation: Book Explorer Application

Project Title: Implementation of a Book Explorer

Role: Front-End Developer Intern – Dynamic Search & API Rendering

Technology Stack: React, Material UI, useState, useEffect, useRef, Fetch API

Objective

To design and implement a dynamic book search application that allows users to search, view, and load book results using data fetched from an external API. The project introduces interns to live API integration, infinite scrolling, and input throttling using React hooks and Material UI components.

Task Overview

As part of your hands-on learning, you are required to build a book explorer interface that allows users to:

- Search for books using a keyword
- Display matching books in a card-based grid layout
- Load more books as the user scrolls (infinite scroll)
- View essential book details like title, author, and cover thumbnail

Task Requirements

1. **Functionality:**

- Display a search input using <TextField>
- Use useState to manage the search term and book list
- Use useEffect to fetch books from https://openlibrary.org/search.json?q=
- Use useRef to debounce input and prevent frequent API calls
- Append new results on scroll for infinite loading

2. **User Interface (UI):**

- Create a clean, responsive layout using Material UI
- Display results as cards in a <Grid>
- Use components like <Card>, <Typography>, and <CircularProgress>
- Show loading indicators while fetching data
- Apply hover effects and spacing for improved user experience

3. **Code Structure:**

- App.jsx – Main layout, state, and API logic

- BookCard.jsx – Component for individual book display
- api.js – (Optional) utility file for fetching data

Note: Add inline comments to explain:

- How the input debounce logic works with useRef
- How pagination and infinite scroll are implemented

Bonus (Optional):

- Add filters like “Author” or “Publish Year”
- Open a modal with more details when clicking a book
- Use IntersectionObserver for scroll detection

Deliverables

Submit a project folder that includes:

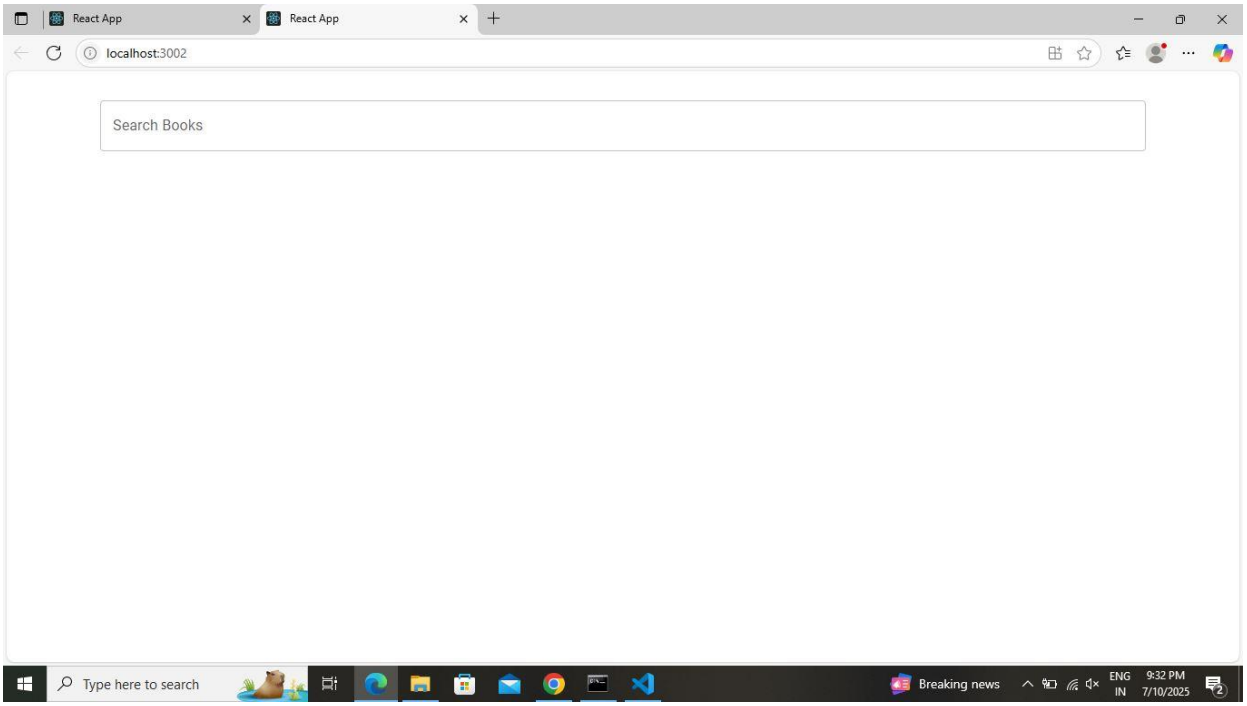
- App.jsx – Layout and search logic
- BookCard.jsx – Displays individual book info
- api.js – API interaction logic

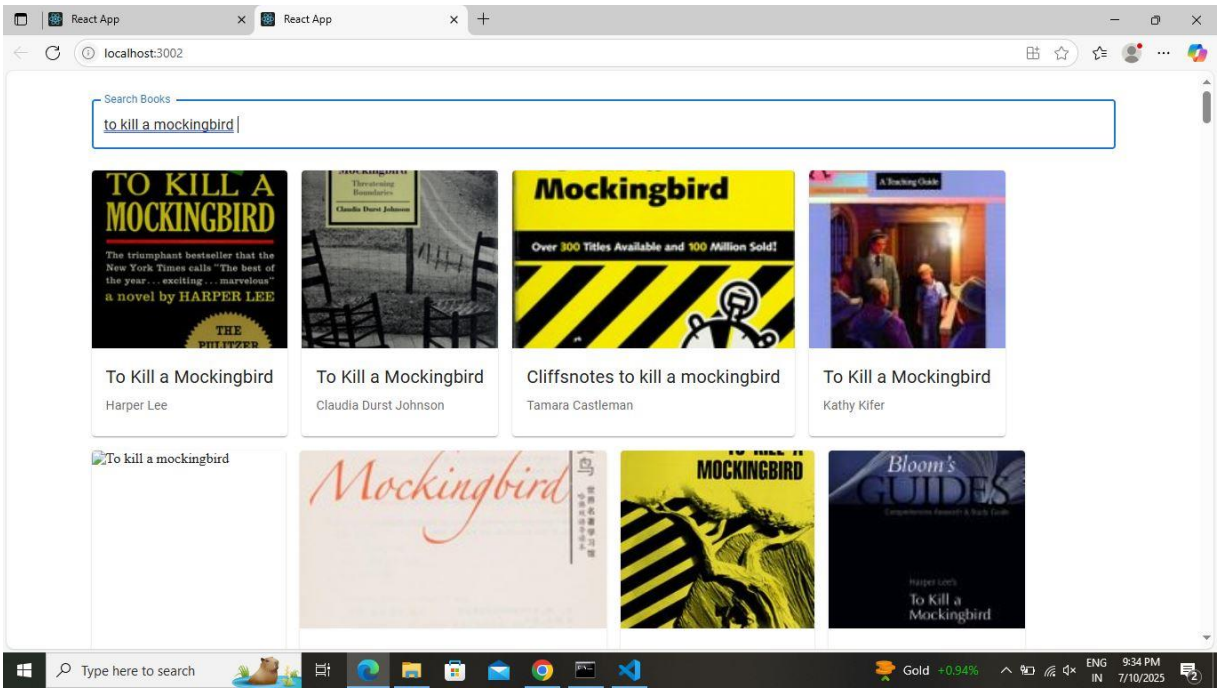
Note: Use only Material UI components. Do not use raw HTML or custom CSS.

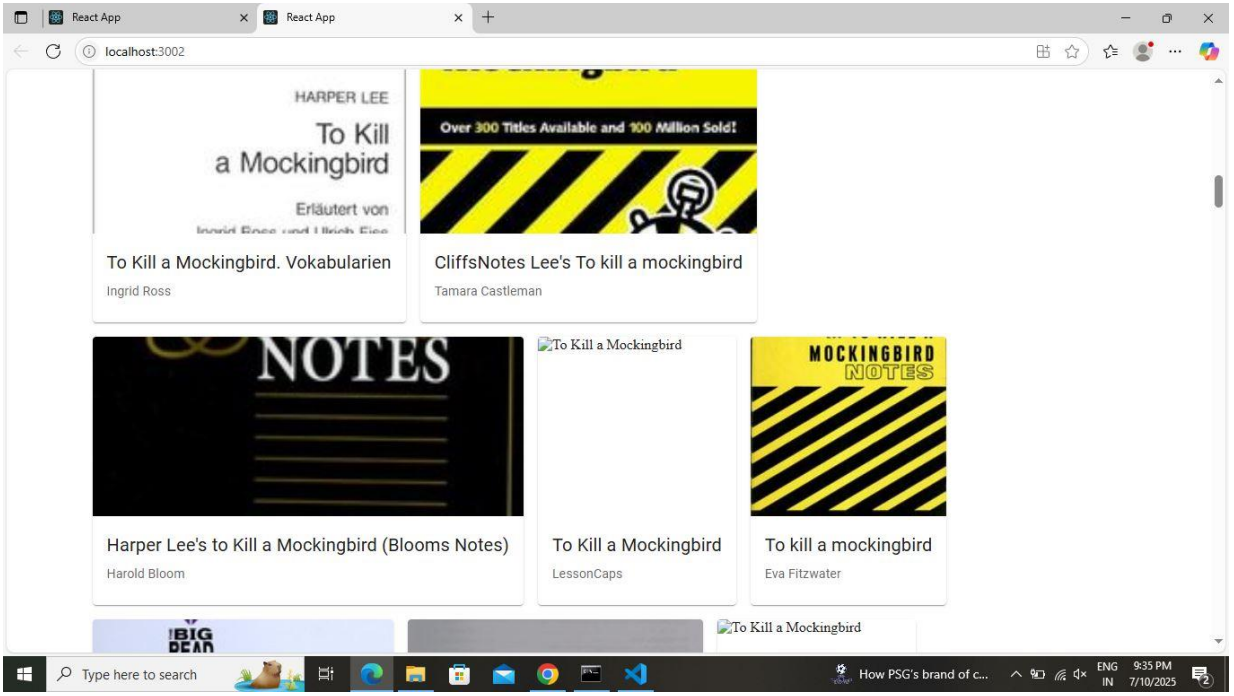
Learning Outcomes

By completing this task, you will gain practical experience in:

- Integrating external APIs using Fetch
- Managing user input and component re-renders with hooks
- Creating modular and reusable UI components
- Implementing infinite scroll in a React application
- Using Material UI to build responsive and modern layouts.




React App
React App
localhost:3002
Search Books
to kill a mockingbird
To Kill a Mockingbird
Harper Lee
To Kill a Mockingbird
Claudia Durst Johnson
Cliffsnotes to kill a mockingbird
Tamara Castleman
To Kill a Mockingbird
Kathy Kifer
To kill a mockingbird
Type here to search
Gold +0.94%
ENG
IN
9:34 PM
7/10/2025


React App
React App
localhost:3002
To Kill a Mockingbird. Vokabularien
Ingrid Ross
CliffsNotes Lee's To kill a mockingbird
Tamara Castleman
Harper Lee's to Kill a Mockingbird (Blooms Notes)
Harold Bloom
To Kill a Mockingbird
LessonCaps
To kill a mockingbird
Eva Fitzwater
To Kill a Mockingbird
Type here to search
How PSG's brand of c...
ENG
IN
9:35 PM
7/10/2025