Status Report

Summary
Stacks is a web-based library discovery platform that allows users to search for books across multiple library systems and discover what's available in their local libraries.

Current Status: Development in Progress - Core Features Implemented

Work Completed So Far:
1. Core Application Infrastructure
- Set up React 18 + Vite development environment
- Implemented react-router-dom for client-side routing
- Created navigation system with responsive design
- Established component directory structure
- Configured build and deployment pipeline
2. Library System Integration
- Scraped Houston Public Library catalog
- Scraped LA County Library catalog
- Implemented category-based navigation UI
- Created library detail pages with item listings
- Set up automated monitoring for LA County updates
3. Discover Feature Implementation
- Ported Discover page from Next.js TypeScript to React JavaScript
- Integrated Open Library API adapter for book search
- Implemented California library registry
- Added geolocation-based library finder
- Created Nominatim geocoding integration
- Converted all TypeScript code to JavaScript
- Fixed ES module imports and file extensions
- Implemented distance calculation using Haversine formula
4. Web Scraping Development
- Initial BeautifulSoup scrapers (failed - JS-rendered content)
- Pivoted to Selenium-based scraping approach
- Created working scrapers for 3 California libraries:
- Orange County Public Libraries
- Irvine Public Libraries
- Rancho Cucamonga Public Library
- Encountered pagination and stability issues (Chrome crashes)
Current Goals
- Testing and debugging Discover page functionality

- Validating book search with Open Library API
- Testing geolocation features across browsers
- Verifying library data loads correctly
- Complete user profile functionality
- Implement wishlist/favorites feature
- Add search result filtering and sorting
- Create comparison view for multiple libraries

Identified Problems:

BeautifulSoup scrapers failed because library catalog websites use JavaScript framework that render content client-side. No data was collected from libraries so I used Selenium WebDriver with headless Chrome and was able to scrape a few libraries