

NextPick Capstone Documentation

By: Shelby Larrucea, Justin Kurz, Alex Cardoso, Imran Imran, Sebastian Perez Fematt

Quick Links: [Poster](#) - [PowerPoint](#) - [Intro Video](#) - [Demo Video](#) - [NextPick](#)

Summary

Our capstone project is an AI-driven website designed to change the way readers discover books. A user inputs their favorite or last read book, and our site takes care of the rest. The site uses a semantic search over a vectorized database of books, which is powered by OpenAI embeddings and Pinecone, to then return a ranked list of recommendations to the user. Our app also includes two features to alter results, excluding certain books by title/ISBN and an exclude series choice. Upon user input, our site displays recommendation cards with book cover, genre, ISBN, and an AI-optimized description of each result. We aim to prove how modern retrieval and AI can be combined to bring readers their next best pick.

Problem & Requirements

Avid readers reach a point where they aren't sure what to read next. Not because of a lack of books but due to the difficulty of selecting another book that aligns with their personal preferences. Other traditional book sites recommend based on keywords, genres, and authors. These recommendations often do not align as they fail to capture the deeper "vibes" in books. Our project addresses this gap in recommendations by creating a system that focuses on conceptual similarity rather than matching authors and genres.

Features

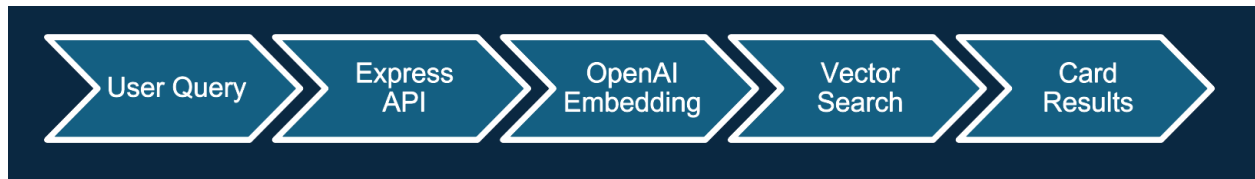
Functional:

- Natural-language / title search
- Personalized recommendations
- Exclusion Filtering
- Dynamic recommendation display with book metadata

Non-Functional:

- Responsive Web UI
- Secure environment configuration
- Rate limiting + CORS Control

System Design & Implementation



The frontend was built with HTML, CSS, and JavaScript. The backend runs on Node.js and Express, connecting the UI to the text model and Pinecone via the API. Python was also used for dataset cleaning, vectorizing, and uploading. Pinecone is what stores our vectorized database, while we use the OpenAI text embedding model. Vercel takes care of the hosting and deployment. The text embedding model is also what creates the short and simplified descriptions for each book result shown.

Security & Privacy

For security/privacy reasons, we have implemented rate limiting to prevent API abuse. The site also has CORS control. At NextPick, we do not collect any data from users and have also opted out of including auth/user accounts to reduce concerns and maintain a more load-and-go approach. No user preferences or input history is stored, keeping inputs anonymized and unreachable.

Deployment and Operations

Our system is fully automated through GitHub and Vercel integration. Any push to the main branch will cause Vercel to trigger a build and deploy to production. As for our vectorized database, the process is manually handled using Python, and all books currently stored have been personally cleaned, vectorized, and uploaded to Pinecone.

Limitations & Future Work

We currently face a limited number of books due to monetary limitations regarding database pricing. Due to these constraints, we only have about 180,000 books stored. Although this can be seen as a large number, when it comes to book databases this is a needle in a stack of hay. In the future we would love to increase our resources in order to hold a competitively sized database. We would also like to include user accounts in order to implement wishlist, favoriting, and voting features on site. Other future work includes multilingual support and implementation of a Goodreads API for user library syncing.