



Introduction

Snapseeker is a web application where users can scan a product, and the app will display all the results, comparing all the prices that are available in the market. Users can click a link that would redirect them to that product listing.

Features

- Users are able to search real products data all around the web
- Search with auto text suggestions
- Search products by taking an image
- Users are able to click the product and get redirected to the product posting
- Compare real time product prices from different websites
- Create an account with passwords
- Users are able to reset passwords via one time link sent through their email
- Users are able to change account information within their dashboard
- Responsive web app that allows users to use the app via mobile or desktop

Security

- Passwords are hashed in the database
- API calls with sensitive keys are only ran in the server side
- Users are always verified when making API calls to the database

Login

Let's get you saving!

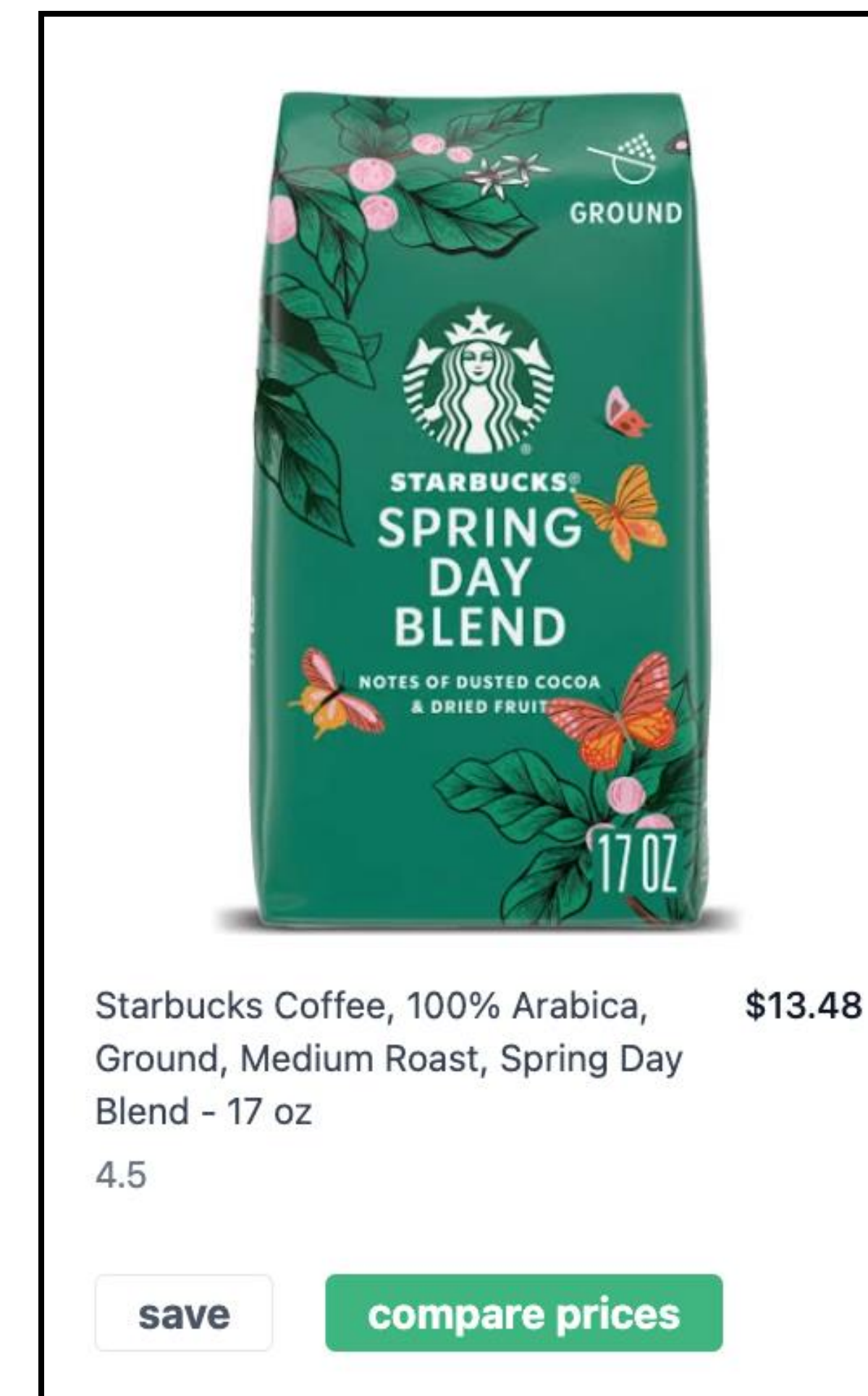
Email Address

Password

[Forgot your password?](#)

Already have an account? [Sign up.](#)

Product Card



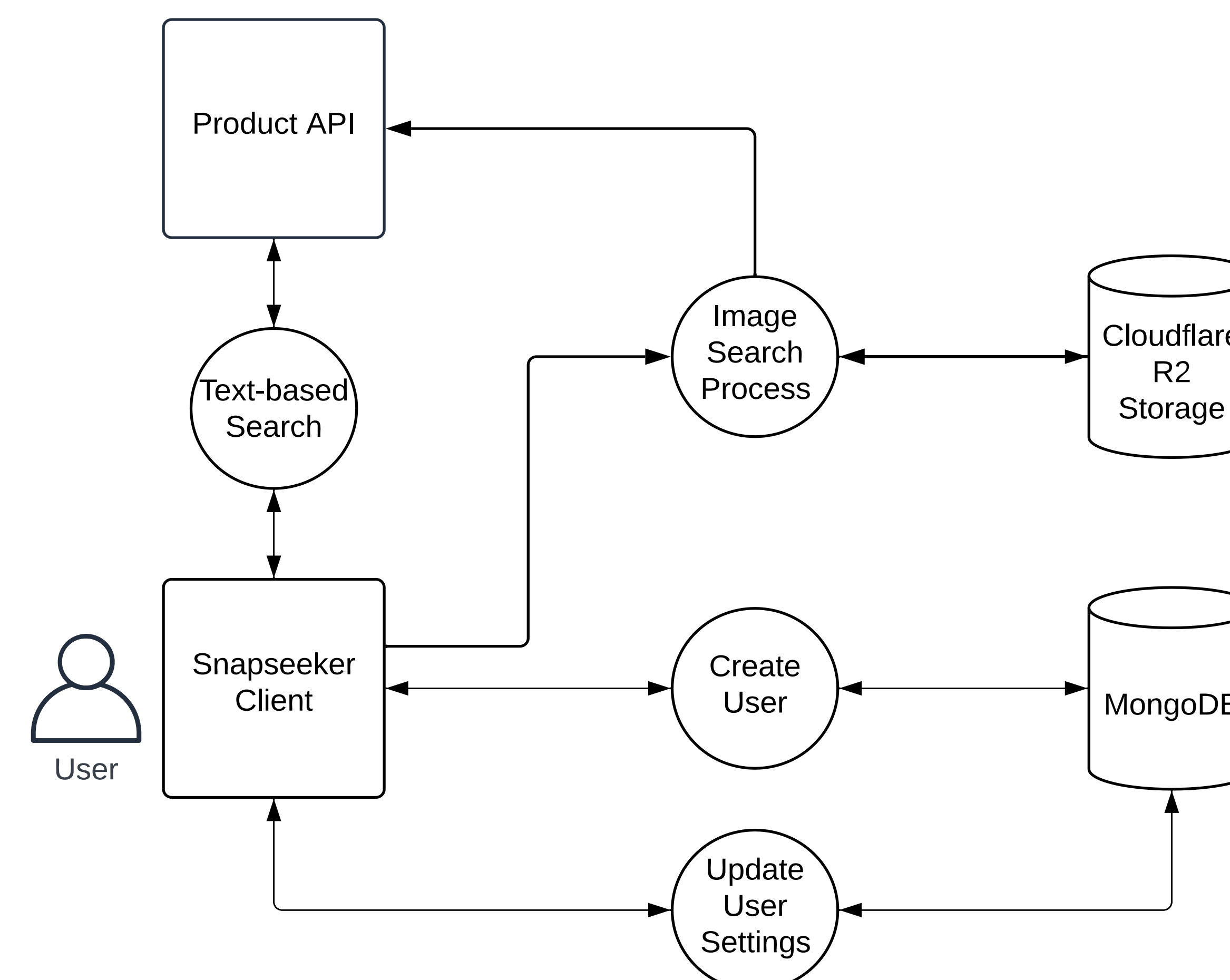
Comparisons

Product Comparisons

Walmart	\$17.39
Target	\$10.88
CVS Pharmacy	\$19.90
Vitacost.com	\$21.75
Kroger	\$15.21
Fred Meyer	\$15.21

Search Bar

- starbucks cups blue
- starbucks cups christmas version
- starbucks cups collections
- starbucks cups amazon



Technologies

Snapseeker was mainly built with Next.js, a React framework that supports server-side rendering and has more SEO and performance benefits over using plain React. The team also used MongoDB as its primary database and used cloud services such as Cloudflare to store images and external APIs.

Project Timeline

Fall

- Conceptual design for the project
- Research on what framework and database to use
- Establish implementation timeline for Spring

Spring

- Requirement gathering, check what tools to use and research about needed APIs
- Build the base architecture of the web application
- Database integration and front-end implementation
- Testing and deployment

Challenges

- How to get started with Next.js and connecting it with the database
- How to fetch real product without paying for a subscription service for data
- How to implement Image search and integrating it with text-based search

Project Links

Preview: <https://snapseeker.vercel.app>

GitHub: <https://github.com/Tagaca19e/snapseeker>