



Knight Foundation School of Computing and Information Sciences

Summer Semester 2024 Senior Design Project

Material Atlas

Students: Daniel Brizuela , Laura Penza , Thuong Nguyen , Hansel Soberao

Mentor: Lyla Wu, FIU

Instructor/Faculty: *Dr.Masoud Sadjadi*, Florida International University

Abstract

Our team aims to improve the user experience on Material Atlas and deliver on site promises. The platform exists to serve as a knowledge-sharing platform between engineers/architects. Previous efforts on the project built the foundations for application. During this cycle, our team made progress in delivering on general use cases. These changes prepare Material Atlas to be shared with its intended audience, and delivers features essential to the crowd-sourced nature of the webpage.

Technologies used:

- Next.js
- React
- Neo4j
- Visual Studio Code
- Vercel

Standout features:

- User Authentication
- Email Verification
- File Upload and Storage
- Privileged User actions
- Data Visualization

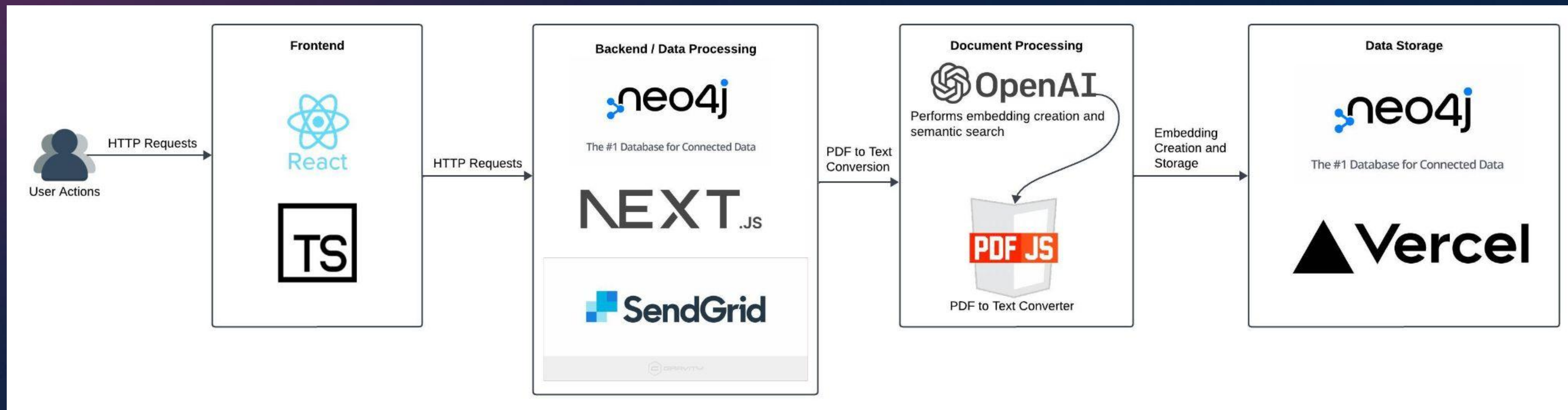
Challenges & Problem

- User Experience - The website lacked essential features such as functional login, password recovery and email validation, which are critical features that a website should have for user registration and login.
- Hosting - Initially, there were hosting issues due to a switch from Vercel to SiteGround. While SiteGround is excellent for WordPress, it was incompatible with the project's foundation, which included Python and TypeScript. Additionally it did not support the database system used for this project which is Neo4j, a graphical database, since it only supports PHP databases.
- Technology compatibility - TypeScript does not support the full functionality of OpenAI API request. There were difficulties making the requests for OpenAI from the .tsx file to upload the pdf and a work around was needed and implemented.

User Stories & Implementations

- Provide a robust and well refined platform for material scientists and engineers new to the field so they can share and improve their knowledge and skill
- Assist new researchers that have no access to big databases for material construction to start practicing and building up their skills
- Include website features like “Forgotten Password” , “Email Validation” and “Email Verification” to properly secure the website from bots or other malicious internet threats and improving the average user experience
- Add features like User and Admin Dashboards needed for a website to correctly handle users and their permission, while also tracking their contribution to the website
- Produce a clean and easy to navigate User Interface that allow new users to experience no trouble getting used to the website and giving them no issue when it comes to accessibility, being able to access the website from any of their device

Material Atlas System Design



Front-end:

- React: Handles user interface and interactions
- TypeScript: Ensure type safety and code reliability

Back-end/Data Processing:

- Next.js: Server-side rendering and API routes
- Neo4j: Stores and manages connected data
- SendGrid: Handles email services

Document Processing:

- PDF.js: Converts PDF to Text
- OpenAI: Performs text embedding creation and semantic search

Data Storage:

- Neo4j: Stores processes data for efficient querying and retrieval
- Vercel: Hosts the application, ensuring scalability and high performance

Thuong's Contributions

File:

- Implemented file upload functionality using react-dropzone and FormData
- Managed file uploads to Vercel's server, handling various document types (PDF, Word)
- Built functionality to delete uploaded files

OpenAI API Integration:

- Developed a module to interact with the OpenAI API using fetch, handling both successful responses and errors
- Extracted and processed text from PDF files using PDF.js, converting text to prompts for the API

Hansel's Contributions

Mobile Friendly Webpage layout:

- Updates to web page layout so Mobile experience is same across platforms.

Privileged User Sessions:

- Implemented user session management and an admin check to restrict access to sensitive pages in our web application. Ensuring that only users with administrative privileges can access certain functionalities

User Administrator Dashboard:

- Designed and Developed an Admin Dashboard for privileged actions
- User Directory with contact information of registered users
- Functionality to grant Admin permissions to other users

Daniel's Contributions

Website Hosting and Maintenance

- Transferring back from SiteGround to Vercel for hosting and maintenance of the website, and creating a Github for the team to work on.

Solving Connectivity Issues

- Properly connecting the Upload page to the Neo4j Cloud database.
- Fixing connectivity issues that caused the database to not display properly in the Database page.

Website Additions and Improvements

- Creating a Material Input page that focuses on manually inputting mixes into the database to facilitate the growth of the database for future updates.
- Improve existing queries robustness and security for uploading data in the Upload page by adding new unique values to make sure there are no duplicates relationships that may cause the database to crash or display wrong results in the display.

Laura's Contributions

Registration and Sign In

- Created a robust registration system allowing new users to create accounts and implemented input validation to ensure data integrity and prevent invalid data submissions
- Used Neo4j to store user information securely, ensuring compliance with data protection
- Added functionality to send verification email through Twilio SendGrid to new users' requiring email confirmation before activating their accounts
- Integrated a system to send password reset links through Twilio SendGrid to users' registered email addresses
- Developed a secure sign-in process using standard encryption methods to ensure data protection
- Designed and developed a responsive and user-friendly sign-in and forgot my password interface

User Dashboard

- Designed and developed a user dashboard to display relevant user information and metric
- Interactive elements and features to the dashboard, improving user engagement and functionality
- Fetch and display user-specific data from the Neo4j database in real-time

Summary

User Experience

- User-friendly design for easy navigation and interaction
- Real-time data display and interactive elements

Hosting

- *Scalable Environment*: Robust hosting ensuring high availability and performance
- *Optimized Handling*: Efficient management of multiple user requests

Technology stack

- *Front-end*: React with TypeScript for a responsive UI
- *Back-end*:
 - Neo4j for secure data management
 - Vercel for storage of files and hosting
 - Next.js for server-side rendering
 - Next-auth for authentication
- *Additional*:
 - Twilio SendGrid for email services
 - Blob for connecting to Vercel's serverless function
 - OpenAI for interpreting files
 - PDF.js for converting pdfs to texts