Anthony Konas
CSPB 3112

My API Journey

Introduction

API's are an integral part of today's software communications. They assist companies with data collection to make educated decisions, they also are utilized to help companies communicate changes. For example an API can be used to change the wording you see on a bus route, construction site, or freeway signs. The company I work for allows an open API on our environmental health and safety software where groups can build out their scripts to pull data on forms that have been filled out to make informed decisions to improve their safety programs. These same API's at my company are also used to pull data on employee OSHA safety training completion to show who's up to date and who needs to be pushed to complete their training assignments.

The vision for my project changed about halfway through. In the beginning I thought I was going to learn everything there is to know about APIs. Given my bandwidth and motivation, this just wasn't the case unfortunately. I felt like I was pushing too hard and ultimately was regressing because I was so overwhelmed by the information that is out there. Instead I needed to change directions and walk a path that was less stressful and had better objectives. The LinkedIn Learning project portion of this class came at a great time. I discovered a great class on APIs on this platform and since about week 9, it became a great pivot to introduce myself into APIs.

Background

While my project goal was originally to develop my first API, instead it became what is in the title, "My API Journey". The goal was simple: put myself out there and learn as much as I can. Where the project idea came from was actually my frustrations of dealing with API questions by my clients when we aren't trained on how to write API scripts as Client Success Managers. It just isn't a part of our job (currently). But I felt that it would be a better time to take advantage of this lack of knowledge, because at the end of the day I want to make myself in my role as valuable as possible. This project opportunity also opened my eyes to different career paths that I've taken an interest in over the course of the semester like Product Managers and Solutions Engineers. The value this brings not only improves my ability to connect with my current clients, but also helps me expand my knowledge of APIs as I look to transition into other fields that require this type of background knowledge.

Methodology, Materials and Methods

In the beginning I was using FreeCodeCamp to learn how to write API's, but it was hard to stay focused. I switched over to the LinkedIn Learning module on API's and used that as my resource moving forward. Once I got the basic command codes written and a functioning API that could run on my local server, I then expanded into my company's open API documentation to work on scripting code that would allow me to do both a GET and POST request from my demo site into MongoDB. Other resources I used were Studio 3T Community Edition to see that my SQL DB was saving the information I was requesting and Postman to make my API requests.

My development environment included Node.js with Express framework for building the API server, MongoDB for data storage, and various npm packages including axios for making HTTP requests, mongoose for database modeling, and babel-node for JavaScript transpilation. I used Visual Studio Code as my primary development environment and tested all API endpoints using Postman.

Results

There was a good amount I learned from this project. Aside from writing the code, which LinkedIn Learning really did a great job explaining, I also gained an understanding of the types of packages needed to run these JSON requests, and how to design my Controller, Model, Routes, and Services. I'm sure everyone handles these documentations differently, but my crmController file was used to house all of my code requests, crmModel was used to POST the type of information I was looking to upload into my DB, crmRoutes was utilized to pull the code from my Controller and run those commands in Postman, then my kpaServices was used to handle all communications with KPA's API and handle those requests within one file.

I successfully built a functioning API integration that can pull user data from KPA Flex's API and store it in my local MongoDB database. The system includes proper error handling, data transformation between different schemas, and duplicate prevention based on employee numbers. I can demonstrate my learning through the working code that successfully retrieves real user data from my company's production API and stores it locally.

I had three learning objectives:

API Fundamentals (~15 hours): Complete online tutorials and courses using resources like FreeCodeCamp, Postman Learning Center, and Power BI Playground to understand REST API principles, HTTP methods (GET, POST, PUT, DELETE), and JSON data structures.

Hands-on Practice (~20 hours): Create and test API calls using Postman or similar tools to interact with publicly available APIs. Document at least 10 different API interactions, including authentication, data retrieval, and error handling scenarios.

Documentation and Portfolio Development (~10 hours): Maintain weekly blog posts on my GitHub Pages site documenting learning progress, challenges encountered, and solutions discovered. Create a comprehensive resource that could help other beginners in similar situations.

While I did pivot from FreeCodeCamp and Power BI to LinkedIn Learning and Postman, the course still remained on task. I still managed to work in JSON formatting, I got plenty of hands-on practice of making requests and ensuring that my code was functioning properly, and lastly I felt I did a decent job staying on top of my documentation even though there were some delays here and there.

Discussion / Reflection

Overall I felt like I did what my title's intent states, I went out on a journey to better understand APIs. While I may not be a master of them, I'm really pleased with what I started and the direction that it allows me to go moving forward. I look at this project as a kick-starter that will allow me to build on my API within my company's open API and work into Power BI projects so I can become even more of an asset to both the company and clients.

I successfully met my learning objectives, though the path changed from my original plan. What allowed me to reach these goals was being flexible with my approach and switching to LinkedIn Learning when FreeCodeCamp wasn't working for me. The hands-on integration with my company's actual API gave me real-world experience that was much more valuable than just theoretical learning.

This project also taught me a ton about how I like to learn. I prefer the hardest part first, which to me is setting up the environment and writing the code, then from there I can dive into the intricacies of the material I'm learning as I go. Oftentimes I think learning comes with too much hand holding and then I lose interest. If I can get to coding faster and apply knowledge as I go, then I become more engaged in the activity. If I had to go back and start over, I'd dive right into the LinkedIn Learning module and then expand my knowledge into other areas like my companies open API a lot sooner.

The biggest challenge was debugging the API authentication issues, which taught me the importance of attention to detail when working with tokens and endpoints. This troubleshooting experience was actually more valuable than if everything had worked perfectly from the start.

Conclusion

This project confirmed my interest in technical roles and showed me that I can learn complex technical concepts when I approach them in a way that matches my learning style. Moving from a Client Success Manager role, I now have concrete API experience that makes me more valuable to both my current company and potential future employers in Product Management or Solutions Engineering roles.

Looking to the present, I have a working foundation that I can build upon. The API integration I created is just the beginning - I can now expand it to include more KPA endpoints, add batch processing capabilities, or integrate it with other systems like Power BI for data visualization.

Looking to the future, this project opens several paths. I can continue building more sophisticated integrations within my current company, pursue technical certifications to complement this hands-on experience, or use this foundation to transition into more technical roles. The confidence I gained from successfully completing a real-world API integration gives me the motivation to tackle more complex technical challenges.

What I could do next includes expanding the current integration to handle multiple users at once, adding scheduled synchronization, building a user interface for the API, or exploring how to integrate this data with Power BI for reporting and analytics. Overall I’m extremely excited about the future possibilities.

References

LinkedIn Learning. (n.d.). *Building RESTful APIs with Node.js and Express*. https://www.linkedin.com/learning/building-restful-apis-with-node-js-and-express-16069959/restful-apis-with-node-and-express

KPA Solutions. (n.d.). *KPA Flex API Documentation*. https://api.kpaehs.com/

MongoDB, Inc. (n.d.). *MongoDB Documentation*. https://docs.mongodb.com/

Postman, Inc. (n.d.). *Postman API Platform*. https://www.postman.com/

Studio 3T. (n.d.). *Studio 3T Community Edition*. https://studio3t.com/

Node.js Foundation. (n.d.). *Node.js Documentation*. https://nodejs.org/en/docs/

Express.js. (n.d.). *Express.js Documentation*. https://expressjs.com/

Axios. (n.d.). *Axios HTTP Client Documentation*. https://axios-http.com/docs/intro