## 1. Briefly describe the artifact. What is it? When was it created?

Global Traveler is a travel website that I created just prior to the commencement of CS 499. I intend to turn it into a real travel blog. The features that are worked on specifically for this class are the “Destinations” page, which lists where I’ve been, what the cost was and when the best time to go is.

## 2. Justify the inclusion of the artifact in your ePortfolio. Why did you select this item? What specific components of the artifact showcase your skills and abilities in databases? How was the artifact improved?

The main justification for including this artifact is that it is a full-stack application and therefore has components that meet all three criteria to enhance. I selected it because it was the most current of any of my projects, and I want to turn it into my own personal website later. The components that showcase my skills and abilities in database primarily present in that my “Destinations” page data is no longer hard coded in as a JSON. For this enhancement, I added the MongoDB database.

## 3. Did you meet the course objectives you planned to meet with this enhancement in Module One? Do you have any updates to your outcome-coverage plans?

I did meet my course objectives that I planned. One update that I would make is that I have also met Course Objective 5 (in terms of security) by implementing an apiKeyMiddleware function that checks for a valid API key in the request headers before allowing access to the API route. If the API key does not match the one stored in your environment variables, the middleware returns a 403 Forbidden response, preventing unauthorized access.

*Figure 1: Middleware implemented*



## 4. Reflect on the process of enhancing and modifying the artifact. What did you learn as you were creating it and improving it? What challenges did you face?

I think the most difficult part of this enhancement was the securing of the database. I wase very hesitant to do it, and I thought I had to turn it into a whole user authentication affair where I had to log in using Firebase, etc. However, this was not the case, I am happy to know I had other options. API key protection was most direct, and it was easy to implement after I understood it.