

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

The original artifact I am basing my improvements on is the CS-465 travlr project, which was a full stack web application I created in July of 2023.

2. Justify the inclusion of the artifact in your ePortfolio. Why did you select this item?

I chose this item because creating a full stack web application would provide an excellent portfolio item for applying to web development jobs.

3. What specific components of the artifact showcase your skills and abilities in software development?

The artifact required me to use the MEAN stack, showing the ability to use a MongoDB database, angular frontend and node/express backend. As a part of that it also showed skills with Javascript.

4. How was the artifact improved?

My improvements to the artifact begin with using the Phoenix Framework and the elixir language. The Elixir language is a functional language which is quite different from Javascript, and more niche. The Phoenix framework likewise is very different from using the MEAN stack, and being less popular is more difficult to learn due to a paucity of resources.

5. Did you meet the course outcomes you planned to meet with this enhancement in Module One?

In the Module One Assignment, I planned to create a web app which allowed for a .csv file containing an underwriter's quotes to be uploaded. I also planned for this information to populate on the webpage and be displayed, in an analogous way to the travlr artifact I am very loosely basing the full stack web app on. This enhancement meets that functionality. It has the ability for a .csv file containing quote information to be uploaded, and then imported to the database. It can

then be viewed as a list in a format which is much easier on the eyes than a Looker report or reading a .csv file in Excel or Google Sheets.

6. Do you have any updates to your outcome-coverage plans?

Yes, the web app ended up having expanded functionality beyond that which I scoped out in the Module One Assignment. For example, due to the way the Phoenix framework expects you to implement the MVC(Model View Controller) architecture, I added functionality for quotes to be viewed individually, edited and also deleted. While I did not have that as a part of the initial plan, the added functionality adds to the usefulness of the application for the business use case. It would allow an Underwriter to, for example, update the quote effective date upon being informed by the broker that the account is renewing on 9/1/24. Deleting a quote is also a useful feature as it allows quotes which have either been bound or which the Underwriter knows we won't win to be removed from the system. This reduces the visual clutter on the quotes display and allows the Underwriter to just focus on active deals.

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

In creating and modifying the artifact I had to learn a lot about the Elixir language and particularly the Phoenix Framework. I had to learn how to create a Phoenix application in the first place, as well as learn how to implement the MVC pattern. I had to learn enough about the Elixir language to successfully do this. Moreover, the .csv importing functionality required me to learn about the new "live view" feature of the Phoenix Framework, as the only resource I could find showing how to do that used that new technology.

8. What challenges did you face?

Learning the Elixir I needed for the Phoenix Framework was not particularly difficult, and largely consisted of googling how to do a certain task using Elixir, for instance googling how to compare strings in Elixir. The Phoenix Framework was certainly more difficult however. This was in large part due to a paucity of resources, and in particular a paucity of resources which are up to date with the latest or even a recent version of the Phoenix Framework. This meant that I had to learn using the guides on the official docs and going through other people's projects on GitHub. While this\ was challenging, I feel a sense of accomplishment for having learned enough this way to get the app to have the functionality I wanted.

Checkpoint	Software Design and Engineering	Algorithms and Data Structures	Databases
Name of Artifact Used	Artifact Name: TravlR, travel full stack application Origin: CS-465 CS-465 Final Project	CS-465 Final Project	CS-465 Final Project
Status of Initial Enhancement	Complete	Complete	Complete
Submission Status	Submitted	Submitted	Submitted
Status of Final Enhancement	Complete	Complete	Complete
Uploaded to ePortfolio	Yes	Yes	Yes

Status of Finalized ePortfolio	Complete	Complete	Complete
---	----------	----------	----------