The artifact I chose for the Software Engineering and Design portion of the final project is my Final Project from Computer Science 340. This project created a RESTful API that linked to a Mongo Database. The REST API was created using Python, Jupyter Notebook, and Plotly Dash to create a dashboard to display database information, a pie chart, and location data for the table created from the database information. This project was created between January and February of 2021, and through the course of the project creation I learned about REST APIs and learned how to do client-side scripting for client-server development. This was an area that I hadn’t worked with much before, and the project as designed only reads from the database and doesn’t make use of the rest of the CRUD module created for it.

This artifact was included in my ePortfolio because it was one of the most difficult projects I worked on during the course of my degree, and it shows my ability to create a product that meets a client’s objectives. This artifact was created with the intent of enabling a rescue training company to work with local animal shelters to find animals that meet specific criteria to begin rescue training. This artifact also showed my ability to use innovative skills and techniques for the design solutions. My original artifact is already a good example of this because I show how I moved from using buttons to radio selection items when I realized that the radio items would work better for the requested features.

Currently, I am still completing the work needed to display that I am able to complete the ideas I articulated. I was able to update the software design to include a user entry section, but there is still improvement needed for the logic and troubleshooting to determine why entries are not being added to the database. I have met some of the course objectives I initially laid out in Module 1 and through my Code Review, but have not achieved all of the goals I had set for myself with this Milestone. I spent more time focused on building the environment than I should have, and I should have started this earlier instead of waiting and trying to gain access to an older environment.

While updating and modifying this artifact I had to modify it to work on the new operating system and update the authentication portion of the CRUD module created to ensure that it was able to successfully access the database. I also modified the naming convention for the CRUD module so that it made more sense with the project than keeping the names used to submit homework. After getting my environment up and running and ensuring the original project was able to run without issues I began making my enhancements. My large planned enhancement for this project was adding a user entry section so that users could add entries to the database to keep this project in continued use. While I was adding this enhancement I learned more about the core components of Plotly Dash, and learned how to add more sections to the interface. I also learned how to use those input components of Plotly Dash, and realized that to enable user entries I would have to use a separate input section for each of the dictionary components and then write those entries into a new dictionary before adding it to the database. I then added some logic for the button, so that the entry would be added if the number of button clicks was greater than a comparison number. After the button clicks n\_clicks would increase, and then theoretically it would go through the IF loop I created to add that entry to the database and re-read the whole database to the dataframe being used to display the data. The design of the dashboard has been updated so this new section appears to the user, but I cannot determine why when the Create method is being called the program is not adding the entry to the database. This is a challenge that I still have not been able to overcome, but I am working on looking up resources and re-evaluating the logic I am using to determine where the error is.

During my updates I also re-evaluated the location data logic, and cannot determine why two of my radio selections only update the pie-chart and not the location data. The pie-chart and the location data follow the same logic to get the data from the viewData being created by the table, so I am not sure why one updates and the other doesn’t. This logic is still being evaluated. I feel confident that given more time I will be able to address the issue with my user entries and gain a better understanding of the location data issue. The past two to three weeks I have focused more on gaining access to my old environment instead of just rebuilding and starting on the project, so given more time I am sure that I will be able to course-correct and achieve my enhancement goals.