1-4 ePortfolio Selection and Refinement Plan

Michael P Clisbee

CS 499 Computer Science Capstone

Southern New Hampshire University

**Category One: Software Engineering/Design**

For Category One: Software Engineering/Design, I will select work done in CS320 – Software Testing, Automation and Quality Assurance, which focuses on locating and resolving software security vulnerabilities by creating secure code and testing procedures to locate issues before code implementation. My enhancement plan will include expanding the complexity of my Unit Test files created in Java and edited and implemented using Visual Studio, Terminal and XCode. In addition, enhancements will be made to the associated ContactTest.java and ContactServiceTest.java files linked to their respective java files. This will demonstrate skills learned in altering Java code to be more secure, as well as in the creation of efficient test files, created specifically for their respective java files for testing. As seen in the Structural Model representation below, each unit test requirement will have attributes and methods included which will complete the process or unit test needs.

Diagram

Description automatically generated

**Category Two: Algorithms and Data Structures**

For Category Two: Algorithms and Data Structures, I will select work done in CS250 – Software Development LifeCycle (SDLC), which focuses on the Agile team roles and how their teamwork results in an overall rounded functional Java application where the code includes added functionality and imagery for a better user experience. My enhancement plan will include expanding the current API complexity of my TopFiveTravelDestination.java and SlideShow.java files with added resource folder imagery to give the user a better visual experience viewing destination information through a simple GUI that is user friendly. This will be done through the enhancement of current Java code to include better code documentation to elaborate on code functionality resulting in minimal errors and a fluid transition between screens. This will demonstrate knowledge and skills gained in the use of Java coding to create simple applications that the user can easily manipulate but not corrupt. Code will be altered and tested using Java IDE, XCode application and Terminal applications for Mac.

Diagram

Description automatically generated

**Category Three: Databases**

For Category Three: Databases, I will select work done in CS340 – Advanced Programming Concepts in Client/Server Development. For this course I initially created a dynamic dashboard linking several databases in the Jupyter Notebook using .csv, .py, and .ipynb Python and Javascript language files in MongoDB. The key databases to be used will be ProjectTwoDashboard.ipynb and Animal\_App.py, with a aac\_shelter\_outcomes.csv file. The objective is to apply database systems concepts and principles to create a client/server database application that interfaces with client-side code. This will pull data to display geo-mapping, statistics, and other user data from a database powered by MongDB. My enhancement plan will include expanding on the current MongoDB API (application programming interface) via JavaScript to make it more user friendly, as well as provide more selection options through improving on my current project code. This includes enhancing CRUD (create, read, update, delete) elements of my code. This will demonstrate skills learned in Python code when creating user interfaces and efficient CRUD implementation. As shown below in the Behavior Model, the database will be improved upon to have more input options which in turn will have more varied results.

Diagram

Description automatically generated

**ePortfolio Overall**

If I fulfill my enhancement plans, it will demonstrate skills learned in the languages of Java and Python that I would apply toward improving on my current code and using that knowledge to improve on the various API the code is controlling. Namely the ability to create effective test files for their respective java files to ensure they function correctly. As well, creating detailed CRUD functionality, adding Create, Read, Update and Delete functionality with safeguards included. The only limitations I can see would be the lack of full expert knowledge in these languages which will hinder the code from being as expansive as it should be. As well, if how are each of these to be demonstrated or presented in the GitHub site without access to specific applications. They are not stand-alone apps and have to be run within their associated programs. The only thing that will be viewed would be the source code itself, which might be enough.