Module 4 – Milestone Three

Kenneth Lockhart

Southern New Hampshire University

CS-499-13459-M01 Computer Science Capstone

Gene Bryant

March 30, 2025

I have chosen for category two, Algorithms and Data Structure, to focus on an artifact from CS 360. The artifact from CS 360 was an android application that was designed to be an Event Planner. This Event Planner was coded in Android Studio using Java as the coding language. The Event Planner had basic functionality in the form of being able to create, edit, and delete events as well as display, reminded, and store those events. In order to complete such a task, the program made use of two databases, one for users and one for events. These databases were then used to complete the as for mentioned tasks. This project was started at the tail end of 2024 and reached its minimal viable product on 12/13/2024 with a few cut features that were not required for submitting.

The primary reason for choosing this artifact for my ePortfolio was to continue the original goal of the project. As stated during the process of developing the application a few features were cut to ensure that the minimal viable product could be delivered. As a result, I want to continue the design and engineering of the application to bring the project to a better closure point. In order to continue the design and engineering of the application, I had to expand the use of the current data structure and design and algorithm to present that data structure in a good format for the end user. Additionally, I wanted to gain additional practice in an unfamiliar environment of mobile development. I was able to through the creation of the daily planner portion of this artifact demonstrating my familiar skills of Java and stretch my knowledge mobile development. In order to complete the development of the daily planner fragment I had to completely create a new method to display events by expanding the ability of the existing data structure. This improvement not only greatly improves the functionality of the end product, but it also brings the application to a much better closure and feeling of completeness. When I looked

at the application before I always felt it was lacking and felt very unfinished. Now I look at with a different opinion, I can see a full functional product that is ready for actual use.

The primary outcome I set out to achieve with this enhancement was to demonstrate my ability to use my techniques, skills, and tools to deliver value and accomplished industry-specific goals. This outcome was by far the most important to this, as stated I felt that the application did not reach those goals when it was submitted in December. Now I feel like it finally reached that goal, and it was done through the use of well-found and innovative techniques, skills, and tools. However, it was not the only outcome reach, I had the opportunity to correct a few minor details within the code during process. During the process I design, develop, and deliver clear visual improvement to the daily planner which will allow it to be easily understood while being technically sound for different audiences or context. An example of this was making the format of the daily planner easy to read, it follows the current time zone no matter where you are, and it informs the user of overlapping events. Additionally, during the review of the code I noticed many details that could be adjusted to better improve the application, specifically in the name of security. Some variables were left accessible to other classes, or as a field when they could have been more locked down. I went through and fixed though variables to help ensure that those fields do not get exploited. Overall, I was surprised that despite focusing on one outcome just how much I was able to relate parts of it to other outcomes and while it might not be easy to pick out all the details, I can say a majority of the outcome can apply in some capacity to the artifact.

In summary of this enhancement and modification of the artifact there are several things I had to face and learn. First, it is important to note that I have a decent background of Java experience with little to no mobile development practice. This means that while I understand the language itself and how it functions, I struggle with understanding it in a mobile development

mindset. The very first challenge is that not a whole lot of people have made a daily planner from scratch. This limited my ability to research ideas as all I got was to use some library someone else created. Previously, I had mentioned my biggest challenge in design and engineering of the application to get a good visual for the application. While this is still trying another part of this challenge, I saved to talk about category two was the algorithm that was needed to generate the view and overlay graphics. First off, in order to optimize the algorithm, I only have it act on data that should be displayed rather than running it on all data. Like the calendar and upcoming events view I had a separate function store the events into arrays of data if they are relevant to the user and if the event time is part of the current date selected. Combining the use of the database and arrays allows for a quick way to not only clear data when needed but to also iterate over that data without querying the database. With the data prepared we can then clear the views and lockout list and run the algorithm on the acquired data structure from the database. Starting by formatting the date into two categories, time and date. These are then used to determine 4 important flags for the algorithm so it can correctly generate time blocks. They are the start day the selected, is the current day after the start day, is the current day before the end day, and lastly is the end day the current day. If the start day was the current day, we need to set the start of the block from 0 to current hour from the start time. If the end day is the current day, then we need to set the end block to the end hours instead of 24. In the event that neither is true given previous preparation the vent should span the whole day but as a double check the algorithm verifies this by checking if both after start and before end flags are true and setting the start to 0 and end to 24. Finally with the start and end times we can figure out the difference which is used for the size of the block. From here the algorithm just scales the blocks based on the display and size between hours to get the positions. This portion of the code was the most difficult part of the

process of developing this application as it required insuring that all the numbers lined up, the elimination of daylight-saving factors, time zone factors, and display scaling. The process of adding the ability to display to events which was an idea from the beginning was rather simple by adding a data structure in the form of two array list work as a lockout list by adding the hours consumed by the block and by multiplying float by 100, I was ability to make it place event very close to each other in a single list before moving to the second list. Ultimately, I know that the algorithm can still be enhanced and optimized as I did not add a feature to allow overlapping lists or more than two lists. However, given the limited details on this topic on the internet I was rather impressed with the solution I found. The biggest improvement I can see to this algorithm is in the multievent view as it takes time to perform lockout while minor, I could see having a lot of events slowing down the process. Ideally, I would like to reduce the amount of nest loops to help produce a better algorithm but that felt outside the scope of this project and would take far longer to design from scratch then this project has available. Overall, in the end I think the improvements came out very well given the limited time spent on it. I believe it really shows how far I have come in mobile development since December and shows how I can overcome challenges and design algorithm and data structures to solve problems. Additionally, it shows that I am always improving and thinking critically about my work.