Daniel Ruppert

Professor Steaanelli

CS-499

05/26/2025

Enhancement Two: Data Structures and Algorithms

The artifact I selected for the Data Structures and Algorithms category is the same Android application used throughout this project, originally developed in CS-360. I chose this artifact because it offered significant opportunities to apply and refine algorithmic principles and data structure usage in a meaningful, real-world context. The application manages user-created events, which made it a natural candidate for enhancements such as caching, sorting, and recurrence logic, which rely on efficient algorithms and proper data modeling.

I selected this artifact for inclusion in my ePortfolio because it demonstrates several core competencies in algorithms and data structures, particularly in the context of building scalable and efficient Android applications. This artifact was enhanced to include features such as an LRU (Least Recently Used) cache to optimize daily event retrieval, sorting algorithms to display events chronologically, and logic for calculating recurring event intervals. I also implemented an enum structure to handle recurrence types more effectively, improving both code maintainability and input validation. Additionally, I developed logic for handling series-based event deletions and normalized phone number inputs to align with the constraints of Android’s SMS manager. These improvements illustrate my ability to apply algorithmic thinking in real-world scenarios, balancing performance, clarity, and robustness. The enhancements were made with a clear intent to strengthen runtime efficiency and user experience, making this artifact a strong representation of my growth in this area.

With this enhancement, I fully met the course outcomes I originally planned to target. These included Outcome 3, which focuses on designing and evaluating computing solutions that solve a given problem using algorithmic principles and computer science practices while managing design trade-offs, and Outcome 4, which emphasizes the use of well-founded and innovative techniques, skills, and tools to implement solutions that deliver value and align with industry-specific goals. Each improvement I made was intentional, involving careful evaluation of not only the technical implementation but also considerations around runtime efficiency, readability, and scalability. These enhancements directly reflect my ability to make thoughtful design decisions grounded in algorithmic reasoning and real-world applicability. As such, no updates were necessary to my outcome coverage plan, as the completed work remains aligned with the originally identified objectives.

Enhancing and modifying this artifact offered valuable opportunities to deepen my understanding of both algorithm design and practical implementation within an Android environment. One major learning experience was implementing an LRU cache for optimized event retrieval, which required a balance between memory efficiency and lookup performance. I also learned how to calculate recurrence intervals algorithmically, reinforcing my understanding of temporal logic and data structure alignment. Another key area of growth was developing logic to manage entire series of events, which challenged me to think about parent-child relationships and efficient deletion across related records. Beyond core algorithms, I explored the use of Android’s InputFilter interface to restrict user input at the UI level, something I hadn’t used before, which improved validation without requiring extra logic in backend processing. Overall, the enhancement process helped strengthen my understanding of trade-offs in algorithm selection, system performance, and user experience.