Jeffrey Flores

September 21, 2025

CS-499 – Milestone Two – Narrative

Weight Trackit is a mobile application designed to make weight logging and goal tracking fast, private, and motivating. Core features include daily weight entry, goal setting, progress visualization, real-time sync via Firebase (so entries persist across devices), and milestone notifications when targets are hit. The product features a clean, low-friction UI designed for users who prioritize functionality over clutter. The initial Android version launched during my Mobile Architect and Programming course in April 2025, and I created the iOS version in September 2025, bringing the project to a multi-platform state.

I selected this app because it represents the kind of work I want to do professionally, mobile engineering tied to cloud data and user-centric design. It showcases my skills with algorithms and data structures. It shows my ability to make the algorithms more efficient by using sorted insertion and binary search for entries. Since I am dealing with data for my progress view feature, I am able to show how I can manipulate data structures to be able to fetch the data more quickly by using user Ids as keys to my dictionaries as one strategy.

The largest improvement was achieving O(1) on my screen refreshes when the user switched the views from weekly to monthly view. I enhanced the app by adding a progress view that allows users to see their entered weights in comparison by weekly view and monthly view. This specific enhancement with fetching the data more efficiently made the app work faster and be more responsive to the user’s interaction with it. Instead of the UI being stalled for poor connection to internet it can work faster to get the data so it’s not affected by these other factors.

One of the course outcomes that I met with this enhancement and with this artifact was the algorithm design and evaluation. This allowed me to model time series entries, compute trends/targets, and choose NoSQL schemas that optimize reads for charts. I was also able to hit the Engineering tools and practice outcome by using Firebase SDKs, dependency managers, linting, and crash/analytics dashboards. I was able to organize code into testable modules. Another outcome I achieved with this enhancement is solving a given problem using algorithmic principles. I was able to manage tradeoffs with binary search or deep nested search.

The first issue I ran into when implementing my ProgressView feature was that I didn’t have any real data to test the view with. I ended up having to add data manually to the backend database through creating JSON files to upload to my database to test with. Another issue I ran into was displaying the week data on my line chart for my progress view. When displaying the date, it was too long to fit and display properly. The solution for me was to change the naming convention of the weeks to W1 – W52 so we can cover all the weeks in a year, and it is a smaller real estate to show the week data. I learned about the tradeoffs of having data structures that were properly matched to search queries set up rather than having issues with key value pairs that made a search difficult to make more efficient.