Erin Kiesel
CS 499: 4-2 Milestone Three: Enhancement Two: Algorithms and Data Structure

Artifact Description

The artifact I selected for the Algorithms and Data Structures category was the Weight Tracking mobile application that I originally developed in CS 360: Mobile Architecture and Programming that I took in Fall 2024. The app allows a user to create an account to log their daily weight, set a goal weight, and view past weight entries as a list. The original application was very basic and offered opportunities for enhancements.

Justification for Inclusion

I chose this artifact because as it was designed in CS 360, it had limited functionality, and I felt that there were opportunities to enhance the application to provide a more meaningful user experience. At the end of CS 360, the application simply displayed a list of weight entries to the user; they couldn't draw any meaningful conclusions from the data other than receiving a notification when they reached their goal.

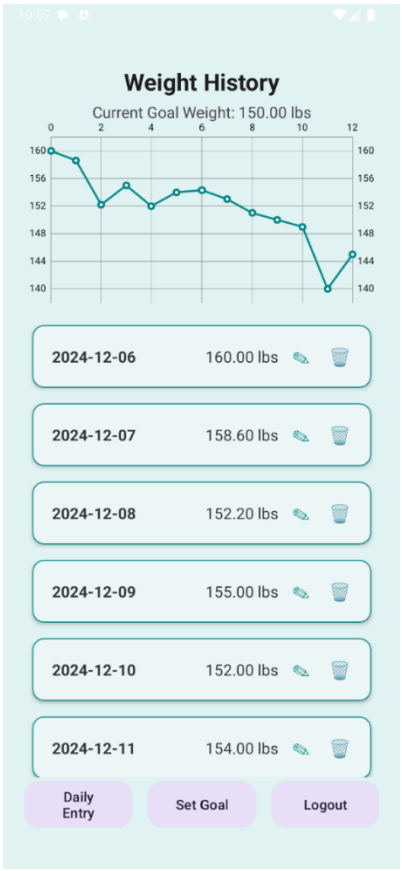
I added custom max and min heap classes to track the highest and lowest weights efficiently. I also created a WeightTrendAnalyzer class that uses logic to detect the longest streak of consecutive days a user logs their weight, the longest streak of days their weight decreased, and their longest weight plateau where they saw very little change. So now instead of only seeing a list of entries, the user sees a visual graph of their weight progress as well as some quick stats regarding their min and max weight, as well as some streak data. There is evidence that streaks encourage users to engage more with apps (Anizoba, 2025), so including this as an enhancement could encourage the user to meet their weight goals and use the app more often.

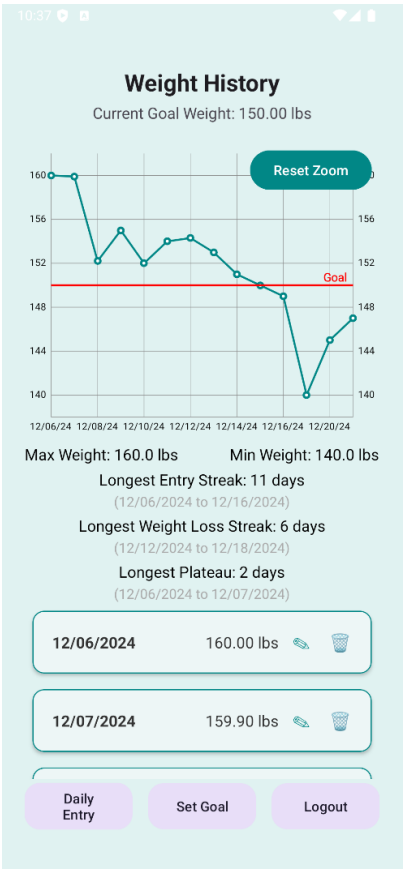
I had originally planned on adding a calendar view as well, but I have not gotten to that. I also received feedback that it wouldn't necessarily be an enhancement that would meet the requirements for this category, so I instead spent time on the custom max and min heap classes. I may still implement the calendar later, time permitting.

Below you can see the progress of the development of the application. The first image is at the end of CS 360, very basic, just a list of entries with dates. The second image is at the end of Milestone 2, overall, a better, more professional looking app with some meaningful information with the addition of a chart. Then the third image is the app currently, I've formatted the chart better with more meaningful date labels on the x-axis, it's zoomable and scrollable, and has a clear goal marker on it. There are also some quick and meaningful stats for the user right below the chart and then the scrollable list is below that.

Erin Kiesel
CS 499: 4-2 Milestone Three: Enhancement Two: Algorithms and Data Structure

I also formatted the dates to be in the format that most Americans are used to: month/date/year.





Course Outcomes

Yes, I believe I met the course outcome related to designing and evaluating solutions using algorithms and data structures. I didn't end up using HashMaps or TreeMaps like I initially planned, I added custom logic to analyze weight trends like downward streaks and plateaus, and I created my own MinHeap and MaxHeap classes to help display the minimum and maximum weights. I also used sorting and handled some edge cases around the dates and formatting. I tried to make design choices that made sense for what I was building without overcomplicating things.

Reflection

Enhancing and modifying this artifact taught me a lot about how to take a simple app and start adding more meaningful features behind the scenes. Figuring out how to calculate

trends like streaks and plateaus pushed me to think more carefully about how to structure the data and how to compare dates and weights efficiently.

One challenge I ran into was formatting and displaying the results clearly, especially when dealing with date ranges and keeping the UI consistent. Another was making sure I didn't break anything while adding logic that handled edge cases, like entries with the same weight or missing days.

I also realized that I have a habit of writing a lot of logic directly in HomeActivity, even when it would probably be cleaner and easier to manage in its own class. I'm definitely still learning how to better organize my code. Working through these improvements helped me get better at writing code that's not just functional, but also more maintainable and user-friendly.

References

Anizoba, P. (2025, April 14). *Designing for User Retention: The Psychology Behind Streaks*. Retrieved from Medium: https://medium.com/design-bootcamp/designing-for-user-retention-the-psychology-behind-streaks-cf0fd84b8ff9