CS 499 Milestone Three: Enhancement Two (Algorithms and Data Structures)

Mohamed Aziz Zaghdoudi

## 1. Briefly describe the artifact

This artifact is a flight scheduling and assignment algorithm developed as part of my Airline Dispatcher application. It was originally created as part of my CS 499 enhancements to extend my existing airline scheduling app into a full dispatcher/agent system. The core function assigns flights (arrivals and departures) to available gate agents by ordering based on ETA/ETD and balancing fairness across agents.

## 2. Justify inclusion in your ePortfolio

I selected this artifact because it clearly demonstrates my ability to design and implement algorithms and apply data structures to solve real-world scheduling problems. The enhancement showcases my skills in reasoning about algorithmic efficiency, working with time-based constraints, and structuring data flow between different roles (dispatcher and agent).

## 3. How was the artifact improved?

Originally, the app had static data and no assignment logic. In the enhancement, I introduced algorithms that use arrays, maps, and queues to manage flights and agents. I refined the assignment logic to consider ETA for arrivals and ETD for departures, while fairly distributing flights among available agents. I also connected this logic to the UI so dispatchers can run scheduling, preview draft assignments, and publish them, while agents only see their own tasks for now.

## 4. Did you meet the course outcomes you planned in Module One?

Yes. This enhancement demonstrates my ability to design and evaluate computing solutions using algorithmic principles. It also shows my ability to use innovative techniques and tools in practice. Compared to my Module One plan, I added more focus on fairness and data structure trade-offs (maps vs heaps) to make the algorithm more realistic.

## 5. Reflect on the process of enhancing and modifying the artifact

Through this enhancement, I learned how to design scheduling logic that works in both dispatcher and agent contexts, and how to manage shared state effectively with React Context. The biggest challenge was balancing simplicity with scalability ensuring the algorithm was understandable but also realistic. Debugging and ensuring IDs aligned between sample flights, agents, and assignments was also tricky, but it reinforced my understanding of why unique identifiers matter in data structures.