**Software Engineering ACS56000**

**HW-12**

**Design Patterns**

I have applied the Strategy Pattern to my first HW HealthCalculation code, which is a behavioral design pattern that enables selecting an algorithm’s behavior at runtime. This pattern is suitable here because we can separate different calculation strategies for health metrics.

The Strategy Pattern has been applied here by:

1. Creating a CalculationStrategy interface that defines the common interface for all calculation strategies
2. Implementing there concrete strategies:
   1. DurationSumStrategy: Calculates the sum of all durations
   2. CalcoriesMeanStategy: Calculates the mean of all calories
   3. MaxCaloriesForDurationStrategy: Find maximum calories for a specific duration
3. Adding a HealthCalculator class that:
   1. Uses the strategies to perform calculations
   2. Allows switching between different calculation strategies at runtime
   3. Provides a clean interface for executing calculations

The main effect of implementing this pattern is to provide flexibility in adding new calculation without modifying the existing code. Each calculation strategy is also encapsulated in its own class. It has easy maintainability and testability because the calculation logic is separate from the main business logic and each strategy can be tested independently.

