Mission 3 Grading Form

Total: 100 marks

Breakdown of Marks:

1. Use Case Model (50 marks total)
    1.1. Use case diagrams – 15 marks
        1.1.1. Diagrams should be consistent with the rest of the use case model (naming, etc.)
        1.1.2. Proper use of <<include>>, <<extend>>, and inheritance relationships
    1.2. List of actors – 5 marks
        1.2.1. Brief descriptions are clear and concise
    1.3. List of use-cases – 5 marks
        1.3.1. Brief descriptions are clear and concise
    1.4. Detailed description of 15 use cases – 25 marks
        1.4.1. Most important use cases are identified
        1.4.2. Flow of events is clear and concise with an appropriate level of detail
        1.4.3. Entry and exit conditions are clearly defined
2. Architecture (15 marks total)
    2.1. Describes system architecture and subsystem decomposition – 10 marks
        2.1.1. Clearly explains the service provided by each subsystem
        2.1.2. Clarifies relationship between subsystems
    2.2. Explains rationale for this architecture – 5 marks
        2.2.1. Lists any tradeoffs taken into consideration
        2.2.2. Lists advantages and disadvantages of the architecture chosen
3. Static Model (20 marks total)
    3.1. List of main classes – 10 marks
        3.1.1. Only important, meaningful classes are included
        3.1.2. Each class is properly named
        3.1.3. The role of each class is given
        3.1.4. Traceability – references are given to use cases that involve the class
    3.2. Class diagrams – 10 marks
        3.2.1. Diagrams are clear, correct, and consistent with the rest of the document
        3.2.2. Aggregation, composition, and generalization relationships are used properly.
4. Requirement Traceability Matrix (15 marks total)
    4.1. Traceability matrix – 15 marks
        4.1.1. Complete – includes all use cases from part A, all classes from part C, and all requirements as listed in the most recent version of the requirements spec document.
        4.1.2. Comprehensive – every requirement and class is illustrated or used in at least one use-case.