Homework 4
3 pages
52 points

Assigned 29OC01
Due 09NO01

- Problems 1-4 are group effort, and are worth 8 points each. The format of group work will be explained in class.
- Problems 5-6 are individual effort, and are worth 10 points.
- All diagrams must be drawn using Visio, or similar tool. Subsequent groups will be improving on existing diagrams, so exchange format must be decided on Monday, 29OC01.
- The system should be understandable based upon viewing the UML diagrams. Minimal additional text may be included in the diagrams to aid understanding. No separate textual description is required.
- Final versions of diagrams are required *from* student groups by 6:00pm, Tuesday, 06NO01 for completion of problems 5 and 6.
- Final versions of diagrams will be *returned to* the students by midnight, Tuesday, 06NO01 for completion of problems 5 and 6.

- Four groups;
- Progressive improvements to diagrams during 29OC01-09NO01;
- Groups will evaluate individual members participation.

	29OC01	31OC01	02NO01	05NO01
Group Sub1	UseCase	Statechart	Sequence	Class
Group Sub2	Class	UseCase	Statechart	Sequence
Group Sub3	Sequence	Class	UseCase	Statechart
Group Sub4	Statechart	Sequence	Class	UseCase

Group Work: Comprehensive System Design

The following homework evaluates your comprehensive knowledge of UML, and your ability to apply UML to decompose and complete a preliminary design of the software for a common system. The system is described below.

LF8R Software Company

The objective of the software system is to control an elevator system with the following constraints.

- A 4-story building needs a set of 3 elevators to service it.
- The 3 elevators have well-defined individual behavior, as well as synchronized control from a global control software. As an example, suppose that all three elevators were on the 1st floor and that two passengers on the 2nd and 3rd floors, respectively, called the elevators. Then one elevator would go to the second floor, one elevator would go to the third floor, and one elevator would remain parked on the first floor. The control of which elevator goes where is the responsibility of the global control software.
- On each floor, to the exterior of the elevators is an up arrow and a down arrow to beckon the elevators, based upon whether the passenger desires to increase or decrease altitude.
- The individual elevators have the following functionality:
  - Doors open and close automatically on floors to which they have been beckoned, or to which they are delivering passengers. The current state of the door (open or closed) can be overridden by pressing the 'close door' or 'open door' button inside the elevator;
  - Each elevator has only one door;
  - Each elevator has a button for each floor (floors 1 through 4);
  - Each elevator has an alarm button which sounds an alarm while pressed, but offers no other functionality;
  - Each elevator has an 'emergency stop' pull button, that stops the elevator immediately (even between floors). Pushing the button in restores it to normal operation, with the following exception: If the button is pulled and pushed three times within a period of three days, then the elevator delivers passengers to the closest floor and ceases operation with the doors open. Essentially, the elevator control makes the assumption that if the emergency button is tampered with too much, then there must be an inherent error in the system;
  - There are several override keys (actual physical keys) that may be used on the elevators:
    - Key 1: 'on/off' key which disables operation of an individual elevator on the 1st floor with the doors open;
    - Key 2: 'fire on/off' key which, when activated in any of the three elevators, delivers all elevators to the first floor, opens doors, and leaves them open.

Notes:

- List all assumptions regarding the behavior described above.
- Note the *context* of the diagram is described in the following problem statements. That is, don't draw a statechart diagram for a single elevator if the problem asks for the state behavior of the entire system.
- Hint: Look for clues in the EECS Atrium.

1) Draw the use case diagram for the context of the entire system, including all three elevators and the global control. Include appropriate actors and relationships between use cases.

2) Draw a system level class diagram that includes the hardware context of the system (elevators, doors, buttons, etc.), including classes, responsibilities, and appropriate relationships. However, attributes and operations are not required in this diagram. Note that question 5) is a design level class diagram for the elevator class, so clever system level analysis now may save design level effort later.

3) Draw a sequence diagram for the system in normal operation given the following scenario. Hint: include at least these objects: elevator1, elevator2, elevator3, passengerA, passengerB, and globalControl.

   - Elevator 1 is on the 4th floor.
   - Elevator 2 is on the 3rd floor.
   - Elevator 3 is on the 2nd floor.
   - A passenger is on the 1st floor.
   - A passenger is on the 3rd floor.
   - A passenger on the 1st floor beckons an elevator.
   - The global control software sends the closest elevator (elevator 3) to the passenger.
   - The passenger boards the elevator and presses floor 3.
   - Another passenger beckons an elevator to the 3rd floor.
   - Global control software determines that elevator 2 is already on the 3rd floor and opens its door.
   - As the passenger on the 3rd floor is boarding elevator 2, elevator 3 arrives on the third floor and its door opens.
   - The passenger on elevator 2 recognizes a friend getting off of elevator 3 and presses the open door button on elevator 2.
   - The passenger from elevator 3 boards elevator 2 with the friend, presses floor 1 and both students ride the elevator to the 1st floor where the door opens.

4) Draw a statechart diagram for the context of an individual elevator under both normal and emergency operation. The statechart for the individual elevator must account for all behaviors described in the problem description for an individual elevator. Note that this is not a statechart diagram for the entire system, but rather for an individual elevator. That is, do not attempt to decompose the coordinated state behavior required to control three elevators simultaneously.

<u>Individual Work: Design Level Class and Package Diagram</u>

5) Based upon all of the above diagrams in problems 1-4, draw a detailed, design-level class for the elevator class. Include all relevant attributes and operations. Only name and description fields for each attribute and operation are necessary. That is, there is no need for ranges, resolutions, etc.

6) Organize the class diagram from Problem 2 into packages. As appropriate, include proper usage of import, export, and generalization. Also include correct visibility.