

QUESTION 1

Create a Class Diagram using good OO techniques (like inheritance, interfaces, abstract, overriding, overloading, constructor state modification, etc.) for the following problem:

A ferryboat can carry up to 100 human passengers, exactly 50 crew members, up to 25 cars and up to 5 buses. A ferryboat can be empty: no passengers and/or no vehicles, but a crew must always be present. Every passenger must have a ticket. Every vehicle must belong to exactly 1 passenger. A car can carry at least 1 passenger and at most 6 passengers. A bus can carry at most 50 passengers, plus one mandatory driver (who is a passenger). Company “Mighty Boats” owns exactly 5 ferryboats, as described above. Vehicles can be added to and removed from a ferryboat, but to be added they must have a passenger with a ticket. Passengers can be added to and removed from a ferryboat, but to be added they must have a ticket. Crew members can be added to and removed from a ferryboat. “Might Boats” has a single data structure (database), in RAM, of all the active people and vehicles for all of its boats. Crew members want to be able to access passenger and vehicle information from within the ferryboat they are currently on without seeing information from other boats. “Might Boat” staff on shore want to be able to access passenger and vehicle information from any boat.

QUESTION 2

Draw a UML Activity Diagram for the following Watch & Alarm problem description:

A user is creating an app that displays a digital clock. The app has two screens: a screen that displays the digital clock, and a screen that permits the user to set an alarm (the user can set multiple alarms).

The digital clock display screen shows the current time, formatted this way → hours:minutes:seconds. The clock increments each second and the display is updated each second. A picture of an alarm is displayed if the user has set an alarm. The screen changes to red when the alarm is ringing.

The alarm screen permits the user to create one or more alarms stored in a list data structure. The user presses ADD to create an alarm and DELETE to remove an alarm. Pressing ADD asks the user for the time and then adds the time to the list. If the list is empty, then there are no alarms. If the list has one or more items, then the digital clock display will turn red for 1 minute when the alarm sounds. Pressing delete asks the user for the alarm order number and then garbage collects that entry from the list, if it was a valid order number.

Not all the information is useful for your diagram. Make sure your diagram is well-formed using all the techniques covered in class: diagrammatically, well-formed objects, contracts, etc.

QUESTION 3

Draw a class diagram that uses design patterns to solve the following problem:

AAA News publishes an email newsletter. They have subscribers and publishers. Subscribers are users who receive the email each day. Publishers are reporters that submit news stories. AAA News automatically assembles the new stories into an email and sends that off to their subscribers once a day.