SYS466 Fall 2018

Professor: Barb Czegel

**Lab 6 (1% of Final Grade): Introduction to Object level Sequence Diagrams**

Due: At the end of the lab session

**Objectives:**

- Learn how to use the Entity Manager control object to model persistence in sequence diagrams:
    - Getting data from a persistent store
    - Creating new data and putting it into a persistent store
    - Removing data from a persistent store.
- Learn how to add reference attributes to an object in a SD
- Learn how to indicate multiple instances of objects in a SD

This is an individual lab and must be done in the lab room itself. **INDIVIDUAL SUBMISSIONS ONLY. PLEASE DO NOT USE INTERNET EXPLORER TO ACCESS THIS LAB – USE FIREFOX OR CHROME.**

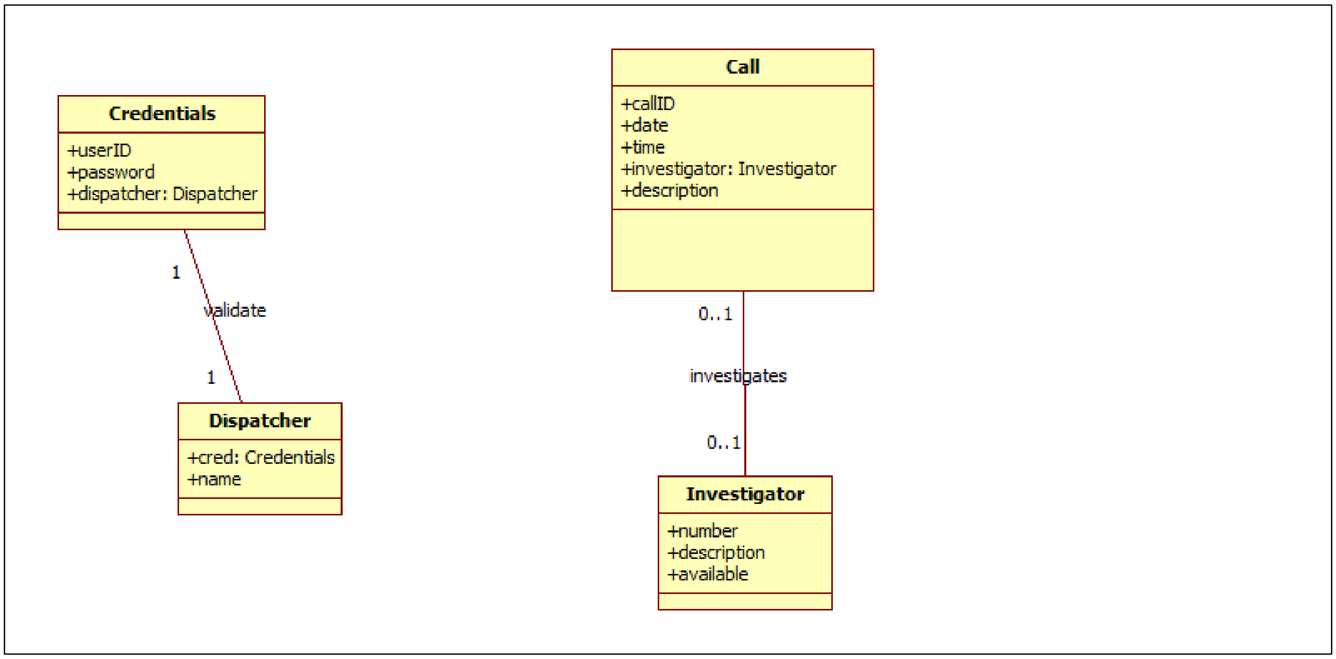
**Submission and Credit:**

- Only those students who sign in will get credit for the lab. Submission is to be done via Blackboard. Email submissions WILL NOT be accepted.
- **Name your submission name_Lab6.uml, for example JannaSmith_Lab6.uml**

**Special Instructions For This Lab:**

- You will be working in the **Investigate Model of the uml file you have been given**. In that model you will find class diagrams and empty sequence diagrams. Please use the sequence diagrams for your answers. **Do not create any new diagrams**

**The "Investigate" case study:**
Calls come in to a dispatcher for incidents that need to be investigated – for security purposes the investigator name does not appear anywhere in the system, only the number. Each incident is assigned one investigator.



# Exercise 1:

**Create the SD for this scenario in the diagram SignIn.**

**Scenario: Sign in**

| Actor – Dispatch Worker | System |
|---|---|
| Enters userid and password | Authenticates user information in the database. It is valid. (Note…you don't need to use a guard condition because this scenario is only for the case in which the credentials match)<br>Displays the dispatcher name as well as all calls, showing id, description, and investigator numbers for each call |

Hints and Notes:

- When you are dealing with sets of objects in a SD, it is often useful to use the "multi instance" feature. Pull your objects into a SD, select the one you want to be multi instance and in the properties box check the multi instance option. The object's appearance will change.
- You will probably want to do this for :Call
- When you have designated objects as multi instance you don't have to specifically show loops in most cases; multi-instance implies that you are looping through all instances.

# Exercise 2:

**Create the SD for this scenario in the diagram NewCall.**

**Scenario: New Call**

| Actor – Dispatch Worker | System |
|---|---|
| Requests to start a call. | Displays the numbers of all available investigators.<br>*Hint: when you are getting investigators from the DB, one option is to specify "available = true". This simplifies processing* |
| Enters date, time, description and selects an investigator | Creates a new call, adds the selected investigator. Persists to the DB.<br>Sets investigator availability to false. Persists to DB. |

Hints and Notes:

- In this scenario you may want to create a separate investigator object named something like i so that you can set availability to false.
- This will also allow you to pass that object to Call when you add it to call.
- You can assume that the Domain Controller uses the investigator number to pick that investigator out of the set or you can show that specifically. One way to do this (there are many) might be a message to self e.g. i:Investigator = findInvestigator(number).

# Exercise 3:

**Create the SD for this scenario in the diagram UpdateCall.**

**Scenario: Update Call -- Remove Investigator from call**

| Actor – Dispatch Worker | System |
|---|---|
| Enters call id | Displays call id, description and investigator number. |
| Asks to remove investigator | Removes the investigator from the call<br>Set investigator availability to true<br>Persists call and investigator.<br>Gives warning message: this call has no investigator. |

Hints and Notes:

- You are not deleting the investigator, you are simply removing the investigator from the call. This probably involves setting the reference attribute to null or the equivalent.

## Exercise 4:

**Create the SD for this scenario in the diagram DeleteInvestigator.**

**Scenario: Delete Investigator**

| Actor – Dispatch Worker | System |
|---|---|
| Enters investigator number | Retrieves investigator from DB, Displays investigator number, description, availability.<br>Checks to make sure availability is true; in this scenario it is true. |
| Asks to delete investigator | Removes the investigator from the DB.<br>Deletes the Investigator object.<br>*Hint: to delete an object use the destroy message type* |

## Exercise 5:

In the Controller diagram update the DomainController as follows:

- Add all operations that the controller will need to carry out the 4 sequence diagrams in this lab.