**CS3431: Project Description**
**C-Term, 2013**
**Building a Database Application**
**Phase 3: Complete Hospital Application**

**Release Date:** Feb. 11, 2013.
**Due Date:** Feb. 21, 2013 (11:59PM).
**Teams:** The project is done in teams of two.

**Description:**
In this phase you will complete the design of the ***Hospital System Database*** by ensuring that the application requirements and constraints are all met.

**Part 1 (50 Points): Database Views (10 Points each requirement)**

- Create a database view named ***CriticalCases*** that selects the patients who have been admitted to Intensive Care Unit (ICU) at least 2 times. The view columns should be: *Patient_SSN, firstName, lastName, numberOfAdmissionsToICU.*
    - ***Hint:*** *ICU is a service that is stored in table 'RoomService'*

- Create a database view named ***DoctorsLoad*** that reports for each doctor whether this doctor has an overload or not. A doctor has an overload if (s)he has examined more than 10 admission cases, otherwise the doctor has an underload. The view columns should be: *DoctorID, gender, specialty, load.*
    - The *load* column should have either of these two values 'Overloaded', or 'Underloaded' according to the definition above.

- Use the views created above (in addition to the original tables) to report critical-case patients with number of admissions to ICU greater than 4.

- Use the views created above (in addition to the original tables) to report the female overloaded doctors. You should report the doctor ID, firstName, and lastName.

- Use the views created above (in addition to the original tables) to report the comments inserted by underloaded doctors when examining critical-case patients. You should report the doctor Id, patient SSN, and the comment.

**Part 2 (70 Points): Database Triggers [10 Points each requirement)**

Given the following requirements, create one or more triggers that ensure that these requirements are always satisfied. Think of the three events *Insert, Update,* and *Delete* because you may need triggers on one or more of these events to meet the requirements below.

***Hint:*** *You are allowed to combine several requirements in one trigger*

- Any room in the hospital cannot offer more than three services
- The insurance payment for any admission covers 70% of the total payment
- Ensure that regular employees (with rank 0) must have their supervisors as division managers (with rank 1). Also each regular employee must have a supervisor at all times.
- Similarly, division managers (with rank 1) must have their supervisors as general managers (with rank 2). Division managers must have supervisors at all times.
- General managers should not have supervisors at all times.
- When a patient is admitted to ICU room on date *D*, the futureVisitDate should be automatically set to 3 months after that date (D + 3 months). The futureVisitDate may be manually changed later, but when the ICU admission happens, the date should be set as mentioned above.
- When a patient is admitted to the hospital, i.e., a new record is inserted into the *Admission* table; the system should print out the names of the doctors who previously examined this patient (if any).
    - Hint: Use function dbms_output.put_line()

## Grading:

The maximum grade is 120 Points. Late submissions follow the rules stated on the website.

## Deliverables:

Each team should deliver two files as follows:

1) Text file (.sql) that contains all SQL commands from Part1 and Part2 above. Also ensure that the file includes the creation of all needed tables (copy it from the submission of Phase 2). The file must be executable from SQL using command:
   SQL > @<fileName>

   The file must run correctly, creates all tables, and then the views and triggers. If the file's syntax was not correct (and the file did not run), you will loose 10 points in addition to any other deductions.

2) Any comments or assumptions that you have, you can write them to a separate .doc or .pdf file.

   ***Important:*** In this file include for each of the requirements in Part 2, what trigger(s) did you create to enforce this requirement. You need to state the event type (Before or After), the operation (Insert, Update, or Delete), and the granularity (For each row, or For each statement) and on which table.

3) Put the two files in a single zip file that will be submitted as below.

## Submission (Each team give one submission):

- Submit electronically by the due date via blackboard.wpi.edu website. Make sure your two files (the text file .sql, and your report .doc or .pdf) are zipped, and you upload one file.
- Each team submits one copy (from either of the team members).
- ***No hard copy submissions***