LABORATORY MANUAL

CZ2007: Introduction to Databases
CSC206/CPE303: Database Systems
Software Lab 2 (Location: N4-01c-06)

Implementation of a Database Application

SESSION 2015/2016
SEMESTER 2

SCHOOL OF COMPUTER ENGINEERING
NANYANG TECHNOLOGICAL UNIVERSITY

1. OBJECTIVES

Upon completion of the assignment, the student should be able to:

- Construct an entity-relationship model at a conceptual level
- Map the model into a schema of a relational DBMS
- Implement the given schema on a relational DBMS
- Use a database language (SQL) for manipulating and updating data

2. LABORATORY

This is a group assignment. Each group consists of **four to five** members from your laboratory group. You have the choice of selecting your group members. However, all the names within your group must be given to the laboratory technician(s) during your first laboratory session. Name lists with respect to each laboratory group are available in the "Public Folders" towards the beginning of the semester.

Note that the laboratory will start from the **third week** of the semester onwards and that you might need more than the mentioned five sessions for the actual implementation. You are also encouraged to **start early** with your assignment (as soon as the topic is covered in the lectures).

Attendance is taken for all supervised laboratory sessions. It is within the responsibility of each student to sign-in at the beginning of each session. **Failing to sign-in for the first, third, or fifth lab session may result in F grade for the respective assessment.**

3. INTRODUCTION

The assignment covers the portion of the course concerning data modelling, database design and implementation from the user's viewpoint. Thus, the assignment involves modelling as well as implementation aspects of the database course.

The overall aim of the laboratory is to develop an application based on a given data model using a given database management system. This exercise will bring you through a crucial first part of the life cycle of a database application. It is assumed that the data analysis has been performed. Note that this manual provides you with more information than is required for the first laboratory session, e.g., not all constraints can be modelled in the beginning but are included at a later implementation stage. In contrast you might require additional information for the understanding of the application. Proceed by stating your assumptions in written form and / or ask your laboratory supervisor.

4. DESCRIPTION OF THE ASSIGNMENT

The description of the application is given in the appendices. This includes the back-ground and general requirements of the application, conceptual information about the system and its users as well as a list of queries that must be fulfilled as a minimum.

Note that teamwork is required. Every team has to submit one solution. **No individual submission is allowed.**

4.1 First Laboratory Session: Creating an ER Diagram

Appendix A gives conceptual information about the project obtained after a partial system analysis was performed. Based on the appendices, construct a suitable ER diagram. Analyse the cardinality of relationships, the usage of weak entity sets, choice of entity sets etc. and compare them with alternative solutions. The laboratory technicians will provide the necessary information at the beginning of the lab session.

You need to submit the followings at latest **three working days** after the first laboratory session.

- A hard copy of your ER diagram and written discussion of your solution (maximum one page), which highlights the reasons for the chosen design.

4.2 Second Laboratory Session: Finalization of the ER Diagram

There is no submission due with respect to the second laboratory session. In this lab, each group should finalize their database design based on the feedback received from their lab supervisor. Please note that the second laboratory session is a free-access session, i.e., attendance is not compulsory (but recommended in case the group has questions).

4.3 Third Laboratory Session: Generation of Normalized Database Schema

In this lab you must ensure that the database is at least in 3NF. Follow the general guidelines covered during the lectures and tutorials to produce suitable normalized relations. For each relation, the key(s), primary key, and functional dependencies must be specified. If a relation is generated due to normalization of an original relation then the normalization steps must be presented.

You need to submit the followings at latest **three working days** after the third laboratory session.

- A hardcopy of the normalized database schema and FDs associated with each relation. If a relation created from the ER diagram violates the 3NF form then this should be highlighted along with the decomposed normalized relations. Note that for this lab no SQL code should be submitted. Hence, the structure of your solution shall be similar to the following example:

  `R1(A,B,C,D)`

  Keys: `AB`, `AD`
  Primary Key: `AB`
  FDs: `AB → CD`, `A → D`
  The relation is in 3NF.

4.4 Fourth Laboratory Session: Implementation of the database schema

There is no submission due with respect to the third laboratory session. In this lab, the finalized database schema must be implemented using SQL DDL commands. Your implementation should clearly incorportate the primary and foreign keys, data types,

integrity constraints, value-based and tuple-based constraints. Solve the implementation by using the MS SQL Server software.

Please note that the fourth laboratory session is a free-access session, i.e., attendance is not compulsory (but recommended in case the group has questions).

4.5 Fifth Laboratory Session: Final demonstration

The fifth session is the final assessment of your implementation. The implementation obtained from the previous laboratory session has to be extended by incorporating necessary triggers and additional constraints. In addition, you have to formulate the SQL statements for the sample queries in Appendix B.

This session has two components. First, **at the beginning of the lab** a hardcopy of the schema implemented using the SQL DDL commands together with constraints and sample queries need to be submitted. Auto-generated relations are not permitted. Hence, the structure of your solution for the database schema definition shall be similar to the following example and written by yourselves:

```
CREATE TABLE name (
        attr1 datatype NOT NULL,
        attr2 datatype,
        ...
        PRIMARY KEY (attr1),
        FOREIGN KEY (attr3) REFERENCES name(attr1)
                ON DELETE ... ON UPDATE ...,
);
```

The second component of this session involves **demonstration** of your system. All team members are required to contribute actively during the demonstration session. Additionally, the laboratory supervisor will ask individual questions. During the demo session, the evaluation shall be based on the following points:

- Implementation and execution of additional queries on the spot
- Answers on and understanding of the design and related issues
- Demonstration of the proper working of your implementation
- Additional effort in terms of implementation etc.
- Presentation quality

Note that your group might be required to begin the presentation at any time during the fifth laboratory session, i.e., one team will be asked to present at the beginning of the session. All applications should run on the provided hardware and software components of the Software Laboratory 2 using MS SQL Server.

APPENDIX A

"SnatchCar" is a mobile app that allows users to book "taxi" trips from drivers who provide service with their own cars. Suppose that you are to construct a database for SnatchCar, and the requirements as follows:

- Every user is uniquely identified by his/her id (e.g., NRIC). In addition, each user has a user name of his/her choice. The user's phone number and gender are also recorded.
- Each driver is uniquely identified by his/her id (e.g., NRIC). In addition, the driver's first name, last name, DOB, gender, phone number, and driver's license number are recorded.
- Each driver must register his/her car with SnatchCar before he/she can serve any trip. Each car is uniquely identified by its car plate number. In addition, the car's brand, model, color, and number of seats are recorded.
- If a user intends to book a trip, he/she needs to submit a booking request via the SnatchCar app. Each booking request has a pickup location and a drop-off location, each of which is represented as a (latitude, longitude) pair. In addition, the number of passengers to be picked up is also recorded with the booking request.
- Once SnatchCar receives a booking request, it would compute the price of the trip and prompt the user about it. If the user agrees with the price, then the request is broadcasted to all available drivers (if any) who are currently near the pickup location. The first driver that responds is assigned to handle the request, upon which the booking request becomes a confirmed booking, and the user is notified about the driver's name, phone number, and car plate number. On the other hand, if no driver responds within 3 minutes, then the booking request becomes a failed request.
- Whenever a user completes a trip booked via SnatchCar, he/her receives 1 merit points. The total merit point of a user is visible to the drivers when they consider a booking request from the user.
- A user may cancel his/her booking request or confirmed booking any time before the deriver arrives at the pickup location. However, cancelling a confirmed booking would give the user 1 demerit point. The total demerit point of the user is also shown to the drivers when whenever user submits a booking request. Furthermore, if the user keeps a confirmed booking but does not show up at the pickup location, then he/she receives 5 demerit points.
- A user is allowed to rate the drivers that handle his/her confirmed bookings. At most one rating can be made for each confirmed booking. Each rating is an integer in the range of 1 to 5.
- If a user is unhappy with the service provided by the drivers that handles his/her confirmed bookings, the user may file a complaint to SnatchCar. Each complaint has a text, and is associated with a confirmed booking. Each complaint is handled by a SnatchCar employee. Once the complaint is processed, the user is notified about the result.
- Each SnatchCar employee is uniquely identified by his/her ID (e.g., NRIC). In addition, the employee's first name, last name, gender, DOB, and salary are recorded.
- The database should support the queries listed in Appendix B.

Note that the provided information may not be complete. Many aspects of the system's functions and details may have been omitted. It is expected that the teams come up with their own solutions in case of inconsistencies or missing information. However, you have to keep track

of these aspects and explain your assumptions if asked for the reasons. Extensions to the implementation of the basic system are encouraged.

APPENDIX B
Queries

1. Find the average price of trips booked from 6pm to 9pm in December 2015.
2. Find the drivers who receive more than 100 ratings of 5 in the month of December 2015, and order them by their average ratings.
3. For all confirmed bookings in December 2015 that were eventually cancelled, find the average time from booking confirmation to booking cancellation.
4. Let us define the "latency" of an employee by the average that he/she takes to process a complaint. Find the employee with the smallest latency.
5. Let us consider the car models used by SnatchCar drivers. Produce a list that contains all car models and, for each model, the number of cars used for SnatchCar.