INTRODB

Machine Project: Developing a Small-Scale Database Application Release Date: July 7, 2014

Learning Outcome

This machine project on *Developing a Small Scale Database System* provides a venue for students to achieve the following learning outcomes:

LO1. Design conceptual, logical and physical data models to represent the information requirements of the organization used in the Field Work;

LO2. Formulate SQL queries to provide access to data (creation, update, manipulation, retrieval) to support the information requirements of the organization;

LO3. Design and implement a small-scale database application based on the output of the Field Work.

Description

Team compositions from the Field Work activity will be retained. The teams will design and implement the features of a prototype database application based on the requirements that they have gathered and documented in their respective Field Work reports. Throughout the software development project, the team will iteratively design data models to represent various levels of the data and information requirements of the organization, including the associated business rules. The team will also develop a database application to provide various users of the organization access to the database using well-formulated SQL query statements and based on the identified business processes.

Milestones and Deliverables

The following deliverables must be submitted on the stated schedule:

A. Revised Requirements Specification

The Requirements Specification document is a revision of the Field Work report and contains the following:

- Forms for data access by various user groups
  - The forms are screenshots of your database application's user interface, designed using your target language. You can use any of the following languages to develop your INTRODB database application - C, Java, C#, Visual Basic, and Python.
  - Data access includes facilities for creating new records, updating and viewing existing records, as well as for viewing the generated detailed and summary reports.
- Conceptual Data Model

The Requirements Specification document is a type-written, single-spaced report printed on short-bond paper. It contains the title page and the following sections:

1. Introduction

   In paragraph form, provide an overview of the database application that you will be developing. This is based on your Field Work but should be written from your own point of view. The following serves as your checklist for the items that should be present here:
   - What database application are you developing?
   - What features will your application provide? What business processes are these features supporting? What classes of users will use these features?
   - What data can your users access from your application?

2. Data Requirements of the <*XYZ System*>

In this chapter, the conceptual data model representing the data requirements of your business is presented, along with the description of the software features that you will implement in your database application.

2.1 Conceptual Data Model

Draw the conceptual data model.

Describe each entity type and the attributes it contains (including optional, composite, derived and key attributes; and attribute domain if applicable). You can use a tabular format to summarize the description of each entity type.

Describe the business rules, which are *statements from policies, processes or events in the organization that define or constrain its data*. These include the following (refer to the lecture slides for more details):

- The attribute domain, if applicable, for an entity type
- The entity integrity and key constraints of an entity type
- The relationships between entity types
- The multiplicity constraints between entity types

Some example possible business rules may include the following and may impact both your database design and application software.

- A password must have at least 6 alphanumeric characters;
- A password must be changed every 30 days (in which case, you will need to store the date when the password was last changed);
- A user cannot reuse his last three passwords (thus, you need a password history table);
- A user is given three attempts to login to the system (in which case, you will need to store the number of failed attempts and a status field indicating if a user has been locked out);
- A user can have three challenge questions (in which case, you will need to store the challenge questions and the corresponding answers for each user account).
- An employee can be one of three types, namely *salaried*, *hourly*, and *commissioned*.
- An employee can belong to only one department.
- An employee may be assigned to work on one or more projects.

Use examples (entity and relationship instances) to clarify your discussion as necessary.

2.2 Software Features

Provide a detailed description of the features that you will develop in your database application. Such features should represent business processes and data requirements that your application will be supporting. Follow the format as stated in the Field Work, but you must present your own user interface design.

Forms for displaying the listing and summary reports to be generated by the application software are included in this section. For each report, provide (a) a brief discussion of its purpose; (b) its users; and (c) the data/information that will be generated. There should be at least 6 reports.

Note the following:

- Your user interface should be neat and should show evidence of careful analysis in designing forms that will allow users access to relevant data and information while ensuring data integrity constraints are observed (through proper DB design and correct choice of UI elements).
- Grading will be based on user interfaces that support the user tasks / business processes.
- Graphics and colorful interfaces will not merit extra grades, as these do not achieve the intended learning outcomes.
- Reports will be graded based on their content and relevance to the organization.

Deliverable: Requirements Specification document.

- A printed copy of the Requirements Specification document to be submitted on **July 28/29, 2014**
- An email copy of the document with the subject heading **INTRODB MP Requirements Specification**, with filename **INTRODB_<section>_<lastnames of members>.doc**

B. Logical and Physical Data Models and Queries

The second set of deliverables includes the logical data models and the query statements.

1. Derive the relational data model for your conceptual data model.
2. For each form in your application that provides access to the database (creation, update, retrieval), show the query statement. Explain the query as needed.
3. For each report to be generated by your database application, present and explain the corresponding query statements that will be used.

Deliverable: Print and submit the Relational Model and Queries report on **August 4/5, 2014**

Penalties for the late submission of deliverables for A & B.

- For each late submission of required deliverables, 10 points will be deducted from the final project grade. Incomplete submission will also merit the same deductions.
- No late submission will be accepted beyond a day after the stated due date.

C. Final Software

Each team will present its database application on August 22 - 23, 2014. The actual schedule will be provided by your respective INTRODB teachers.

Deliverables: On August 22, 2014, ALL groups must submit a "virus-free" thumb drive containing the following items to their respective teachers. The thumb drive must be properly labelled.

- The application code and executable file
- The database scripts (or the database itself) needed to re-construct and populate the database
  - Prior to submission, clean your database. Remove all your test data. Then populate each table with at least 5 rows of valid and meaningful data.

* * * No submission = No software presentation * * *

The actual schedule of the presentation of your database application will be provided by your respective INTRODB teacher. During the software presentation:

- Each group must bring its own laptop, with the corresponding tools (IDE, compiler, mySQL server) already installed and ready for use.
- The application code / software will be installed and the database will be generated on the actual demo schedule, using the contents from the submitted thumb drive.
- All members must be present during the demo and must be ready to show evidence of involvement and understanding of the database system project.
- The team must prepare and use a test script during the presentation. The test script is a step-by-step guide simulating an actual business process that involves access to data in the database using the features provided by the application.
- A student may receive a grade of 0 if he fails to show up for the presentation, late for presentation, or cannot answer the questions.

Format of the Title Page (for Requirements Specification)

<Name of your Database Application>

A Requirements Specification Document
for the course on
Introduction to Databases
(INTRODB)

Submitted by

<lastname, firstname> of all group members, in alphabetical order

<Teacher's Name>
Teacher

July 28, 2014

Criteria for Grading

The Machine Project comprises 25pts of your final grade for INTRODB. Your Machine Project has 100 points subdivided into the following components:

Milestone 1: Requirements Specification	20pts
Milestone 2: Logical Model and Queries	20pts
Milestone 3: Database Application and Demo	60pts

Individual grading may be imposed for members who are not actively contributing to the fulfillment of the project requirements.