CIS 353 (Database)
Specifications for the Database Design and Implementation Project

In an immediate first meeting with your group, read and discuss these specs very carefully. Throughout the semester, if you have any questions, please ask as early as possible.

INTRODUCTION

- The purpose of this project is to integrate the phases of database design that we have, or will, study in this class. It includes: writing database specifications, ER modeling, ER-to-relational mapping, and database implementation and testing.
- This is a team project; each team consists of 4 members.
- Each team chooses its own application domain subject to the requirements listed later in this document.
- Every member of the team must work on every phase of the project. It is counterproductive to have person A design the ERD, person B design the relational schema, person C write the queries, and person D do the writing since the overriding consideration is that everyone must experience the learning opportunity offered by every phase of the project.
- The proposal is worth (20 points); and the final report is worth (80 points).
- This document specifies specific formats for the deliverables. The professional quality and organization of all presented material does count in grading.

Discussions among different teams are allowed and highly encouraged.

IMPORTANT DATES

Refer to the syllabus for milestones and dates.

PROJECT TASKS

The project involves the following tasks. The tasks are listed first and then explained in more detail.

1. Choosing an application.
2. Submitting a proposal.
   Also, Submitting individual evaluations of team members – *see the evaluation form at the end of this document.* The evaluations are to be submitted directly to me by each team member.

3. Receiving feedback (from me and from another team).

4. Implementing the proposal, using Oracle.
5. Submitting a final **printed** report.
6. Submitting a final **electronic** report.

7. Submitting individual evaluations of team members – *see the evaluation form at the end of this document.* The evaluations are to be submitted directly to me by each team member.

1. **Choosing an application**

Identify an application that interests you. As we have seen through the examples in class, one has to make reasonable simplifying assumptions about the real world in order to keep the size of the project manageable.

Structure

Your application must result in an **ER diagram** that includes:

1. At least four regular entity types.
2. At least one weak entity type.
3. At least one multi-valued attribute.
4. At least one of each type of relationship cardinalities: 1:1, 1: N, and M:N.
5. Some relationships must have attributes of their own.
6. Participation constraints must be a reasonable mixture of partial and total constraints.
7. Sufficient variety of attributes to answer the type of queries required in this project (see later).

Integrity constraint types

The application must have an interesting set of ICs that include at a minimum the following types:

1. **Foreign keys**.
2. **1-attribute CHECK** (i.e. a CHECK IC that refers to a single attribute).
3. **2-attribute, 1-row CHECK**: A CHECK IC that relates more than one attribute in the same row. The IC must reflect a logical condition between the 2 attributes. For example, in the following relation:
   Student (Gnumber, …, dateOfBirth, dateEnrolled, classification, CreditHoursCompleted)
   Unacceptable IC: dateEnrolled must be after dateOfBirth.
   Acceptable IC: A sophomore must have completed at least 25 credit hours.

Query types

At a minimum, your queries must demonstrate the features listed below. *You may of course demonstrate more than one feature in any one query* and thus end up having to write fewer, but more interesting, queries.

1. A join involving at least four relations.
2. A self-join.
3. UNION, INTERSECT, and/or MINUS.
4. SUM, AVG, MAX, and/or MIN.
5. GROUP BY, HAVING, and ORDER BY, all appearing in the same query
6. A correlated subquery.
7. A non-correlated subquery.
8. A relational DIVISION query.
9. An outer join query.
10. A RANK query.
11. A Top-N query.

2. Submitting the Proposal

The proposal shall consist of the following items:

1. **A cover page that includes**: project title, names of team members, and the class/semester.
2. **Database requirements**: Two to three pages of text that describe precisely and concisely the database requirements. In order to improve readability, structure your description as follows:
   - One or two paragraphs to introduce the nature of the application (i.e. the big picture).
   - One paragraph per entity type. The paragraph should have the name of the entity as its title. The body of the paragraph should describe the entity and its relationships to other entities. The description should be in user terms rather than in technical terms. For example say: *A department may control several projects but a project can be controlled by one department only* rather than saying *Departments are related to projects by a 1:N relationship*.
3. **EER diagram**: Using the notation that we have used in class.
4. **Basic relational schema**: Just as we did in class where each relation that results from converting your ER schema is represented by one line of the form **R (A, B, C, D)**.
5. **Integrity constraints:** State, in English one type of ICs in your application in the format of the following table. (Please note that you'll have many ICs in your proposal. However, you need to list only one IC of each type in the table below.)

IC name & table(s)	IC type	English statement
	Key	
	Foreign Key	
	1-attribute	
	2-attribute, 1-row	

The proposal shall be submitted in printed and electronic formats by the beginning of class on the due date.

- The **printed** version must be single-sides and stapled at the top left corner (please don't use folders or covers). Use 1.5 line spacing and make sure that your pages are numbered.

- The **electronic** version ***must be a single file*** that is uploaded to BB by the team coordinator.

Individual Submission

In addition to the team submission above, each team member is also required to submit (directly to me) a filled 'Team-Skills Survey Form' which is the last page of these specs.

3. Receiving feedback (from me and from another team).

I'll post, within 48 hours of proposal submission, all the proposals on Blackboard. Everyone is expected to study all the proposals in preparation for the scheduled discussion. In addition every team will be assigned to review and give written feedback on the proposal of another team.

NOTE

You are of course allowed to fine-tune your proposal based upon the feedback that you receive.

4. Implementing your proposal using Oracle

After you have developed and tested all the components of your project, consolidate your work into one SQL command file and call it `project.sql`. ***The file must have the following contents***:

Make sure to place the `highlighted` commands exactly as per the skeleton below

```
SPOOL project.out
SET ECHO ON
/*
CIS 353 - Database Design Project
<One line per team member name>
*/
< The SQL/DDL code that creates your schema >
     In the DDL, every IC must have a unique name; e.g. IC5, IC10, IC15, etc.
--
SET FEEDBACK OFF
< The INSERT statements that populate the tables>
     Important: Keep the number of rows in each table small enough so that the results of your
     queries can be verified by hand. See the Sailors database as an example.
SET FEEDBACK ON
COMMIT;
--
< One query (per table) of the form: SELECT * FROM table; in order to print out your
database >
--
< The SQL queries>. Include the following for each query:
     1.  A comment line stating the query number and the feature(s) it demonstrates
          (e.g. – Q25 – correlated subquery).
     2.  A comment line stating the query in English.
     3.  The SQL code for the query.
--
< The insert/delete/update statements to test the enforcement of ICs >
     Include the following items for every IC that you test (Important: see the next section titled
     "Submit a final report" regarding which ICs to test).
     –   A comment line stating: Testing: < IC name>
     –   A SQL INSERT, DELETE, or UPDATE that will test the IC.
     COMMIT;
--
SPOOL OFF
```

Run the above command file. It will produce an output file called `project.out`. Do the following:

1. Edit the resulting spooled output file `project.out` by changing its **font size to 9**.
2. Print your file `project.out` – ***use single-sided printing***
3. Number the pages of the printout by hand as A1, A2, A3, … and attach them as an appendix to your final report which is described in the next section.

5. Submitting a final PRINTED report

The final PRINTED report must be a single-sided printout that contains the following items in the order in which they are listed below:

1. A cover page that includes:
   - Project title
   - Team member names
   - Course (section), semester, and year
   - Table of contents (with page numbers for each section – see below for sections)
2. The database requirements specifications (in English).
3. The EER diagram.
4. The basic relational schema which resulted from converting your ER.
5. The BCNF relational schema if your schema in (4) above was not already in BCNF.
6. A table containing a list of 4 integrity constraints (***one of each type***) that you have implemented and tested. The table must have the following format:

IC name &table(s)	IC type	English statement	Page # where implemented	Page # where tested
	Key			
	Foreign Key			
	1-attribute			
	2-attribute 1-row			

Notes about the above IC table

- Of course you will likely have implemented several ICs. However, the above table requires you to state only one of each type.
- The page numbers in the table refer to the page numbers of the appendix that you printed in section 4 above.

7. The printout of your **`project.out`** that you printed in the previous section.

IMPORTANT

If a query (or feature) doesn't work, write a note (by hand) next to that query or feature. You may get partial credit for a serious attempt. On the other hand, if your query doesn't work and I find that out by inspecting your results, then no partial credit will be given.

6. Submitting a final ELECTRONIC report

Create a file containing the following:

a. Items 1-6 of the previous section.
b. Your file **`project.sql`** (**not `project.out`**).

Upload the file to BB. The file is to be uploaded by the team coordinator.

Individual Submission

In addition to the team submission above, each team member is also required to submit (directly to me) a filled 'Team-Skills Survey Form' which is the last page of these specs.

This page is intentionally left blank

Team-Skills Survey Form

Date ___________________ Team Number __________

Course number (section): _________(____) Semester/Year:_______/______ Name: _______________________________

…Your Team Members' LAST Names …

Why this survey is important
In your career, you will be asked periodically to evaluate your coworkers. This survey should help you start thinking of and practicing for, that task. The School is extremely interested in developing students' team skills. This survey will give the School important feedback regarding what it needs to emphasize in the future.

Legend
SA=Strongly Agree
A=Agree
D=Disagree
SD=Strongly Disagree
NA=Not Applicable or
No Opinion

		#1	#2	#3	#4
1	**This team member contributed to group discussions.** *Issues to consider include:* – *Participated constructively in meetings.* – *Contributed good ideas to the project.*	*SA A D SD N*	*SA A D SD N*	*SA A D SD N*	*SA A D SD N*
2	**This team member was dependable.** *Issues to consider include:* – *Worked around his/her schedule in order to set up meeting times.* – *Attended all meetings.* – *Delivered assigned tasks professionally and on time.*	*SA A D SD N*	*SA A D SD N*	*SA A D SD N*	*SA A D SD N*
3	**This team member showed initiative**. *Issues to consider include:* – *Acted to move forward rather than waited to be told what to do.* – *Was essential in the formulation of new ideas and solutions.*	*SA A D SD N*	*SA A D SD N*	*SA A D SD N*	*SA A D SD N*
4	**This team member was easy to work with**. *Issues to consider include:* – *Was objective in evaluating other members' ideas.* – *Accepted critique of his ideas graciously.*	*SA A D SD N*	*SA A D SD N*	*SA A D SD N*	*SA A D SD N*
5	**This team member contributed fair share of work.** *Issues to consider include:* – *Did at least as much work as any other member.*	*SA A D SD N*	*SA A D SD N*	*SA A D SD N*	*SA A D SD N*
6	**This team member communicated effectively in writing**. *Issues to consider include:* – *Expressed ideas clearly in writing (reports, emails, blogs, chats, etc.).*	*SA A D SD N*	*SA A D SD N*	*SA A D SD N*	*SA A D SD N*
7	**This team member communicated well when speaking**. *Issues to consider include:* – *Expressed ideas clearly during discussions and presentations.*	*SA A D SD N*	*SA A D SD N*	*SA A D SD N*	*SA A D SD N*
8	**This team member behaved in an ethical manner**. *Issues to consider include:* – *Is trustworthy.* – *Understands and upholds the standards of academic honesty.*	*SA A D SD N*	*SA A D SD N*	*SA A D SD N*	*SA A D SD N*
9	**Given the choice, I would work with this team member again** *Issues to consider include:* – *Should another opportunity arise in this (or in another) class, I'll be willing to work again with this team member.* – *I'll consider myself fortunate if this team member and I are hired in the same company after graduation.* – *If I were working as a team leader in a company, I'll be glad to hire this member in my team.*	*SA A D SD N*	*SA A D SD N*	*SA A D SD N*	*SA A D SD N*

***Additional Comments**: Please feel free to use the back of this form or attach another sheet of paper if you wish.*