

CS331- Project Assignment

Problem Description

Project 1

International Student Information System

Problem Description

The International Student office at Bellevue College is interested in developing a database to maintain information about the international students. The database will keep information about the courses the students have taken, their visa status, their on-campus jobs, etc. The International Student office is interested in creating a new database to carry this information because there are certain rules and regulations that apply to international students only, and therefore using the university's database would be inconvenient. The following are some of the rules and regulations that are specific to international students: (a) international undergraduate students should take at least 12 credit hours per semester; (b) international graduate students should take at least 9 credit hours per semester; (c) the I-20 should not be expired; (d) international students can work up to 20 hours per week; (e) international students should be working on-campus. They would need special permission from USCIS (US Citizenship and Immigration Services) to work off campus.

Database Design

We present the main entity types of this database. For each entity type, we provide some of the corresponding attributes. Use this information in order to: (a) Build an Enhanced E-R diagram; (b) Transform the Enhanced E-R diagram to a relational database. Identify the primary key(s) and the foreign key(s) for each relation. Draw the relational integrality constraints;

1. College: The main attributes are college identification number, name, address, name and address of the dean, etc.
2. Country: The main attributes are name, language, capital city, ethnicity, etc.
3. Course: The main attributes are course identification number, name, number of credits, etc.
4. Department: The main attributes are department identification number, name, address, name and address of the department head, etc.
5. Degree: The main attributes are name, description, etc.
6. Job: The main attributes are job name, job type, hours/week, employer's name, employer's address, employer's telephone number, etc.
7. Regulations and Laws: The main attributes are identification number, name, description, etc. refer to problem description above.
8. Student: The main attributes are social security number, name, birthday, gender, nationality, address, enrollment date, type (graduate or undergraduate student), visa type, major department, college, degree sought, etc. Note the following:
 - a. A student has an educational history. The history consists of the name of the schools attended, degree earned, GPA, etc;
 - b. The visa status for current students could be F-1, J-1, etc. The visa status for students who have already graduated could be H-1, OPT (optional practical training), etc;

- c. The relationship between student, degree, and college is a “*for the team to define*” relationship since a student may get a dual degree or get minors from different departments;
- d. The relationship between student and country is a “*for the team to define*” relationship since a student may have dual citizenship.

Application Development

Please note you will not be asked to implement all of the following queries, forms or reports during the Project. These are only to add some details on how the system can be use. During the project implementation you will be assigned a specific set of queries to implement.

The following are some of the queries, forms, and reports one can create in order to increase the functionality of the database:

Queries:

1. The following queries help to retrieve information about undergraduate students:
 - a. List the social security number, name, address, major department, and college for all the undergraduate students.
 - b. Create a query that prompts for the name of a country and returns a list consisting of the social security numbers, names and addresses of the undergraduate students from this country.
 - c. Create a query that prompts for the name of a college and returns a list with the social security numbers and names of the undergraduate students enrolled in this college.
 - d. List the social security numbers, names, and nationalities of the undergraduate students holding a J-1 visa.
 - e. List the social security numbers, names, and addresses of undergraduate students who are currently working. For each student, report the number of working hours per week.
 - f. Create a query that prompts for the social security number of a student and returns that student’s educational history (the name of the schools attended, degree earned, GPA, etc.).
2. Create the same queries as the ones described in part (1) to retrieve information about the international graduate students.
3. Present the name of the country that has the majority of the international students.
4. List the countries that have at least one representative (graduate or undergraduate student) in this university.
5. List the number of graduate and undergraduate students enrolled in this university each year for the last ten years.
6. Present the names of the college and department with the highest enrollment of international students.
7. Present the overall average GPA earned so far by undergraduate and graduate international students.
8. Present the average GPA earned so far by undergraduate and graduate international students. Group this information by country of citizenship.

Forms:

1. Create a user sign-in form together with a registration form for new users.
2. Create the following data entry forms that are used for database administrative functions: students, courses, departments, jobs, degrees, etc. These forms allow the user to add, update, and delete information about students, courses, departments, jobs, degrees, etc.
3. Create a form that presents academic information and contact information of the international students. The form should present the following academic information for each student: social security number, name, date of birth, gender, educational history, current education, and courses taken. The contact information consists of the following: student's address, e-mail, current phone number, permanent address and permanent phone number.
4. Create a form that presents detailed information about the courses that an international student is taking currently. Insert in this form a combo box to allow the user to choose the social security number of a student. Insert a textbox that presents the name of the selected student. Insert a subform that lists the identification number, name, and corresponding credit hours of the courses that the selected student is currently taking. Insert a textbox that presents the total number of credit hours the student is currently taking. In case that this number is fewer than 12 credit hours for an undergraduate student or fewer than 9 credit hours for a graduate student, display a message notifying the user that the student is enrolled for fewer than the required number of credits.
5. Create a form that presents information about the employment of an international student. Insert a combo box that allows the user to select the social security number of a student. Insert a textbox that presents the name of the selected student. Insert a subform that presents information about the current employment of the student. The subform should present the following: job type, employer's name, employer's telephone number, and employer's address. Insert a textbox that presents the total number of working hours per week.
6. Create a form that allows the user to browse through the regulations and laws related to international students. Create a subform that presents detailed descriptions of the selected regulation/law.

Reports

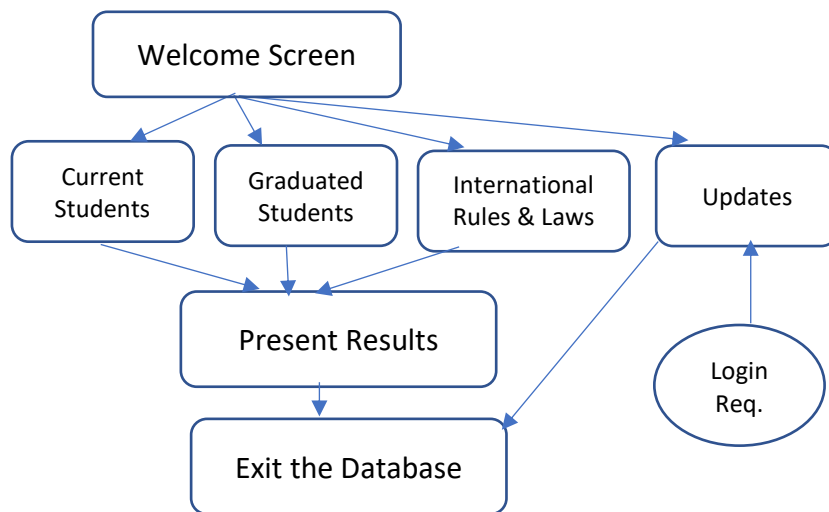
1. Report the following information for all the undergraduate international students: social security number, name, address, nationality, major department, college, GPA, and total number of credits earned so far. Group this information by nationality. Within each group, sort the information in descending order of credit hours earned.
2. Report the following information for all the graduate international students: social security number, name, address, nationality, major department, college, GPA, and total number of credits earned so far. Group this information by nationality. Within each group, sort the information in descending order of credit hours earned.
3. Report the following information about the international students who have already graduated: social security number, name, current visa type, job title, and the name and phone number of their current employer.
4. Report the following information about the students who will be graduating this coming spring: social security number, name, degree earned, name of the major department, college, enrollment date, graduation date, and GPA.
5. Report the following information about the students enrolled in the current semester: social security number, name, and a list of the courses the student is currently taking. For each course, present the following: course identification number, name, name of the department that is

offering the course, and the number of credits. Calculate the total number of credits each student is taking in the current semester.

6. Report the following information about the current employment of the international students: job title, employer's name, employer's address, and the total number of working hours per week. Group the information by students' social security number and name.
7. Report the following information about the visa status of each international student: visa type and starting date and expiration date of the visa. Group this information by student social security number and name.

Java Application Development

This database application can be used by the employees of the International Student office, the database administrator, etc. In the following figure we present a tentative layout of the system.



Graduated Students: The user browses this part of the database to learn about the following:

- (a) academic information,
- (b) contact information,
- (c) visa status, and
- (d) current employment of the international students who have already graduated.

International Rules & Laws: The user browses this part of the database to learn about rules and laws concerning international students. For example, there are restrictions about the following: the total number of required credit hours per semester, the total number of working hours per semester, etc.

Update: The update form requires an administrator login name and password. This form allows the user to add/delete/update the information kept in this database about students, departments, regulations, etc.

Java Application Required scenarios and Query sets

Your team will be responsible to complete the implementation and demonstrate the application behavior for the following 3 Scenarios functionalities in the application.

Please Note: For the application UI, I will not include complete screenshots here as I want each group to think how they want to present their application. Main screen UI Sample is provided as well as some others

Figure 1: Main Menu interface option

```
*****
                        *****
                        Welcome to the
International Student Information System
                        *****
*****
                        1. Current Students
                        2. Graduated Students
                        3. International Rules & Laws
                        4. Updates
                        5. Quit
Type in your option: 4
Enter User ID: raj
Enter password: raj
*****
```

Scenario 1

In Figure 1: Option 4 is used for database administrative functions. When the user choose **option 4 (Updates)** from the Main Menu interface “Figure 1”: The user should be prompted to enter the admin login information-- username and password. The application verifies the information and if correct, that means this person is an admin and he/she is authorized to do updates on the database. If username or password is incorrect the user will be prompted to try again with the valid admin login info or quit to the main screen.

Once the user successfully logs in, the application will show a new UI with the options below

1. **Option A: Insert new information** which includes 4 new sub options
 - Add new student information
 - Add new course information
 - Add new department information
 - Register a student in a course
2. **Option B: Delete some information** which includes 4 new sub options
 - Delete specific student information

- Delete specific course information
 - Delete department information
 - Withdraw a student from a course
3. **Option C: Update current information** which includes 4 new sub options
- Update specific student information
 - Update specific course information
 - Update department information
 - Update a student Course registration

For this Scenario, the team is responsible to implement and demonstrate the functionality for **only Option A (Insert new information)** and all **the 4 sub options**. The video recording should show how the application can be used to insert one example of each sub option under option A.

Also, in the case of registering a student in a course, after completing the registration a confirmation message will show and it should include how many credits the student has been registered in so far. In case this number is fewer than 12 credit hours for an undergraduate student, display a message notifying the user that the student is enrolled for fewer than the required number of credits.

Scenario 2

Figure 2: Current Students option interface

```
*****
Welcome to the
International Student Information System

1. Current Students
*****
1. Academic information
2. Contact information
3. visa status
4. Quit

Type in your option:
```

When the user chooses *option 1 (Current Students)* from the Main Menu interface “Figure 1”. It will show a new UI similar to Figure 2.

Implement and demonstrate the application behavior for the following options in **Figure 2**

1. When the user chooses *option 1(Academic information)*: It should take you to a new UI interface that gives the option to retrieve the following:
 - a. **Option A (Educational history for a student)**: This option should prompt for the social security number or student ID of a student and returns that student’s educational history (the name of the schools attended, degree earned, GPA, etc.).
 - a. **Option B (Collage Students information)**: This option should prompt for the name of a college and returns a list with the social security numbers and names of the undergraduate students enrolled in this college. If your system is designed for only one college then show the information for a department.
2. When the user choose *option 2 (Contact information)* from the main interface it should take you to a new UI interface that gives the option to retrieve the following
 - a. **Option A (All student contact details)**: It should return the list of the social security number, name, address, major department, and college for all the undergraduate students.
 - b. **Option B (Student from a Country)**: It should prompt for the name of a country and returns a list consisting of the social security numbers, names and addresses of the undergraduate students from this country
 - c. **Option C (Working students contact information)**: It lists the social security numbers, names, and addresses of undergraduate students who are currently working, and for each student, report the number of working hours per week
3. When the user chooses *option 3 (Visa Status)* from the main interface it should take you to new UI interface that gives the option to retrieve the following:

- a. **Option A (Visa Status Report)**: It should Report the following information about the visa status of each international student: visa type and starting date and expiration date of the visa. Group this information by student social security number and name.
 - b. **Option B (F-1 visa)**: List the social security numbers, names, and nationalities of the undergraduate students holding a F-1 visa.
4. When the user chooses ***option 4 (Quit)***: It should return the application to the main interface in figure 1

Scenario 3

When the user choose ***option 3 (International Rules & Laws)*** from the Main Menu interface “Figure 1”. It should allow the user to list through the regulations and laws related to international students with a brief description of the regulation/law

CS331- Project Assignment
(Task 3: Data Preparation and Design modification)
Due: Saturday Nov 21st, 2020 @ 11:59 Pm

Introduction:

Now and after you completed the project design, you are ready to start the implementation phase. The first task of the implementation is to provide some data samples for testing purposes and to confirm that your assumptions in the design are correctly mapped.

Objectives:

1. Data Preparation for implementation testing and debugging
2. Test design assumption
3. Opportunity to modify design schema accordingly

Tasks:

1. **Data Preparation:** Load some initial data (dummy data) into the database tables that you created during Task 2 of the project (Analysis and Design Phase) of the assignment.
 - a. Please provide at least seven rows in each table. The data needs to maintain data integrity constraints and a valid state for the DB.
 - b. Using MySQL, please write a script with insert statements to load all the data into your schema tables. The insert statement should be in the correct order that will not break the integrity constraints.
 - c. For each table, export the inserted data into a separate .csv file, and that it easily be reloaded during debugging.
2. **Design adjustment if needed:**

As you work on creating your testing data sample, you might have cases that the initial submitted design does not capture it correctly, or it is missing from the logical schema. Now is the chance to modify your design model and your logical schema to show the correct system requirements and the team design assumptions to work as expected.

Please, apply any adjustments to your EER and your Logical schema if needed.

What to submit

1. The row data loading script (.sql file)
2. The final .csv files for all tables data and that the team will use for the application testing in the next steps
3. Modified EER model and Modified Schema script if any changes made after the initial submission
4. A team reflection to share how the team managed this task together, upload as .doc, or .pdf file

Grading:

Here are the evaluation criteria:

1. All requested files are submitted.
2. Insert statements are in the correct order for the data dependency.
3. The table's data are uploaded correctly to the database schema, and it should maintain data integrity constraints and provide a valid data state to your system.
4. The inserted data works with the modified schema if there is a new one that is submitted by the team. If no modification, the initial schema shared during the last submission is going to be used for data insertion testing instead.