

## Project

### Purpose

To analyze the requirements, design, implement, document and test a database application for Jupiter College. The User Requirements of the database application are given in **Appendix A** below.

### Materials to submit

You will be asked to submit reports at various milestones as major phases of the project are completed. Further, you will also need to demonstrate the database application you designed and developed. Details about the milestones of the project, the phases they include and their due dates will be posted on the class webpage.

### Project Report/Phases

The project consists of the following phases. In addition, you will develop and maintain a Project Report. Your Project report should consist of a section for each phase, as well as an executive summary, introduction, and conclusion.

A. Analysis of the requirements of the project and a high-level description of the tasks involved.

B. Conceptual Design.

This stage involves the following

- Develop an Entity-Relationship model detailing the relations involved.
- Identify the attributes of the entities and the relations along with the primary key for each entity.
- List the constraints for each relation and entity.
- You should be able to explain the reasons for the particular design approach you have chosen.

C. Logical Design

This stage involves the mapping of your conceptual design above onto the relational data model. In this stage, you will design the tables for all your entities and relations. You should apply all normalizations you find useful and/or necessary. Ensure that your design still satisfies the user requirements. Justify your design choices.

D. Physical Design

This stage involves the following:

- Design an MySQL database based on the design developed in section B.
- Implement SQL tables for the relations and the constraints. Maintain scripts for the creation and deletion of tables.
- Maintain scripts for loading data into your tables.
- Design the user interfaces for your application.
- Ensure that your design still satisfies the user requirements.
- Justify your design choices.

## E. Prototype, Development, and Testing.

This consists of the following:

- Write Python code to access, update and administer the SQL tables made.
- Develop user interfaces, using Python, which satisfy all functional user requirements.
- Create indices for the database application. Justify the reasons.

## F. Make a user's guide for the database application.

**Project Demonstration**

Populate the tables with data for

- At least five courses offerings, three courses, and two degree programs
- At least five faculty, three departments, and ten students
- At least fifteen enrollments, and eight course grades

The demonstration should include all the functional user requirements.

**Miscellaneous**

The final project report should document all the activities with appropriate E-R diagrams, relation schema, etc. It should also give a list of the limitations of the application and give possibilities for improvement.

Features and functions other than specified in the document can also be added but should be documented clearly and demonstrated as well.

## **Appendix A**

### **Jupiter College User Requirements**

Jupiter College (aka Jupiter) is interested in developing a database application to help it in managing graduate admissions. Jupiter is seeking to manage the information for applicants to its graduate degree programs.

#### **System Scope**

The users of the application are the employees of Jupiter. It is assumed that all users have network computers capable of running Web browsers and Java applications.

#### **Data Requirements**

##### **Applicants**

For each applicant the following information is maintained: unique student id, name, contact info [address, phone number(s), email address], gender, date of birth.

##### **Applications**

Each applicant submits an application to a graduate degree program. The application contains the following information: degree program and semester/year applying for, GRE scores (verbal, quantitative, and analytical) and date GRE test was taken, prior education (college/university, major, degree (BS, BA, MS, MA), graduation date, and GPA), statement of purpose (essay), and the emails of three references. Each application also contains the applicant's answers to the program—specific admission requirement questions (see below).

##### **Graduate Degree Programs**

Each graduate degree program has a unique name, department offering the program, director (which is a faculty member of that department), phone number and email to contact for assistance/questions.

##### **Admissions Requirements**

Each graduate program has a list of additional requirements for its applicants. Each such requirement is expressed as a question to each applicant, together with possible answers.

##### **Admission Rubrics**

Each degree program has a rubric for evaluating prospective graduate students. The rubric contains information with the criteria used to evaluate applicants (eg, undergraduate education, prerequisite knowledge, letters of recommendation, standardized scores, statement of purpose), and possible scores for each criterion together with a short description of its meaning.

##### **Application Evaluation**

Each application is evaluated at least two professors of sought graduate degree program. The evaluation consists of the scores assigned to the application by the reviewers based on the admissions rubric for the program. For each application, there is a final decision on whether to admit/deny admission for the applicant to the requested program. The evaluations and final decision are timestamped with the date and time they were made.

#### **Functional Requirements**

Enter, update, and delete the details of

- Applicants
- Applications
- Graduate Degree Programs
- Admissions Requirements
- Admissions Rubrics
- Application Evaluations

Queries and Reports

1. List the names of all applicants for a given degree program and period (semester/year).
2. Find the total number of applicants by program and period.
3. Find the most “popular” major among the applicants in a given year.
4. Find those applicants with the lowest GPA that have been admitted in the current period.
5. Find the number of students applying to a degree program, for each degree program by prior degree.
6. Find the applications that have not been evaluated by anyone till a certain date.
7. Find the number of applications submitted/accepted/denied by program and year.
8. Find the emails of those references that resulted in the most acceptance decisions.
9. Find the minimum, maximum, average, and standard deviation of the GRE scores of the applicants, by program and application year.
10. Find the college(s) attended by most applicants over the last 5 years.