**Prestige University Student Course Registration ER Diagram Design Report**

Hang Yang

SYS 5013: System Engineering Analysis

Prof. Hitam Zidan

Oct 06, 2024

**Introduction**

To complete the assignment, we've built an ER diagram with four main entities: Student, Program, Course, Faculty. This ER diagram for Prestige University's enrollment system is designed to capture essential data related to students, courses, faculty, and programs, along with their interrelationships. The design of ER diagram is to help the Prestige University registration management team to maintain easy data retrieving and manipulation, as well as data consistency, integrity, and accountability.

**Overview of the ER Diagram**

The ER diagram (refer to **Figure 1**) illustrates the primary entities within the student course registration system and their interrelationships. Key relationships include the requirement for each student to enroll in a specific program, ensuring that all students are associated with a defined academic path. Additionally, the diagram specifies that each course must be assigned a designated faculty member, establishing a clear link between courses and instructors. Below are the specifications:

**Figure 1**

*ER Diagram of Student Course Registration*

A diagram of a company

Description automatically generated

*Note*. picture by Hang Yang in 2024. Own work.

**Entity Descriptions**

* **STUDENT** entity holds the personal details of each student, including basic attributes like StudentID, FullName, Date\_Of\_Birth, Email, and composite attribute like Address (with sub-attributes, like Street, City, State, and ZipCode). The StudentID serves as the primary key, ensuring each student record is unique. Sometimes, email attribute may also be seen as an unique filter when retrieving a student information.
* **PROGRAM** entity represents the academic programs available at the university, uniquely identified by a Program\_Number. A student must enroll in one program as a credited student and to complete a program and achieved related degree, the student must complete all the required credits for the specific program.
* **COURSE** entity includes a primary key attribute CourseID. It connects to both students and faculty, playing a central role in the enrollment process.
* **FACULTY** entity stores information like Name, Email, StartDate, Salary about faculty member. The title attribute of a faculty member can be a multivalued attribute with university-specific titles.
* **PREREQUISITES** entity represents the prerequisites structure between courses, and it allows for recursive relationships, where one course can only be registered if the student has taken all its required prerequisites courses.

**Relationship Descriptions**

The ER diagram highlights several key relationships within the student course registration system. There is a 1-to-many relationship for the Program and Student entities. It indicates each program can include many students, but one student can only be enrolled in one program. The Student and Course entities are connected through a many-to-many relationship. This means that the students can enroll in multiple courses, and in the meanwhile, the courses can accommodate numerous students. Also, the relationship between Course entity and Faculty entity via the Teach relationship, showing that each course is taught by a specific faculty member, while a faculty member can teach multiple courses. Specially, a recursive relationship is present in the form of Prerequisites, which links courses to other courses required as prerequisites, thus establishing a layered structure for course progression.

**Design Highlights**

In designing this system, maintaining data integrity, and reducing redundancy were primary considerations. Each entity has a clear purpose, with attributes defined to capture essential details, such as StudentID, CourseID, and FacultyID for unique identification. Relationships were chosen based on logical connections between entities to ensure efficient data management. The use of many-to-many relationships via associative entities like Register enables scalable student-course assignments, while recursive relationships, such as Prerequisites, accommodate complex course requirements. The system’s design is intended to facilitate easy querying, supporting quick access to student profiles, their enrolled courses, and faculty teaching assignments, while preserving the flexibility to manage course prerequisites effectively.

**Conclusion**

This ER diagram for Prestige University is designed to handle core requirements in student enrollment, course management, and faculty assignment, with flexibility for future scalability. The use of a variety of relationships, including one-to-many, many-to-many, and recursive relationships, supports robust data management while maintaining clarity and efficiency. The design ensures that the enrollment system can accommodate a growing student body and provide accurate, accessible information for administrative and educational purposes.

**References**

Neso Academy. (2021, June 02). Database Design Process [Video]. Youtube. <https://www.youtube.com/watch?v=7m6gXeMDaHc&list=PLBlnK6fEyqRi_CUQ-FXxgzKQ1dwr_ZJWZ&index=12>