**DATABASE PROJECT REPORT**

**University Management System**

**Members of Group “t.a.y”:**

**Yousuf Moiz Ali (Group Leader)**

**Muhammad Talha Shahab**

**Arsalan Ahmed**

1. **INTRODUCTION**

We have chosen to implement a university management system for our database project. Universities are very large and complex organizational structures with many employees and many different sorts of operations to be executed. University management system will be used to create, manage and perform all the activities related to the database of a given university. The database consists of information about the university and all of its associated people, including the students, staff and faculty. Starting from registration of a new student in the college, it maintains all the details regarding the attendance and marks of the students. For faculty, we also maintain important details like the courses being taught, the status of the faculty and the salary being offered.

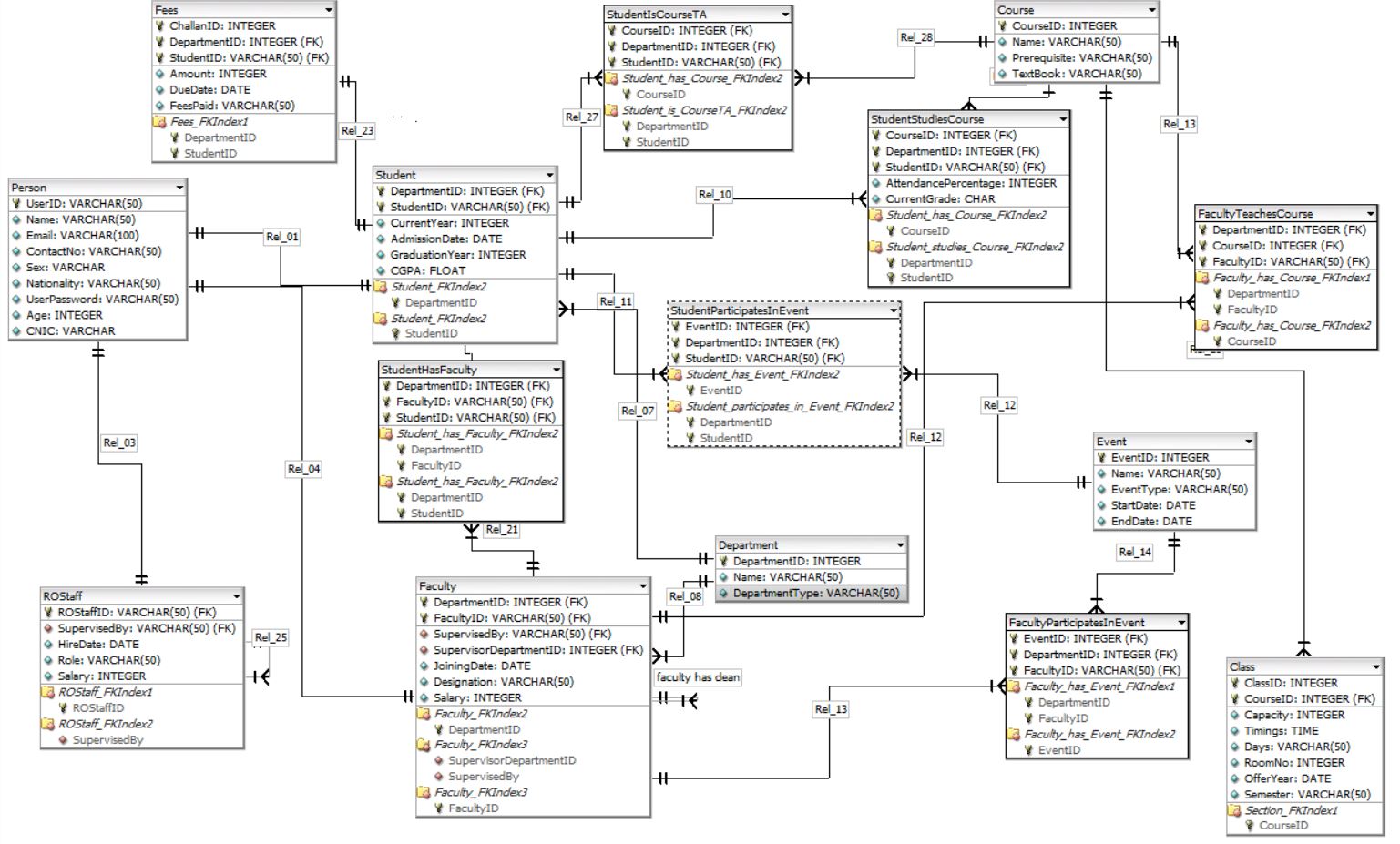
1. **BUSINESS SCENARIO**
2. **MODULES IMPLEMENTED**

We have implemented many different functionalities in our project which were slightly discussed above. For students, we have implemented the options of showing all the university related activities of a student including g=his basic information, courses he is taking, his course history and their associated grades and his financial record. On the metacurricular side, his participation in events is also shown and the option for him to enroll in another event is also provided.

For faculty, we also have some basic information like their department name, the courses they are teaching and have taught, and their salary. Moreover, for each course we have taught, we have computed some logistics for those courses like the number of students in a particular offering and the average grade.

For staff, we have also implemented a similar scheme. Furthermore, the most important feature of our project is the option to add and remove different object with ease. For example, we can add courses, events, faculty members or students after filling out the required information in the Windows forms. We also have the facility of deleting students, faculty and events. There is no delete course button since we assume that if a course will not be offered in any particular semester, it will not be in the corresponding course catalogue.

1. **ERD**



The picture attached above is of the final ERD. Over time we have made many changes but most of these were technical rather than structural. For example, we changed the attributes of some types of variables so that they would take up less space and others were changed so that the types were such that they would be more understandable to the user and would parallel the existing university management system at Habib better. In some tables, we adjusted the auto increment property of the primary keys so for example StudentID is supposed to have the format of the initials of his first and last name plus a 5-digit number so it is not auto incremented. Initially, we had planned to incorporate all staff but due to some ambiguities in their actual roles, we reverted to just including the RO staff. Another feature to be discussed here is that the days of the Class entity will be entered so that they contain the initials of the days on which the pertinent class will be held. For example, MWR means classes will be held on Monday, Wednesday and Thursday.

1. **GUI**

The user interface is implemented using simple Windows forms. Initially, we had created a different form for each of students, faculty and staff but after the instructor’s suggestion that the users should have to make minimal clicks, we implemented a dashboard approach so that all the functionalities pertaining to a certain group are available in a single form. Thus, all the forms have its related queries linked to it on the click of a button. For example, the “Add Course” button has an insert query on the back end linked to the relevant table(s).

1. TEST CASES CHECKED