User Manual and Design Manual for University Database Application

CS 460 Final Project

Matthew Bell
Matthew Huestis
Alif Jakir
Conor Miller-Lynch

Table of Contents

User Manual

Introduction

  The University Database Application allows users to access information in a university database. Users are categorized as university administrators, professors, or students, and are granted different levels of permissions to access relevant information.

Starting the Server

To start the server, navigate to the project directory and run the following command:

`python3 Univ_application/manage.py runserver`

Logging In

  When the application starts, users will be prompted to enter a username and password to access the appropriate page for their user type. Click "Log Out" at any time to return to the login screen.

Features By Group

  University Administrator

- Instructor Index: View a list of professors sorted by name, by department, or by salary.

- Department Index: View a list of departments. Click on a department for more information.

- Course Index: View the list of all courses.

- Professor Performance search: Given a professor's name, an academic year, and a semester, view the following for the professor: the list of sections taught during the semester and the number of students taught in each section, the all-time total dollar amount of funding through grants, and the all-time list of publications for which the professor is an author.

USERNAME: ADMINISTRATION

Instructor Index

Department Index

Course Index

Professor Performance search

Professor

- View the list of course sections and the number of students enrolled in each section that the professor taught in a given semester.

- View the list of students enrolled in a course section taught by the professor in a given semester.

USERNAME: TEST_INSTRUCTOR

Professor Name: [   ]

Semester: [Fall ▾]

[Search]

Course Name: [   ]

Section: [01 ▾]

Semester: [Fall ▾]

Year: [   ]

[Search]

Student

- View the list of sections offered by a department in a given year and semester.

USERNAME: TEST_STUDENT

Department: [   ]

Semester: [Fall ▾]

Year: [   ]

[Search]

Administration

Users can be added and removed using the Django administration system by going to the /admin route and logging in as an IT administrator.

Design Manual

Overview

The University Database Application stores data that may be of interest to university administrators, instructors, and students. The entities are **department**, **instructor**, **student**, **course**, **section**, **paper**, and **fund_grants**. Many-to-many relationships between entities are implemented with the tables **takes**, **teaches**, and **pre_req**.

Instructor information is stored in the **instructor** table. The table **teaches** records which instructors taught which sections during each year and semester. The table **fund_grant** records the grants given to each instructor. **fund_grant** was chosen as the name for this table since "grant" is a keyword in MySQL. The table **papers** records information about publications.

The table **course** records information shared between all instances of a course at the university. The table **pre_req** records the prerequisites of courses that have them. The table **section** records the section number of the class along with the year and semester of the course.

The Table **student** records the information for each individual student. The table **takes** records the section and grade of the class a student has taken.

Tables that are used by Django to validate users and their information should be accessed through Django's administration system.

Schema / ER Diagram
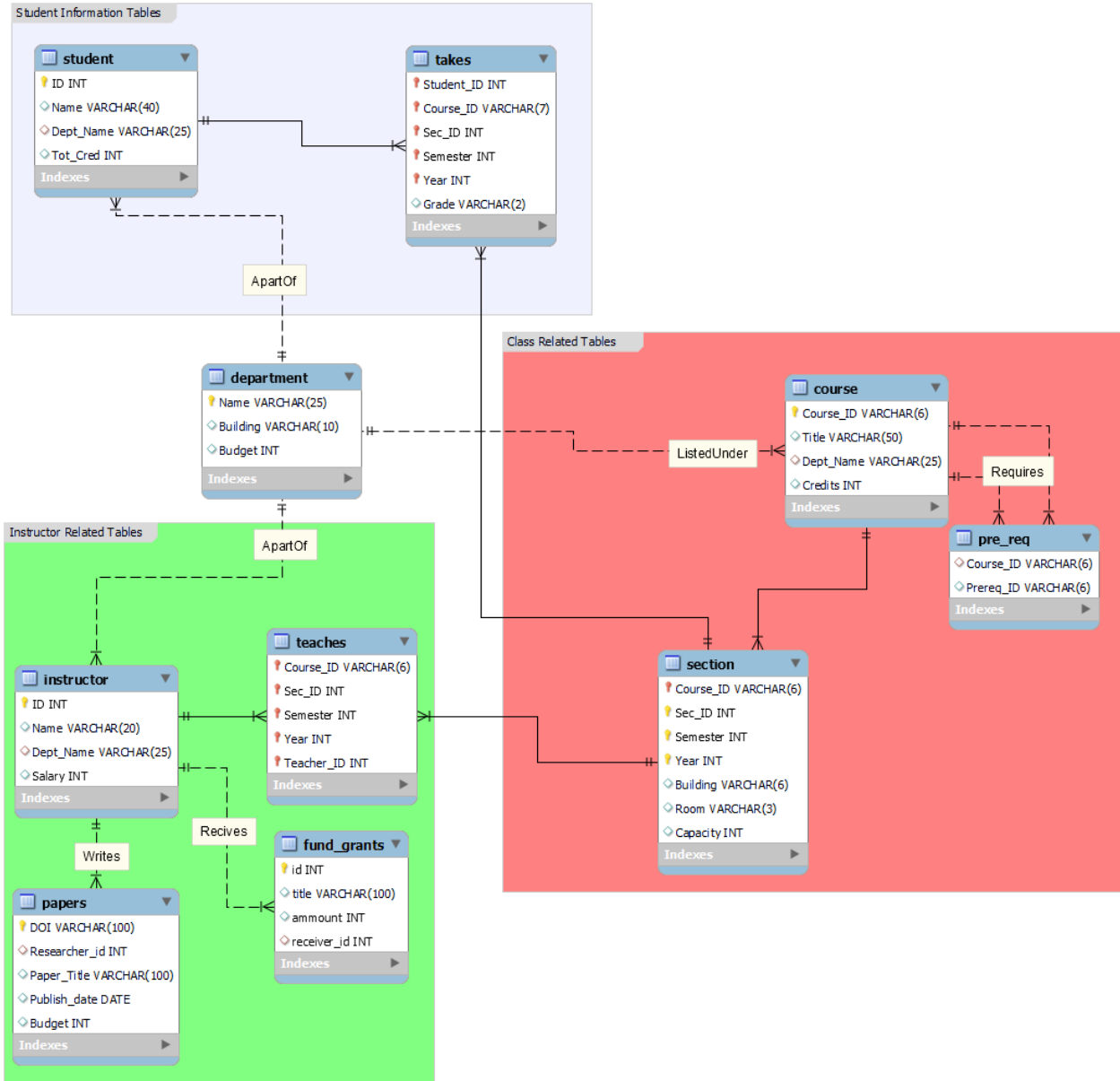
Security Measures

      The University Database Application implements several security measures to prevent unauthorized access. If a user is not logged in or tries to access a page they are not authorized to view, the server will redirect them to either the login screen or their home screen.

Modifying users and groups is restricted to the IT administrator, who can perform these actions by logging into the Django administration page. Most queries are executed through Django's query system. User input for queried data is obtained through drop-down select boxes and text input boxes. The drop-down inputs have pre-selected values and are not a security concern. Text inputs for professor performance and other pages only query specific tables and columns. Moreover, views that display and query the database are designed to show an error message in case of unexpected input.