CMS Login System
CST363: Intro to database Systems
Midterm Project

Mark Mariscal
Christopher Piwarski
Wais Robleh

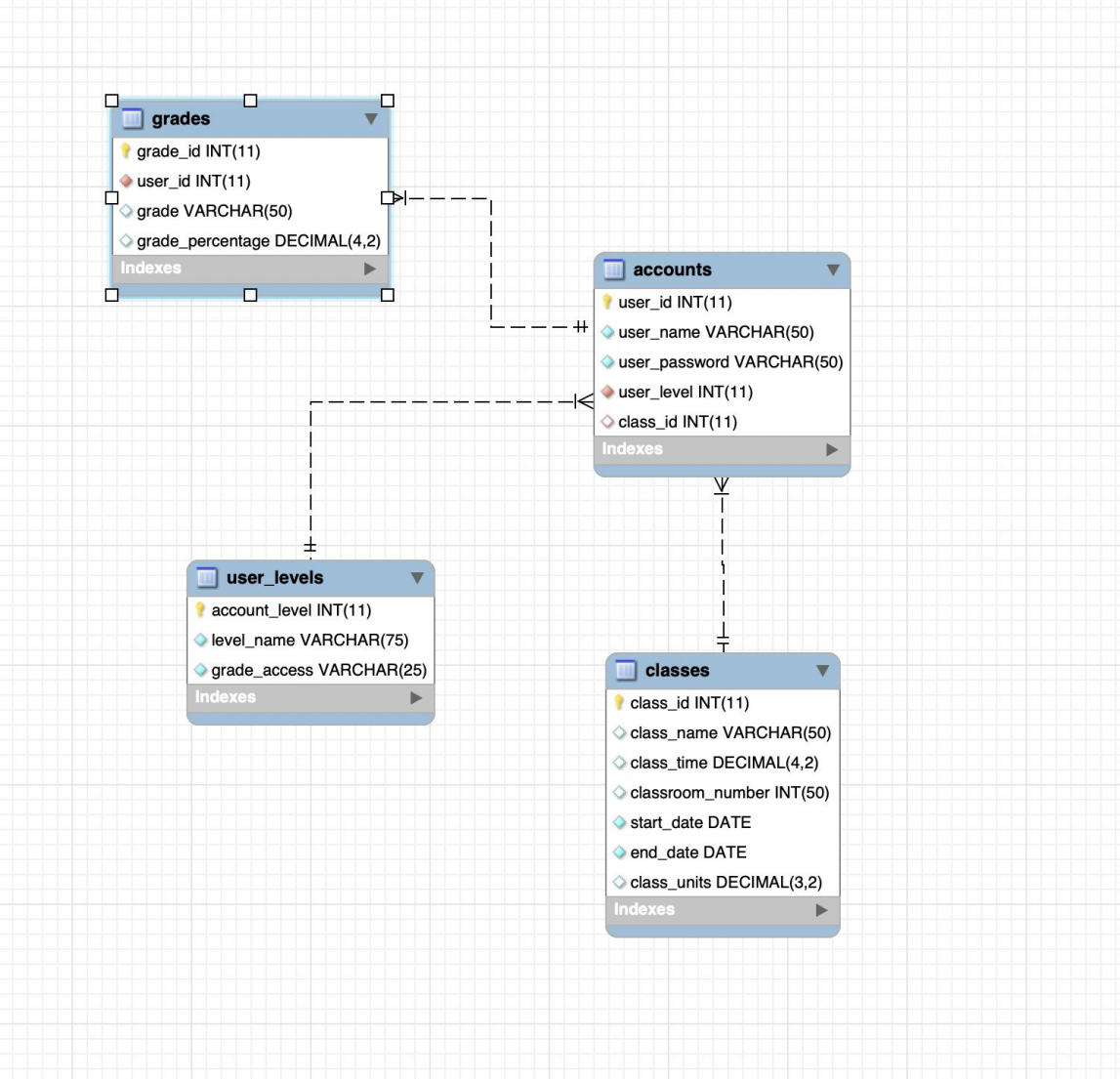
Midterm Project

For our midterm project we chose to make a course management system (CMS) similar to iLearn. The goal is to create a web application that users log into; based on their user level (student, teacher, administrator), they receive different informational views relevant to their needs. Students will receive access to their course list, and individual grade. Teachers will be able to view grades for all their students. The administrator will be able to create new accounts, but not view courses or grades. The project is centered around displaying different information based on account access levels. This type of project would provide ample application of database technology.

Our application performs several different database operations to provide the user with the before mentioned feature set. SQL queries are used to confirm user identity during login. Passwords are compared against usernames for authentication. For the three other features of our website, queries are used to determine the type of user, and what information they have access to. Once the user type is known, more queries are conducted to display to them the appropriate information. Cookies are implemented so the user does not have to re-authenticate during their session.

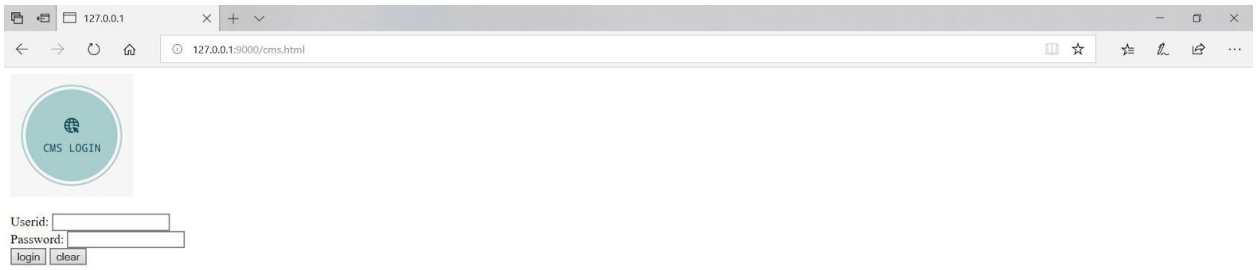
The database for this project features a table for account levels, classes, accounts, and grades. The primary and foreign key relationship will be used to connect all the data. An Entity-Relationship Diagram (ERD) is attached to this document to show an outline of our data structure. Also attached are images that display the various pages of our website. Constraints are generated during database creation to guarantee the foreign key relationship. All of the source files can be retrieved from our project github account at https://github.com/evilmurries/cst363_midterm.

One of the requirements for this project is that our database adheres to Third Normal Form (3NF). According to the course text, there are three criteria the database needs to conform to in order to be considered in 3NF. Each column should contain a unique scalar value. The columns for each table should only depend on the primary key. Referring to the ERD, one can see that all the column data is unique save for the foreign keys. Each account references another table in order to retrieve data specific to courses, grades, and user privileges.

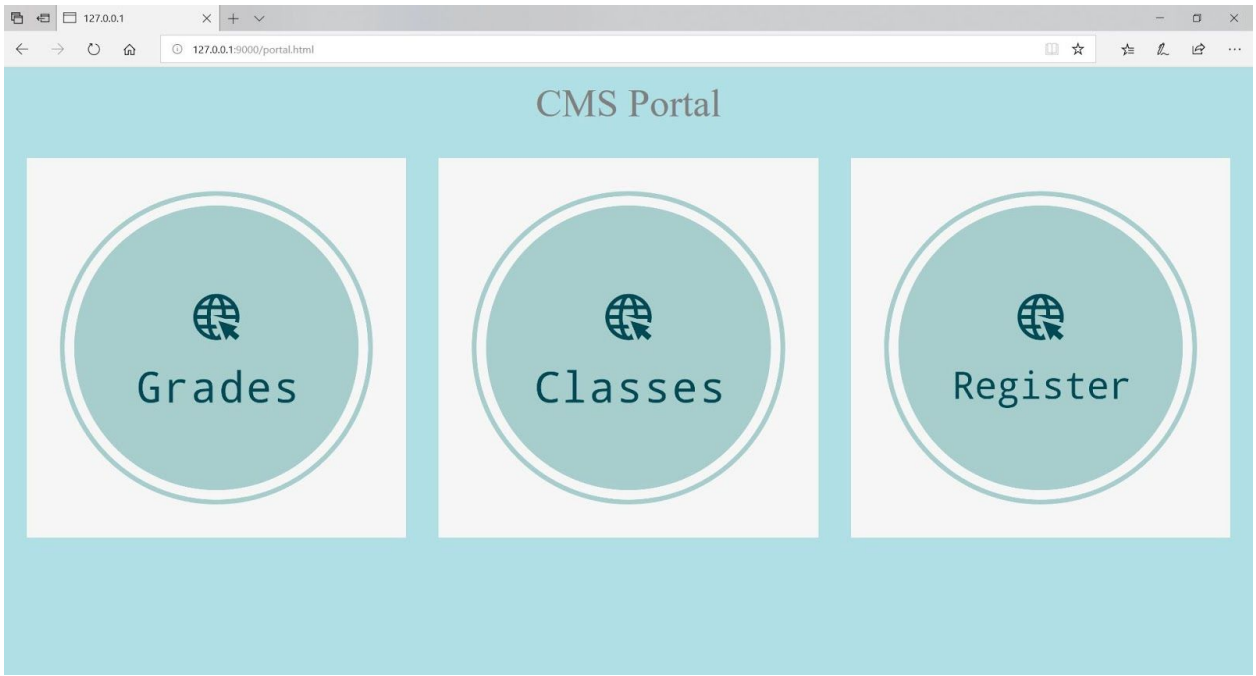


Entity-Relationship Diagram for our project.

We hope that you enjoy our project as much as we enjoyed creating it. It was an excellent opportunity to implement the lessons we have learned from the previous course modules. Some website features required practicing more advanced SQL skills such as subquery and complex joins. The project was also an opportunity to study and implement HTML and Python Common Gateway Interface (CGI) programming. Python3 was the tool that was used to implement the database interactions beyond the initial HTML homepage. HTML code was used to add style to the various pages as well as create display features such as tables. The project has provided experience that will prove invaluable in our future careers. Attached are screen captures that provide a walk through of our course site. Thank you.



The login home page - cms.html

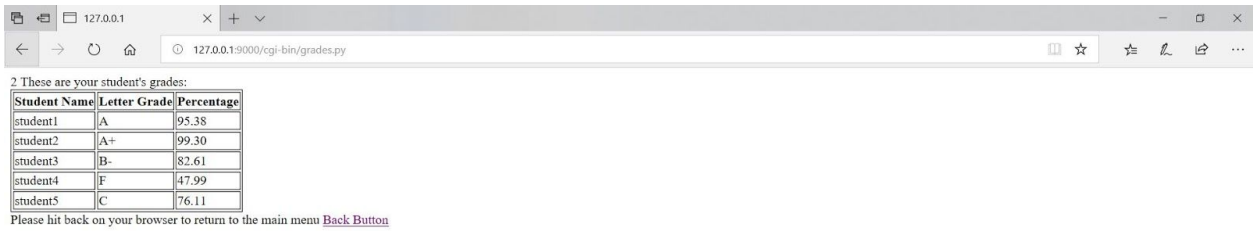


The portal page when the user logs in - login.py

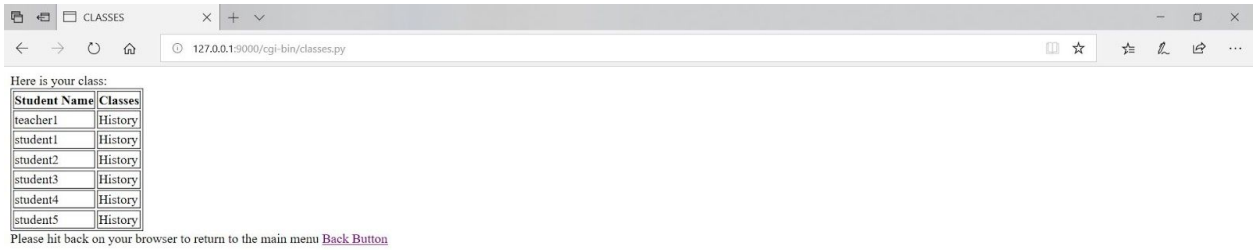
A grade printout for a student user - grades.py

A course printout for a student user - classes.py

A student attempting to register new users - register.py



A teacher viewing their student’s grades - grades.py



A course print out for a teacher user - classes.py

A teacher attempting to access registration - register.py

An admin attempting to view student’s grades - grades.py

An admin attempting to view student’s classes - classes.py

An admin attempting to register new users - register.py