

Design Document

CMPT 391 Project 1

Group Members

Karley Yachimec
Justin Thai
Aidan Mark
Jackson Steed
McKenzy Ison

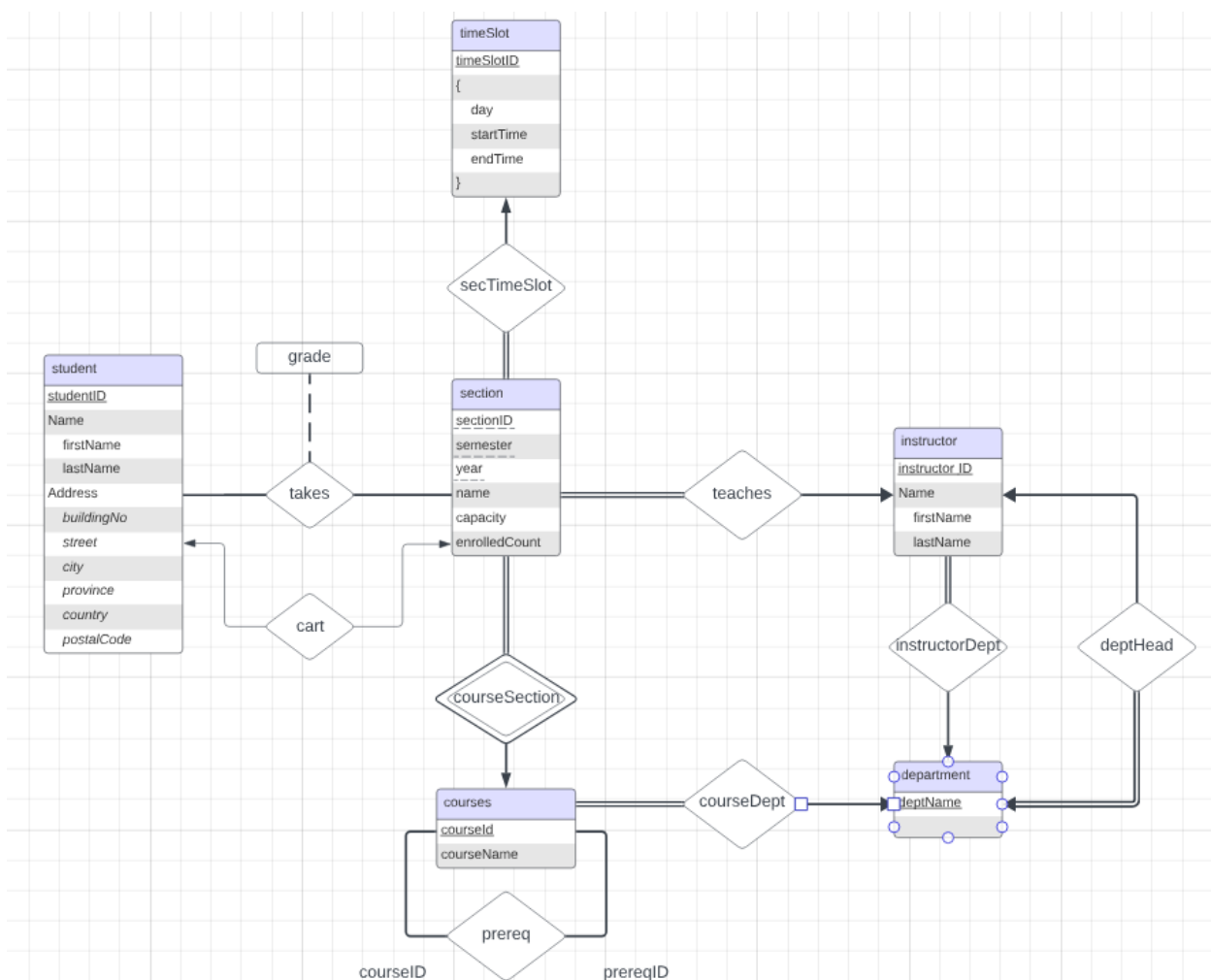
Front End

Visual Studio .NET Core 6.0
Windows Forms C# Project template

Back End

Microsoft SQL Server Management Studio
19.3.4.0 or equivalent
Database created with dbo schema

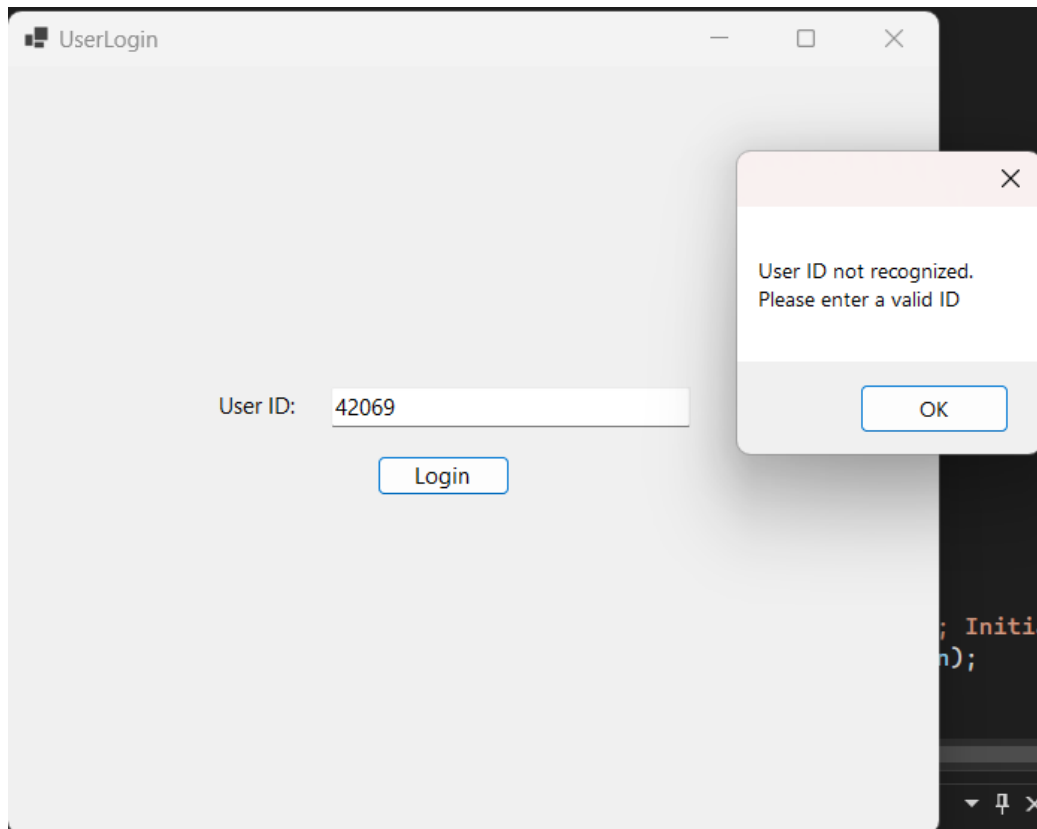
Entity-Relationship Diagram



Project Description

User Login

A student will log in by entering their student ID and clicking “Login”. The event will call the database connection and execute the stored procedure “checkStudentID” with the ID entered by the student. This stored procedure compares the parameter with all values in the student table. If the ID is not found within the table, the student is prompted to enter an ID that exists within the table. Otherwise, the user continues to the main screen.



Main Screen

The main screen contains a tabControl system to separate the 3 different processes: viewing currently enrolled courses (My Courses), searching for courses to enroll in (Search Courses), and a tab to formally enroll in courses (Shopping Cart).

My Courses

Form1

Fall 2024 Apply

My Courses Search Courses Shopping Cart

Course ID	Section ID	Semester	Year
CMPT 101	KD00	Fall	2024

View Course Details

This tab displays courses that the user is enrolled in for the selected semester and year in a listBox object. Using the studentID of the user, a stored procedure “showMyCourses” being a transaction to read the section information using the selected semester and year as parameters. A user can select an index from the list of courses and run a transaction to see more detailed course information by pressing the View More Information button. This transaction takes the semester, year, courseID, and sectionID and reads data from the course table, the section table, the timeSlot table, and the instructor table.

Search Courses

Form1

Semester Year

cmpt

CMPT 101	Introduction to Computing I AS01	1	50	Darren Tellier	Monday	08:00:00 08:50:00	Fall 2024
CMPT 101	Introduction to Computing I GO61	0	50	Dr. Indratmo Indratmo	Monday	10:00:00 11:20:00	Summer 2024
CMPT 101	Introduction to Computing I KD00	0	50	Dr. Indratmo Indratmo	Monday	08:00:00 08:50:00	Fall 2024
CMPT 101	Introduction to Computing I RJ47	0	50	Dr. Indratmo Indratmo	Monday	10:00:00 10:50:00	Spring 2024
CMPT 101	Introduction to Computing I VC27	0	50	Dr. Indratmo Indratmo	Monday	08:00:00 09:20:00	Winter 2024

A student can search for courses in this tab. Using the `viewAvailableCourses` transaction to find best matches from a user-inputted string, a list of relevant courses is read, the courses read are stored in a materialized view. Because much course information is gathered across many tables, we save query time by storing data this way.

The user is then able to select an index from the list of courses and click the Add to Cart button. A transaction `addToCart` is invoked, and the courses are compared to ensure that the user has no time slot conflicts.

Shopping Cart

A list of courses currently in the shopping cart is shown through the procedure `showMyCart`. A user can choose to enroll in courses that are in their shopping cart by confirming they would like to perform the action. First, confirming there is space in the course to accommodate the student and, then using a transaction stored procedure `enrollInSection` to add to the students table of courses and to remove them from the cart. If there is no space for the student, the transaction will rollback and the student will be able to try again.

Design Plans - Lab 3

A student needs to log in, select courses, and enroll. System needs to check if the student has prerequisites and check if there are time conflicts. If all requirements are met, a transaction must occur between system and student where all courses are enrolled. Database must be populated with thousands of records in order to test the capability of our system.

1. Create a login screen with username and password.
2. Create a course search screen with a search option and possibly filters.
3. Create a course selection screen after clicking on the desired course where a student can see additional info and click the "add to cart" button.
4. Create a cart screen where the student can see all courses in their cart and decide to enroll in the courses, or remove them from the cart.
5. Create a confirmation screen that lets the student know which courses were successfully enrolled.

Split tasks into two:

1. Allow a student to add courses to cart (and check for prerequisites and time conflicts)
 - a. Want to use a materialized view in this step because of how many tables we may need to join all the time
 2. Enroll in courses (take the courses from CART table and move them to TAKES)
 - a. This is a very simple task, all we need to do is move the 5 rows from one table to another and double check that there is space in the course
- Need a new table called cart that is unique to each student
 - courseID
 - sectionID
 - studentID

Future Plans

- Implement a confirmation when a student is able to enroll from the shopping cart.
- Add Course Information to the Search tab.
- Add error checking to account for previously taken courses or currently enrolled courses.
- Fix database data to avoid “Ship of Theseus” issue.
- Clearer communication when one of us is working on something.
- Making sure forks are up-to-date before we begin working to avoid hefty conflicts
- Implement materialized views at the database level for course information
- Confirm no conflict with class size when enrolling from shopping cart
- Add lock to capacity check to ensure multiple people do not see availability from cart

Task Distribution

- Populate tables with data scripts
- Search query stored procedure
 - Check time conflicts when adding to the cart
 - Check prerequisites
- Login page interface design
- Login query stored procedure
 - Use studentID to login and store for use
- Enrollment transaction in Shopping Cart
 - Stored procedure for capacity/count
- Interface design
- Stored procedure for My Courses

Tabs

- My Courses
 - A list of all currently enrolled courses
 - A drop option for a selected course
- Search
 - A list of courses that match a search
 - Connect to shopping cart with a button
 - Add selected courses to shopping cart
 - Check time conflicts
 - Check prerequisites
- Shopping cart
 - A list of courses selected and added to cart
 - Enroll/cancel option
 - When enroll add to My Courses by adding to takes and setting active to yes

Notes

All lists of courses should have a minimum amount of information:

Course Name
Course Section
Instructor
Time

Make a view when searching all available courses. This limits the number of joins between tables. When adding to cart, check prerequisites and time conflicts. The cart will hold courses that a student wished to enroll in, and our transaction will be attempting to transfer those rows into takes from the cart.