Student Portal

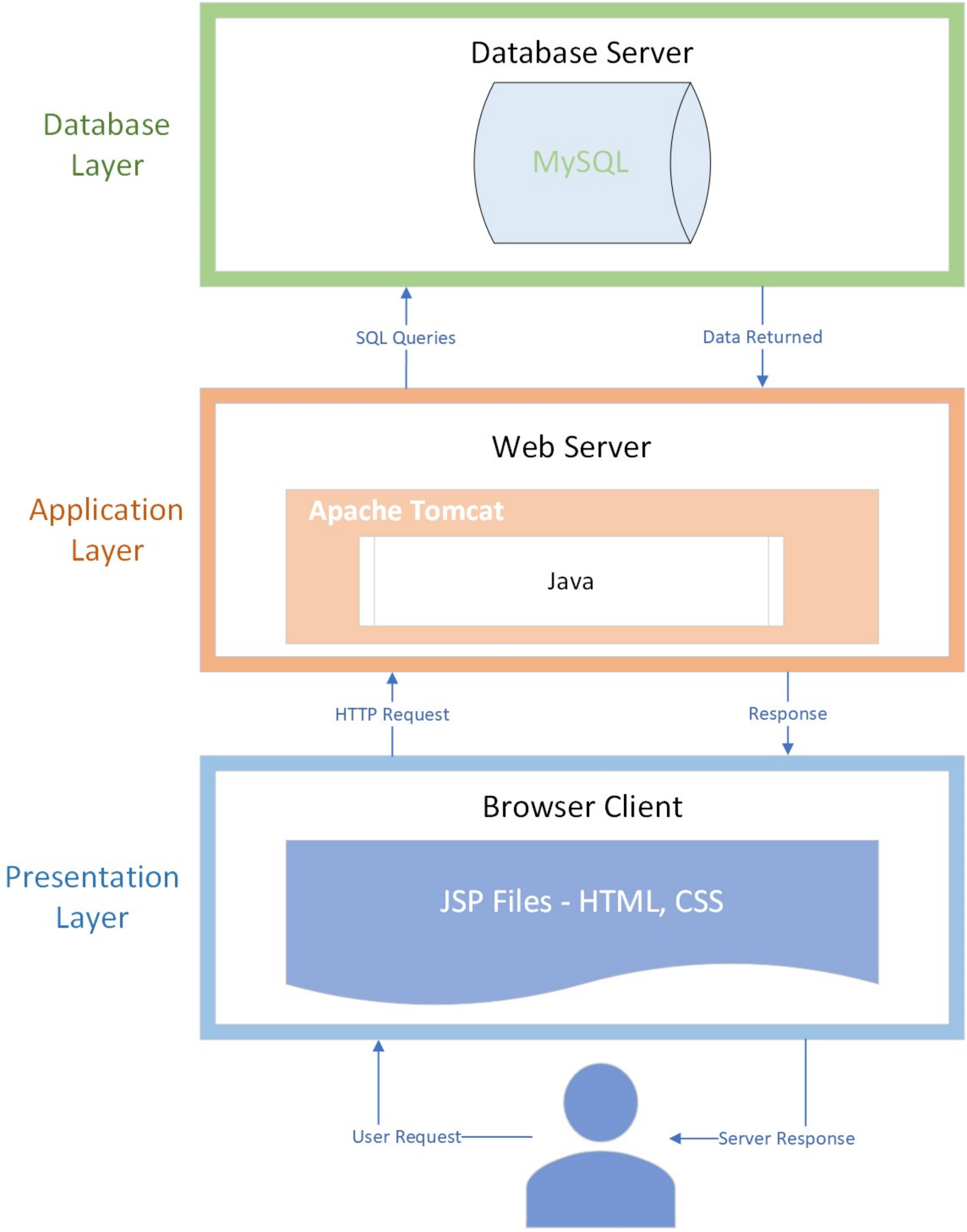
Mario Alkhoury, Youhao Chen

Team 5

Project Description:

This project will be a college database application that will manage campus activities. The primary goal will be to allow students to manage their classes and records. Students will be able to find classes according to their needs. They can choose to sign up and enroll in classes, drop classes, or simply see the class description. They can use this information to help plan out their college roadmap for their desired degree. The classes they look up will be input by the administrators, who have special admin account privileges enabling them to add classes to the list of available classes, specify their details (department, course number, CRN, time, instructor, mode, etc) as well as remove them. The Graphical User Interface (GUI) will be simple, elegant, and straightforward to navigate providing the students and admins alike with an intuitive interface with as few distractions as possible. The motivation for this project is to make a modern, high-quality management system that solves the pain points and limitations of current management systems and reduces costs for universities and community colleges alike. The stakeholders for this project will be the students who will manage their class schedules, universities that want to upgrade their current systems, Dr. Wu as the class instructor, and Mario, and Youhao as developers of the system. The application domain is the education and academic sector as the system aims to simplify college class management. Similar to the motivation, the goal of the project is to provide a hassle-free experience for students and universities alike to reduce the time overhead, improve user experience, and avoid errors and mistakes.

System Environment:



Hardware/Software:

Random Laptops running Windows 10, MySQL Workbench 8.0 CE

HTTP Server:

Apache Tomcat 9.0.64

RDBMS:

MySQL Server 8.0.29

Languages:

Java, SQL, HTML, CSS, Javascript

Functional Requirements:

This application is designed to be used by different users: Administrators, Instructors, and primarily Students. Administrators can add/remove other users and add/remove/modify classes. Instructors can submit grades for students in their class as well as drop them. Finally, students are able to search for classes based on what they still need for their major or by department, as well as add and drop classes for themselves.

Functions:

Login:

- Administrators, students, and instructors will have to login to be able to securely access their account.

Modify Profile:

- Students, instructors, and admins will be able to make changes to their profile and contact information.

Add User:

- Administrators will be able to add other students, instructors, and admins.

Delete User:

- Administrators will be able to delete the accounts of students and instructors.

Add Class:

- Administrators will be able to add classes to the database.
- Administrators are only able to add classes that are available from the list of courses in the university catalog.
- The system will check to make sure that there are no time conflicts with a given instructor.

Modify Class:

- Administrators will also be able to make changes to existing classes.
- Same checks as Add Class

Delete Class:

- Administrators will also be able to delete existing classes

Add Course

- Administrators can add courses to the catalog of courses.

Add Department

- Administrators are also able to add new departments.

Look up Classes:

- Students will be able to search the list of available classes (to be able to select the ones they need to graduate)

Save Classes

- After a student finds a class they want, they can save that class into a “shopping basket” for later registration.
- Students can add or remove classes in this basket.

Display Class Information

- A student will be able to select a class to get detailed information about a class after using the “Look up Classes” function.
- The class display will open in a new tab or popup to avoid having to find classes all over again.

Register for Classes:

- Students can add classes by CRN or a list of classes from “Save Classes”
- System makes sure there are no time conflicts between classes

Display Class Schedule

- Students will be able to see a list of their current classes for the semester.

Grade section:

- Students also have the ability to view their grades

Comment section:

- Students can leave a comment to detail anything they want about the class to the administration.

Drop Classes:

- Instructors are able to drop students from their class

- Students can also drop the classes that they are enrolled in

Submit Grades:

- Instructors have the ability to submit grades for each student in their class.

Non-Functional Issues:

Graphical User Interface:

The GUI will be a relatively simple HTML webpage with clearly defined links to college related activities. The actual links shown are decided by the user's status as an Administrator/Student/Instructor.

The student page will have a list of functions that a user might browse or use. Put simply, all the functions mentioned above in the functional requirements will be a button on the student's homepage. Clicking any of them will redirect the student to a different page according to what function the user clicked. To go back to the homepage and choose a different function, the user can simply click the back button in the top left corner of the page.

Administrators will have access to static class and user forms. They will also have pages for the managing and removal of classes and users. Finally, they will also be able to add catalog courses and departments.

Instructors will have a form allowing them to submit final grades and another form allowing them to drop students in their class.

Security:

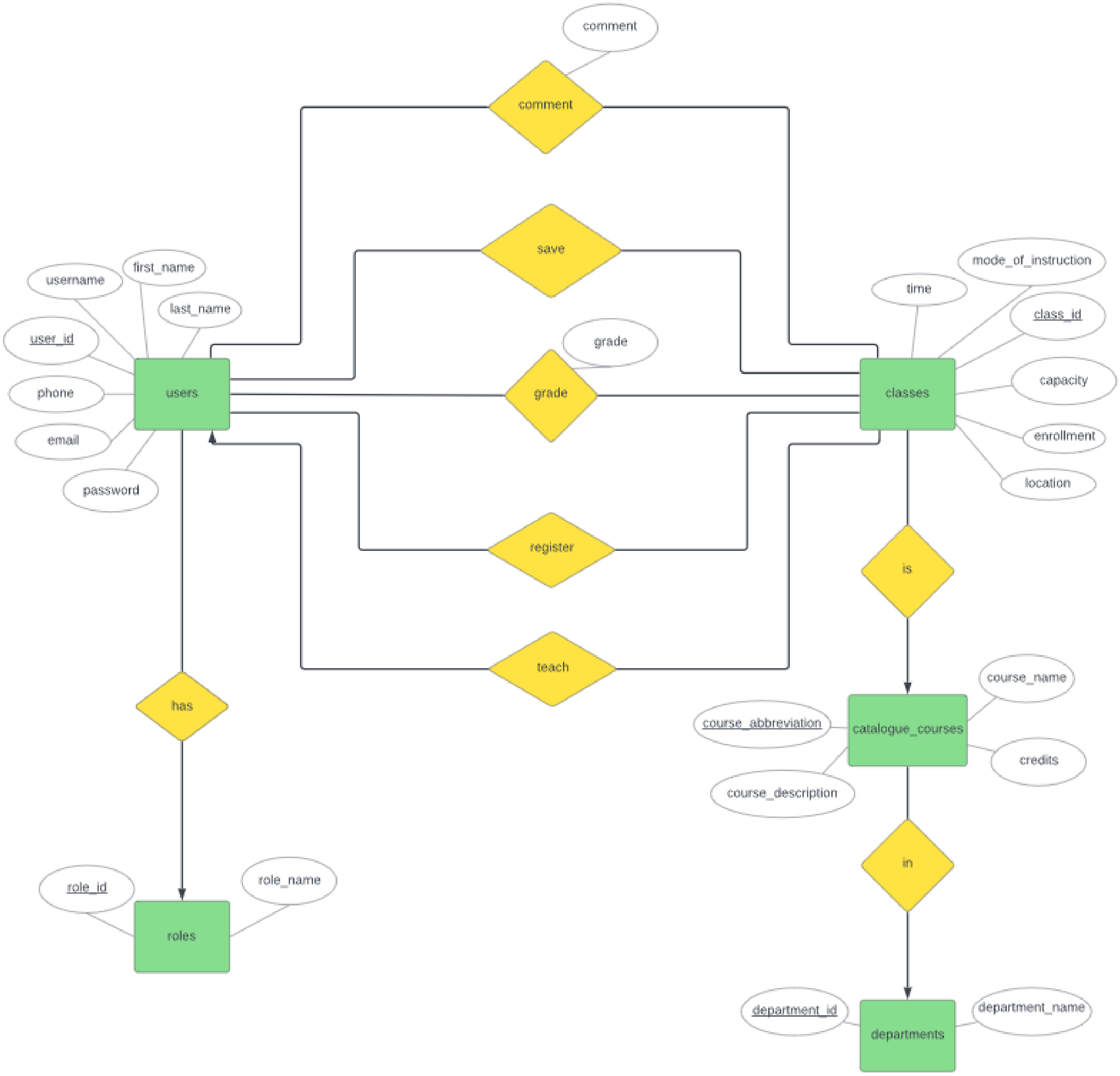
All users of the application will have to use a unique username and a password. This is done for security as well as privacy. User accounts protected by passwords will prevent attackers from gaining access to user accounts and profile information.

Access Control:

Users are divided into Administrators/Instructors/Students, each has access to different actions.

- Administrators can do everything from creating users to modifying classes. However, Administrators cannot remove other Administrators.
- Instructors can submit grades for students in their classes as well as drop them from their classes. Instructors cannot add, drop, or save classes.
- Students have the ability to add classes, drop classes, and save classes.

ERD diagram:


comment
comment
save
grade
grade
register
teach
username
first_name
last_name
user_id
phone
email
password
users
classes
time
mode_of_instruction
class_id
capacity
enrollment
location
is
has
catalogue_courses
course_name
course_abbreviation
credits
course_description
in
role_id
role_name
roles
department_id
department_name
departments

Entity Set and Relationship:

Primary Key - PK
Foreign Key - FK

1. **User entity**

   The user entity set is responsible for keeping a list of the different users in our database. The PK is user_id.

   User entity relationships:

   - A user "**has**" one and only one role and its PK is role_id. Many users can have the same role. We set a many-to-one relationship between **users** and **roles**.
   - A user can "**comment**" on many classes, and a class can have comments from many users. This is why we set a many-to-many relationship between the **user** and the **class** table, which has class_id as its PK.
   - A user can "**save**" on many classes, and a class can have comments from many users. This is why we set a many-to-many relationship between the **user** and the **class** table, which has class_id as its PK.
   - A user can get a "**grade**" for many classes, and a class can have grades for many users. This is why we set a many-to-many relationship between the **user** and the **class** table, which has class_id as its PK.
   - A user can "**register**" for many classes, and a class can have many people registered for it. This is why we set a many-to-many relationship between the **user** and the **class** table, which has class_id as its PK.
   - A user can "**teach**" many classes, but a class can only be taught by 1 teacher. This is why we set a one-to-many relationship between the **user** and the **class** table, which has class_id as its PK.

2. **Roles entity**

   The role entity set contains 3 roles: student, instructor, administrator.

   Roles entity relationships:

   - A role can be "**had**" by many users (PK user_id), and many users can have the same role. This is why the relationship between **roles** and **user** is one-to-many.

3. **Classes entity**

The class entity has a list of all available classes for the courses offered.The PK is class_id.

Classes entity relationships:

- A catalogue_course has many sections/classes. (A class “**is**” a catalogue_course). So the relationship between **class** and **catalogue_course** is many-to-one.
- Users have “**grades**” from classes. There may be many students getting grades from a class, so the relationship between **class** and **user** is many-to-many.
- Users “**register**” for classes. There may be many students registered for a class, a student could also register many classes, so there’s a many-to-many relationship between the **class** and **user** tables.
- A class can be “**saved**” by many users, a user can also save many classes, so the relationship between **class** and **users** is many-to-many.
- A user can “**comment**” on many classes, and a class may be commented on by many students. Therefore, the relationship between **class** and **users** is many-to-many.
- A user can “**teach**” many classes, but a class can only be taught by 1 teacher. This is why we set a many-to-one relationship between **class** and **user**.

4. **Catalogue_courses entity**

The Catalogue_courses entity contains the list of courses along with their description. Its PK is course_abbreviation.

Courses entity relationships:

- A catalogue_course can have many classes (PK class_id), but several classes can have only 1 catalogue_course (A class “**is**” a catalogue_course). This is why the relationship between **catalogue_courses** and **classes** is one-to-many.
- Many catalogue_courses can be “**in**” 1 department (PK department_id), but multiple departments cannot contain the same catalogue_course. This is why the relationship between **catalogue_courses** and **departments** is many-to-one.

5. **Departments entity**

The departments entity contains the list of all departments in the university. Its PK is department_id.

Departments entity relationships:

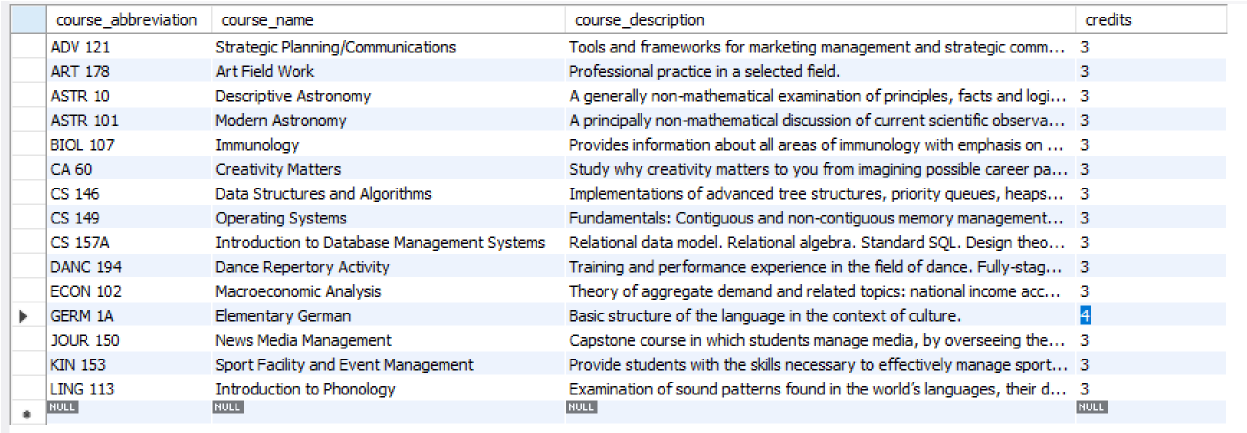
- A department can have many catalogue_course (PK course_abbreviation), but a catalogue_course is “**in**” only 1 department. This is why the relationship between **departments** and **courses** is one-to-many.

ERD Schemas:

- catalogue_courses (**course_abbreviation**, course_name, course_description, credits)
- classes (**class_id**, enrollment, capacity, mode_of_instruction, location, time)
- comment (**user_id**, **class_id**, comment)
- departments (**department_id**, department_name)
- grade (**user_id**, **class_id**, grade)
- has(**user_id**, **role_id**)
- in(**course_abbreviation**, **department_id**)
- is(**class_id**, **course_abbreviation**)
- register (**user_id**, **class_id**)
- roles (**role_id**, role_name)
- save (**user_id**, **class_id**)
- teach (**user_id**, **class_id**)
- users (**user_id**, username, password, first_name, last_name, email, phone)

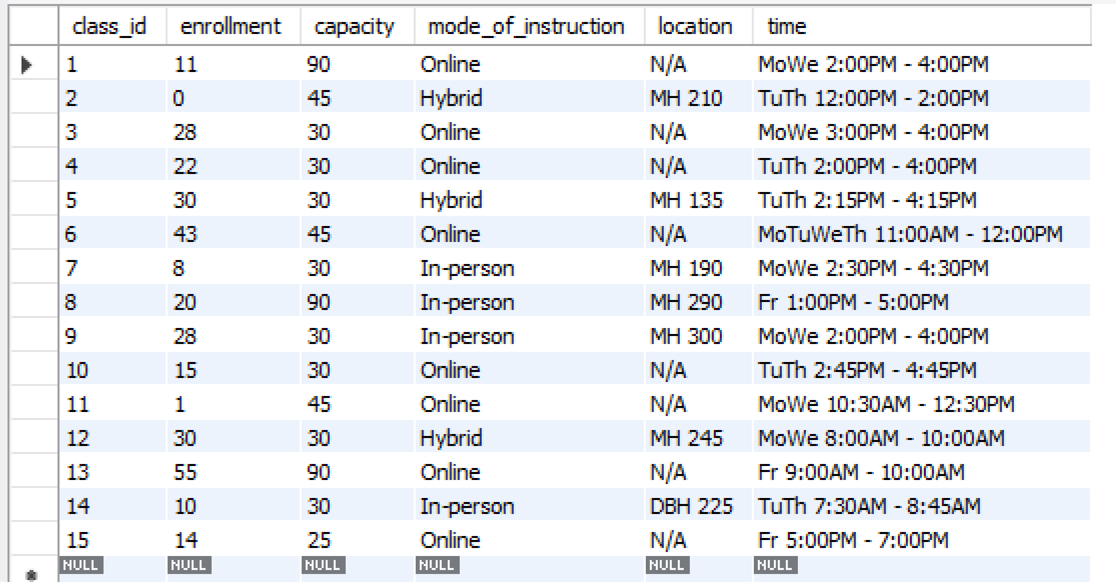
Workbench Screenshots:

Catalogue_courses:



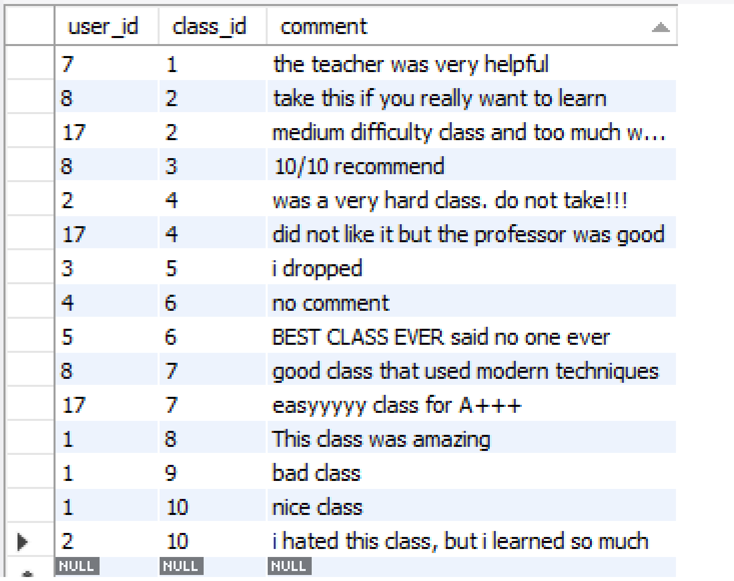
	course_abbreviation	course_name	course_description	credits
	ADV 121	Strategic Planning/Communications	Tools and frameworks for marketing management and strategic comm...	3
	ART 178	Art Field Work	Professional practice in a selected field.	3
	ASTR 10	Descriptive Astronomy	A generally non-mathematical examination of principles, facts and logi...	3
	ASTR 101	Modern Astronomy	A principally non-mathematical discussion of current scientific observa...	3
	BIOL 107	Immunology	Provides information about all areas of immunology with emphasis on ...	3
	CA 60	Creativity Matters	Study why creativity matters to you from imagining possible career pa...	3
	CS 146	Data Structures and Algorithms	Implementations of advanced tree structures, priority queues, heaps...	3
	CS 149	Operating Systems	Fundamentals: Contiguous and non-contiguous memory management...	3
	CS 157A	Introduction to Database Management Systems	Relational data model. Relational algebra. Standard SQL. Design theo...	3
	DANC 194	Dance Repertory Activity	Training and performance experience in the field of dance. Fully-stag...	3
	ECON 102	Macroeconomic Analysis	Theory of aggregate demand and related topics: national income acc...	3
▶	GERM 1A	Elementary German	Basic structure of the language in the context of culture.	4
	JOUR 150	News Media Management	Capstone course in which students manage media, by overseeing the...	3
	KIN 153	Sport Facility and Event Management	Provide students with the skills necessary to effectively manage sport...	3
	LING 113	Introduction to Phonology	Examination of sound patterns found in the world’s languages, their d...	3
*	NULL	NULL	NULL	NULL

Classes:



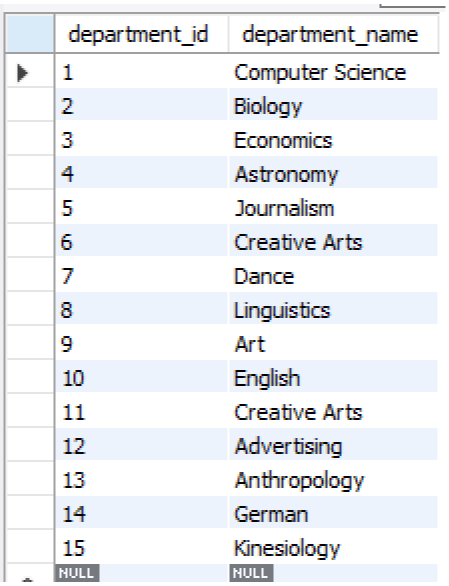
class_id	enrollment	capacity	mode_of_instruction	location	time
1	11	90	Online	N/A	MoWe 2:00PM - 4:00PM
2	0	45	Hybrid	MH 210	TuTh 12:00PM - 2:00PM
3	28	30	Online	N/A	MoWe 3:00PM - 4:00PM
4	22	30	Online	N/A	TuTh 2:00PM - 4:00PM
5	30	30	Hybrid	MH 135	TuTh 2:15PM - 4:15PM
6	43	45	Online	N/A	MoTuWeTh 11:00AM - 12:00PM
7	8	30	In-person	MH 190	MoWe 2:30PM - 4:30PM
8	20	90	In-person	MH 290	Fr 1:00PM - 5:00PM
9	28	30	In-person	MH 300	MoWe 2:00PM - 4:00PM
10	15	30	Online	N/A	TuTh 2:45PM - 4:45PM
11	1	45	Online	N/A	MoWe 10:30AM - 12:30PM
12	30	30	Hybrid	MH 245	MoWe 8:00AM - 10:00AM
13	55	90	Online	N/A	Fr 9:00AM - 10:00AM
14	10	30	In-person	DBH 225	TuTh 7:30AM - 8:45AM
15	14	25	Online	N/A	Fr 5:00PM - 7:00PM
NULL	NULL	NULL	NULL	NULL	NULL

Comment:



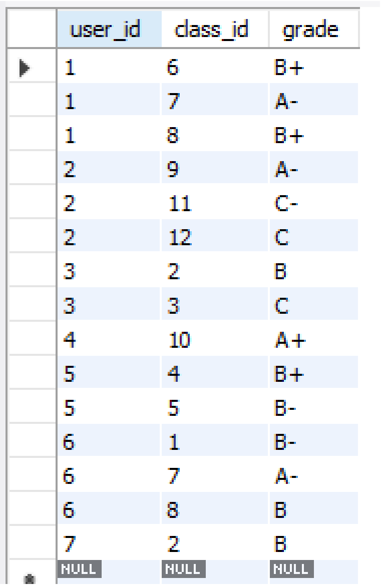
user_id	class_id	comment
7	1	the teacher was very helpful
8	2	take this if you really want to learn
17	2	medium difficulty class and too much w...
8	3	10/10 recommend
2	4	was a very hard class. do not take!!!
17	4	did not like it but the professor was good
3	5	i dropped
4	6	no comment
5	6	BEST CLASS EVER said no one ever
8	7	good class that used modern techniques
17	7	easyyyyy class for A+++
1	8	This class was amazing
1	9	bad class
1	10	nice class
2	10	i hated this class, but i learned so much
NULL	NULL	NULL

Departments:



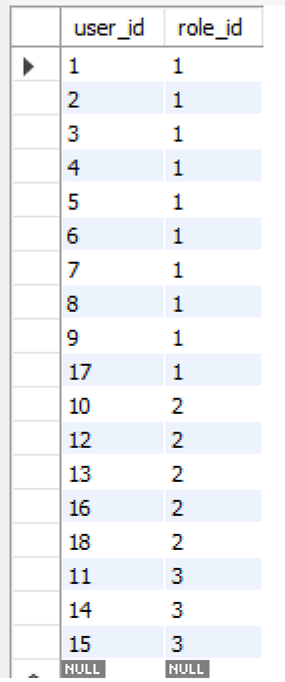
department_id	department_name
1	Computer Science
2	Biology
3	Economics
4	Astronomy
5	Journalism
6	Creative Arts
7	Dance
8	Linguistics
9	Art
10	English
11	Creative Arts
12	Advertising
13	Anthropology
14	German
15	Kinesiology
NULL	NULL

Grade:



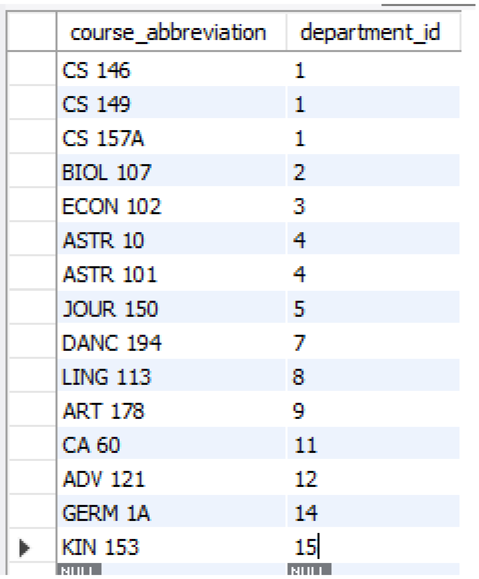
user_id	class_id	grade
1	6	B+
1	7	A-
1	8	B+
2	9	A-
2	11	C-
2	12	C
3	2	B
3	3	C
4	10	A+
5	4	B+
5	5	B-
6	1	B-
6	7	A-
6	8	B
7	2	B
NULL	NULL	NULL

Has:



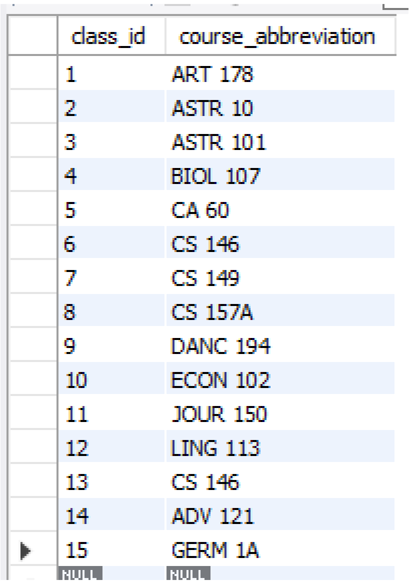
user_id	role_id
1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	1
9	1
17	1
10	2
12	2
13	2
16	2
18	2
11	3
14	3
15	3
NULL	NULL

In:



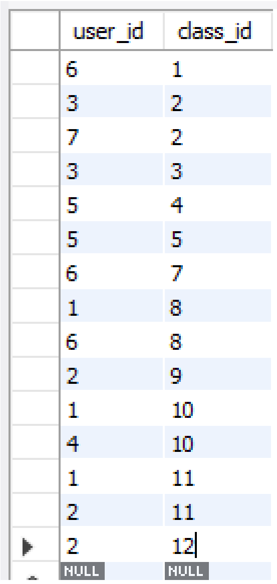
course_abbreviation	department_id
CS 146	1
CS 149	1
CS 157A	1
BIOL 107	2
ECON 102	3
ASTR 10	4
ASTR 101	4
JOUR 150	5
DANC 194	7
LING 113	8
ART 178	9
CA 60	11
ADV 121	12
GERM 1A	14
KIN 153	15
NULL	NULL

Is:



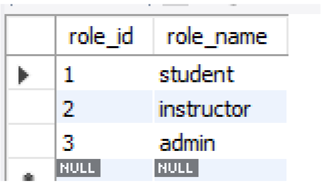
class_id	course_abbreviation
1	ART 178
2	ASTR 10
3	ASTR 101
4	BIOL 107
5	CA 60
6	CS 146
7	CS 149
8	CS 157A
9	DANC 194
10	ECON 102
11	JOUR 150
12	LING 113
13	CS 146
14	ADV 121
15	GERM 1A
NULL	NULL

Register:



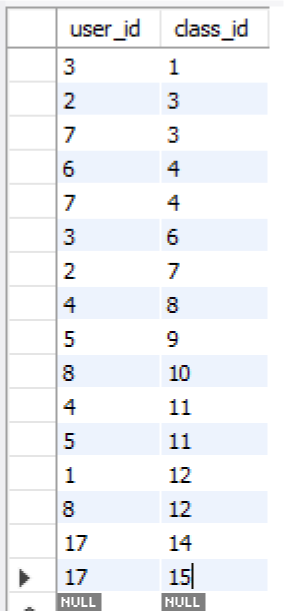
user_id	class_id
6	1
3	2
7	2
3	3
5	4
5	5
6	7
1	8
6	8
2	9
1	10
4	10
1	11
2	11
2	12
NULL	NULL

Roles:



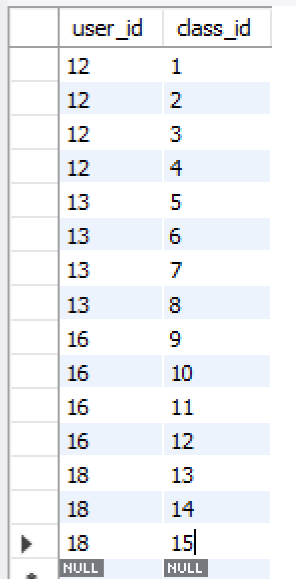
role_id	role_name
1	student
2	instructor
3	admin
NULL	NULL

Save:



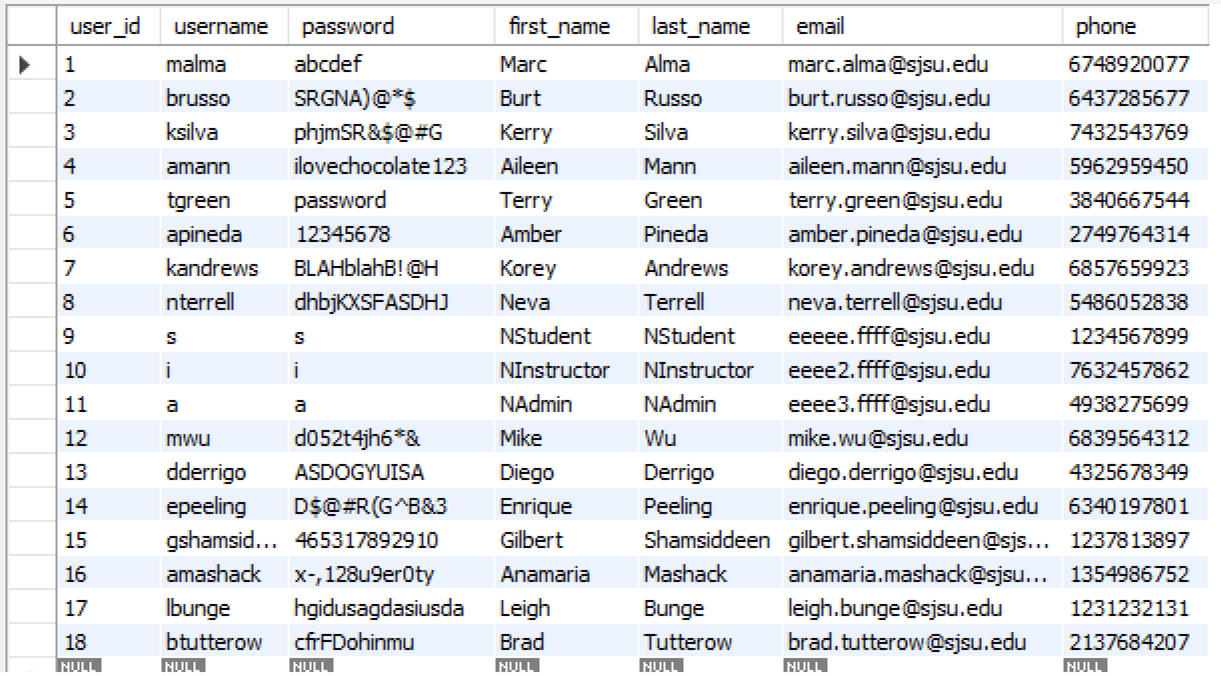
user_id	class_id
3	1
2	3
7	3
6	4
7	4
3	6
2	7
4	8
5	9
8	10
4	11
5	11
1	12
8	12
17	14
17	15
NULL	NULL

Teach:



user_id	class_id
12	1
12	2
12	3
12	4
13	5
13	6
13	7
13	8
16	9
16	10
16	11
16	12
18	13
18	14
18	15
NULL	NULL

Users:

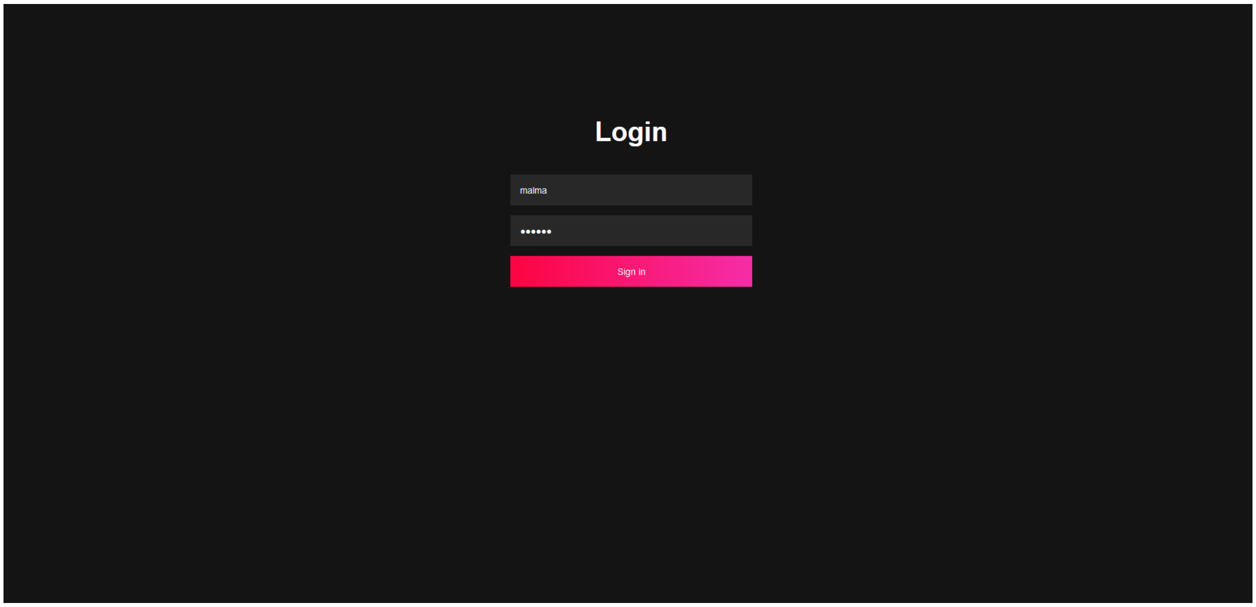


user_id	username	password	first_name	last_name	email	phone
1	malma	abcdef	Marc	Alma	marc.alma@sjsu.edu	6748920077
2	brusso	SRGNA)@*$	Burt	Russo	burt.russo@sjsu.edu	6437285677
3	ksilva	phjmSR&$@#G	Kerry	Silva	kerry.silva@sjsu.edu	7432543769
4	amann	ilovechocolate123	Aileen	Mann	aileen.mann@sjsu.edu	5962959450
5	tgreen	password	Terry	Green	terry.green@sjsu.edu	3840667544
6	apineda	12345678	Amber	Pineda	amber.pineda@sjsu.edu	2749764314
7	kandrews	BLAHblahB!@H	Korey	Andrews	korey.andrews@sjsu.edu	6857659923
8	nterrell	dhbjKXSFASDHJ	Neva	Terrell	neva.terrell@sjsu.edu	5486052838
9	s	s	NStudent	NStudent	eeeee.ffff@sjsu.edu	1234567899
10	i	i	NInstructor	NInstructor	eeee2.ffff@sjsu.edu	7632457862
11	a	a	NAdmin	NAdmin	eeee3.ffff@sjsu.edu	4938275699
12	mwu	d052t4jh6*&	Mike	Wu	mike.wu@sjsu.edu	6839564312
13	dderrigo	ASDOGYUISA	Diego	Derrigo	diego.derrigo@sjsu.edu	4325678349
14	epeeling	D$@#R(G^B&3	Enrique	Peeling	enrique.peeling@sjsu.edu	6340197801
15	gshamsid...	465317892910	Gilbert	Shamsiddeen	gilbert.shamsiddeen@sjs...	1237813897
16	amashack	x-,128u9er0ty	Anamaria	Mashack	anamaria.mashack@sjsu...	1354986752
17	lbunge	hgidusagdasiusda	Leigh	Bunge	leigh.bunge@sjsu.edu	1231232131
18	btutterow	cfrFDohinmu	Brad	Tutterow	brad.tutterow@sjsu.edu	2137684207
NULL	NULL	NULL	NULL	NULL	NULL	NULL

Screenshots:

Here, we will demonstrate how our project works by demonstrating how to use a few functions

This is where users start



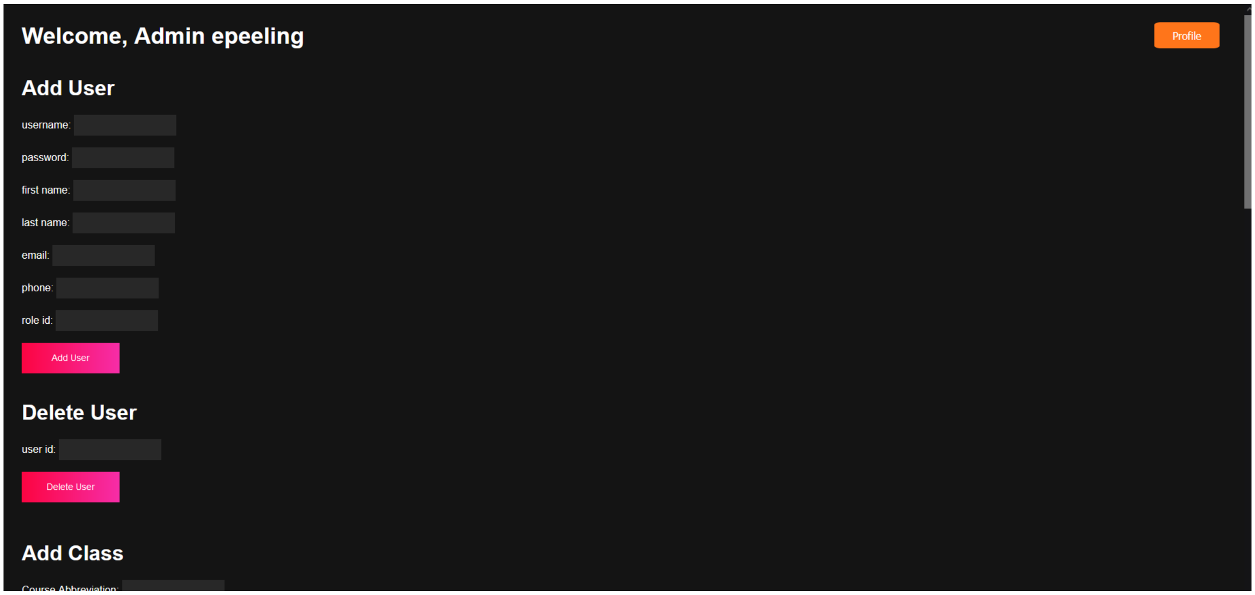
Login

malma

••••••

3 types of users can login: admin, instructors, students
Depending on the type of user logged in, they will be redirected to different pages

Admin page:



Welcome, Admin epeeling

Add User

username:

password:

first name:

last name:

email:

phone:

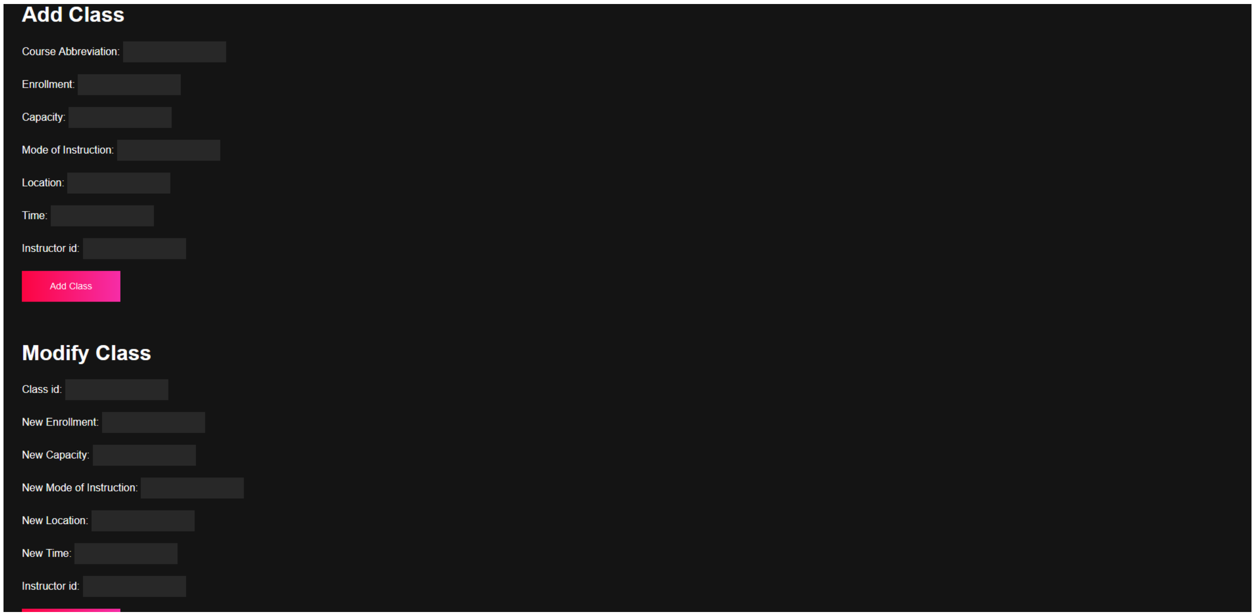
role id:

Delete User

user id:

Add Class

Course Abbreviation:



Add Class

Course Abbreviation:

Enrollment:

Capacity:

Mode of Instruction:

Location:

Time:

Instructor id:

Modify Class

Class id:

New Enrollment:

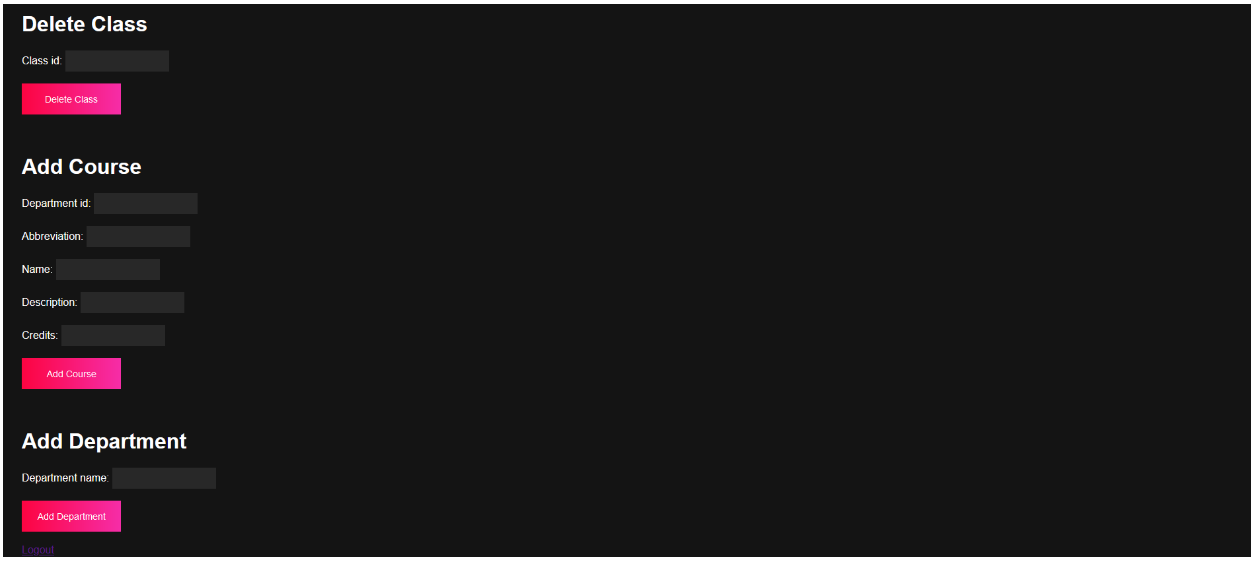
New Capacity:

New Mode of Instruction:

New Location:

New Time:

Instructor id:



Delete Class

Class id:

Add Course

Department id:

Abbreviation:

Name:

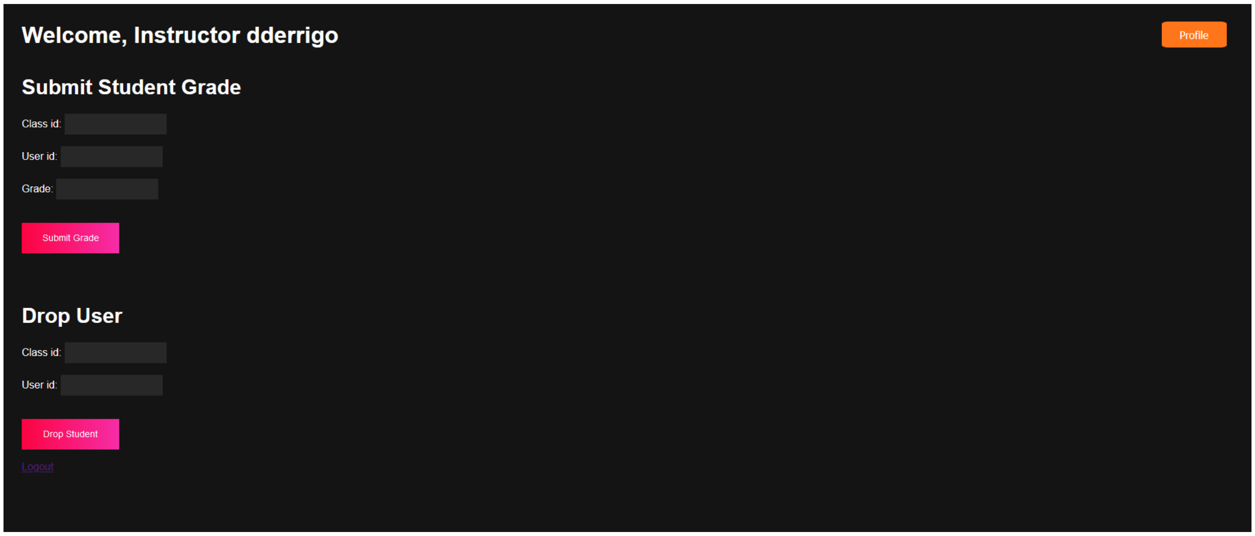
Description:

Credits:

Add Department

Department name:

Instructor page:


Welcome, Instructor dderrigo

Submit Student Grade

Class id:

User id:

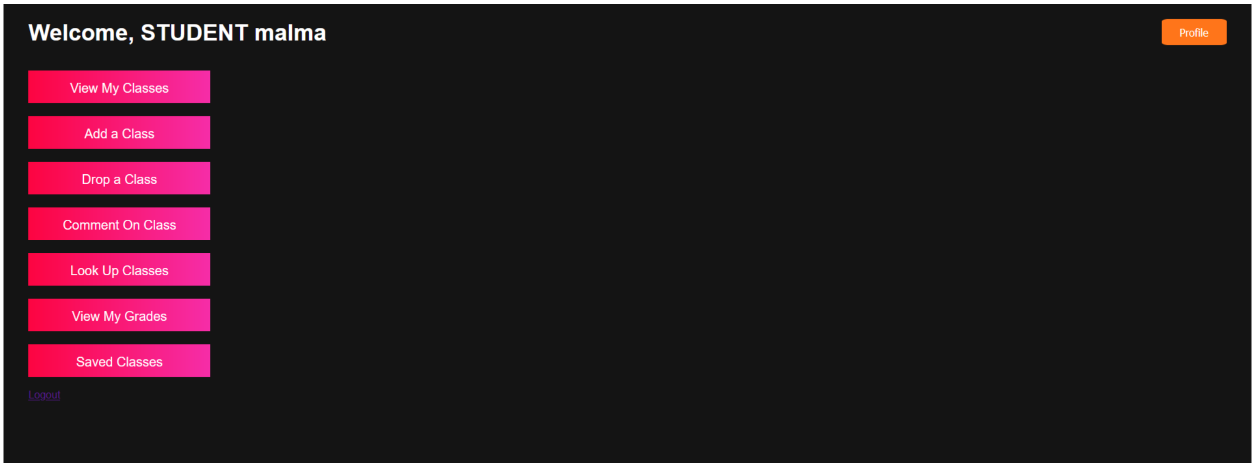
Grade:

Drop User

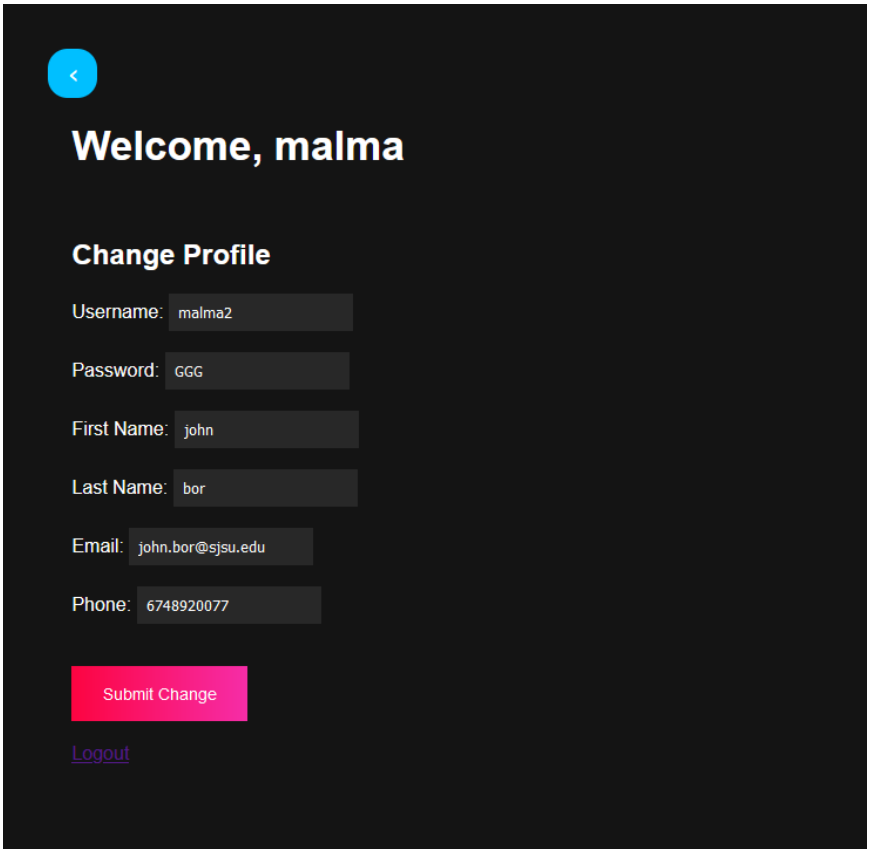
Class id:

User id:

Student page:


Welcome, STUDENT malma

Let’s login as student and change our profile by clicking the profile button on the top right



‹

Welcome, malma

Change Profile

Username: malma2

Password: GGG

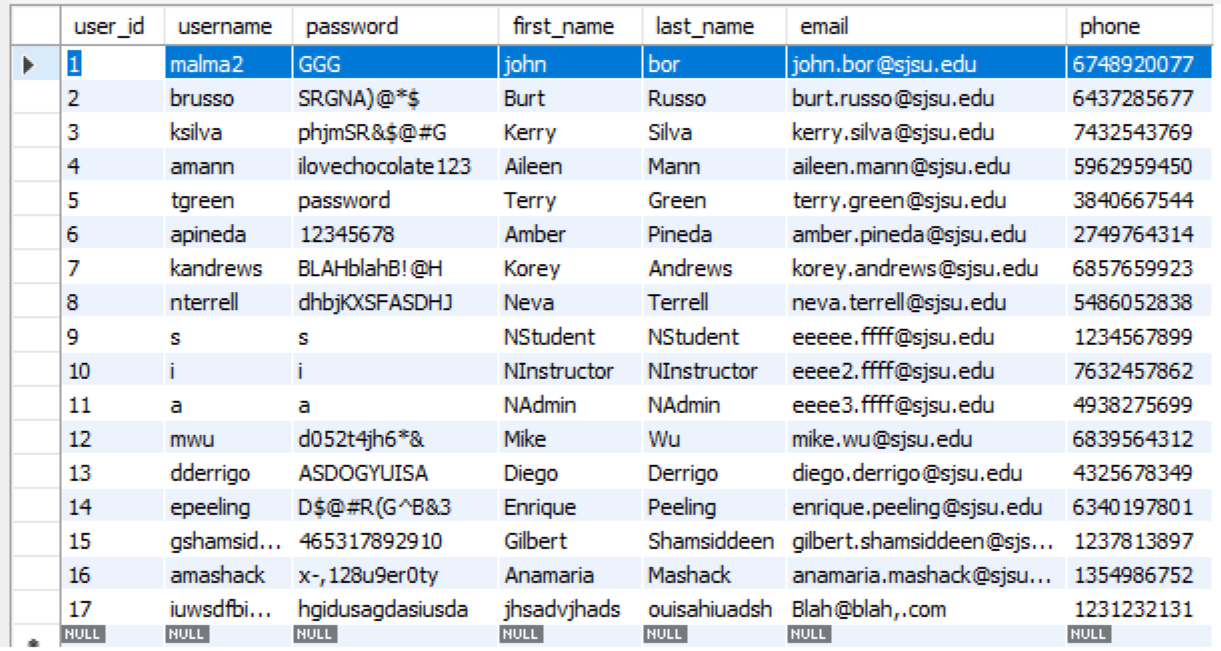
First Name: john

Last Name: bor

Email: john.bor@sjsu.edu

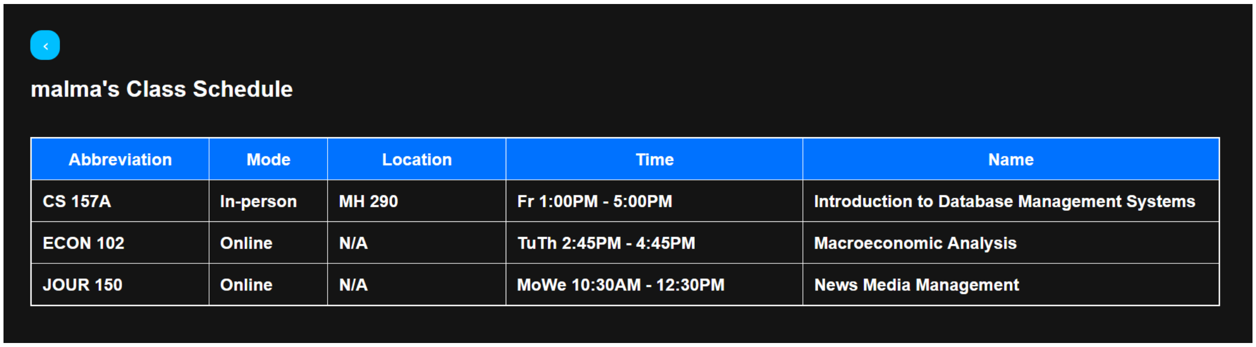
Phone: 6748920077

As we can see the users table updated



	user_id	username	password	first_name	last_name	email	phone
▶	1	malma2	GGG	john	bor	john.bor@sjsu.edu	6748920077
	2	brusso	SRGNA)@*$	Burt	Russo	burt.russo@sjsu.edu	6437285677
	3	ksilva	phjmSR&$@#G	Kerry	Silva	kerry.silva@sjsu.edu	7432543769
	4	amann	ilovechocolate123	Aileen	Mann	aileen.mann@sjsu.edu	5962959450
	5	tgreen	password	Terry	Green	terry.green@sjsu.edu	3840667544
	6	apineda	12345678	Amber	Pineda	amber.pineda@sjsu.edu	2749764314
	7	kandrews	BLAHblahB!@H	Korey	Andrews	korey.andrews@sjsu.edu	6857659923
	8	nterrell	dhbjKXSFASDHJ	Neva	Terrell	neva.terrell@sjsu.edu	5486052838
	9	s	s	NStudent	NStudent	eeeee.ffff@sjsu.edu	1234567899
	10	i	i	NInstructor	NInstructor	eeee2.ffff@sjsu.edu	7632457862
	11	a	a	NAdmin	NAdmin	eeee3.ffff@sjsu.edu	4938275699
	12	mwu	d052t4jh6*&	Mike	Wu	mike.wu@sjsu.edu	6839564312
	13	dderrigo	ASDOGYUISA	Diego	Derrigo	diego.derrigo@sjsu.edu	4325678349
	14	epeeling	D$@#R(G^B&3	Enrique	Peeling	enrique.peeling@sjsu.edu	6340197801
	15	gshamsid...	465317892910	Gilbert	Shamsiddeen	gilbert.shamsiddeen@sjs...	1237813897
	16	amashack	x-,128u9er0ty	Anamaria	Mashack	anamaria.mashack@sjsu...	1354986752
	17	iuwsdfbi...	hgidusagdasiusda	jhsadvjhads	ouisahiuadsh	Blah@blah,.com	1231232131
	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Let’s see what classes we have registered. We can do that by clicking “View My Classes”

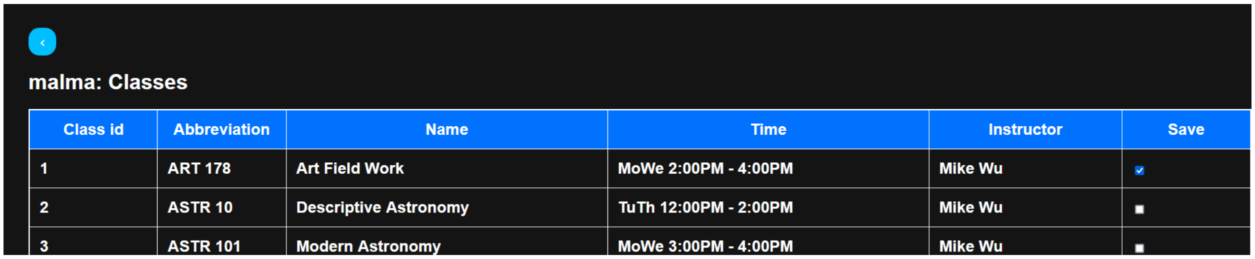


malma's Class Schedule

Abbreviation	Mode	Location	Time	Name
CS 157A	In-person	MH 290	Fr 1:00PM - 5:00PM	Introduction to Database Management Systems
ECON 102	Online	N/A	TuTh 2:45PM - 4:45PM	Macroeconomic Analysis
JOUR 150	Online	N/A	MoWe 10:30AM - 12:30PM	News Media Management

These are the classes that show up

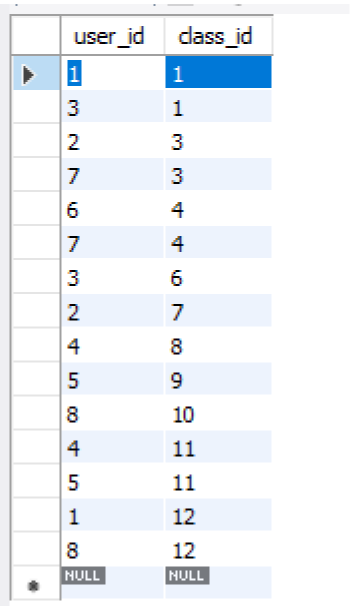
Now let’s say we want to add a new class. We click “Look Up Classes” and tick the ones we like



malma: Classes

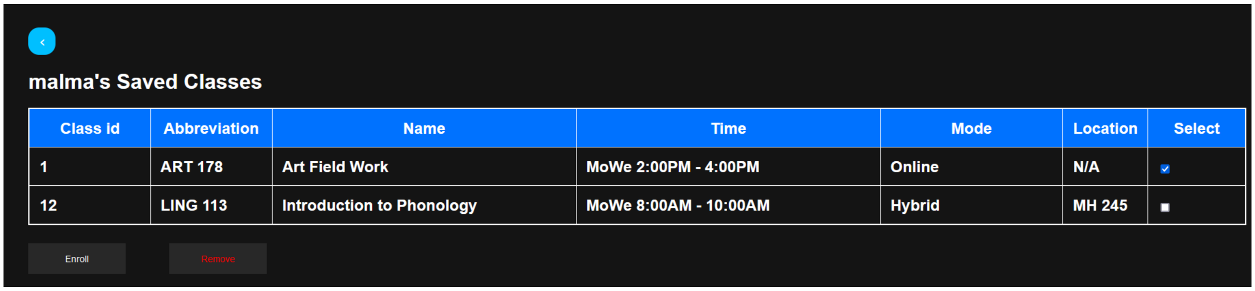
Class id	Abbreviation	Name	Time	Instructor	Save
1	ART 178	Art Field Work	MoWe 2:00PM - 4:00PM	Mike Wu	☑
2	ASTR 10	Descriptive Astronomy	TuTh 12:00PM - 2:00PM	Mike Wu	□
3	ASTR 101	Modern Astronomy	MoWe 3:00PM - 4:00PM	Mike Wu	□

This saved table will now be updated



	user_id	class_id
▶	1	1
	3	1
	2	3
	7	3
	6	4
	7	4
	3	6
	2	7
	4	8
	5	9
	8	10
	4	11
	5	11
	1	12
	8	12
*	NULL	NULL

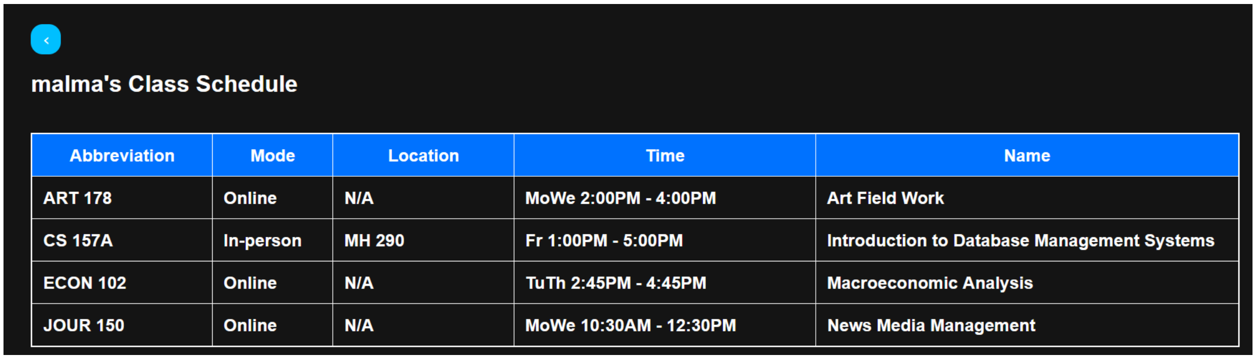
We can now go to Saved Classes and select the class we just added and proceed to enroll.



malma's Saved Classes

Class id	Abbreviation	Name	Time	Mode	Location	Select
1	ART 178	Art Field Work	MoWe 2:00PM - 4:00PM	Online	N/A	☑
12	LING 113	Introduction to Phonology	MoWe 8:00AM - 10:00AM	Hybrid	MH 245	□

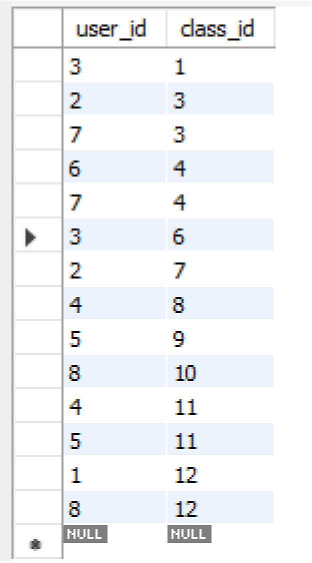
If we go back to View My Classes, we can see that the class was successfully added



malma's Class Schedule

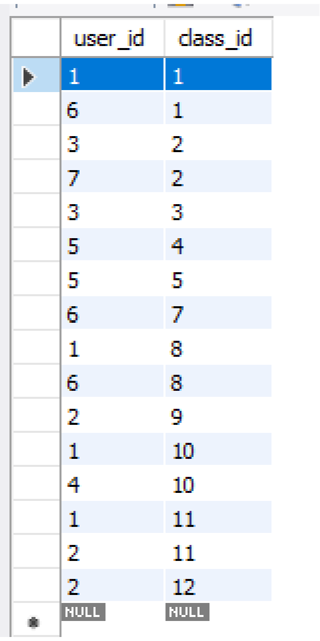
Abbreviation	Mode	Location	Time	Name
ART 178	Online	N/A	MoWe 2:00PM - 4:00PM	Art Field Work
CS 157A	In-person	MH 290	Fr 1:00PM - 5:00PM	Introduction to Database Management Systems
ECON 102	Online	N/A	TuTh 2:45PM - 4:45PM	Macroeconomic Analysis
JOUR 150	Online	N/A	MoWe 10:30AM - 12:30PM	News Media Management

The class will be removed from the save table



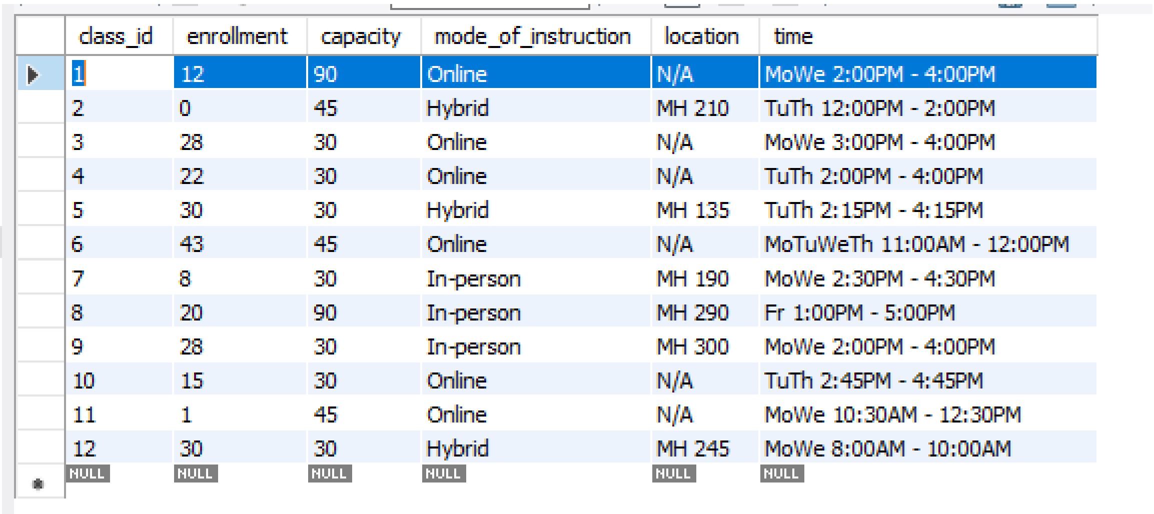
user_id	class_id
3	1
2	3
7	3
6	4
7	4
3	6
2	7
4	8
5	9
8	10
4	11
5	11
1	12
8	12
NULL	NULL

The register table is now also updated



user_id	class_id
1	1
6	1
3	2
7	2
3	3
5	4
5	5
6	7
1	8
6	8
2	9
1	10
4	10
1	11
2	11
2	12
NULL	NULL

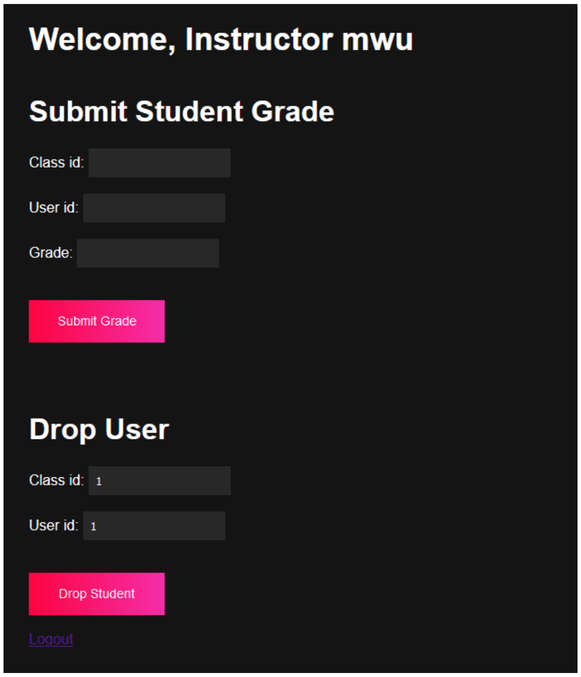
In the classes table, the number of students enrolled has also gone up by 1 (11 to 12)



class_id	enrollment	capacity	mode_of_instruction	location	time
1	12	90	Online	N/A	MoWe 2:00PM - 4:00PM
2	0	45	Hybrid	MH 210	TuTh 12:00PM - 2:00PM
3	28	30	Online	N/A	MoWe 3:00PM - 4:00PM
4	22	30	Online	N/A	TuTh 2:00PM - 4:00PM
5	30	30	Hybrid	MH 135	TuTh 2:15PM - 4:15PM
6	43	45	Online	N/A	MoTuWeTh 11:00AM - 12:00PM
7	8	30	In-person	MH 190	MoWe 2:30PM - 4:30PM
8	20	90	In-person	MH 290	Fr 1:00PM - 5:00PM
9	28	30	In-person	MH 300	MoWe 2:00PM - 4:00PM
10	15	30	Online	N/A	TuTh 2:45PM - 4:45PM
11	1	45	Online	N/A	MoWe 10:30AM - 12:30PM
12	30	30	Hybrid	MH 245	MoWe 8:00AM - 10:00AM
NULL	NULL	NULL	NULL	NULL	NULL

The user could've also instead bypassed all these steps and simply clicked the "Add Class" button to quickly add the new class using only its class id.

Now let's click logout in the bottom to logout from student and log back in as an instructor
Let's attempt to drop the student who just registered



Welcome, Instructor mwu

Submit Student Grade

Class id:

User id:

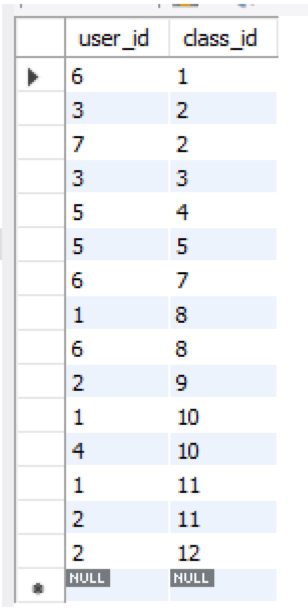
Grade:

Drop User

Class id: 1

User id: 1

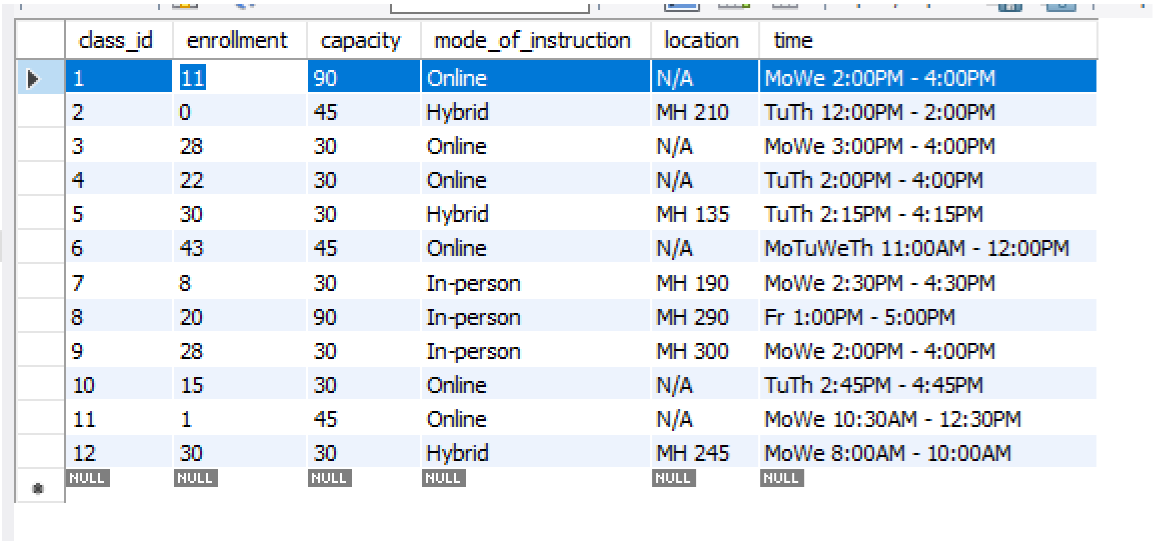
As we can see the user has been removed from the class from the registered table



user_id	class_id
6	1
3	2
7	2
3	3
5	4
5	5
6	7
1	8
6	8
2	9
1	10
4	10
1	11
2	11
2	12
NULL	NULL

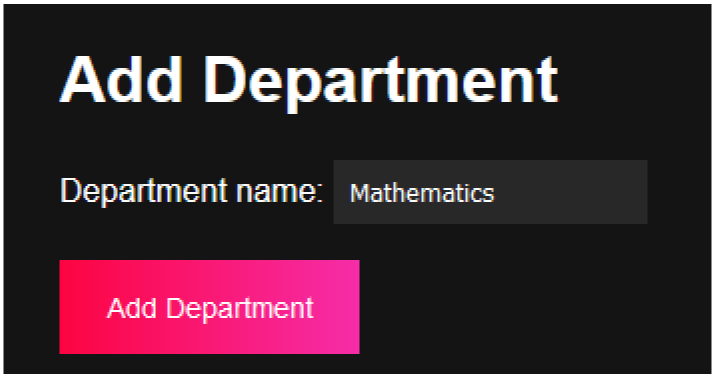
The user's grade was also removed (in this case, the user still didn't have a grade for that class so nothing happens)

Enrollment also decreases by 1 for that class



class_id	enrollment	capacity	mode_of_instruction	location	time
1	11	90	Online	N/A	MoWe 2:00PM - 4:00PM
2	0	45	Hybrid	MH 210	TuTh 12:00PM - 2:00PM
3	28	30	Online	N/A	MoWe 3:00PM - 4:00PM
4	22	30	Online	N/A	TuTh 2:00PM - 4:00PM
5	30	30	Hybrid	MH 135	TuTh 2:15PM - 4:15PM
6	43	45	Online	N/A	MoTuWeTh 11:00AM - 12:00PM
7	8	30	In-person	MH 190	MoWe 2:30PM - 4:30PM
8	20	90	In-person	MH 290	Fr 1:00PM - 5:00PM
9	28	30	In-person	MH 300	MoWe 2:00PM - 4:00PM
10	15	30	Online	N/A	TuTh 2:45PM - 4:45PM
11	1	45	Online	N/A	MoWe 10:30AM - 12:30PM
12	30	30	Hybrid	MH 245	MoWe 8:00AM - 10:00AM
NULL	NULL	NULL	NULL	NULL	NULL

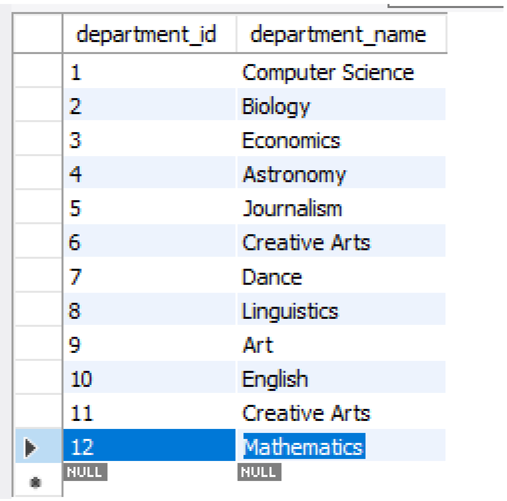
Now let's logout again and try a function from admin



Add Department

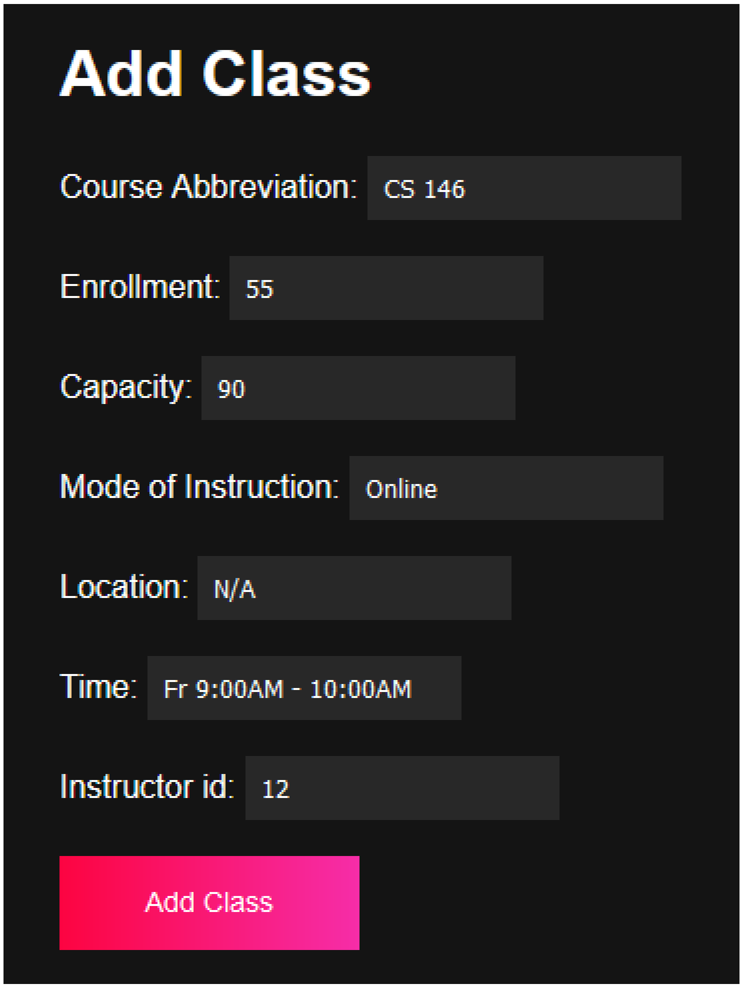
Department name: Mathematics

As we can see, a new department has been added



department_id	department_name
1	Computer Science
2	Biology
3	Economics
4	Astronomy
5	Journalism
6	Creative Arts
7	Dance
8	Linguistics
9	Art
10	English
11	Creative Arts
12	Mathematics
NULL	NULL

Let’s try another function. Let’s try to add a new class.



Add Class

Course Abbreviation: CS 146

Enrollment: 55

Capacity: 90

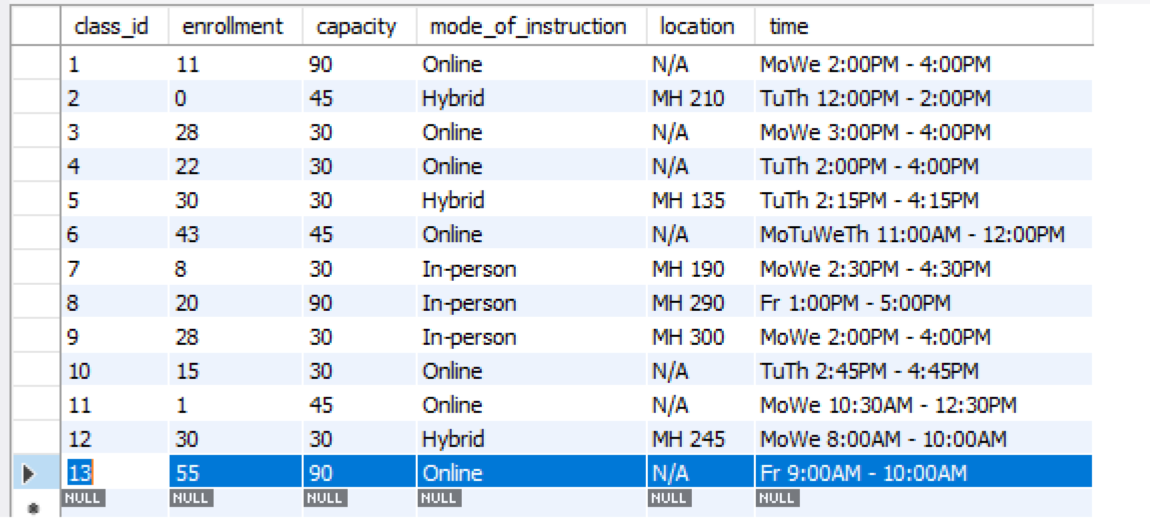
Mode of Instruction: Online

Location: N/A

Time: Fr 9:00AM - 10:00AM

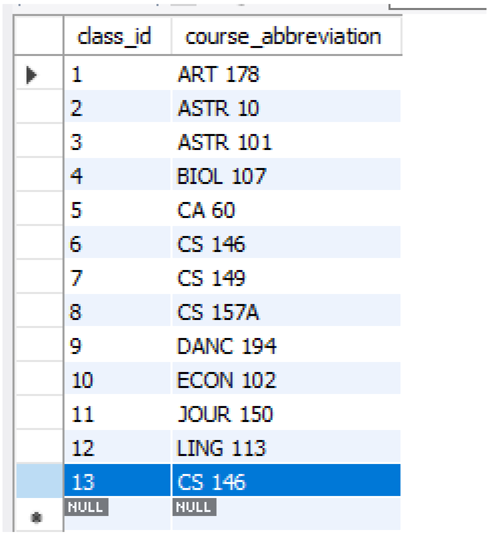
Instructor id: 12

Classes table is now updated:



	class_id	enrollment	capacity	mode_of_instruction	location	time
	1	11	90	Online	N/A	MoWe 2:00PM - 4:00PM
	2	0	45	Hybrid	MH 210	TuTh 12:00PM - 2:00PM
	3	28	30	Online	N/A	MoWe 3:00PM - 4:00PM
	4	22	30	Online	N/A	TuTh 2:00PM - 4:00PM
	5	30	30	Hybrid	MH 135	TuTh 2:15PM - 4:15PM
	6	43	45	Online	N/A	MoTuWeTh 11:00AM - 12:00PM
	7	8	30	In-person	MH 190	MoWe 2:30PM - 4:30PM
	8	20	90	In-person	MH 290	Fr 1:00PM - 5:00PM
	9	28	30	In-person	MH 300	MoWe 2:00PM - 4:00PM
	10	15	30	Online	N/A	TuTh 2:45PM - 4:45PM
	11	1	45	Online	N/A	MoWe 10:30AM - 12:30PM
	12	30	30	Hybrid	MH 245	MoWe 8:00AM - 10:00AM
▶	13	55	90	Online	N/A	Fr 9:00AM - 10:00AM
*	NULL	NULL	NULL	NULL	NULL	NULL

Is table is also updated



class_id	course_abbreviation
1	ART 178
2	ASTR 10
3	ASTR 101
4	BIOL 107
5	CA 60
6	CS 146
7	CS 149
8	CS 157A
9	DANC 194
10	ECON 102
11	JOUR 150
12	LING 113
13	CS 146
NULL	NULL

Lesson Learned:

Mario Alkhoury:

I learned a few things from this project. First and foremost, this project has strengthened my sql and MySQL knowledge by applying concepts to the real world. This project allowed me to learn about the uses of databases in real world applications for the very first time. I learned how a 3 tier architecture works and how to set it up. I also strengthened my foundation in basic frontend languages (HTML/CSS/javascript) and learned how to use them in tandem with Java to create jsp pages. I also learned about entities, relationships, and how to design efficient ER diagrams so that I can translate them into relational schema. I have gained a great foundation in database systems that I can use to further my knowledge in other applications as well as advance further into database systems.

Youhao Chen:

I have learned a lot.

First of all, the use process of JDBC, how to construct a three-tier system is the most important. This includes: How to read and operate the database by calling the sql class library in jsp. How to build server and use tomcat.

2. The basic operation of jsp: display java data on the web page and insert data into the html table. How to store global variables through session, and how to pass variables between pages.

3. Database design, how to design a better and more scientific database. The production process of ER diagram, the proficient application of SQL language, and the use of Mysql workbrench.

4. Front-end HTML language, the use of css files.