Group- Oracles Eleven

Assignment-4

3.1 Responsibility of G1:

1. The G1 takes two feedbacks from the stakeholders, one initial feedback (on or before 30th March 11:59 PM), and then makes relevant changes as suggested per the first feedback, then final feedback (on or before 10th April 11:59 PM) post changes. The write-up/documentation should have screenshots before the first feedback, after the first feedback, and after the second feedback. If a team discusses with multiple stakeholders, please fill out the forms again (Initial and final feedback forms). **(30 Points)**

Feedback 1 :-

1.Use a single login page. Store the data of all users in a single table and later segregate into student, professor etc.

2. Define the budget and number of students that can be taken.

3. use bootstrap for data tables.

4. operation wise its correct.

5. work on design.

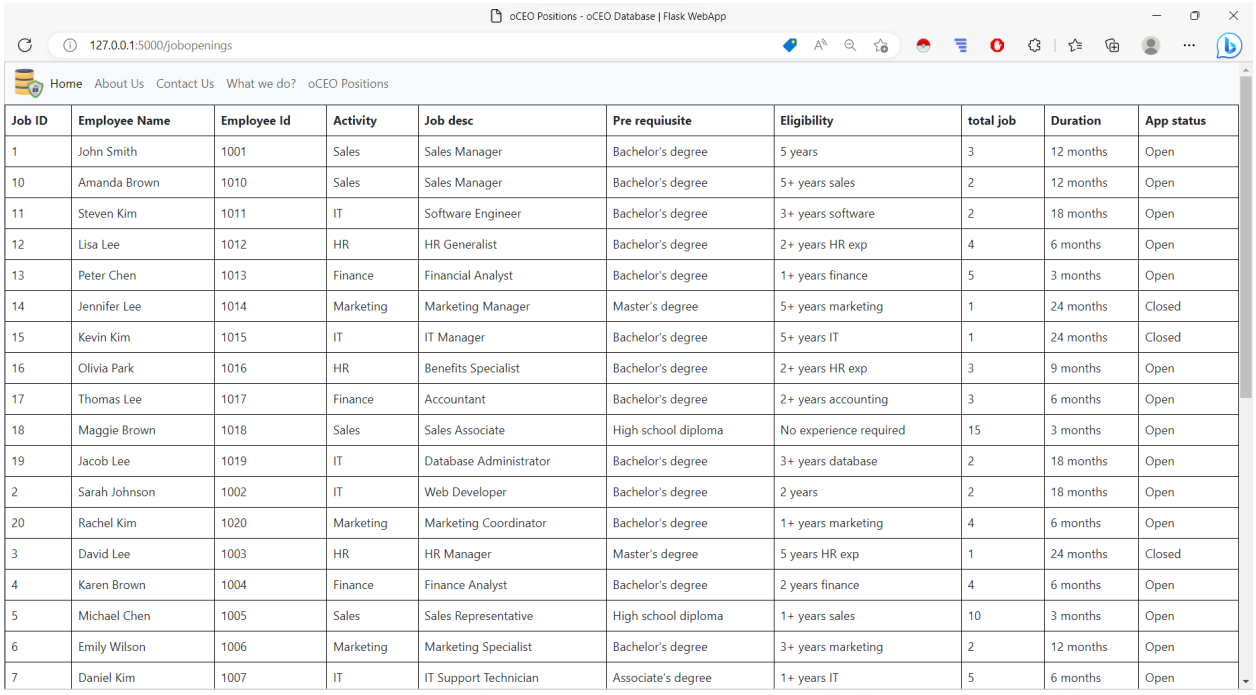
6..Don't keep passwords in the frontend as admins can see them directly, and have access to change them.

7. Do not keep cpi in the frontend as that is personal for every student.

8. Rename the tables name

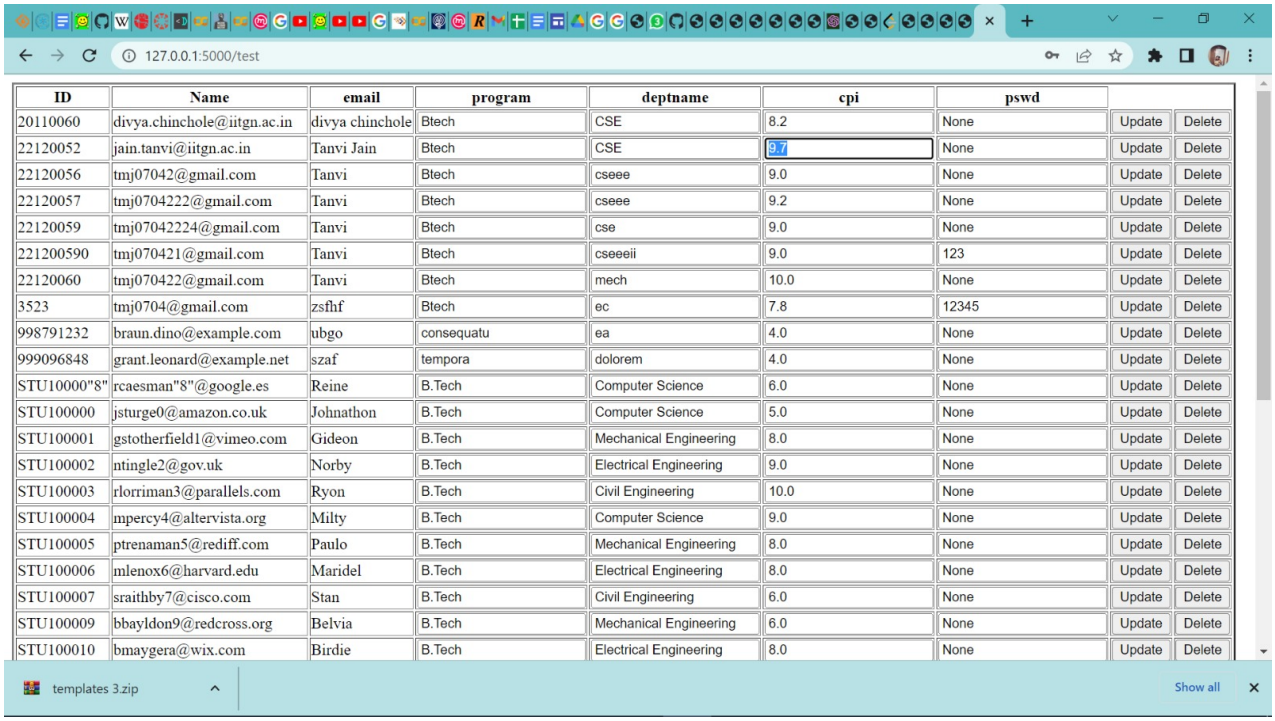
Before first feedback

1. No direct button for the application process for the specific job



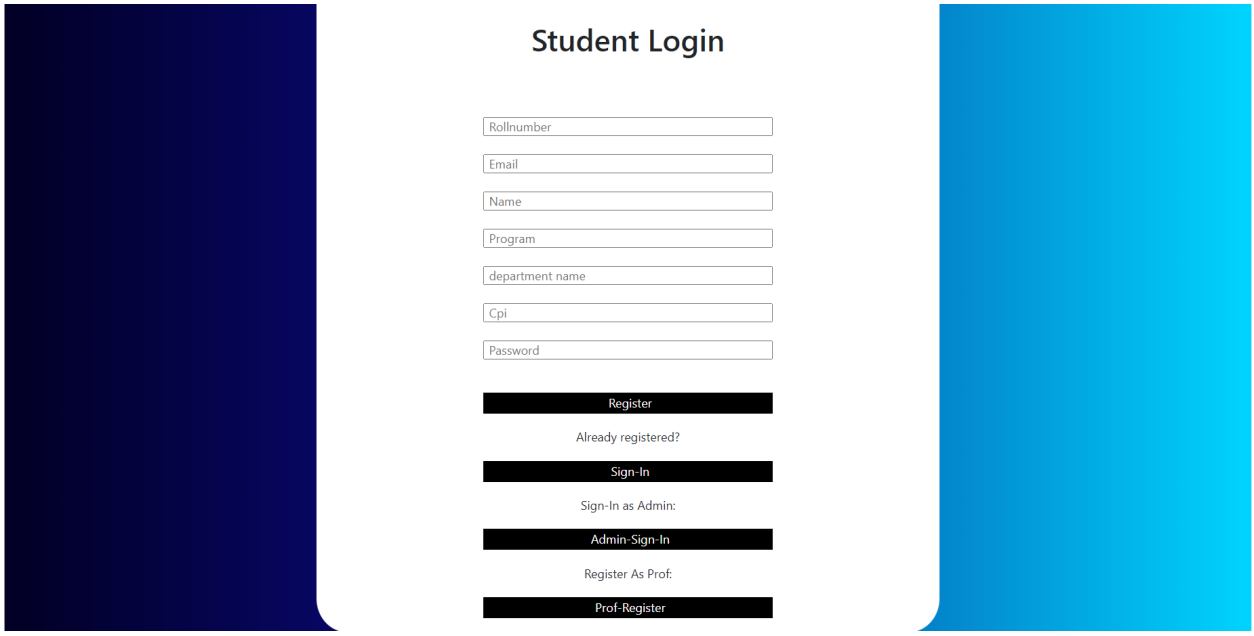
Job ID	Employee Name	Employee Id	Activity	Job desc	Pre requiusite	Eligibility	total job	Duration	App status
1	John Smith	1001	Sales	Sales Manager	Bachelor's degree	5 years	3	12 months	Open
10	Amanda Brown	1010	Sales	Sales Manager	Bachelor's degree	5+ years sales	2	12 months	Open
11	Steven Kim	1011	IT	Software Engineer	Bachelor's degree	3+ years software	2	18 months	Open
12	Lisa Lee	1012	HR	HR Generalist	Bachelor's degree	2+ years HR exp	4	6 months	Open
13	Peter Chen	1013	Finance	Financial Analyst	Bachelor's degree	1+ years finance	5	3 months	Open
14	Jennifer Lee	1014	Marketing	Marketing Manager	Master's degree	5+ years marketing	1	24 months	Closed
15	Kevin Kim	1015	IT	IT Manager	Bachelor's degree	5+ years IT	1	24 months	Closed
16	Olivia Park	1016	HR	Benefits Specialist	Bachelor's degree	2+ years HR exp	3	9 months	Open
17	Thomas Lee	1017	Finance	Accountant	Bachelor's degree	2+ years accounting	3	6 months	Open
18	Maggie Brown	1018	Sales	Sales Associate	High school diploma	No experience required	15	3 months	Open
19	Jacob Lee	1019	IT	Database Administrator	Bachelor's degree	3+ years database	2	18 months	Open
2	Sarah Johnson	1002	IT	Web Developer	Bachelor's degree	2 years	2	18 months	Open
20	Rachel Kim	1020	Marketing	Marketing Coordinator	Bachelor's degree	1+ years marketing	4	6 months	Open
3	David Lee	1003	HR	HR Manager	Master's degree	5 years HR exp	1	24 months	Closed
4	Karen Brown	1004	Finance	Finance Analyst	Bachelor's degree	2 years finance	4	6 months	Open
5	Michael Chen	1005	Sales	Sales Representative	High school diploma	1+ years sales	10	3 months	Open
6	Emily Wilson	1006	Marketing	Marketing Specialist	Bachelor's degree	3+ years marketing	2	12 months	Open
7	Daniel Kim	1007	IT	IT Support Technician	Associate's degree	1+ years IT	5	6 months	Open

2. Cpi and password visible for the admin view of the student.



ID	Name	email	program	deptname	cpi	pswd		
20110060	divya.chinchole@iitgn.ac.in	divya chinchole	Btech	CSE	8.2	None	Update	Delete
22120052	jain.tanvi@iitgn.ac.in	Tanvi Jain	Btech	CSE	9.7	None	Update	Delete
22120056	tmj07042@gmail.com	Tanvi	Btech	cseee	9.0	None	Update	Delete
22120057	tmj0704222@gmail.com	Tanvi	Btech	cseee	9.2	None	Update	Delete
22120059	tmj07042224@gmail.com	Tanvi	Btech	cse	9.0	None	Update	Delete
221200590	tmj070421@gmail.com	Tanvi	Btech	cseeeii	9.0	123	Update	Delete
22120060	tmj070422@gmail.com	Tanvi	Btech	mech	10.0	None	Update	Delete
3523	tmj0704@gmail.com	zsfhf	Btech	ec	7.8	12345	Update	Delete
998791232	braun.dino@example.com	ubgo	consequatu	ea	4.0	None	Update	Delete
999096848	grant.leonard@example.net	szaf	tempora	dolorem	4.0	None	Update	Delete
STU10000"8"	rcaesman"8"@google.es	Reine	B.Tech	Computer Science	6.0	None	Update	Delete
STU100000	jsturge0@amazon.co.uk	Johnathon	B.Tech	Computer Science	5.0	None	Update	Delete
STU100001	gstotherfield1@vimeo.com	Gideon	B.Tech	Mechanical Engineering	8.0	None	Update	Delete
STU100002	ntingle2@gov.uk	Norby	B.Tech	Electrical Engineering	9.0	None	Update	Delete
STU100003	rlorriman3@parallels.com	Ryon	B.Tech	Civil Engineering	10.0	None	Update	Delete
STU100004	mpercy4@altervista.org	Milty	B.Tech	Computer Science	9.0	None	Update	Delete
STU100005	ptrenaman5@rediff.com	Paulo	B.Tech	Mechanical Engineering	8.0	None	Update	Delete
STU100006	mlenox6@harvard.edu	Maridel	B.Tech	Electrical Engineering	8.0	None	Update	Delete
STU100007	sraithby7@cisco.com	Stan	B.Tech	Civil Engineering	6.0	None	Update	Delete
STU100009	bbayldon9@redcross.org	Belvia	B.Tech	Mechanical Engineering	6.0	None	Update	Delete
STU100010	bmaygera@wix.com	Birdie	B.Tech	Electrical Engineering	8.0	None	Update	Delete

3. All login pages displayed at a single place



4. Renaming the tables name of student and professor to student_details and professor_details

```
78 •  show create table student;
79 •  desc student;
80 •  desc professor;
81
82 •  alter table student rename student_details;
83 •  alter table professor rename professor_details;
```

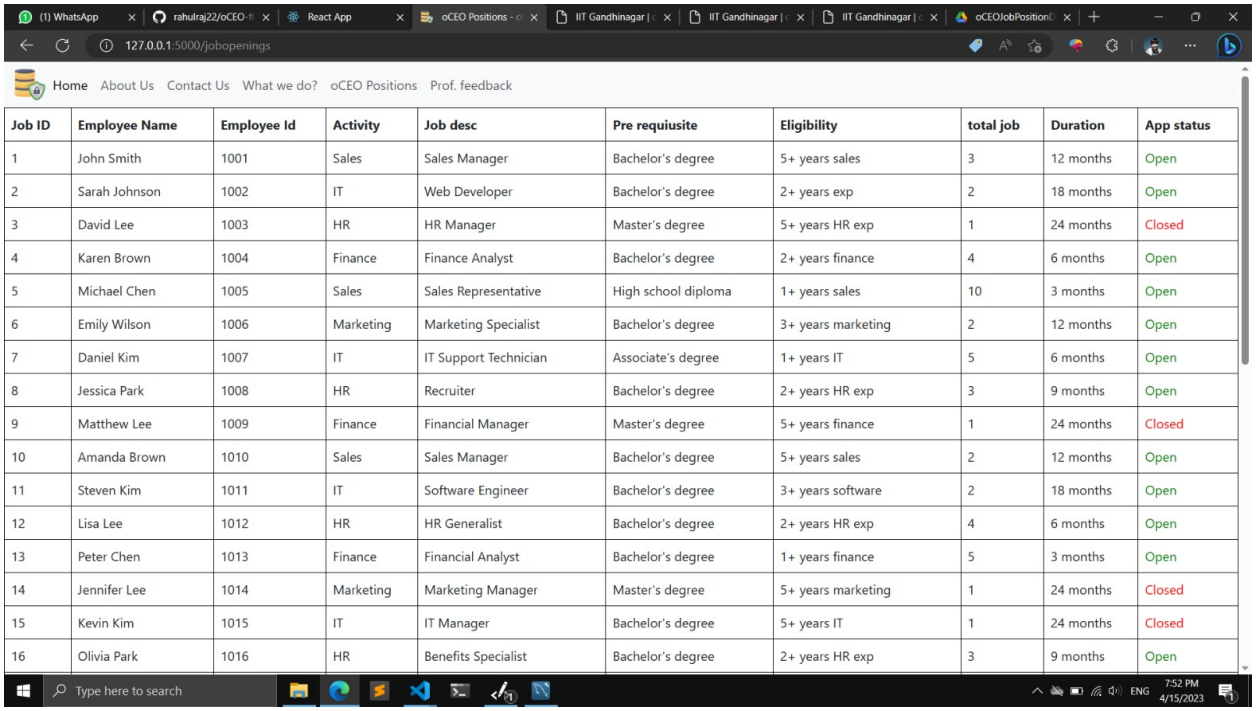
Output

Action Output

#	Time	Action	Message
39	19:48:41	create table professor (email varchar(200), emp_id varchar(20), emp_name char(30), pswd varchar(200), primar...	0 row(s) affected
40	19:48:50	INSERT INTO professor (email, emp_id, emp_name, pswd) VALUES ('professor1@university.edu', 'EMP10000...	5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0
41	19:51:08	desc oceo_positions	10 row(s) returned
42	19:59:23	desc student desc professor	Error Code: 1064. You have an error in your SQL syntax; check the manual that corresponds t
43	19:59:31	desc student	7 row(s) returned
44	20:00:18	alter table student rename student_details	0 row(s) affected

After first feedback

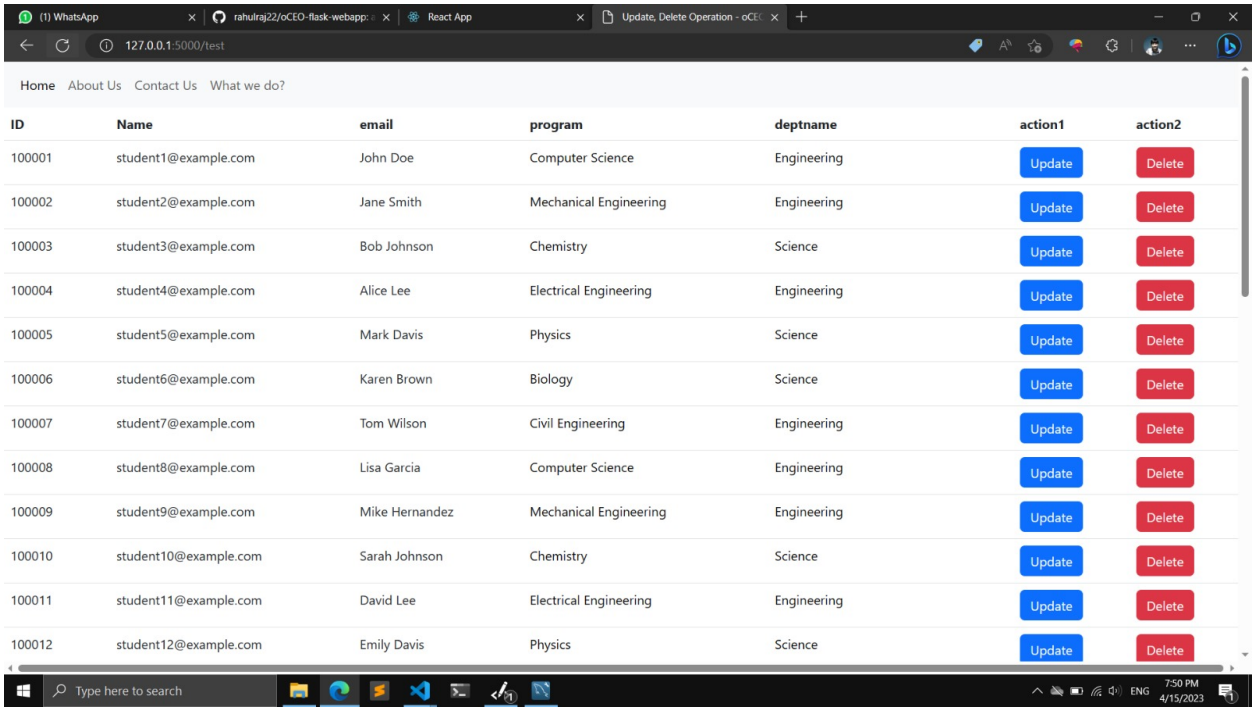
1. Added a hyperlinked button on the open/close button for direct application process



Home About Us Contact Us What we do? oCEO Positions Prof. feedback

Job ID	Employee Name	Employee Id	Activity	Job desc	Pre requiusite	Eligibility	total job	Duration	App status
1	John Smith	1001	Sales	Sales Manager	Bachelor's degree	5+ years sales	3	12 months	Open
2	Sarah Johnson	1002	IT	Web Developer	Bachelor's degree	2+ years exp	2	18 months	Open
3	David Lee	1003	HR	HR Manager	Master's degree	5+ years HR exp	1	24 months	Closed
4	Karen Brown	1004	Finance	Finance Analyst	Bachelor's degree	2+ years finance	4	6 months	Open
5	Michael Chen	1005	Sales	Sales Representative	High school diploma	1+ years sales	10	3 months	Open
6	Emily Wilson	1006	Marketing	Marketing Specialist	Bachelor's degree	3+ years marketing	2	12 months	Open
7	Daniel Kim	1007	IT	IT Support Technician	Associate's degree	1+ years IT	5	6 months	Open
8	Jessica Park	1008	HR	Recruiter	Bachelor's degree	2+ years HR exp	3	9 months	Open
9	Matthew Lee	1009	Finance	Financial Manager	Master's degree	5+ years finance	1	24 months	Closed
10	Amanda Brown	1010	Sales	Sales Manager	Bachelor's degree	5+ years sales	2	12 months	Open
11	Steven Kim	1011	IT	Software Engineer	Bachelor's degree	3+ years software	2	18 months	Open
12	Lisa Lee	1012	HR	HR Generalist	Bachelor's degree	2+ years HR exp	4	6 months	Open
13	Peter Chen	1013	Finance	Financial Analyst	Bachelor's degree	1+ years finance	5	3 months	Open
14	Jennifer Lee	1014	Marketing	Marketing Manager	Master's degree	5+ years marketing	1	24 months	Closed
15	Kevin Kim	1015	IT	IT Manager	Bachelor's degree	5+ years IT	1	24 months	Closed
16	Olivia Park	1016	HR	Benefits Specialist	Bachelor's degree	2+ years HR exp	3	9 months	Open

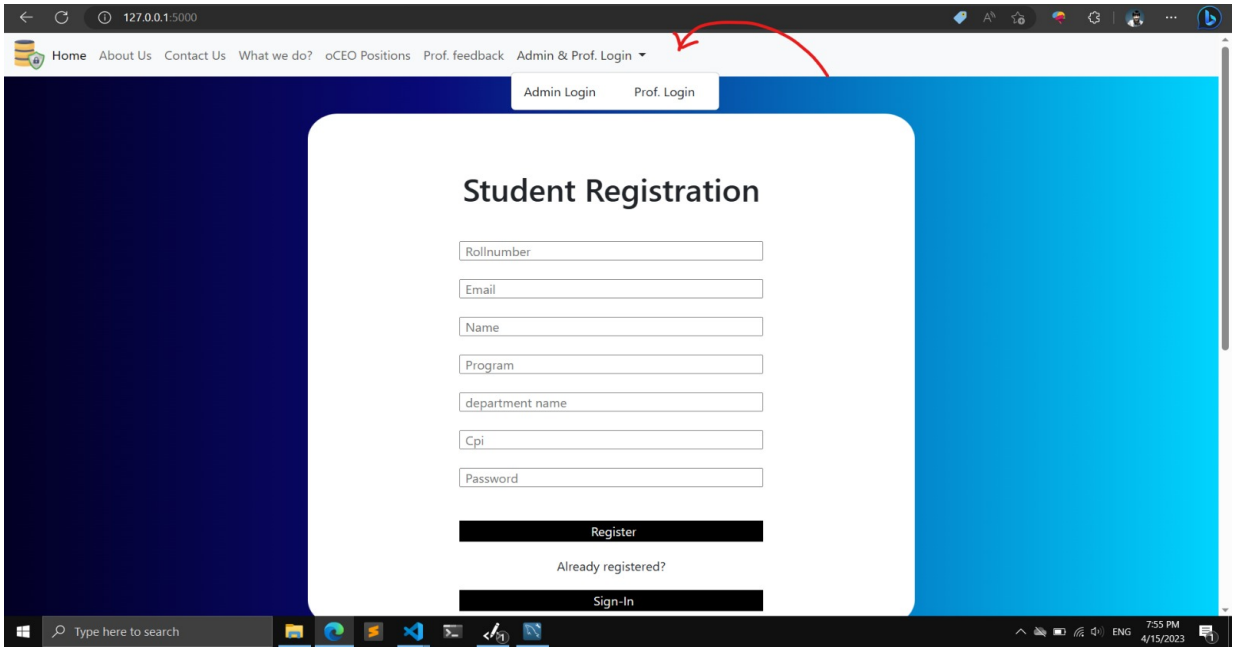
2. Removed the cpi and password for the admin view of the student and improved the css



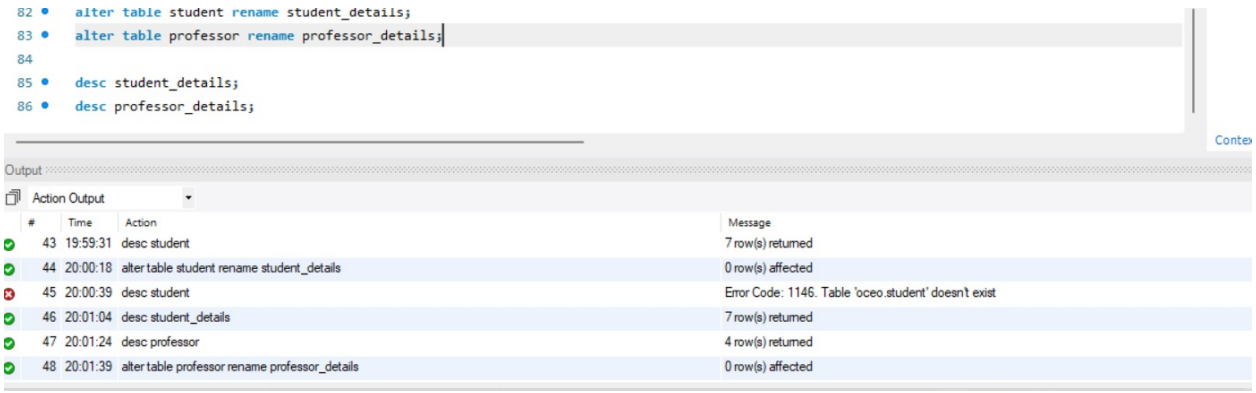
Home About Us Contact Us What we do?

ID	Name	email	program	deptname	action1	action2
100001	student1@example.com	John Doe	Computer Science	Engineering	Update	Delete
100002	student2@example.com	Jane Smith	Mechanical Engineering	Engineering	Update	Delete
100003	student3@example.com	Bob Johnson	Chemistry	Science	Update	Delete
100004	student4@example.com	Alice Lee	Electrical Engineering	Engineering	Update	Delete
100005	student5@example.com	Mark Davis	Physics	Science	Update	Delete
100006	student6@example.com	Karen Brown	Biology	Science	Update	Delete
100007	student7@example.com	Tom Wilson	Civil Engineering	Engineering	Update	Delete
100008	student8@example.com	Lisa Garcia	Computer Science	Engineering	Update	Delete
100009	student9@example.com	Mike Hernandez	Mechanical Engineering	Engineering	Update	Delete
100010	student10@example.com	Sarah Johnson	Chemistry	Science	Update	Delete
100011	student11@example.com	David Lee	Electrical Engineering	Engineering	Update	Delete
100012	student12@example.com	Emily Davis	Physics	Science	Update	Delete

3. Added a dropdown menu for login



4. Renaming the tables name of student and professor to student_details and professor_details



Why did we not implement a single login page?

1. Having multiple login pages for each stakeholder, such as students and professors, is better than having one single database for several reasons. Firstly, having separate login pages provides greater security by restricting access to the system based on specific user roles and permissions. This ensures that sensitive data is protected, and users can only access the information they need to perform their tasks.

2. Having separate login pages makes it easier to manage the database as the system can be designed and optimized based on the specific needs of each stakeholder. For example, the user interface and functionalities can be tailored to meet the requirements of each group, leading to a more efficient and user-friendly system.

3. As the number of students and professors increases in the future, having multiple login pages ensures that the system can handle the growing number of users without compromising on performance. With a single database, the increase in users can lead to slower response times and increased processing times, resulting in reduced efficiency.

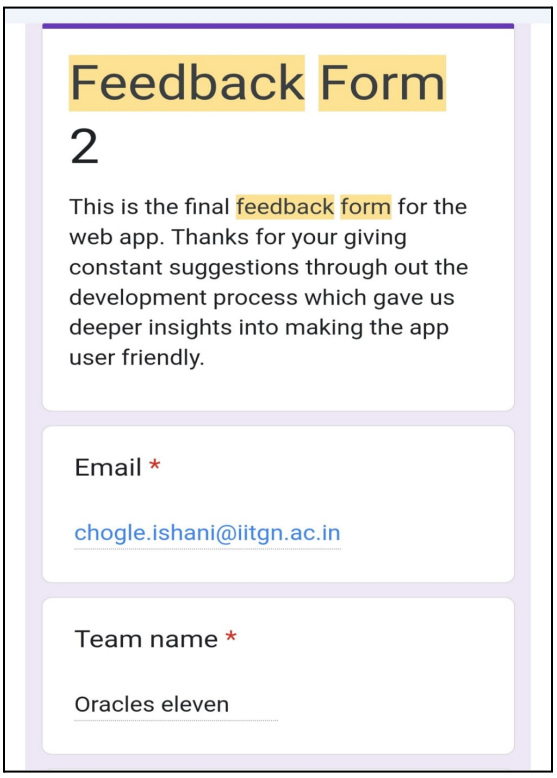
Why is CPI being shown on the front end?

The CPI attribute present on the frontend is only for the registration process for the student. Once the student has logged in , then he can see his CPI as it is a deciding factor for the qualification of the student. The registered students can directly login and check the job opportunities.

Defining the annual budget for oCEO positions is done by the administration for which we cannot hold any accountability as of now.

Feedback 2:-

Screenshots of the feedback from shared:



Feedback Form 2

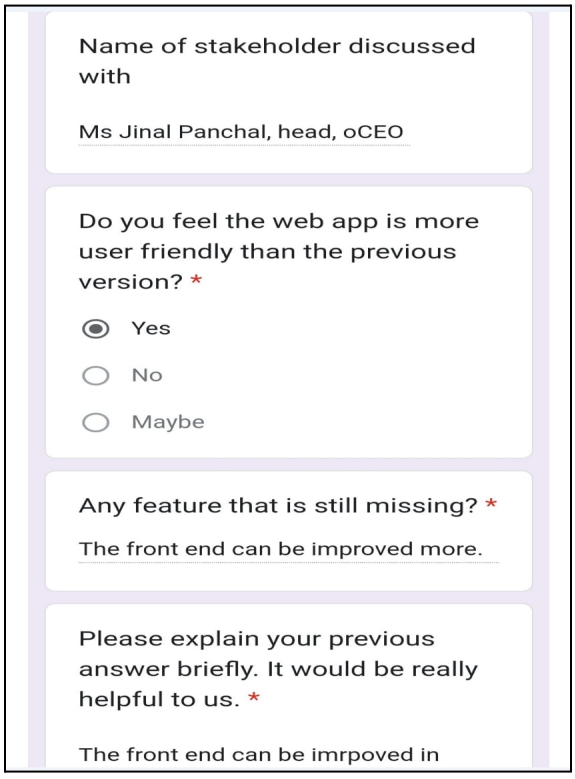
This is the final feedback form for the web app. Thanks for your giving constant suggestions through out the development process which gave us deeper insights into making the app user friendly.

Email *

chogle.ishani@iitgn.ac.in

Team name *

Oracles eleven



Name of stakeholder discussed with

Ms Jinal Panchal, head, oCEO

Do you feel the web app is more user friendly than the previous version? *

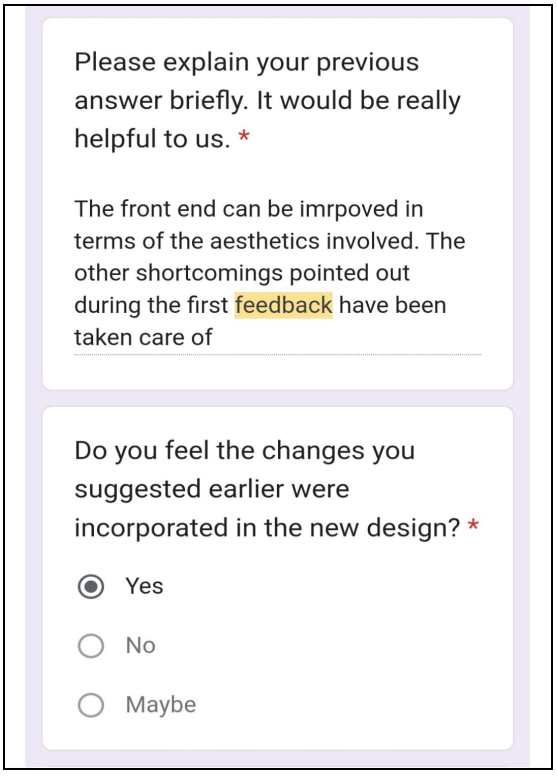
- (•) Yes
- ( ) No
- ( ) Maybe

Any feature that is still missing? *

The front end can be improved more.

Please explain your previous answer briefly. It would be really helpful to us. *

The front end can be imrpoved in

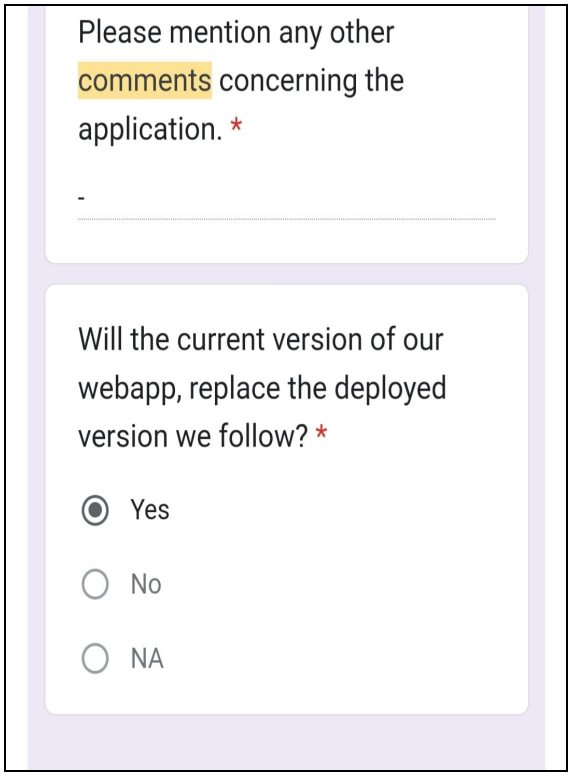


Please explain your previous answer briefly. It would be really helpful to us. *

The front end can be imrpoved in terms of the aesthetics involved. The other shortcomings pointed out during the first feedback have been taken care of

Do you feel the changes you suggested earlier were incorporated in the new design? *

- (•) Yes
- ( ) No
- ( ) Maybe



Please mention any other comments concerning the application. *

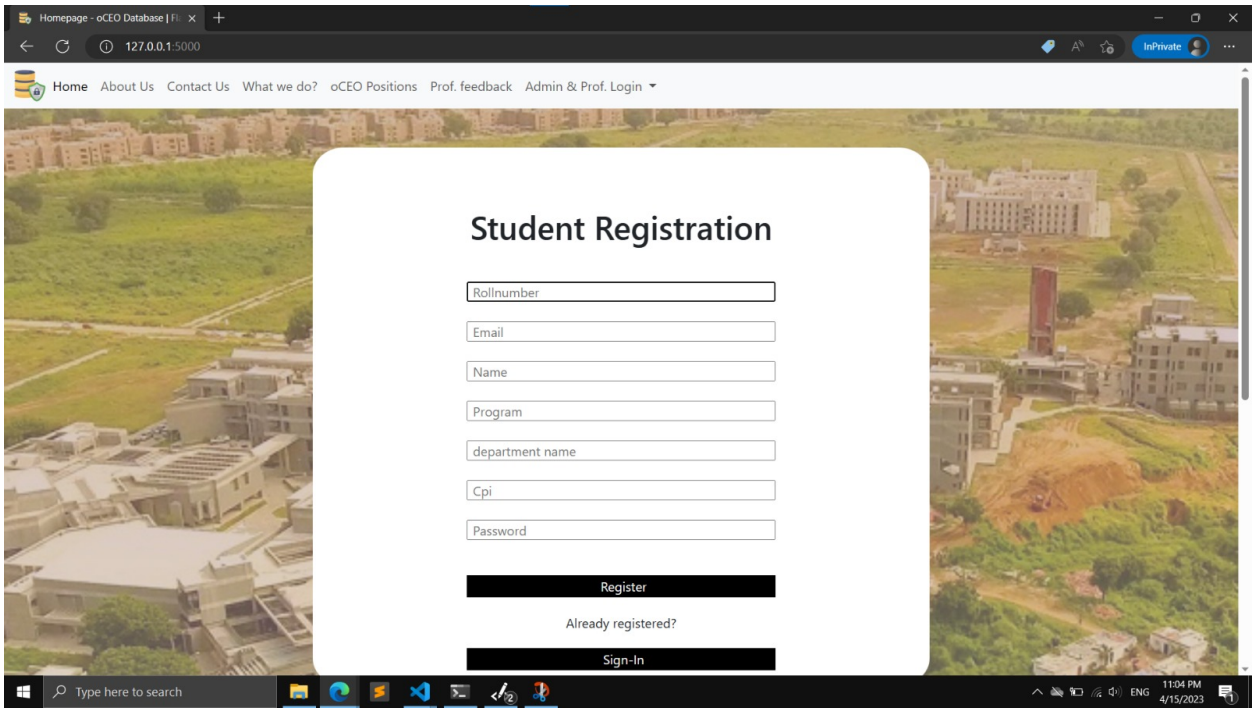
-

Will the current version of our webapp, replace the deployed version we follow? *

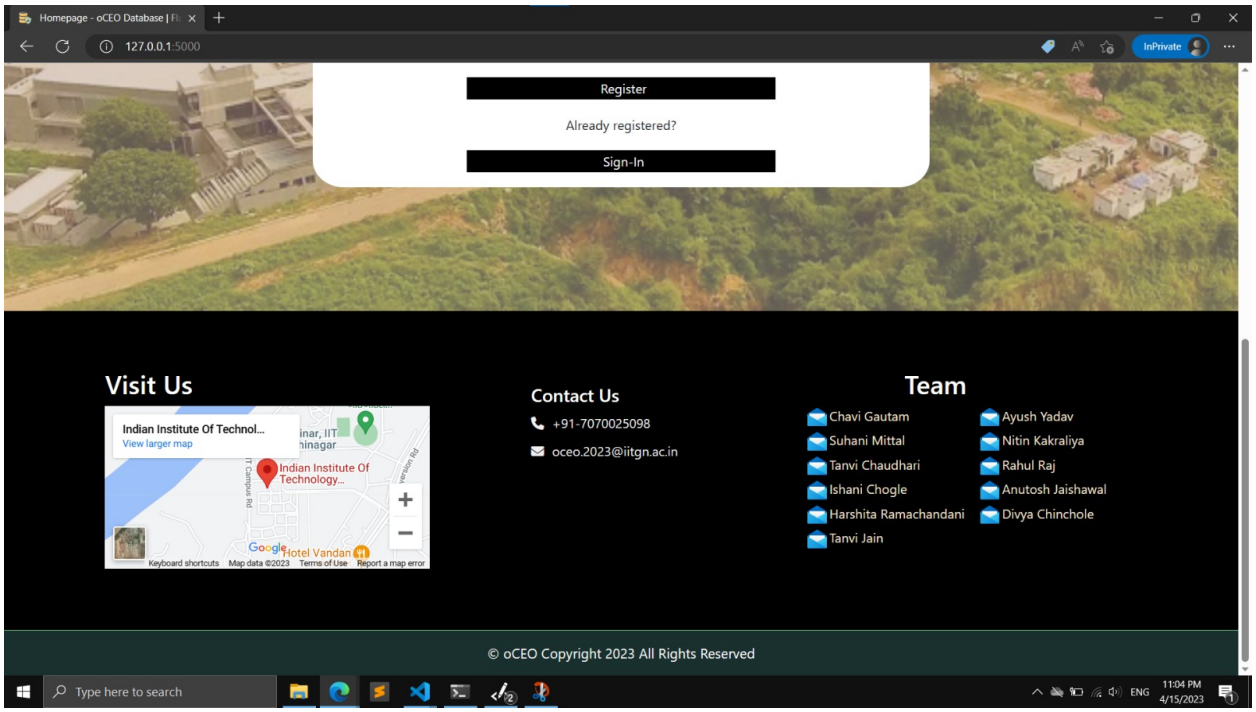
- (•) Yes
- ( ) No
- ( ) NA

After second feedback

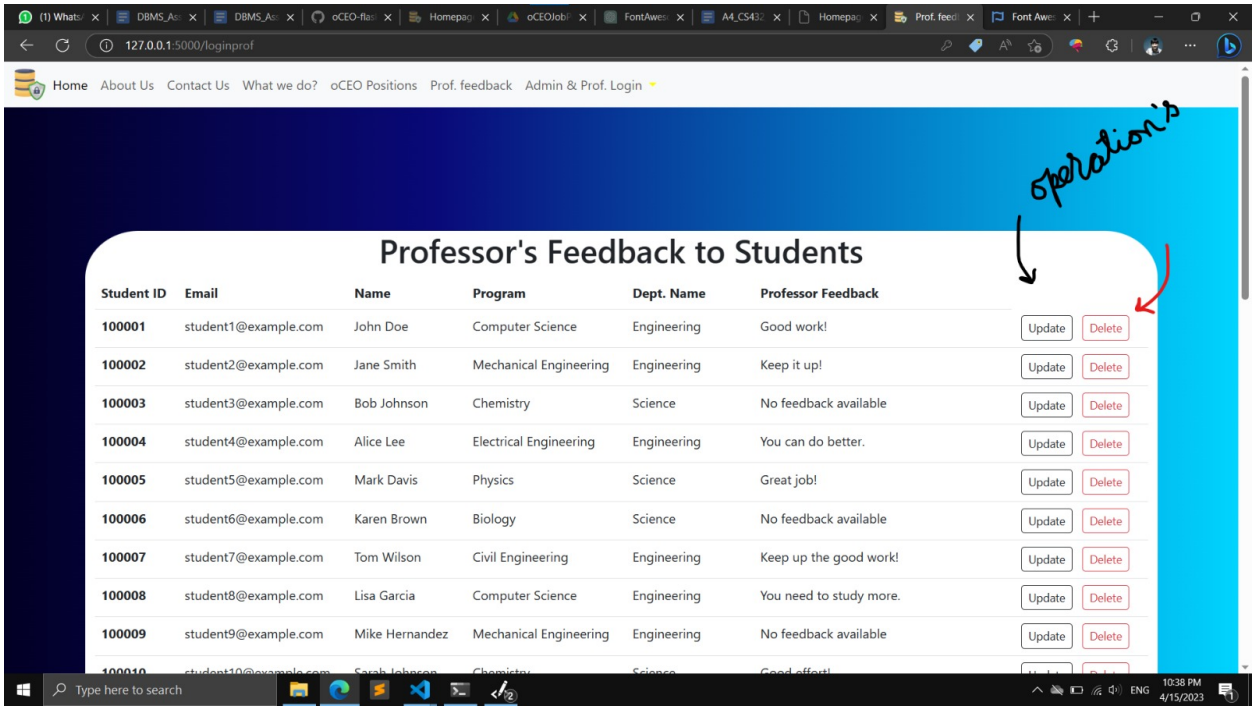
After making changes in frontend and adding professor feedback:-



Footer Section:



We also hyperlinked the email id's of all the team members and changed the background image for a better user experience.


127.0.0.1:5000/loginprof
Home
About Us
Contact Us
What we do?
oCEO Positions
Prof. feedback
Admin & Prof. Login
Operation's
Professor's Feedback to Students
Student ID
Email
Name
Program
Dept. Name
Professor Feedback
100001 student1@example.com John Doe Computer Science Engineering Good work!
100002 student2@example.com Jane Smith Mechanical Engineering Engineering Keep it up!
100003 student3@example.com Bob Johnson Chemistry Science No feedback available
100004 student4@example.com Alice Lee Electrical Engineering Engineering You can do better.
100005 student5@example.com Mark Davis Physics Science Great job!
100006 student6@example.com Karen Brown Biology Science No feedback available
100007 student7@example.com Tom Wilson Civil Engineering Engineering Keep up the good work!
100008 student8@example.com Lisa Garcia Computer Science Engineering You need to study more.
100009 student9@example.com Mike Hernandez Mechanical Engineering Engineering No feedback available
Update
Delete
Type here to search
ENG
10:38 PM
4/15/2023

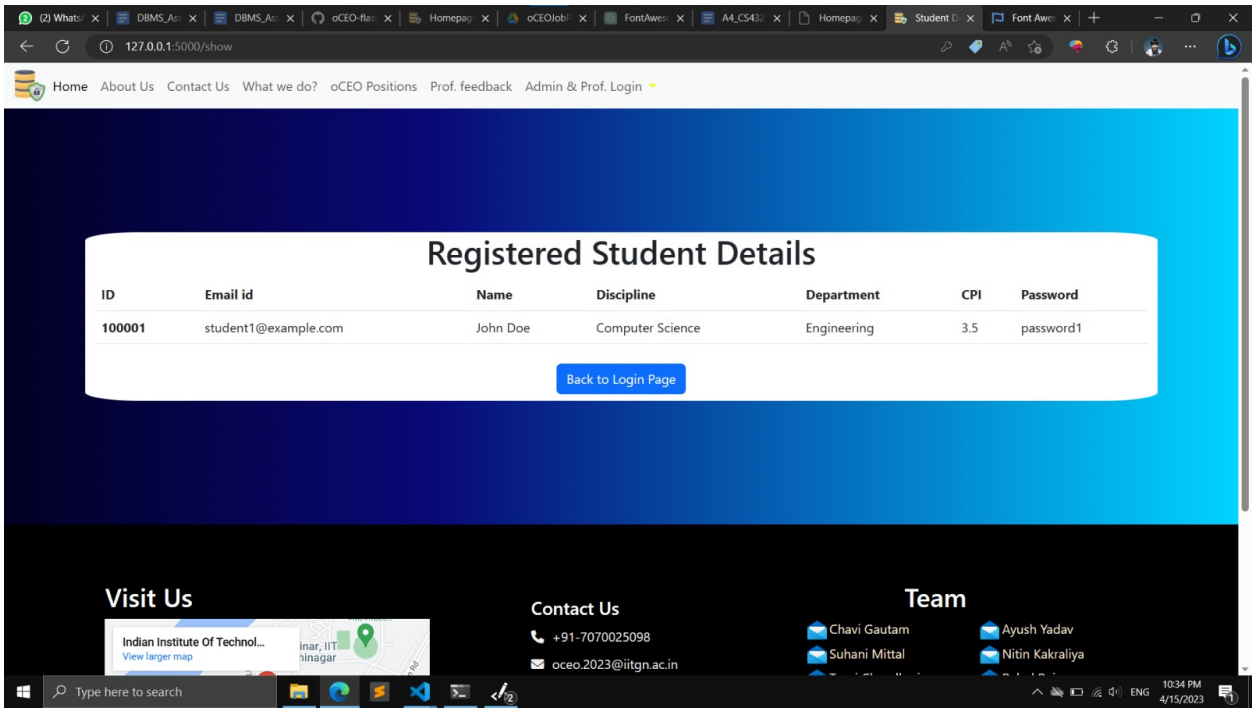
3.1 G1:

1. The feedback taken from the stakeholders has been shared above.
2. Attach screenshots of different views [along with a write-up on their privileges] of the database as seen by different classes of users.

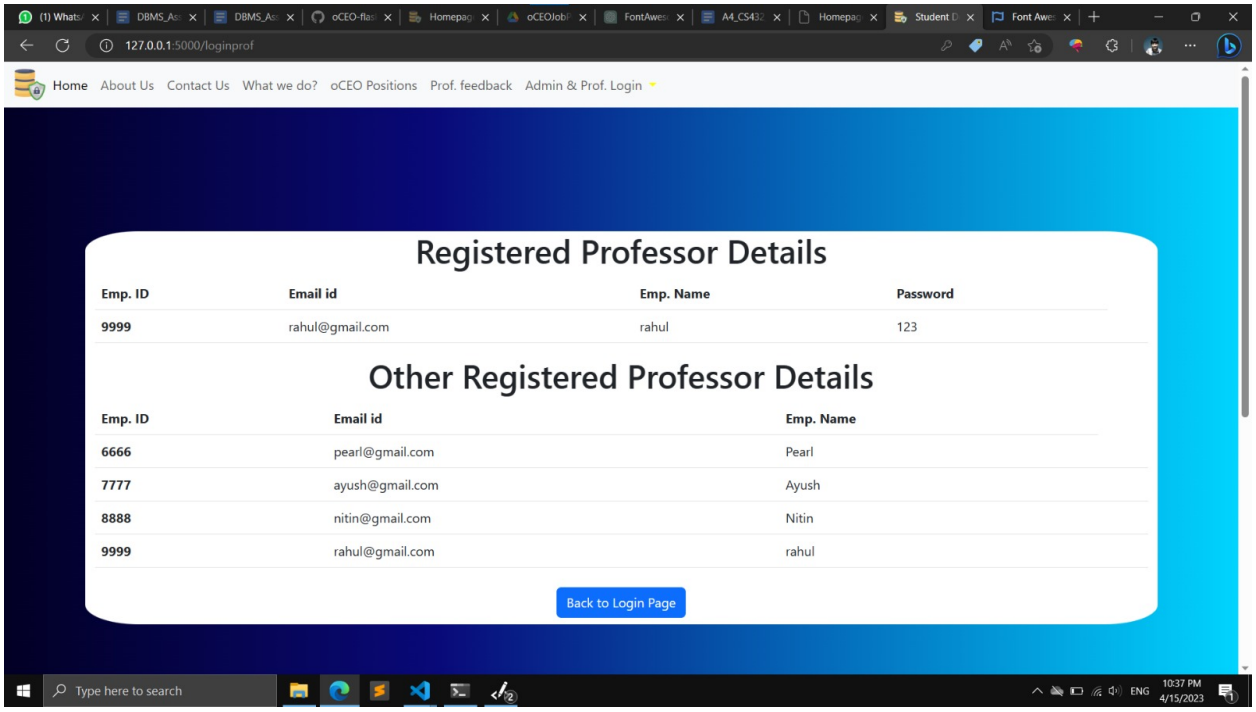
Answer:

View for Students:-

The student view allows the student to view and apply for job opportunities, update their profile, and view their application status. This view should be accessible and user-friendly to ensure that students can easily navigate the site and apply for job opportunities.

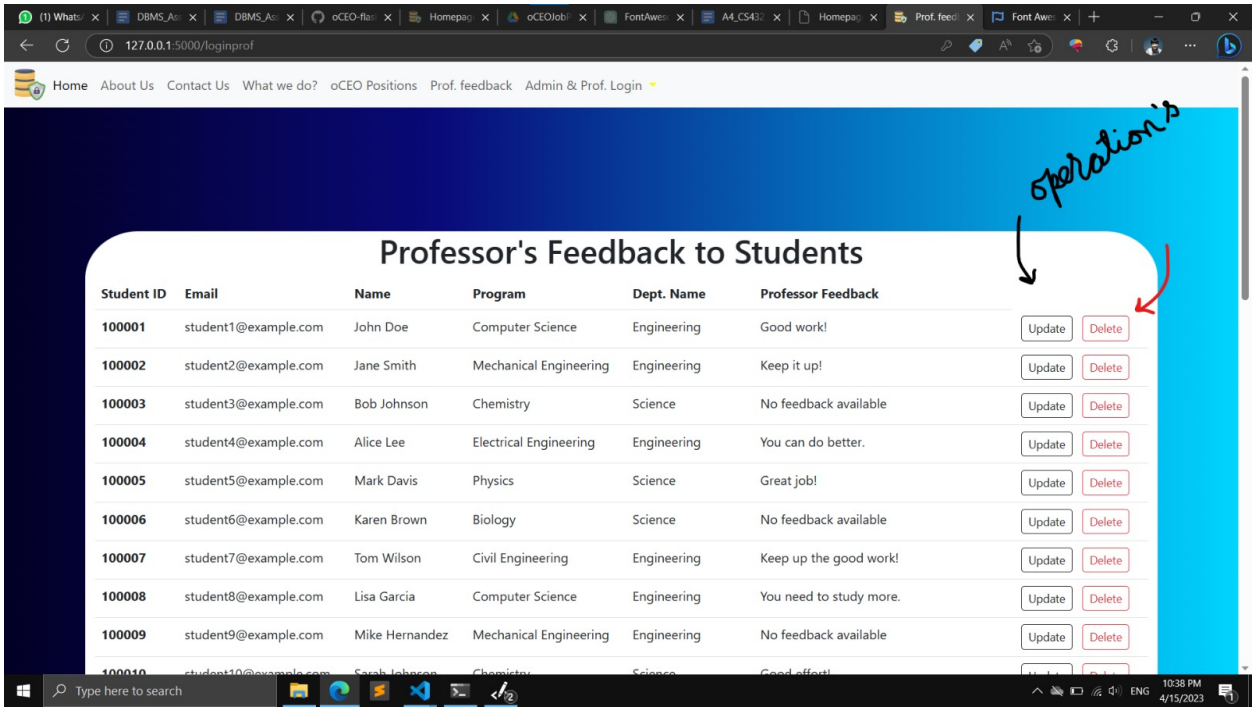


View for Professors:-

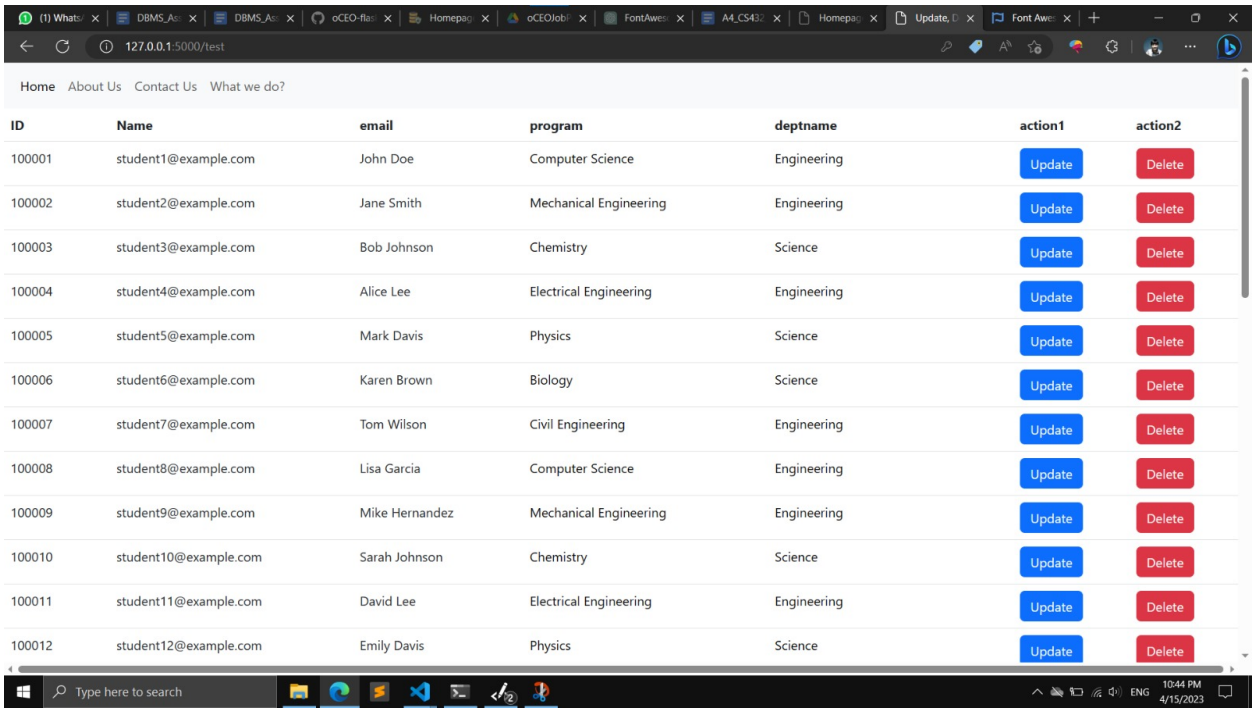


The professor view allows the professor to provide feedback on student job performance, view student profiles, and recommend job opportunities to their students. This view should also be highly secure to ensure that no unauthorized access occurs and that only designated professors can view and provide feedback on student performance.

After going on Check Feedback(For Professor):-



View for Admins:-



The admin view has the highest privileges and allows the admin to manage job postings, student profiles, and application statuses. This view should be highly secure to ensure that no unauthorized access occurs

and that only designated admins can manage the site's data. **The operation of the update and delete buttons are already shown in assignment 3 along with the images.**

3.2 Responsibility of G2:

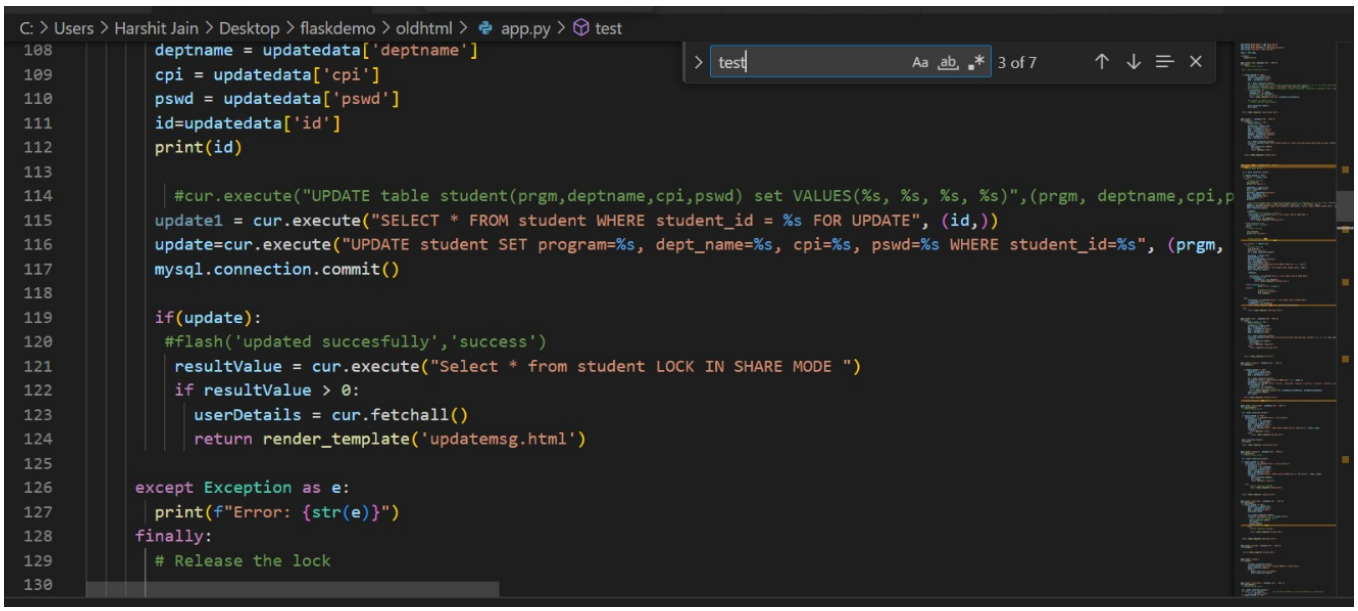
1. Concurrent multi-user access: Multiple users with different roles can access and update the database concurrently. In such a scenario, the same item can not be updated by two different users. For example, locks can be applied to tables in MySQL. **(10 Points)**

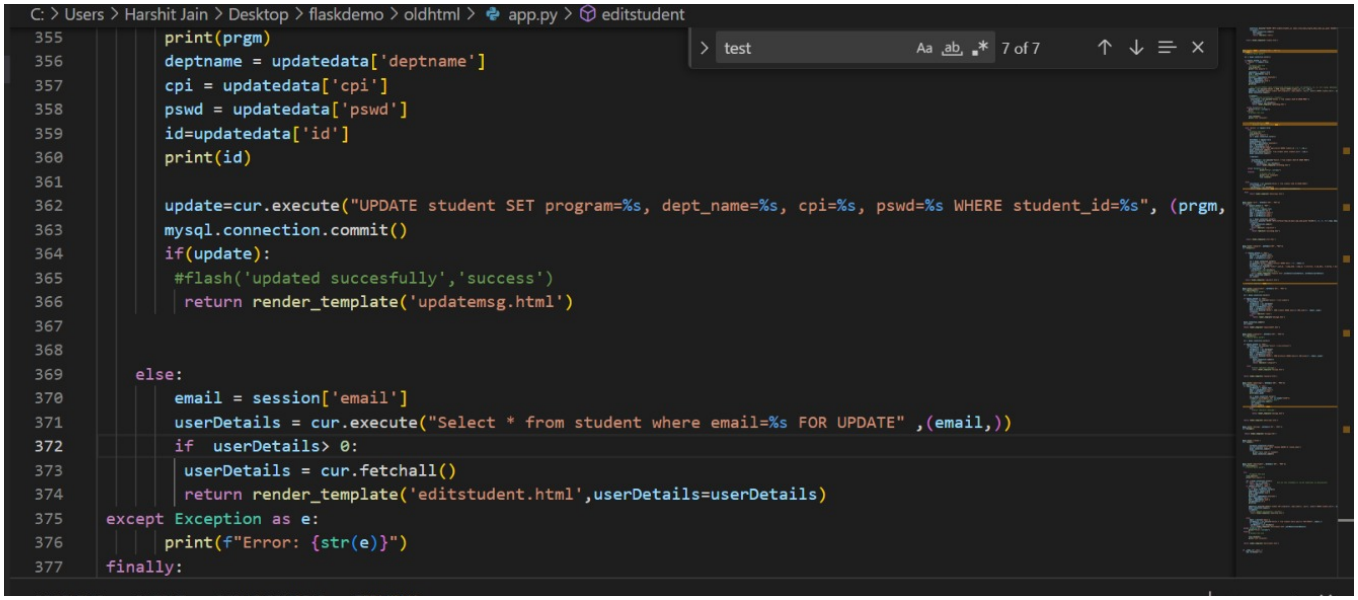
   A shared lock is a type of lock that allows multiple users to read the same data simultaneously but prevents any of them from modifying it. This type of lock is used to ensure that data is not changed while it is being accessed by multiple users, which can lead to data inconsistencies and errors.

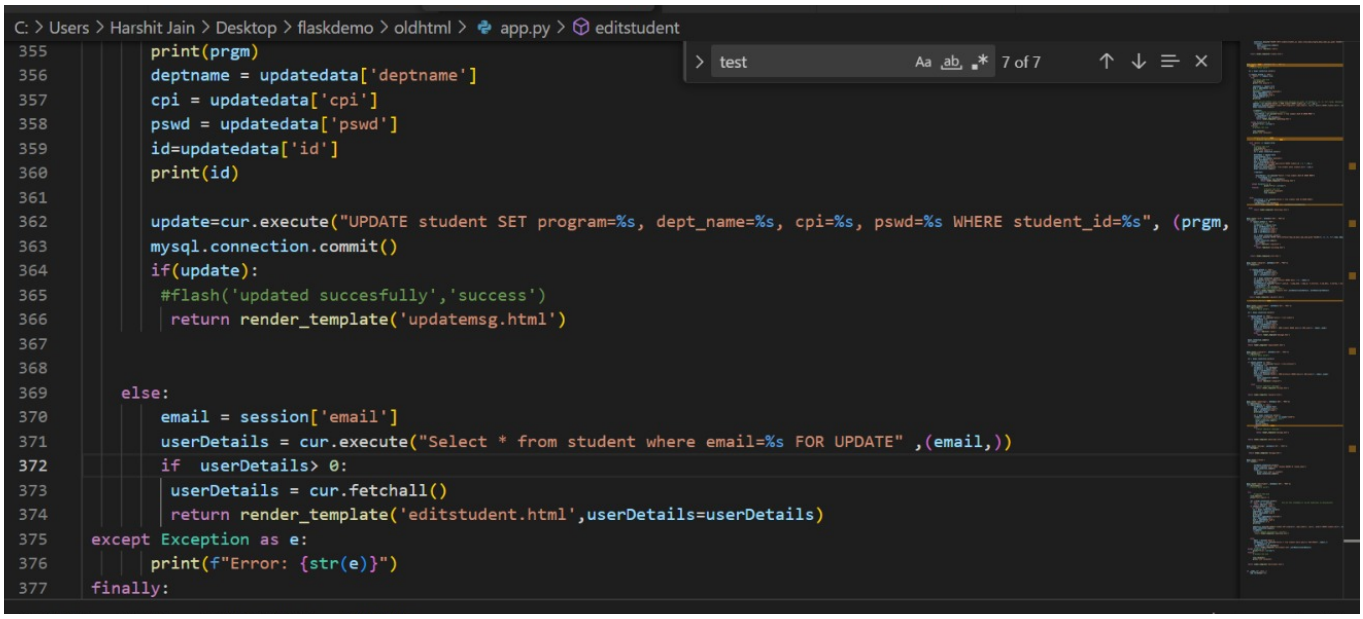
   An exclusive lock, on the other hand, is a type of lock that prevents any other user from accessing or modifying the data while the lock is in place. This type of lock is used when a user needs to modify the data and ensure that no other user can access or modify it until the modification is complete.

   In general, shared locks are used for read-only operations, while exclusive locks are used for write operations. By using these locks, the database management system ensures that multiple users can access and modify data simultaneously without causing data inconsistencies or errors. However, if these locks are not used correctly, they can lead to deadlocks, where multiple users are waiting for each other to release their locks, resulting in a system freeze. Therefore, it is essential to use these locks appropriately and carefully to ensure optimal database performance and prevent potential issues.

   Code:- IN SHARED MODE FOR UPDATES



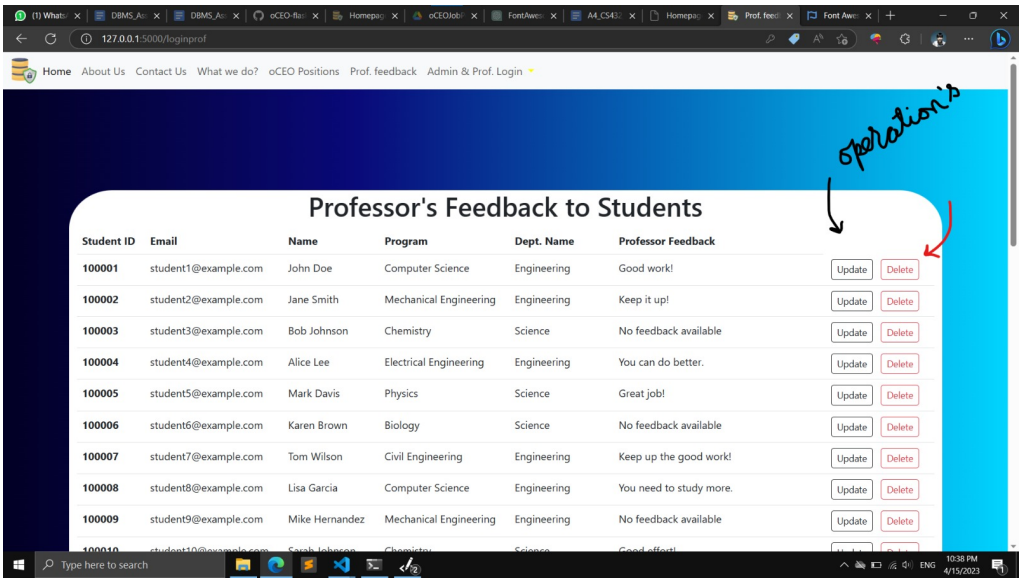

C: > Users > Harshit Jain > Desktop > flaskdemo > oldhtml > app.py > editstudent
test
7 of 7
print(prgm)
deptname = updatedata['deptname']
cpi = updatedata['cpi']
pswd = updatedata['pswd']
id=updatedata['id']
print(id)
update=cur.execute("UPDATE student SET program=%s, dept_name=%s, cpi=%s, pswd=%s WHERE student_id=%s", (prgm,
mysql.connection.commit()
if(update):
#flash('updated succesfully','success')
return render_template('updatemsg.html')
else:
email = session['email']
userDetails = cur.execute("Select * from student where email=%s FOR UPDATE" ,(email,))
if userDetails> 0:
userDetails = cur.fetchall()
return render_template('editstudent.html',userDetails=userDetails)
except Exception as e:
print(f"Error: {str(e)}")
finally:


C: > Users > Harshit Jain > Desktop > flaskdemo > oldhtml > app.py > editstudent
test
7 of 7
print(prgm)
deptname = updatedata['deptname']
cpi = updatedata['cpi']
pswd = updatedata['pswd']
id=updatedata['id']
print(id)
update=cur.execute("UPDATE student SET program=%s, dept_name=%s, cpi=%s, pswd=%s WHERE student_id=%s", (prgm,
mysql.connection.commit()
if(update):
#flash('updated succesfully','success')
return render_template('updatemsg.html')
else:
email = session['email']
userDetails = cur.execute("Select * from student where email=%s FOR UPDATE" ,(email,))
if userDetails> 0:
userDetails = cur.fetchall()
return render_template('editstudent.html',userDetails=userDetails)
except Exception as e:
print(f"Error: {str(e)}")
finally:

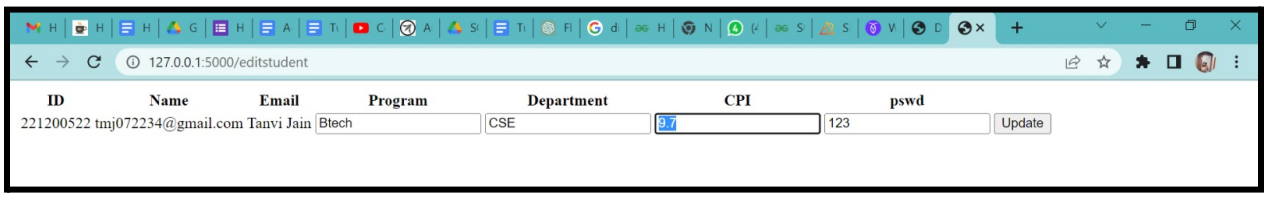
G2:

2. Implement the changes in the database as per the feedback received from stakeholders. **(30 Points)**

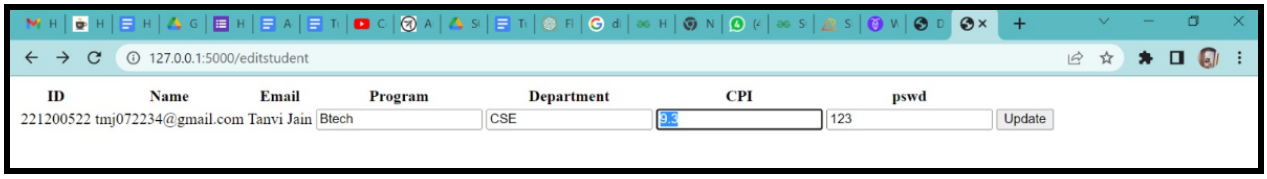
    1. **Professor Feedback Page:-**
    After receiving feedback from stakeholders, we added a page for professor feedback on our website. This page allows professors to provide feedback to students on their job performance. This feedback can help students understand their strengths and weaknesses, and it can also provide valuable insights for future job opportunities. The professor feedback page is designed to be user-friendly and accessible, allowing professors to quickly and easily provide feedback on student work. Overall, this new feature enhances the communication and collaboration between students and professors, contributing to a better learning and job-seeking experience.



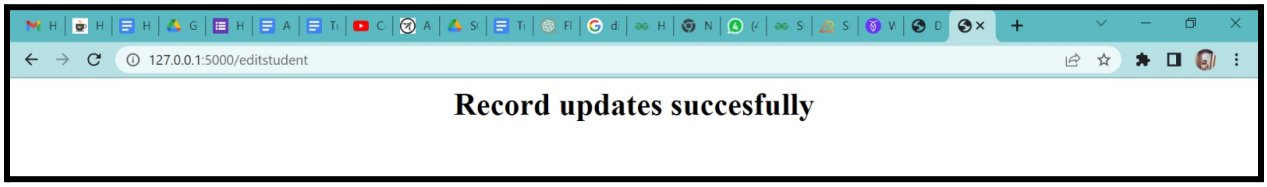
    2. **Edit student's profile**
    We added a feature that allows students to edit their profiles. With this feature, students can now easily update their information such as their name, mobile number, CPI, etc. This gives them greater control over their personal information and ensures that their profile remains accurate and up-to-date at all times. By allowing students to edit their profiles, we are providing them with a more user-friendly and personalized experience on our platform.



We update the profile of the student.



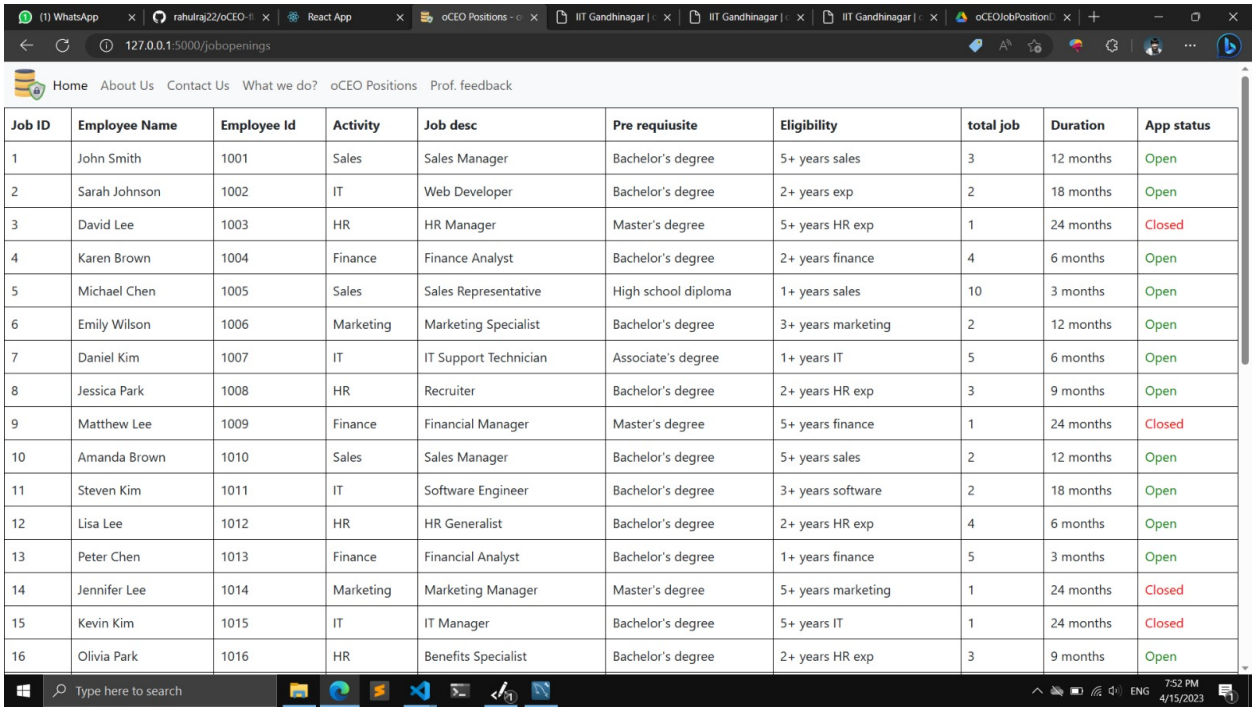
The ‘record updated successfully’ message pops up.



In the current system, a user or student may update their CPI at any moment by selecting the "update" button, which enables CPI modification. This functionality exemplifies dynamic execution since it enables database updates based on user input in real-time.

3. **Student application process**

   We added an apply option(hyperlink) after each job row. With this new feature, students can simply click on the apply option that is conveniently located next to each job listing, and they will be directed to the application form for that particular job. This feature will make it much easier for students to apply for jobs that they are interested in. Students will no longer have to navigate through several pages or scroll through a long list of jobs to find the one they want to apply for. This will save students time and effort and make the job application process more seamless and efficient.



Job ID	Employee Name	Employee Id	Activity	Job desc	Pre requiusite	Eligibility	total job	Duration	App status
1	John Smith	1001	Sales	Sales Manager	Bachelor's degree	5+ years sales	3	12 months	Open
2	Sarah Johnson	1002	IT	Web Developer	Bachelor's degree	2+ years exp	2	18 months	Open
3	David Lee	1003	HR	HR Manager	Master's degree	5+ years HR exp	1	24 months	Closed
4	Karen Brown	1004	Finance	Finance Analyst	Bachelor's degree	2+ years finance	4	6 months	Open
5	Michael Chen	1005	Sales	Sales Representative	High school diploma	1+ years sales	10	3 months	Open
6	Emily Wilson	1006	Marketing	Marketing Specialist	Bachelor's degree	3+ years marketing	2	12 months	Open
7	Daniel Kim	1007	IT	IT Support Technician	Associate's degree	1+ years IT	5	6 months	Open
8	Jessica Park	1008	HR	Recruiter	Bachelor's degree	2+ years HR exp	3	9 months	Open
9	Matthew Lee	1009	Finance	Financial Manager	Master's degree	5+ years finance	1	24 months	Closed
10	Amanda Brown	1010	Sales	Sales Manager	Bachelor's degree	5+ years sales	2	12 months	Open
11	Steven Kim	1011	IT	Software Engineer	Bachelor's degree	3+ years software	2	18 months	Open
12	Lisa Lee	1012	HR	HR Generalist	Bachelor's degree	2+ years HR exp	4	6 months	Open
13	Peter Chen	1013	Finance	Financial Analyst	Bachelor's degree	1+ years finance	5	3 months	Open
14	Jennifer Lee	1014	Marketing	Marketing Manager	Master's degree	5+ years marketing	1	24 months	Closed
15	Kevin Kim	1015	IT	IT Manager	Bachelor's degree	5+ years IT	1	24 months	Closed
16	Olivia Park	1016	HR	Benefits Specialist	Bachelor's degree	2+ years HR exp	3	9 months	Open

4. **Student and Professor Information:**

Student information

ID	Name	email	program	deptname	action1	action2		
001	johndoe@gmail.com	John Doe	BSCS	Computer Science	None	None	Update	Delete
002	janedoe@yahoo.com	Jane Doe	BBA	Business Administration	None	None	Update	Delete
003	joshuasmith@gmail.com	Joshua Smith	BSCE	Civil Engineering	None	None	Update	Delete
004	sarahbrown@gmail.com	Sarah Brown	BSEE	Electrical Engineering	None	None	Update	Delete
005	peterparker@gmail.com	Peter Parker	BSCS	Computer Science	None	None	Update	Delete
006	clarkkent@yahoo.com	Clark Kent	BSEE	Electrical Engineering	None	None	Update	Delete
007	brucewayne@gmail.com	Bruce Wayne	BBA	Business Administration	None	None	Update	Delete
008	dianaprince@yahoo.com	Diana Prince	BSCS	Computer Science	None	None	Update	Delete
009	tonystark@gmail.com	Tony Stark	BSEE	Electrical Engineering	None	None	Update	Delete
010	stephenstrange@yahoo.com	Stephen Strange	BSCE	Civil Engineering	None	None	Update	Delete
21	abc@gmail.com	Yam	Diploma	Product ENginerring	9.0	fadfaiji	Update	Delete

Professor Information

emp_ID	email	emp_Name	action1	action2	
001	johndoe@gmail.com	John Doe	BSCS	Update	Delete
002	janedoe@yahoo.com	Jane Doe	BBA	Update	Delete
003	joshuasmith@gmail.com	Joshua Smith	BSCE	Update	Delete
004	sarahbrown@gmail.com	Sarah Brown	BSEE	Update	Delete
005	peterparker@gmail.com	Peter Parker	BSCS	Update	Delete
006	clarkkent@yahoo.com	Clark Kent	BSEE	Update	Delete
007	brucewayne@gmail.com	Bruce Wayne	BBA	Update	Delete
008	dianaprince@yahoo.com	Diana Prince	BSCS	Update	Delete
009	tonystark@gmail.com	Tony Stark	BSEE	Update	Delete

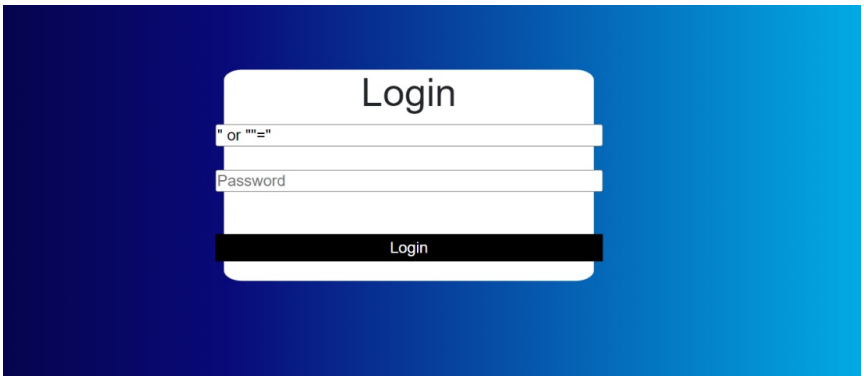
Before there were only student information on the admin page, now admin can see both student and professor information and update or delete them as per requirement.

3.3 Responsibility of G1 & G2: 20 Pts

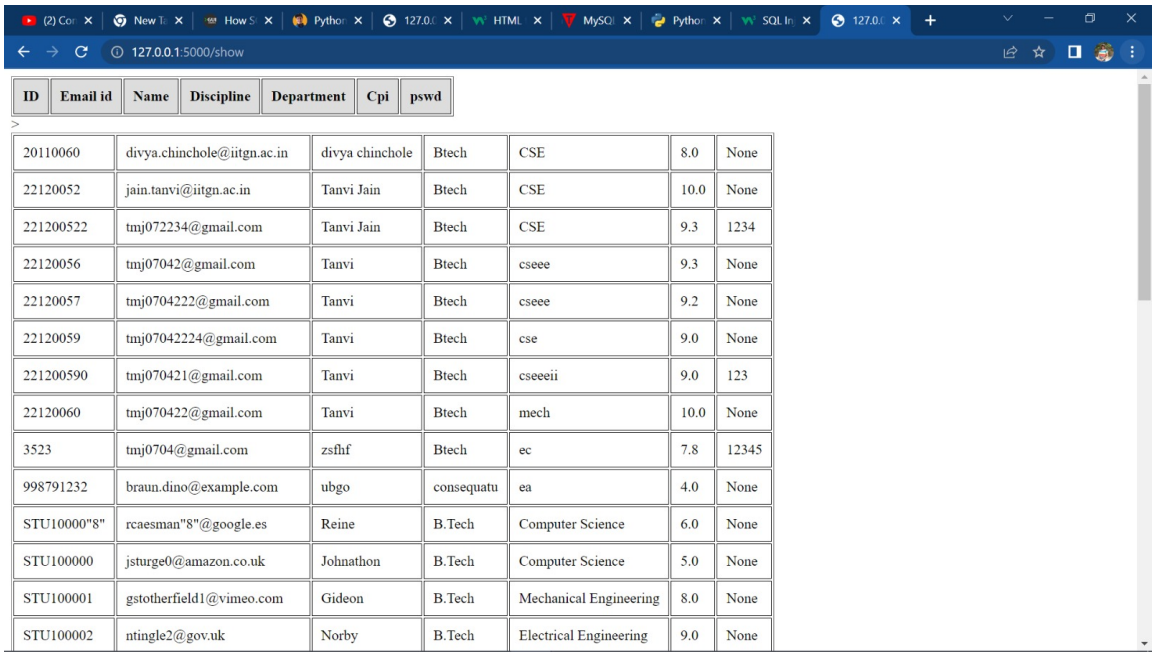
1. Documentation and screenshots of a total of 2 attacks [SQL Injection and XSS] performed and the defenses against those attacks. **(15 points)**
    a. Additional attack and defense will lead to 10 **bonus** points for the team. *(Maximum 20 bonus points)*

Ans:

1. SQL Injection:



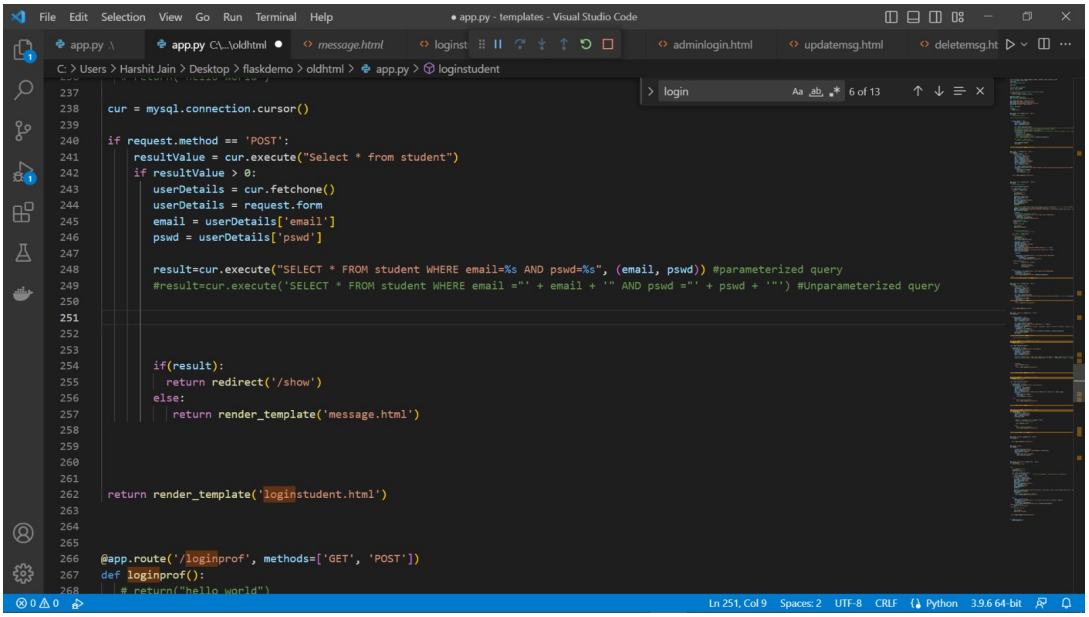
The query "or" "=" is often used as part of SQL injection attacks where an attacker tries to bypass authentication or access sensitive data by manipulating the query to modify its behavior.
The OR condition with an empty string value will always be true, effectively bypassing the authentication check and allowing the attacker to login as the admin user.
On running this query in input,



ID	Email id	Name	Discipline	Department	Cpi	pswd
20110060	divya.chinchole@iitgn.ac.in	divya chinchole	Btech	CSE	8.0	None
22120052	jain.tanvi@iitgn.ac.in	Tanvi Jain	Btech	CSE	10.0	None
221200522	tmj072234@gmail.com	Tanvi Jain	Btech	CSE	9.3	1234
22120056	tmj07042@gmail.com	Tanvi	Btech	cseee	9.3	None
22120057	tmj0704222@gmail.com	Tanvi	Btech	cseee	9.2	None
22120059	tmj07042224@gmail.com	Tanvi	Btech	cse	9.0	None
221200590	tmj070421@gmail.com	Tanvi	Btech	cseeeii	9.0	123
22120060	tmj070422@gmail.com	Tanvi	Btech	mech	10.0	None
3523	tmj0704@gmail.com	zsfhf	Btech	ec	7.8	12345
998791232	braun.dino@example.com	ubgo	consequatu	ea	4.0	None
STU10000"8"	rcaesman"8"@google.es	Reine	B.Tech	Computer Science	6.0	None
STU100000	jsturge0@amazon.co.uk	Johnathon	B.Tech	Computer Science	5.0	None
STU100001	gstotherfield1@vimeo.com	Gideon	B.Tech	Mechanical Engineering	8.0	None
STU100002	ntingle2@gov.uk	Norby	B.Tech	Electrical Engineering	9.0	None

Defense against these attacks:

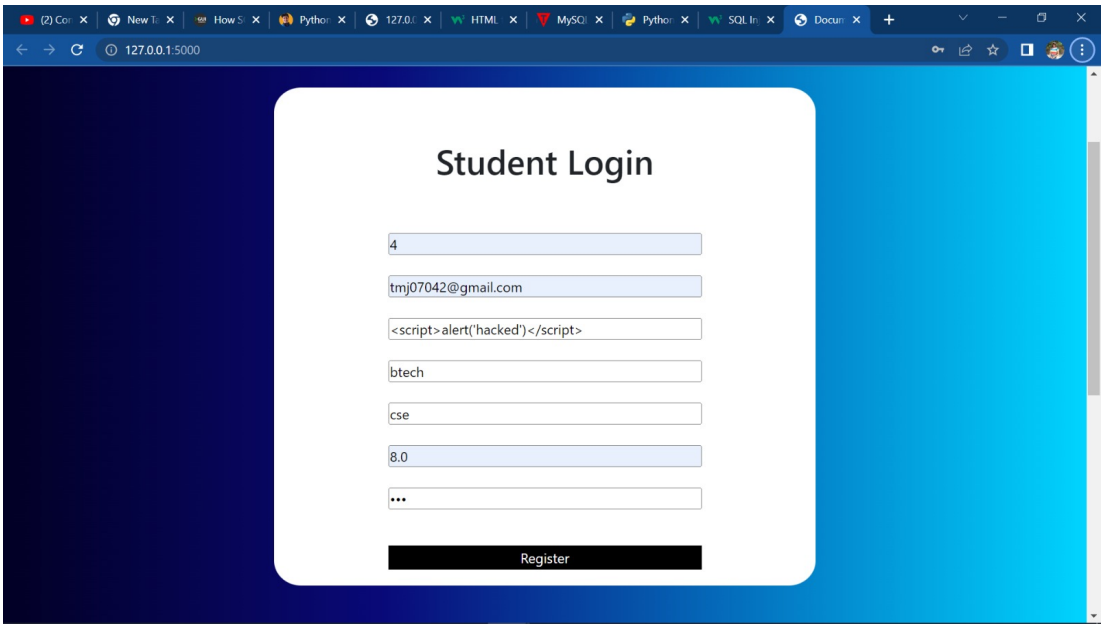
Parameterized SQL queries are an effective way to prevent SQL injection attacks in web applications.



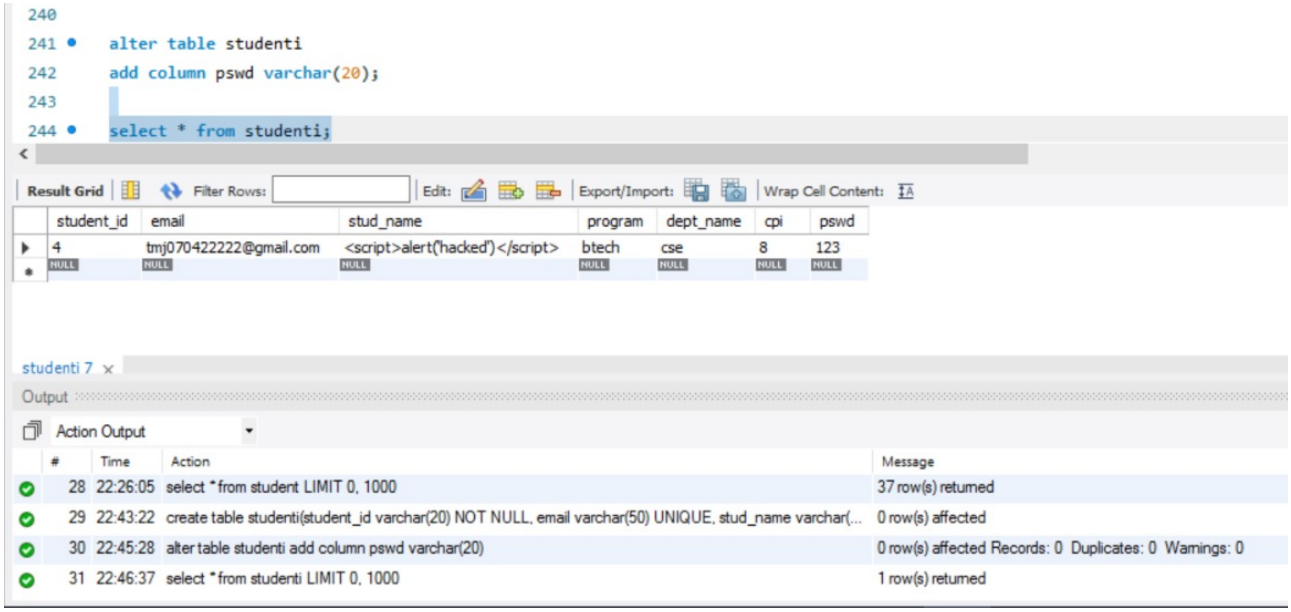
#the parameterized query used:
*"SELECT * FROM student WHERE email= %s AND pswd = %s", (email,pswd)"*

2. XSS Attack:

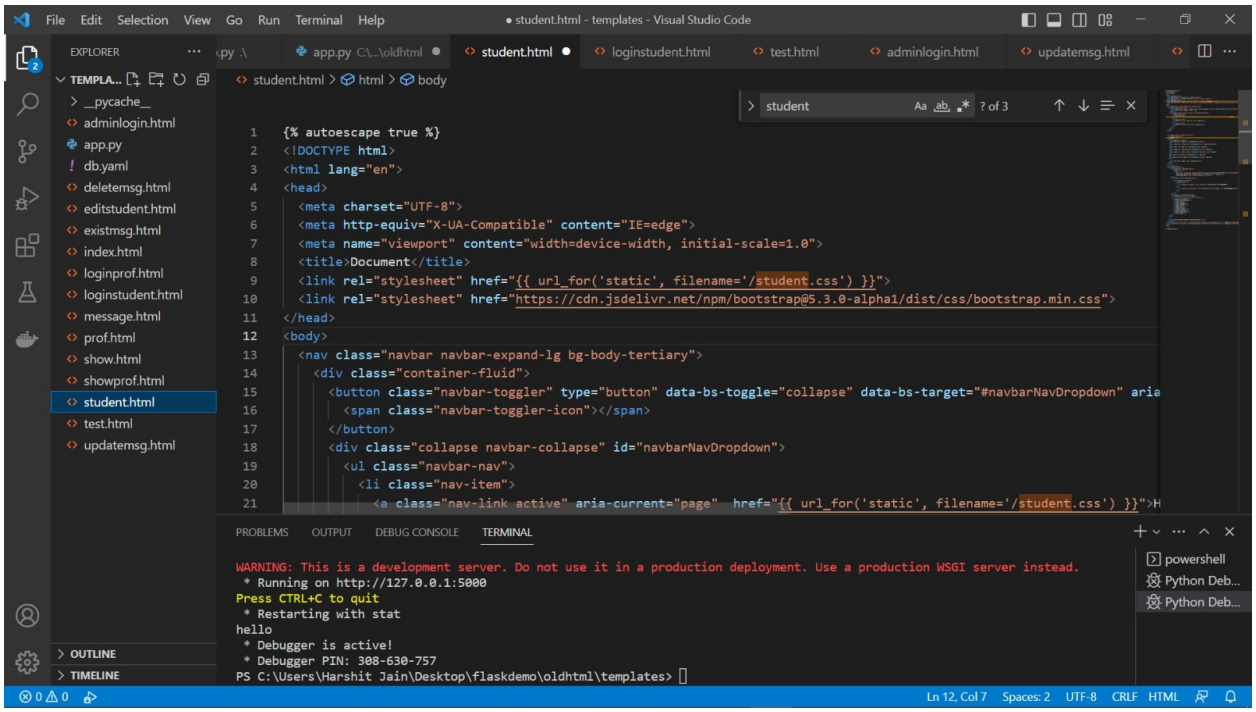
On giving input as:
<script>alert('hacked')</script>



The input gets saved as it is in our database.

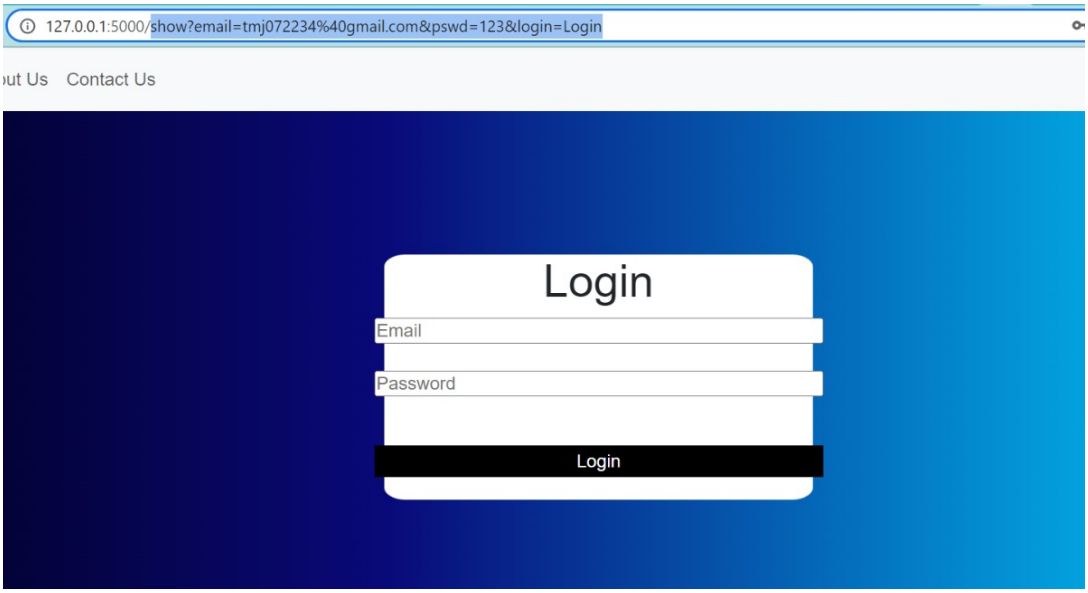


To prevent XSS attack, autoescape is used,

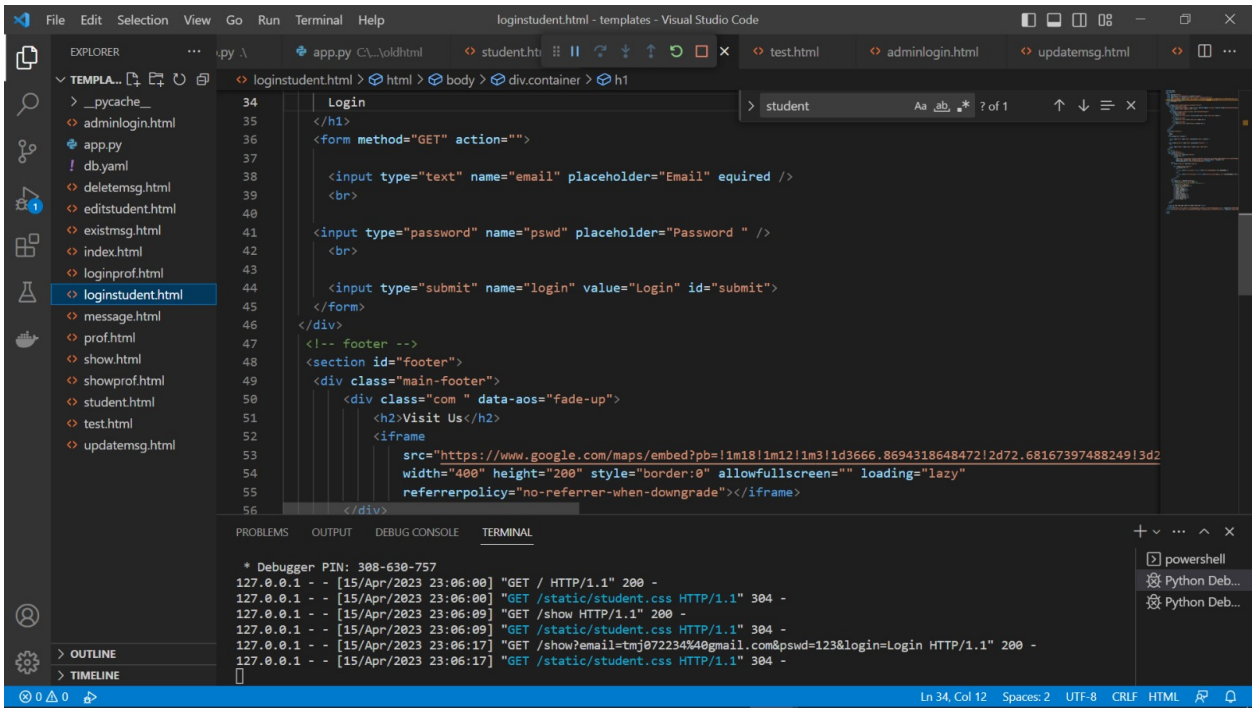


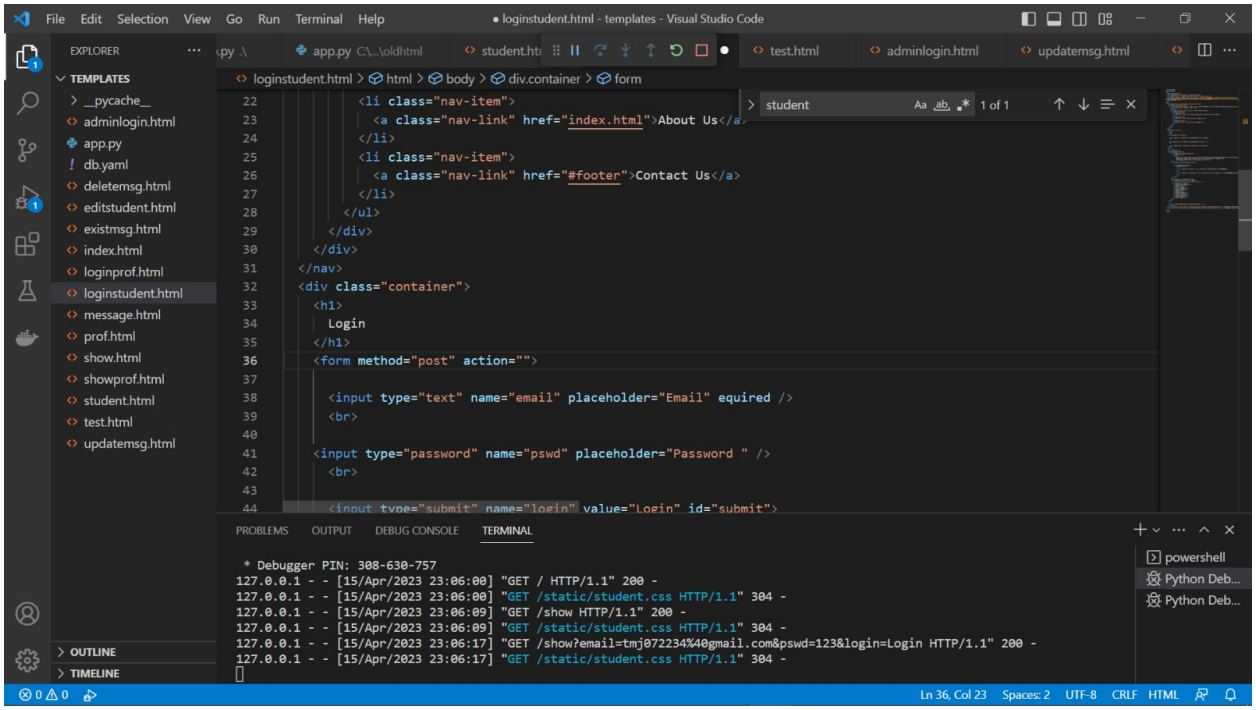
When autoescape is enabled, all user input that is rendered on a web page is automatically cleaned up so that characters like, >, and & are substituted with their respective HTML entities. These HTML entities will be displayed as plain text on the page rather than being executed as part of a script. This lessens the likelihood of malicious code injection and other XSS attacks.

3. GET POST updation:
   We have used GET before which shows email and password in the query string which can be harmful. As a solution we have used POST method that will not show the data in the query string like the password.



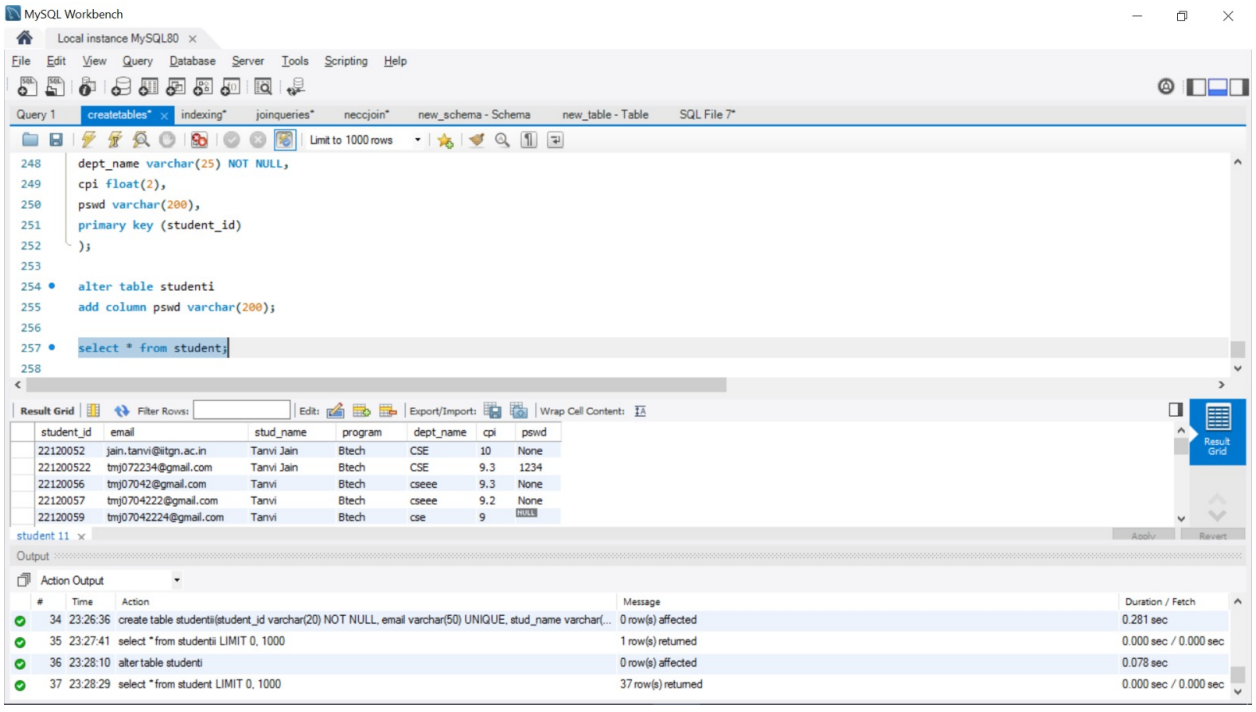
All the data (highlighted in the web page route) is in the querystring. Before, GET :





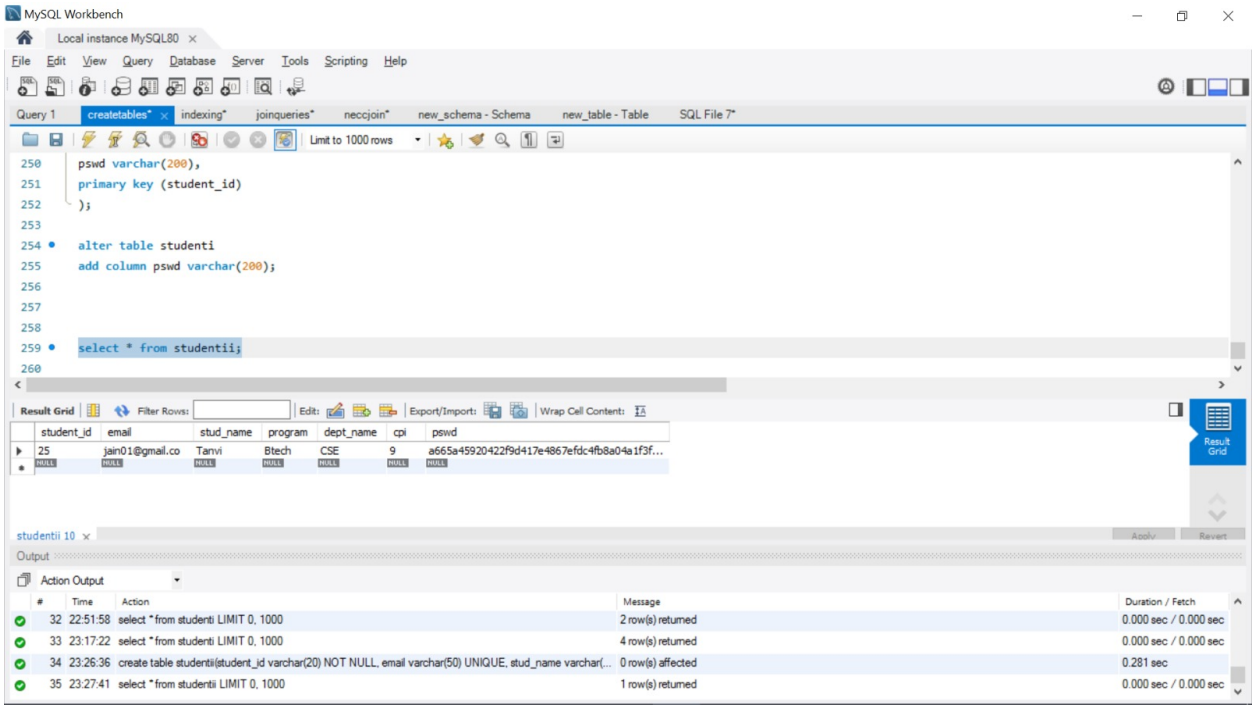
4. To prevent hackers from accessing the passwords of your website's users, password encryption is required. A user's password is encrypted into a hash when they establish an account, making it difficult to decode the password back to its original form.

Before: we havent encrypted the pswd which can be harmful for privacy

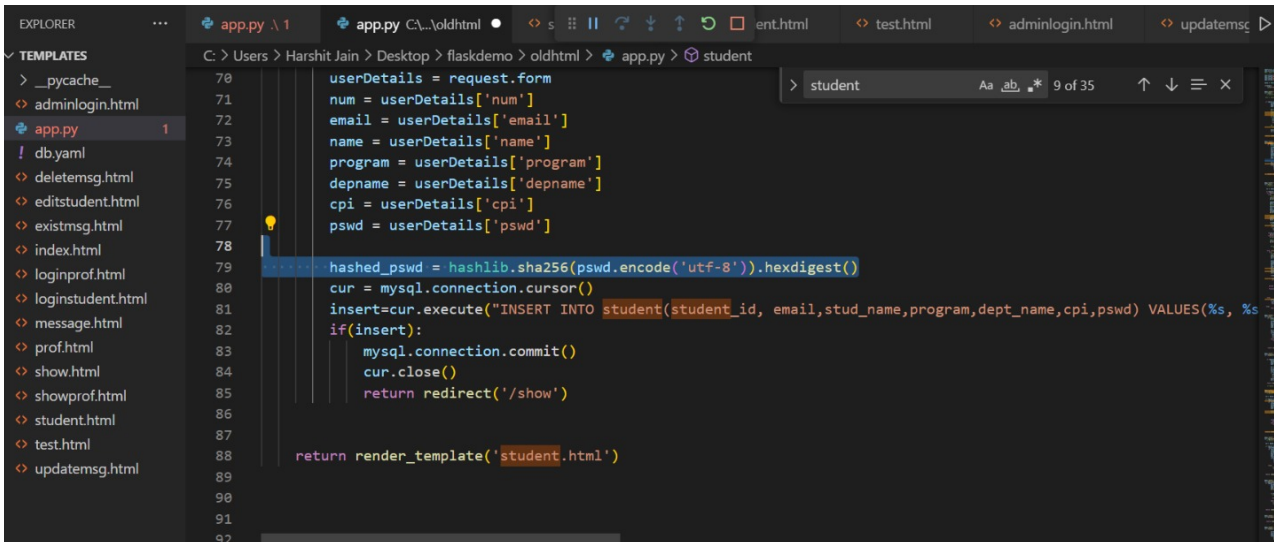


By doing this, you can make sure that even if an attacker manages to access the database of your website, they won't be able to see the real passwords. For password encryption, hashing algorithms like bcrypt, scrypt, and Argon2 are frequently used. However, to fully safeguard your website and its users, you should also implement other security measures like strict password policies and multi-factor authentication.

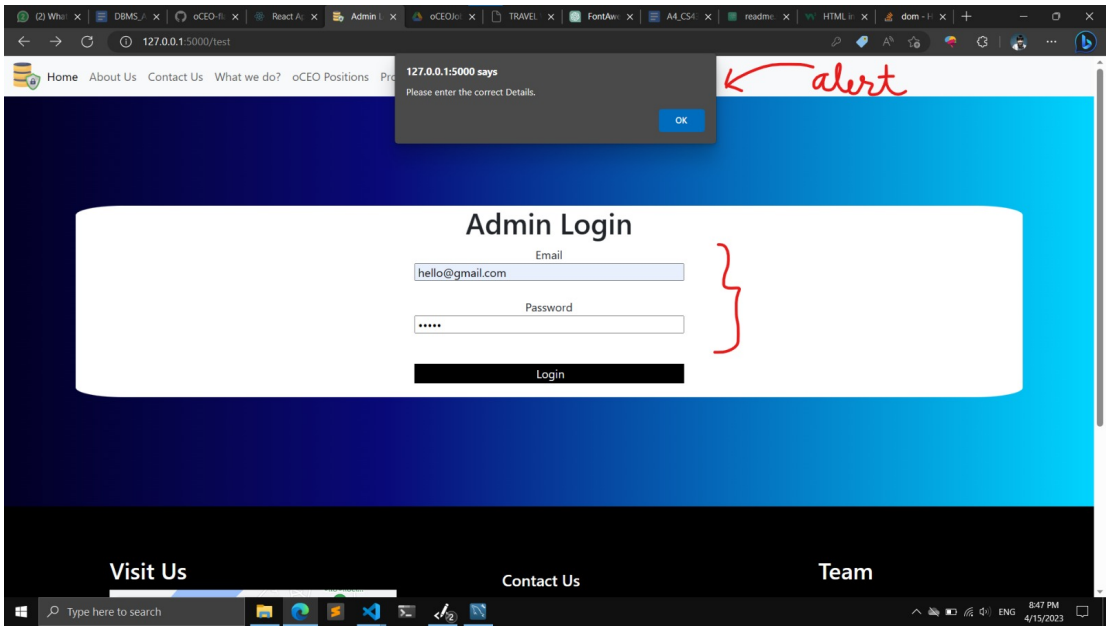
After Encryption: It was not required earlier but it was made required after encryption.

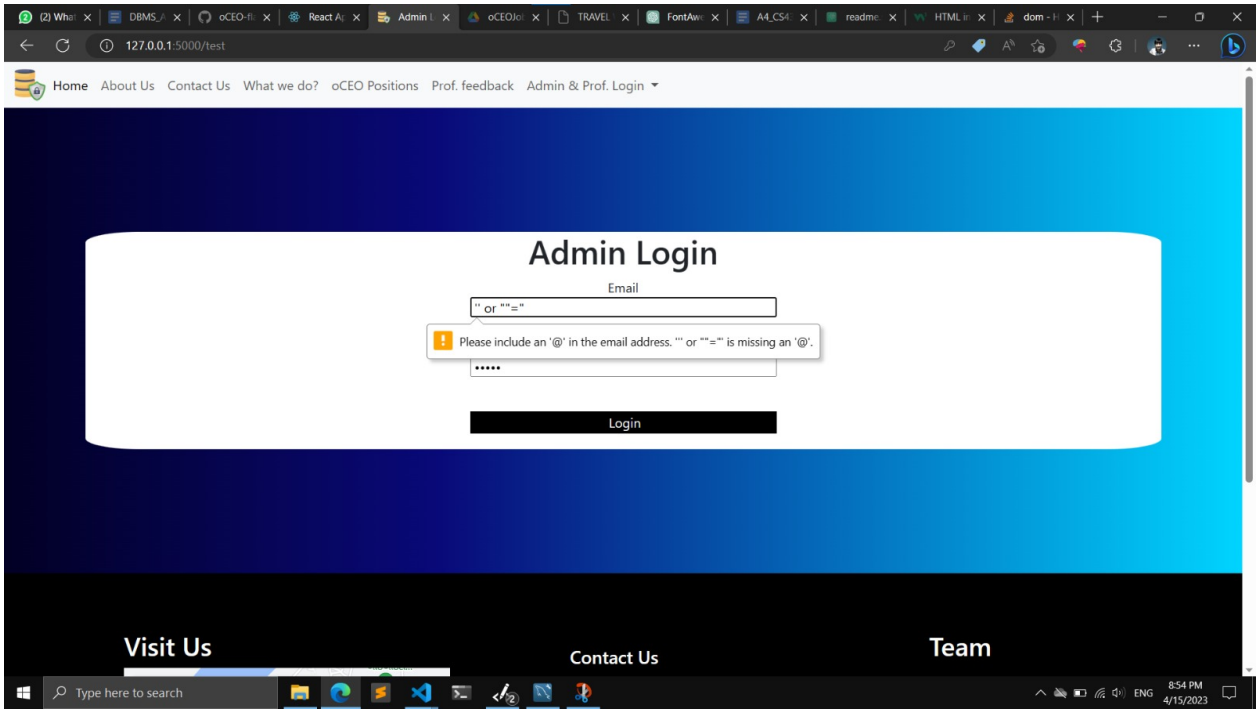


The code used for it is given below:



5. We have implemented several security measures on our admin page to ensure the safety of sensitive information. A login option has been added to restrict access to authorized users only. An alert box pops up when invalid login credentials are entered, preventing potential brute force attacks. Email and password input validation has been set up to only accept valid input and an alert box notifies the user if an SQL injection or incorrect details are entered. These measures work together to minimize unauthorized access to our admin page,



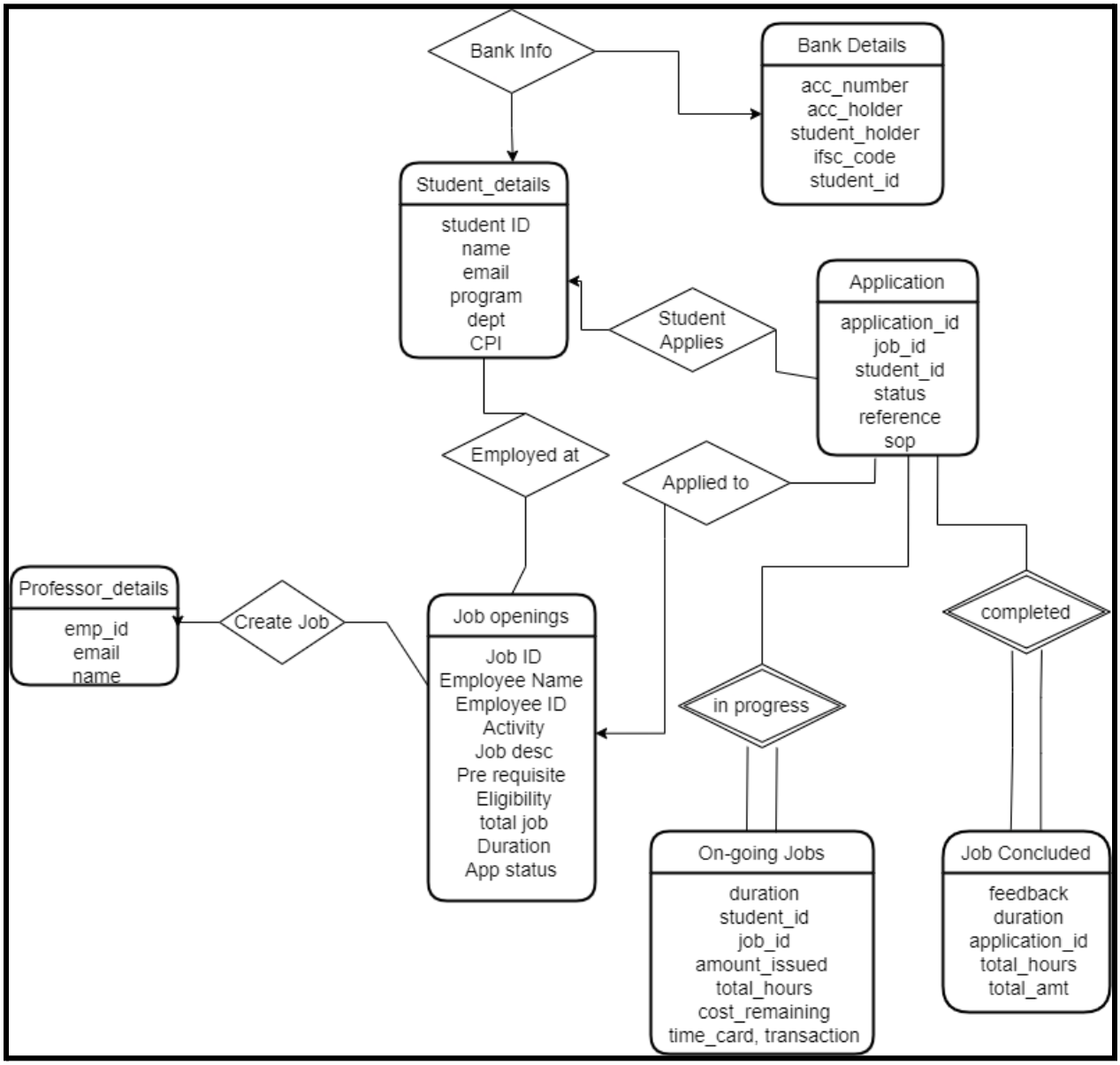


G1&G2:

2. Show that all the relations and their constraints, finalized after the second feedback, are present and valid as per the ER diagram constructed in Assignment 1.

Answer:

This is new ER diagram



And this table shows the all relationships between different tables as per in Assignment 1:

Entity	Relationships	Attributes
Student enrolled	Job Application: One to many(since one student can apply for more jobs) To professor: Many to one Ongoing and completed jobs– Many to one	Roll no ,program, student name ,program Dept,email,bank account, Previous job record
job	To student– many to one To applications–one to many	Eligibility, no. of seats, prerequisites, type of activity,job desc,current status
Applications	To student-one to one To professor–many to one	Jobid,studentid,reference,current status ,SOP
On going jobs	To students-one to many or many to one To professor–many to one	duration,student id,job id ,cost given,no of hours remaining,cost remaining ,transaction
Completed jobs	To students– one to many	Feedback,duration,student id ,job id,no of hours completed ,total amount
Bank details	To student id: one to one To jobs completed : one to one(use of foreign key)	Bank name,IFSC code,bank account no ,account holder name,student id
Professor associated	To jobs–One to many(since each professor can roll over many position)	Name,employee ,id department, Email ID

It is made sure that all constraints are held while running sql queries for our web app.

Contributions:-

Name	Contribution
Harshita Ramchandani	Assisted in the backend and verification of documentation.
Divya Chinchole	Assisted in the final documentation and checked the correctness of the final document. Also did the research on attacks and SQL injection.
Ishani Chogle	Assisted in front end coding and documentation
Tanvi Chaudhari	Assisted in the backend.
Chhavi Gautam	Integrated the whole work along with pictures and did the documentation and answer the questions . Helped with G1 and G1 &G2
Suhani Mittal	Helped in locating the earliest issues in the web app, contributed to the creation of the final document by responding to the questions in the G1, and G2, and compiling all the information together with images or screenshots.
Anutosh Jaishwal	Helped in finding the initial bugs throughout the web app. Contributed in the making of the final document including answering the questions of G1 and G1&G2 sections and compilation of all the data along with pictures.
Tanvi Jain	Contributed in backend, performed all attacks [SQL Injection and XSS and additional too] and the defenses against those attacks . Implemented multi user access(G2) and edit profile feature.
Rahul Raj	Improved the backend from SQL injection using truthy values (admin' OR '1' = '1) and similar other injections, also added strict check while admin login(through form). Added professor feedback for all students and

	also actions like update(to update feedback) and delete(to delete the feedback, etc)
Ayush Yadav	Helped in changing the frontend like changing the background ,the login form.Improved the contact/footer section and included various features for better and efficient user experience.
Nitin kakraliya	Helped in the frontend part especially the questions raised in the second part like removing three different login tabs on the front page and converting them into single navbar for better user experience.