



Indian Institute of Technology, Gandhinagar

## **CS 432: DATABASES**

---

### **Assignment 3: Building the Web App**

#### **Institute Phone Directory**

---

#### **Group - 14**

Aravind Krishna - 20110021

Arun Mani - 20110023

Hrishikesh CP -20110078

Mihika Jadhav - 20110113

Madhav Biswam -20110103

Rahul R Pai - 20120155

Shrija Agrawal - 20110197

Abhiram Geddam - 19110149

Kanishk Singh - 19110089

Ruthvik Mula - 19110124

Ninad Shah - 19110126

Prateek Kumar Jha - 19110159

***under the guidance of Prof. Mayank Singh***

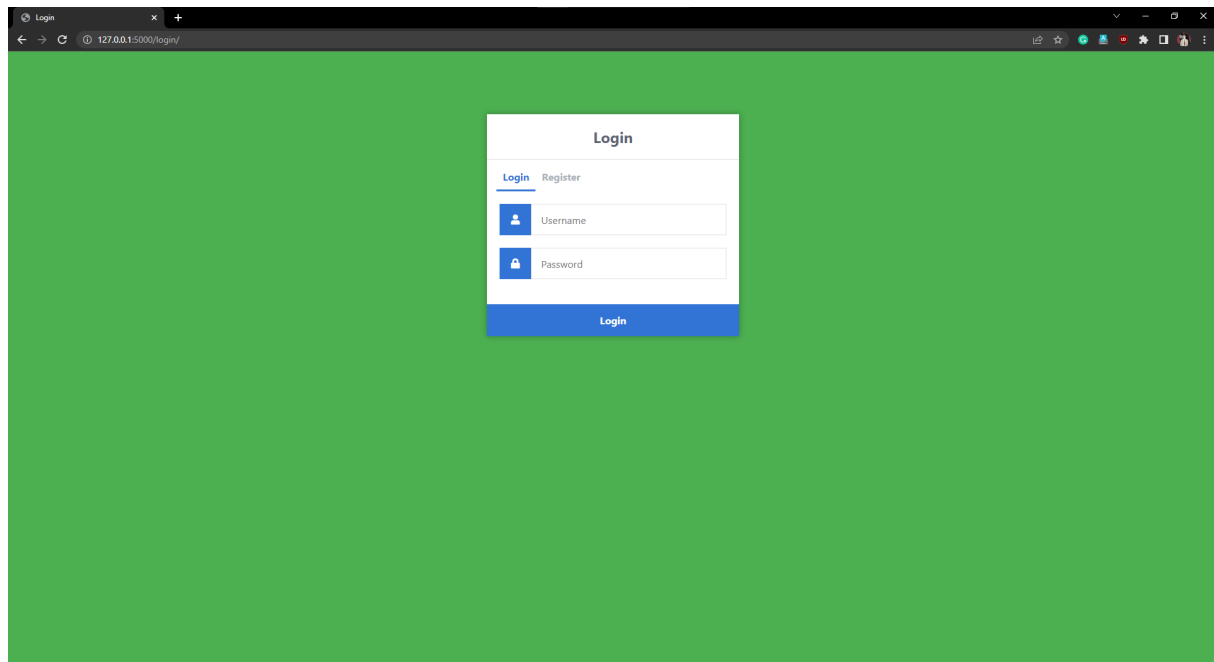
### 3. Tasks

#### 3.1 Responsibility of G1:

40 Pts.

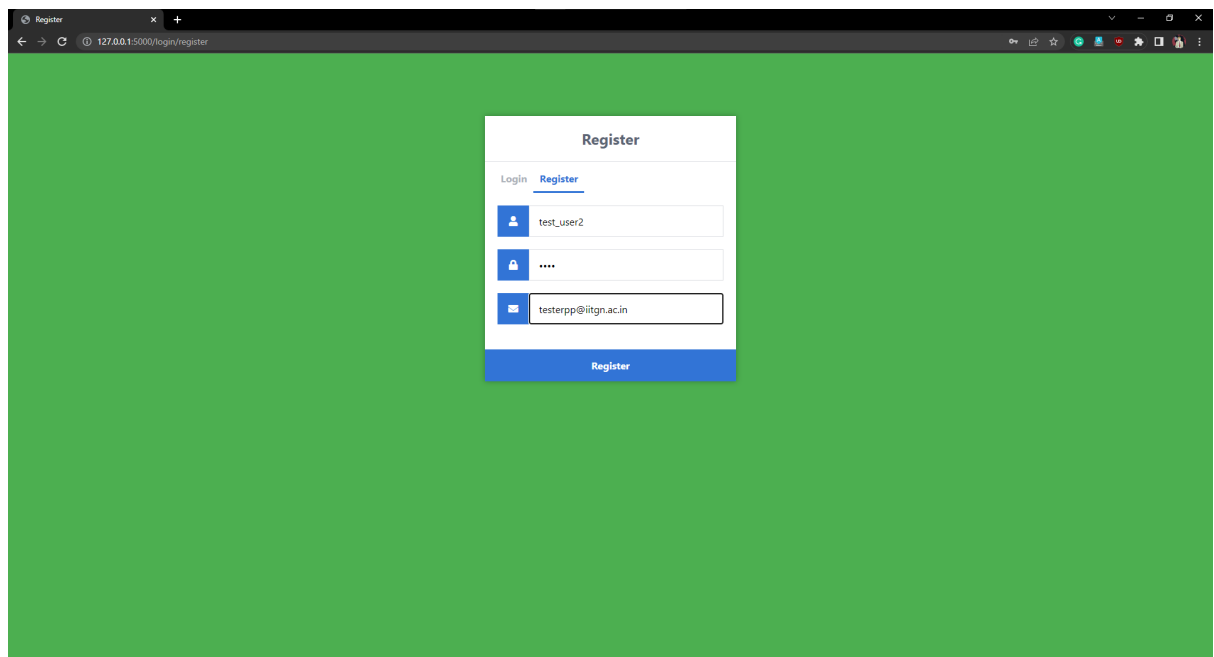
G1 is supposed to design the front end and beautify accordingly using HTML/CSS/JS, you may use any other libraries for the front end, such as Bootstrap or jQuery.

##### Startup Page



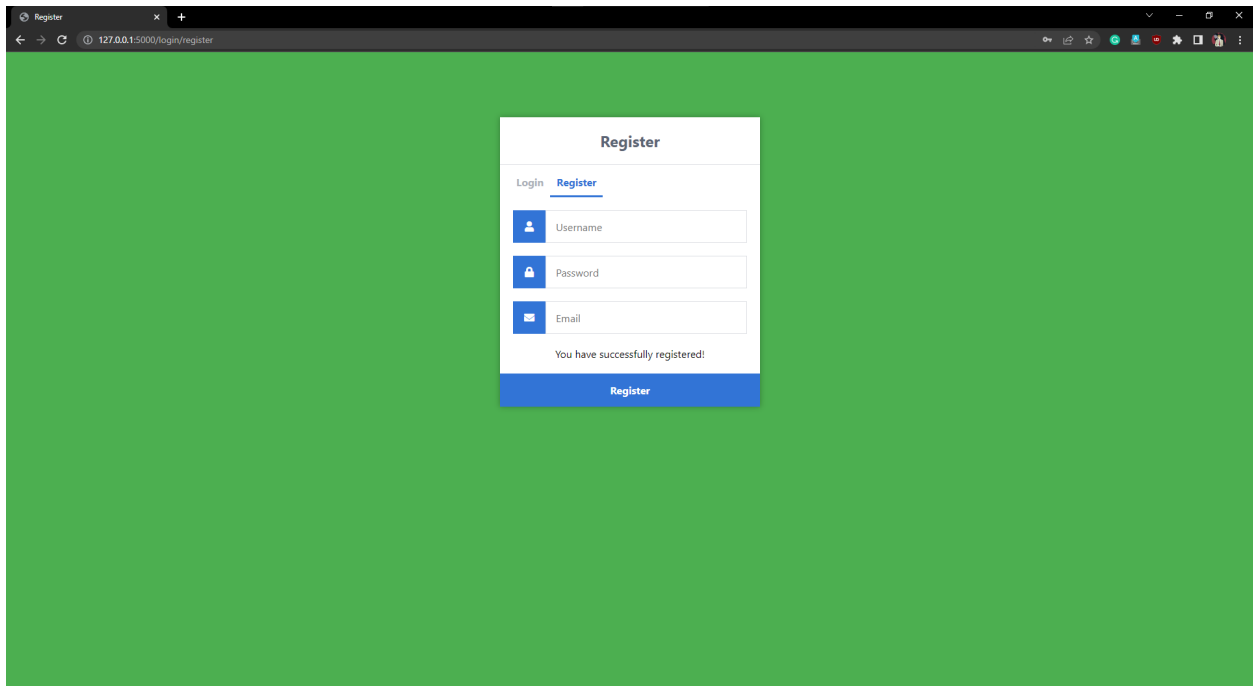
The screenshot shows a web browser window with the address bar displaying "127.0.0.1:5000/login/". The page has a solid green background. In the center, there is a white rectangular form titled "Login". Inside the form, there are two tabs: "Login" (which is selected and underlined) and "Register". Below the tabs, there are two input fields: "Username" with a blue user icon on the left, and "Password" with a blue lock icon on the left. At the bottom of the form is a blue button labeled "Login".

##### Register Page



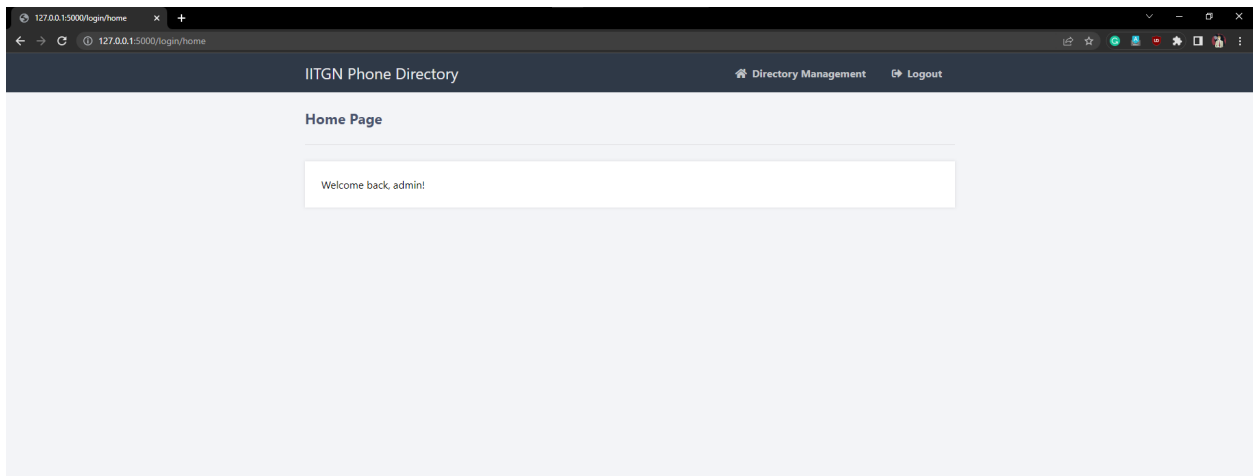
The screenshot shows a web browser window with the address bar displaying "127.0.0.1:5000/register". The page has a solid green background. In the center, there is a white rectangular form titled "Register". Inside the form, there are two tabs: "Login" and "Register" (which is selected and underlined). Below the tabs, there are three input fields: "test\_user2" with a blue user icon on the left, a password field with four asterisks and a blue lock icon on the left, and an email field with "testerpp@itgn.ac.in" and a blue envelope icon on the left. At the bottom of the form is a blue button labeled "Register".

## Registered successfully



The screenshot shows a web browser window with the address bar displaying "127.0.0.1:5000/login/register". The page has a solid green background. In the center, there is a white rectangular box titled "Register". Inside this box, there are two tabs: "Login" and "Register", with "Register" being the active tab. Below the tabs, there are three input fields, each with a blue icon on the left: a person icon for "Username", a lock icon for "Password", and an envelope icon for "Email". Below these fields, a message states "You have successfully registered!". At the bottom of the white box is a blue button labeled "Register".

## Admin's Home Page

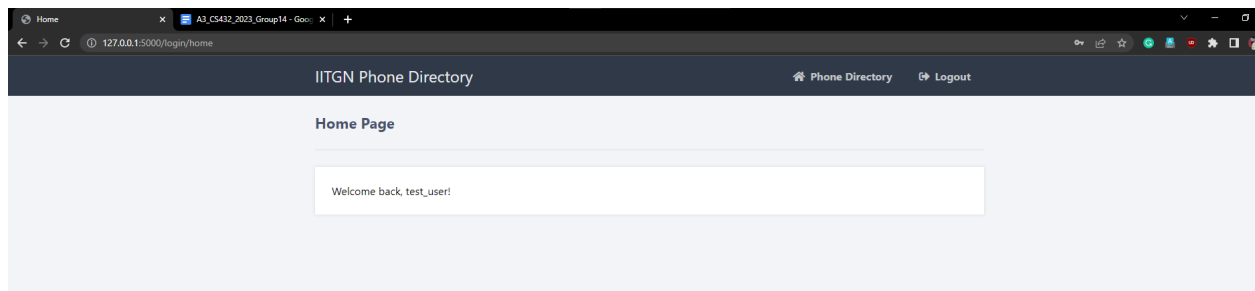


The screenshot shows a web browser window with the address bar displaying "127.0.0.1:5000/login/home". The page has a dark blue header bar with the text "IITGN Phone Directory" on the left and "Directory Management" and "Logout" on the right. Below the header, the page is titled "Home Page". In the center of the page, there is a white rectangular box containing the text "Welcome back, admin!".

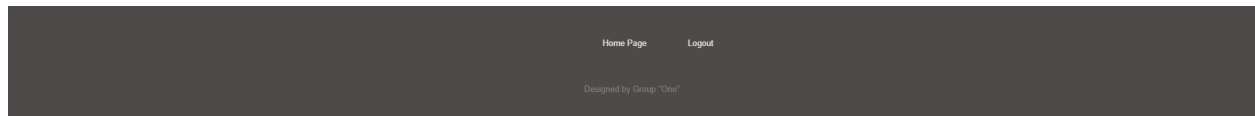
## On clicking **Directory Management**



## Non-admin user's home page



Some quick links are included in the footer in every page for easy navigation. The links navigate the user/admin back to the home page and facilitate logout respectively:



### 3.2 Responsibility of G2:

40Pts.

G2 has to work on integrating MySQL in WebApp. This group works on the proper working of the backend.

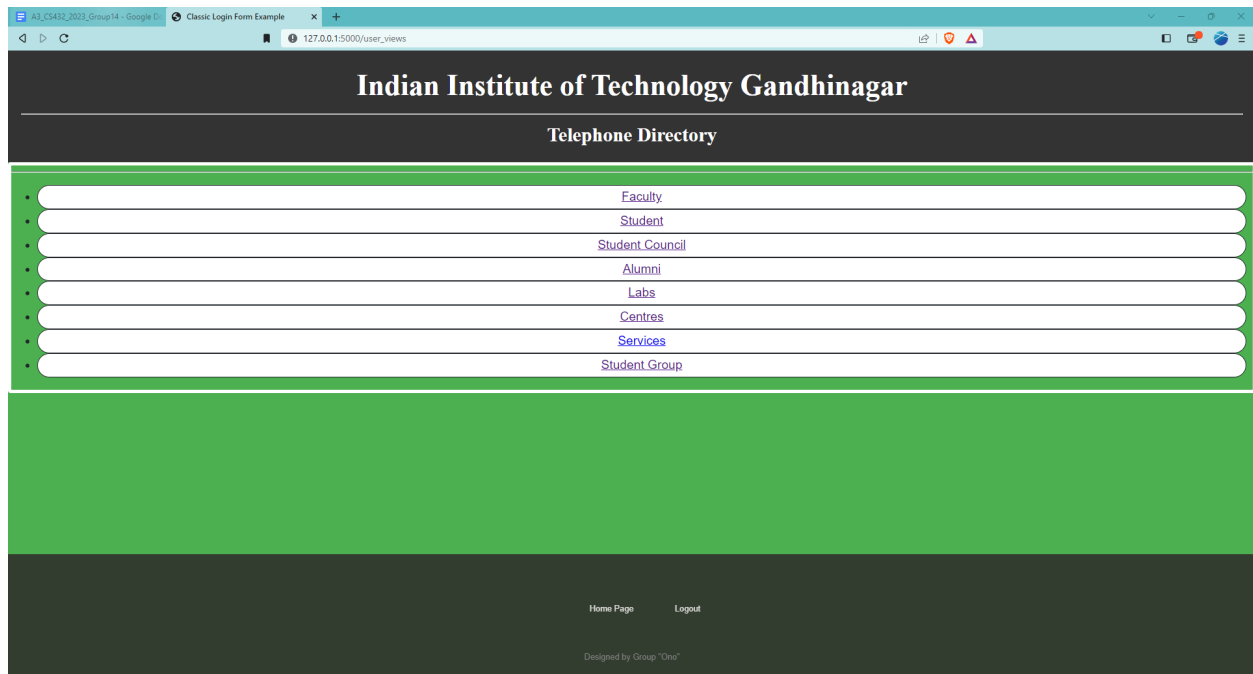
### 3.3 Responsibility of G1 & G2:

20Pts.

1. Create the database of the designed logical schema in your previous assignment number 2, enter dummy/imaginary values into it, and show these dummy entries on the web page. (10M)

The dummy data for the database was taken from CSV files and a sql dump file was generated. The database was integrated into the webpage using the flask app mysql connector. The further operations being pulled on the webpage are dynamically executed.

The user view of the website has the following options:



The tables are joined coherently to match the functionality defined in the previous assignments. The dummy data values can be seen in the tables:

A3\_CS432\_2022\_Group14 - Google D...

Faculty At IITGN Table

127.0.0.1:5000/faculty

# Indian Institute of Technology Gandhinagar

## Telephone Directory

Faculty At IITGN

faculty_id	first_name	last_name	dept_name	designation	block_no	room_no	office_phone_number	Contact Details
10861895	Lucy	Goodwin	Biological Engineering	Associate Professor	3	101	9206468795	Contact
11643101	Jarett	Hessel	Civil Engineering	Associate Professor	3	102	9885201804	Contact
12340912	Lauren	Blick	Chemical Engineering	Assistant Professor	3	103	9556788048	Contact
12340913	Nona	Wehner	Computer Science and Engineering	Department Head	3	104	7598279544	Contact
12340914	Kailyn	Kuhlman	Electrical Engineering	Department Head	3	303	7313061424	Contact
12340915	Brandt	Rohan	Mechanical Engineering	Assistant Professor	3	304	7613606677	Contact
12340916	Max	Lowe	Materials Engineering	Associate Professor	3	305	7249562277	Contact
12340918	Barry	Yost	Mathematics	Department Head	4	201	9866162808	Contact
12340919	Mikayla	Windler	Physics	Associate Professor	4	202	9063577650	Contact
12340920	Brandy	Nienow	Cognitive Science	Assistant Professor	4	203	8924582281	Contact
12340921	Vivien	Bashirian	Earth Sciences	Associate Professor	4	203	8924582281	Contact
12340922	Jaunita	Stracker	Humanities and Social Sciences	Department Head	5	301	8536023832	Contact

A3\_CS432\_2022\_Group14 - Google D...

Student Table

127.0.0.1:5000/student

# Indian Institute of Technology Gandhinagar

## Telephone Directory

Student

student_id	first_name	last_name	program_name	dept_name	Contact Details
19110001	Casey	Weissnat	BTech	Electrical Engineering	Contact
19110002	Yasmine	Graham	MTech	Computer Science and Engineering	Contact
19110003	Catalina	Lind	MSc	Cognitive Science	Contact
19110004	Felcher	Bogan	MASC	Humanities and Social Sciences	Contact
19110005	Camelo	Parisian	BTech	Materials Engineering	Contact
19110006	Kavon	Trump	MTech	Mechanical Engineering	Contact
19110007	Orpha	Jones	MSc	Mathematics	Contact
19110008	Walter	White	MSc	Physics	Contact
20110001	Jesse	Pinkman	BTech	Civil Engineering	Contact
20110002	Skyler	White	MTech	Chemical Engineering	Contact
20110003	Fabian	Hane	MSc	Chemistry	Contact
20110004	Cedrick	Swamiwaski	MASC	Humanities and Social Sciences	Contact
.....	-	-	-	-	Contact

Indian Institute of Technology Gandhinagar						
Telephone Directory						
Labs At IITGN						
lab_name	email_id	dept_name	block_no	room_no	office_phone_number	Member List
Experimental Complex Fluids Lab	ecfl@iitgn.ac.in	Chemical Engineering	6	320	7442323179	<a href="#">Members</a>
Design and Innovation Lab	designinnovationlab.pvtgroup@iitgn.ac.in	Civil Engineering	6	330	7792291105	<a href="#">Members</a>
Brain & Informatics	brainlab.pvtgroup@iitgn.ac.in	Cognitive Science	5	302	7635270492	<a href="#">Members</a>
Computational Nanoelectronic Lab	conrel@iitgn.ac.in	Computer Science and Engineering	6	315	9943835412	<a href="#">Members</a>
Computer Vision Imaging and Graphics Lab	cvig@iitgn.ac.in	Computer Science and Engineering	5	303	8951151328	<a href="#">Members</a>
Soil Testing Lab	touchgrass@iitgn.ac.in	Earth Sciences	3	204	7661836773	<a href="#">Members</a>
Analog CKT Lab	analog-ckt-lab.pvtgroup@iitgn.ac.in	Electrical Engineering	5	304	9117583346	<a href="#">Members</a>
Electrical Engineering Lab	celabs@iitgn.ac.in	Electrical Engineering	3	201	9232280679	<a href="#">Members</a>
Material Characterization Lab	matchlab@iitgn.ac.in	Materials Engineering	3	202	9556196434	<a href="#">Members</a>
<a href="#">Home Page</a> <a href="#">Logout</a>						

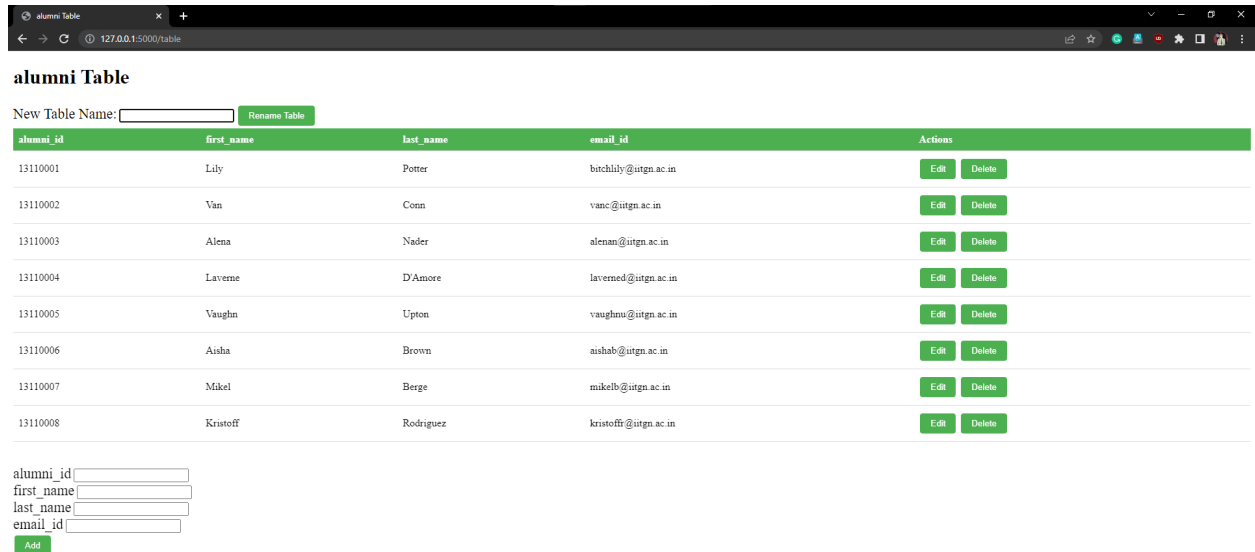
Indian Institute of Technology Gandhinagar								
Telephone Directory								
Centres At IITGN								
centre_name	email_id	block_no	room_no	office_phone_number	first_name	last_name	designation	Member List
Maker Bhavan	maker.bhavan@iitgn.ac.in	5	106	7726598799	Makayla	Windle	Making Coordinator	<a href="#">Members</a>
Nyasa	nyasa@iitgn.ac.in	5	107	8376777959	Brandy	Nienow	Child Lover	<a href="#">Members</a>
Neev	neev@iitgn.ac.in	5	108	8296921100	Vivien	Bashirian	Art and Craft Lover	<a href="#">Members</a>
CCL	ccl@iitgn.ac.in	6	230	7891685388	Jasmita	Stracker	Creative Learner	<a href="#">Members</a>
<a href="#">Home Page</a> <a href="#">Logout</a>								
Designed by Group "One"								

Similarly, all other pages reflect similar data. The rest of the functionality (along with SS of populated tables) is explained in the subsequent tasks.

## 2. Implementing the tasks as per the section [Section 2: Problem Statement and Requirements](#). (15M)

- INSERT

### Original table



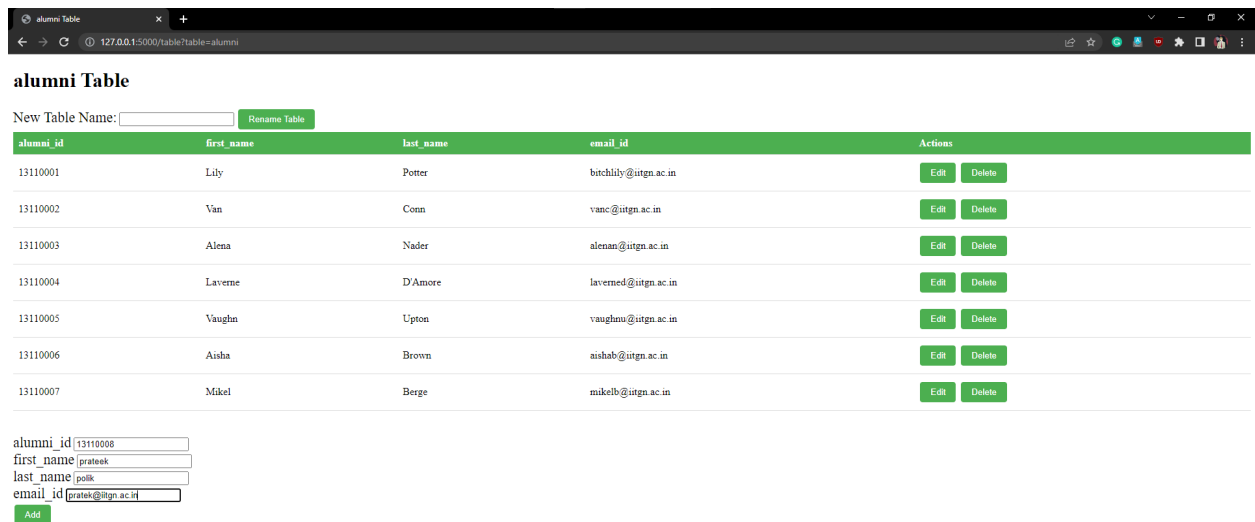
alumni Table

New Table Name:  Rename Table

alumni_id	first_name	last_name	email_id	Actions
13110001	Lily	Potter	bitchlily@utgn.ac.in	<span>Edit</span> <span>Delete</span>
13110002	Van	Conn	vanc@utgn.ac.in	<span>Edit</span> <span>Delete</span>
13110003	Alena	Nader	alenan@utgn.ac.in	<span>Edit</span> <span>Delete</span>
13110004	Laverne	D'Amore	laverned@utgn.ac.in	<span>Edit</span> <span>Delete</span>
13110005	Vaughn	Upton	vaughnu@utgn.ac.in	<span>Edit</span> <span>Delete</span>
13110006	Aisha	Brown	aishab@utgn.ac.in	<span>Edit</span> <span>Delete</span>
13110007	Mikel	Berge	mikelb@utgn.ac.in	<span>Edit</span> <span>Delete</span>
13110008	Kristoff	Rodriguez	kristoffr@utgn.ac.in	<span>Edit</span> <span>Delete</span>

alumni\_id   
first\_name   
last\_name   
email\_id   
Add

### Inserting 'Prateek Polik' with alumni id 13110008 into the alumni table



alumni Table

New Table Name:  Rename Table

alumni_id	first_name	last_name	email_id	Actions
13110001	Lily	Potter	bitchlily@utgn.ac.in	<span>Edit</span> <span>Delete</span>
13110002	Van	Conn	vanc@utgn.ac.in	<span>Edit</span> <span>Delete</span>
13110003	Alena	Nader	alenan@utgn.ac.in	<span>Edit</span> <span>Delete</span>
13110004	Laverne	D'Amore	laverned@utgn.ac.in	<span>Edit</span> <span>Delete</span>
13110005	Vaughn	Upton	vaughnu@utgn.ac.in	<span>Edit</span> <span>Delete</span>
13110006	Aisha	Brown	aishab@utgn.ac.in	<span>Edit</span> <span>Delete</span>
13110007	Mikel	Berge	mikelb@utgn.ac.in	<span>Edit</span> <span>Delete</span>

alumni\_id   
first\_name   
last\_name   
email\_id   
Add



## Inserted Entry

alumni Table

127.0.0.1:5000/table?table=alumni

alumni Table

New Table Name:

Rename Table

alumni_id	first_name	last_name	email_id	Actions
13110001	Lily	Potter	bitchlily@iitgn.ac.in	<div>EditDelete</div>
13110002	Van	Conn	vanc@iitgn.ac.in	<div>EditDelete</div>
13110003	Alena	Nader	alenan@iitgn.ac.in	<div>EditDelete</div>
13110004	Laverne	D'Amore	laverned@iitgn.ac.in	<div>EditDelete</div>
13110005	Vaughn	Upton	vaughnu@iitgn.ac.in	<div>EditDelete</div>
13110006	Aisha	Brown	aishab@iitgn.ac.in	<div>EditDelete</div>
13110007	Mikel	Berge	mikelb@iitgn.ac.in	<div>EditDelete</div>
13110008	prateek	polik	pratek@iitgn.ac.in	<div>EditDelete</div>

alumni\_id

first\_name

last\_name

email\_id

Add

- UPDATE

Edit button opens up the shown form

Edit Entry

127.0.0.1:5000/edit\_entry

Edit Entry

alumni\_id

13110001

first\_name

Lily

last\_name

Potter

email\_id

bitchlily@iitgn.ac.in

Update

Editing the entry

Edit Entry

127.0.0.1:5000/edit\_entry

Edit Entry

alumni\_id

13110001

first\_name

Harry

last\_name

Potter

email\_id

harrypp@iitgn.ac.in

Update

Updated table with “Harry” instead of “Lily”

alumni Table

127.0.0.1:5000/table?table=alumni

New Table Name:

Rename Table

alumni_id	first_name	last_name	email_id	Actions
13110001	Harry	Potter	harrypp@utgn.ac.in	<div>EditDelete</div>
13110002	Van	Conn	vanc@utgn.ac.in	<div>EditDelete</div>
13110003	Alena	Nader	alenan@utgn.ac.in	<div>EditDelete</div>
13110004	Laverne	D'Amore	laverned@utgn.ac.in	<div>EditDelete</div>
13110005	Vaughn	Upton	vaughnu@utgn.ac.in	<div>EditDelete</div>
13110006	Aisha	Brown	aishab@utgn.ac.in	<div>EditDelete</div>
13110007	Mikel	Berge	mikelb@utgn.ac.in	<div>EditDelete</div>
13110008	prateek	polik	pratek@utgn.ac.in	<div>EditDelete</div>

alumni\_id

first\_name

last\_name

email\_id

Add

- DELETE

Entry is deleted in the database table screenshot

alumni Table

127.0.0.1:5000/table?table=alumni

New Table Name:

Rename Table

alumni_id	first_name	last_name	email_id	Actions
13110001	Lily	Potter	bitchlily@utgn.ac.in	<div>EditDelete</div>
13110002	Van	Conn	vanc@utgn.ac.in	<div>EditDelete</div>
13110003	Alena	Nader	alenan@utgn.ac.in	<div>EditDelete</div>
13110004	Laverne	D'Amore	laverned@utgn.ac.in	<div>EditDelete</div>
13110005	Vaughn	Upton	vaughnu@utgn.ac.in	<div>EditDelete</div>
13110006	Aisha	Brown	aishab@utgn.ac.in	<div>EditDelete</div>
13110007	Mikel	Berge	mikelb@utgn.ac.in	<div>EditDelete</div>

alumni\_id

first\_name

last\_name

email\_id

Add

- RENAME

## Changing original table name

alumni Table

127.0.0.1:5000/table?table=alumni

New Table Name:

Rename Table

alumni_id	first_name	last_name	email_id	Actions
13110001	Harry	Potter	harrypp@uitgn.ac.in	<div>EditDelete</div>
13110002	Van	Conn	vanc@uitgn.ac.in	<div>EditDelete</div>
13110003	Alena	Nader	alenan@uitgn.ac.in	<div>EditDelete</div>

## Renamed the table

dumbname Table

127.0.0.1:5000/table

New Table Name:

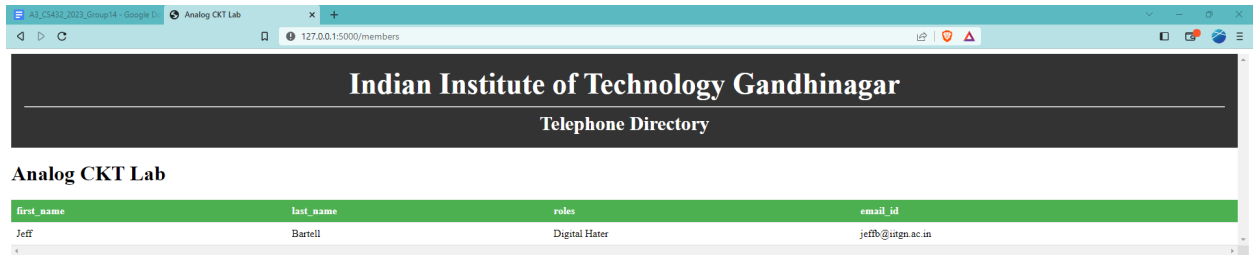
Rename Table

alumni_id	first_name	last_name	email_id	Actions
13110001	Harry	Potter	harrypp@uitgn.ac.in	<div>EditDelete</div>
13110002	Van	Conn	vanc@uitgn.ac.in	<div>EditDelete</div>
13110003	Alena	Nader	alenan@uitgn.ac.in	<div>EditDelete</div>
13110004	Laverne	D'Amore	laverned@uitgn.ac.in	<div>EditDelete</div>
13110005	Vaughn	Upton	vaughnu@uitgn.ac.in	<div>EditDelete</div>

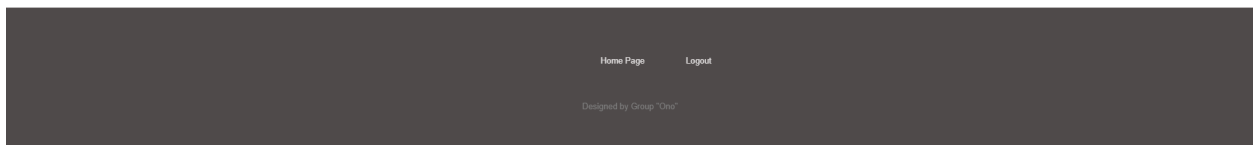
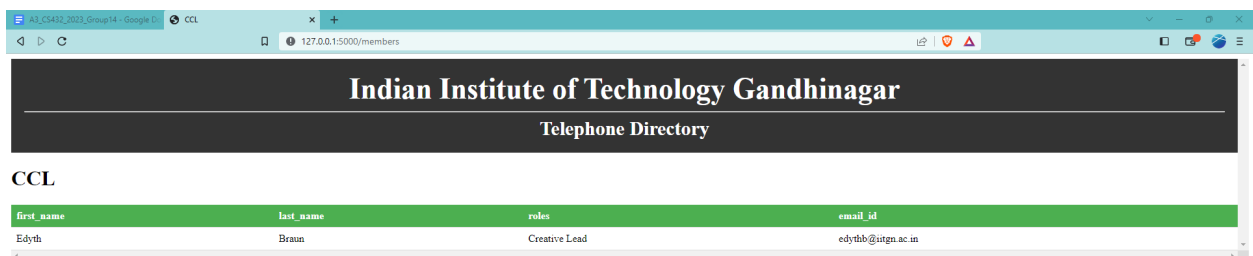
- WHERE clause

The WHERE clause is used as a query in the backend to display:

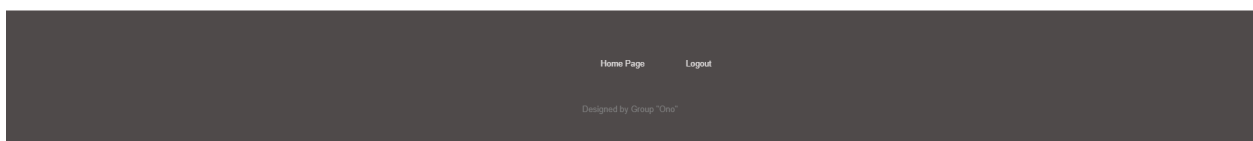
1. list of students who are in Student Council, Student Groups and list of staff working in Services, Centres and Labs. These can be accessed by clicking the 'Members' button in the respective pages.



first_name	last_name	roles	email_id
Jeff	Bartell	Digital Hater	jeffb@iitgn.ac.in

first_name	last_name	roles	email_id
Edyth	Braun	Creative Lead	edythb@iitgn.ac.in



The code in the backend with WHERE clause is:

```

query1 = f"select student.first_name, student.last_name, student_council_member.position, student_enrolled.program_name, student_enrolled.dept_name, student.email_id " \
f"from student join student_council_member on student.student_id = student_council_member.student_id " \
f"join student_enrolled " \
f"on student.student_id = student_enrolled.student_id " \
f"where student_council_member.council_name = '{value}'"

query2 = f"select student.first_name, student.last_name, student_group_member.position, student_enrolled.program_name, student_enrolled.dept_name, student.email_id " \
f"from student " \
f"join student_group_member " \
f"on student.student_id = student_group_member.student_id " \
f"join student_enrolled " \
f"on student.student_id = student_enrolled.student_id " \
f"where student_group_member.group_name = '{value}'"

query3 = f"select staff.first_name, staff.last_name, service_staff.roles from staff, staff.email_id " \
f"join service_staff " \
f"on staff.staff_id = service_staff.staff_id " \
f"where service_staff.service_name = '{value}'"

query4 = f"select staff.first_name, staff.last_name, centre_staff.roles, staff.email_id " \
f"from staff " \
f"join centre_staff " \
f"on staff.staff_id = centre_staff.staff_id " \
f"where centre_staff.centre_name = '{value}'"

query5 = f"select staff.first_name, staff.last_name, lab_staff.roles, staff.email_id " \
f"from staff " \
f"join lab_staff " \
f"on staff.staff_id = lab_staff.staff_id " \
f"where lab_staff.lab_name = '{value}'"

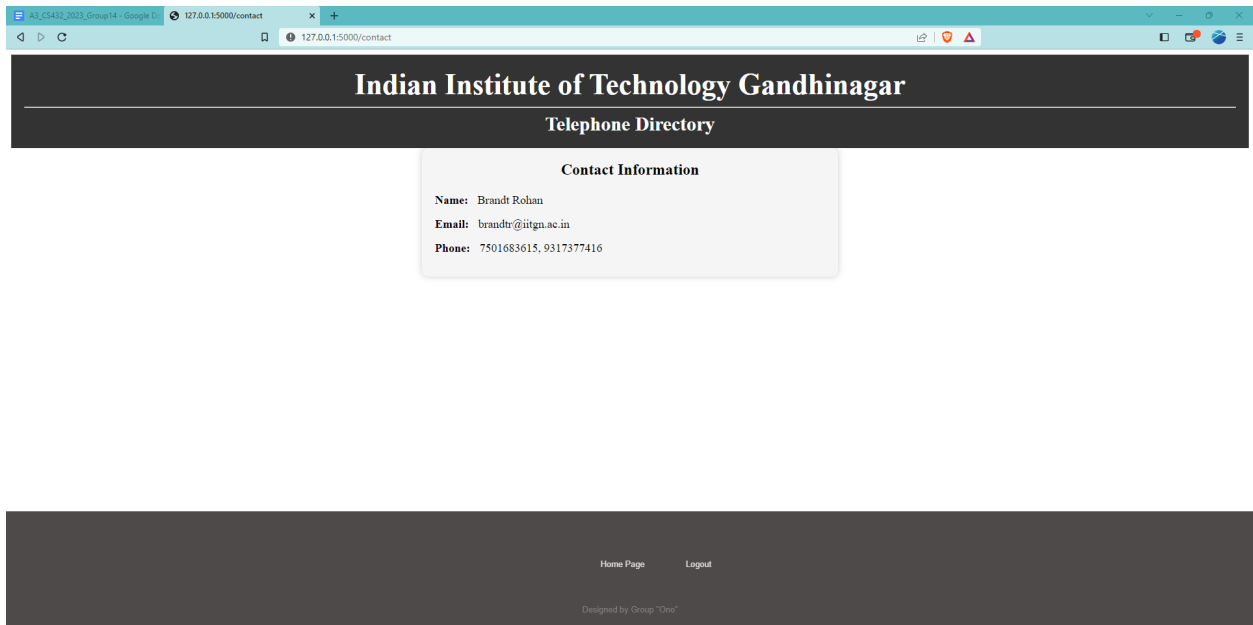
```

- the contact details of students, faculty and alumni members when clicked on the 'Contact' button in their respective pages.

The screenshot shows a web browser window with the address bar displaying "127.0.0.1:5000/contact". The page content includes the header "Indian Institute of Technology Gandhinagar" and "Telephone Directory". Below this, a "Contact Information" box displays the following details:

- Name:** Casey Weissnat
- Email:** casey.weissnat@iitgn.ac.in
- Phone:** 8159769661, 8165028752, 8804152916, 9398906745

At the bottom of the page, there are links for "Home Page" and "Logout", and a footer that reads "Designed by Group 'One'".



The code in the backend with WHERE clause is:

```
@app.route('/contact', methods=['POST'])
def contact():
    table = request.form['table']
    primary_key = request.form['primary_key']
    value = request.form['value']

    print("CONTACTS:", "1", table, "2", primary_key, "3", value)
    cursor = db.cursor()

    query1 = f"select student.first_name, student.last_name, student.email_id, group_concat(student_phone_number.phone_number separator ',') as phone_numbers " \
    f"from student " \
    f"join student_phone_number " \
    f"on student.student_id = student_phone_number.student_id " \
    f"where student.student_id = '{value}'"

    query2 = f"select alumni.first_name, alumni.last_name, alumni.email_id, group_concat(alumni_phone_number.phone_number separator ',') as phone_numbers " \
    f"from alumni " \
    f"join alumni_phone_number " \
    f"on alumni.alumni_id = alumni_phone_number.alumni_id " \
    f"where alumni.alumni_id = '{value}'"

    query3 = f"select faculty.first_name, faculty.last_name, faculty.email_id, group_concat(faculty_phone_number.phone_number separator ',') as phone_numbers " \
    f"from faculty " \
    f"join faculty_phone_number " \
    f"on faculty.faculty_id = faculty_phone_number.faculty_id " \
    f"where faculty.faculty_id = '{value}'"

    queries = {
        'Student': query1,
        'Alumni': query2,
        'Faculty At IITGN': query3
    }
}
```

GitHub link : [https://github.com/TwoGunTitan/Group\\_14\\_Assignment\\_3](https://github.com/TwoGunTitan/Group_14_Assignment_3)

3. Write-up in the README file, which should include the following: (5M)
  - a. Steps to run your web app [initial setup, executing the required tasks, etc.].
  - b. Screenshots of successful execution of the dynamic operations defined in [Section 2: Problem Statement and Requirements](#). You can design the web app in such a way that a dialogue box pops up that the query has been successfully executed. Also, you will have to post these snapshots of the dialogue box (in the readme file) for every respective query. Or else, you can directly show the before and after changes reflected in your concerned table in the web app in the README file.

Step 1: Create a virtual python environment and proceed to download the required packages and dependencies as mentioned in the requirements document. Clone the GitHub repository at this created virtual environment through the above mentioned link.

Step 2: Perform 'flask run' app.py script to find the webpage link where you can interact with the telephone directory database based on your authentication.

The original database is active now. If you have admin privileges, one can perform the view, insert, update, delete, rename, and where operations through the webpage altering the backend database. If you do not have these admin privileges, one can only view while performing where operation.

On the login page, logging in with the username and password as 'admin', one gets admin privileges while logging in using other credentials, one gets user privileges. You need to register first before logging in if you are a user. We are navigated to the home page. where if you are a user, clicking on 'Phone Directory' will lead to the database view page. If you are admin, clicking on 'Directory Management' will lead you to the page where you can perform dynamic updates on the database.

The executions of the operations in the webpage can be performed as shown through the screenshots above.

## Contributions

---

Name	Group	Contributions
Aravind Krishna - 20110021	G1	8.33%
Arun Mani - 20110023	G1	8.33%
Hrishikesh CP -20110078	G1	8.33%
Mihika Jadhav - 20110113	G1	8.33%
Madhav Biswam -20110103	G1	8.33%
Rahul R Pai - 20120155	G1	8.33%
Shrija Agrawal - 20110197	G1	8.33%
Abhiram Geddam - 19110149	G2	8.33%
Kanishk Singh - 19110089	G2	8.33%
Ruthvik Mula - 19110124	G2	8.33%
Ninad Shah - 19110126	G2	8.33%
Prateek Kumar Jha - 19110159	G2	8.33%