

BACK-END CODE

```
from flask import Flask,render_template,request,session,url_for,redirect,flash
from flask_mysqlldb import MySQL
import mysql.connector
import smtplib
from email.message import EmailMessage
import razorpay
```

```
myapp = Flask(__name__)
```

```
myapp.config['SECRET_KEY'] = '_5#y2L"F4Q8z\n\xec]/'
myapp.config['MYSQL_HOST'] = 'localhost'
myapp.config['MYSQL_USER'] = 'root'
myapp.config['MYSQL_PASSWORD'] = 'Betaop$17'
myapp.config['MYSQL_DB'] = 'connectsport'
```

```
mysql = MySQL(myapp)
```

```
razorpay_key_id = 'rzp_test_dYStpQwu6bs3hh'
razorpay_key_secret = 'mTN9AofGfNT330nrs19Pm0RZ'
razorpay_client = razorpay.Client(auth=(razorpay_key_id,
razorpay_key_secret))
```

```
@myapp.route('/')
def index():
```

```
    return render_template('login.html')
```

```
@myapp.route('/home')
```

```

def home():
    if 'loggedin' in session:
        return render_template('home.html', username=session['username'])
    return redirect(url_for('index'))

@app.route('/login',methods=['GET','POST'])
def login():
    msg=""
    if request.method=='POST':
        username = request.form['username']
        password = request.form['passwd']
        cursor = mysql.connection.cursor()
        cursor.execute('SELECT * FROM login WHERE email=%s AND
password=%s',(username,password))
        record = cursor.fetchone()
        if record:
            session['loggedin']=True
            session['username']=record[0]
            return redirect(url_for('home'))
        else:
            msg='Incorrect username/password, Try again'
            flash(msg, 'error')
    return render_template('login.html,msg=msg)

@app.route('/logout')
def logout():
    session.pop('loggedin', None)
    session.pop('username', None)

```

```

return redirect(url_for('index'))

@myapp.route('/registration', methods=['GET', 'POST'])
def registration():
    msg = "
    if request.method == 'POST':
        username = request.form['userid']
        password = request.form['passwd']
        email = request.form['email']
        uname = request.form['uname']
        phone_no = request.form['phnum']
        roles = request.form['role']
        cursor = mysql.connection.cursor()
        cursor.execute('SELECT * FROM login WHERE userid=%s', (username,))
        record = cursor.fetchone()

        if record:
            flash('Username already exists, please choose a different one.')
            return redirect(url_for('registration'))

        cursor.execute('INSERT INTO login (userid, password, email, uname,
phone_no,role) VALUES (%s, %s, %s, %s, %s,%s)',
                        (username, password, email, uname, phone_no,roles,))
        mysql.connection.commit()
        cursor.close()

        flash('Registration successful! You can now log in.')
        return redirect(url_for('index'))

```

```

return render_template('reg.html')

def registration_mail(us_id):
    smtp_server = 'smtp.gmail.com'
    smtp_port = 587
    smtp_username = 'superstarsuperstar123123@gmail.com'
    smtp_password = 'vfm ugim qljz jedd'

    cursor = mysql.connection.cursor()
    cursor.execute('SELECT uname, email FROM login WHERE userid=%s',
(us_id,))
    player = cursor.fetchone()

    if player:
        msg = EmailMessage()
        msg.set_content(f"Hello {player[0]},\n\n"
            f"Welcome To Connectport Buddy! "
            f"\n Hope you had a great time"
            f"\nBest Regards"
            f"\nCONNECTSPORT TEAM")

        msg['Subject'] = 'Tournament Participation Details'
        msg['From'] = smtp_username
        msg['To'] = player[1]
        print(player[1]
            ])

```

```

with smtplib.SMTP(smtp_server, smtp_port) as server:
    server.starttls()
    server.login(smtp_username, smtp_password)
    server.send_message(msg)

cursor.close()
return 'successful'
else:
    return 'Player not found'

@app.route('/profiles')
def profiles():
    if 'loggedin' in session:
        username = session['username']
        cursor = mysql.connection.cursor()
        cursor.execute('SELECT userid, email, uname, role, phone_no FROM
login WHERE userid = %s', (username,))
        user_details = cursor.fetchone()
        cursor.close()
        if user_details:
            role = user_details[3]
            if role == 'tcc':
                return render_template('cprofile.html', user_details=user_details)
            elif role == 'play':
                return render_template('plyprofile.html', user_details=user_details)
            else:
                return "Invalid user role."
        else:

```

```
return "User details not found."
```

```
@myapp.route('/create_tournament', methods=['POST'])
```

```
def create_tournament():
```

```
    if request.method == 'POST':
```

```
        cursor = mysql.connection.cursor()
```

```
        orgid = session['username']
```

```
        tournament_name = request.form['tname']
```

```
        category = request.form['category']
```

```
        game_name = request.form['gameName']
```

```
        age_limit = request.form['agelimit']
```

```
        reg_end_date = request.form['rd']
```

```
        match_date = request.form['td']
```

```
        location = request.form['location']
```

```
        entry_fee = request.form['efee']
```

```
        gender = request.form['gender']
```

```
        cursor.execute('INSERT INTO tournament(org_id,tname, category,  
gameName, agelimit, registration_ed, match_date, location, entry_fee, gender)  
VALUES (%s,%s, %s, %s, %s, %s, %s, %s, %s, %s)',
```

```
        (orgid,tournament_name, category, game_name, age_limit,  
reg_end_date, match_date, location, entry_fee, gender,))
```

```
        mysql.connection.commit()
```

```
        cursor.close()
```

```
        flash('Tournament created successfully!', 'success')
```

```
        return redirect(url_for('profiles'))
```

```
@myapp.route('/view_conducted')
```

```

def view_conducted():
    conduct_id = session['username']
    cursor = mysql.connection.cursor()
    cursor.execute('SELECT * FROM tournament WHERE org_id = %s',
(conduct_id,))

    conducted_tournaments = cursor.fetchall()
    formatted_tournaments = []
    for tournament in conducted_tournaments:
        tournament_id = tournament[0]
        cursor.execute("""
            SELECT COUNT(*) AS participant_count
            FROM playertournamentinfo ti
            WHERE ti.tournament_id = %s
        """, (tournament_id,))
        participant_count = cursor.fetchone()[0]

        formatted_tournament = {
            'tournament_details': tournament,
            'participant_count': participant_count
        }
        formatted_tournaments.append(formatted_tournament)

    cursor.close()

    return render_template('concards.html',
conducted_tournaments=formatted_tournaments)

@app.route('/about')
def about():

```

```

return render_template('about.html')

@app.route('/tournament_card' )
def tournament_card():
    cursor = mysql.connection.cursor()
    cursor.execute('SELECT * FROM tournament')
    tournaments = cursor.fetchall()
    cursor.close()
    return render_template('card.html', tournaments=tournaments)

def send_email(name, email, message):
    smtp_server = 'smtp.gmail.com'
    smtp_port = 587
    smtp_username = 'superstarsuperstar123123@gmail.com'
    smtp_password = 'vfim ugim qljz jedd'

    msg = EmailMessage()
    msg.set_content(f'From: {name}\nEmail: {email}\n\n{message}')
    msg['Subject'] = 'Contact Form Submission'
    msg['From'] = email
    msg['To'] = 'connectsports18@gmail.com'

    with smtplib.SMTP(smtp_server, smtp_port) as server:
        server.starttls()
        server.login(smtp_username, smtp_password)
        server.send_message(msg)

```



```
@myapp.route('/contact',methods=['GET','POST'])
```

```
def contact():
```

```
    if request.method == 'POST':
```

```
        name = request.form['name']
```

```
        email = request.form['email']
```

```
        message = request.form['message']
```

```
        send_email(name, email, message)
```

```
        return 'Thank you for your message!'
```

```
    return render_template('contact.html')
```

```
@myapp.route('/message', methods=['GET', 'POST'])
```

```
def message():
```

```
    if request.method == 'POST':
```

```
        try:
```

```
            cursor = mysql.connection.cursor()
```

```
            uid = session['username']
```

```
            msgs = request.form['content']
```

```
            print(uid)
```

```
            print(msgs)
```

```
            cursor.execute('INSERT INTO connectsport.posts(user_id, content)
VALUES (%s, %s)', (uid, msgs,))
```

```
            mysql.connection.commit()
```

```
            cursor.close()
```

```
            flash('Message Posted!', 'post_success')
```

```
            return redirect('/message')
```

```
        except Exception as e:
```

```
            flash('An error occurred while posting the message', 'merror')
```

```
            print('Error:', e)
```

```

    return render_template('message.html')

@app.route('/postmsg')
def postmsg():
    cursor=mysql.connection.cursor()
    cursor.execute('SELECT * FROM posts')
    mesgs = cursor.fetchall()
    cursor.close()
    return render_template('postmsg.html',mesgs=mesgs)

def user_check(user_id):
    cursor = mysql.connection.cursor()
    cursor.execute('SELECT role FROM login WHERE userid=%s', (user_id,))
    user_role = cursor.fetchone()
    cursor.close()
    if user_role and user_role[0] == 'tcc':
        return True
    else:
        return False

@app.route('/playerinfo', methods=['POST'])
def playerinfo():
    if request.method == 'POST':
        u_id = session['username']
        t_id = request.form['tournamentId']
        teamname = request.form['teamName']
        user = request.form['yourName']

```

```

    loca = request.form['address']
    cursor = mysql.connection.cursor()
    cursor.execute('SELECT * FROM login WHERE userid=%s', (u_id,))
    user_role = cursor.fetchone()
    role = user_role[5]
    print(role)
    if role == 'tcc':
        flash('Tournament conductor cannot participate in the
        tournaments','error')
        return redirect(url_for('home'))
    elif role == 'play':
        cursor.execute('INSERT INTO playertournamentinfo (userid,
        tournament_id, teamname, name, location) VALUES (%s, %s, %s, %s, %s)',
        (u_id, t_id, teamname, user, loca))
        mysql.connection.commit()
        cursor.close()

        send_participation_mail(session['username'])
    else:
        return 'Invalid User Role'

    return 'Player information inserted successfully'
else:
    return 'Method not allowed'

@app.route('/payment', methods=['GET', 'POST'])
def payment():
    if request.method == 'POST':

```

```

payment_id = request.form['razorpay_payment_id']
payment_order_id = session.get('payment_order_id')

if payment_order_id:
    try:
        payment = razorpay_client.payment.fetch(payment_id)
        if payment['status'] == 'captured':
            u_id = session['username']
            t_id = request.form['tournamentId']
            teamname = request.form['teamName']
            user = request.form['yourName']
            loca = request.form['address']

            cursor = mysql.connection.cursor()
            cursor.execute('INSERT INTO playertournamentinfo (userid,
tournament_id, teamname, name, location) VALUES (%s, %s, %s, %s, %s)',
(u_id, t_id, teamname, user, loca))
            mysql.connection.commit()
            cursor.close()

            # Send participation mail
            send_participation_mail(session['username'])

            flash('Payment successful. Player information inserted
successfully.', 'success')
            return redirect(url_for('home'))
        else:
            flash('Payment verification failed. Please contact support.', 'error')

```

```

        return redirect(url_for('home'))
    except Exception as e:
        print('Error in payment verification:', e)
        flash('Payment verification failed. Please contact support.', 'error')
        return redirect(url_for('home'))
    else:
        flash('Payment order not found. Please try again.', 'error')
        return redirect(url_for('home'))
    else:
        # Render payment.html template for displaying Razorpay checkout form
        return render_template('payment.html')

@app.route('/payment_failure')
def payment_failure():
    flash('Payment failed. Please try again.', 'error')
    return redirect(url_for('home'))

@app.route('/ply_tournaments')
def ply_tournaments():
    ply_id = session['username']
    cursor = mysql.connection.cursor()

    cursor.execute('SELECT ti.userid, ti.tournament_id, ti.name AS
player_name, ti.teamname, ti.location AS player_location, '
                  't.org_id, t.tname AS tournament_name, t.gameName AS
game_name, t.match_date, '
                  't.location AS tournament_location, t.entry_fee AS efee '
                  'FROM playertournamentinfo ti '
                  'JOIN tournament t ON ti.tournament_id = t.tournament_id ')

```

```

        'WHERE ti.userid = %s', (ply_id,))

player_tournament_details = cursor.fetchall()

cursor.close()

return render_template('ply_cards.html',
player_tournament_details=player_tournament_details)

def send_participation_mail(player_id):

    smtp_server = 'smtp.gmail.com'

    smtp_port = 587

    smtp_username = 'superstarsuperstar123123@gmail.com'

    smtp_password = 'vfm ugim qljz jedd'

    cursor = mysql.connection.cursor()

    cursor.execute('SELECT uname, email FROM login WHERE userid=%s',
(player_id,))

    player = cursor.fetchone()

    if player:

        msg = EmailMessage()

        msg.set_content(f"Hello {player[0]},\n\n"

            f"Thank you for participating in the tournament.Use this mail in
Matchday"

            f"\n Hope you had a great time"

            f"\nBest Regards"

            f"\nCONNECTSPORT TEAM")

        msg['Subject'] = 'Tournament Participation Details'

        msg['From'] = smtp_username

```

```

msg['To'] = player[1]
print(player[1])

with smtplib.SMTP(smtp_server, smtp_port) as server:
    server.starttls()
    server.login(smtp_username, smtp_password)
    server.send_message(msg)

cursor.close()
return 'successful'
else:
    return 'Player not found'

if __name__ == "__main__":
    myapp.run(debug=True)
{% for chunk in tournaments|batch(4) %}
<div class="card-container">
    {% for tournament in chunk %}
    <div class="card">
        <h1 class="tournament-name">NAME :{{ tournament[2] }}</h1>
        <h1 class="tournament-name">GAME :{{ tournament[4] }}</h1>
        <h1 class="tournament-date">REG-ENDDATE :{{ tournament[6]
    }}</h1>
        <h1 class="tournament-date">MATCH-DATE :{{ tournament[7]
    }}</h1>
        <h1 class="tournament-date">ENTRY-FEE :{{ tournament[9] }}</h1>
        <button type="submit" onclick="url_for('playerinfo')" class="register-
btn"

```

```

        data-tournament-id="{ { tournament[0] } }" data-entry-fee="{ {
tournament[9] } }">Register Now</button>

</div>

{ % endfor % }

</div>

{ % endfor % }

<script src="https://checkout.razorpay.com/v1/checkout.js"></script>
<script>

    document.addEventListener("DOMContentLoaded", function () {

        var popupContainer = document.getElementById('popupContainer');
        var registerBtns = document.getElementsByClassName("register-btn");

        console.log("Register buttons found:", registerBtns.length);

        for (var i = 0; i < registerBtns.length; i++) {
            registerBtns[i].addEventListener('click', function () {
                console.log("Register button clicked");
                var tournamentId = this.getAttribute("data-tournament-id");
                var entryFee = this.getAttribute("data-entry-fee");
                popupContainer.style.display = "flex";
                document.getElementById('tournamentId').value = tournamentId;
                document.getElementById('entryFee').value = entryFee;
            });
        }
    });

    window.onclick = function (event) {
        if (event.target == popupContainer) {

```



```

        popupContainer.style.display = "none";
    }
}

document.getElementById('payButton').onclick = function () {
    var teamName = document.getElementById('teamName').value;
    var yourName = document.getElementById('yourName').value;
    var address = document.getElementById('address').value;
    var tournamentId = document.getElementById('tournamentId').value;
    var entryFee = document.getElementById('entryFee').value;

    var options = {
        "key": "rzp_test_dYStpQwu6bs3hh",
        "amount": entryFee * 100,
        "currency": "INR",
        "name": "Your Company Name",
        "description": "Tournament Registration Fee",
        "handler": function (response) {
            alert("Payment successful: " + response.razorpay_payment_id);
            document.getElementById('registrationForm').submit();
        },
        "prefill": {
            "name": yourName,
            "email": "connectsport18@gmail.com",
            "contact": "9500797305"
        }
    };
};

```

```

        var rzp = new Razorpay(options);
        rzp.open();
    };
</script>
</body>

{% for formatted_tournament in conducted_tournaments %}
    {% if loop.index % 4 == 1 %}
        <div class="row">
            {% endif %}

            <div class="card">
                <h2>Tournament Name: {{
formatted_tournament['tournament_details'][1] }}</h2>
                <p>Game: {{ formatted_tournament['tournament_details'][4] }}</p>
                <p>Registration End Date: {{
formatted_tournament['tournament_details'][6] }}</p>
                <p>Match Date: {{ formatted_tournament['tournament_details'][7]
}}</p>
                <p>Entry Fee: {{ formatted_tournament['tournament_details'][9] }}</p>
                <p>Participant Count: {{ formatted_tournament['participant_count']
}}</p>
            </div>

            {% if loop.index % 4 == 0 or loop.last %}
                </div>
            {% endif %}
        {% endfor %}

```

```

div class="profile-info">
    <h2>Profile Information</h2>
    <p>userid: {{ user_details[0] }}</p>
    <p>role: {{ user_details[3] }}</p>
    <p>name:{{ user_details[2] }}</p>
    <p>Email: {{ user_details[1] }}</p>
    <p>Phone: {{ user_details[4] }}</p>
</div>
</div>
<h2>COMMUNICATION CENTER</h2>
<div class="message-container">
    {% for ms in mesgs %}
    <div class="message">
        <span class="username">Username :{{ ms[0] }}</span>
        <span class="content">Message :{{ ms[1] }}</span>
    </div>
    {% endfor %}
document.addEventListener("DOMContentLoaded", function () {
    var flashMessages = document.querySelectorAll(".flash-message");

    flashMessages.forEach(function (message) {
        setTimeout(function () {
            message.style.display = "none";
        }, 5000);
    });
});

```

