

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width,
initial-scale=1.0">
  <title>Sribhumi eSports Registration</title>
  <!-- Load Tailwind CSS -->
  <script src="https://cdn.tailwindcss.com"></script>
  <!-- Use Inter font -->
  <link
href="https://fonts.googleapis.com/css2?family=Inter:wght@400;600;800&
display=swap" rel="stylesheet">
  <style>
    body {
      font-family: 'Inter', sans-serif;
      background-color: #121212; /* Dark background */
    }
    /* Custom neon border effect for buttons and input focus */
    .neon-border {
      border: 2px solid transparent;
      transition: all 0.2s ease-in-out;
    }
    .neon-button {
      background-color: #059669; /* Green/accent color for
Sribhumi */
      color: #ffffff;
      font-weight: 700;
      transition: background-color 0.2s, box-shadow 0.2s;
    }
    .neon-button:hover {
      background-color: #047857;
      box-shadow: 0 0 10px #059669, 0 0 20px #047857;
    }
    .input-field:focus {
      border-color: #059669;
      box-shadow: 0 0 5px rgba(5, 150, 105, 0.5);
      outline: none;
    }
    .scroll-container::-webkit-scrollbar {
      width: 8px;
    }
    .scroll-container::-webkit-scrollbar-thumb {
      background-color: #059669;
      border-radius: 4px;
    }
    .scroll-container::-webkit-scrollbar-track {
      background: #2D3748;
    }
  </style>
</head>
<body>
```

```

    }
    .modal-bg {
        background-color: rgba(0, 0, 0, 0.8);
    }
    .payment-box {
        background-color: #1f2937;
        border: 2px solid #059669;
    }
</style>
</head>
<body class="min-h-screen flex items-center justify-center p-4">

    <!-- Tournament Registration Card -->
    <div class="w-full max-w-lg bg-gray-800 p-6 sm:p-8 rounded-xl
shadow-2xl border border-green-700/50">

        <h1 class="text-3xl font-extrabold text-white text-center mb-1
tracking-wider uppercase">
            Sribhumi eSports Championship
        </h1>
        <p class="text-xl font-light text-green-400 text-center mb-2">
            Prize Pool: ₹1,600 | Entry: ₹200 per Squad
        </p>

        <!-- Dynamic Status Message -->
        <p id="form-status" class="text-center text-sm mb-4
text-yellow-400">
            Initializing connection...
        </p>

        <!-- Payment Instructions -->
        <div class="payment-box p-4 rounded-lg mb-6 text-center">
            <h2 class="text-lg font-bold text-green-400 mb-2">PAYMENT
IS MANDATORY TO REGISTER</h2>
            <p class="text-gray-200 text-sm mb-2">
                Pay **₹200** per squad using the UPI ID below:
            </p>
            <p class="text-2xl font-extrabold text-white mb-3
tracking-wider bg-gray-900 py-2 rounded">
                8011866993@ptsbi
            </p>
            <p class="text-yellow-300 text-xs font-semibold">
                **ACTION REQUIRED:** After payment, send the
screenshot to WhatsApp number:
                <a href="https://wa.me/918011866993" target="_blank"
class="text-green-300 hover:text-green-500 underline
font-extrabold">8011866993 (Click to Chat)</a>
            </p>

```

```

</div>

<form id="registration-form" class="space-y-4">
  <!-- Team Information Section -->
  <div class="space-y-4 p-4 bg-gray-700/50 rounded-lg">
    <h2 class="text-lg font-semibold text-white border-b
border-green-600 pb-2">Tournament Details</h2>

    <div>
      <label for="gameTitle" class="block text-sm
font-medium text-gray-300">Game Title</label>
      <select id="gameTitle" name="gameTitle" required
class="mt-1 block w-full p-3 bg-gray-900
text-white border-gray-600 rounded-lg input-field neon-border">
        <option value="">Select Game (e.g., BGMI, Free
Fire, Valorant)</option>
        <option value="BGMI">Battlegrounds Mobile
India (BGMI)</option>
        <option value="FreeFire">Free Fire</option>
        <option value="Valorant">Valorant</option>
        <option value="LoL">League of Legends</option>
        <option value="CS2">Counter-Strike 2</option>
      </select>
    </div>

    <div>
      <label for="teamName" class="block text-sm
font-medium text-gray-300">Squad Name</label>
      <input type="text" id="teamName" name="teamName"
placeholder="Enter your squad's name" required
class="mt-1 block w-full p-3 bg-gray-900
text-white border border-gray-600 rounded-lg input-field neon-border">
    </div>

    <div>
      <label for="transactionId" class="block text-sm
font-medium text-gray-300">Payment Transaction ID (Mandatory)</label>
      <input type="text" id="transactionId"
name="transactionId" placeholder="Enter UPI/Payment Reference ID
(e.g., T2406151234567)" required
class="mt-1 block w-full p-3 bg-gray-900
text-white border border-gray-600 rounded-lg input-field neon-border">
      <p class="text-xs text-yellow-400 mt-1">This ID
must match the payment screenshot sent to WhatsApp.</p>
    </div>
  </div>

  <!-- Captain/Contact Information Section -->

```

```

        <div class="space-y-4 p-4 bg-gray-700/50 rounded-lg">
            <h2 class="text-lg font-semibold text-white border-b
border-green-600 pb-2">Squad Captain (Contact Person)</h2>

            <div>
                <label for="captainName" class="block text-sm
font-medium text-gray-300">Captain's Real Name</label>
                <input type="text" id="captainName"
name="captainName" placeholder="Your full legal name" required
                    class="mt-1 block w-full p-3 bg-gray-900
text-white border border-gray-600 rounded-lg input-field neon-border">
            </div>

            <div class="grid grid-cols-1 sm:grid-cols-2 gap-4">
                <div>
                    <label for="mobileNumber" class="block text-sm
font-medium text-gray-300">Mobile Number</label>
                    <input type="tel" id="mobileNumber"
name="mobileNumber" placeholder="Your 10-digit mobile number" required
pattern="[0-9]{10}"
                        class="mt-1 block w-full p-3 bg-gray-900
text-white border border-gray-600 rounded-lg input-field neon-border">
                </div>
                <div>
                    <label for="email" class="block text-sm
font-medium text-gray-300">Email Address (Optional)</label>
                    <input type="email" id="email" name="email"
placeholder="email@example.com"
                        class="mt-1 block w-full p-3 bg-gray-900
text-white border border-gray-600 rounded-lg input-field neon-border">
                </div>
            </div>

            <div>
                <label for="gamerTag" class="block text-sm
font-medium text-gray-300">Captain's In-Game Name / Tag</label>
                <input type="text" id="gamerTag" name="gamerTag"
placeholder="Your in-game display name" required
                    class="mt-1 block w-full p-3 bg-gray-900
text-white border border-gray-600 rounded-lg input-field neon-border">
            </div>
        </div>

        <!-- Roster Information Section -->
        <div class="space-y-4 p-4 bg-gray-700/50 rounded-lg">
            <h2 class="text-lg font-semibold text-white border-b
border-green-600 pb-2">Full Squad Roster (Including Captain)</h2>

```

```

        <div>
            <label for="roster" class="block text-sm
font-medium text-gray-300">
                Enter all player details, one player per line,
in the format:
                <span class="text-green-300 font-mono block
mt-1">Real Name | Game Name | UID</span>
            </label>
            <textarea id="roster" name="roster" rows="6"
placeholder="Example: Captain Real Name | Captain IGN | 1234567890
Player 2 Real Name | Player 2 IGN | 0987654321 Player 3 Real Name |
Player 3 IGN | 1122334455 Player 4 Real Name | Player 4 IGN |
2233445566" required
                class="mt-1 block w-full p-3 bg-gray-900
text-white border border-gray-600 rounded-lg input-field neon-border
scroll-container"></textarea>
            <p class="text-xs text-gray-400 mt-1">Ensure the
Captain's details are included in the roster list.</p>
        </div>
    </div>

    <!-- Checkbox and Submit -->
    <div class="flex items-start">
        <input id="waiver" name="waiver" type="checkbox"
required
            class="h-4 w-4 text-green-600 bg-gray-900
border-gray-600 rounded focus:ring-green-500 mt-1">
        <label for="waiver" class="ml-3 text-sm
text-gray-400">
            I confirm payment of ₹200 and agree to the <a
href="#" class="text-green-400 hover:text-green-300
font-medium">Tournament Rules</a>.
        </label>
    </div>

    <button type="submit" id="submit-button" class="w-full p-3
rounded-lg neon-button">
        Complete Registration
    </button>
    <div id="loading-indicator" class="hidden text-center
text-green-400 font-semibold">Processing Registration and Verifying
Payment...</div>
</form>
</div>

    <!-- Custom Modal for Success/Error -->
    <div id="status-modal" class="hidden fixed inset-0 z-50 flex
items-center justify-center modal-bg">

```

```

        <div class="bg-gray-800 p-8 rounded-xl shadow-2xl border
border-green-500 max-w-sm w-11/12 text-center">
            <h3 id="modal-title" class="text-2xl font-bold mb-4
text-white"></h3>
            <p id="modal-message" class="text-gray-300 mb-6"></p>
            <button id="modal-close-button" class="w-full p-3
rounded-lg neon-button">Close</button>
        </div>
    </div>

    <!-- Firebase SDKs and Logic -->
    <script type="module">
        import { initializeApp } from
"https://www.gstatic.com/firebasejs/11.6.1/firebase-app.js";
        import { getAuth, signInAnonymously, signInWithCustomToken }
from "https://www.gstatic.com/firebasejs/11.6.1/firebase-auth.js";
        import { getFirestore, collection, addDoc, setLogLevel } from
"https://www.gstatic.com/firebasejs/11.6.1/firebase-firestore.js";

        // Constants and Global State
        const submitButton = document.getElementById('submit-button');
        const loadingIndicator =
document.getElementById('loading-indicator');
        const formStatus = document.getElementById('form-status');
        const registrationForm =
document.getElementById('registration-form');
        const modal = document.getElementById('status-modal');
        const modalTitle = document.getElementById('modal-title');
        const modalMessage = document.getElementById('modal-message');
        const modalCloseButton =
document.getElementById('modal-close-button');

        let db, auth;
        let isAuthReady = false;

        // --- FIREBASE INITIALIZATION ---

        // Global variables provided by the environment
        const appId = typeof __app_id !== 'undefined' ? __app_id :
'default-app-id';
        const firebaseConfig = typeof __firebase_config !==
'undefined' ? JSON.parse(__firebase_config) : {};
        const initialAuthToken = typeof __initial_auth_token !==
'undefined' ? __initial_auth_token : null;

        /**
         * Initializes Firebase and authenticates the user.
         */

```

```

    async function initializeFirebase() {
        try {
            setLogLevel('Debug');
            const app = initializeApp(firebaseConfig);
            db = getFirestore(app);
            auth = getAuth(app);

            formStatus.textContent = 'Authenticating...';

            if (initialAuthToken) {
                await signInWithCustomToken(auth,
initialAuthToken);
            } else {
                await signInAnonymously(auth);
            }

            isAuthenticated = true;
            formStatus.textContent = 'Ready to Register! Database
Connection Established.';
            formStatus.classList.remove('text-yellow-400',
'text-red-500');
            formStatus.classList.add('text-green-400');
        } catch (error) {
            console.error("Firebase Initialization/Auth Error:",
error);

            formStatus.textContent = 'Connection Error.
Registration data will not be saved.';
            formStatus.classList.remove('text-yellow-400',
'text-green-400');
            formStatus.classList.add('text-red-500');
        }
    }

    // --- UTILITY FUNCTIONS ---

    /**
     * Shows the custom modal with a message.
     */
    function showModal(title, message, color = 'border-red-500') {
        modalTitle.textContent = title;
        modalMessage.innerHTML = message;

        modal.querySelector('div').classList.remove('border-red-500',
'border-green-500');
        modal.querySelector('div').classList.add(color);
        modal.classList.remove('hidden');
    }

```

```

    /**
     * Parses the raw roster text into a structured array of
player objects.
     * Expected format: Real Name | Game Name | UID
    */
    function parseRoster(rosterText) {
        const lines = rosterText.split('\n').map(line =>
line.trim()).filter(line => line.length > 0);
        const roster = [];
        for (const line of lines) {
            const parts = line.split('|').map(part =>
part.trim());
            if (parts.length >= 3) {
                roster.push({
                    realName: parts[0],
                    gameName: parts[1],
                    uid: parts[2]
                });
            } else {
                console.warn("Skipping roster line due to
incorrect format:", line);
            }
        }
        return roster;
    }

// --- FORM SUBMISSION HANDLER ---

/**
 * Handles the form submission and saves data to Firestore.
 */
async function handleRegistration(event) {
    event.preventDefault();

    if (!isAuthReady) {
        showModal("Connection Error", "The database connection
is not ready. Please refresh the page.", 'border-red-500');
        return;
    }

    submitButton.disabled = true;
    submitButton.textContent = 'Submitting...';
    loadingIndicator.classList.remove('hidden');

    const formData = new FormData(registrationForm);
    const rosterData = parseRoster(formData.get('roster'));

    if (rosterData.length === 0) {

```



```

        showModal("Roster Missing", "Please enter valid
player details in the Roster field using the required format (Real
Name | Game Name | UID).", 'border-red-500');
        submitButton.disabled = false;
        submitButton.textContent = 'Complete Registration';
        loadingIndicator.classList.add('hidden');
        return;
    }

    const teamData = {
        tournamentName: "Sribhumi eSports Championship",
        gameTitle: formData.get('gameTitle'),
        teamName: formData.get('teamName'),
        // Captain Details
        captainRealName: formData.get('captainName'),
        captainMobile: formData.get('mobileNumber'),
        captainEmail: formData.get('email') || 'N/A',
        captainGameTag: formData.get('gamerTag'),
        // Payment Details
        entryFee: '₹200',
        prizePool: '₹1,600',
        paymentTransactionId: formData.get('transactionId'),
        // Roster & Admin Data
        fullRoster: rosterData,
        waiverAccepted: formData.get('waiver') === 'on',
        registeredAt: new Date().toISOString(),
        userId: auth.currentUser?.uid || 'anonymous',
        appId: appId
    };

    try {
        // Public collection for organizer viewing/team
listing
        const collectionPath =
`/artifacts/${appId}/public/data/sribhumi_esports_registrations`;

        const docRef = await addDoc(collection(db,
collectionPath), teamData);

        console.log("Registration successfully submitted with
ID:", docRef.id);

        // Show success message with payment confirmation
instructions
        const successMessage = `
            <p class="text-lg font-semibold
text-green-300">${teamData.teamName} is registered!</p>
            <p class="mt-2">Your Registration ID:

```

```

<strong>${docRef.id}</strong></p>
      <p class="mt-4 text-sm text-yellow-300">
        **FINAL STEP:** Make sure you have sent the
**payment screenshot** to the WhatsApp number
        <a href="https://wa.me/918011866993"
target="_blank" class="text-green-300 hover:text-green-500 underline
font-extrabold">8011866993</a> for verification.
        Your registration will not be confirmed until
payment proof is verified.
      </p>
    `;
    showModal("Registration Successful!", successMessage,
'border-green-500');

    // Clear the form after successful submission
    registrationForm.reset();

    } catch (e) {
      console.error("Error submitting registration:", e);
      showModal("Submission Error",
        "There was an issue submitting your
registration. Error: " + e.message,
        'border-red-500');
    } finally {
      submitButton.disabled = false;
      submitButton.textContent = 'Complete Registration';
      loadingIndicator.classList.add('hidden');
    }
  }

  // --- EVENT LISTENERS & STARTUP ---

  // Attach event listener to the form
  registrationForm.addEventListener('submit',
handleRegistration);

  // Attach event listener to modal close button
  modalCloseButton.addEventListener('click', () => {
    modal.classList.add('hidden');
  });

  // Start Firebase initialization when the script loads
  initializeFirebase();

</script>
</body>
</html>

```