

demo\src\main\resources\static\admin_dashboard.js

```
const menuToggle = document.getElementById('menuToggle');const sidebar =
document.getElementById('sidebar');const mainContent =
document.getElementById('mainContent');const logoutBtn =
document.getElementById('logout');const sessionLoading =
document.getElementById('sessionLoading');const sessionTable =
document.getElementById('sessionTable');const userId =
sessionStorage.getItem("userId");function showModule(moduleId) {const
allModules = document.querySelectorAll('.module');allModules.forEach(mod =>
mod.style.display = 'none'); // hide allconst selectedModule =
document.getElementById(moduleId);if (selectedModule) {
selectedModule.style.display = 'block'; // show the selected one// Optional
logic for individual modulesif (moduleId === 'manageUsersModule') {
loadUsers(); // Function to load users dynamicallyif (moduleId ===
'reportsModule') {loadRecent20Users();}if (moduleId === 'adminProfileModule')
{loadUserProfile();}if (moduleId === 'settingsModule') {
initializeSettingsModule(); // + p This will run only when you open sett}
if (moduleId === 'chatModule') {loadChatMessages();}
// Toggle sidebar
menuToggle.addEventListener('click', () => {
sidebar.classList.toggle('active');
mainContent.classList.toggle('expanded');});function
formatDateTime(dateTimeStr) {if (!dateTimeStr) return 'N/A';const
date = new Date(dateTimeStr);return new Intl.DateTimeFormat('en-US', {
year: 'numeric',month: 'short',day: 'numeric',
hour: '2-digit',minute: '2-digit',second: '2-digit'
}).format(date);}function loadAdminDashboardData() {if (!userId) {
alert("Admin ID not found in session. Please login again.");
window.location.href = "adminSignin.html";return;}
sessionLoading.style.display = 'flex';sessionTable.style.display = 'none';
fetch(`http://localhost:8080/api/admin/${userId}`)
.then(response
=> {if (!response.ok) throw new Error('Failed to fetch admin
session');return response.json();})
.then(data => {
sessionLoading.style.display = 'none';
sessionTable.style.display = 'table';
document.getElementById("userIdDisplay").textContent = data.userId || "N/A";
document.getElementById("userNameDisplay").textContent =
data.userName || "N/A";
document.getElementById("signinTimeDisplay").textContent =
formatDateTime(data.signinTime);
document.getElementById("logoutTimeDisplay").textContent = data.logoutTime
? formatDateTime(data.logoutTime): "Active";
document.getElementById("roleDisplay").textContent = data.role ||
"ADMIN";const statusDisplay =
document.getElementById("statusDisplay");const sidebarStatus =
document.getElementById("adminStatus");if (data.isActive) {
statusDisplay.innerHTML = '<span style="color:
green;">Online</span>';sidebarStatus.textContent = "Online";
sidebarStatus.style.color = "green";} else {
statusDisplay.innerHTML = '<span style="color: red;">Offline</
span>';sidebarStatus.textContent = "Offline";
sidebarStatus.style.color = "red";}
document.getElementById("sidebarAdminName").textContent = data.userName ||
"Unknown";document.getElementById("headerAdminName").textContent
= data.userName || "Unknown";
document.getElementById("sidebarAdminRole").textContent = data.role ||
"Admin";document.getElementById("headerAdminRole").textContent =
`${data.role || "Admin"}`;const avatar = data.userName ?
data.userName.charAt(0).toUpperCase() : "?";
document.getElementById("adminAvatar").textContent = avatar;
sessionStorage.setItem("role", data.role);}
```

```

        console.error("Error loading admin dashboard:", error);
        sessionLoading.style.display = 'none';
        alert("Failed to load dashboard data. Please login again.");
        window.location.href = "adminSignin.html";
        logoutBtn.addEventListener('click', function () {
            if (!userId) {
                alert("No admin ID found. Already logged out.");
                window.location.href = "adminSignin.html";
                return;
            }
            fetch('http://localhost:8080/api/admin/logout', {
                method: 'POST',
                headers: {
                    'Content-Type': 'application/json'
                },
                body: JSON.stringify({
                    userId: userId
                })
            }).then(response => {
                if (!response.ok) throw new Error('Logout failed');
                sessionStorage.clear();
                window.location.href = "adminSignin.html";
            }).catch(error => {
                console.error('Logout error:', error);
                alert('Logout failed. Try again.');
```

```

        return (User.userId.toLowerCase().includes(query)
        || User.userName.toLowerCase().includes(query)) || !
        user.role.toLowerCase().includes(query));
    }
    if (sortBy &&
        filteredUsers.length > 0) {
        filteredUsers.sort((a, b) => {
            const aVal =
            (a[sortBy] || "").toLowerCase();
            const bVal = (b[sortBy] || "").toLowerCase();
            return aVal.localeCompare(bVal);
        });
        displayAllUsers(filteredUsers);
    }
    document.addEventListener("DOMContentLoaded", () => {
        loadUsers();
        document.getElementById("allUserSearchInput").addEventListener("input",
        filterAndSortAllUsers);
        document.getElementById("allUserSortSelect").addEventListener("change",
        filterAndSortAllUsers);
    });
    let recentUsers = [];
    async function
    loadRecent20Users() {
        const loadingDiv =
        document.getElementById("recentUsersLoading");
        const table =
        document.getElementById("recentUsersTable");
        const tbody =
        document.getElementById("recentUsersTableBody");
        loadingDiv.style.display =
        "block";
        table.style.display = "none";
        try {
            const response = await
            fetch("http://localhost:8080/api/users/recent");
            if (!response.ok) throw new
            Error("Failed to fetch users");
            recentUsers = await response.json();
            // make
            sure this gets filled
            displayUsers(recentUsers);
        } catch (error) {
            console.error("Error loading users:", error);
            loadingDiv.innerHTML = "Error
            loading user data.";
        }
        function displayUsers(users) {
            const tbody =
            document.getElementById("recentUsersTableBody");
            const loadingDiv =
            document.getElementById("recentUsersLoading");
            const table =
            document.getElementById("recentUsersTable");
            tbody.innerHTML = "";
            if
            (users.length === 0) {
                tbody.innerHTML = `<tr><td colspan="4">No matching
                users found.</td></tr>`;
            } else {
                users.forEach(user => {
                    const row =
                    document.createElement("tr");
                    row.innerHTML = `<td>${user.userId}</td>
                    <td>${user.userName}</td>
                    <td>${user.role}</td>
                    <td>${formatDate(user.createdAt)}</td>`;
                    tbody.appendChild(row);
                });
            }
            loadingDiv.style.display = "none";
            table.style.display = "table";
        }
        function
        formatDate(dateString) {
            const date = new Date(dateString);
            return
            date.toLocaleString();
        }
        function filterAndSortUsers() {
            const query =
            document.getElementById("searchInput").value.toLowerCase();
            const sortBy =
            document.getElementById("sortSelect").value;
            let filteredUsers =
            recentUsers.filter(user => {
                return (User.userId.toLowerCase().includes(query)
                || User.userName.toLowerCase().includes(query)) || !
                user.role.toLowerCase().includes(query);
            });
            if (sortBy &&
                filteredUsers.length > 0) {
                filteredUsers.sort((a, b) => {
                    const aVal =
                    (a[sortBy] || "").toLowerCase();
                    const bVal = (b[sortBy] || "").toLowerCase();
                    return aVal.localeCompare(bVal);
                });
                displayUsers(filteredUsers);
            }
            document.addEventListener("DOMContentLoaded", () => {
                loadRecent20Users();
                document.getElementById("searchInput").addEventListener("input",
                filterAndSortUsers);
                document.getElementById("sortSelect").addEventListener("change",
                filterAndSortUsers);
            });
            async function downloadUserReport() {
                const { jsPDF } =
                window.jspdf;
                const doc = new jsPDF();
                const title = "User Management Report";
                const date = new Date().toLocaleString();
                // Add optional logo (uncomment and
                use base64 image if needed)
                // doc.addImage("data:image/png;base64,...",
                "PNG", 10, 10, 30, 30);
                // Add Title
                doc.setFontSize(18);
                doc.text(title, 14,
                20);
                // Add Date
                doc.setFontSize(10);
                doc.setTextColor(100);
                doc.text("Generated
                on: " + date, 14, 27);
                // Grab the user table
                const table =
                document.getElementById("userTable");
                const headers =
                Array.from(table.querySelectorAll("thead th")).map(th => th.innerHTML);
                const
                rows = Array.from(table.querySelectorAll("tbody tr")).map(tr =>
                Array.from(tr.querySelectorAll("td")).map(td => td.innerHTML));
                // Create the
                PDF table
                doc.autoTable({
                    head: [headers],
                    body: rows,
                    startY: 35,
                    theme: "grid",
                    styles: {
                        fontSize: 10,
                        cellPadding: 3,
                        headStyles: {
                            fillColor: [60, 141, 188],
                            textColor: 255,
                            fontStyle: 'bold',
                        },
                        alternateRowStyles: {
                            fillColor: [245, 245,
                            245]
                        }
                    }
                });
                // Download file
                const fileName = "User_Report_" + new
                Date().toISOString().slice(0, 10) + ".pdf";
            }
        }
    }

```

```

doc.save(fileName);}); async function
downloadAllUsersReport() {const { jsPDF } = window.jspdf;const doc = new
jsPDF();const title = "All Users Report";const date = new
Date().toLocaleString();// Optional: Add your logo (base64 format image)//
doc.addImage("data:image/png;base64,...", "PNG", 10, 10, 30, 30);// Titledoc
doc.setFontSize(18);doc.text(title, 14, 20);// Datedoc.setFontSize(10);doc
doc.setTextColor(100);doc.text("Generated on: " + date, 14, 27);// Fetch
latest all users datadry {const response = await fetch("http://localhost:8080/
api/all-users");if (!response.ok) throw new Error("Failed to fetch users");const
users = await response.json();if (!users.length) {alert("No users found
to generate the report.");return;}// Format headers and rowsconst headers =
[["User ID", "User Name", "Email", "Phone", "Role"]];const rows =
users.map(user => [User.userId,User.userName,User.email,User.phone,User.roledoc
]);// Create the table in the PDFdoc.autoTable({head: headers,body: rows,doc
startY: 35,doc.theme: "grid",doc.styles: {doc.fontSize: 10,doc.cellPadding: 3,doc.headStyles: {doc
fillColor: [60, 141, 188],doc.textColor: 255,doc.fontStyle: 'bold',doc.alternateRowStyles: {doc.fillColor: [245, 245, 245]}doc});// Save PDFconst fileName
= "All_Users_Report_" + new Date().toISOString().slice(0, 10) + ".pdf";doc
doc.save(fileName);doc catch (error) {console.error("Error generating PDF:",
error);alert("Failed to generate report.");function loadUserProfile() {const
userId = sessionStorage.getItem("userId");if (!userId) {doc
console.error("No userId found in sessionStorage");return;}fetch(`http://
localhost:8080/api/profile/${userId}`)then(response => {if (!response.ok) {doc
throw new Error("Failed to fetch user profile");return response.json();}doc
.then(data => {document.getElementById("profileUserId").textContent =
data.userId || "N/A";document.getElementById("profileUserName").textContent =
data.userName || "N/A";document.getElementById("profileEmail").textContent =
data.email || "N/A";document.getElementById("profilePhone").textContent =
data.phone || "N/A";document.getElementById("profileRole").textContent =
data.role || "N/A";document.getElementById("profileReferralCode").textContent
= data.referralCode || "N/A";document.getElementById("profileCreatedAt").textContent = data.createdAt ||
"N/A";const pic = localStorage.getItem("adminProfileImage");if (pic) {doc
document.getElementById("profileImage").src = pic;doc
document.getElementById("settingsImage").src = pic;doc
document.getElementById("profileImage").classList.add("visible");doc
document.getElementById("settingsImage").classList.add("visible");doc }doc }doc
.catch(error => {console.error("Error loading profile:", error);doc }doc // Load
data on page loadfunction initializeSettingsModule() {const userId =
sessionStorage.getItem("userId");if (!userId) {doc console.error("No
userId found in sessionStorage");doc showStatus("No user session found",
true);doc return;doc }doc fetch(`http://localhost:8080/api/profile/
${userId}`)then(res => res.json())then(data => {doc
document.getElementById("currentUsername").textContent = data.userName || "N/
A";doc document.getElementById("currentPhone").textContent =
data.phone || "N/A";doc })doc .catch(() => showStatus("Failed to load
profile", true));doc // Edit profile (triggers both username and phone edit)doc
function editProfile() {doc editUsername();doc editPhone();doc
document.getElementById("editProfileBtn").style.display = "none";doc // Edit
usernamefunction editUsername() {doc
document.getElementById("usernameEdit").classList.add("active");doc
document.getElementById("usernameInput").value =
document.getElementById("currentUsername").textContent;doc
document.getElementById("currentUsername").style.display = "none";doc // Edit
phonefunction editPhone() {doc
document.getElementById("phoneEdit").classList.add("active");doc
document.getElementById("phoneInput").value =
document.getElementById("currentPhone").textContent;doc
document.getElementById("currentPhone").style.display = "none";doc // Cancel editdoc
function cancelEdit(field) {doc

```

```

        if (field === 'username') {
document.getElementById("usernameEdit").classList.remove("active");
document.getElementById("currentUsername").style.display = "inline-block";
        } else if (field === 'phone') {
document.getElementById("phoneEdit").classList.remove("active");
document.getElementById("currentPhone").style.display = "inline-block";
        }
        // Show Edit button only if both edit sections are inactive
        if (!document.getElementById("usernameEdit").classList.contains("active") &&
            !document.getElementById("phoneEdit").classList.contains("active")) {
document.getElementById("editProfileBtn").style.display = "inline-block";
        }
        // Save username with validation
function saveUsername() {
const updatedName = document.getElementById("usernameInput").value.trim();
const userId = sessionStorage.getItem("userId");
const usernameRegex = /^(?!\\s)(?!.*\\s){2,}[a-zA-Z0-9 ]+(?!\\s)$/;
if (!updatedName) {
showStatus("Username cannot be empty", true);
return;
}
if (!usernameRegex.test(updatedName)) {
showStatus("Username must contain only letters, numbers, and single spaces, and not start/end with space", true);
return;
}
fetch("http://localhost:8080/api/admin/update-username", {
method: "POST",
headers: { "Content-Type": "application/json" },
body: JSON.stringify({ userId, username: updatedName })
})
.then(res => {
if (!res.ok) throw new Error("Failed to update");
return res.json();
})
.then(data => {
document.getElementById("currentUsername").textContent = data.userName;

document.getElementById("usernameEdit").classList.remove("active");
document.getElementById("currentUsername").style.display = "inline-block";
showStatus("Username updated successfully.");
        // Show Edit button if both edit sections are inactive
        if (!document.getElementById("phoneEdit").classList.contains("active")) {
document.getElementById("editProfileBtn").style.display = "inline-block";
        }
    })
    .catch(() => showStatus("Failed to update username", true));
}
        // Save phone with validation
function savePhone() {
const updatedPhone = document.getElementById("phoneInput").value.trim();
const userId = sessionStorage.getItem("userId");
const phoneRegex = /^[6-9]\\d{9}$/;
if (!updatedPhone) {
showStatus("Phone number cannot be empty", true);
return;
}
if (!phoneRegex.test(updatedPhone)) {
showStatus("Phone number must start with 6-9 and be 10 digits", true);
return;
}
fetch("http://localhost:8080/api/admin/update-phone", {
method: "POST",
headers: { "Content-Type": "application/json" },
body: JSON.stringify({ userId, phone: updatedPhone })
})
.then(res => {
if (!res.ok) throw new Error("Failed to update");
return res.json();
})
.then(data => {
document.getElementById("currentPhone").textContent = data.phone;
document.getElementById("phoneEdit").classList.remove("active");
document.getElementById("currentPhone").style.display = "inline-block";
showStatus("Phone number updated successfully.");
        // Show Edit button if both edit sections are inactive
        if (!document.getElementById("usernameEdit").classList.contains("active")) {
document.getElementById("editProfileBtn").style.display = "inline-block";
        }
    })
    .catch(() => showStatus("Failed to update phone number", true));
}
        // Show status
function showStatus(message, isError = false) {
const el = document.getElementById("settingsStatus");
el.textContent = message;
el.style.color = isError ? "red" : "green";
el.classList.add("visible");
setTimeout(() => {
el.classList.remove("visible");
el.textContent = "";
}, 3000);
}
        // Show success popup
function showSuccessPopup(message) {
const popup = document.getElementById("successPopup");
const messageEl = document.getElementById("successMessage");

```

```

        messageEl.textContent = message;
        popup.style.display = "block";
        setTimeout(() => {
            closePopup();
        }, 3000); // Auto-close after 3 seconds
    }
    // Close popup function
    closePopup() {
        const popup = document.getElementById("successPopup");
        popup.style.display = "none";
    }
    // Show status for failures
    showStatus(message, isError = false) {
        const el = document.getElementById("settingsStatus");
        el.textContent = message;
        el.style.color = isError ? "red" : "green";
        el.classList.add("visible");
        setTimeout(() => {
            el.classList.remove("visible");
        }, 3000);
    }
    // Profile picture
    const profilePic = document.getElementById("profilePic");
    const settingsPic = document.getElementById("settingsPic");
    const uploadInput = document.getElementById("uploadPicInput");
    const pic = localStorage.getItem("adminProfileImage");
    if (pic) {
        document.getElementById("profileImage").src = pic;
        document.getElementById("settingsImage").src = pic;
        document.getElementById("profileImage").classList.add("visible");
        document.getElementById("settingsImage").classList.add("visible");
    }
    // Profile picture
    document.getElementById("uploadInput").addEventListener("change", function (e) {
        const file = e.target.files[0];
        if (file && file.type.startsWith("image/")) {
            const reader = new FileReader();
            reader.onload = function (event) {
                const base64Image = event.target.result;
                localStorage.setItem("adminProfileImage", base64Image);
                document.getElementById("profileImage").src = base64Image;
                document.getElementById("settingsImage").src = base64Image;
                document.getElementById("profileImage").classList.add("visible");
                document.getElementById("settingsImage").classList.add("visible");
            };
            reader.readAsDataURL(file);
        }
    });
    // Run on page load
    initializeSettingsModule();
    let currentAdminId = 'ADMIN_0001'; // Replace with actual admin ID
    let isAdmin = true; // true for admin
    let receiverId = 'USER_0001'; // Replace with the logged-in user's ID
    // Fetch chat messages from backend
    function loadChatMessages() {
        fetch(`http://localhost:8080/chat/conversation?userId=${receiverId}&adminId=${currentAdminId}`)
            .then(res => res.json())
            .then(data => {
                const chatBox = document.getElementById('chatBox');
                chatBox.innerHTML = ''; // Clear previous messages
                data.forEach(msg => {
                    const div = document.createElement('div');
                    div.classList.add('chat-message');
                    if (msg.senderId === currentAdminId) {
                        div.classList.add('admin-message');
                    } else {
                        div.classList.add('user-message');
                    }
                    // Format timestamp to HH:MM AM/PM
                    const timestamp = new Date(msg.timestamp);
                    const timeString = timestamp.toLocaleTimeString('en-US', {
                        hour12: true,
                        timeZone: 'Asia/Kolkata' // Change to your preferred time zone
                    });
                    div.innerHTML = `<span>${msg.message}</span><br><small class="timestamp">${timeString}</small>`;
                    chatBox.appendChild(div);
                });
                chatBox.scrollTop = chatBox.scrollHeight; // Scroll to bottom
            })
            .catch(err => {
                console.error('Error loading chat messages:', err);
            });
    }
    // Send chat message
    function sendChatMessage() {
        const input = document.getElementById('chatInput');
        const message = input.value.trim();
        if (!message) return;
        const payload = {
            senderId: currentAdminId,
            receiverId: receiverId,
            message: message,
            timestamp: new Date().toISOString()
        };
        fetch('http://localhost:8080/chat/send', {
            method: 'POST',
            headers: { 'Content-Type': 'application/json' },
            body: JSON.stringify(payload)
        })
            .then(res => {
                if (res.ok) {
                    input.value = ''; // Clear input field
                    loadChatMessages(); // Refresh chat
                }
            })
            .catch(err => {
                console.error('Error sending message:', err);
            });
    }
    // Auto-refresh chat every 5 seconds
    setInterval(loadChatMessages, 10000);
    window.onload = loadChatMessages;

```

demo\src\main\resources\static\user_dashboard3.js

```

    const menuToggle = document.getElementById('menuToggle');
    const sidebar = document.getElementById('sidebar');
    const mainContent = document.getElementById('mainContent');
    const logoutBtn = document.getElementById('logout');
    const sessionLoading = document.getElementById('sessionLoading');
    const sessionTable = document.getElementById('sessionTable');
    const userId = sessionStorage.getItem("userId");
    function showModule(moduleId) {
        const allModules = document.querySelectorAll('.module');
        allModules.forEach(mod => {
            mod.style.display = 'none'; // hide all
        });
        const selectedModule = document.getElementById(moduleId);
        if (selectedModule) {
            selectedModule.style.display = 'block'; // show the selected one
        }
        if (moduleId === 'userProfileModule') {
            loadUserProfile();
        }
        if (moduleId === 'settingsModule') {
            initializeSettingsModule(); // + p This will run only when you open settings
        }
        if (moduleId === 'chatModule') {
            loadChatMessages();
        }
        // Toggle sidebar
        menuToggle.addEventListener('click', () => {
            sidebar.classList.toggle('active');
            mainContent.classList.toggle('expanded');
        });
        // Format datetime for display
        function formatDateTime(dateTimeStr) {
            if (!dateTimeStr) return 'N/A';
            const date = new Date(dateTimeStr);
            return new Intl.DateTimeFormat('en-US', {
                year: 'numeric',
                month: 'short',
                day: 'numeric',
                hour: '2-digit',
                minute: '2-digit',
                second: '2-digit'
            }).format(date);
        }
        // Load dashboard data
        function loadDashboardData() {
            if (!userId) {
                alert("User ID not found in session. Please login again.");
                return;
            }
            sessionLoading.style.display = 'flex';
            sessionTable.style.display = 'none';
            fetch(`http://localhost:8080/api/user_sessions/${userId}`)
                .then(response => {
                    if (!response.ok) throw new Error('Failed to fetch session');
                    return response.json();
                })
                .then(data => {
                    sessionLoading.style.display = 'none';
                    sessionTable.style.display = 'table'; // Session data
                    document.getElementById("userIdDisplay").textContent = data.userId || "N/A";
                    document.getElementById("userNameDisplay").textContent = data.userName || "N/A";
                    document.getElementById("signinTimeDisplay").textContent = formatDateTime(data.signinTime);
                    document.getElementById("logoutTimeDisplay").textContent = data.logoutTime ? formatDateTime(data.logoutTime) : "Active";
                    document.getElementById("roleDisplay").textContent = data.role || "USER";
                    // Status
                    const statusDisplay = document.getElementById("statusDisplay");
                    sidebarUserStatus = document.getElementById("userStatus");
                    if (data.isActive) {
                        statusDisplay.innerHTML = '<span style="color: green;">Online</span>';
                        sidebarUserStatus.textContent = "Online";
                        sidebarUserStatus.style.color = "green";
                    } else {
                        statusDisplay.innerHTML = '<span style="color: red;">Offline</span>';
                        sidebarUserStatus.textContent = "Offline";
                        sidebarUserStatus.style.color = "red";
                    }
                });
            // User Info
            document.getElementById("sidebarUserName").textContent = data.userName ||

```



```

"Unknown";
document.getElementById("headerUserName").textContent = data.userName ||
"Unknown";
document.getElementById("sidebarUserRole").textContent = data.role || "User";
document.getElementById("headerUserRole").textContent =
`${data.role || "User"}`;
const avatar =
data.userName ? data.userName.charAt(0).toUpperCase() : "?";
document.getElementById("userAvatar").textContent =
avatar;
sessionStorage.setItem("role", data.role);
})
.catch(error => {
console.error("Error loading dashboard data:", error);
sessionLoading.style.display = 'none';
loadDashboardData().then(() => {
loadDashboardData().Please login again.");
window.location.href = "signin.html";
});
// Handle logout
logoutBtn.addEventListener('click', function () {
if (!userId) {
alert("No user ID found.
Already logged out.");
window.location.href = "signin.html";
return;
}
fetch('http://localhost:8080/api/logout', {
method: 'POST',
headers: {
'Content-Type': 'application/json',
body: JSON.stringify({ userId: userId })
})
.then(response => {
if (!response.ok) throw new Error('Logout failed');
sessionStorage.clear();
window.location.href = "signin.html";
})
.catch(error => {
console.error('Logout error:', error);
alert('Logout failed. Try again.');
});
// Init dashboard
document.addEventListener('DOMContentLoaded', function () {
loadDashboardData();
setInterval(loadDashboardData, 30000); // Refresh every 30 seconds
function loadUserProfile() {
const userId = sessionStorage.getItem("userId");
if (!userId) {
console.error("No userId found in sessionStorage");
return;
}
fetch(`http://localhost:8080/api/profile/${userId}`)
.then(response => {
if (!response.ok) {
throw new Error("Failed to fetch user profile");
}
return response.json();
})
.then(data => {
document.getElementById("profileUserId").textContent = data.userId || "N/A";
document.getElementById("profileUserName").textContent = data.userName || "N/A";
document.getElementById("profileEmail").textContent = data.email || "N/A";
document.getElementById("profilePhone").textContent = data.phone || "N/A";
document.getElementById("profileRole").textContent = data.role || "N/A";
document.getElementById("profileReferralCode").textContent = data.referralCode || "N/A";
document.getElementById("profileCreatedAt").textContent = data.createdAt || "N/A";
const pic = localStorage.getItem("userProfileImage");
if (pic) {
document.getElementById("profileImage").src = pic;
document.getElementById("settingsImage").src = pic;
document.getElementById("profileImage").classList.add("visible");
document.getElementById("settingsImage").classList.add("visible");
}
})
.catch(error => {
console.error("Error loading profile:", error);
});
// Load data on page load
function initializeSettingsModule() {
const userId = sessionStorage.getItem("userId");
if (!userId) {
console.error("No user ID found in sessionStorage");
showStatus("No user session found", true);
return;
}
fetch(`http://localhost:8080/api/profile/${userId}`)
.then(res => res.json())
.then(data => {
document.getElementById("currentUsername").textContent = data.userName || "N/A";

```

```

document.getElementById("currentPhone").textContent = data.phone || "N/A";
    })
    .catch(() => showStatus("Failed to load
profile", true));
    }
    // Edit profile (triggers both username and
phone edit)
    function editProfile() {
        editUsername();
        editPhone();
    }
document.getElementById("editProfileBtn").style.display = "none";
    // Edit username
    function editUsername() {
document.getElementById("usernameEdit").classList.add("active");
document.getElementById("usernameInput").value =
document.getElementById("currentUsername").textContent;
document.getElementById("currentUsername").style.display = "none";
    // Edit phone
    function editPhone() {
document.getElementById("phoneEdit").classList.add("active");
document.getElementById("phoneInput").value =
document.getElementById("currentPhone").textContent;
document.getElementById("currentPhone").style.display = "none";
    // Cancel edit
    function cancelEdit(field) {
        if
        (field === 'username') {
document.getElementById("usernameEdit").classList.remove("active");
            document.getElementById("currentUsername").style.display =
"inline-block";
        } else if (field === 'phone') {
document.getElementById("phoneEdit").classList.remove("active");
            document.getElementById("currentPhone").style.display =
"inline-block";
        }
        // Show Edit button only if both edit
sections are inactive
        if (!
document.getElementById("usernameEdit").classList.contains("active") &&
!
document.getElementById("phoneEdit").classList.contains("active")) {
            document.getElementById("editProfileBtn").style.display =
"inline-block";
        }
        // Save username with validation
        function saveUsername() {
            const updatedName =
document.getElementById("usernameInput").value.trim();
            const
userId = sessionStorage.getItem("userId");
            const usernameRegex = /
^(?!\\s)(?!.*\\s){2,}[a-zA-Z0-9 ]+(?!\\s)$/;
            if (!updatedName) {
showStatus("Username cannot be empty", true);
            }
            if (!usernameRegex.test(updatedName)) {
showStatus("Username must contain only letters, numbers, and
single spaces, and not start/end with space", true);
            }
            return;
        }
        fetch("http://localhost:8080/api/update-username", {
            method: "POST",
            headers: { "Content-Type":
"application/json" },
            body: JSON.stringify({ userId, userName:
updatedName })
        })
        .then(res => {
            if (!res.ok) throw new Error("Failed to update");
            return res.json();
        })
        .then(data => {
document.getElementById("currentUsername").textContent = data.userName;

document.getElementById("usernameEdit").classList.remove("active");
            document.getElementById("currentUsername").style.display
= "inline-block";
            showStatus("Username updated
successfully.");
            // Show Edit button if both edit sections
are inactive
            if (!
document.getElementById("phoneEdit").classList.contains("active")) {
document.getElementById("editProfileBtn").style.display = "inline-block";
            }
        })
        .catch(() =>
showStatus("Failed to update username", true));
        // Save phone
with validation
        function savePhone() {
            const updatedPhone =
document.getElementById("phoneInput").value.trim();

```

```

const userId =
sessionStorage.getItem("userId");
const phoneRegex = /
^[6-9]\d{9}$/;
if (!updatedPhone) {
showStatus("Phone number cannot be empty", true);
return;
}
if (!phoneRegex.test(updatedPhone)) {
showStatus("Phone number must start with 6-9 and be 10 digits", true);
return;
}
fetch("http://localhost:8080/
api/update-phone", {
method: "POST",
headers:
{ "Content-Type": "application/json" },
body:
JSON.stringify({ userId, phone: updatedPhone })
})
.then(res => {
if (!res.ok) throw new
Error("Failed to update");
return res.json();
})
.then(data => {
document.getElementById("currentPhone").textContent = data.phone;

document.getElementById("phoneEdit").classList.remove("active");
document.getElementById("currentPhone").style.display =
"inline-block";
showStatus("Phone number updated
successfully.");
// Show Edit button if both edit sections
are inactive
if (!
document.getElementById("usernameEdit").classList.contains("active")) {

document.getElementById("editProfileBtn").style.display = "inline-block";
}
})
.catch(() =>
showStatus("Failed to update phone number", true));
}
//
Show status
function showStatus(message, isError = false) {
const el = document.getElementById("settingsStatus");
el.textContent = message;
el.style.color = isError ? "red" :
"green";
el.classList.add("visible");
setTimeout(() => {
el.classList.remove("visible");
el.textContent
= "";
}, 3000);
}
// Show success popup
function showSuccessPopup(message) {
const popup =
document.getElementById("successPopup");
const messageEl =
document.getElementById("successMessage");
messageEl.textContent =
message;
popup.style.display = "block";
setTimeout(()
=> {
closePopup();
}, 3000);
// Auto-close after 3
seconds
}
// Close popup
function closePopup() {
const popup = document.getElementById("successPopup");
popup.style.display = "none";
}
// Show status for failures
function showStatus(message, isError = false) {
const el =
document.getElementById("settingsStatus");
el.textContent =
message;
el.style.color = isError ? "red" : "green";
el.classList.add("visible");
setTimeout(() => {
el.classList.remove("visible");
el.textContent = "";
}, 3000);
}
const profilePic =
document.getElementById("profilePic");
const settingsPic =
document.getElementById("settingsPic");
const uploadInput =
document.getElementById("uploadPicInput");
const pic =
localStorage.getItem("userProfileImage");
if (pic) {
document.getElementById("profileImage").src = pic;
document.getElementById("settingsImage").src = pic;
document.getElementById("profileImage").classList.add("visible");
document.getElementById("settingsImage").classList.add("visible");
}
// Profile picture
document.getElementById("uploadInput").addEventListener("change", function
(e) {
const file = e.target.files[0];
if (file &&
file.type.startsWith("image/")) {
const reader = new FileReader();
reader.onload = function (event) {
const base64Image =
event.target.result;
localStorage.setItem("userProfileImage",
base64Image);
}
}
}

```

```

        document.getElementById("profileImage").src =
base64Image;
        document.getElementById("settingsImage").src =
base64Image;
document.getElementById("profileImage").classList.add("visible");
document.getElementById("settingsImage").classList.add("visible");
showStatus("Profile picture updated successfully.");
reader.readAsDataURL(file);
initializeSettingsModule();
function showCustomNotification() {
const notificationArea =
document.getElementById("notificationArea");
notificationArea.style.display
= "block"; // Show the notification
// Event listener to hide the
notification when clicked
notificationArea.addEventListener('click', () => {
notificationArea.style.display = "none"; // Hide on click
// Hide
the notification after 3 seconds if not clicked
setTimeout(() => {
if
(notificationArea.style.display !== "none") {
notificationArea.style.display = "none";
}
}, 3000);
const currentUserId =
'USER_0001'; // Replace with actual logged-in ID
const isAdmin = false; // true
if admin, false if user
const receiverId = 'ADMIN_0001'; // Replace with actual
receiver (admin ID)
// Fetch chat messages from backend
function
loadChatMessages() {
fetch('http://localhost:8080/chat/conversation?
userId=${currentUserId}&adminId=ADMIN_0001') // Pass userId and adminId
.then(res => res.json())
.then(data => {
const chatBox =
document.getElementById('chatBox');
chatBox.innerHTML = ''; // Clear
previous messages
data.forEach(msg => {
const div =
document.createElement('div');
div.classList.add('chat-message');
// Add different class based on sender
if (msg.senderId ===
currentUserId) {
div.classList.add('user-message');
} else {
div.classList.add('admin-message');
}
// Format
timestamp to time only
const timeOnly = new
Date(msg.timestamp).toLocaleTimeString('en-IN', {
hour: '2-digit',
minute: '2-digit',
hour12: true,
timeZone: 'Asia/
Kolkata'
});
// Combine message and time
div.innerHTML =
`<span>${msg.message}</span>
<br>
<small
class="timestamp">${timeOnly}</small>`;
chatBox.appendChild(div);
});
chatBox.scrollTop =
chatBox.scrollHeight; // Scroll to the bottom of the chat
}).catch(err
=> {
console.error('Error loading chat messages:', err);
});
// Send
chat message
function sendChatMessage() {
const input =
document.getElementById('chatInput');
const message = input.value.trim();
if (!message) return;
const payload = {
senderId: currentUserId,
receiverId: receiverId,
message: message,
timestamp: new
Date().toISOString() // Add timestamp
};
fetch('http://localhost:8080/chat/
send', {
method: 'POST',
headers: { 'Content-Type': 'application/
json' },
body: JSON.stringify(payload)
})
.then(res => {
if
(res.ok) {
input.value = ''; // Clear the input field
loadChatMessages(); // Reload chat after sending
}).catch(err
=> {
console.error('Error sending message:', err);
});
// Auto-
refresh chat every 5 seconds
setInterval(loadChatMessages, 10000);
window.onload = loadChatMessages; // Load chat on page load

```

demo\target\classes\static\admin_dashboard.js

```
const menuToggle = document.getElementById('menuToggle');const sidebar =
document.getElementById('sidebar');const mainContent =
document.getElementById('mainContent');const logoutBtn =
document.getElementById('logout');const sessionLoading =
document.getElementById('sessionLoading');const sessionTable =
document.getElementById('sessionTable');const userId =
sessionStorage.getItem("userId");function showModule(moduleId) {const
allModules = document.querySelectorAll('.module');allModules.forEach(mod =>
mod.style.display = 'none'); // hide allconst selectedModule =
document.getElementById(moduleId);if (selectedModule) {
selectedModule.style.display = 'block'; // show the selected one// Optional
logic for individual modulesif (moduleId === 'manageUsersModule') {
loadUsers(); // Function to load users dynamicallyif (moduleId ===
'reportsModule') {loadRecent20Users();}if (moduleId === 'adminProfileModule')
{loadUserProfile();}if (moduleId === 'settingsModule') {
initializeSettingsModule(); // + p This will run only when you open sett}
if (moduleId === 'chatModule') {loadChatMessages();}
// Toggle sidebar
menuToggle.addEventListener('click', () => {
sidebar.classList.toggle('active');
mainContent.classList.toggle('expanded');});function
formatDateTime(dateTimeStr) {if (!dateTimeStr) return 'N/A';const
date = new Date(dateTimeStr);return new Intl.DateTimeFormat('en-US', {
year: 'numeric',month: 'short',day: 'numeric',
hour: '2-digit',minute: '2-digit',second: '2-digit'
}).format(date);}function loadAdminDashboardData() {if (!userId) {
alert("Admin ID not found in session. Please login again.");
window.location.href = "adminSignin.html";return;}
sessionLoading.style.display = 'flex';sessionTable.style.display = 'none';
fetch(`http://localhost:8080/api/admin/${userId}`)
.then(response => {
if (!response.ok) throw new Error('Failed to fetch admin
session');return response.json();})
.then(data => {
sessionLoading.style.display = 'none';
sessionTable.style.display = 'table';
document.getElementById("userIdDisplay").textContent = data.userId || "N/A";
document.getElementById("userNameDisplay").textContent =
data.userName || "N/A";
document.getElementById("signinTimeDisplay").textContent =
formatDateTime(data.signinTime);
document.getElementById("logoutTimeDisplay").textContent = data.logoutTime
? formatDateTime(data.logoutTime) : "Active";
document.getElementById("roleDisplay").textContent = data.role ||
"ADMIN";const statusDisplay =
document.getElementById("statusDisplay");const sidebarStatus =
document.getElementById("adminStatus");if (data.isActive) {
statusDisplay.innerHTML = '<span style="color:
green;">Online</span>';
sidebarStatus.textContent = "Online";
sidebarStatus.style.color = "green";
} else {
statusDisplay.innerHTML = '<span style="color: red;">Offline</
span>';
sidebarStatus.textContent = "Offline";
}
document.getElementById("sidebarAdminName").textContent = data.userName ||
"Unknown";
document.getElementById("headerAdminName").textContent
= data.userName || "Unknown";
document.getElementById("sidebarAdminRole").textContent = data.role ||
"Admin";
document.getElementById("headerAdminRole").textContent =
`${data.role || "Admin"}`;const avatar = data.userName ?
data.userName.charAt(0).toUpperCase() : "?";
document.getElementById("adminAvatar").textContent = avatar;
sessionStorage.setItem("role", data.role);}
```

```

        console.error("Error loading admin dashboard:", error);
        sessionLoading.style.display = 'none';
        alert("Failed to load dashboard data. Please login again.");
        window.location.href = "adminSignin.html";
        logoutBtn.addEventListener('click', function () {
            if (!userId) {
                alert("No admin ID found. Already logged out.");
                window.location.href = "adminSignin.html";
                return;
            }
            fetch('http://localhost:8080/api/admin/logout', {
                method: 'POST',
                headers: {
                    'Content-Type': 'application/json'
                },
                body: JSON.stringify({
                    userId: userId
                })
            }).then(response => {
                if (!response.ok) throw new Error('Logout failed');
                sessionStorage.clear();
                window.location.href = "adminSignin.html";
            }).catch(error => {
                console.error('Logout error:', error);
                alert('Logout failed. Try again.');
```

```

        return (User.userId.toLowerCase().includes(query)
        || User.userName.toLowerCase().includes(query)) || !
        user.role.toLowerCase().includes(query));
    if (sortBy &&
        filteredUsers.length > 0) {
        filteredUsers.sort((a, b) => {
            const aVal =
            (a[sortBy] || "").toLowerCase();
            const bVal = (b[sortBy] || "").toLowerCase();
            return aVal.localeCompare(bVal);
        });
        displayAllUsers(filteredUsers);
    }
    document.addEventListener("DOMContentLoaded", () => {
        loadUsers();
        document.getElementById("allUserSearchInput").addEventListener("input",
        filterAndSortAllUsers);
        document.getElementById("allUserSortSelect").addEventListener("change",
        filterAndSortAllUsers);
    });
    let recentUsers = [];
    async function
    loadRecent20Users() {
        const loadingDiv =
        document.getElementById("recentUsersLoading");
        const table =
        document.getElementById("recentUsersTable");
        const tbody =
        document.getElementById("recentUsersTableBody");
        loadingDiv.style.display =
        "block";
        table.style.display = "none";
        try {
            const response = await
            fetch("http://localhost:8080/api/users/recent");
            if (!response.ok) throw new
            Error("Failed to fetch users");
            recentUsers = await response.json();
            // make
            sure this gets filled
            displayUsers(recentUsers);
        } catch (error) {
            console.error("Error loading users:", error);
            loadingDiv.innerHTML = "Error
            loading user data.";
        }
        function displayUsers(users) {
            const tbody =
            document.getElementById("recentUsersTableBody");
            const loadingDiv =
            document.getElementById("recentUsersLoading");
            const table =
            document.getElementById("recentUsersTable");
            tbody.innerHTML = "";
            if
            (users.length === 0) {
                tbody.innerHTML = `<tr><td colspan="4">No matching
                users found.</td></tr>`;
            } else {
                users.forEach(user => {
                    const row =
                    document.createElement("tr");
                    row.innerHTML = `<td>${user.userId}</td>
                    <td>${user.userName}</td>
                    <td>${user.role}</td>
                    <td>${formatDate(user.createdAt)}</td>`;
                    tbody.appendChild(row);
                });
            }
            loadingDiv.style.display = "none";
            table.style.display = "table";
        }
        function
        formatDate(dateString) {
            const date = new Date(dateString);
            return
            date.toLocaleString();
        }
        function filterAndSortUsers() {
            const query =
            document.getElementById("searchInput").value.toLowerCase();
            const sortBy =
            document.getElementById("sortSelect").value;
            let filteredUsers =
            recentUsers.filter(user => {
                return (User.userId.toLowerCase().includes(query)
                || User.userName.toLowerCase().includes(query)) || !
                user.role.toLowerCase().includes(query);
            });
            if (sortBy &&
                filteredUsers.length > 0) {
                filteredUsers.sort((a, b) => {
                    const aVal =
                    (a[sortBy] || "").toLowerCase();
                    const bVal = (b[sortBy] || "").toLowerCase();
                    return aVal.localeCompare(bVal);
                });
                displayUsers(filteredUsers);
            }
            document.addEventListener("DOMContentLoaded", () => {
                loadRecent20Users();
                document.getElementById("searchInput").addEventListener("input",
                filterAndSortUsers);
                document.getElementById("sortSelect").addEventListener("change",
                filterAndSortUsers);
            });
            async function downloadUserReport() {
                const { jsPDF } =
                window.jspdf;
                const doc = new jsPDF();
                const title = "User Management Report";
                const date = new Date().toLocaleString();
                // Add optional logo (uncomment and
                use base64 image if needed)
                // doc.addImage("data:image/png;base64,...",
                "PNG", 10, 10, 30, 30);
                // Add Title
                doc.setFontSize(18);
                doc.text(title, 14,
                20);
                // Add Date
                doc.setFontSize(10);
                doc.setTextColor(100);
                doc.text("Generated
                on: " + date, 14, 27);
                // Grab the user table
                const table =
                document.getElementById("userTable");
                const headers =
                Array.from(table.querySelectorAll("thead th")).map(th => th.innerHTML);
                const
                rows = Array.from(table.querySelectorAll("tbody tr")).map(tr =>
                Array.from(tr.querySelectorAll("td")).map(td => td.innerHTML));
                // Create the
                PDF table
                doc.autoTable({
                    head: [headers],
                    body: rows,
                    startY: 35,
                    theme: "grid",
                    styles: {
                        fontSize: 10,
                        cellPadding: 3,
                        headStyles: {
                            fillColor: [60, 141, 188],
                            textColor: 255,
                            fontStyle: 'bold',
                        },
                        alternateRowStyles: {
                            fillColor: [245, 245,
                            245]
                        }
                    }
                });
                // Download file
                const fileName = "User_Report_" + new
                Date().toISOString().slice(0, 10) + ".pdf";
            }
        }
    }

```

```

doc.save(fileName);}); async function
downloadAllUsersReport() {const { jsPDF } = window.jspdf;const doc = new
jsPDF();const title = "All Users Report";const date = new
Date().toLocaleString();// Optional: Add your logo (base64 format image)//
doc.addImage("data:image/png;base64,...", "PNG", 10, 10, 30, 30);// Titledoc
doc.setFontSize(18);doc.text(title, 14, 20);// Datedoc.setFontSize(10);doc
doc.setTextColor(100);doc.text("Generated on: " + date, 14, 27);// Fetch
latest all users datatry {const response = await fetch("http://localhost:8080/
api/all-users");if (!response.ok) throw new Error("Failed to fetch users");const
users = await response.json();if (!users.length) {alert("No users found
to generate the report.");return;// Format headers and rowsconst headers =
[["User ID", "User Name", "Email", "Phone", "Role"]];const rows =
users.map(user => [User.userId,User.userName,User.email,User.phone,User.roledoc
]);// Create the table in the PDFdoc.autoTable({head: headers,body: rows,startY: 35,theme: "grid",styles: {fontSize: 10,cellPadding: 3,headStyles: {fillColor: [60, 141, 188],textColor: 255,fontStyle: 'bold',alternateRowStyles: {fillColor: [245, 245, 245]}doc
});// Save PDFconst fileName
= "All_Users_Report_" + new Date().toISOString().slice(0, 10) + ".pdf";doc
doc.save(fileName);catch (error) {console.error("Error generating PDF:",
error);alert("Failed to generate report.");// Function loadUserProfile() {const
userId = sessionStorage.getItem("userId");if (!userId) {console
.error("No userId found in sessionStorage");return;fetch(`http://
localhost:8080/api/profile/${userId}`)then(response => {if (!response.ok) {throw
new Error("Failed to fetch user profile");return response.json();doc
}.then(data => {document.getElementById("profileUserId").textContent =
data.userId || "N/A";document.getElementById("profileUserName").textContent =
data.userName || "N/A";document.getElementById("profileEmail").textContent =
data.email || "N/A";document.getElementById("profilePhone").textContent =
data.phone || "N/A";document.getElementById("profileRole").textContent =
data.role || "N/A";document.getElementById("profileReferralCode").textContent
= data.referralCode || "N/A";document.getElementById("profileCreatedAt").textContent = data.createdAt ||
"N/A";const pic = localStorage.getItem("adminProfileImage");if (pic) {document
.getElementById("profileImage").src = pic;document.getElementById("settingsImage").src = pic;document
.getElementById("profileImage").classList.add("visible");document
.getElementById("settingsImage").classList.add("visible");doc
}.catch(error => {console.error("Error loading profile:", error);// Load
data on page loadfunction initializeSettingsModule() {const userId =
sessionStorage.getItem("userId");if (!userId) {console.error("No
userId found in sessionStorage");showStatus("No user session found",
true);return;doc
}.fetch(`http://localhost:8080/api/profile/
${userId}`)then(res => res.json())then(data => {document
.getElementById("currentUsername").textContent = data.userName || "N/
A";document.getElementById("currentPhone").textContent =
data.phone || "N/A";doc
}.catch(() => showStatus("Failed to load
profile", true));// Edit profile (triggers both username and phone edit)function
editProfile() {editUsername();editPhone();document
.getElementById("editProfileBtn").style.display = "none";// Edit
usernamefunction editUsername() {document
.getElementById("usernameEdit").classList.add("active");document
.getElementById("usernameInput").value =
document.getElementById("currentUsername").textContent;document
.getElementById("currentUsername").style.display = "none";// Edit
phonefunction editPhone() {document
.getElementById("phoneEdit").classList.add("active");document
.getElementById("phoneInput").value =
document.getElementById("currentPhone").textContent;document
.getElementById("currentPhone").style.display = "none";// Cancel editfunction
cancelEdit(field) {doc

```



```

        if (field === 'username') {
document.getElementById("usernameEdit").classList.remove("active");
document.getElementById("currentUsername").style.display = "inline-block";
        } else if (field === 'phone') {
document.getElementById("phoneEdit").classList.remove("active");
document.getElementById("currentPhone").style.display = "inline-block";
        }
        // Show Edit button only if both edit sections are inactive
        if (!document.getElementById("usernameEdit").classList.contains("active") &&
            !document.getElementById("phoneEdit").classList.contains("active")) {
document.getElementById("editProfileBtn").style.display = "inline-block";
        }
        // Save username with validation
function saveUsername() {
const updatedName = document.getElementById("usernameInput").value.trim();
const userId = sessionStorage.getItem("userId");
const usernameRegex = /^(?!\\s)(?!.*\\s){2,}[a-zA-Z0-9 ]+(?!\\s)$/;
if (!updatedName) {
showStatus("Username cannot be empty", true);
return;
}
if (!usernameRegex.test(updatedName)) {
showStatus("Username must contain only letters, numbers, and single spaces, and not start/end with space", true);
return;
}
fetch("http://localhost:8080/api/admin/update-username", {
method: "POST",
headers: { "Content-Type": "application/json" },
body: JSON.stringify({ userId, username: updatedName })
})
.then(res => {
if (!res.ok) throw new Error("Failed to update");
return res.json();
})
.then(data => {
document.getElementById("currentUsername").textContent = data.userName;

document.getElementById("usernameEdit").classList.remove("active");
document.getElementById("currentUsername").style.display = "inline-block";
showStatus("Username updated successfully.");
        // Show Edit button if both edit sections are inactive
        if (!document.getElementById("phoneEdit").classList.contains("active")) {
document.getElementById("editProfileBtn").style.display = "inline-block";
        }
    })
    .catch(() => showStatus("Failed to update username", true));
}
        // Save phone with validation
function savePhone() {
const updatedPhone = document.getElementById("phoneInput").value.trim();
const userId = sessionStorage.getItem("userId");
const phoneRegex = /^[6-9]\\d{9}$/;
if (!updatedPhone) {
showStatus("Phone number cannot be empty", true);
return;
}
if (!phoneRegex.test(updatedPhone)) {
showStatus("Phone number must start with 6-9 and be 10 digits", true);
return;
}
fetch("http://localhost:8080/api/admin/update-phone", {
method: "POST",
headers: { "Content-Type": "application/json" },
body: JSON.stringify({ userId, phone: updatedPhone })
})
.then(res => {
if (!res.ok) throw new Error("Failed to update");
return res.json();
})
.then(data => {
document.getElementById("currentPhone").textContent = data.phone;
document.getElementById("phoneEdit").classList.remove("active");
document.getElementById("currentPhone").style.display = "inline-block";
showStatus("Phone number updated successfully.");
        // Show Edit button if both edit sections are inactive
        if (!document.getElementById("usernameEdit").classList.contains("active")) {
document.getElementById("editProfileBtn").style.display = "inline-block";
        }
    })
    .catch(() => showStatus("Failed to update phone number", true));
}
        // Show status
function showStatus(message, isError = false) {
const el = document.getElementById("settingsStatus");
el.textContent = message;
el.style.color = isError ? "red" : "green";
el.classList.add("visible");
setTimeout(() => {
el.classList.remove("visible");
el.textContent = "";
}, 3000);
}
        // Show success popup
function showSuccessPopup(message) {
const popup = document.getElementById("successPopup");
const messageEl = document.getElementById("successMessage");

```

```

        messageEl.textContent = message;
        popup.style.display = "block";
        setTimeout(() => {
            closePopup();
        }, 3000); // Auto-close after 3 seconds
    }
    // Close popup function
    closePopup() {
        const popup = document.getElementById("successPopup");
        popup.style.display = "none";
    }
    // Show status for failures
    showStatus(message, isError = false) {
        const el = document.getElementById("settingsStatus");
        el.textContent = message;
        el.style.color = isError ? "red" : "green";
        el.classList.add("visible");
        setTimeout(() => {
            el.classList.remove("visible");
        }, 3000);
    }
    // Profile picture
    const profilePic = document.getElementById("profilePic");
    const settingsPic = document.getElementById("settingsPic");
    const uploadInput = document.getElementById("uploadPicInput");
    const pic = localStorage.getItem("adminProfileImage");
    if (pic) {
        document.getElementById("profileImage").src = pic;
        document.getElementById("settingsImage").src = pic;
        document.getElementById("profileImage").classList.add("visible");
        document.getElementById("settingsImage").classList.add("visible");
    }
    // Profile picture
    document.getElementById("uploadInput").addEventListener("change", function (e) {
        const file = e.target.files[0];
        if (file && file.type.startsWith("image/")) {
            const reader = new FileReader();
            reader.onload = function (event) {
                const base64Image = event.target.result;
                localStorage.setItem("adminProfileImage", base64Image);
                document.getElementById("profileImage").src = base64Image;
                document.getElementById("settingsImage").src = base64Image;
                document.getElementById("profileImage").classList.add("visible");
                document.getElementById("settingsImage").classList.add("visible");
                showStatus("Profile picture updated successfully.");
            };
            reader.readAsDataURL(file);
        }
    });
    // Run on page load
    initializeSettingsModule();
    let currentAdminId = 'ADMIN_0001'; // Replace with actual admin ID
    let isAdmin = true; // true for admin
    let receiverId = 'USER_0001'; // Replace with the logged-in user's ID
    // Fetch chat messages from backend
    function loadChatMessages() {
        fetch(`http://localhost:8080/chat/conversation?userId=${receiverId}&adminId=${currentAdminId}`)
            .then(res => res.json())
            .then(data => {
                const chatBox = document.getElementById('chatBox');
                chatBox.innerHTML = ''; // Clear previous messages
                data.forEach(msg => {
                    const div = document.createElement('div');
                    div.classList.add('chat-message');
                    if (msg.senderId === currentAdminId) {
                        div.classList.add('admin-message');
                    } else {
                        div.classList.add('user-message');
                    }
                    // Format timestamp to HH:MM AM/PM
                    const timestamp = new Date(msg.timestamp);
                    const timeString = timestamp.toLocaleTimeString('en-US', {
                        hour12: true,
                        timeZone: 'Asia/Kolkata' // Change to your preferred time zone
                    });
                    div.innerHTML = `<span>${msg.message}</span><br><small class="timestamp">${timeString}</small>`;
                    chatBox.appendChild(div);
                });
                chatBox.scrollTop = chatBox.scrollHeight; // Scroll to bottom
            })
            .catch(err => {
                console.error('Error loading chat messages:', err);
            });
    }
    // Send chat message
    function sendChatMessage() {
        const input = document.getElementById('chatInput');
        const message = input.value.trim();
        if (!message) return;
        const payload = {
            senderId: currentAdminId,
            receiverId: receiverId,
            message: message,
            timestamp: new Date().toISOString()
        };
        fetch('http://localhost:8080/chat/send', {
            method: 'POST',
            headers: { 'Content-Type': 'application/json' },
            body: JSON.stringify(payload)
        })
            .then(res => {
                if (res.ok) {
                    input.value = ''; // Clear input field
                    loadChatMessages(); // Refresh chat
                }
            })
            .catch(err => {
                console.error('Error sending message:', err);
            });
    }
    // Auto-refresh chat every 5 seconds
    setInterval(loadChatMessages, 10000);
    window.onload = loadChatMessages;

```

demo\target\classes\static\user_dashboard3.js

```

    const menuToggle = document.getElementById('menuToggle');
    const sidebar = document.getElementById('sidebar');
    const mainContent = document.getElementById('mainContent');
    const logoutBtn = document.getElementById('logout');
    const sessionLoading = document.getElementById('sessionLoading');
    const sessionTable = document.getElementById('sessionTable');
    const userId = sessionStorage.getItem("userId");
    function showModule(moduleId) {
        const allModules = document.querySelectorAll('.module');
        allModules.forEach(mod => {
            mod.style.display = 'none'; // hide all
        });
        const selectedModule = document.getElementById(moduleId);
        if (selectedModule) {
            selectedModule.style.display = 'block'; // show the selected one
        }
        if (moduleId === 'userProfileModule') {
            loadUserProfile();
        }
        if (moduleId === 'settingsModule') {
            initializeSettingsModule(); // + p This will run only when you open settings
        }
        if (moduleId === 'chatModule') {
            loadChatMessages();
        }
        // Toggle sidebar
        menuToggle.addEventListener('click', () => {
            sidebar.classList.toggle('active');
            mainContent.classList.toggle('expanded');
        });
        // Format datetime for display
        function formatDateTime(dateTimeStr) {
            if (!dateTimeStr) return 'N/A';
            const date = new Date(dateTimeStr);
            return new Intl.DateTimeFormat('en-US', {
                year: 'numeric',
                month: 'short',
                day: 'numeric',
                hour: '2-digit',
                minute: '2-digit',
                second: '2-digit'
            }).format(date);
        }
        // Load dashboard data
        function loadDashboardData() {
            if (!userId) {
                alert("User ID not found in session. Please login again.");
                return;
            }
            sessionLoading.style.display = 'flex';
            sessionTable.style.display = 'none';
            fetch(`http://localhost:8080/api/user_sessions/${userId}`)
                .then(response => {
                    if (!response.ok) throw new Error('Failed to fetch session');
                    return response.json();
                })
                .then(data => {
                    sessionLoading.style.display = 'none';
                    sessionTable.style.display = 'table'; // Session data
                    document.getElementById("userIdDisplay").textContent = data.userId || "N/A";
                    document.getElementById("userNameDisplay").textContent = data.userName || "N/A";
                    document.getElementById("signinTimeDisplay").textContent = formatDateTime(data.signinTime);
                    document.getElementById("logoutTimeDisplay").textContent = data.logoutTime
                        ? formatDateTime(data.logoutTime)
                        : "Active";
                    document.getElementById("roleDisplay").textContent = data.role || "USER";
                    // Status
                    const statusDisplay = document.getElementById("statusDisplay");
                    sidebarUserStatus = document.getElementById("userStatus");
                    if (data.isActive) {
                        statusDisplay.innerHTML = '<span style="color: green;">Online</span>';
                        sidebarUserStatus.textContent = "Online";
                        sidebarUserStatus.style.color = "green";
                    } else {
                        statusDisplay.innerHTML = '<span style="color: red;">Offline</span>';
                        sidebarUserStatus.textContent = "Offline";
                        sidebarUserStatus.style.color = "red";
                    }
                });
            // User Info
            document.getElementById("sidebarUserName").textContent = data.userName ||

```

```

"Unknown";
document.getElementById("headerUserName").textContent = data.userName ||
"Unknown";
document.getElementById("sidebarUserRole").textContent = data.role || "User";
document.getElementById("headerUserRole").textContent =
`${data.role || "User"}`;
const avatar =
data.userName ? data.userName.charAt(0).toUpperCase() : "?";
document.getElementById("userAvatar").textContent =
avatar;
sessionStorage.setItem("role", data.role);
})
.catch(error => {
console.error("Error loading dashboard data:", error);
sessionLoading.style.display = 'none';
loadDashboardData().then(() => {
loadDashboardData().Please login again.");
window.location.href = "signin.html";
});
// Handle logout
logoutBtn.addEventListener('click', function () {
if (!userId) {
alert("No user ID found.
Already logged out.");
window.location.href = "signin.html";
return;
}
fetch('http://localhost:8080/api/logout', {
method: 'POST',
headers: {
'Content-Type': 'application/json',
body: JSON.stringify({ userId: userId })
})
.then(response => {
if (!response.ok) throw new Error('Logout failed');
sessionStorage.clear();
window.location.href = "signin.html";
})
.catch(error => {
console.error('Logout error:', error);
alert('Logout failed. Try again.');
});
// Init dashboard
document.addEventListener('DOMContentLoaded', function () {
loadDashboardData();
setInterval(loadDashboardData, 30000); // Refresh every 30 seconds
function loadUserProfile() {
const userId = sessionStorage.getItem("userId");
if (!userId) {
console.error("No userId found in sessionStorage");
return;
}
fetch(`http://localhost:8080/api/profile/${userId}`)
.then(response => {
if (!response.ok) {
throw new Error("Failed to fetch user profile");
}
return response.json();
})
.then(data => {
document.getElementById("profileUserId").textContent = data.userId || "N/A";
document.getElementById("profileUserName").textContent = data.userName || "N/A";
document.getElementById("profileEmail").textContent = data.email || "N/A";
document.getElementById("profilePhone").textContent = data.phone || "N/A";
document.getElementById("profileRole").textContent = data.role || "N/A";
document.getElementById("profileReferralCode").textContent = data.referralCode || "N/A";
document.getElementById("profileCreatedAt").textContent = data.createdAt || "N/A";
const pic = localStorage.getItem("userProfileImage");
if (pic) {
document.getElementById("profileImage").src = pic;
document.getElementById("settingsImage").src = pic;
document.getElementById("profileImage").classList.add("visible");
document.getElementById("settingsImage").classList.add("visible");
}
})
.catch(error => {
console.error("Error loading profile:", error);
});
// Load data on page load
function initializeSettingsModule() {
const userId = sessionStorage.getItem("userId");
if (!userId) {
console.error("No user ID found in sessionStorage");
showStatus("No user session found", true);
return;
}
fetch(`http://localhost:8080/api/profile/${userId}`)
.then(res => res.json())
.then(data => {
document.getElementById("currentUsername").textContent = data.userName || "N/A";

```

```

document.getElementById("currentPhone").textContent = data.phone || "N/A";
    })
    .catch(() => showStatus("Failed to load
profile", true));
    }
    // Edit profile (triggers both username and
phone edit)
    function editProfile() {
        editUsername();
        editPhone();
    }
document.getElementById("editProfileBtn").style.display = "none";
    // Edit username
    function editUsername() {
document.getElementById("usernameEdit").classList.add("active");
document.getElementById("usernameInput").value =
document.getElementById("currentUsername").textContent;
document.getElementById("currentUsername").style.display = "none";
    // Edit phone
    function editPhone() {
document.getElementById("phoneEdit").classList.add("active");
document.getElementById("phoneInput").value =
document.getElementById("currentPhone").textContent;
document.getElementById("currentPhone").style.display = "none";
    // Cancel edit
    function cancelEdit(field) {
        if
        (field === 'username') {
document.getElementById("usernameEdit").classList.remove("active");
            document.getElementById("currentUsername").style.display =
"inline-block";
        } else if (field === 'phone') {
document.getElementById("phoneEdit").classList.remove("active");
            document.getElementById("currentPhone").style.display =
"inline-block";
        }
        // Show Edit button only if both edit
sections are inactive
        if (!
document.getElementById("usernameEdit").classList.contains("active") &&
!
document.getElementById("phoneEdit").classList.contains("active")) {
            document.getElementById("editProfileBtn").style.display =
"inline-block";
        }
        // Save username with validation
        function saveUsername() {
            const updatedName =
document.getElementById("usernameInput").value.trim();
            const
userId = sessionStorage.getItem("userId");
            const usernameRegex = /
^(?!\\s)(?!.*\\s){2,}[a-zA-Z0-9 ]+(?!\\s)$/;
            if (!updatedName) {
                showStatus("Username cannot be empty", true);
            }
            if (!usernameRegex.test(updatedName)) {
                showStatus("Username must contain only letters, numbers, and
single spaces, and not start/end with space", true);
            }
            return;
        }
        fetch("http://localhost:8080/api/update-username", {
            method: "POST",
            headers: { "Content-Type":
"application/json" },
            body: JSON.stringify({ userId, userName:
updatedName })
        })
        .then(res => {
            if (!res.ok) throw new Error("Failed to update");
            return res.json();
        })
        .then(data => {
document.getElementById("currentUsername").textContent = data.userName;

document.getElementById("usernameEdit").classList.remove("active");
            document.getElementById("currentUsername").style.display
= "inline-block";
            showStatus("Username updated
successfully.");
            // Show Edit button if both edit sections
are inactive
            if (!
document.getElementById("phoneEdit").classList.contains("active")) {
document.getElementById("editProfileBtn").style.display = "inline-block";
            }
        })
        .catch(() =>
showStatus("Failed to update username", true));
        // Save phone
with validation
        function savePhone() {
            const updatedPhone =
document.getElementById("phoneInput").value.trim();

```

```

const userId =
sessionStorage.getItem("userId");
const phoneRegex = /
^[6-9]\d{9}$/;
if (!updatedPhone) {
showStatus("Phone number cannot be empty", true);
return;
}
if (!phoneRegex.test(updatedPhone)) {
showStatus("Phone number must start with 6-9 and be 10 digits", true);
return;
}
fetch("http://localhost:8080/api/update-phone", {
method: "POST",
headers: {
"Content-Type": "application/json"
},
body:
JSON.stringify({ userId, phone: updatedPhone })
})
.then(res => {
if (!res.ok) throw new
Error("Failed to update");
return res.json();
})
.then(data => {
document.getElementById("currentPhone").textContent = data.phone;

document.getElementById("phoneEdit").classList.remove("active");
document.getElementById("currentPhone").style.display =
"inline-block";
showStatus("Phone number updated
successfully.");
// Show Edit button if both edit sections
are inactive
if (!
document.getElementById("usernameEdit").classList.contains("active")) {

document.getElementById("editProfileBtn").style.display = "inline-block";
}
})
.catch(() =>
showStatus("Failed to update phone number", true));
}
// Show status
function showStatus(message, isError = false) {
const el = document.getElementById("settingsStatus");
el.textContent = message;
el.style.color = isError ? "red" :
"green";
el.classList.add("visible");
setTimeout(() => {
el.classList.remove("visible");
el.textContent = "";
}, 3000);
}
// Show success popup
function showSuccessPopup(message) {
const popup =
document.getElementById("successPopup");
const messageEl =
document.getElementById("successMessage");
messageEl.textContent =
message;
popup.style.display = "block";
setTimeout(()
=> {
closePopup();
}, 3000);
// Auto-close after 3
seconds
}
function closePopup() {
const popup = document.getElementById("successPopup");
popup.style.display = "none";
}
// Show status for failures
function showStatus(message, isError = false) {
const el =
document.getElementById("settingsStatus");
el.textContent =
message;
el.style.color = isError ? "red" : "green";
el.classList.add("visible");
setTimeout(() => {
el.classList.remove("visible");
el.textContent = "";
}, 3000);
}
const profilePic =
document.getElementById("profilePic");
const settingsPic =
document.getElementById("settingsPic");
const uploadInput =
document.getElementById("uploadPicInput");
const pic =
localStorage.getItem("userProfileImage");
if (pic) {
document.getElementById("profileImage").src = pic;
document.getElementById("settingsImage").src = pic;
document.getElementById("profileImage").classList.add("visible");
document.getElementById("settingsImage").classList.add("visible");
}
// Profile picture
document.getElementById("uploadInput").addEventListener("change", function
(e) {
const file = e.target.files[0];
if (file &&
file.type.startsWith("image/")) {
const reader = new FileReader();
reader.onload = function (event) {
const base64Image =
event.target.result;
localStorage.setItem("userProfileImage",
base64Image);
}
}
}

```

```

        document.getElementById("profileImage").src =
base64Image;
        document.getElementById("settingsImage").src =
base64Image;
document.getElementById("profileImage").classList.add("visible");
document.getElementById("settingsImage").classList.add("visible");
showStatus("Profile picture updated successfully.");
reader.readAsDataURL(file);
initializeSettingsModule();
function showCustomNotification() {
const notificationArea =
document.getElementById("notificationArea");
notificationArea.style.display
= "block"; // Show the notification
// Event listener to hide the
notification when clicked
notificationArea.addEventListener('click', () => {
notificationArea.style.display = "none"; // Hide on click
// Hide
the notification after 3 seconds if not clicked
setTimeout(() => {
if
(notificationArea.style.display !== "none") {
notificationArea.style.display = "none";
}
}, 3000);
const currentUserId =
'USER_0001'; // Replace with actual logged-in ID
const isAdmin = false; // true
if admin, false if user
const receiverId = 'ADMIN_0001'; // Replace with actual
receiver (admin ID)
// Fetch chat messages from backend
function
loadChatMessages() {
fetch('http://localhost:8080/chat/conversation?
userId=${currentUserId}&adminId=ADMIN_0001') // Pass userId and adminId
.then(res => res.json())
.then(data => {
const chatBox =
document.getElementById('chatBox');
chatBox.innerHTML = ''; // Clear
previous messages
data.forEach(msg => {
const div =
document.createElement('div');
div.classList.add('chat-message');
// Add different class based on sender
if (msg.senderId ===
currentUserId) {
div.classList.add('user-message');
} else {
div.classList.add('admin-message');
}
// Format
timestamp to time only
const timeOnly = new
Date(msg.timestamp).toLocaleTimeString('en-IN', {
hour: '2-digit',
minute: '2-digit',
hour12: true,
timeZone: 'Asia/
Kolkata'
});
// Combine message and time
div.innerHTML =
`<span>${msg.message}</span>
<br>
<small
class="timestamp">${timeOnly}</small>`;
chatBox.appendChild(div);
});
chatBox.scrollTop =
chatBox.scrollHeight; // Scroll to the bottom of the chat
}).catch(err
=> {
console.error('Error loading chat messages:', err);
});
// Send
chat message
function sendChatMessage() {
const input =
document.getElementById('chatInput');
const message = input.value.trim();
if (!message) return;
const payload = {
senderId: currentUserId,
receiverId: receiverId,
message: message,
timestamp: new
Date().toISOString() // Add timestamp
};
fetch('http://localhost:8080/chat/
send', {
method: 'POST',
headers: { 'Content-Type': 'application/
json' },
body: JSON.stringify(payload)
})
.then(res => {
if
(res.ok) {
input.value = ''; // Clear the input field
loadChatMessages(); // Reload chat after sending
}
}).catch(err
=> {
console.error('Error sending message:', err);
});
// Auto-
refresh chat every 5 seconds
setInterval(loadChatMessages, 10000);
window.onload = loadChatMessages; // Load chat on page load

```

```
<title> Admin Signin Page</title><div class="text-  
overlay"></div><div class="image-section"></div>  
</div><div class="form-section"><div class="form-box">  
<h2>Welcome Back!</h2><p>Login to Continue</p><form  
id="adminSigninForm"><div class="input-group">  
    <label for="adminid">Admin Username </label>  
    <input type="text" id="adminid" name="adminid"  
placeholder="Enter Admin Username or Email" required >  
<div class="error" id="adminidError"></div>  
    <div class="input-group"><label  
for="adminPassword">Password</label>  
    <input type="password" id="adminPassword" name="adminPassword" placeholder="Enter  
Password" required>  
<div class="error" id="adminPasswordError"></div>  
    <div class="remember-me"><input type="checkbox"  
id="adminRememberMe"><label  
for="adminRememberMe">Remember Me</label>  
<h5><a href="forgot.html" class="forgot">Forgot Password?</a></h5>  
</div><button type="submit" class="btn">Sign In</button>  
<div id="adminResult"></div></form>  
</div></div><script>  
document.getElementById("adminSigninForm").addEventListener("submit", async  
function (event) {  
    event.preventDefault(); // Prevent default form  
    submission  
    let adminEnteredValue =  
document.getElementById("adminid").value.trim();  
let  
adminEnteredPassword = document.getElementById("adminPassword").value.trim();  
let adminResultDiv = document.getElementById("adminResult");  
// Clear previous errors  
document.getElementById("adminidError").textContent = "";  
document.getElementById("adminPasswordError").textContent = "";  
adminResultDiv.textContent = "";  
if  
(adminEnteredValue === '' || adminEnteredPassword === '') {  
    adminResultDiv.textContent = '"L Please fill in all fields.";  
    adminResultDiv.style.color = "red";  
return;  
}  
try {  
    let response = await fetch("http://localhost:8080/api/admin/  
signin", {  
        method: "POST",  
        headers: {  
            "Content-Type": "application/json"  
        },  
        body: JSON.stringify({  
            identifier:  
adminEnteredValue, // ' Use "identifier" instead of "adminid"  
password: adminEnteredPassword  
})  
});  
let result = await response.json();  
if (response.ok) { // ' Check response status instead of message  
    adminResultDiv.textContent = '" Login successful!  
Redirecting...";  
    adminResultDiv.style.color = "green";  
    // ' Store in sessionStorage  
sessionStorage.setItem("userId", result.userId);  
sessionStorage.setItem("userName", result.userName);  
sessionStorage.setItem("role", result.role);  
// Store  
admin details in localStorage  
localStorage.setItem("userId",  
result.userId);  
localStorage.setItem("userName",  
result.userName);  
localStorage.setItem("role",  
result.role);  
// ' Store admin role  
// Redirect  
after a short delay  
setTimeout(() => {  
window.location.href = "admin_dashboard.html";  
, 1500);  
} else {  
    adminResultDiv.textContent = '"L ' +  
result.message;<br>  
    adminResultDiv.style.color = "red";  
} catch (error) {  
    console.error("Error:",  
error);  
    adminResultDiv.textContent = '"L Server error. Please try  
again later.";
```



```

adminResultDiv.style.color = "red";
</script>
style {
  * {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
    font-family: "Poppins", sans-serif;
    display: flex;
    justify-content: center;
    align-items: center;
    height: 100vh;
    background: white;
    .container {
      display: flex;
      width: 100%;
      background: black;
      max-width: 1700px;
      height: 700px;
      box-shadow: 0px 4px 10px rgba(0, 0, 0, 0.3);
      border-radius: 15px;
      overflow: hidden;
      background-color: #f0f5ff;
      background-image: radial-gradient(circle at 100% 150%, #f0f5ff 24%, #c8d7ff 25%, #c8d7ff 28%, #f0f5ff 29%, #f0f5ff 36%, #c8d7ff 36%, #c8d7ff 40%, transparent 40%, transparent), radial-gradient(circle at 0 150%, #f0f5ff 24%, #c8d7ff 25%, #c8d7ff 28%, #f0f5ff 29%, #f0f5ff 36%, #c8d7ff 36%, #c8d7ff 40%, transparent 40%, transparent), radial-gradient(circle at 50% 100%, #c8d7ff 10%, #f0f5ff 11%, #f0f5ff 23%, #c8d7ff 24%, #c8d7ff 30%, #f0f5ff 31%, #f0f5ff 43%, #c8d7ff 44%, #c8d7ff 50%, #f0f5ff 51%, #f0f5ff 63%, #c8d7ff 64%, #c8d7ff 71%, transparent 71%, transparent), radial-gradient(circle at 100% 50%, #c8d7ff 5%, #f0f5ff 6%, #f0f5ff 15%, #c8d7ff 16%, #c8d7ff 20%, #f0f5ff 21%, #f0f5ff 30%, #c8d7ff 31%, #c8d7ff 35%, #f0f5ff 36%, #f0f5ff 45%, #c8d7ff 46%, #c8d7ff 49%, transparent 50%, transparent), radial-gradient(circle at 0 50%, #c8d7ff 5%, #f0f5ff 6%, #f0f5ff 15%, #c8d7ff 16%, #c8d7ff 20%, #f0f5ff 21%, #f0f5ff 30%, #c8d7ff 31%, #c8d7ff 35%, #f0f5ff 36%, #f0f5ff 45%, #c8d7ff 46%, #c8d7ff 49%, transparent 50%, transparent);
      background-size: 100px 50px;
      .image-section {
        width: 600px;
        flex: 1;
        background: url("pics/trading\ with\ coin.gif") no-repeat center / 100% 100%;
        margin-top: 100px;
        margin-bottom: 100px;
        margin-left: 70px;
        border-bottom-left-radius: 5%;
        border-top-left-radius: 5%;
        box-shadow: -5px 0 10px -2px rgba(0, 0, 0, 0.4);
        .form-section {
          width: 50%;
          display: flex;
          align-items: center;
          justify-content: center;
          backdrop-filter: blur(10px);
          margin-top: 100px;
          margin-bottom: 100px;
          margin-right: 60px;
          border-top-right-radius: 5%;
          border-bottom-right-radius: 5%;
          box-shadow: 5px 0 4px -2px rgba(0, 0, 0, 0.3);
          .form-box {
            background: white;
            padding: 80px;
            width: 100%;
            height: 100%;
            text-align: center;
            color: white;
            border-top-right-radius: 5%;
            border-bottom-right-radius: 5%;
            box-shadow: 5px 0 4px -2px rgba(0, 0, 0, 0.3);
            h2 {
              font-family: 'Times New Roman', Times, serif;
              text-align: center;
              font-size: 40px;
              font-weight: bold;
              color: black;
              text-shadow: 0px 4px 10px rgba(0, 0, 0, 0.1);
            }
            p {
              font-family: 'Times New Roman', Times, serif;
              text-align: center;
              font-size: 20px;
              font-weight: bold;
              color: black;
              margin-bottom: 30px;
            }
            h4 {
              font-family: 'Franklin Gothic Medium', 'Arial Narrow', Arial, sans-serif;
              color: black;
              font-size: 19px;
              font-weight: bold;
              padding-bottom: 20px;
            }
            h5 {
              color: black;
              font-weight: bold;
              font-size: 15px;
            }
            .signup {
              text-decoration: none;
            }
            .signup:hover {
              text-decoration: underline;
              transform: scale(1.05);
            }
            .input-group {
              margin-bottom: 15px;
              text-align: left;
              label {
                font-family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida Sans Unicode', Geneva, Verdana, sans-serif;
                padding-bottom: 10px;
                font-weight: bold;
                letter-spacing: 1px;
                display: block;
                font-size: 16px;
                color: black;
            }
            input {
              font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;
              width: 100%;
              padding: 12px 15px;
              border: none;
              border-radius: 6px;
              font-size: 16px;
              border: 2px solid #ccc;
            }
            .input-group input:hover {
              border-color: #667eea;
              box-shadow: 0px 0px 10px rgba(102, 126, 234, 0.5);
            }
          }
        }
      }
    }
  }
}

```

```

transform:
scale(1.02);@ }@ input:focus {@ outline: none;@ transform:
scale(1.05);@ box-shadow: 0px 0px 10px rgba(255, 255, 255, 0.4);@ }@
.error {@ color: red;@ font-size: 14px;@ margin-top:
5px;@ }@ @ .remember-me {@ display: flex;@ @ align-
items: center;@ @ @ @ }@ input[type=checkbox]{@ -
webkit-appearance: checkbox;@ -moz-appearance: checkbox;@
appearance: checkbox;@ display: inline-block;@ width: auto;@ }@
.remember-me input {@ margin-right: 8px;@ cursor: pointer;@
left: 0;@ @ }@ .btn {@ width: 100%;@
background: linear-gradient(to right, #667eea, #764ba2);@ color: white;@
padding: 14px;@ border: none;@ border-radius: 8px;@
font-size: 18px;@ cursor: pointer;@ transition: all 0.3s ease-in-
out;@ font-weight: bold;@ text-transform: uppercase;@
letter-spacing: 1px;@ box-shadow: 0px 5px 15px rgba(102, 126, 234, 0.3);@
}@ @ @ @ .btn:hover {@ background: linear-
gradient(to right, #764ba2, #667eea);@ transform: scale(1.05);@
box-shadow: 0px 8px 20px rgba(102, 126, 234, 0.5);@ }@ #result {@
font-weight: bold;@ margin-top: 15px;@ color: #00ff99;@ }@
.forgot {@ padding-left: 230px;@ text-decoration: none;@ }@
.forgot :hover{@ background: #ff4e88;@ transform:
scale(1.05);@ }@ @ @ @ @media (max-width: 768px) {@ .container {@
flex-direction: column;@ height: auto;@ }@
.image-section {@ height: 200px;@ }@ .form-
section {@ height: auto;@ padding: 20px;@ }@
.form-box {@ width: 90%;@ }@ }@ .text-overlay h2
{position: absolute;@ top: 6%;@ left: 25%;@ transform: translate(-50%, -50%);@ text-
align: center;@ padding: 20px;@ border-radius: 10px;@ }@ .text-overlay h2{@ color:
black;@ font-weight: bold;@ font-size: 24px;@ text-shadow: 0 0 1px #101111, 0 0 1px
#0e0f0f, 0 0 1px #121313;@ }@ .text-overlay p {position: absolute;@ top: 94%;@ left:
25%;@ transform: translate(-50%, -50%);@ text-align: center;@ color: black;@ padding:
20px;@ border-radius: 10px;@ }@ .text-overlay p{@ font-weight: bold;@ font-size: 20px;@
text-shadow: 0 0 1px #0a1013, 0 0 1px #13171d, 0 0 1px #0e1011;@ }@ /style>@ @

```

demo\src\main\resources\static\adminSignup.html

```
<title>Admin Signup Page</title>&#xA0;<div class="container">&#xA0;<div class="text-
overlay">&#xA0;<h2>Start Your Trading Journey with Confidence</h2>&#xA0;
<p>Trade Smart. Grow Fast.</p>&#xA0;</div>&#xA0;<div class="image-section">&#xA0;
&#xA0;</div>&#xA0;<div
class="form-section">&#xA0;<div class="form-box">&#xA0;<h2>Admin Sign up</h2>&#xA0;
<form id="adminSignupForm">&#xA0;<div class="input-group">&#xA0;<label
for="adminName">Admin Name</label>&#xA0;<input type="text" class="input-
style" id="userName" name="adminName" required placeholder="Admin Name">&#xA0;
<div class="error" id="adminNameError"></div>&#xA0;</div>&#xA0;<div
class="input-group">&#xA0;<label for="adminEmail">Email</label>&#xA0;
<input type="email" id="email" class="input-style" name="email" required
placeholder="Email">&#xA0;<div class="error" id="adminEmailError"></div>&#xA0;
</div>&#xA0;<div class="input-group">&#xA0;<label for="adminPhone">Phone
Number</label>&#xA0;<input type="tel" id="phone" class="input-style"
name="phone" required placeholder="Phone Number">&#xA0;<div class="error"
id="adminPhoneError"></div>&#xA0;</div>&#xA0;<div class="input-group">&#xA0;
<label for="tradingId">Trading View ID</label>&#xA0;<input type="text"
id="tradingId" class="input-style" name="tradingId" required
placeholder="Trading View ID">&#xA0;</div>&#xA0;<div class="input-group">&#xA0;
<label for="adminPassword">Password</label>&#xA0;<input type="password"
id="password" class="input-style" name="password" required placeholder="Enter
Password">&#xA0;<div class="error" id="adminPasswordError"></div>&#xA0;</div>&#xA0;
<div class="input-group">&#xA0;<label for="confirmPassword">Confirm
Password</label>&#xA0;<input type="password" id="confirmPassword"
class="input-style" name="confirmPassword" required placeholder="Repeat
Password">&#xA0;<div class="error" id="confirmPasswordError"></div>&#xA0;</
div>&#xA0;<div class="referral-section">&#xA0;<input type="checkbox"
id="referralCheckbox">&#xA0;<label for="referralCheckbox">Have a Referral
ID?</label>&#xA0;</div>&#xA0;<div class="input-group" id="referralId"
style="display: none;">&#xA0;<label for="referral">Referral ID</label>&#xA0;
<input type="text" class="input-style" id="referral" name="referral"
placeholder="Referral ID">&#xA0;</div>&#xA0;<button type="submit"
class="btn">Register Admin</button>&#xA0;</form>&#xA0;<div class="account">&#xA0;
<h4>Already registered? <a class="signin" href="adminSignin.html">Login</a></
h4>&#xA0;</div>&#xA0;</div>&#xA0;</div>&#xA0;</div>&#xA0;</div>&#xA0;</div>&#xA0;</div>
document.getElementById("referralCheckbox").addEventListener("change",
function () {&#xA0; document.getElementById("referralId").style.display =
this.checked ? "block" : "none";&#xA0;});&#xA0;
document.getElementById("adminSignupForm").addEventListener("submit", async
function (event) {&#xA0; event.preventDefault();&#xA0; let userName =
document.getElementById("userName").value.trim();&#xA0; let email =
document.getElementById("email").value.trim();&#xA0; let phone =
document.getElementById("phone").value.trim();&#xA0; let tradingId =
document.getElementById("tradingId").value.trim();&#xA0; let password =
document.getElementById("password").value.trim();&#xA0; let confirmPassword =
document.getElementById("confirmPassword").value.trim();&#xA0; let referralId =
document.getElementById("referral").value.trim() || null;&#xA0; let namePattern
= /^(?!\\s)(?!.*\\s{2,})[a-zA-Z0-9 ]+(?!\\s)$/;&#xA0; let emailPattern = /^[a-zA-
Z0-9._%+-]+@[a-zA-Z0-9.-]+\\.[a-zA-Z]{2,4}$/;&#xA0; let phonePattern = /
^[6-9]\\d{9}$/;&#xA0; let passwordPattern = /^(?=.*[a-z])(?=.*[A-Z])(?=.*\\d)(?
=.*[@$!%*?&])[A-Za-z\\d@$!%*?&]{8,}$/;&#xA0; let isValid = true;&#xA0; // Clear
previous errors&#xA0; document.getElementById("adminNameError").textContent = "&#xA0;";&#xA0;
document.getElementById("adminEmailError").textContent = "&#xA0;";&#xA0;
document.getElementById("adminPhoneError").textContent = "&#xA0;";&#xA0;
document.getElementById("adminPasswordError").textContent = "&#xA0;";&#xA0;
document.getElementById("confirmPasswordError").textContent = "&#xA0;";&#xA0; if (!
namePattern.test(userName)) {&#xA0;
document.getElementById("adminNameError").textContent = "Invalid name!";&#xA0;
isValid = false;&#xA0;

```

```

        }ð if (!emailPattern.test(email)) {ð
document.getElementById("adminEmailError").textContent = "Invalid Email
Format!";ð isValid = false;ð }ð if (!phonePattern.test(phone)) {ð
    document.getElementById("adminPhoneError").textContent = "Invalid
    Phone Number!";ð isValid = false;ð }ð if (!
passwordPattern.test(password)) {ð
document.getElementById("adminPasswordError").textContent =ð
"Password must have at least 8 characters, 1 special char, 1 uppercase, 1
lowercase, and 1 number!";ð isValid = false;ð }ð if (password !==
confirmPassword) {ð
document.getElementById("confirmPasswordError").textContent = "Passwords do
not match!";ð isValid = false;ð }ð if (!isValid) return;ð let
requestData = {ð username: userName,ð email: email,ð phone:
phone,ð tradingId: tradingId,ð password: password,ð
referralId: referralIdð };ð try {ð let response = await
fetch("http://localhost:8080/api/admin/signup", {ð method: "POST",ð
headers: { "Content-Type": "application/json" },ð body:
JSON.stringify(requestData)ð });ð let data;ð try {ð
    data = await response.json();ð } catch (jsonError) {ð
        console.error("Invalid JSON response", jsonError);ð
throw new Error("Invalid server response");ð }ð if (response.ok) {ð
    alert(`Admin Registered Successfully! \n\nAdmin Name:
    ${data.userName} \nAdmin ID: ${data.userId} \nReferral Code:
    ${data.referralCode}`);ð window.location.href = "adminSignin.html";ð
    } else {ð
document.getElementById("adminEmailError").textContent = data.message ||
"Registration failed!";ð }ð } catch (error) {ð
console.error("Error:", error);ð alert("An error occurred while
registering. Please try again.");ð }ð);ð</script>ðstyle>ð * {ð
margin: 0;ð padding: 0;ð box-sizing: border-box;ð
    font-family: "Poppins", sans-serif;ð }ð body {ð
    height: 150vh;ð display: flex;ð justify-
content: center;ð align-items: center;ð background: #ffff;ð
    }ð ð .container {ð display: flex;ð
width: 100%;ð max-width: 1500px;ð height: 950px;ð
    box-shadow: 0px 10px 30px rgba(0, 0, 0, 0.1);ð border-
radius: 15px;ð overflow: hidden;ð background-color:
#f5f5f5;ð background-image: linear-gradient(30deg, #ecec 12%,
transparent 12.5%, transparent 87%, #ecec 87.5%, #ecec),ð
linear-gradient(150deg, #ecec 12%, transparent 12.5%, transparent 87%,
#ecec 87.5%, #ecec),ð linear-gradient(30deg, #ecec 12%,
transparent 12.5%, transparent 87%, #ecec 87.5%, #ecec),ð
linear-gradient(150deg, #ecec 12%, transparent 12.5%, transparent 87%,
#ecec 87.5%, #ecec),ð linear-gradient(60deg, #dddddd 25%,
transparent 25.5%, transparent 75%, #dddddd 75%, #dddddd),ð
linear-gradient(60deg, #dddddd 25%, transparent 25.5%, transparent 75%,
#dddddd 75%, #dddddd);ð background-size: 40px 70px;ð
background-position: 0 0, 0 0, 20px 35px, 20px 35px, 0 0, 20px 35px;ð }ð
    ð .image-section {ð flex: 1;ð
width: 65%;ð background: white;ð margin-top: 60px;ð
    margin-bottom: 20px;ð margin-left: 70px;ð
border-bottom-left-radius: 5%;ð border-top-left-radius: 5%;ð box-
shadow: -5px 0 10px -2px rgba(0, 0, 0, 0.4);ð }ð .image{ð
    height: 370px;ð width: 38%;ð justify-items:
center;ð position: absolute;ð top: 70%;ð left:
24%;ð transform: translate(-50%, -50%);ð }ð .form-
section {ð width: 50%;ð padding-bottom: 20px;ð
display: flex;ð align-items: center;ð justify-content:
center;ð background: white;ð backdrop-filter: blur(10px);ð
    margin-top: 60px;ð

```

```
margin-right: 80px;#
margin-bottom: 20px;# border-top-right-radius: 5%;#
border-bottom-right-radius: 5%;# box-shadow: 5px 0 4px rgba(0, 0,
0, 0.3);# }# .form-box {# # background: white;#
padding: 60px;# width: 100%;# height: 100%;#
justify-content: center;# # text-align: center;#
# border-top-right-radius: 5%;# border-bottom-
right-radius: 5%;# # # # #
h2{# font-family: 'Times New Roman', Times, serif;#
text-align: center;# font-size: 40px;# font-
weight: bold;# margin-bottom: 20px;# color: black;#
text-shadow: 0px 4px 10px rgba(0, 0, 0, 0.1);# # }#
h4{# font-family: 'Franklin Gothic Medium', 'Arial Narrow',
Arial, sans-serif;# color: black;# font-size: 19px;#
font-weight: bold;# padding-bottom: 10px;# }#
# # .input-group{# margin-bottom: 15px;#
text-align: left;# }# label {# font-family:
'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida Sans Unicode',
Geneva, Verdana, sans-serif;# font-weight: bold;#
display: block;# margin-bottom: 5px;# font-size: 14px;#
color: black;# letter-spacing: 1px;# }#
input {# font-family: 'Gill Sans', 'Gill Sans MT', Calibri,
'Trebuchet MS', sans-serif;# width: 100%;# padding: 12px
15px;# border: 2px solid #ccc;# border-radius: 8px;#
font-size: 16px;# transition: 0.3 ease;# #
}# # .input-group input:hover{# border-color:
#667eea;# box-shadow: 0px 0px 10px rgba(102, 126, 234, 0.5);#
transform: scale(1.02);# }# # # #
.error {# color: red;# font-size: 12px;#
margin-top: 5px;# }# .referral-section{#
display: flex;# align-items: center;# gap: 10px;#
margin-bottom: 10px;# }# #
input[type=checkbox]{# -webkit-appearance: checkbox;# -
moz-appearance: checkbox;# appearance: checkbox;#
display: inline-block;# width: auto;# }# .btn {#
width: 100%;# background: linear-gradient(to right,
#667eea, #764ba2);# color: white;# padding: 14px;#
border: none;# border-radius: 8px;# font-
size: 18px;# cursor: pointer;# transition: all 0.3s ease-
in-out;# font-weight: bold;# # letter-spacing: 1px;#
box-shadow: 0px 5px 15px rgba(102, 126, 234, 0.3);# }#
# .btn:hover {# background: linear-gradient(to
right, #764ba2, #667eea);# transform: scale(1.05);# box-
shadow: 0px 8px 20px rgba(102, 126, 234, 0.5);# # #
#referralId {# display: none;# }# # .account h4{#
padding-top: 10px;# font-size: 16px;# }# .signin{#
text-decoration: none;# font-size: 16px;# }#
.signin :hover{# text-decoration: underline;# color:
#667eea;# transition: scale(1.02);# }# # # /*
Responsive */# @media (max-width: 900px) {# .main-container {#
flex-direction: column;# width: 90%;#
height: auto;# }# .form-container, .image-
container {# width: 100%;# }# .image-
container {# height: 300px;# }# }# .text-
overlay {# position: absolute;# top: 23%;# left: 26%;# transform:
translate(-50%, -50%);# text-align: center;# color: white;# padding:
20px;# border-radius: 10px;# }# text-overlay h2{# color: black;# # font-
weight: bold;# font-size: 25px;# text-shadow: 0 0 1px #141616, 0 0 1px
#131314, 0 0 1px #121414;# overflow: hidden;# opacity: 0;# transform:
translateX(30px);# }
```

```
        animation: fadeSlideUp 1.2s ease forwards;# .text-
overlay p{# color: black;# font-weight: bold;# font-size: 22px;# text-
shadow: 0 0 1px #101111, 0 0 1px #000000, 0 0 1px #16191b;# opacity: 0;#
transform: translateX(30px);# animation: fadeSlideUp 1.2s ease forwards;# }#
@keyframes fadeSlideUp {# to {# opacity: 1;# transform: translateY(0);#
}#/#style>#
```

demo\src\main\resources\static\admin_dashboard.html

```
<!DOCTYPE html>ðhtml lang="en">ðhead>ð    <meta charset="UTF-8">ð
<title>Admin Dashboard</title>ð    <link rel="stylesheet" href="https://
cdnjs.cloudflare.com/ajax/libs/font-awesome/6.5.0/css/all.min.css">ð
<script src="https://kit.fontawesome.com/a076d05399.js"
crossorigin="anonymous"></script>ð    <script src="https://
cdnjs.cloudflare.com/ajax/libs/jspdf/2.5.1/jspdf.umd.min.js"></script>ð<script
src="https://cdnjs.cloudflare.com/ajax/libs/jspdf-autotable/3.5.28/
jspdf.plugin.autotable.min.js"></script>ð<link rel="stylesheet"
href="admin_dashboard.css">ð/head>ðbody>ð    <!-- HEADER -->ð    <header
class="header">ð        <button class="hamburger" id="menuToggle">ð
<i class="fas fa-bars"></i>ð        </button>ð        <div class="logo"></div>ð        <div
class="header-title">Admin Management System</div>ð        <div class="user-
info-mini">ð            <div class="notification-icon"
onclick="toggleNotifications()">ð                Ø=Ÿ    <span
id="notificationCount">0</span>ð            </div>ð            <div
class="user-data">ð                <span id="headerAdminName"
class="username">Loading...</span>ð                <span id="headerAdminRole"
class="name" style="font-size: 10px; color: black; margin-left:
4px;">(Admin)</span>ð            </div>ð        </div>ð    <!--
SIDEBAR -->ð    <aside class="sidebar" id="sidebar">ð        <div
class="sidebar-header">ð            <div class="user-info">ð
<div class="user-avatar" id="adminAvatar">A</div>ð            <div>ð
                <div class="user-name" id="sidebarAdminName">Loading...</
div>ð                <div class="user-role" id="sidebarAdminRole">Admin</
div>ð                <div class="user-status" id="adminStatus">Online</div>ð
            </div>ð            </div>ð            </div>ð            <ul class="nav-
menu">ð                <li><a href="#" onclick="showModule('dashboardModule')">
<i class="fas fa-tachometer-alt"></i> Dashboard</a></li>ð                <li><a
href="#" onclick="showModule('manageUsersModule')"><i class="fas fa-user-
cog"></i> Manage Users</a></li>ð                <li><a href="#"
onclick="showModule('reportsModule')"><i class="fas fa-chart-line"></i>
Reports</a></li>ð                <li><a href="#"
onclick="showModule('adminProfileModule')"><i class="fas fa-user"></i>
Profile</a></li>ð                <li><a href="#"
onclick="showModule('settingsModule')"><i class="fas fa-tools"></i>
Settings</a></li>ð                <li><a href="#"
onclick="showModule('chatModule')"><i class="fas fa-comment-sms"></i> Chat</
a></li>ð            </ul>ð            <button id="logout" class="logout-btn">ð
                <i class="fas fa-sign-out-alt"></i> Logoutð            </button>ð    </
aside>ð    <!-- MAIN CONTENT -->ð    <main class="main-content"
id="mainContent">ð        <div id="dashboardModule" class="module"
style="display: block;">ð            <div class="dashboard-header">ð                <h1
class="dashboard-title">Admin Dashboard</h1>ð                <div
class="breadcrumb">Home &gt; Dashboard</div>ð            </div>ð            <div
class="data-card fade-in">ð                <h2>Session Information</h2>ð                <div
id="sessionDataContainer">ð                    <div class="loading"
id="sessionLoading">Loading...</div>ð                    <table class="session-
table" id="sessionTable" style="display: none;">ð                        <thead>ð
                            <tr>ð                                <th>Admin ID</th>ð
                                <th>Admin Name</th>ð
                                <th>Sign-in Time</th>ð                                <th>Role</th>ð
                                <th>Logout Time</th>ð                                <th>Status</th>ð
                            </tr>ð                        <tbody>ð
                            <tr>ð                                <td id="userIdDisplay"></td>ð
                                <td id="userNameDisplay"></td>ð
                                <td id="signinTimeDisplay"></td>ð
                                <td id="logoutTimeDisplay"></td>ð
```

```

                <td id="roleDisplay"></td>
                <td id="statusDisplay"></td>
            </tr>
        </tbody>
    </table>
</div>
</div>
</div>
    <!--
Manage Users Section -->
<div id="manageUsersModule" class="module"
style="display: none;">
    <div class="dashboard-header">
        <h1
class="dashboard-title">Manage Users</h1>
        <div class="breadcrumb">Home
        &gt; Manage Users</div>
    </div>
    <!-- Search Bar -->
    <!-- Search and
Sort Controls for All Users -->
<div style="margin-bottom: 15px; display:
flex; gap: 10px; align-items: center;">
    <div style="position: relative;">
        <input type="text" id="allUserSearchInput" placeholder="Search by User
ID, Name, or Role" style="padding-left: 30px;">
        <span style="position:
absolute; left: 8px; top: 50%; transform: translateY(-50%); color: #aaa;">
        Ø=Ý </span>
    </div>
    <select id="allUserSortSelect">
<option value="">Sort by</option>
    <option value="userId">User ID</option>
    <option value="userName">User Name</option>
    <option
value="role">Role</option>
    </select>
    <button
onclick="downloadAllUsersReport()" class="download-btn"> Ø=ÜÄ Download All Users</
button>
    </div>
    <div class="data-card fade-in">
        <h2>User
List</h2>
        <div id="userLoading">Loading users...</div>
        <table
class="session-table" id="userTable" style="display: none;">
<thead>
        <tr>
            <th>User ID</th>
            <th>User Name</th>
            <th>Email</th>
            <th>Phone</th>
            <th>Role</th>
        </tr>
        </thead>
        <tbody
id="userTableBody">
            <!-- User rows will be inserted here -->
        </tbody>
        </table>
    </div>
    <div
id="reportsModule" class="module" style="display: none; padding: 20px;">
        <h2 style="font-size: 24px; margin-bottom: 20px;">Recent User
Registrations</h2>
        <div style="margin-bottom: 15px; display:
flex; gap: 10px; align-items: center;">
            <div style="position:
relative;">
                <input type="text" id="searchInput"
placeholder="Search by User ID, Name, or Role" style="padding-left: 30px;">
                <span style="position: absolute; left: 8px; top: 50%;
transform: translateY(-50%); color: #aaa;">
                Ø=Ý </span>
            </div>
            <select id="sortSelect">
<option value="">Sort by</option>
            <option value="userId">User
ID</option>
            <option value="userName">User Name</option>
            <option value="role">Role</option>
            </select>
            <button onclick="downloadUserReport()" class="download-btn">Ø=ÜÄ
Download User Report (PDF)</button>
        </div>
        <div
id="recentUsersLoading" class="loading-message">Loading recent users...</div>
        <table id="recentUsersTable" class="user-table">
            <thead>
                <tr>
                    <th>User ID</th>
                    <th>User Name</th>
                    <th>Created At</th>
                </tr>
            </thead>
            <tbody id="recentUsersTableBody">
            </tbody>
        </table>
    </div>
    <div id="adminProfileModule" class="module"
style="display: none;">
        <div class="dashboard-header">
            <h1
class="dashboard-title">Admin Profile</h1>
            <div
class="breadcrumb">Home &gt; Profile</div>
        </div>
        <div
class="profile-card">
            <div class="profile-photo">
                
            </div>
            <div class="profile-details">
                <div class="profile-row">
                    <span class="profile-label">User ID:</span>
                    <span
id="profileUserId">Loading...</span>
                </div>
                <div
class="profile-row">
                    <span class="profile-label">Name:</span>
                    <span id="profileUserName">Loading...</span>
                </div>
                <div class="profile-row">

```



```

                                <span class="profile-
label">Email:</span>Đ                                <span id="profileEmail">Loading...</span>Đ
                                </div>Đ                                <div class="profile-row">Đ                                <span
class="profile-label">Phone:</span>Đ                                <span
id="profilePhone">Loading...</span>Đ                                </div>Đ                                <div
class="profile-row">Đ                                <span class="profile-label">Role:</span>Đ
                                <span id="profileRole">Loading...</span>Đ                                </div>Đ
                                <div class="profile-row">Đ                                <span class="profile-
label">Referral Code:</span>Đ                                <span
id="profileReferralCode">Loading...</span>Đ                                </div>Đ                                <div
class="profile-row">Đ                                <span class="profile-label">Created At:</
span>Đ                                <span id="profileCreatedAt">Loading...</span>Đ                                </
div>Đ                                </div>Đ                                </div>Đ                                <div
id="settingsModule" class="module" style="display: none;">Đ                                <div
class="dashboard-header">Đ                                <h1 class="dashboard-title">Admin
Settings</h1>Đ                                <div class="breadcrumb">Home > Settings</div>Đ
</div>Đ                                <!-- Username Section -->Đ                                <div
class="settings-section">Đ                                <label>Username:</label>Đ
<span id="currentUsername" class="current-value">Loading...</span>Đ
<div id="usernameEdit" class="edit-section">Đ                                <input
type="text" id="usernameInput" placeholder="Enter new username" />Đ
                                <button class="save-button" onclick="saveUsername()">Save</
button>Đ                                <button class="cancel-button"
onclick="cancelEdit('username')">Cancel</button>Đ                                </div>Đ                                Đ
                                </div>Đ                                <!-- Phone Section -->Đ                                <div class="settings-
section">Đ                                <label>Phone:</label>Đ                                <span id="currentPhone"
class="current-value">Loading...</span>Đ                                <div id="phoneEdit"
class="edit-section">Đ                                <input type="tel" id="phoneInput"
placeholder="Enter new phone number" />Đ                                <button class="save-
button" onclick="savePhone()">Save</button>Đ                                <button
class="cancel-button" onclick="cancelEdit('phone')">Cancel</button>Đ
                                </div>Đ                                </div>Đ                                <!-- Single Edit
Button -->Đ                                <div class="settings-section">Đ                                <button class="edit-button"
id="editProfileBtn" onclick="editProfile()">Edit Profile</button>Đ                                </div>Đ
                                <!-- Profile Picture -->Đ                                <div class="settings-section">Đ
                                <label>Change Profile Picture:</label>Đ                                Đ
                                <input type="file" id="uploadInput" accept="image/*" />Đ                                </div>Đ                                Đ
                                <div id="settingsStatus" class="status-msg"></div>Đ                                Đ                                Đ
                                <div id="successPopup">Đ                                <p id="successMessage"></p>Đ                                <button
onclick="closePopup()">OK</button>Đ                                </div>Đ                                </div>Đ                                Đ                                Đ                                Đ
                                <div id="chatModule" class="module" style="display: none;">Đ
                                <h2>Chat</h2>Đ                                <!-- Chat messages will be displayed here -->Đ
                                <div class="chat-box" id="chatBox"></div>Đ                                <!-- Chat
input area -->Đ                                <div class="chat-input-area">Đ                                <input
type="text" id="chatInput" placeholder="Type your message..." />Đ
                                <button onclick="sendChatMessage()">Send</button>Đ                                </div>Đ                                Đ
                                <!-- Loading indicator for fetching chat -->Đ                                <div
id="loadingIndicator" style="display: none;">Loading...</div>Đ                                </div>Đ                                Đ
                                <!-- Notifications Panel -->Đ                                <div id="notificationPanel"
class="notification-panel">Đ                                <h3>Đ=Ÿ Notifications</h3>Đ                                <ul
id="adminNotificationList"></ul>Đ                                <button
onclick="closeNotifications()" style="Đ                                position: absolute;Đ
top: 10px;Đ                                right: 10px;Đ                                background: none;Đ                                border: none;Đ
font-size: 20px;Đ                                color: red;Đ                                cursor: pointer;Đ
font-weight: bold;Đ                                ">x</button>Đ                                </div>Đ                                Đ                                </main>Đ                                Đ
<script src="admin_dashboard.js"></script>ĐĐ/body>Đ/html>Đ

```

demo\src\main\resources\static\admin_dashboard2.html

```
<!DOCTYPE html>ðhtml lang="en">ðhead>ð    <meta charset="UTF-8">ð    <meta
name="viewport" content="width=device-width, initial-scale=1.0">ð
<title>Admin Dashboard</title>ð    <link rel="stylesheet"
href="admin_dashboard.css">ð/head>ðbody>ð    <div class="admin-container">ð
    <!-- Sidebar -->ð        <aside class="sidebar">ð                <h2>Admin
Panel</h2>ð                <ul>ð                    <li><a href="#">ð<ßà Dashboard</a></li>ð
                    <li><a href="#">ð<Û Users</a></li>ð                    <li><a
href="#">ð<™p Settings</a></li>ð                    <li><a href="#">ð<Ûa Logout</a></li>ð
                </ul>ð        </aside>ð        <!-- Main Content -->ð        <main
class="main-content">ð            <header>ð                <h1>Welcome, Admin</
h1>ð                <div class="notification-icon"
onclick="toggleNotifications()">ð                    ð=Ý    <span
id="notificationCount">0</span>ð                </div>ð            </header>ð
            <!-- Recent Users Table -->ð                <section class="dashboard-
section">ð                    <h2>ð=Û Recent User Signups</h2>ð                    <div
class="table-container">ð                        <table>ð
<thead>ð                            <tr>ð                                <th>Name</th>ð
                                <th>Email</th>ð
                                <th>Phone</th>ð                                <th>Signup Time</th>ð
                            </tr>ð                        </thead>ð
                            <tbody id="userList">ð                            <!--
Users will be loaded here dynamically -->ð                                </tbody>ð
                        </table>ð                    </div>ð                </section>ð
            <!-- Notifications Panel -->ð                <div
id="notificationPanel" class="notification-panel">ð                    <h3>ð=Ý
Notifications</h3>ð                    <ul id="adminNotificationList"></ul>ð
                    <button onclick="closeNotifications()">Close</button>ð
                </div>ð        </main>ð        </div>ð        <script>ð            // Fetch
Recent Usersðasync function fetchRecentUsers() {ð        try {ð            const
response = await fetch("http://localhost:8080/api/recent-signups");ð
            const users = await response.json();ð            const userList =
document.getElementById("userList");ð            userList.innerHTML = "";ð
            users.forEach(user => {ð                const row = document.createElement("tr");ð
                row.innerHTML = `ð                    <td>${user.userId}</td>ð
                <td>${user.userName}</td>ð                    <td>${user.email}</
td>ð                    <td>${user.phone}</td>ð                    <td>${new
Date(user.createdAt).toLocaleString()}</td>ð                `;ð
                userList.appendChild(row);ð            });ð        } catch (error) {ð
            console.error("Error fetching users:", error);ð        }ðð/ Fetch Admin
Notificationsðasync function fetchAdminNotifications() {ð        try {ð            const
response = await fetch("http://localhost:8080/admin-notifications/unread");ð
            const notifications = await response.json();ð            const
notificationList = document.getElementById("adminNotificationList");ð
            const notificationCount = document.getElementById("notificationCount");ð
            notificationList.innerHTML = "";ð            notificationCount.textContent
= notifications.length;ð            notifications.forEach(notification => {ð
                const item = document.createElement("li");ð
                item.innerText = notification.message;ð                item.onclick = async () => {ð
                    await markAsRead(notification.id);ð
                    fetchAdminNotifications();ð                };ð
                notificationList.appendChild(item);ð            });ð        } catch (error) {ð
            console.error("Error fetching notifications:", error);ð        }ðð/ ' Mark
notification as read in the databaseðasync function markAsRead(notificationId)
{ð        try {ð            await fetch(`http://localhost:8080/admin-notifications/mark-
as-read/${notificationId}`, {ð                method: "POST",ð                headers:
{ "Content-Type": "application/json" }ð            });ð        } catch (error) {ð
            console.error("Error marking notification as read:", error);ð        }ðð/ Close
Notification Panelð
```

```

function closeNotifications() {
document.getElementById("notificationPanel").style.display = "none";
Toggle Notifications Panel
function toggleNotifications() {
const panel =
document.getElementById("notificationPanel");
panel.style.display =
panel.style.display === "block" ? "none" : "block";
// Load Initial Data
fetchRecentUsers();
fetchAdminNotifications();
}
</script>
<style>
/*
General Styling
*/
body {
font-family: "Poppins", sans-serif;
margin: 0;
padding: 0;
display: flex;
background: #f4f4f4;
}
/* Dashboard
Container */
.admin-container {
display: flex;
width: 100%;
}
/* Sidebar */
.sidebar {
width: 200px;
background: #2c3e50;
color: white;
padding: 20px;
height: 100vh;
position: fixed;
left: 0;
top: 0;
animation: slideIn 0.5s ease-in-out;
}
/* Sidebar h2 */
.sidebar h2 {
text-align: center;
padding: 15px;
transition: background 0.3s;
}
/* Sidebar ul li */
.sidebar ul li {
padding: 15px;
background: #34495e;
}
/* Sidebar ul li a */
.sidebar ul li a {
color: white;
text-decoration: none;
font-weight: bold;
}
/* Main Content */
.main-content {
flex: 1;
margin-left: 250px;
padding: 20px;
animation: fadeIn 0.8s ease-in-out;
}
/* Header */
.header {
display: flex;
justify-content: space-between;
align-items: center;
background: #fff;
padding: 15px;
border-radius: 10px;
box-shadow: 0px 4px 10px rgba(0, 0, 0, 0.1);
}
/* Notification Icon */
.notification-icon {
cursor: pointer;
font-size: 22px;
position: relative;
}
/* Notification Count */
.notification-count {
background: red;
color: white;
padding: 5px 8px;
border-radius: 50%;
position: absolute;
top: -8px;
right: -8px;
font-size: 14px;
}
/* Dashboard Section */
.dashboard-section {
margin-top: 20px;
background: white;
padding: 20px;
border-radius: 10px;
box-shadow: 0px 4px 10px rgba(0, 0, 0, 0.1);
}
/* Table */
table-container {
overflow-x: auto;
}
table {
width: 100%;
border-collapse: collapse;
margin-top: 10px;
}
thead, td {
border: 1px solid #ccc;
padding: 12px;
text-align: left;
}
tbody tr {
background: #2c3e50;
color: white;
}
/* Notifications */
.notification-panel {
display: none;
position: fixed;
top: 50px;
right: 20px;
background: white;
border-radius: 8px;
width: 320px;
box-shadow: 0px 4px 10px rgba(0, 0, 0, 0.2);
padding: 15px;
}
/* Notification-panel ul */
.notification-panel ul {
list-style: none;
padding: 0;
}
/* Notification-panel ul li */
.notification-panel ul li {
padding: 10px;
border-bottom: 1px solid #ddd;
cursor: pointer;
}
/* Animations */
@keyframes slideIn {
from {
transform: translateX(-100%);
}
to {
transform: translateX(0);
}
}
@keyframes fadeIn {
from {
opacity: 0;
}
to {
opacity: 1;
}
}
</style>
</body>
</html>

```

demo\src\main\resources\static\forgot.html

```
<title>Forgot Page</title><div class="container"><div class="text-
overlay"></div><div class="form-section"><div class="form-box">
  <h2>Forgotten Password</h2><p>Create your new
password</p><!-- Email Verification Section --><div
id="emailSection"><div class="input-group"><label
for="email">Enter your registered Email</label><input
type="email" id="email" required placeholder="Email"><div
class="error" id="emailError"></div><button
class="btn" onclick="checkEmail()">Verify Email</button></div>
<!-- Password Reset Section (Hidden Initially) --><div
id="resetPasswordSection" style="display: none;"><div
class="input-group"><label for="newPassword">New Password</
label><input type="password" id="newPassword" required
placeholder="New Password"><div class="error"
id="passwordError"></div><div class="input-
group"><label for="confirmNewPassword">Confirm Password</
label><input type="password" id="confirmNewPassword" required
placeholder="Confirm Password"><div class="error"
id="confirmPasswordError"></div><button
class="btn" onclick="resetPassword()">Reset Password</button></div>
<div class="account"><h5>Remembered your password? <a
class="signin" href="signin.html">Sign In</a></h5></div>
</div></div><script>
  // Verify if the email exists in the
  databasefunction checkEmail() {
    let email =
    document.getElementById("email").value.trim();
    let emailError =
    document.getElementById("emailError");
    // Clear previous error messages
    emailError.textContent = "";
    fetch("http://localhost:8080/api/verify-
    email", {
      method: "POST",
      headers: { "Content-Type":
    "application/json" },
      body: JSON.stringify({ email: email })
    })
    .then(response => response.json())
    .then(data => {
      if
    (data.status === "error") {
      emailError.textContent = data.message;
    } else {
      // Hide email section and show password reset
      section
      document.getElementById("emailSection").style.display =
      "none";
      document.getElementById("resetPasswordSection").style.display = "block";
    }
  })
  .catch(error => {
    console.error("Error:", error);
    emailError.textContent = "Server error. Please try again."
  });
  //
  Reset the password after validationfunction resetPassword() {
    let
    newPassword = document.getElementById("newPassword").value.trim();
    let
    confirmNewPassword =
    document.getElementById("confirmNewPassword").value.trim();
    let
    passwordError = document.getElementById("passwordError");
    let
    confirmPasswordError = document.getElementById("confirmPasswordError");
    //
    Password validation pattern
    let passwordPattern = /^(?=.*[A-Z])(?=.*[a-z])
    (?!.*\d)(?!.*[@$!%*?&])[A-Za-z\d@$!%*?&]{8,}$/;
    // Validate new password
    if (!passwordPattern.test(newPassword)) {
      passwordError.textContent = "Password must be 8+ chars, 1 uppercase, 1
      lowercase, 1 number & 1 special char.";
      return;
    } else {
      passwordError.textContent = "";
    }
    // Confirm password match
    if
    (newPassword !== confirmNewPassword) {
      confirmPasswordError.textContent = "Passwords do not match!";
      return;
    } else {
      confirmPasswordError.textContent = "";
    }
    // Send
    the new password to the server for updating
    let email =
    document.getElementById("email").value.trim();
    fetch("http://
    localhost:8080/api/reset-password", {
      method: "POST",
      headers:
    { "Content-Type": "application/json",
      body: JSON.stringify({
        email: email,
        newPassword: newPassword
      })
    })
    .then(response => response.json())
  }
```

```

        .then(data => {
            if (data.status === "success") {
                alert("Password updated successfully! Redirecting to login...");
                window.location.href = "signin.html"; // Redirect to login page
            } else {
                alert(data.message);
            }
        }).catch(error => {
            console.error("Error:", error);
            alert("Server error. Please try again.");
        });
    </script>
    <style>
        * {
            margin: 0;
            padding: 0;
            box-sizing: border-box;
            font-family: "poppins", sans-serif;
        }
        body {
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
            background: white;
        }
        .container {
            display: flex;
            width: 100%;
            max-width: 1700px;
            height: 700px;
            border-radius: 8px;
            overflow: hidden;
            box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1);
            background-color: #fafafa;
            background-image: linear-gradient(135deg, #f5f5f5 25%, transparent 25%),
                linear-gradient(225deg, #f5f5f5 25%, transparent 25%),
                linear-gradient(45deg, #f5f5f5 25%, transparent 25%),
                linear-gradient(315deg, #f5f5f5 25%, transparent 25%);
            background-position: 10px 0, 10px 0, 0 0, 0 0;
            background-size: 20px 20px;
            background-repeat: repeat;
        }
        .image-section {
            width: 65%;
            flex: 1;
            background: url("pics/forex\ flaticon.gif ") no-repeat center / 100% 100%;
            margin-top: 100px;
            margin-bottom: 100px;
            margin-left: 80px;
            border-bottom-left-radius: 5%;
            border-top-left-radius: 5%;
            box-shadow: -5px 0 10px -2px rgba(0, 0, 0, 0.2);
        }
        .form-section {
            width: 50%;
            display: flex;
            justify-content: center;
            align-items: center;
            backdrop-filter: blur(10px);
            margin-top: 100px;
            margin-bottom: 100px;
            margin-right: 80px;
            height: 100%;
            background: white;
            padding: 80px;
            text-align: center;
            border-top-right-radius: 5%;
            border-bottom-right-radius: 5%;
            box-shadow: 5px 0 4px -2px rgba(0, 0, 0, 0.3);
        }
        h2 {
            padding-bottom: 10px;
            text-align: center;
            font-size: 30px;
            font-weight: bold;
            color: black;
            text-shadow: 0px 4px 10px rgba(0, 0, 0, 0.2);
            font-family: 'Franklin Gothic Medium', 'Arial Narrow', Arial, sans-serif;
        }
        p {
            text-align: center;
            font-size: 20px;
            color: black;
            margin-bottom: 20px;
            text-align: center;
            margin-bottom: 10px;
            font-weight: bold;
            font-size: 20px;
        }
        .heading {
            padding-bottom: 50px;
        }
        .input-group {
            margin-bottom: 15px;
            text-align: left;
        }
        label {
            font-family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida Sans Unicode', Geneva, Verdana, sans-serif;
            font-weight: bold;
            display: block;
            margin-bottom: 5px;
            font-size: 14px;
            color: black;
            letter-spacing: 1px;
        }
        input {
            font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;
            width: 100%;
            padding: 12px 15px;
            border: 2px solid #ccc;
            border-radius: 8px;
            font-size: 16px;
            transition: 0.3 ease;
        }
        .input-group input:hover {
            border-color: #667eea;
            box-shadow: 0px 0px 10px rgba(102, 126, 234, 0.5);
            transform: scale(1.02);
        }
        .error {
            color: red;
            font-size: 12px;
        }
        .btn {
            width: 100%;
            background: linear-gradient(to right, #667eea, #764ba2);
            color: black;
            padding: 14px;
            border: none;
            border-radius: 8px;
            font-size: 16px;
            cursor: pointer;
            transition: all 0.3s ease-in-out;
            font-weight: bold;
            letter-spacing: 1px;
            box-shadow: 0px 5px 15px rgba(102, 126, 234, 0.3);
        }
        .btn:hover {
            background: linear-gradient(to right, #764ba2, #667eea);
        }
    </style>

```

```

        transform: scale(1.05);@        box-shadow: 0px 8px
20px rgba(102, 126, 234, 0.5);@    }@ #resetPasswordSection {@
display: none;@    }@    @ .account{@        padding-top: 20px;@        text-
align: center;@    }@    .account h5{@        font-size: 16px;@    }@    .signin{@
        text-decoration: none;@        @        font-weight: bold;@        font-
size: 16px@    }@    @ .signin :hover{@        text-decoration: underline;@
        transform: scale(1.05);@    @    }@    .text-overlay h2 {@position:
absolute;@top: 6%;@left: 28%;@transform: translate(-50%, -50%);@text-align:
center;@padding: 20px;@border-radius: 10px;@}@text-overlay h2{@color: black;@font-
weight: bold;@font-size: 23px;@text-shadow: 0 0 1px #121414, 0 0 1px #181b1b, 0
0 1px #0e0f0f;@animation: fadeInOut 2.5s ease-in-out infinite;@display:
inline-block;@}@text-overlay p {@position: absolute;@top: 91%;@left: 28%;@
transform: translate(-50%, -50%);@text-align: center;@padding: 20px;@border-
radius: 10px;@}@text-overlay p{@color: black;@font-weight: bold;@font-size: 21px;@
text-shadow: 0 0 1px #161718, 0 0 1px #0f1011, 0 0 1px #171a1b;@animation:
fadeInOut 2.5s ease-in-out infinite;@display: inline-block;@}@keyframes
fadeInOut {@0% {@opacity: 0;@}@50% {@opacity: 1;@}@100% {@
opacity: 0;@}@}@/style>@

```

demo\src\main\resources\static\homepage.html

```
<!DOCTYPE html>ðhtml lang="en">ðhead>ð <meta charset="UTF-8">ð <meta
name="viewport" content="width=device-width, initial-scale=1.0">ð
<title>TradePro - Advanced Trading Platform</title>ð <style>ð  * {ð
margin: 0;ð    padding: 0;ð    box-sizing: border-box;ð    font-family:
'Segoe UI',Tahoma, Geneva, Verdana, sans-serif;ð    }ð    ð body {ð
background-color: #f0f2f5;ð    color: #333;ð    }ð    ð .navbar {ð
display: flex;ð    justify-content: space-between;ð    align-items: center;ð
    background-color: #0a2647;ð    padding: 1rem 2rem;ð    color: white;ð
    box-shadow: 0 2px 4px rgba(0,0,0,0.1);ð    }ð    ð .logo {ð    font-
size: 1.5rem;ð    font-weight: bold;ð    color: white;ð    text-
decoration: none;ð    }ð    ð .logo span {ð    color: #38bdf8;ð    }ð    ð
.nav-links {ð    display: flex;ð    gap: 1.5rem;ð    }ð    ð .nav-links
a {ð    color: white;ð    text-decoration: none;ð    font-weight: 500;ð
    transition: color 0.3s ease;ð    }ð    ð .nav-links a:hover {ð
color: #38bdf8;ð    }ð    ð .auth-buttons {ð    display: flex;ð    gap:
1rem;ð    }ð    ð .auth-buttons button {ð    padding: 0.5rem 1rem;ð
border-radius: 4px;ð    font-weight: 500;ð    cursor: pointer;ð
transition: all 0.3s ease;ð    }ð    ð .login-btn {ð    background:
transparent;ð    border: 1px solid white;ð    color: white;ð    }ð    ð
.login-btn:hover {ð    background: rgba(255, 255, 255, 0.1);ð    }ð    ð
.signup-btn {ð    background: #38bdf8;ð    border: none;ð    color:
#0a2647;ð    }ð    ð .signup-btn:hover {ð    background: #0ea5e9;ð    }ð    ð
.hero {ð    padding: 2rem;ð    background: #0a2647;ð    color: white;ð
    text-align: center;ð    padding-bottom: 4rem;ð    }ð    ð .hero h1 {ð
font-size: 2.5rem;ð    margin-bottom: 1rem;ð    }ð    ð .hero p {ð
font-size: 1.1rem;ð    max-width: 800px;ð    margin: 0 auto;ð    margin-
bottom: 2rem;ð    color: #cbd5e1;ð    }ð    ð .get-started {ð    padding:
0.75rem 1.5rem;ð    background: #38bdf8;ð    color: #0a2647;ð    border:
none;ð    border-radius: 4px;ð    font-weight: 600;ð    font-size: 1rem;ð
    cursor: pointer;ð    transition: background 0.3s ease;ð    }ð    ð
.get-started:hover {ð    background: #0ea5e9;ð    }ð    ð .dashboard {ð
max-width: 1200px;ð    margin: -2rem auto 2rem;ð    padding: 1rem;ð
display: grid;ð    grid-template-columns: repeat(12, 1fr);ð    grid-
gap: 1rem;ð    }ð    ð .card {ð    background: white;ð    border-radius:
8px;ð    box-shadow: 0 4px 6px rgba(0,0,0,0.1);ð    padding: 1.5rem;ð
transition: transform 0.3s ease, box-shadow 0.3s ease;ð    }ð    ð
.card:hover {ð    transform: translateY(-5px);ð    box-shadow: 0 8px
15px rgba(0,0,0,0.1);ð    }ð    ð .price-chart {ð    grid-column: span 8;ð
    }ð    ð .market-summary {ð    grid-column: span 4;ð    }ð    ð .volume-
chart, .technical-indicators {ð    grid-column: span 6;ð    }ð    ð .order-
book, .recent-trades {ð    grid-column: span 6;ð    }ð    ð .news-feed {ð
grid-column: span 12;ð    }ð    ð .card h2 {ð    margin-bottom: 1rem;ð
font-size: 1.25rem;ð    color: #0a2647;ð    display: flex;ð
justify-content: space-between;ð    align-items: center;ð    }ð    ð .card
h2 .icon {ð    color: #64748b;ð    font-size: 1rem;ð    cursor: pointer;ð
    }ð    ð .card h2 .icon:hover {ð    color: #0a2647;ð    }ð    ð .table {ð
width: 100%;ð    border-collapse: collapse;ð    }ð    ð .table
th, .table td {ð    padding: 0.75rem;ð    text-align: left;ð    border-
bottom: 1px solid #e2e8f0;ð    }ð    ð .table th {ð    font-weight: 600;ð
color: #64748b;ð    }ð    ð .up {ð    color: #10b981;ð    }ð    ð
.down {ð    color: #ef4444;ð    }ð    ð .news-item {ð    display: flex;ð
gap: 1rem;ð    margin-bottom: 1rem;ð    padding-bottom: 1rem;ð
border-bottom: 1px solid #e2e8f0;ð    }ð    ð .news-item:last-child {ð
margin-bottom: 0;ð    padding-bottom: 0;ð    border-bottom: none;ð    }ð    ð
.news-image {ð    width: 80px;ð    height: 80px;ð    background-color:
#e2e8f0;ð    border-radius: 4px;ð    overflow: hidden;ð    }ð    ð .news-
content h3 {ð    font-size: 1rem;ð    margin-bottom: 0.5rem;ð    }ð    ð
.news-content p {ð    font-size: 0.875rem;ð    color: #64748b;ð
margin-bottom: 0.5rem;ð
```

```

    } @media (max-width: 768px) {
      .news-content .timestamp { font-size: 0.75rem; color: #94a3b8; }
      .footer { background: #0f172a; color: white; padding: 3rem 2rem; }
      .footer-content { max-width: 1200px; margin: 0 auto; display: grid; grid-template-columns: repeat(4, 1fr); gap: 2rem; }
      .footer-section h3 { margin-bottom: 1rem; font-size: 1.1rem; }
      .footer-section a { display: block; color: #cbd5e1; text-decoration: none; margin-bottom: 0.5rem; transition: color 0.3s ease; }
      .footer-section a:hover { color: #38bdf8; }
      .copyright { text-align: center; margin-top: 2rem; padding-top: 2rem; border-top: 1px solid #1e293b; color: #94a3b8; font-size: 0.875rem; }
    }
    @media (max-width: 768px) {
      .dashboard { grid-template-columns: 1fr; }
      .price-chart, .market-summary, .volume-chart, .technical-indicators, .order-book, .recent-trades, .news-feed { grid-column: span 1; }
    }
    .footer-content { grid-template-columns: repeat(2, 1fr); }
    @media (max-width: 480px) {
      .nav-links { display: none; }
      .footer-content { grid-template-columns: 1fr; }
    }
  
```

</style></head><body>
 <nav class="navbar">
 Trade
 <div class="nav-links">
 Dashboard
 Markets
 Trading
 Portfolio
 Analytics
 </div>
 <div class="auth-buttons">
 <button class="login-btn">Login</button>
 <button class="signup-btn">Sign Up</button>
 </div>
 </nav>
 <section class="hero">
 <h1>Professional Trading at Your Fingertips</h1>
 <p>Access real-time market data, advanced charting tools, and insightful analytics to make informed trading decisions.</p>
 <button class="get-started">Get Started Now</button>
 </section>
 <div class="dashboard">
 <div class="card price-chart">
 <h2>BTC/USD Price Chart</h2>
 <div id="price-chart-container" style="height: 320px;></div>
 </div>
 <div class="card market-summary">
 <h2>Market Summary</h2>
 <table class="table">
 <thead>
 <tr>
 <th>Symbol</th>
 <th>Price</th>
 <th>24h Change</th>
 </tr>
 </thead>
 <tbody>
 <tr>
 <td>BTC/USD</td>
 <td>48,236.75</td>
 <td class="up">+2.34%</td>
 </tr>
 <tr>
 <td>ETH/USD</td>
 <td>3,152.89</td>
 <td class="up">+1.78%</td>
 </tr>
 <tr>
 <td>SOL/USD</td>
 <td>103.42</td>
 <td class="down">-0.86%</td>
 </tr>
 <tr>
 <td>XRP/USD</td>
 <td>0.5724</td>
 <td class="up">+0.43%</td>
 </tr>
 <tr>
 <td>DOT/USD</td>
 <td>6.87</td>
 <td class="down">-1.21%</td>
 </tr>
 </tbody>
 </table>
 </div>
 <div class="card volume-chart">
 <h2>Trading Volume</h2>
 <div id="volume-chart-container" style="height: 250px;></div>
 </div>
 <div class="card technical-indicators">
 <h2>Technical Indicators</h2>
 <div id="indicators-chart-container" style="height: 250px;></div>
 </div>
 <div class="card order-book">
 <h2>Order Book</h2>
 <div id="order-book-chart" style="height: 250px;></div>
 </div>
 <div class="card recent-trades">
 <h2>Recent Trades</h2>
 <table class="table">
 <thead>
 <tr>
 <th>Amount</th>
 <th>Time</th>
 <th>Price</th>
 </tr>
 </thead>
 <tbody>
 <tr>
 <td>14:32:45</td>
 <td class="up">48,236.75</td>
 <td>0.0546</td>
 </tr>
 <tr>
 <td>14:32:35</td>
 <td class="down">48,231.20</td>
 <td>0.1250</td>
 </tr>
 <tr>
 <td>14:31:58</td>
 <td class="down">48,228.35</td>
 <td></td>
 </tr>
 </tbody>
 </table>
 </div>
 </div>


```

<td>0.0387</td>
</tr>
<tr>
<td>14:31:42</td>
<td>
class="up">48,240.60</td>
<td>0.2145</td>
</tr>
<tr>
<td>14:31:23</td>
<td class="up">48,245.15</td>
<td>0.0833</td>
</tr>
</tbody>
</table>
</div>
<div class="footer">
<div class="footer-section">
<h3>TradePro</h3>
<a href="#">About Us</a>
<a href="#">Careers</a>
<a href="#">Press</a>
<a href="#">Blog</a>
</div>
<div class="footer-section">
<h3>Products</h3>
<a href="#">Exchange</a>
<a href="#">Institutional Services</a>
<a href="#">API</a>
<a href="#">Mobile App</a>
</div>
<div class="footer-section">
<h3>Support</h3>
<a href="#">Help Center</a>
<a href="#">Contact Us</a>
<a href="#">FAQ</a>
<a href="#">Security</a>
</div>
<div class="footer-section">
<h3>Legal</h3>
<a href="#">Terms of Service</a>
<a href="#">Privacy Policy</a>
<a href="#">Risk Disclosure</a>
<a href="#">Cookie Policy</a>
</div>
</div>
<div class="copyright">
&copy; 2025 TradePro. All rights reserved.
</div>
</footer>
<script src="https://cdn.jsdelivr.net/npm/chart.js/3.9.1/chart.min.js"></script>
// Price Chart
const priceCtx = document.createElement('canvas');
document.getElementById('price-chart-container').appendChild(priceCtx);
const priceLabels = Array.from({length: 30}, (_, i) => i + 1);
const priceData = [46720, 46850, 47100, 46900, 47300, 47200, 47500, 47450, 47800, 47750, 47900, 47850, 48000, 48150, 48050, 48200, 48100, 48350, 48250, 48400, 48300, 48150, 48250, 48350, 48237, 48300, 48150, 48237, 48150, 48236];
new Chart(priceCtx, {
type: 'line',
data: {
labels: priceLabels,
datasets: [{
label: 'BTC/USD Price',
data: priceData,
borderColor: '#38bdf8',
backgroundColor: 'rgba(56, 189, 248, 0.1)',
tension: 0.4,
fill: true,
pointRadius: 0,
borderWidth: 2
}]
},
options: {
responsive: true,
maintainAspectRatio: false,
plugins: {
legend: {
display: false
},
tooltip: {
mode: 'index',
intersect: false
},
scales: {
x: {
display: true
},
y: {
display: true,
grid: {
display: false
}
}
}
},
color: 'rgba(0, 0, 0, 0.05)'
});
// Volume Chart
const volumeCtx = document.createElement('canvas');
document.getElementById('volume-chart-container').appendChild(volumeCtx);
const volumeLabels = ['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec'];
const volumeData = [850, 750, 820, 780, 880, 930, 920, 1020, 980, 925, 970, 1050];
new Chart(volumeCtx, {
type: 'bar',
data: {
labels: volumeLabels,
datasets: [{
label: 'Trading Volume (millions USD)',
data: volumeData,
backgroundColor: '#38bdf8'
}]
},
borderRadius: 4,
options: {
responsive: true,
maintainAspectRatio: false,
plugins: {
legend: {
display: false
},
scales: {
x: {
display: true
},
y: {
display: true,
grid: {
display: false
}
}
},
color: 'rgba(0, 0, 0, 0.05)'
},
// Technical Indicators Chart
const indicatorsCtx = document.createElement('canvas');
document.getElementById('indicators-chart-container').appendChild(indicatorsCtx);
const indicatorLabels = Array.from({length: 20}, (_, i) => i + 1);
const priceDataShort = priceData.slice(-20);
const smaData = [];
const emaData = [];
// Calculate SMA and EMA
for (let i = 0; i < priceDataShort.length; i++) {
if (i >= 4) {

```

```

        // Simple 5-day moving average
        const sma =
(priceDataShort[i] + priceDataShort[i-1] + priceDataShort[i-2] +
priceDataShort[i-3] + priceDataShort[i-4]) / 5;
        smaData.push(sma);
    } else {
        smaData.push(null);
    }
    // Simple EMA
    approximation
    if (i === 0) {
        emaData.push(priceDataShort[0]);
    } else {
        const k = 2 / (10 + 1); // 10-day EMA factor
        const ema = priceDataShort[i] * k + emaData[i-1] * (1-k);
        emaData.push(ema);
    }
    new Chart(indicatorsCtx, {
        type:
'line',
        data: {
            labels: indicatorLabels,
            datasets: [
                {
                    label: 'Price',
                    data: priceDataShort,
                    borderColor: '#38bdf8',
                    borderWidth: 2,
                    tension: 0.1,
                    pointRadius: 0
                },
                {
                    label: 'SMA (5)',
                    data: smaData,
                    borderColor: '#ef4444',
                    borderWidth: 2,
                    tension: 0.1,
                    pointRadius: 0
                },
                {
                    label: 'EMA (10)',
                    data: emaData,
                    borderColor: '#10b981',
                    borderWidth: 2,
                    tension: 0.1,
                    pointRadius: 0
                }
            ],
            options: {
                responsive: true,
                maintainAspectRatio:
false,
                plugins: {
                    tooltip: {
                        mode: 'index',
                        intersect: false,
                        display: false
                    },
                    scales: {
                        x: {
                            grid: {
                                display: false
                            }
                        },
                        y: {
                            grid: {
                                color: 'rgba(0, 0, 0, 0.05)'
                            }
                        }
                    }
                }
            }
        },
        // Order Book Chart
        const orderBookCtx = document.createElement('canvas');
        document.getElementById('order-book-chart').appendChild(orderBookCtx);
        const askPrices = [48240, 48250, 48260, 48270, 48280, 48290, 48300, 48310,
48320, 48330];
        const askVolumes = [1.2, 2.5, 3.8, 4.2, 5.5, 6.1, 7.4, 8.2,
9.5, 10.8];
        const bidPrices = [48230, 48220, 48210, 48200, 48190,
48180, 48170, 48160, 48150, 48140];
        const bidVolumes = [1.5, 2.8, 3.6,
4.5, 5.3, 6.7, 7.9, 8.6, 9.3, 10.5];
        new Chart(orderBookCtx, {
            type: 'bar',
            data: {
                labels:
[...bidPrices.reverse(), ...askPrices],
                datasets: [
                    {
                        label: 'Bids',
                        data:
[...bidVolumes.reverse(), ...Array(askPrices.length).fill(null)],
                        backgroundColor: '#10b981'
                    },
                    {
                        label: 'Asks',
                        data:
[...Array(bidPrices.length).fill(null), ...askVolumes],
                        backgroundColor: '#ef4444'
                    }
                ],
            },
            options: {
                responsive: true,
                maintainAspectRatio: false,
                indexAxis: 'x',
                plugins: {
                    tooltip: {
                        callbacks: {
                            title: function(tooltipItems) {
                                return 'R' +
tooltipItems[0].label;
                            },
                            scales: {
                                x: {
                                    stacked: true,
                                    display: false,
                                    ticks: {
                                        callback: function(value, index) {
                                            const allLabels =
[...bidPrices.reverse(), ...askPrices];
                                            // Show fewer ticks
                                            for readability
                                            if (index === 0 || index === 9 || index === 10
|| index === 19) {
                                                return 'R' + allLabels[index];
                                            }
                                            return '';
                                        }
                                    },
                                    y: {
                                        stacked: true,
                                        grid: {
                                            color: 'rgba(0, 0, 0, 0.05)'
                                        },
                                        display: true,
                                        text: 'Volume'
                                    }
                                }
                            }
                        }
                    }
                }
            }
        });
    </script>
</body>
</html>

```

demo\src\main\resources\static\manageUsers.html

```
<!-- manageUsers.html (already correct!) --><div class="dashboard-header"><div class="dashboard-title">Manage Users</div><div class="breadcrumb">Home > Manage Users</div></div><div class="data-card fade-in"><h2>User List</h2><div id="userLoading">Loading users...</div><table class="session-table" id="userTable" style="display: none;"><thead><tr><th>Name</th><th>Role</th><th>Email</th><th>Phone</th><th>User ID</th></tr></thead><tbody><tr><td colspan="5"><!-- User rows will be inserted here --></td></tr></tbody></table></div><script>async function loadUsers() {const loadingDiv = document.getElementById("userLoading");const table = document.getElementById("userTable");const tbody = document.getElementById("userTableBody");loadingDiv.style.display = "block";table.style.display = "none";try {const response = await fetch("http://localhost:8080/api/all-users");if (!response.ok) throw new Error("Failed to fetch users");const users = await response.json();tbody.innerHTML = "";users.forEach(user => {const row = document.createElement("tr");row.innerHTML = `<td>${user.userId}</td><td>${user.userName}</td><td>${user.email}</td><td>${user.phone}</td><td>${user.role}</td>`;tbody.appendChild(row);});loadingDiv.style.display = "none";table.style.display = "table";} catch (error) {console.error("Error loading users:", error);loadingDiv.innerText = "Error loading user data.";}document.addEventListener("DOMContentLoaded", loadUsers);</script><style>* {margin: 0; padding: 0; box-sizing: border-box; font-family: 'Segoe UI', sans-serif;}body {display: flex; background: #f5f7fa; height: 100vh; overflow: hidden;}header {position: fixed; top: 0; left: 0; width: 100%; height: 60px; background: #2e3a59; color: white; display: flex; align-items: center; padding: 0 20px; z-index: 1000; box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);}user-role {font-size: 12px; color: #888; margin-top: 2px;}hamburger {background: none; border: none; color: white; font-size: 22px; cursor: pointer; margin-right: 15px;}header-title {font-size: 20px; flex: 1;}user-info-mini {display: flex; align-items: center; font-size: 16px; font-weight: 600; justify-content: center; gap: 10px;}user-info-mini .name {text-align: center;}sidebar {position: fixed; top: 60px; left: 0; width: 240px; height: calc(100% - 60px); background: #1c2536; color: #fff; transition: transform 0.3s ease-in-out; overflow-y: auto; z-index: 999;}sidebar.active {transform: translateX(-100%);}sidebar-header {padding: 20px; border-bottom: 1px solid #333;}user-info {display: flex; align-items: center;}user-avatar {background: #f39c12; color: white; font-weight: bold; width: 45px; height: 45px; border-radius: 50%; display: flex; align-items: center; justify-content: center; margin-right: 15px; font-size: 20px; animation: popIn 0.4s ease;}user-name {font-size: 16px; font-weight: bold;}user-status {font-size: 13px; color: lightgreen;}nav-menu {list-style: none; padding: 10px 0;}nav-menu li a {display: flex; align-items: center; padding: 12px 20px; color: #ccc; text-decoration: none; transition: background 0.3s; border-bottom: 1px solid #ccc;}nav-menu li a:active {background: #34495e; color: #fff;}nav-menu i {margin-right: 12px;}logout-btn {width: 90%; margin: 20px auto; display: flex; align-items: center; justify-content: center; background: #e74c3c; color: white; border: none; padding: 12px; border-radius: 6px; font-size: 16px; cursor: pointer; transition: background 0.3s ease;}logout-btn:hover {background: #c0392b;}main-content {margin-left: 240px; padding: 80px 30px 30px 30px; width: 100%; overflow-y: auto; transition: margin-left 0.3s ease-in-out;}main-content.expanded {margin-left: 0;}dashboard-header {margin-bottom: 20px; animation: fadeInUp 0.5s ease;}dashboard-title {font-size: 28px; margin-bottom: 5px;}breadcrumb {color: #888;}
```

```

}data-card {background: white;padding: 25px;border-radius: 10px;box-
shadow: 0 3px 10px rgba(0,0,0,0.05);animation: fadeIn 1s ease;#loading {#
padding: 10px;color: #333;#session-table {width: 100%;border-collapse:
collapse;margin-top: 15px;#session-table th,#session-table td {padding: 12px;#
border: 1px solid #ddd;text-align: center;#session-table th {background:
#2e3a59;color: white;#session-table td {font-size: 15px;##* Notification
Icon */notification-icon {cursor: pointer;font-size: 22px;position: relative;#
}notificationCount {background: red;color: white;padding: 2px 6px;border-
radius: 50%;position: absolute;#op: -8px;#ight: -8px;font-size: 12px;##*
Notifications */notification-panel {display: none;position: fixed;#op: 50px;#
right: 20px;background: white;border-radius: 8px;#idth: 320px;box-shadow: 0px
4px 10px rgba(0, 0, 0, 0.2);padding: 15px;#notification-panel ul {list-
style: none;padding: 0;#notification-panel ul li {padding: 10px;border-
bottom: 1px solid #ddd;cursor: pointer;##keyframes fadeIn {from {opacity: 0;
transform: translateY(10px);}to {opacity: 1; transform: translateY(0);}##
@keyframes fadeInUp {from {opacity: 0; transform: translateY(20px);}to
{opacity: 1; transform: translateY(0);}##keyframes popIn {0% { transform:
scale(0); opacity: 0; }100% { transform: scale(1); opacity: 1; }##*
Responsive */@media (max-width: 768px) {#sidebar {# transform:
translateX(-100%);#sidebar.active {# transform: translateX(0);#main-
content {# margin-left: 0;# padding: 70px 20px;##/style>#/style>#

```

demo\src\main\resources\static\modules\manageUsers.html

```
<!-- manageUsers.html (already correct!) --><div class="dashboard-header"><div class="dashboard-title">Manage Users</div><div class="breadcrumb">Home > Manage Users</div></div><div class="data-card fade-in"><h2>User List</h2><div id="userLoading">Loading users...</div><table class="session-table" id="userTable" style="display: none;"><thead><tr><th>Name</th><th>Role</th><th>Email</th><th>Phone</th></tr></thead><tbody id="userTableBody"><!-- User rows will be inserted here --></tbody></table><script>async function loadUsers() {const loadingDiv = document.getElementById("userLoading");const table = document.getElementById("userTable");const tbody = document.getElementById("userTableBody");loadingDiv.style.display = "block";table.style.display = "none";try {const response = await fetch("http://localhost:8080/api/all-users");if (!response.ok) throw new Error("Failed to fetch users");const users = await response.json();tbody.innerHTML = "";users.forEach(user => {const row = document.createElement("tr");row.innerHTML = `<td>${user.userId}</td><td>${user.userName}</td><td>${user.email}</td><td>${user.phone}</td></tr>`;tbody.appendChild(row);});loadingDiv.style.display = "none";table.style.display = "table";} catch (error) {console.error("Error loading users:", error);loadingDiv.innerText = "Error loading user data.";}document.addEventListener("DOMContentLoaded", loadUsers);</script><style>* {margin: 0;padding: 0;box-sizing: border-box;font-family: 'Segoe UI', sans-serif;}body {display: flex;background: #f5f7fa;height: 100vh;overflow: hidden;}header {position: fixed;top: 0;left: 0;width: 100%;height: 60px;background: #2e3a59;color: white;display: flex;align-items: center;padding: 0 20px;z-index: 1000;box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);}user-role {font-size: 12px;color: #888;margin-top: 2px;}hamburger {background: none;border: none;color: white;font-size: 22px;cursor: pointer;margin-right: 15px;}header-title {font-size: 20px;flex: 1;}user-info-mini {display: flex;align-items: center;font-size: 16px;font-weight: 600;justify-content: center;gap: 10px;}user-info-mini .name {text-align: center;}sidebar {position: fixed;top: 60px;left: 0;width: 240px;height: calc(100% - 60px);background: #1c2536;color: #fff;transition: transform 0.3s ease-in-out;overflow-y: auto;z-index: 999;}sidebar.active {transform: translateX(-100%);}sidebar-header {padding: 20px;border-bottom: 1px solid #333;}user-info {display: flex;align-items: center;}user-avatar {background: #f39c12;color: white;font-weight: bold;width: 45px;height: 45px;border-radius: 50%;display: flex;align-items: center;justify-content: center;margin-right: 15px;font-size: 20px;animation: popIn 0.4s ease;}user-name {font-size: 16px;font-weight: bold;}user-status {font-size: 13px;color: lightgreen;}nav-menu {list-style: none;padding: 10px 0;}nav-menu li a {display: flex;align-items: center;padding: 12px 20px;color: #ccc;text-decoration: none;transition: background 0.3s;}nav-menu li a:active {background: #34495e;color: #fff;}nav-menu i {margin-right: 12px;}logout-btn {width: 90%;margin: 20px auto;display: flex;align-items: center;justify-content: center;background: #e74c3c;color: white;border: none;padding: 12px;border-radius: 6px;font-size: 16px;cursor: pointer;transition: background 0.3s;}logout-btn:hover {background: #c0392b;}main-content {margin-left: 240px;padding: 80px 30px 30px 30px;width: 100%;overflow-y: auto;transition: margin-left 0.3s ease-in-out;}main-content.expanded {margin-left: 0;}dashboard-header {margin-bottom: 20px;animation: fadeInUp 0.5s ease;}dashboard-title {font-size: 28px;margin-bottom: 5px;}breadcrumb {color: #888;}
```

```

}data-card {background: white;padding: 25px;border-radius: 10px;box-
shadow: 0 3px 10px rgba(0,0,0,0.05);animation: fadeIn 1s ease;#loading {#
padding: 10px;color: #333;#session-table {width: 100%;border-collapse:
collapse;margin-top: 15px;#session-table th,#session-table td {padding: 12px;#
border: 1px solid #ddd;text-align: center;#session-table th {background:
#2e3a59;color: white;#session-table td {font-size: 15px;##* Notification
Icon */notification-icon {cursor: pointer;font-size: 22px;position: relative;#
}notificationCount {background: red;color: white;padding: 2px 6px;border-
radius: 50%;position: absolute;#op: -8px;#ight: -8px;font-size: 12px;##*
Notifications */notification-panel {display: none;position: fixed;#op: 50px;#
right: 20px;background: white;border-radius: 8px;#width: 320px;box-shadow: 0px
4px 10px rgba(0, 0, 0, 0.2);padding: 15px;#notification-panel ul {list-
style: none;padding: 0;#notification-panel ul li {padding: 10px;border-
bottom: 1px solid #ddd;cursor: pointer;##keyframes fadeIn {from {opacity: 0;
transform: translateY(10px);}to {opacity: 1; transform: translateY(0);}##
@keyframes fadeInUp {from {opacity: 0; transform: translateY(20px);}to
{opacity: 1; transform: translateY(0);}##keyframes popIn {0% { transform:
scale(0); opacity: 0; }100% { transform: scale(1); opacity: 1; }##*
Responsive */@media (max-width: 768px) {#sidebar {# transform:
translateX(-100%);#sidebar.active {# transform: translateX(0);#main-
content {# margin-left: 0;# padding: 70px 20px;##/style>#/style>#

```

```

<title>Signin Page</title><div class="container"><div
class="text-overlay"></div><div class="image-section">
</div><div class="form-section"><div
class="form-box"><h2>Welcome Back!</h2><p>Login to
Continue</p><form id="signinForm">
<div class="input-group"><label
for="userid">Username</label><input type="text"
id="userid" name="userid" placeholder="Enter Username, Email, or Phone"
required><div class="error" id="useridError"></div>
</div><div class="input-group"><label for="password">Password</label>
<input type="password" id="password" name="password" placeholder="Enter
Password" required><div class="error"
id="passwordError"></div><div
class="remember-me"><input type="checkbox"
id="rememberMe"><label for="rememberMe">Remember Me</
label><h5><a href="forgot.html" class="forgot">Forgot
Password?</a></h5></div><button type="submit"
class="btn">Sign In</button><div id="result"></div>
<h5>Don't have an account? <a class="signup"
href="signup.html">Sign Up</a></h5></form>
</div><script>
document.getElementById("signinForm").addEventListener("submit", async
function (event) {<div event.preventDefault(); // Prevent default form
submission<div let enteredValue =
document.getElementById("userid").value.trim();<div let enteredPassword =
document.getElementById("password").value.trim();<div let resultDiv =
document.getElementById("result");<div // Clear previous errors<div
document.getElementById("useridError").textContent = "";<div
document.getElementById("passwordError").textContent = "";<div
resultDiv.textContent = "";<div if (enteredValue === "" || enteredPassword ===
"") {<div resultDiv.textContent = "Please fill in all fields."<div
resultDiv.style.color = "red";<div return;<div }<div try {<div let
response = await fetch("http://localhost:8080/api/signin", {<div
method: "POST",<div headers: {<div "Content-Type":
"application/json"<div },<div body: JSON.stringify({<div
userid: enteredValue,<div password: enteredPassword<div
})<div });<div let result = await response.json();<div
if (result.message === "Login successful") {<div resultDiv.textContent
= "Login successful! Redirecting..."<div resultDiv.style.color =
"green";<div // ' Store data in sessionStorage<div
sessionStorage.setItem("userId", result.userId);<div
sessionStorage.setItem("userName", result.userName);<div
localStorage.setItem("userId", result.userId);<div
localStorage.setItem("userName", result.userName);<div
localStorage.setItem("loggedInUser", enteredValue);<div
setTimeout(() => {<div window.location.href
="user_dashboard3.html"; // Redirect to dashboard<div }, 1500);<div
} else {<div resultDiv.textContent = result.message;<div
resultDiv.style.color = "red";<div }<div } catch (error) {<div
console.error("Error:", error);<div resultDiv.textContent =
"Server error. Please try again later."<div resultDiv.style.color = "red";<div
})<div }</script><div style><div * {<div margin: 0;<div padding: 0;<div
box-sizing: border-box;<div font-family: "Poppins", sans-serif;<div
}<div body {<div display: flex;<div justify-content: center;<div
align-items: center;<div height: 100vh;<div background: #ffff;<div }<div
.container {<div display: flex;<div width: 100%;<div max-width:
1700px;<div height: 700px;<div box-shadow: 0px 4px 10px rgba(0, 0, 0,
0.1);<div

```

```

border-radius: 15px;@ overflow: hidden;@ @
background-color: #f0f5ff;@ background-image: @
radial-gradient(circle at 100% 150%, #f0f5ff 24%, #c8d7ff 25%, #c8d7ff 28%,
#f0f5ff 29%, #f0f5ff 36%, #c8d7ff 36%, #c8d7ff 40%, transparent 40%,
transparent),@ radial-gradient(circle at 0 150%, #f0f5ff 24%,
#c8d7ff 25%, #c8d7ff 28%, #f0f5ff 29%, #f0f5ff 36%, #c8d7ff 36%, #c8d7ff 40%,
transparent 40%, transparent),@ radial-gradient(circle at 50%
100%, #c8d7ff 10%, #f0f5ff 11%, #f0f5ff 23%, #c8d7ff 24%, #c8d7ff 30%,
#f0f5ff 31%, #f0f5ff 43%, #c8d7ff 44%, #c8d7ff 50%, #f0f5ff 51%, #f0f5ff 63%,
#c8d7ff 64%, #c8d7ff 71%, transparent 71%, transparent),@
radial-gradient(circle at 100% 50%, #c8d7ff 5%, #f0f5ff 6%, #f0f5ff 15%,
#c8d7ff 16%, #c8d7ff 20%, #f0f5ff 21%, #f0f5ff 30%, #c8d7ff 31%, #c8d7ff 35%,
#f0f5ff 36%, #f0f5ff 45%, #c8d7ff 46%, #c8d7ff 49%, transparent 50%,
transparent),@ radial-gradient(circle at 0% 50%, #c8d7ff 5%,
#f0f5ff 6%, #f0f5ff 15%, #c8d7ff 16%, #c8d7ff 20%, #f0f5ff 21%, #f0f5ff 30%,
#c8d7ff 31%, #c8d7ff 35%, #f0f5ff 36%, #f0f5ff 45%, #c8d7ff 46%, #c8d7ff 49%,
transparent 50%, transparent);@ background-size: 100px 50px;@ @
@@ @@@ @ @ .image-section {@ width: 700px;@
flex: 1;@ @ background: url("pics/unnamed\ flaticon.gif") no-
repeat center / auto ;@ @ margin-top: 100px;@ margin-bottom:
100px;@ margin-left: 80px;@ border-bottom-left-radius: 5%;@
border-top-left-radius: 5%;@ box-shadow: -5px 0 10px -2px rgba(0, 0, 0,
0.4);@ }@ .form-section {@ width: 50%;@ @ display: flex;@
align-items: center;@ justify-content: center;@ @
backdrop-filter: blur(10px);@ margin-top: 100px;@ margin-
bottom: 100px;@ margin-right: 80px;@ @ }@ .form-box {@ @
background: white;@ padding: 80px;@ width: 100%;@ height:
100%;@ @ @ text-align: center;@ border-top-right-radius:
5%;@ border-bottom-right-radius: 5%;@ box-shadow: 5px 0 4px -2px
rgba(0, 0, 0, 0.3);@ @ }@ h2 {@ font-family: 'Times New Roman',
Times, serif;@ text-align: center;@ font-size: 40px;@ font-
weight: bold;@ @ color: black;@ text-shadow: 0px 4px 10px
rgba(0, 0, 0, 0.1);@ @ }@ p {@ font-family: 'Times New Roman',
Times, serif;@ text-align: center;@ font-size: 20px;@ font-
weight: bold;@ color: black;@ }@ h4 {@ font-family: 'Franklin
Gothic Medium', 'Arial Narrow', Arial, sans-serif;@ color: black;@
font-size: 19px;@ font-weight: bold;@ padding-bottom:
20px;@ @ }@ h5 {@ color: black;@ font-weight: bold;@
font-size: 15px;@ }@ .signup {@ text-decoration: none;@ color:
red;@ }@ @ .signup :hover {@ text-decoration: underline;@
transform: scale(1.05);@ color: red;@ }@ .input-group {@ @
margin-bottom: 15px;@ text-align: left;@ }@ label {@ font-
family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida Sans
Unicode', Geneva, Verdana, sans-serif;@ padding-bottom: 10px;@
font-weight: bold;@ letter-spacing: 1px;@ display: block;@ @
font-size: 16px;@ color: black;@ }@ input {@ font-
family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;@
width: 100%;@ padding: 12px 15px;@ border: none;@
border-radius: 6px;@ font-size: 16px;@ border: 2px solid #ccc;@
color: black;@ }@ @ .input-type {@ color:
black;@ }@ .input-group input:hover {@ border-color: #667eea;@
box-shadow: 0px 0px 10px rgba(102, 126, 234, 0.5);@ transform:
scale(1.02);@ }@ input:focus {@ outline: none;@ transform:
scale(1.05);@ box-shadow: 0px 0px 10px rgba(255, 255, 0.4);@ }@
.error {@ color: red;@ font-size: 14px;@ margin-top:
5px;@ }@ @ .remember-me {@ display: flex;@ @ align-
items: center;@ @ }@ input[type=checkbox] {@ -webkit-
appearance: checkbox;@ -moz-appearance: checkbox;@ appearance:
checkbox;@ display: inline-block;@ width: auto;@ }@ .remember-
me input {@

```



```

        margin-right: 8px;@        cursor: pointer;@        left: 0;@
    @ }@ .btn {@        width: 100%;@        background: linear-
gradient(to right, #667eea, #764ba2);@        color: white;@        padding:
14px;@        border: none;@        border-radius: 8px;@        font-size: 18px;@
        cursor: pointer;@        transition: all 0.3s ease-in-out;@        font-
weight: bold;@        text-transform: uppercase;@        letter-spacing: 1px;@
        box-shadow: 0px 5px 15px rgba(102, 126, 234, 0.3);@ }@ @
    @ @ .btn:hover {@        background: linear-gradient(to right,
#764ba2, #667eea);@        transform: scale(1.05);@        box-shadow: 0px 8px
20px rgba(102, 126, 234, 0.5);@ }@ #result {@        font-weight: bold;@
        margin-top: 15px;@        color: #00ff99;@ }@ .forgot {@
padding-left: 230px;@        text-decoration: none;@        color: blue;@ }@
.forgot :hover{@        background: #ff4e88;@        transform:
scale(1.05);@ }@ @ @ @media (max-width: 768px) {@        .container {@
        flex-direction: column;@        height: auto;@        }@
        .image-section {@        height: 200px;@        }@        .form-
section {@        height: auto;@        padding: 20px;@        }@
        .form-box {@        width: 90%;@        }@ }@ .text-overlay h2
{position: absolute;@top: 6%;@left: 25%;@transform: translate(-50%, -50%);@text-
align: center;@color: black;@padding: 20px;@border-radius: 10px;@}@.text-overlay
h2{font-weight: bold;@font-size: 24px;@text-shadow: 0 0 1px #000000, 0 0 1px
#0e1213, 0 0 1px #131616;@animation: glowFade 2.5s ease-in-out infinite;@
display: inline-block;@}@.text-overlay p {position: absolute;@top: 94%;@left: 26%;@
transform: translate(-50%, -50%);@text-align: center;@color: black;@padding:
20px;@border-radius: 10px;@}@.text-overlay p{font-weight: bold;@font-size: 20px;@
text-shadow: 0 0 1px #080a0a, 0 0 1px #101214, 0 0 1px #121416;@animation:
glowFade 2.5s ease-in-out infinite;@display: inline-block;@}@keyframes
glowFade {@0%, 100% {@opacity: 0.8;@text-shadow: 0 0 5px #ffffff;@}@
50% {@opacity: 1;@text-shadow: 0 0 15px #00ffff;@}@}@/style>@

```

demo\src\main\resources\static\signin2.html

```
<!DOCTYPE html>ðhtml lang="en">ðhead>ð <meta charset="UTF-8">ð <meta
name="viewport" content="width=device-width, initial-scale=1.0">ð
<title>Creative Authentication System</title>ð <style>ð :root {ð
  --primary: #4776E6;ð --secondary: #8E54E9;ð --
light: #f8f9fa;ð --dark: #343a40;ð --success: #28a745;ð
  --error: #dc3545;ð --transition: all 0.4s ease;ð }ð
  * {ð box-sizing: border-box;ð margin: 0;ð
padding: 0;ð font-family: 'Segoe UI', Tahoma, Geneva,
Verdana, sans-serif;ð }ð body, html {ð height: 100%;ð
overflow: hidden;ð background-color: var(--light);ð
}ð .navbar {ð position: fixed;ð display:
flex;ð justify-content: space-between;ð padding: 15px
30px;ð background: linear-gradient(135deg, var(--primary), var(--
secondary));ð border-radius: 30px;ð top: 20px;ð
left: 50%;ð transform: translateX(-50%);ð width: 90%;ð
max-width: 600px;ð z-index: 1000;ð box-
shadow: 0 10px 30px rgba(0, 0, 0, 0.1);ð }ð .nav-item {ð
color: white;ð text-align: center;ð padding:
10px 20px;ð text-decoration: none;ð font-weight: 500;ð
border-radius: 20px;ð transition: var(--transition);ð
}ð .nav-item:hover {ð background-color: rgba(255,
255, 255, 0.2);ð transform: translateY(-3px);ð }ð .auth-
container {ð position: relative;ð width: 100%;ð
height: 100vh;ð overflow: hidden;ð display: flex;ð
align-items: center;ð justify-content: center;ð
background: linear-gradient(135deg, #f5f7fa, #c3cfe2);ð }ð
.form-container {ð position: relative;ð width:
100%;ð height: 100%;ð transition: transform 0.8s cubic-
bezier(0.68, -0.55, 0.265, 1.55);ð }ð .form-section {ð
position: absolute;ð width: 100%;ð height: 100%;ð
display: flex;ð flex-direction: column;ð
justify-content: center;ð align-items: center;ð
transition: var(--transition);ð opacity: 0;ð visibility:
hidden;ð }ð .form-section.active {ð opacity: 1;ð
visibility: visible;ð }ð .form-card {ð
width: 90%;ð max-width: 450px;ð padding: 40px;ð
background: white;ð border-radius: 20px;ð box-shadow: 0
15px 35px rgba(0, 0, 0, 0.1);ð transform: translateY(50px);ð
opacity: 0;ð transition: var(--transition);ð }ð
.form-section.active .form-card {ð transform: translateY(0);ð
opacity: 1;ð }ð .form-header {ð text-align:
center;ð margin-bottom: 30px;ð }ð .form-title {ð
font-size: 28px;ð color: var(--dark);ð margin-
bottom: 10px;ð font-weight: 600;ð }ð .form-subtitle {ð
color: #6c757d;ð font-size: 16px;ð }ð .form-
group {ð margin-bottom: 25px;ð position: relative;ð
}ð .form-group label {ð position: absolute;ð
top: 0;ð left: 0;ð font-size: 16px;ð
color: #6c757d;ð padding: 18px 15px;ð pointer-
events: none;ð transition: var(--transition);ð }ð .form-
group input {ð width: 100%;ð padding: 18px 15px;ð
font-size: 16px;ð border: none;ð border-
bottom: 2px solid #e9ecef;ð background-color: transparent;ð
outline: none;ð transition: var(--transition);ð }ð
.form-group input:focus,ð .form-group input:valid {ð
border-bottom-color: var(--primary);ð }ð .form-group input:focus
+ label,ð .form-group input:valid + label {ð top: -25px;ð
left: 0;ð font-size: 14px;ð color: var(--
primary);ð }
```

```

    }@ .error-message {@ color: var(--error);@
    font-size: 14px;@ margin-top: 5px;@ display:
none;@ }@ .submit-btn {@ width: 100%;@
padding: 15px;@ background: linear-gradient(135deg, var(--primary),
var(--secondary));@ color: white;@ border: none;@
border-radius: 30px;@ font-size: 16px;@ font-
weight: 600;@ cursor: pointer;@ transition: var(--
transition);@ box-shadow: 0 5px 15px rgba(0, 0, 0, 0.1);@ }@
.submit-btn:hover {@ transform: translateY(-3px);@
box-shadow: 0 8px 25px rgba(0, 0, 0, 0.15);@ }@ .form-
footer {@ text-align: center;@ margin-top: 25px;@
font-size: 15px;@ color: #6c757d;@ }@ .form-
link {@ color: var(--primary);@ text-decoration: none;@
font-weight: 600;@ transition: var(--transition);@
}@ .form-link:hover {@ color: var(--secondary);@
}@ .bg-shapes {@ position: absolute;@
width: 100%;@ height: 100%;@ overflow: hidden;@
z-index: -1;@ }@ .shape {@ position: absolute;@
border-radius: 50%;@ background: linear-
gradient(135deg, rgba(71, 118, 230, 0.1), rgba(142, 84, 233, 0.1));@
animation: float 10s infinite ease-in-out;@ }@
.shape:nth-child(1) {@ width: 300px;@ height:
300px;@ top: -100px;@ right: -100px;@ animation-
delay: 0s;@ }@ .shape:nth-child(2) {@ width: 200px;@
height: 200px;@ bottom: 50px;@ left: 30px;@
animation-delay: 2s;@ }@ .shape:nth-child(3) {@
width: 150px;@ height: 150px;@ bottom: 150px;@
right: 100px;@ animation-delay: 4s;@ }@
.shape:nth-child(4) {@ width: 100px;@ height:
100px;@ top: 120px;@ left: 150px;@ animation-
delay: 6s;@ }@ @keyframes float {@ 0% {@
transform: translate(0, 0) rotate(0deg);@ }@ 50% {@
transform: translate(20px, 20px) rotate(10deg);@ }@
100% {@ transform: translate(0, 0) rotate(0deg);@
}@ }@ .success-message {@ display: none;@
position: fixed;@ top: 20px;@ left: 50%;@
transform: translateX(-50%);@ background-color: var(--
success);@ color: white;@ padding: 15px 25px;@
border-radius: 30px;@ box-shadow: 0 5px 15px rgba(0, 0, 0, 0.2);@
z-index: 1000;@ animation: slideDown 0.5s forwards;@
}@ @keyframes slideDown {@ from {@ top:
-50px;@ opacity: 0;@ }@ to {@ top:
20px;@ opacity: 1;@ }@ }@ .progress-
bar {@ position: absolute;@ bottom: 0;@ left: 0;@
height: 5px;@ background: linear-gradient(90deg, var(--
primary), var(--secondary));@ transition: width 0.5s ease;@ }@
.indicator-container {@ display: flex;@ justify-
content: center;@ margin-top: 20px;@ }@ .form-indicator
{@ width: 10px;@ height: 10px;@ border-radius:
50%;@ background-color: #e9ecef;@ margin: 0 5px;@
transition: var(--transition);@ }@ .form-
indicator.active {@ background: linear-gradient(135deg, var(--
primary), var(--secondary));@ transform: scale(1.3);@ }@
/* Social Login Buttons */@ .social-login {@ display:
flex;@ justify-content: center;@ margin: 20px 0;@
gap: 15px;@ }@ .social-btn {@ display: flex;@
justify-content: center;@ align-items: center;@
width: 50px;@ height: 50px;@ border-radius:
50%;@ background-color: white;@ border: 1px solid #e9ecef;@
cursor: pointer;@

```

```

        transition: var(--transition);
        box-shadow: 0 3px 10px rgba(0, 0, 0, 0.1);
    }
    .social-btn:hover {
        transform: translateY(-3px);
        box-shadow: 0 5px 15px rgba(0, 0, 0, 0.15);
    }
    .social-icon {
        width: 24px;
        height: 24px;
    }
    .divider {
        display: flex;
        align-items: center;
        margin: 25px 0;
    }
    .divider-line {
        flex-grow: 1;
        height: 1px;
        background-color: #e9ecef;
    }
    .divider-text {
        margin: 0 15px;
        color: #6c757d;
        font-size: 14px;
    }
</style></head><body>
    <!-- Success Message -->
    <div class="success-message" id="success-message">
        <!-- Background Shapes -->
        <div class="bg-shapes">
            <div class="shape"></div>
            <div class="shape"></div>
            <div class="shape"></div>
            <div class="shape"></div>
        </div>
        <!-- Forms Container -->
        <div class="forms-container">
            <!-- Sign In Form -->
            <div class="form-section" id="signin-section">
                <div class="form-card">
                    <div class="form-header">
                        <h2 class="form-title">Welcome Back</h2>
                        <p class="form-subtitle">Sign in to continue your journey</p>
                    </div>
                    <div class="divider">
                        <div class="divider-line"></div>
                        <span class="divider-text">OR</span>
                        <div class="divider-line"></div>
                    </div>
                    <form id="signin-form">
                        <div class="form-group">
                            <input type="email" id="signin-email" required>
                            <label for="signin-email">Email Address</label>
                            <div class="error-message" id="signin-email-error">Please enter a valid email address</div>
                        </div>
                        <div class="form-group">
                            <input type="password" id="signin-password" required>
                            <label for="signin-password">Password</label>
                            <div class="error-message" id="signin-password-error">Password must be at least 6 characters</div>
                        </div>
                        <button type="submit" class="submit-btn">Sign In</button>
                    </form>
                    <div class="form-footer">
                        <a href="#" class="form-link" id="signin-to-forgot">Forgot Password?</a>
                        <div class="form-footer">
                            Don't have an account? <a href="#" class="form-link" id="signin-to-signup">Sign Up</a>
                        </div>
                    </div>
                    <div class="indicator-container">
                        <div class="form-indicator active"></div>
                        <div class="form-indicator"></div>
                    </div>
                </div>
            </div>
            <!-- Sign Up Form -->
            <div class="form-section" id="signup-section">
                <div class="form-card">
                    <div class="form-header">
                        <h2 class="form-title">Create Account</h2>
                    </div>
                    <form id="signup-form">
                        <div class="form-group">
                            <input type="text" id="signup-name" required>
                            <label for="signup-name">Full Name</label>
                            <div class="error-message" id="signup-name-error">Please enter your name</div>
                        </div>
                        <div class="form-group">
                            <input type="email" id="signup-email" required>
                            <label for="signup-email">Email Address</label>
                            <div class="error-message" id="signup-email-error">Please enter a valid email address</div>
                        </div>
                    </form>
                </div>
            </div>
        </div>
    </div>
</body>
</html>

```

```

</div>ð
        <div class="form-group">ð
<input type="password" id="signup-password" required>ð
        <label for="signup-password">Password</label>ð
        <div class="error-message" id="signup-password-
error">Password must be at least 8 characters with letters and numbers</div>ð
        </div>ð        <div class="form-group">ð
        <input type="password" id="signup-confirm"
required>ð        <label for="signup-confirm">Confirm
Password</label>ð        <div class="error-message"
id="signup-confirm-error">Passwords do not match</div>ð
</div>ð        <button type="submit" class="submit-btn">Create
Account</button>ð        </form>ð        <div
class="form-footer">ð        Already have an account? <a
href="#" class="form-link" id="signup-to-signin">Sign In</a>ð
        </div>ð        <div class="indicator-
container">ð        <div class="form-indicator"></div>ð
        <div class="form-indicator active"></div>ð
        <div class="form-indicator"></div>ð
</div>ð        </div>ð        </div>ð        <!-- Forgot
Password Form -->ð        <div class="form-section" id="forgot-section">ð
        <div class="form-card">ð        <div class="form-
header">ð        <h2 class="form-title">Reset Password</h2>ð
        <p class="form-subtitle">We'll send you a reset link</
p>ð        </div>ð        <form id="forgot-form">ð
        <div class="form-group">ð
<input type="email" id="forgot-email" required>ð
<label for="forgot-email">Email Address</label>ð
<div class="error-message" id="forgot-email-error">Please enter a valid email
address</div>ð        </div>ð        <button
type="submit" class="submit-btn">Send Reset Link</button>ð
</form>ð        <div class="form-footer">ð
Remember your password? <a href="#" class="form-link" id="forgot-to-
signin">Sign In</a>ð        </div>ð        <div
class="indicator-container">ð        <div class="form-
indicator"></div>ð        <div class="form-indicator"></div>ð
        <div class="form-indicator active"></div>ð
        </div>ð        </div>ð        </div>ð
<!-- Progress Bar -->ð        <div class="progress-bar" id="progress-
bar"></div>ð        </div>ð        </div>ð        <script>ð        // Form Elementsð
        const signinSection = document.getElementById('signin-section');ð
        const signupSection = document.getElementById('signup-section');ð
        const forgotSection = document.getElementById('forgot-section');ð
        const progressBar = document.getElementById('progress-bar');ð
const successMessage = document.getElementById('success-message');ð        ð
        // Navigationð        document.getElementById('signin-to-
signup').addEventListener('click', function(e) {ð
e.preventDefault();ð        slideToForm('signup');ð        });ð        ð
        document.getElementById('signup-to-signin').addEventListener('click',
function(e) {ð        e.preventDefault();ð        slideToForm('signin');ð
        });ð        ð        document.getElementById('signin-to-
forgot').addEventListener('click', function(e) {ð
e.preventDefault();ð        slideToForm('forgot');ð        });ð        ð
        document.getElementById('forgot-to-signin').addEventListener('click',
function(e) {ð        e.preventDefault();ð        slideToForm('signin');ð
        });ð        ð        // Form Switching Functionð        function
slideToForm(formName) {ð        // Remove active class from all sectionsð
        signinSection.classList.remove('active');ð
        signupSection.classList.remove('active');ð
        forgotSection.classList.remove('active');ð

```

```

                                ð // Update
indicatorsð const indicators = document.querySelectorAll('.form-
indicator');ð indicators.forEach(ind =>
ind.classList.remove('active'));ð // Set active form
and update progress barð if (formName === 'signin') {ð
    signinSection.classList.add('active');ð
indicators[0].classList.add('active');ð progressBar.style.width
= '33.3%';ð } else if (formName === 'signup') {ð
signupSection.classList.add('active');ð
indicators[1].classList.add('active');ð progressBar.style.width
= '66.6%';ð } else if (formName === 'forgot') {ð
forgotSection.classList.add('active');ð
indicators[2].classList.add('active');ð progressBar.style.width
= '100%';ð }ð // Initializedð
window.addEventListener('load', function() {ð slideToForm('signin');ð
    });ð // Form Validationsð
document.getElementById('signin-form').addEventListener('submit', function(e)
{ð e.preventDefault();ð let isValid = true;ð
    ð // Email validationð const email =
document.getElementById('signin-email').value;ð if (!
validateEmail(email)) {ð document.getElementById('signin-email-
error').style.display = 'block';ð isValid = false;ð }
else {ð document.getElementById('signin-email-
error').style.display = 'none';ð }ð //
Password validationð const password =
document.getElementById('signin-password').value;ð if
(password.length < 6) {ð document.getElementById('signin-
password-error').style.display = 'block';ð isValid = false;ð
    } else {ð document.getElementById('signin-password-
error').style.display = 'none';ð }ð if
(isValid) {ð showSuccessMessage('Signed in successfully!');ð
    // Here you would typically send the form data to your serverð
    }ð });ð document.getElementById('signup-
form').addEventListener('submit', function(e) {ð e.preventDefault();ð
    ð let isValid = true;ð // Name
validationð const name = document.getElementById('signup-
name').value;ð if (name.trim() === '') {ð
document.getElementById('signup-name-error').style.display = 'block';ð
    isValid = false;ð } else {ð
document.getElementById('signup-name-error').style.display = 'none';ð
    }ð // Email validationð const
email = document.getElementById('signup-email').value;ð if (!
validateEmail(email)) {ð document.getElementById('signup-email-
error').style.display = 'block';ð isValid = false;ð }
else {ð document.getElementById('signup-email-
error').style.display = 'none';ð }ð //
Password validationð const password =
document.getElementById('signup-password').value;ð if
(password.length < 8 || ![a-zA-Z]/.test(password) || ![
0-9]/.test(password)) {ð document.getElementById('signup-
password-error').style.display = 'block';ð isValid = false;ð
    } else {ð document.getElementById('signup-password-
error').style.display = 'none';ð }ð //
Password confirmationð const confirm =
document.getElementById('signup-confirm').value;ð if (confirm !==
password) {ð document.getElementById('signup-confirm-
error').style.display = 'block';ð isValid = false;ð }
else {ð document.getElementById('signup-confirm-
error').style.display = 'none';ð }ð if
(isValid) {ð

```

```

        showSuccessMessage('Account created
successfully!');ð
        // Here you would typically send the form
data to your serverð
        setTimeout(() => {ð
slideToForm('signin');ð
        }, 2000);ð
        }ð
        });ð
        document.getElementById('forgot-
form').addEventListener('submit', function(e) {ð
        e.preventDefault();ð
        let isValid = true;ð
        // Email
validationð
        const email = document.getElementById('forgot-
email').value;ð
        if (!validateEmail(email)) {ð
document.getElementById('forgot-email-error').style.display = 'block';ð
        isValid = false;ð
        } else {ð
document.getElementById('forgot-email-error').style.display = 'none';ð
        }ð
        if (isValid) {ð
showSuccessMessage('Password reset link sent! Check your email.');

```

demo\src\main\resources\static\signin3.html

```
<!DOCTYPE html>ðhtml lang="en"ðheadð <meta charset="UTF-8" />ð <meta
name="viewport" content="width=device-width, initial-scale=1.0" />ð
<title>Signin/Signup Page</title>ð <style>ð    * {ð        margin: 0;ð
padding: 0;ð        box-sizing: border-box;ð        font-family: "Poppins", sans-
serif;ð    }ð    body {ð        display: flex;ð        justify-content: center;ð
align-items: center;ð        ð        background: #fff;ð    }ð    .container {ð
display: flex;ð        width: 100%;ð        max-width: 1400px;ð        height: 850px;ð
        background: lightgoldenrodyellow;ð        box-shadow: 0px 4px 10px rgba(0,
0, 0, 0.3);ð        border-radius: 15px;ð        overflow: hidden;ð    }ð    .image-
section {ð        width: 50%;ð        background: url("pics/art-rachen-sM4r-swmcoY-
unsplash-Photoroom.png") no-repeat center center/cover;ð    }ð    .form-section
{ð        height: 850px;ð        width: 50%;ð        background: white;ð    ð
display: flex;ð        align-items: center;ð        justify-content: center;ð    }ð
    .form-box {ð        width: 100%;ð        max-width: 400px;ð        display: none;ð
    }ð    .form-box.active {ð        display: block;ð    }ð    h2 {ð        font-size:
32px;ð        font-weight: bold;ð        color: black;ð        margin-bottom: 20px;ð
        text-align: center;ð    }ð    .input-group {ð        margin-bottom: 15px;ð
        text-align: left;ð    }ð    label {ð        font-weight: bold;ð        font-
size: 16px;ð        color: black;ð        display: block;ð        margin-bottom: 5px;ð
    }ð    input {ð        width: 100%;ð        padding: 12px 15px;ð        font-size:
16px;ð        border: 2px solid #ccc;ð        border-radius: 6px;ð    }ð
input:hover {ð        border-color: #667eea;ð        box-shadow: 0px 0px 10px
rgba(102, 126, 234, 0.5);ð        transform: scale(1.02);ð    }ð    input:focus {ð
        outline: none;ð        transform: scale(1.05);ð        box-shadow: 0px 0px
10px rgba(255, 255, 255, 0.4);ð    }ð    .error {ð        color: red;ð        font-
size: 14px;ð        margin-top: 5px;ð    }ð    .remember-me {ð        display: flex;ð
        align-items: center;ð        ð    }ð    input[type=checkbox]{ð        -
webkit-appearance: checkbox;ð        -moz-appearance: checkbox;ð
appearance: checkbox;ð        display: inline-block;ð        width: auto;ð    }ð
    .remember-me input {ð        margin-right: 8px;ð        cursor: pointer;ð
        left: 0;ð        ð    }ð    .btn {ð        width: 100%;ð        background:
linear-gradient(to right, #667eea, #764ba2);ð        color: white;ð        padding:
14px;ð        border: none;ð        border-radius: 8px;ð        font-size: 18px;ð
        cursor: pointer;ð        font-weight: bold;ð        text-transform: uppercase;ð
        letter-spacing: 1px;ð        transition: all 0.3s ease-in-out;ð        margin-top:
10px;ð    }ð    .btn:hover {ð        background: linear-gradient(to right,
#764ba2, #667eea);ð        transform: scale(1.05);ð        box-shadow: 0px 8px 20px
rgba(102, 126, 234, 0.5);ð    }ð    .switch {ð        text-align: center;ð
margin-top: 15px;ð        color: blue;ð        cursor: pointer;ð        text-
decoration: underline;ð    }ð    .forgot {ð        font-size: 14px;ð        text-
decoration: none;ð        color: #333;ð    }ð    .forgot:hover {ð        text-
decoration: underline;ð    }ð    .referral-section {ð        text-align: left;ð
margin-bottom: 10px;ð    }ð    @media (max-width: 768px) {ð
        .container {ð            flex-direction: column;ð            height: auto;ð        }ð
        .image-section {ð            height: 200px;ð            width: 100%;ð        }ð
        .form-section {ð            width: 100%;ð            padding: 30px;ð        }ð
        .form-box {ð            max-width: 100%;ð        }ð    }ð </style>ðð/head>ð
<body>ð <div class="container">ð    <div class="image-section"></div>ð    <div
class="form-section">ð        <div class="form-box active" id="signInBox">ð
            <h2>Sign In</h2>ð            <form id="signInForm">ð                <div
class="input-group">ð                    <label for="userid">Username</label>ð
                    <input type="text" id="userid" name="userid" placeholder="Enter
Username, Email, or Phone" required>ð                    <div class="error"
id="useridError"></div>ð                </div>ð                <div class="input-group">ð
                    <label for="signinPassword">Password</label>ð                    <input
type="password" id="signinPassword" name="password" placeholder="Enter
Password" required>ð                    <div class="error" id="passwordError"></div>ð
                </div>ð                <div class="remember-me">ð                    <div>ð
                    <input type="checkbox" id="rememberMe">ð                </div>ð            </div>ð        </div>ð    </div>ð </body>
```



```

                                <label
for="rememberMe">Remember Me</label>ð                                </div>ð                                <a href="#"
class="forgot">Forgot Password?</a>ð                                </div>ð                                <button
type="submit" class="btn">Sign In</button>ð                                <div class="switch"
onclick="showForm('signup')">Don't have an account? Sign Up</div>ð                                </
form>ð                                </div>ð                                <div class="form-box" id="signupBox">ð                                <h2>Sign
Up</h2>ð                                <form id="signupForm">ð                                <div class="input-group">ð
                                <label for="userName">Username</label>ð                                <input
type="text" id="userName" name="username" required placeholder="User Name">ð
                                <div class="error" id="userIdError"></div>ð                                </div>ð
                                <div class="input-group">ð                                <label for="email">Email</
label>ð                                <input type="email" id="email" name="email" required
placeholder="Email">ð                                <div class="error" id="emailError"></div>ð
                                </div>ð                                <div class="input-group">ð                                <label
for="phone">Phone Number</label>ð                                <input type="tel" id="phone"
name="phone" required placeholder="Phone Number">ð                                <div
class="error" id="phoneError"></div>ð                                </div>ð                                <div
class="input-group">ð                                <label for="tradingId">Trading View ID</label>ð
                                <input type="text" id="tradingId" name="tradingId" required
placeholder="Trading View ID">ð                                </div>ð                                <div class="input-
group">ð                                <label for="signupPassword">Password</label>ð
                                <input type="password" id="signupPassword" name="password" required
placeholder="Password">ð                                <div class="error" id="passwordError"></
div>ð                                </div>ð                                <div class="input-group">ð                                <label
for="confirmPassword">Repeat Password</label>ð                                <input
type="password" id="confirmPassword" name="confirmPassword" required
placeholder="Confirm Password">ð                                <div class="error"
id="confirmPasswordError"></div>ð                                </div>ð                                <div
class="referral-section">ð                                <input type="checkbox"
id="referralCheckbox" onchange="toggleReferral()">ð                                <label
for="referralCheckbox">Have a Referral ID?</label>ð                                </div>ð
                                <div class="input-group" id="referralId" style="display: none;">ð
                                <label for="referral">Referral ID</label>ð                                <input type="text"
id="referral" name="referral" placeholder="Referral ID">ð                                </div>ð
                                <button type="submit" class="btn">Sign Up</button>ð                                <div
class="switch" onclick="showForm('signin')">Already have an account? Sign In</
div>ð                                </form>ð                                </div>ð                                </div>ð                                </div>ð                                <script>ð                                function
showForm(type) {ð                                const signin = document.getElementById("signinBox");ð
                                const signup = document.getElementById("signupBox");ð                                if (type ===
"signup") {ð                                signin.classList.remove("active");ð
                                signup.classList.add("active");ð                                } else {ð
                                signup.classList.remove("active");ð                                signin.classList.add("active");ð
                                }ð                                }ð                                function toggleReferral() {ð                                const referralInput =
document.getElementById("referralId");ð                                referralInput.style.display =
document.getElementById("referralCheckbox").checked ? "block" : "none";ð                                }ð
                                </script>ð                                </body>ð                                </html>ð

```

demo\src\main\resources\static\signin4.html

```
<!DOCTYPE html>ðhtml lang="en">ðhead>ð <meta charset="UTF-8">ð <meta
name="viewport" content="width=device-width, initial-scale=1.0">ð
<title>Creative Login Page</title>ð <style>ð    * {ð        margin: 0;ð
padding: 0;ð        box-sizing: border-box;ð        font-family: "Segoe UI",
Tahoma, Geneva, Verdana, sans-serif;ð    }ð    ð    body {ð        display: flex;ð
        justify-content: center;ð        align-items: center;ð        height: 100vh;ð
        background-color: #6c3bdf;ð        margin: 0;ð    }ð    ð    .container {ð
        display: flex;ð        width: 800px;ð        height: 500px;ð        background:
white;ð        border-radius: 10px;ð        overflow: hidden;ð        box-shadow: 0
10px 25px rgba(0, 0, 0, 0.1);ð    }ð    ð    .image-section {ð        flex: 1;ð
padding: 20px;ð        display: flex;ð        justify-content: center;ð        align-
items: center;ð        position: relative;ð    }ð    ð    .form-section {ð
flex: 1;ð        padding: 40px;ð        display: flex;ð        flex-direction: column;ð
        justify-content: center;ð    }ð    ð    .header {ð        display: flex;ð
justify-content: space-between;ð        align-items: center;ð        margin-bottom:
30px;ð    }ð    ð    .logo {ð        font-weight: bold;ð        font-size: 24px;ð
color: #6c3bdf;ð    }ð    ð    .signup-link {ð        color: #6c3bdf;ð        text-
decoration: none;ð        font-size: 14px;ð    }ð    ð    h1 {ð        font-size: 28px;ð
margin-bottom: 5px;ð        color: #333;ð    }ð    ð    .subtitle {ð
color: #888;ð        margin-bottom: 30px;ð        font-size: 14px;ð    }ð    ð
        .input-field {ð        margin-bottom: 20px;ð        width: 100%;ð    }ð    ð
input {ð        width: 100%;ð        padding: 15px;ð        border: 1px solid #ddd;ð
        border-radius: 6px;ð        font-size: 14px;ð        outline: none;ð    }ð    ð
        .input-field input:focus {ð        border-color: #6c3bdf;ð    }ð    ð    .login-
btn {ð        width: 100%;ð        padding: 15px;ð        background-color: #6c3bdf;ð
        color: white;ð        border: none;ð        border-radius: 30px;ð        font-
size: 16px;ð        font-weight: bold;ð        cursor: pointer;ð        margin-bottom:
20px;ð        text-transform: uppercase;ð    }ð    ð    .forgot-password {ð
text-align: right;ð        color: #888;ð        font-size: 14px;ð        text-
decoration: none;ð        margin-bottom: 20px;ð    }ð    ð    .plant {ð
position: absolute;ð        bottom: 10px;ð        left: 10px;ð    }ð</style>ð/head>ð
<body>ð<div class="container">ð    <div class="image-section">ð        ð
<div class="plant">ð        ð    </div>ð    </div>ð    ð    <div class="form-section">ð
<div class="header">ð        <div class="logo">Creative</div>ð        <a
href="#" class="signup-link">New User? Sign Up</a>ð    </div>ð    ð
<h1>Welcome Back!</h1>ð        <div class="subtitle">Login to continue</div>ð
ð        <div class="input-field">ð            <input type="email"
placeholder="username11@gmail.com">ð        </div>ð        ð        <div class="input-
field">ð            <input type="password" placeholder="Enter Password">ð        </
div>ð        ð        <button class="login-btn">LOGIN</button>ð        ð        <div
style="text-align: right;">ð            <a href="#" class="forgot-password">FORGOT
PASSWORD?</a>ð        </div>ð    </div>ð</div>ð</body>ð</html>
```

demo\src\main\resources\static\signup.html

```
<title>Signup Page</title>ð <div class="container">ð <div class="text-
overlay">ð <h2>Start Your Trading Journey with Confidence</h2>ð
<p>Trade Smart. Grow Fast.</p>ð </div>ð <div class="image-
section">ð ð </div>ð
<div class="form-section">ð <div class="form-box">ð <h2>Sign up</
h2>ð <form id="signupForm">ð <div class="input-group">ð
<label for="username">Username</label>ð <input type="text"
class="input-style" id="userName" name="username" required placeholder="User
Name">ð <div class="error" id="userIdError"></div>ð </div>ð
<div class="input-group">ð <label for="email">Email</label>ð
<input type="email" id="email" class="input-style" name="email"
required placeholder="Email">ð <div class="error" id="emailError"></
div>ð </div>ð <div class="input-group">ð <label
for="phone">Phone Number</label>ð <input type="tel" id="phone"
class="input-style" name="phone" required placeholder="Phone Number">ð
<div class="error" id="phoneError"></div>ð </div>ð
<div class="input-group">ð <label for="tradingId">Trading View ID</
label>ð <input type="text" id="tradingId" class="input-style"
name="tradingId" required placeholder="Trading View ID">ð </div>ð
<div class="input-group">ð <label for="password">Password</label>ð
<input type="password" id="password" class="input-style"
name="password" required placeholder="Password">ð <div
class="error" id="passwordError"></div>ð </div>ð <div
class="input-group">ð <label for="confirmPassword">Repeat Password</
label>ð <input type="password" class="input-style"
id="confirmPassword" name="confirmPassword" required placeholder="Confirm
Password">ð <div class="error" id="confirmPasswordError"></div>ð
</div>ð <div class="referral-section">ð <input
type="checkbox" id="referralCheckbox">ð <label
for="referralCheckbox">Have a Referral ID?</label>ð </div>ð <div
class="input-group" id="referralId" style="display: none;">ð <label
for="referral">Referral ID</label>ð <input type="text" class="input-
style" id="referral" name="referral" placeholder="Referral ID">ð </div>ð
<button type="submit" class="btn">Sign Up</button>ð </form>ð <div
class="account">ð <h4>Already have an account? <a class="signin"
href="signin.html">Sign in</a></h4>ð </div>ð </div>ð </div>ð
<script>ð // Show/Hide Referral ID inputð
document.getElementById("referralCheckbox").addEventListener("change",
function () {ð let referralInput =
document.getElementById("referralId");ð referralInput.style.display
= this.checked ? "block" : "none";ð });ð
document.getElementById("signupForm").addEventListener("submit", async
function (event) {ð event.preventDefault(); // Prevent default form
submissionð let userName =
document.getElementById("userName").value.trim();ð let email =
document.getElementById("email").value.trim();ð let phone =
document.getElementById("phone").value.trim();ð let tradingId =
document.getElementById("tradingId").value.trim();ð let password =
document.getElementById("password").value.trim();ð let
confirmPassword = document.getElementById("confirmPassword").value.trim();ð
let referral = document.getElementById("referral").value.trim()
|| null;ð // Validation Patternsð let usernamePattern = /
^(?!\\s)(?!.*\\s{2,})[a-zA-Z0-9 ]+(?!\\s)$/;ð let emailPattern = /^[a-
zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\\.[a-zA-Z]{2,4}$/;ð let phonePattern
= /^[6-9]\\d{9}$/;ð let passwordPattern = /^(?=.*[A-Z])(?=.*[a-z])(?
=.*\\d)(?=.*[@$!%*?&])[A-Za-z\\d$!%*?&]{8,}$/;ð let isValid = true;ð
// Clear previous errorsð
document.getElementById("userIdError").textContent = "";ð
document.getElementById("emailError").textContent = "";ð
```

```

document.getElementById("phoneError").textContent = "";
document.getElementById("passwordError").textContent = "";
document.getElementById("confirmPasswordError").textContent = "";
// Validate username
if (!usernamePattern.test(userName)) {
document.getElementById("userIdError").textContent = "Invalid username!";
isValid = false;
} // Validate email
if (!emailPattern.test(email)) {
document.getElementById("emailError").textContent = "Invalid Email Format!";
isValid = false;
} // Validate phone number
if (!phonePattern.test(phone)) {
document.getElementById("phoneError").textContent = "Invalid Phone Number!";
isValid = false;
} // Validate password
if (!passwordPattern.test(password)) {
document.getElementById("passwordError").textContent = "Password must be 8+ characters with 1 uppercase, 1 lowercase, 1 number & 1 special character.";
isValid = false;
} // Confirm password
if (password !== confirmPassword) {
document.getElementById("confirmPasswordError").textContent = "Passwords do not match!";
isValid = false;
} if (!isValid) return;
// Send data to Spring Boot Backend
let requestData = {
  userName: userName,
  email: email,
  phone: phone,
  tradingId: tradingId,
  password: password,
  referralId: referralId
};
try {
let response = await
fetch("http://localhost:8080/api/signup", {
  method: "POST",
  headers: { "Content-Type": "application/json" },
  body: JSON.stringify(requestData)
});
let data = await response.json();
if (response.ok) {
// ' Show user details
in alert
alert(`Signup Successful! \n\nUsername: ${data.userName} \nUser ID: ${data.userId} \nReferral Code: ${data.referralCode}`);
window.location.href = "signin.html";
} else {
document.getElementById("emailError").textContent = data.message || "Signup failed!";
} catch (error) {
console.error("Error:", error);
alert("An error occurred while signing up. Please try again.");
}
}
</script>
<style>
* {
margin: 0;
padding: 0;
box-sizing: border-box;
font-family: "Poppins", sans-serif;
}
body {
height: 150vh;
display: flex;
justify-content: center;
align-items: center;
background: white;
}
.container {
display: flex;
width: 100%;
max-width: 1500px;
height: 950px;
box-shadow: 0px 10px 30px rgba(0, 0, 0, 0.1);
border-radius: 15px;
background-color: #f5f5f5;
background-image: linear-gradient(30deg, #ecec 12%, transparent 12.5%, transparent 87%, #ecec 87.5%, #ecec), linear-gradient(150deg, #ecec 12%, transparent 12.5%, transparent 87%, #ecec 87.5%, #ecec), linear-gradient(30deg, #ecec 12%, transparent 12.5%, transparent 87%, #ecec 87.5%, #ecec), linear-gradient(150deg, #ecec 12%, transparent 12.5%, transparent 87%, #ecec 87.5%, #ecec), linear-gradient(60deg, #ddd 25%, transparent 25.5%, transparent 75%, #ddd 75%, #ddd), linear-gradient(60deg, #ddd 25%, transparent 25.5%, transparent 75%, #ddd 75%, #ddd);
background-size: 40px 70px;
background-position: 0 0, 0 0, 20px 35px, 20px 35px, 0 0, 20px 35px;
}
.image-section {
flex: 1;
width: 65%;
background: white;

```

```
margin-top: 60px;D margin-left: 60px;D
margin-bottom: 30px;D D border-bottom-left-radius: 5%;D
border-top-left-radius: 5%;D box-shadow: -5px 0 10px -2px rgba(0,
0, 0, 0.4);D }D .image{D height: 420px;D
width: 48%;D justify-items: center;D position: absolute;D
top: 70%;D left: 28%;D transform:
translate(-50%, -50%);D D }D .form-section {D
width: 45%;D D display: flex;D align-items: center;D
justify-content: center;D background: #fff;D
backdrop-filter: blur(10px);D margin-top: 60px;D margin-
right: 60px;D margin-bottom: 30px;D D border-top-right-
radius: 5%;D border-bottom-right-radius: 5%;D box-shadow: 5px 0
4px -2px rgba(0, 0, 0, 0.3);D }D .form-box {D D
background: rgba(255, 255, 255, 0.2);D padding: 50px;D
width: 100%;D height: 100%;D justify-content: center;D
D text-align: center;D D color:
white;D D border-top-right-radius: 5%;D border-bottom-right-
radius: 5%;D }D D h2{D font-family:
'Times New Roman', Times, serif;D text-align: center;D
font-size: 40px;D font-weight: bold;D margin-bottom: 20px;D
color: black;D text-shadow: 0px 4px 10px rgba(0, 0, 0,
0.1);D D h4{D font-family: 'Franklin
Gothic Medium', 'Arial Narrow', Arial, sans-serif;D color: black;D
font-size: 19px;D font-weight: bold;D padding-
bottom: 10px;D }D D .input-group{D margin-
bottom: 15px;D text-align: left;D }D label {D
font-family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida
Sans Unicode', Geneva, Verdana, sans-serif;D font-weight: bold;D
display: block;D margin-bottom: 5px;D font-
size: 14px;D color: black;D letter-spacing: 1px;D }D
input {D font-family: 'Gill Sans', 'Gill Sans MT', Calibri,
'Trebuchet MS', sans-serif;D width: 100%;D padding: 12px
15px;D border: 2px solid #ccc;D border-radius: 8px;D
font-size: 16px;D transition: 0.3 ease;D D
}D D .input-group input:hover{D border-color:
#667eea;D box-shadow: 0px 0px 10px rgba(102, 126, 234, 0.5);D
transform: scale(1.02);D }D D D
.error {D color: red;D font-size: 12px;D
margin-top: 5px;D }D .referral-section{D
display: flex;D align-items: center;D gap: 10px;D
margin-bottom: 10px;D }D D
input[type=checkbox]{D -webkit-appearance: checkbox;D
moz-appearance: checkbox;D appearance: checkbox;D
display: inline-block;D width: auto;D .btn {D
width: 100%;D background: linear-gradient(to right,
#667eea, #764ba2);D color: white;D padding: 14px;D
border: none;D border-radius: 8px;D font-
size: 18px;D cursor: pointer;D transition: all 0.3s ease-
in-out;D font-weight: bold;D D letter-spacing: 1px;D
box-shadow: 0px 5px 15px rgba(102, 126, 234, 0.3);D }D
D .btn:hover {D background: linear-gradient(to
right, #764ba2, #667eea);D transform: scale(1.05);D box-
shadow: 0px 8px 20px rgba(102, 126, 234, 0.5);D }D D
#referralId {D display: none;D }D D .account h4{D
padding-top: 10px;D font-size: 16px;D }D .signin{D
text-decoration: none;D font-size: 16px;D }D
.signin :hover{D text-decoration: underline;D color:
#667eea;D transition: scale(1.02);D }D D /*
Responsive */D @media (max-width: 900px) {D .main-container {D
flex-direction: column;D
```

```

width: 90%;
height: auto;
}
.container {
width: 100%;
}
.container {
height: 300px;
}
.overlay {
position: absolute;
top: 23%;
left: 27%;
transform:
translate(-50%, -50%);
text-align: center;
color: white;
padding:
20px;
border-radius: 10px;
}
text-overlay h2 {
color: black;
font-
weight: bold;
font-size: 27px;
text-shadow: 0 0 1px #141616, 0 0 1px
#131314, 0 0 1px #121414;
overflow: hidden;
opacity: 0;
transform:
translateX(30px);
animation: fadeSlideUp 1.2s ease forwards;
}
.text-
overlay p {
color: black;
font-weight: bold;
font-size: 22px;
text-
shadow: 0 0 1px #101111, 0 0 1px #000000, 0 0 1px #16191b;
opacity: 0;
transform: translateX(30px);
animation: fadeSlideUp 1.2s ease forwards;
}
@keyframes fadeSlideUp {
to {
opacity: 1;
transform: translateY(0);
}
}
</style>

```

demo\src\main\resources\static\user_dashboard.html

```
<!DOCTYPE html>ðhtml lang="en">ðhead>ð  <meta charset="UTF-8">ð  <meta
name="viewport" content="width=device-width, initial-scale=1.0">ð
<title>Trading Homepage</title>ð  <link rel="stylesheet" href="styles.css">ð
  <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>ð  <style>ð
    body {ð              margin: 0;ð              font-family: Arial, sans-
serif;ð              background: #0d0d0d;ð              color: #fff;ð
overflow-x: hidden;ð              scroll-behavior: smooth;ð              }ð
header {ð              display: flex;ð              justify-content: space-between;ð
              align-items: center;ð              padding: 20px;ð
background: rgba(0, 0, 0, 0.8);ð              position: fixed;ð              width:
100%;ð              top: 0;ð              left: 0;ð              animation: fadeIn 1s
ease-in-out;ð              z-index: 1000;ð              }ð              .logo {ð
font-size: 24px;ð              font-weight: bold;ð              color: #ff9800;ð
              }ð              nav ul {ð              list-style: none;ð              display:
flex;ð              gap: 20px;ð              }ð              nav ul li a {ð              text-
decoration: none;ð              color: white;ð              transition: color 0.3s;ð
              }ð              nav ul li a:hover {ð              color: #ff9800;ð              }ð
              .hero {ð              height: 100vh;ð              display: flex;ð
              align-items: center;ð              justify-content: center;ð
              background: url('trading-bg.jpg') no-repeat center center/cover;ð
              text-align: center;ð              animation: zoomIn 1.5s ease-in-out;ð
              }ð              .hero-content {ð              max-width: 600px;ð
animation: slideUp 1.5s ease-in-out;ð              }ð              .hero h1 {ð
font-size: 3rem;ð              margin-bottom: 10px;ð              text-transform:
uppercase;ð              }ð              .hero button {ð              padding: 10px 20px;ð
              font-size: 1.2rem;ð              border: none;ð              cursor:
pointer;ð              background: #ff9800;ð              color: black;ð
transition: background 0.3s;ð              }ð              .hero button:hover {ð
background: #e68900;ð              }ð              .features, .testimonials, .trading-graph
{ð              display: flex;ð              flex-direction: column;ð
align-items: center;ð              text-align: center;ð              padding: 80px
20px;ð              background: linear-gradient(to bottom, #1a1a1a, #0d0d0d);ð
              animation: fadeIn 2s ease-in-out;ð              }ð              .feature {ð
              max-width: 300px;ð              margin: 20px;ð              }ð
              .testimonial {ð              margin: 20px auto;ð              max-width:
400px;ð              font-style: italic;ð              }ð              @keyframes fadeIn {ð
              from { opacity: 0; }ð              to { opacity: 1; }ð              }ð
              @keyframes slideUp {ð              from { transform: translateY(50px);
opacity: 0; }ð              to { transform: translateY(0); opacity: 1; }ð              }ð
              @keyframes zoomIn {ð              from { transform: scale(0.8); opacity:
0; }ð              to { transform: scale(1); opacity: 1; }ð              }ð  </style>ð/
head>ðbody>ð  <header>ð    <div class="logo">TradingX</div>ð    <nav>ð
      <ul>ð        <li><a href="#hero">Home</a></li>ð
        <li><a href="#features">Markets</a></li>ð
        <li><a href="#testimonials">Insights</a></li>ð        <li><a
href="#trading-graph">Live Charts</a></li>ð        <li><a
href="#about">About</a></li>ð        <li><a href="#contact">Contact</
a></li>ð      </ul>ð    </nav>ð  </header>ð  ð  <section id="hero"
class="hero">ð    <div class="hero-content">ð      <h1>Trade Smarter,
Trade Faster</h1>ð      <p>Harness AI-driven trading strategies for
market dominance.</p>ð      <button>Get Started</button>ð    </div>ð
  </section>ð  ð  <section id="features" class="features">ð    <div
class="feature">ð      <h2>Live Market Analysis</h2>ð      <p>Stay
ahead with real-time charts and analytics.</p>ð    </div>ð    <div
class="feature">ð      <h2>AI-Powered Strategies</h2>ð      <p>Leverage machine learning for precise trading decisions.</p>ð    </div>ð
    <div class="feature">ð      <h2>Secure Transactions</h2>ð      <p>Experience top-tier security for all your trades.</p>ð    </div>ð
  </section>ð</body>
```

```

        </section>Đ    <section id="testimonials" class="testimonials">Đ
        <h2>What Traders Say</h2>Đ        <div class="testimonial">Đ
        <p>"TradingX revolutionized my trading game!"</p>Đ        <span>- Alex T.</
        span>Đ        </div>Đ        <div class="testimonial">Đ        <p>"Best AI
        tools for market analysis hands down."</p>Đ        <span>- Sarah K.</span>Đ
        </div>Đ    </section>Đ    <section id="trading-graph" class="trading-
        graph">Đ        <h2>Live Trading Graph</h2>Đ        <canvas id="tradingChart"></
        canvas>Đ    </section>Đ    Đ    <script>Đ        document.querySelectorAll('nav
        ul li a').forEach(anchor => {Đ        anchor.addEventListener('click',
        function(event) {Đ        event.preventDefault();Đ        const
        targetId = this.getAttribute('href').substring(1);Đ        const
        targetElement = document.getElementById(targetId);Đ        if
        (targetElement) {Đ        window.scrollTo({Đ
        top: targetElement.offsetTop - 50,Đ        behavior: 'smooth'Đ
        });Đ        });Đ        });Đ        Đ
    </script>Đ</body>Đ</html>

```


demo\src\main\resources\static\User_dashboard2.html

```
<!DOCTYPE html>ðhtml lang="en">ðhead>ð  <meta charset="UTF-8">ð  <meta
name="viewport" content="width=device-width, initial-scale=1.0">ð  <title>
Trading Platform</title>ð  <style>ð    /* Reset and Base Styles */ð
* {ð      margin: 0;ð      padding: 0;ð      box-sizing:
border-box;ð      font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-
serif;ð    }ð    ð    body {ð      background-color: #0a0e17;ð
      color: #e6e6e6;ð      overflow-x: hidden;ð    }ð    ð
    /* Header Styles */ð    header {ð      background: linear-
gradient(135deg, #131927, #1d2533);ð      padding: 1.5rem 0;ð
position: fixed;ð      width: 100%;ð      top: 0;ð      z-
index: 1000;ð      box-shadow: 0 4px 20px rgba(0, 0, 0, 0.3);ð
animation: headerSlideDown 0.8s ease-out;ð    }ð    ð    .header-
container {ð      max-width: 1200px;ð      margin: 0 auto;ð
      display: flex;ð      justify-content: space-between;ð
      align-items: center;ð      padding: 0 2rem;ð    }ð    ð
    .logo {ð      display: flex;ð      align-items: center;ð
    }ð    ð    .logo h1 {ð      color: #4deeea;ð
font-size: 1.8rem;ð      margin-left: 1rem;ð      font-weight: 700;ð
      animation: logoPulse 2s infinite alternate;ð    }ð    ð
    .logo-icon {ð      width: 40px;ð      height: 40px;ð
      background: linear-gradient(45deg, #4deeea, #4d8eea);ð
border-radius: 8px;ð      display: flex;ð      align-items: center;ð
      justify-content: center;ð      animation: logoRotate 5s
linear infinite;ð    }ð    ð    .logo-icon::before {ð
content: "¿";ð      color: #131927;ð      font-size: 24px;ð
      font-weight: bold;ð    }ð    ð    ð    nav ul {ð
display: flex;ð      list-style: none;ð    }ð    ð    nav ul li
{ð      margin-left: 2rem;ð      position: relative;ð    }ð
    ð    nav ul li a {ð      color: #e6e6e6;ð      text-
decoration: none;ð      font-size: 1rem;ð      font-weight: 500;ð
      transition: all 0.3s ease;ð      padding: 0.5rem 0;ð    }ð
    ð    nav ul li a:hover {ð      color: #4deeea;ð    }ð
    ð    nav ul li a::after {ð      content: '';ð
position: absolute;ð      bottom: -5px;ð      left: 0;ð
width: 0%;ð      height: 2px;ð      background-color: #4deeea;ð
      transition: width 0.3s ease;ð    }ð    ð    nav ul li
a:hover::after {ð      width: 100%;ð    }ð    ð    .cta-btn {ð
      background: linear-gradient(45deg, #4deeea, #4d8eea);ð
color: #131927;ð      border: none;ð      padding: 0.8rem 1.5rem;ð
      border-radius: 30px;ð      font-weight: 600;ð
cursor: pointer;ð      transition: all 0.3s ease;ð      animation:
ctaPulse 2s infinite alternate;ð    }ð    ð    .cta-btn:hover {ð
      transform: translateY(-3px);ð      box-shadow: 0 10px 20px
rgba(77, 238, 234, 0.3);ð    }ð    ð    /* Hero Section */ð
    .hero {ð      height: 100vh;ð      display: flex;ð
      align-items: center;ð      background: url("/api/
placeholder/1200/800") center/cover;ð      background-blend-mode:
overlay;ð      background-color: rgba(13, 17, 28, 0.85);ð
padding-top: 80px;ð      position: relative;ð      overflow: hidden;ð
    }ð    ð    .hero::before {ð      content: '';ð
position: absolute;ð      top: 0;ð      left: 0;ð      width:
100%;ð      height: 100%;ð      background: linear-gradient(135deg,
rgba(13, 17, 28, 0.9), rgba(13, 17, 28, 0.6));ð      z-index: 1;ð
    }ð    ð    .hero-content {ð      max-width: 1200px;ð
margin: 0 auto;ð      padding: 0 2rem;ð      display: flex;ð
      justify-content: space-between;ð      align-items: center;ð
      width: 100%;ð      position: relative;ð      z-index: 2;ð
    }ð    ð    .hero-text {ð      flex: 0 0 50%;ð
animation: heroTextSlide 1s ease-out;ð
```

```

        }@ @ .hero-text h2 {@
font-size: 3.5rem;@ font-weight: 700;@ margin-
bottom: 1.5rem;@ line-height: 1.2;@ color: #ffffff;@
}@ @ .hero-text h2 span {@ background: linear-
gradient(45deg, #4deeea, #4d8eea);@ -webkit-background-clip: text;@
background-clip: text;@ color: transparent;@
position: relative;@ display: inline-block;@ }@ @
.hero-text p {@ font-size: 1.1rem;@ line-height:
1.6;@ margin-bottom: 2rem;@ color: #bbbbbb;@ }@
@ @ .hero-btns {@ display: flex;@ gap: 1rem;@
}@ @ .hero-btns .cta-btn.secondary {@
background: transparent;@ border: 2px solid #4deeea;@
color: #4deeea;@ }@ @ .hero-btns .cta-btn.secondary:hover {@
background: rgba(77, 238, 234, 0.1);@ }@ @
.hero-image {@ flex: 0 0 45%;@ animation:
floatUpDown 4s ease-in-out infinite, heroImageFadeIn 1.5s ease-out;@ }@
@ @ .hero-image img {@ width: 100%;@ border-
radius: 10px;@ box-shadow: 0 20px 40px rgba(0, 0, 0, 0.4);@ }@
@ /* Features Section */@ .features {@ padding:
6rem 2rem;@ background: linear-gradient(0deg, #0d111c, #131927);@
}@ @ .section-header {@ text-align: center;@
margin-bottom: 4rem;@ }@ @ .section-header h3 {@
font-size: 2.5rem;@ margin-bottom: 1rem;@
position: relative;@ display: inline-block;@ color:
#ffffff;@ }@ @ .section-header h3::after {@
content: '';@ position: absolute;@ bottom: -10px;@
left: 50%;@ transform: translateX(-50%);@
width: 80px;@ height: 3px;@ background: linear-
gradient(45deg, #4deeea, #4d8eea);@ }@ @ .section-header p {@
max-width: 700px;@ margin: 0 auto;@ color:
#bbbbbb;@ font-size: 1.1rem;@ line-height: 1.6;@ }@
@ @ .features-container {@ max-width: 1200px;@
margin: 0 auto;@ display: grid;@ grid-
template-columns: repeat(auto-fit, minmax(300px, 1fr));@ gap: 2rem;@
}@ @ .feature-card {@ background: linear-
gradient(145deg, #131927, #1d2533);@ border-radius: 15px;@
padding: 2rem;@ box-shadow: 0 10px 30px rgba(0, 0, 0,
0.2);@ transition: all 0.4s ease;@ position: relative;@
overflow: hidden;@ z-index: 1;@ }@ @
.feature-card:hover {@ transform: translateY(-10px);@
box-shadow: 0 15px 40px rgba(77, 238, 234, 0.15);@ }@ @
.feature-card::before {@ content: '';@ position:
absolute;@ top: 0;@ left: 0;@ width: 4px;@
height: 0;@ background: linear-gradient(45deg, #4deeea,
#4d8eea);@ transition: height 0.4s ease;@ z-index: -1;@
}@ @ .feature-card:hover::before {@ height:
100%;@ }@ @ .feature-icon {@ width: 60px;@
height: 60px;@ border-radius: 50%;@
background: rgba(77, 238, 234, 0.1);@ display: flex;@
align-items: center;@ justify-content: center;@ margin-
bottom: 1.5rem;@ font-size: 1.8rem;@ color: #4deeea;@
}@ @ .feature-card h4 {@ font-size: 1.5rem;@
margin-bottom: 1rem;@ color: #ffffff;@ }@ @
.feature-card p {@ color: #bbbbbb;@ line-height:
1.6;@ }@ @ /* Trading Charts Section */@ .trading-
charts {@ padding: 6rem 2rem;@ background: #0a0e17;@
position: relative;@ overflow: hidden;@ }@ @
.trading-charts::before {@ content: '';@
position: absolute;@ top: 0;@ left: 0;@ width:
100%;@

```

```

        height: 100%;
        background: url("/api/placeholder/1200/800") center/cover;
        opacity: 0.05;
        z-index: 0;
    }
    .charts-container {
        max-width: 1200px;
        margin: 0 auto;
        position: relative;
        z-index: 1;
    }
    .chart-tabs {
        display: flex;
        justify-content: center;
        gap: 1rem;
        margin-bottom: 2rem;
    }
    .chart-tab {
        padding: 0.8rem 1.5rem;
        background: linear-gradient(145deg, #131927, #1d2533);
        border: none;
        border-radius: 30px;
        color: #bbbbbb;
        cursor: pointer;
        transition: all 0.3s ease;
    }
    .chart-tab.active, .chart-tab:hover {
        background: linear-gradient(45deg, #4deeea, #4d8eea);
        color: #131927;
        font-weight: 600;
    }
    .chart-display {
        background: linear-gradient(145deg, #131927, #1d2533);
        border-radius: 15px;
        padding: 2rem;
        box-shadow: 0 10px 30px rgba(0, 0, 0, 0.2);
        height: 500px;
        position: relative;
        overflow: hidden;
        animation: chartAnimate 1s ease-out;
    }
    .chart-display img {
        width: 100%;
        height: 100%;
        object-fit: cover;
        border-radius: 10px;
    }
    .chart-stats {
        display: grid;
        grid-template-columns: repeat(4, 1fr);
        gap: 1.5rem;
        margin-top: 2rem;
    }
    .stat-card {
        background: linear-gradient(145deg, #131927, #1d2533);
        border-radius: 10px;
        padding: 1.5rem;
        text-align: center;
        box-shadow: 0 5px 15px rgba(0, 0, 0, 0.2);
        animation: fadeInUp 0.8s ease-out;
    }
    .stat-card h5 {
        font-size: 1.1rem;
        color: #bbbbbb;
        margin-bottom: 0.8rem;
    }
    .stat-card .value {
        font-size: 1.8rem;
        font-weight: 700;
        color: #ffffff;
    }
    .stat-card .trend {
        display: flex;
        align-items: center;
        justify-content: center;
        margin-top: 0.5rem;
        font-size: 0.9rem;
    }
    .stat-card .trend.up {
        color: #4dfa7b;
    }
    .stat-card .trend.down {
        color: #fa4d4d;
    }
    /* Testimonials Section */
    .testimonials {
        padding: 6rem 2rem;
        background: linear-gradient(0deg, #131927, #0d111c);
    }
    .testimonials-container {
        max-width: 1200px;
        margin: 0 auto;
    }
    .testimonial-slider {
        display: flex;
        overflow-x: auto;
        scroll-snap-type: x mandatory;
        -ms-overflow-style: none;
        gap: 2rem;
        padding: 1rem 0;
    }
    .testimonial-slider::-webkit-scrollbar {
        display: none;
    }
    .testimonial-card {
        flex: 0 0 350px;
        scroll-snap-align: start;
        background: linear-gradient(145deg, #131927, #1d2533);
        border-radius: 15px;
        padding: 2rem;
        box-shadow: 0 10px 30px rgba(0, 0, 0, 0.2);
        position: relative;
        animation: fadeInRight 1s ease-out;
    }
    .testimonial-card::before {
        position: absolute;
        top: 10px;
        left: 20px;
        font-size: 6rem;
        color: rgba(77, 238, 234, 0.1);
        line-height: 1;
    }
    .testimonial-content {
        position: relative;
        z-index: 1;
        font-style: italic;
        color: #bbbbbb;
        line-height: 1.6;
        margin-bottom: 1.5rem;
    }
    .testimonial-author {
        display: flex;
        align-items: center;
        width: 50px;
        height: 50px;
        border-radius: 50%;
        overflow: hidden;
        margin-right: 1rem;
    }
    .author-avatar {
        width: 50%;
        height: 50%;
        margin-right: 1rem;
    }

```

```

        .author-avatar img {
            width: 100%;
            height: 100%;
            object-fit: cover;
        }
        .author-info h5 {
            font-size: 1.1rem;
            color: #ffffff;
            margin-bottom: 0.2rem;
        }
        .author-info p {
            color: #bbbbbb;
            font-size: 0.9rem;
        }
        /* CTA Section */
        .cta-section {
            padding: 5rem 2rem;
            background: linear-gradient(135deg, #0a0e17, #131927);
            text-align: center;
            position: relative;
            overflow: hidden;
        }
        .cta-section::before {
            content: '';
            position: absolute;
            top: -100px;
            right: -100px;
            width: 300px;
            height: 300px;
            border-radius: 50%;
            background: rgba(77, 238, 234, 0.05);
            z-index: 0;
            animation: floatAroundLarge 15s linear infinite;
        }
        .cta-section::after {
            content: '';
            position: absolute;
            bottom: -80px;
            left: -80px;
            width: 200px;
            height: 200px;
            border-radius: 50%;
            background: rgba(77, 142, 234, 0.05);
            z-index: 0;
            animation: floatAroundSmall 10s linear infinite reverse;
        }
        .cta-container {
            max-width: 800px;
            margin: 0 auto;
            position: relative;
            z-index: 1;
        }
        .cta-container h3 {
            font-size: 2.5rem;
            margin-bottom: 1.5rem;
            color: #ffffff;
        }
        .cta-container p {
            font-size: 1.1rem;
            line-height: 1.6;
            color: #bbbbbb;
            margin-bottom: 2rem;
        }
        .cta-form {
            display: flex;
            max-width: 500px;
            margin: 0 auto;
            position: relative;
        }
        .cta-input {
            flex: 1;
            height: 60px;
            border-radius: 30px;
            border: none;
            padding: 0 2rem;
            background: linear-gradient(145deg, #131927, #1d2533);
            color: #ffffff;
            font-size: 1rem;
            box-shadow: 0 5px 15px rgba(0, 0, 0, 0.2);
        }
        .cta-input:focus {
            box-shadow: 0 5px 20px rgba(77, 238, 234, 0.2);
            outline: none;
        }
        .cta-btn-submit {
            position: absolute;
            right: 5px;
            top: 5px;
            height: 50px;
            border-radius: 25px;
            background: linear-gradient(45deg, #4deeea, #4d8eea);
            color: #131927;
            border: none;
            padding: 0 2rem;
            font-weight: 600;
            cursor: pointer;
            transition: all 0.3s ease;
        }
        .cta-btn-submit:hover {
            transform: translateX(3px);
            box-shadow: 0 5px 15px rgba(77, 238, 234, 0.3);
        }
        /* Footer */
        footer {
            background: #0a0e17;
            padding: 4rem 2rem 2rem;
            color: #bbbbbb;
        }
        .footer-container {
            max-width: 1200px;
            margin: 0 auto;
        }
        .footer-top {
            display: grid;
            grid-template-columns: repeat(auto-fit, minmax(200px, 1fr));
            gap: 2rem;
            margin-bottom: 3rem;
        }
        .footer-logo {
            grid-column: 1 / span 2;
        }
        .footer-logo .logo {
            margin-bottom: 1.5rem;
        }
        .footer-about {
            max-width: 300px;
            line-height: 1.6;
            font-size: 0.95rem;
        }
        .footer-links h4 {
            font-size: 1.2rem;
            color: #ffffff;
            margin-bottom: 1.5rem;
            position: relative;
            display: inline-block;
        }
        .footer-links h4::after {
            content: '';
            position: absolute;
            bottom: -8px;
            left: 0;
            width: 30px;
            height: 2px;
            background: linear-gradient(45deg, #4deeea, #4d8eea);
        }
        .footer-links ul {
            list-style: none;
            margin-bottom: 0.8rem;
        }
        .footer-links ul li {
            margin-bottom: 0.8rem;
        }
        .footer-links ul li a {
            color: #bbbbbb;
            text-decoration: none;
        }

```

```

        font-size: 0.95rem;D        }D        D        .footer-links ul li
a:hover {D        color: #4deeea;D        padding-left: 5px;D        }D
        D        .footer-bottom {D        border-top: 1px solid rgba(255,
255, 255, 0.1);D        padding-top: 2rem;D        display: flex;D
        justify-content: space-between;D        align-items: center;D
        flex-wrap: wrap;D        gap: 1rem;D        }D        D
        .footer-bottom p {D        font-size: 0.9rem;D        }D        D
        .social-links {D        display: flex;D        gap: 1rem;D
        }D        D        .social-link {D        width: 40px;D
height: 40px;D        border-radius: 50%;D        background: linear-
gradient(145deg, #131927, #1d2533);D        display: flex;D
align-items: center;D        justify-content: center;D        color:
#bbbbbb;D        text-decoration: none;D        transition: all 0.3s
ease;D        font-size: 1.2rem;D        }D        D        .social-
link:hover {D        background: linear-gradient(45deg, #4deeea, #4d8eea);D
        color: #131927;D        transform: translateY(-3px);D        }D
        D        /* Animations */D        @keyframes headerSlideDown {D
        from {D        transform: translateY(-100%);D
        opacity: 0;D        }D        to {D
transform: translateY(0);D        opacity: 1;D        }D        }D
        D        @keyframes logoPulse {D        from {D
transform: scale(1);D        }D        to {D        transform:
scale(1.05);D        }D        }D        D        @keyframes logoRotate {D
        from {D        transform: rotate(0deg);D        }D
        to {D        transform: rotate(360deg);D        }D
        }D        D        @keyframes ctaPulse {D        from {D
        box-shadow: 0 0 0 rgba(77, 238, 234, 0);D        }D
        to {D        box-shadow: 0 0 15px rgba(77, 238, 234, 0.5);D
        }D        }D        D        @keyframes heroTextSlide {D
from {D        transform: translateX(-50px);D        opacity: 0;D
        }D        to {D        transform: translateX(0);D
        opacity: 1;D        }D        }D        D        @keyframes
heroImageFadeIn {D        from {D        transform:
translateX(50px);D        opacity: 0;D        }D        to {D
        transform: translateX(0);D        opacity: 1;D
        }D        }D        D        @keyframes floatUpDown {D        0%,
100% {D        transform: translateY(0);D        }D        50% {D
        transform: translateY(-15px);D        }D        }D        D
        @keyframes chartAnimate {D        from {D        opacity: 0;D
        transform: scale(0.98);D        }D        to {D
        opacity: 1;D        transform: scale(1);D        }D
        }D        D        @keyframes fadeInUp {D        from {D
        opacity: 0;D        transform: translateY(20px);D
        }D        to {D        opacity: 1;D
        }D        }D        D        @keyframes
fadeInRight {D        from {D        opacity: 0;D
transform: translateX(-20px);D        }D        to {D
        opacity: 1;D        transform: translateX(0);D        }D        }D
        D        @keyframes floatAroundLarge {D        0% {D
transform: translate(0, 0);D        }D        25% {D
transform: translate(-50px, 50px);D        }D        50% {D
transform: translate(-100px, 0);D        }D        75% {D
transform: translate(-50px, -50px);D        }D        100% {D
transform: translate(0, 0);D        }D        }D
        D        @keyframes floatAroundSmall {D        0% {D
transform: translate(0, 0);D        }D        25% {D
transform: translate(30px, -30px);D        }D        50% {D
transform: translate(60px, 0);D        }D        75% {D
transform: translate(30px, 30px);D        }D

```

```

100% {
  transform: translate(0, 0);
  /* Responsive Styles */
  @media (max-width: 992px) {
    .hero-content {
      flex-direction: column;
      text-align: center;
    }
    .hero-btns {
      justify-content: center;
    }
    .chart-stats {
      grid-template-columns: repeat(2, 1fr);
    }
    @media (max-width: 768px) {
      .hero-text h2 {
        font-size: 2.5rem;
      }
      .header-container {
        flex-direction: column;
        gap: 1rem;
        flex-wrap: wrap;
      }
      nav ul {
        justify-content: center;
      }
      nav ul li {
        margin: 0 1rem;
      }
      .footer-logo {
        grid-column: 1 / -1;
      }
      @media (max-width: 576px) {
        .hero-text h2 {
          font-size: 2rem;
        }
        .cta-form {
          flex-direction: column;
        }
        .cta-btn-submit {
          position: static;
          width: 100%;
        }
        .chart-stats {
          height: 50px;
        }
        .chart-stats {
          grid-template-columns: 1fr;
        }
        .footer-bottom {
          flex-direction: column;
          text-align: center;
        }
        .social-links {
          justify-content: center;
        }
      }
    }
  }
}
</style></head><body>
<!-- Header Section -->
<header>
  <div class="header-container">
    <div class="logo">
      <h1>TradePro</h1>
    </div>
    <nav>
      <ul>
        <li><a href="#home">Home</a></li>
        <li><a href="#features">Features</a></li>
        <li><a href="#markets">Markets</a></li>
        <li><a href="#pricing">Pricing</a></li>
        <li><a href="#about">About</a></li>
      </ul>
    </nav>
    <button class="cta-btn">Start Trading</button>
  </div>
</header>
<!-- Hero Section -->
<section class="hero" id="home">
  <div class="hero-content">
    <div class="hero-text">
      <h2>Trade <span>Smarter</span>, Not Harder</h2>
      <p>Experience a new generation of trading with TradePro's advanced platform. Real-time analysis, AI-powered insights, and commission-free trades to maximize your profits.</p>
    </div>
    <div class="hero-btns">
      <button class="cta-btn">Get Started</button>
      <button class="cta-btn secondary">Learn More</button>
    </div>
  </div>
  
</div>
</section>
<!-- Features Section -->
<section class="features" id="features">
  <div class="section-header">
    <h3>Powerful Features</h3>
  </div>
  <p>Discover the tools that make TradePro the preferred choice of professional traders around the world.</p>
  <div class="features-container">
    <div class="feature-icon">
      <h4>Real-Time Analytics</h4>
      <p>Access comprehensive market analysis and real-time data from global markets to make informed trading decisions.</p>
    </div>
    <div class="feature-card">
      <div class="feature-icon">
        <h4>AI Trading Assistant</h4>
        <p>Our advanced AI analyzes patterns and trends to suggest optimal trading strategies tailored to your risk profile.</p>
      </div>
    </div>
  </div>
</section>

```

```


#### Multi-Platform Access



Trade on the go with our seamless experience across desktop, web, and
mobile applications with real-time synchronization.



#### Bank-Level Security



Your assets are protected with industry-leading encryption, two-factor
authentication, and regular security audits.



#### Advanced Charting



Over 100+
technical indicators and drawing tools to analyze market trends and make
precise predictions.



#### Commission-Free



Trade stocks, ETFs, and
cryptocurrencies without any commission fees. Only pay for premium features
you choose.


```

demo\src\main\resources\static\user_dashboard3.html

```
<!DOCTYPE html>ðhtml lang="en">ðhead>ð    <meta charset="UTF-8">ð
<title>User Dashboard</title>ð    ð    <link rel="stylesheet" href="https://
cdnjs.cloudflare.com/ajax/libs/font-awesome/6.5.0/css/all.min.css">ð
<script src="https://cdn.jsdelivr.net/npm/sockjs-client@1/dist/
sockjs.min.js"></script>ð <script src="https://cdn.jsdelivr.net/npm/
stompjs@2.3.3/lib/stomp.min.js"></script>ð <link rel="stylesheet"
href="user_dashboard3.css">ð/head>ðbody>ð    <header class="header">ð
<button class="hamburger" id="menuToggle">ð                <i class="fas fa-
bars"></i>ð                </button>ð                <div class="logo"></div>ð                <div class="header-
title">User Management System</div>ð                <div class="user-info-mini">ð
ð                <div class="user-data">ð                <span
id="headerUserName" class="username">Loading...</span>ð                <span
id="headerUserRole" class="name" style="font-size: 12px; color: black;
margin-left: 8px;">(Role)</span>ð                </div>ð                </div>ð    </header>ð
ð    <aside class="sidebar" id="sidebar">ð                <div class="sidebar-
header">ð                <div class="user-info">ð                <div class="user-
avatar" id="userAvatar">U</div>ð                <div>ð                <div
class="user-name" id="sidebarUserName">Loading...</div>ð
<div class="user-role" id="sidebarUserRole">Role</div>ð
<div class="user-status" id="userStatus">Online</div>ð                </div>ð
</div>ð                <ul class="nav-menu">ð
<li><a href="#" onclick="showModule('dashboardModule')"><i class="fas fa-
tachometer-alt"></i> Dashboard</a></li>ð                <li><a href="#"
onclick="showModule('userProfileModule')"><i class="fas fa-user"></i>
Profile</a></li>ð                <li><a href="#"
onclick="showModule('settingsModule')"><i class="fas fa-cog"></i> Settings</
a></li>ð                <li><a href="#"><i class="fas fa-chart-bar"></i>
Analytics</a></li>ð                <li><a href="#"
onclick="showModule('chatModule')"><i class="fas fa-comment-sms"></i> Chat</
a></li>ð                </ul>ð                <button id="logout" class="logout-btn">ð
<i class="fas fa-sign-out-alt"></i> Logoutð                </button>ð    </
aside>ð    ð    <main class="main-content" id="mainContent">ð                <div
id="dashboardModule" class="module" style="display: block;">ð                <div
class="dashboard-header">ð                <h1 class="dashboard-title">Dashboard</
h1>ð                <div class="breadcrumb">Home &gt; Dashboard</div>ð                </div>ð
ð                <div class="data-card fade-in">ð                <h2>Session
Information</h2>ð                <div id="sessionDataContainer">ð
<div class="loading" id="sessionLoading">Loading...</div>ð
<table class="session-table" id="sessionTable" style="display: none;">ð
    <thead>ð                <tr>ð
ð                <th>User ID</th>ð
<th>Username</th>ð
<th>Sign-in Time</th>ð                <th>Logout Time</th>ð
<th>Role</th>ð
<th>Status</th>ð                </tr>ð
    </thead>ð                <tbody>ð
ð                <tr>ð
ð                <td
id="userIdDisplay"></td>ð                <td
id="userNameDisplay"></td>ð
id="signinTimeDisplay"></td>ð
id="logoutTimeDisplay"></td>ð
id="roleDisplay"></td>ð                <td id="statusDisplay"></td>ð
    </tr>ð                </tbody>ð                </div>ð
ð                </div>ð                </div>ð                </div>ð    <!-- USER PROFILE
MODULE -->ð    <div id="userProfileModule" class="module" style="display: none;">ð
    <div class="dashboard-header">ð                <h1 class="dashboard-title">User
Profile</h1>ð                <div class="breadcrumb">Home &gt; Profile</div>ð    </div>ð
ð
```



```

        <div class="profile-card">ð        <div class="profile-photo">ð        ð        </div>ð
<div class="profile-details">ð        <div class="profile-row">ð        <span
class="profile-label">User ID:</span>ð        <span
id="profileUserId">Loading...</span>ð        </div>ð        <div class="profile-
row">ð        <span class="profile-label">Name:</span>ð        <span
id="profileUserName">Loading...</span>ð        </div>ð        <div
class="profile-row">ð        <span class="profile-label">Email:</span>ð
        <span id="profileEmail">Loading...</span>ð        </div>ð        <div
class="profile-row">ð        <span class="profile-label">Phone:</span>ð
        <span id="profilePhone">Loading...</span>ð        </div>ð        <div
class="profile-row">ð        <span class="profile-label">Role:</span>ð
        <span id="profileRole">Loading...</span>ð        </div>ð        <div
class="profile-row">ð        <span class="profile-label">Referral Code:</
span>ð        <span id="profileReferralCode">Loading...</span>ð        </div>ð
        <div class="profile-row">ð        <span class="profile-
label">Created At:</span>ð        <span id="profileCreatedAt">Loading...</
span>ð        </div>ð        </div>ð        </div>ð        </div>ð        ð        <div id="settingsModule"
class="module" style="display: none;">ð        <div class="dashboard-header">ð
        <h1 class="dashboard-title">User Settings</h1>ð        <div
class="breadcrumb">Home > Settings</div>ð        </div>ð        ð        <!-- Username
Section -->ð        <div class="settings-section">ð        <label>Username:</label>ð
        <span id="currentUsername" class="current-value">Loading...</span>ð
        <div id="usernameEdit" class="edit-section">ð        <input
type="text" id="usernameInput" placeholder="Enter new username" />ð
        <button class="save-button" onclick="saveUsername()">Save</button>ð
        <button class="cancel-button" onclick="cancelEdit('username')">Cancel</button>ð
        </div>ð        ð        </div>ð        <!-- Phone Section -->ð        <div
class="settings-section">ð        <label>Phone:</label>ð        <span
id="currentPhone" class="current-value">Loading...</span>ð        <div
id="phoneEdit" class="edit-section">ð        <input type="tel"
id="phoneInput" placeholder="Enter new phone number" />ð        <button
class="save-button" onclick="savePhone()">Save</button>ð        <button
class="cancel-button" onclick="cancelEdit('phone')">Cancel</button>ð        </
div>ð        ð        </div>ð        <!-- Single Edit Button -->ð        <div class="settings-
section">ð        <button class="edit-button" id="editProfileBtn"
onclick="editProfile()">Edit Profile</button>ð        </div>ð        <!-- Profile
Picture -->ð        <div class="settings-section">ð        <label>Change Profile
Picture:</label>ð        ð
        <input type="file" id="uploadInput" accept="image/*" />ð        </div>ð        <div
id="settingsStatus" class="status-msg"></div>ð        <div id="successPopup">ð
        <p id="successMessage"></p>ð        <button onclick="closePopup()">OK</button>ð
        </div>ð        </div>ð        ð        ð        ð        ð        ð        ð        <div id="chatModule" class="module"
style="display: none;">ð        <h2>Chat</h2>ð        ð        <!-- Chat Box to display
messages -->ð        <div class="chat-box" id="chatBox">ð        <!-- Messages will
be displayed here -->ð        </div>ð        <!-- Notification Area (Hidden initially)
-->ð        <div id="notificationArea" style="display: none; position: fixed; top:
20px; left: 50%; transform: translateX(-50%); background-color: #4caf50;
color: white; padding: 10px 20px; border-radius: 5px; cursor: pointer;">ð
        New message received!ð        </div>ð        ð        <!-- Loading indicator -->ð        <div
id="loadingIndicator" style="display: none;">Loading...</div>ð        ð        <!--
Message Input Section -->ð        <div class="chat-input-area">ð        <input
type="text" id="chatInput" placeholder="Type your message..." />ð        <button
onclick="sendChatMessage()">Send</button>ð        </div>ð        </div>ð        ð        </main>ð        ð
        ð        ð        ð        <script src="user_dashboard3.js"></script>ð        ð        </body>ð        </html>ð

```

demo\src\main\resources\static\user_dashboard4.html

```
<!DOCTYPE html>ðhtml lang="en">ðhead>ð  <meta charset="UTF-8">ð  <meta
name="viewport" content="width=device-width, initial-scale=1.0">ð
<title>User Dashboard</title>ð  <style>ð      :root {ð          --primary-
color: #4361ee;ð          --secondary-color: #3f37c9;ð          --success-
color: #4caf50;ð          --danger-color: #f44336;ð          --light-color:
#f5f5f5;ð          --dark-color: #333;ð          --sidebar-width: 250px;ð
          --header-height: 60px;ð          --box-shadow: 0 2px 10px
          rgba(0, 0, 0, 0.1);ð          --transition: all 0.3s ease;ð      }ð      *
{ð          margin: 0;ð          padding: 0;ð          box-sizing: border-
box;ð          font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;ð
      }ð      body {ð          background-color: #f8f9fa;ð
color: var(--dark-color);ð          position: relative;ð          min-
height: 100vh;ð      }ð      /* Header styles */ð      .header {ð
          background-color: white;ð          height: var(--header-height);ð
          padding: 0 20px;ð          display: flex;ð          align-
items: center;ð          justify-content: space-between;ð
position: fixed;ð          top: 0;ð          left: 0;ð          right: 0;ð
          z-index: 100;ð          box-shadow: var(--box-shadow);ð      }ð
      .header-title {ð          font-size: 1.2rem;ð          font-weight:
600;ð          color: var(--primary-color);ð      }ð      .hamburger {ð
          display: none;ð          font-size: 24px;ð          cursor:
pointer;ð          color: var(--dark-color);ð          background: none;ð
          border: none;ð      }ð      /* Sidebar styles */ð      .sidebar {ð
          background-color: white;ð          width:
var(--sidebar-width);ð          height: calc(100vh - var(--header-height));ð
          position: fixed;ð          top: var(--header-height);ð
          left: 0;ð          box-shadow: var(--box-shadow);ð          padding-top:
20px;ð          transition: var(--transition);ð          z-index: 99;ð
      }ð      .sidebar.collapsed {ð          left: -250px;ð      }ð
      .sidebar-header {ð          padding: 0 20px 20px;ð          border-
bottom: 1px solid #eee;ð          margin-bottom: 20px;ð      }ð      .user-
info {ð          display: flex;ð          align-items: center;ð      }ð
      .user-avatar {ð          width: 45px;ð          height: 45px;ð
          border-radius: 50%;ð          background-color: var(--primary-
color);ð          color: white;ð          display: flex;ð          align-
items: center;ð          justify-content: center;ð          font-weight:
bold;ð          margin-right: 12px;ð      }ð      .user-name {ð
font-weight: 600;ð          font-size: 1.1rem;ð          margin-bottom: 3px;ð
      }ð      .user-status {ð          font-size: 0.85rem;ð
color: var(--success-color);ð      }ð      .nav-menu {ð          list-
style: none;ð      }ð      .nav-menu li {ð          margin-bottom: 5px;ð
      }ð      .nav-menu a {ð          display: flex;ð          align-
items: center;ð          padding: 12px 20px;ð          color: var(--dark-
color);ð          text-decoration: none;ð          transition: var(--
transition);ð      }ð      .nav-menu a:hover, .nav-menu a.active {ð
          background-color: rgba(67, 97, 238, 0.1);ð          color: var(--
primary-color);ð      }ð      .nav-menu a i {ð          margin-right: 10px;ð
font-size: 1.2rem;ð      }ð      .logout-btn {ð          margin: 20px;ð
padding: 10px;ð          background-color: #f8f9fa;ð
color: var(--danger-color);ð          border: 1px solid #ddd;ð
border-radius: 5px;ð          cursor: pointer;ð          width: calc(100% -
40px);ð          font-weight: 600;ð          display: flex;ð
align-items: center;ð          justify-content: center;ð
transition: var(--transition);ð      }ð      .logout-btn:hover {ð
background-color: var(--danger-color);ð          color: white;ð      }ð
      /* Main content styles */ð      .main-content {ð          margin-
left: var(--sidebar-width);ð          margin-top: var(--header-height);ð
          padding: 20px;ð
```

```

        transition: var(--transition);@
        .main-content.expanded {@
            margin-left: 0;@
        }@
        .dashboard-header {@
            margin-bottom: 20px;@
        }@
        .dashboard-title {@
            font-size: 1.8rem;@
            font-weight: 600;@
            margin-bottom: 5px;@
            color: var(--dark-color);@
        }@
        .breadcrumb {@
            font-size: 0.9rem;@
            color: #666;@
            margin-bottom: 20px;@
        }@
        .data-card {@
            background-color: white;@
            border-radius: 8px;@
            box-shadow: var(--box-shadow);@
            padding: 20px;@
            margin-bottom: 20px;@
        }@
        .data-card h2 {@
            font-size: 1.2rem;@
            margin-bottom: 15px;@
            color: var(--dark-color);@
        }@
        .session-table {@
            border-bottom: 1px solid #eee;@
            padding-bottom: 10px;@
            width: 100%;@
            border-collapse: collapse;@
        }@
        .session-table th, .session-table td {@
            padding: 12px 15px;@
            text-align: left;@
            border-bottom: 1px solid #eee;@
        }@
        .session-table th {@
            font-weight: 600;@
            color: #666;@
            background-color: #f8f9fa;@
        }@
        .session-table tr:last-child td {@
            border-bottom: none;@
        }@
        .session-table .status {@
            padding: 5px 10px;@
            border-radius: 20px;@
            font-size: 0.85rem;@
            font-weight: 500;@
        }@
        .session-table .status.active {@
            background-color: rgba(76, 175, 80, 0.1);@
            color: var(--success-color);@
        }@
        .session-table .status.inactive {@
            background-color: rgba(244, 67, 54, 0.1);@
            color: var(--danger-color);@
        }@
        .loading {@
            display: flex;@
            justify-content: center;@
            align-items: center;@
            height: 100px;@
        }@
        .loading:after {@
            content: " ";@
            display: block;@
            width: 40px;@
            height: 40px;@
            border-radius: 50%;@
            border: 6px solid var(--primary-color);@
            border-color: var(--primary-color) transparent var(--primary-color) transparent;@
            animation: loader 1.2s linear infinite;@
        }@
        @keyframes loader {@
            0% {@
                transform: rotate(0deg);@
            }@
            100% {@
                transform: rotate(360deg);@
            }@
        }@
        /* Responsive styles */@
        @media screen and (max-width: 768px) {@
            .hamburger {@
                display: block;@
            }@
            .sidebar {@
                left: -250px;@
            }@
            .sidebar.active {@
                left: 0;@
            }@
            .main-content {@
                margin-left: 0;@
            }@
            .header-title {@
                margin-left: 20px;@
            }@
        }@
    </style>@
    <!-- Icon library for menu icons -->@
    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/@fortawesome/fontawesome-free@6.4.0/css/all.min.css">@</head>@<body>@
    <!-- Header -->@
    <header class="header">@
        <button class="hamburger" id="menuToggle">@
        <i class="fas fa-bars"></i>@
        </button>@
        <div class="header-title">User Management System</div>@
        <div class="user-info-mini">@
            <span id="headerUserName"></span>@
        </div>@
    </header>@
    <!-- Sidebar -->@
    <aside class="sidebar" id="sidebar">@
        <div class="sidebar-header">@
            <div class="user-info">@
                <div class="user-avatar" id="userAvatar"></div>@
                <div class="user-name" id="sidebarUserName">Loading...</div>@
            </div>@
            <div class="user-status" id="userStatus">Online</div>@
        </div>@
        <div class="nav-menu">@
            <li><a href="#" class="active"><i class="fas fa-tachometer-alt"></i> Dashboard</a></li>@
            <li><a href="#"><i class="fas fa-user"></i> Profile</a></li>@
            <li><a href="#"><i class="fas fa-cog"></i> Settings</a></li>@
            <li><a href="#"><i class="fas fa-bar-chart"></i> Analytics</a></li>@
            <li><a href="#"><i class="fas fa-bell"></i> Notifications</a></li>@
            <li><a href="#"><i class="fas fa-power-off"></i> Logout</a></li>@
        </ul>@
    </div>@
    </body>@
    </html>@

```

```

<i class="fas fa-sign-out-alt" style="margin-right: 10px;"></i> Logout</button></aside><!-- Main Content -->
<main class="main-content" id="mainContent">
  <div class="dashboard-header">
    <h1 class="dashboard-title">Dashboard</h1>
    <div class="breadcrumb">Home > Dashboard</div>
    <div class="data-card">
      <h2>Session Information</h2>
      <div id="sessionDataContainer">
        <div id="sessionLoading"></div>
        <table class="session-table" id="sessionTable" style="display: none;">
          <thead>
            <tr>
              <th>Sign-in Time</th>
              <th>User ID</th>
              <th>Email</th>
              <th>Logout Time</th>
              <th>Phone</th>
              <th>Status</th>
            </tr>
          </thead>
          <tbody>
            <tr>
              <td id="userIdDisplay"></td>
              <td id="signinTimeDisplay"></td>
              <td id="logoutTimeDisplay"></td>
              <td id="emailDisplay"></td>
              <td id="statusDisplay"></td>
            </tr>
          </tbody>
        </table>
      </div>
    </div>
  </main>
  <script>
    // DOM Elements
    const menuToggle = document.getElementById('menuToggle');
    const sidebar = document.getElementById('sidebar');
    const mainContent = document.getElementById('mainContent');
    const logoutBtn = document.getElementById('logout');
    const sessionLoading = document.getElementById('sessionLoading');
    const sessionTable = document.getElementById('sessionTable');
    menuToggle.addEventListener('click', () => {
      sidebar.classList.toggle('active');
      mainContent.classList.toggle('expanded');
    });
    // Format datetime for display
    function formatDate(dateTimeStr) {
      if (!dateTimeStr) return 'N/A';
      const date = new Date(dateTimeStr);
      return Intl.DateTimeFormat('en-US', {
        year: 'numeric',
        month: 'short',
        day: 'numeric',
        hour: '2-digit',
        minute: '2-digit',
        second: '2-digit'
      }).format(date);
    }
    // Check if user is logged in
    function checkUserLoggedIn() {
      const userId = sessionStorage.getItem("userId");
      const userName = sessionStorage.getItem("userName");
      // Display user info in sidebar and header
      document.getElementById("sidebarUserName").textContent = userName;
      document.getElementById("headerUserName").textContent = userName;
      document.getElementById("userAvatar").textContent = userName.charAt(0).toUpperCase();
      return true;
    }
    // Load dashboard data
    function loadDashboardData() {
      if (!checkUserLoggedIn()) return;
      const userId = sessionStorage.getItem("userId");
      // Show loading indicator
      sessionLoading.style.display = 'flex';
      sessionTable.style.display = 'none';
      // Fetch user session data
      fetch(`http://localhost:8080/api/user_sessions/${userId}`)
        .then(response => {
          if (!response.ok) {
            throw new Error('Network response was not ok');
          }
          return response.json();
        })
        .then(data => {
          // Hide loading indicator and show table
          sessionLoading.style.display = 'none';
          sessionTable.style.display = 'table';
          // Populate table with data

```

```

document.getElementById("userIdDisplay").textContent = data.userId || "N/A";
document.getElementById("signinTimeDisplay").textContent
= formatDateTime(data.signinTime);
document.getElementById("logoutTimeDisplay").textContent = data.logoutTime ?
formatDateTime(data.logoutTime) : "Active";
document.getElementById("emailDisplay").textContent = data.email || "N/A";
document.getElementById("phoneDisplay").textContent =
data.phone || "N/A";
const statusDisplay =
document.getElementById("statusDisplay");
if (data.isActive) {
statusDisplay.innerHTML = '<span
class="status active">Online</span>';
document.getElementById("userStatus").textContent = "Online";
document.getElementById("userStatus").style.color =
"var(--success-color)";
} else {
statusDisplay.innerHTML = '<span class="status inactive">Offline</span>';
document.getElementById("userStatus").textContent =
"Offline";
document.getElementById("userStatus").style.color = "var(--danger-color)";
}
.catch(error => {
console.error("Error loading dashboard data:", error);
sessionLoading.style.display = 'none';
alert("Failed to load dashboard data. Please try again.");
});
// Handle logout
logoutBtn.addEventListener('click',
function() {
const userId = sessionStorage.getItem("userId");
if (!userId) {
alert("No active
session found. Redirecting to login page...");
window.location.href = "signin.html";
return;
}
// Disable logout button to prevent multiple clicks
logoutBtn.disabled = true;
logoutBtn.innerHTML = '<i
class="fas fa-spinner fa-spin" style="margin-right: 10px;"></i> Logging
out...';
logoutBtn.addEventListener('click',
function() {
const userId = sessionStorage.getItem("userId");
// Disable logout button to prevent multiple clicks
logoutBtn.disabled = true;
logoutBtn.innerHTML = '<i class="fas fa-spinner fa-spin" style="margin-
right: 10px;"></i> Logging out...';
// Check if we have a userId to
send to the server
if (!userId) {
console.log("No userId found in
session storage");
// Just redirect to login since there's no session
to clear server-side
sessionStorage.clear();
window.location.href = "signin.html";
return;
}
// Send
logout request to server
fetch("http://localhost:8080/api/logout", {
method: "POST",
headers: {
"Content-Type":
"application/json"
},
body: JSON.stringify({ userId: userId })
})
.then(response => {
// Always clear local session data
regardless of server response
sessionStorage.clear();
// Process the response
if (!response.ok) {
console.warn("Server returned error during logout, but continuing with client
logout");
window.location.href = "signin.html";
return
null;
// Don't try to parse JSON if response is not OK
}
return
response.json();
})
.then(data => {
if (data) {
console.log("Logout response:", data);
// Even if
server says "no active session", we still want to logout locally
if (data.message.includes("No active session")) {
console.log("Server reports no active session, but proceeding with client
logout");
}
// Always redirect to login
page
window.location.href = "signin.html";
}
}
).catch(error => {
console.error("Logout error:", error);
// Even if
there's an error, clear session and redirect
sessionStorage.clear();
window.location.href = "signin.html";
}

```

```

    });});
Initialize dashboard on page load
document.addEventListener('DOMContentLoaded', function() {
loadDashboardData();
    }); // Refresh dashboard data every 30
seconds
    setInterval(loadDashboardData, 30000);
</script>
/body>
/html>
```

```

<title> Admin Signin Page</title><div class="text-
overlay"><div class="image-section">
</div><div class="form-section"><div class="form-box">
<h2>Welcome Back!</h2><p>Login to Continue</p>
<form id="adminSigninForm">
<div class="input-group">
<label for="adminid">Admin Username </label>
<input type="text" id="adminid" name="adminid"
placeholder="Enter Admin Username or Email" required >
<div class="error" id="adminidError"></div>
<div class="input-group">
<label
for="adminPassword">Password</label>
<input
type="password" id="adminPassword" name="adminPassword" placeholder="Enter
Password" required>
<div class="error"
id="adminPasswordError"></div>
<div
class="remember-me">
<input type="checkbox"
id="adminRememberMe">
<label
for="adminRememberMe">Remember Me</label>
<h5><a
href="forgot.html" class="forgot">Forgot Password?</a></h5>
</div>
<button type="submit" class="btn">Sign In</button>
<div id="adminResult"></div>
</form>
</div>
</div><script>
document.getElementById("adminSigninForm").addEventListener("submit", async
function (event) {
event.preventDefault(); // Prevent default form
submission
let adminEnteredValue =
document.getElementById("adminid").value.trim();
let
adminEnteredPassword = document.getElementById("adminPassword").value.trim();
let adminResultDiv = document.getElementById("adminResult");
// Clear previous errors
document.getElementById("adminidError").textContent = "";
document.getElementById("adminPasswordError").textContent = "";
adminResultDiv.textContent = ""; // Validate fields
if
(adminEnteredValue === "" || adminEnteredPassword === "") {
adminResultDiv.textContent = "'L Please fill in all fields.'";
adminResultDiv.style.color = "red";
return;
}
try {
let response = await fetch("http://localhost:8080/api/admin/
signin", {
method: "POST",
headers: {
"Content-Type": "application/json"
},
body: JSON.stringify({
identifier:
adminEnteredValue, // ' Use "identifier" instead of "adminid"
password: adminEnteredPassword
});
let result = await response.json();
if (response.ok) { // ' Check response status instead of message
adminResultDiv.textContent = "' Login successful!
Redirecting...";
adminResultDiv.style.color = "green";
// ' Store in sessionStorage
sessionStorage.setItem("userId", result.userId);
sessionStorage.setItem("userName", result.userName);
sessionStorage.setItem("role", result.role); // Store
admin details in localStorage
localStorage.setItem("userId",
result.userId);
localStorage.setItem("userName",
result.userName);
localStorage.setItem("role",
result.role); // ' Store admin role
// Redirect
after a short delay
setTimeout(() => {
window.location.href = "admin_dashboard.html";
}, 1500);
} else {
adminResultDiv.textContent = "'L " +
result.message;
adminResultDiv.style.color = "red";
}
} catch (error) {
console.error("Error:",
error);
adminResultDiv.textContent = "'L Server error. Please try
again later.";
}
}
}
}

```

```

adminResultDiv.style.color = "red";
</script>
style * {
margin: 0;
padding: 0;
box-sizing: border-box;
font-family: "Poppins", sans-serif;
body {
display: flex;
justify-content: center;
align-items: center;
height: 100vh;
background: white;
.container {
display: flex;
width: 100%;
background: black;
max-width: 1700px;
height: 700px;
box-shadow: 0px 4px 10px rgba(0, 0, 0, 0.3);
border-radius: 15px;
overflow: hidden;
background-color: #f0f5ff;
background-image:
radial-gradient(circle at 100% 150%, #f0f5ff 24%, #c8d7ff 25%, #c8d7ff 28%, #f0f5ff 29%, #f0f5ff 36%, #c8d7ff 36%, #c8d7ff 40%, transparent 40%, transparent),
radial-gradient(circle at 0 150%, #f0f5ff 24%, #c8d7ff 25%, #c8d7ff 28%, #f0f5ff 29%, #f0f5ff 36%, #c8d7ff 36%, #c8d7ff 40%, transparent 40%, transparent),
radial-gradient(circle at 50% 100%, #c8d7ff 10%, #f0f5ff 11%, #f0f5ff 23%, #c8d7ff 24%, #c8d7ff 30%, #f0f5ff 31%, #f0f5ff 43%, #c8d7ff 44%, #c8d7ff 50%, #f0f5ff 51%, #f0f5ff 63%, #c8d7ff 64%, #c8d7ff 71%, transparent 71%, transparent),
radial-gradient(circle at 100% 50%, #c8d7ff 5%, #f0f5ff 6%, #f0f5ff 15%, #c8d7ff 16%, #c8d7ff 20%, #f0f5ff 21%, #f0f5ff 30%, #c8d7ff 31%, #c8d7ff 35%, #f0f5ff 36%, #f0f5ff 45%, #c8d7ff 46%, #c8d7ff 49%, transparent 50%, transparent),
radial-gradient(circle at 0 50%, #c8d7ff 5%, #f0f5ff 6%, #f0f5ff 15%, #c8d7ff 16%, #c8d7ff 20%, #f0f5ff 21%, #f0f5ff 30%, #c8d7ff 31%, #c8d7ff 35%, #f0f5ff 36%, #f0f5ff 45%, #c8d7ff 46%, #c8d7ff 49%, transparent 50%, transparent);
background-size: 100px 50px;
}
.image-section {
width: 600px;
flex: 1;
background: url("pics/trading\ with\ coin.gif") no-repeat center / 100% 100%;
margin-top: 100px;
margin-bottom: 100px;
margin-left: 70px;
border-bottom-left-radius: 5%;
border-top-left-radius: 5%;
box-shadow: -5px 0 10px -2px rgba(0, 0, 0, 0.4);
}
.form-section {
width: 50%;
display: flex;
align-items: center;
justify-content: center;
backdrop-filter: blur(10px);
margin-top: 100px;
margin-bottom: 100px;
margin-right: 60px;
border-top-right-radius: 5%;
border-bottom-right-radius: 5%;
box-shadow: 5px 0 4px -2px rgba(0, 0, 0, 0.3);
}
.form-box {
background: white;
padding: 80px;
width: 100%;
height: 100%;
text-align: center;
border-top-right-radius: 5%;
border-bottom-right-radius: 5%;
box-shadow: 5px 0 4px -2px rgba(0, 0, 0, 0.3);
}
h2 {
font-family: 'Times New Roman', Times, serif;
text-align: center;
font-size: 40px;
font-weight: bold;
color: black;
text-shadow: 0px 4px 10px rgba(0, 0, 0, 0.1);
}
p {
font-family: 'Times New Roman', Times, serif;
text-align: center;
font-size: 20px;
font-weight: bold;
color: black;
margin-bottom: 30px;
}
h4 {
font-family: 'Franklin Gothic Medium', 'Arial Narrow', Arial, sans-serif;
color: black;
font-size: 19px;
font-weight: bold;
padding-bottom: 20px;
}
h5 {
color: black;
font-weight: bold;
font-size: 15px;
}
.signup {
text-decoration: none;
}
.signup :hover {
text-decoration: underline;
transform: scale(1.05);
}
.input-group {
margin-bottom: 15px;
text-align: left;
label {
font-family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida Sans Unicode', Geneva, Verdana, sans-serif;
padding-bottom: 10px;
font-weight: bold;
letter-spacing: 1px;
display: block;
font-size: 16px;
color: black;
}
input {
font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;
width: 100%;
padding: 12px 15px;
border: none;
border-radius: 6px;
font-size: 16px;
border: 2px solid #ccc;
}
.input-group input:hover {
border-color: #667eea;
box-shadow: 0px 0px 10px rgba(102, 126, 234, 0.5);
}

```



```

transform:
scale(1.02);@ }@ input:focus {@ outline: none;@ transform:
scale(1.05);@ box-shadow: 0px 0px 10px rgba(255, 255, 255, 0.4);@ }@
.error {@ color: red;@ font-size: 14px;@ margin-top:
5px;@ }@ @ .remember-me {@ display: flex;@ @ align-
items: center;@ @ @ @ }@ input[type=checkbox]{@ -
webkit-appearance: checkbox;@ -moz-appearance: checkbox;@
appearance: checkbox;@ display: inline-block;@ width: auto;@ }@
.remember-me input {@ margin-right: 8px;@ cursor: pointer;@
left: 0;@ @ }@ .btn {@ width: 100%;@
background: linear-gradient(to right, #667eea, #764ba2);@ color: white;@
padding: 14px;@ border: none;@ border-radius: 8px;@
font-size: 18px;@ cursor: pointer;@ transition: all 0.3s ease-in-
out;@ font-weight: bold;@ text-transform: uppercase;@
letter-spacing: 1px;@ box-shadow: 0px 5px 15px rgba(102, 126, 234, 0.3);@
}@ @ @ @ .btn:hover {@ background: linear-
gradient(to right, #764ba2, #667eea);@ transform: scale(1.05);@
box-shadow: 0px 8px 20px rgba(102, 126, 234, 0.5);@ }@ #result {@
font-weight: bold;@ margin-top: 15px;@ color: #00ff99;@ }@
.forgot {@ padding-left: 230px;@ text-decoration: none;@ }@
.forgot :hover{@ background: #ff4e88;@ transform:
scale(1.05);@ }@ @ @ @ @media (max-width: 768px) {@ .container {@
flex-direction: column;@ height: auto;@ }@
.image-section {@ height: 200px;@ }@ .form-
section {@ height: auto;@ padding: 20px;@ }@
.form-box {@ width: 90%;@ }@ }@ .text-overlay h2
{position: absolute;@ top: 6%;@ left: 25%;@ transform: translate(-50%, -50%);@ text-
align: center;@ padding: 20px;@ border-radius: 10px;@ }@ .text-overlay h2{@ color:
black;@ font-weight: bold;@ font-size: 24px;@ text-shadow: 0 0 1px #101111, 0 0 1px
#0e0f0f, 0 0 1px #121313;@ }@ .text-overlay p {position: absolute;@ top: 94%;@ left:
25%;@ transform: translate(-50%, -50%);@ text-align: center;@ color: black;@ padding:
20px;@ border-radius: 10px;@ }@ .text-overlay p{@ font-weight: bold;@ font-size: 20px;@
text-shadow: 0 0 1px #0a1013, 0 0 1px #13171d, 0 0 1px #0e1011;@ }@ /style>@ @

```

demo\target\classes\static\adminSignup.html

```
<title>Admin Signup Page</title><div class="container"><div class="text-
overlay"><h2>Start Your Trading Journey with Confidence</h2><p>Trade Smart. Grow Fast.</p></div><div class="image-section"></div><div
class="form-section"><div class="form-box"><h2>Admin Sign up</h2><form id="adminSignupForm"><div class="input-group"><label
for="adminName">Admin Name</label><input type="text" class="input-
style" id="userName" name="adminName" required placeholder="Admin Name"><div
class="error" id="adminNameError"></div></div><div
class="input-group"><label for="adminEmail">Email</label><input
type="email" id="email" class="input-style" name="email" required
placeholder="Email"><div class="error" id="adminEmailError"></div></div><div
class="input-group"><label for="adminPhone">Phone
Number</label><input type="tel" id="phone" class="input-style"
name="phone" required placeholder="Phone Number"><div class="error"
id="adminPhoneError"></div></div><div class="input-group"><label
for="tradingId">Trading View ID</label><input type="text"
id="tradingId" class="input-style" name="tradingId" required
placeholder="Trading View ID"></div><div class="input-group"><label
for="adminPassword">Password</label><input type="password"
id="password" class="input-style" name="password" required placeholder="Enter
Password"><div class="error" id="adminPasswordError"></div></div><div
class="input-group"><label for="confirmPassword">Confirm
Password</label><input type="password" id="confirmPassword"
class="input-style" name="confirmPassword" required placeholder="Repeat
Password"><div class="error" id="confirmPasswordError"></div></div><div
class="referral-section"><input type="checkbox"
id="referralCheckbox"><label for="referralCheckbox">Have a Referral
ID?</label></div><div class="input-group" id="referralId"
style="display: none;"><label for="referral">Referral ID</label><input
type="text" class="input-style" id="referral" name="referral"
placeholder="Referral ID"></div><button type="submit"
class="btn">Register Admin</button></form></div><div class="account"><h4>Already registered? <a class="signin" href="adminSignin.html">Login</a></h4></div></div></script>
document.getElementById("referralCheckbox").addEventListener("change",
function () {
    document.getElementById("referralId").style.display =
this.checked ? "block" : "none";
});
document.getElementById("adminSignupForm").addEventListener("submit", async
function (event) {
    event.preventDefault();
    let userName =
document.getElementById("userName").value.trim();
    let email =
document.getElementById("email").value.trim();
    let phone =
document.getElementById("phone").value.trim();
    let tradingId =
document.getElementById("tradingId").value.trim();
    let password =
document.getElementById("password").value.trim();
    let confirmPassword =
document.getElementById("confirmPassword").value.trim();
    let referralId =
document.getElementById("referral").value.trim() || null;
    let namePattern = /^(?!\\s)(?!.*\\s{2,})[a-zA-Z0-9 ]+(?!\\s)$/;
    let emailPattern = /^[a-zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,4}$/;
    let phonePattern = /^[6-9]\\d{9}$/;
    let passwordPattern = /^(?=.*[a-z])(?=.*[A-Z])(?=.*\\d)(?=.*[@$!%*?&])[A-Za-z\\d@$!%*?&]{8,}$/;
    let isValid = true;
    // Clear previous errors
    document.getElementById("adminNameError").textContent = "";
    document.getElementById("adminEmailError").textContent = "";
    document.getElementById("adminPhoneError").textContent = "";
    document.getElementById("adminPasswordError").textContent = "";
    document.getElementById("confirmPasswordError").textContent = "";
    if (!
namePattern.test(userName)) {
        document.getElementById("adminNameError").textContent = "Invalid name!";
        isValid = false;
    }

```

```

        }ð if (!emailPattern.test(email)) {ð
document.getElementById("adminEmailError").textContent = "Invalid Email
Format!";ð isValid = false;ð }ð if (!phonePattern.test(phone)) {ð
    document.getElementById("adminPhoneError").textContent = "Invalid
    Phone Number!";ð isValid = false;ð }ð if (!
passwordPattern.test(password)) {ð
document.getElementById("adminPasswordError").textContent =ð
"Password must have at least 8 characters, 1 special char, 1 uppercase, 1
lowercase, and 1 number!";ð isValid = false;ð }ð if (password !==
confirmPassword) {ð
document.getElementById("confirmPasswordError").textContent = "Passwords do
not match!";ð isValid = false;ð }ð if (!isValid) return;ð let
requestData = {ð username: userName,ð email: email,ð phone:
phone,ð tradingId: tradingId,ð password: password,ð
referralId: referralIdð };ð try {ð let response = await
fetch("http://localhost:8080/api/admin/signup", {ð method: "POST",ð
headers: { "Content-Type": "application/json" },ð body:
JSON.stringify(requestData)ð });ð let data;ð try {ð
    data = await response.json();ð } catch (jsonError) {ð
        console.error("Invalid JSON response", jsonError);ð
throw new Error("Invalid server response");ð }ð if (response.ok) {ð
    alert(`Admin Registered Successfully! \n\nAdmin Name:
    ${data.userName} \nAdmin ID: ${data.userId} \nReferral Code:
    ${data.referralCode}`);ð window.location.href = "adminSignin.html";ð
    } else {ð
document.getElementById("adminEmailError").textContent = data.message ||
"Registration failed!";ð }ð } catch (error) {ð
console.error("Error:", error);ð alert("An error occurred while
registering. Please try again.");ð }ð);ð</script>ðstyle>ð * {ð
margin: 0;ð padding: 0;ð box-sizing: border-box;ð
    font-family: "Poppins", sans-serif;ð }ð body {ð
    height: 150vh;ð display: flex;ð justify-
content: center;ð align-items: center;ð background: #ffff;ð
    }ð ð .container {ð display: flex;ð
width: 100%;ð max-width: 1500px;ð height: 950px;ð
    box-shadow: 0px 10px 30px rgba(0, 0, 0, 0.1);ð border-
radius: 15px;ð overflow: hidden;ð background-color:
#f5f5f5;ð background-image: linear-gradient(30deg, #ecec 12%,
transparent 12.5%, transparent 87%, #ecec 87.5%, #ecec),ð
linear-gradient(150deg, #ecec 12%, transparent 12.5%, transparent 87%,
#ecec 87.5%, #ecec),ð linear-gradient(30deg, #ecec 12%,
transparent 12.5%, transparent 87%, #ecec 87.5%, #ecec),ð
linear-gradient(150deg, #ecec 12%, transparent 12.5%, transparent 87%,
#ecec 87.5%, #ecec),ð linear-gradient(60deg, #dddddd 25%,
transparent 25.5%, transparent 75%, #dddddd 75%, #dddddd),ð
linear-gradient(60deg, #dddddd 25%, transparent 25.5%, transparent 75%,
#dddddd 75%, #dddddd);ð background-size: 40px 70px;ð
background-position: 0 0, 0 0, 20px 35px, 20px 35px, 0 0, 20px 35px;ð }ð
    ð .image-section {ð flex: 1;ð
width: 65%;ð background: white;ð margin-top: 60px;ð
    margin-bottom: 20px;ð margin-left: 70px;ð
border-bottom-left-radius: 5%;ð border-top-left-radius: 5%;ð box-
shadow: -5px 0 10px -2px rgba(0, 0, 0, 0.4);ð }ð .image{ð
    height: 370px;ð width: 38%;ð justify-items:
center;ð position: absolute;ð top: 70%;ð left:
24%;ð transform: translate(-50%, -50%);ð }ð .form-
section {ð width: 50%;ð padding-bottom: 20px;ð
display: flex;ð align-items: center;ð justify-content:
center;ð background: white;ð backdrop-filter: blur(10px);ð
    margin-top: 60px;ð

```

```

margin-right: 80px;#
margin-bottom: 20px;# border-top-right-radius: 5%;#
border-bottom-right-radius: 5%;# box-shadow: 5px 0 4px rgba(0, 0,
0, 0.3);# }# .form-box {# # background: white;#
padding: 60px;# width: 100%;# height: 100%;#
justify-content: center;# # text-align: center;#
# border-top-right-radius: 5%;# border-bottom-
right-radius: 5%;# # # # }# # #
h2{# font-family: 'Times New Roman', Times, serif;#
text-align: center;# font-size: 40px;# font-
weight: bold;# margin-bottom: 20px;# color: black;#
text-shadow: 0px 4px 10px rgba(0, 0, 0, 0.1);# # }#
h4{# font-family: 'Franklin Gothic Medium', 'Arial Narrow',
Arial, sans-serif;# color: black;# font-size: 19px;#
font-weight: bold;# padding-bottom: 10px;# }#
# # .input-group{# margin-bottom: 15px;#
text-align: left;# }# label {# font-family:
'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida Sans Unicode',
Geneva, Verdana, sans-serif;# font-weight: bold;#
display: block;# margin-bottom: 5px;# font-size: 14px;#
color: black;# letter-spacing: 1px;# }#
input {# font-family: 'Gill Sans', 'Gill Sans MT', Calibri,
'Trebuchet MS', sans-serif;# width: 100%;# padding: 12px
15px;# border: 2px solid #ccc;# border-radius: 8px;#
font-size: 16px;# transition: 0.3 ease;# #
}# # .input-group input:hover{# border-color:
#667eea;# box-shadow: 0px 0px 10px rgba(102, 126, 234, 0.5);#
transform: scale(1.02);# }# # #
.error {# color: red;# font-size: 12px;#
margin-top: 5px;# }# .referral-section{#
display: flex;# align-items: center;# gap: 10px;#
margin-bottom: 10px;# }# #
input[type=checkbox]{# -webkit-appearance: checkbox;# -
moz-appearance: checkbox;# appearance: checkbox;#
display: inline-block;# width: auto;# }# .btn {#
width: 100%;# background: linear-gradient(to right,
#667eea, #764ba2);# color: white;# padding: 14px;#
border: none;# border-radius: 8px;# font-
size: 18px;# cursor: pointer;# transition: all 0.3s ease-
in-out;# font-weight: bold;# # letter-spacing: 1px;#
box-shadow: 0px 5px 15px rgba(102, 126, 234, 0.3);# }#
# .btn:hover {# background: linear-gradient(to
right, #764ba2, #667eea);# transform: scale(1.05);# box-
shadow: 0px 8px 20px rgba(102, 126, 234, 0.5);# }# #
#referralId {# display: none;# }# # .account h4{#
padding-top: 10px;# font-size: 16px;# }# .signin{#
text-decoration: none;# font-size: 16px;# }#
.signin :hover{# text-decoration: underline;# color:
#667eea;# transition: scale(1.02);# }# # # /*
Responsive */# @media (max-width: 900px) {# .main-container {#
flex-direction: column;# width: 90%;#
height: auto;# }# .form-container, .image-
container {# width: 100%;# }# .image-
container {# height: 300px;# }# }# .text-
overlay {# position: absolute;# top: 23%;# left: 26%;# transform:
translate(-50%, -50%);# text-align: center;# color: white;# padding:
20px;# border-radius: 10px;# }# text-overlay h2{# color: black;# # font-
weight: bold;# font-size: 25px;# text-shadow: 0 0 1px #141616, 0 0 1px
#131314, 0 0 1px #121414;# overflow: hidden;# opacity: 0;# transform:
translateX(30px);#

```

```
        animation: fadeSlideUp 1.2s ease forwards;}}.text-  
overlay p{{    color: black;    font-weight: bold;    font-size: 22px;    text-  
shadow: 0 0 1px #101111, 0 0 1px #000000, 0 0 1px #16191b;    opacity: 0;      
transform: translateX(30px);    animation: fadeSlideUp 1.2s ease forwards;    }}  
    @keyframes fadeSlideUp {{    to {{    opacity: 1;    transform: translateY(0);      
    }}    }}/style>
```

demo\target\classes\static\admin_dashboard.html

```
<!DOCTYPE html>ðhtml lang="en">ðhead>ð    <meta charset="UTF-8">ð
<title>Admin Dashboard</title>ð    <link rel="stylesheet" href="https://
cdnjs.cloudflare.com/ajax/libs/font-awesome/6.5.0/css/all.min.css">ð
<script src="https://kit.fontawesome.com/a076d05399.js"
crossorigin="anonymous"></script>ð    <script src="https://
cdnjs.cloudflare.com/ajax/libs/jspdf/2.5.1/jspdf.umd.min.js"></script>ð<script
src="https://cdnjs.cloudflare.com/ajax/libs/jspdf-autotable/3.5.28/
jspdf.plugin.autotable.min.js"></script>ð<link rel="stylesheet"
href="admin_dashboard.css">ð/head>ðbody>ð    <!-- HEADER -->ð    <header
class="header">ð        <button class="hamburger" id="menuToggle">ð
<i class="fas fa-bars"></i>ð        </button>ð        <div class="logo"></div>ð        <div
class="header-title">Admin Management System</div>ð        <div class="user-
info-mini">ð            <div class="notification-icon"
onclick="toggleNotifications()">ð                Ø=Ÿ    <span
id="notificationCount">0</span>ð            </div>ð            <div
class="user-data">ð                <span id="headerAdminName"
class="username">Loading...</span>ð                <span id="headerAdminRole"
class="name" style="font-size: 10px; color: black; margin-left:
4px;">(Admin)</span>ð            </div>ð        </div>ð    <!--
SIDEBAR -->ð    <aside class="sidebar" id="sidebar">ð        <div
class="sidebar-header">ð            <div class="user-info">ð
<div class="user-avatar" id="adminAvatar">A</div>ð            <div>ð
<div class="user-name" id="sidebarAdminName">Loading...</
div>ð            <div class="user-role" id="sidebarAdminRole">Admin</
div>ð            <div class="user-status" id="adminStatus">Online</div>ð
        </div>ð        </div>ð        </div>ð        <ul class="nav-
menu">ð            <li><a href="#" onclick="showModule('dashboardModule')">
<i class="fas fa-tachometer-alt"></i> Dashboard</a></li>ð            <li><a
href="#" onclick="showModule('manageUsersModule')"><i class="fas fa-user-
cog"></i> Manage Users</a></li>ð                <li><a href="#"
onclick="showModule('reportsModule')"><i class="fas fa-chart-line"></i>
Reports</a></li>ð                <li><a href="#"
onclick="showModule('adminProfileModule')"><i class="fas fa-user"></i>
Profile</a></li>ð                <li><a href="#"
onclick="showModule('settingsModule')"><i class="fas fa-tools"></i>
Settings</a></li>ð                <li><a href="#"
onclick="showModule('chatModule')"><i class="fas fa-comment-sms"></i> Chat</
a></li>ð            </ul>ð            <button id="logout" class="logout-btn">ð
<i class="fas fa-sign-out-alt"></i> Logoutð        </button>ð    </
aside>ð    <!-- MAIN CONTENT -->ð    <main class="main-content"
id="mainContent">ð        <div id="dashboardModule" class="module"
style="display: block;">ð            <div class="dashboard-header">ð                <h1
class="dashboard-title">Admin Dashboard</h1>ð                <div
class="breadcrumb">Home &gt; Dashboard</div>ð            </div>ð            <div
class="data-card fade-in">ð                <h2>Session Information</h2>ð                <div
id="sessionDataContainer">ð                    <div class="loading"
id="sessionLoading">Loading...</div>ð                    <table class="session-
table" id="sessionTable" style="display: none;">ð                        <thead>ð
<tr>ð                            <th>Admin ID</th>ð
<th>Admin Name</th>ð
<th>Sign-in Time</th>ð                            <th>Logout Time</th>ð
<th>Role</th>ð
<th>Status</th>ð                        </tr>ð                        <tbody>ð
<tr>ð                            <td id="userIdDisplay"></td>ð
<td id="userNameDisplay"></td>ð
<td id="signinTimeDisplay"></td>ð
<td id="logoutTimeDisplay"></td>ð
```

```

                <td id="roleDisplay"></td>Ð
                <td id="statusDisplay"></td>Ð
            </tr>Ð
        </tbody>Ð
    </table>Ð
    </div>Ð
    </div>Ð
    </div>Ð
    Ð
    <!--
Manage Users Section -->Ð <div id="manageUsersModule" class="module"
style="display: none;">Ð <div class="dashboard-header">Ð <h1
class="dashboard-title">Manage Users</h1>Ð <div class="breadcrumb">Home
&gt; Manage Users</div>Ð </div>Ð Ð <!-- Search Bar -->Ð <!-- Search and
Sort Controls for All Users -->Ð<div style="margin-bottom: 15px; display:
flex; gap: 10px; align-items: center;">Ð <div style="position: relative;">Ð
    <input type="text" id="allUserSearchInput" placeholder="Search by User
ID, Name, or Role" style="padding-left: 30px;">Ð <span style="position:
absolute; left: 8px; top: 50%; transform: translateY(-50%); color: #aaa;">Ð
    Ø=Ý Ð </span>Ð </div>Ð <select id="allUserSortSelect">Ð
<option value="">Sort by</option>Ð <option value="userId">User ID</option>Ð
<option value="userName">User Name</option>Ð <option
value="role">Role</option>Ð </select>Ð <button
onclick="downloadAllUsersReport()" class="download-btn"> Ø=ÜÄ Download All Users</
button>Ð Ð </div>Ð Ð Ð <div class="data-card fade-in">Ð <h2>User
List</h2>Ð <div id="userLoading">Loading users...</div>Ð <table
class="session-table" id="userTable" style="display: none;">Ð
<thead>Ð
    <tr>Ð
        <th>User ID</th>Ð
        <th>User Name</th>Ð
        <th>Email</th>Ð
        <th>Phone</th>Ð
        <th>Role</th>Ð
    </tr>Ð
    </thead>Ð
    <tbody
id="userTableBody">Ð
        <!-- User rows will be inserted here -->Ð
    </tbody>Ð
</table>Ð </div>Ð </div>Ð Ð Ð <div
id="reportsModule" class="module" style="display: none; padding: 20px;">Ð
    <h2 style="font-size: 24px; margin-bottom: 20px;">Recent User
Registrations</h2>Ð Ð <div style="margin-bottom: 15px; display:
flex; gap: 10px; align-items: center;">Ð <div style="position:
relative;">Ð
    <input type="text" id="searchInput"
placeholder="Search by User ID, Name, or Role" style="padding-left: 30px;">Ð
    <span style="position: absolute; left: 8px; top: 50%;
transform: translateY(-50%); color: #aaa;">Ð
    Ø=Ý Ð </span>Ð </div>Ð <select id="sortSelect">Ð
<option value="">Sort by</option>Ð <option value="userId">User
ID</option>Ð <option value="userName">User Name</option>Ð
<option value="role">Role</option>Ð </select>Ð
    <button onclick="downloadUserReport()" class="download-btn">Ø=ÜÄ
Download User Report (PDF)</button>Ð </div>Ð <div
id="recentUsersLoading" class="loading-message">Loading recent users...</div>Ð
    <table id="recentUsersTable" class="user-table">Ð
        <thead>Ð
            <tr>Ð
                <th>Role</th>Ð
                <th>User ID</th>Ð
                <th>User Name</th>Ð
                <th>Created At</th>Ð
            </tr>Ð
        </thead>Ð
        <tbody id="recentUsersTableBody"></tbody>Ð
    </table>Ð </div>Ð Ð <div id="adminProfileModule" class="module"
style="display: none;">Ð <div class="dashboard-header">Ð <h1
class="dashboard-title">Admin Profile</h1>Ð <div
class="breadcrumb">Home &gt; Profile</div>Ð </div>Ð Ð <div
class="profile-card">Ð <div class="profile-photo">Ð Ð </div>Ð
    <div class="profile-details">Ð <div class="profile-row">Ð
        <span class="profile-label">User ID:</span>Ð <span
id="profileUserId">Loading...</span>Ð </div>Ð <div
class="profile-row">Ð <span class="profile-label">Name:</span>Ð
        <span id="profileUserName">Loading...</span>Ð </div>Ð
    </div>Ð
    <div class="profile-row">Ð

```

```

label">Email:</span>Đ        <span id="profileEmail">Loading...</span>Đ
    </div>Đ        <div class="profile-row">Đ        <span
class="profile-label">Phone:</span>Đ        <span
id="profilePhone">Loading...</span>Đ        </div>Đ        <div
class="profile-row">Đ        <span class="profile-label">Role:</span>Đ
    <span id="profileRole">Loading...</span>Đ        </div>Đ
    <div class="profile-row">Đ        <span class="profile-
label">Referral Code:</span>Đ        <span
id="profileReferralCode">Loading...</span>Đ        </div>Đ        <div
class="profile-row">Đ        <span class="profile-label">Created At:</
span>Đ        <span id="profileCreatedAt">Loading...</span>Đ        </
div>Đ        </div>Đ        </div>Đ        <div
id="settingsModule" class="module" style="display: none;">Đ        <div
class="dashboard-header">Đ        <h1 class="dashboard-title">Admin
Settings</h1>Đ        <div class="breadcrumb">Home > Settings</div>Đ
</div>Đ        Đ        <!-- Username Section -->Đ        <div
class="settings-section">Đ        <label>Username:</label>Đ
<span id="currentUsername" class="current-value">Loading...</span>Đ
<div id="usernameEdit" class="edit-section">Đ        <input
type="text" id="usernameInput" placeholder="Enter new username" />Đ
    <button class="save-button" onclick="saveUsername()">Save</
button>Đ        <button class="cancel-button"
onclick="cancelEdit('username')">Cancel</button>Đ        </div>Đ        Đ
    </div>Đ        <!-- Phone Section -->Đ        <div class="settings-
section">Đ        <label>Phone:</label>Đ        <span id="currentPhone"
class="current-value">Loading...</span>Đ        <div id="phoneEdit"
class="edit-section">Đ        <input type="tel" id="phoneInput"
placeholder="Enter new phone number" />Đ        <button class="save-
button" onclick="savePhone()">Save</button>Đ        <button
class="cancel-button" onclick="cancelEdit('phone')">Cancel</button>Đ
    </div>Đ        Đ        </div>Đ        <!-- Single Edit
Button -->Đ        <div class="settings-section">Đ        <button class="edit-button"
id="editProfileBtn" onclick="editProfile()">Edit Profile</button>Đ        </div>Đ
    <!-- Profile Picture -->Đ        <div class="settings-section">Đ
    <label>Change Profile Picture:</label>Đ        Đ
<input type="file" id="uploadInput" accept="image/*" />Đ        </div>Đ        Đ
    <div id="settingsStatus" class="status-msg"></div>Đ        Đ        Đ
<div id="successPopup">Đ        <p id="successMessage"></p>Đ        <button
onclick="closePopup()">OK</button>Đ        </div>Đ        </div>Đ        Đ        Đ        Đ
    <div id="chatModule" class="module" style="display: none;">Đ
    <h2>Chat</h2>Đ        <!-- Chat messages will be displayed here -->Đ
    <div class="chat-box" id="chatBox"></div>Đ        Đ        <!-- Chat
input area -->Đ        <div class="chat-input-area">Đ        <input
type="text" id="chatInput" placeholder="Type your message..." />Đ
<button onclick="sendChatMessage()">Send</button>Đ        </div>Đ        Đ
    <!-- Loading indicator for fetching chat -->Đ        <div
id="loadingIndicator" style="display: none;">Loading...</div>Đ        </div>Đ        Đ
    <!-- Notifications Panel -->Đ        <div id="notificationPanel"
class="notification-panel">Đ        <h3>Đ=Ÿ Notifications</h3>Đ        <ul
id="adminNotificationList"></ul>Đ        <button
onclick="closeNotifications()" style="Đ        position: absolute;Đ
top: 10px;Đ        right: 10px;Đ        background: none;Đ        border: none;Đ
font-size: 20px;Đ        color: red;Đ        cursor: pointer;Đ
font-weight: bold;Đ        ">x</button>Đ        </div>Đ        Đ        </main>Đ        Đ
<script src="admin_dashboard.js"></script>ĐĐ/body>Đ/html>Đ

```


demo\target\classes\static\admin_dashboard2.html

```
<!DOCTYPE html>ðhtml lang="en">ðhead>ð    <meta charset="UTF-8">ð    <meta
name="viewport" content="width=device-width, initial-scale=1.0">ð
<title>Admin Dashboard</title>ð    <link rel="stylesheet"
href="admin_dashboard.css">ð/head>ðbody>ð    <div class="admin-container">ð
    <!-- Sidebar -->ð        <aside class="sidebar">ð                <h2>Admin
Panel</h2>ð                <ul>ð                    <li><a href="#">ð<ßà Dashboard</a></li>ð
                    <li><a href="#">ð<Û Users</a></li>ð                    <li><a
href="#">ð<™p Settings</a></li>ð                    <li><a href="#">ð<Û Logout</a></li>ð
                    </ul>ð                </aside>ð                <!-- Main Content -->ð                <main
class="main-content">ð                    <header>ð                        <h1>Welcome, Admin</
h1>ð                        <div class="notification-icon"
onclick="toggleNotifications()">ð                            ð=Ý    <span
id="notificationCount">0</span>ð                        </div>ð                    </header>ð
                    <!-- Recent Users Table -->ð                    <section class="dashboard-
section">ð                        <h2>ð=Û Recent User Signups</h2>ð                        <div
class="table-container">ð                            <table>ð
<thead>ð                                <tr>ð                                    <th>Name</th>ð
                                    <th>Email</th>ð
                                    <th>Phone</th>ð                                    <th>Signup Time</th>ð
                                </tr>ð                            </thead>ð
                                <tbody id="userList">ð                                    <!--
Users will be loaded here dynamically -->ð                                    </tbody>ð
                                </table>ð                            </div>ð                        </section>ð
                    <!-- Notifications Panel -->ð                    <div
id="notificationPanel" class="notification-panel">ð                        <h3>ð=Ý
Notifications</h3>ð                        <ul id="adminNotificationList"></ul>ð
                        <button onclick="closeNotifications()">Close</button>ð
                    </div>ð                </main>ð                </div>ð                <script>ð                    // Fetch
Recent Usersðasync function fetchRecentUsers() {ð        try {ð            const
response = await fetch("http://localhost:8080/api/recent-signups");ð
            const users = await response.json();ð            const userList =
document.getElementById("userList");ð            userList.innerHTML = "";ð
            users.forEach(user => {ð                const row = document.createElement("tr");ð
                row.innerHTML = `ð                    <td>${user.userId}</td>ð
                <td>${user.userName}</td>ð                    <td>${user.email}</
td>ð                    <td>${user.phone}</td>ð                    <td>${new
Date(user.createdAt).toLocaleString()}</td>ð                `;ð
                userList.appendChild(row);ð            });ð        } catch (error) {ð
            console.error("Error fetching users:", error);ð        }ðð/ Fetch Admin
Notificationsðasync function fetchAdminNotifications() {ð        try {ð            const
response = await fetch("http://localhost:8080/admin-notifications/unread");ð
            const notifications = await response.json();ð            const
notificationList = document.getElementById("adminNotificationList");ð
            const notificationCount = document.getElementById("notificationCount");ð
            notificationList.innerHTML = "";ð            notificationCount.textContent
= notifications.length;ð            notifications.forEach(notification => {ð
                const item = document.createElement("li");ð
                item.innerText = notification.message;ð                item.onclick = async () => {ð
                    await markAsRead(notification.id);ð
                    fetchAdminNotifications();ð                });ð
                notificationList.appendChild(item);ð            });ð        } catch (error) {ð
            console.error("Error fetching notifications:", error);ð        }ðð/ ' Mark
notification as read in the databaseðasync function markAsRead(notificationId)
{ð        try {ð            await fetch(`http://localhost:8080/admin-notifications/mark-
as-read/${notificationId}`, {ð                method: "POST",ð                headers:
{ "Content-Type": "application/json" }ð            });ð        } catch (error) {ð
            console.error("Error marking notification as read:", error);ð        }ðð/ Close
Notification Panelð
```

```

function closeNotifications() {
document.getElementById("notificationPanel").style.display = "none";
Toggle Notifications Panel
function toggleNotifications() {
const panel =
document.getElementById("notificationPanel");
panel.style.display =
panel.style.display === "block" ? "none" : "block";
// Load Initial Data
fetchRecentUsers();
fetchAdminNotifications();
}
</script>
<style>
/*
General Styling */
body {
font-family: "Poppins", sans-serif;
margin: 0;
padding: 0;
display: flex;
background: #f4f4f4;
}
/* Dashboard
Container */
.admin-container {
display: flex;
width: 100%;
}
/* Sidebar */
.sidebar {
width: 200px;
background: #2c3e50;
color: white;
padding: 20px;
height: 100vh;
position: fixed;
left: 0;
top: 0;
animation: slideIn 0.5s ease-in-out;
}
/* Sidebar h2 */
.sidebar h2 {
text-align: center;
padding: 15px;
transition: background 0.3s;
}
/* Sidebar ul li */
.sidebar ul li {
padding: 15px;
background: #34495e;
}
/* Sidebar ul li a */
.sidebar ul li a {
color: white;
text-decoration: none;
font-weight: bold;
}
/* Main Content */
.main-content {
flex: 1;
margin-left: 250px;
padding: 20px;
animation: fadeIn 0.8s ease-in-out;
}
/* Header */
.header {
display: flex;
justify-content: space-between;
align-items: center;
background: #fff;
padding: 15px;
border-radius: 10px;
box-shadow: 0px 4px 10px rgba(0, 0, 0, 0.1);
}
/* Notification Icon */
.notification-icon {
cursor: pointer;
font-size: 22px;
position: relative;
}
/* Notification Count */
.notification-count {
background: red;
color: white;
padding: 5px 8px;
border-radius: 50%;
position: absolute;
top: -8px;
right: -8px;
font-size: 14px;
}
/* Dashboard Section */
.dashboard-section {
margin-top: 20px;
background: white;
padding: 20px;
border-radius: 10px;
box-shadow: 0px 4px 10px rgba(0, 0, 0, 0.1);
}
/* Table */
table-container {
overflow-x: auto;
}
table {
width: 100%;
border-collapse: collapse;
margin-top: 10px;
}
thead, td {
border: 1px solid #ccc;
padding: 12px;
text-align: left;
}
tbody tr {
background: #2c3e50;
color: white;
}
/* Notifications */
.notification-panel {
display: none;
position: fixed;
top: 50px;
right: 20px;
background: white;
border-radius: 8px;
width: 320px;
box-shadow: 0px 4px 10px rgba(0, 0, 0, 0.2);
padding: 15px;
}
/* Notification-panel ul */
.notification-panel ul {
list-style: none;
padding: 0;
}
/* Notification-panel ul li */
.notification-panel ul li {
padding: 10px;
border-bottom: 1px solid #ddd;
cursor: pointer;
}
/* Animations */
@keyframes slideIn {
from {
transform: translateX(-100%);
}
to {
transform: translateX(0);
}
}
@keyframes fadeIn {
from {
opacity: 0;
}
to {
opacity: 1;
}
}
</style>
</body>
</html>

```

demo\target\classes\static\forgot.html

```
<title>Forgot Page</title><div class="container"><div class="text-
overlay"></div><div class="form-section"><div class="form-box">
  <h2>Forgotten Password</h2><p>Create your new
password</p><!-- Email Verification Section --><div
id="emailSection"><div class="input-group"><label
for="email">Enter your registered Email</label><input
type="email" id="email" required placeholder="Email"><div
class="error" id="emailError"></div><button
class="btn" onclick="checkEmail()">Verify Email</button></div>
<!-- Password Reset Section (Hidden Initially) --><div
id="resetPasswordSection" style="display: none;"><div
class="input-group"><label for="newPassword">New Password</
label><input type="password" id="newPassword" required
placeholder="New Password"><div class="error"
id="passwordError"></div><div class="input-
group"><label for="confirmNewPassword">Confirm Password</
label><input type="password" id="confirmNewPassword" required
placeholder="Confirm Password"><div class="error"
id="confirmPasswordError"></div><button
class="btn" onclick="resetPassword()">Reset Password</button></div>
<div class="account"><h5>Remembered your password? <a
class="signin" href="signin.html">Sign In</a></h5></div>
</div></div><script>
function checkEmail() {
  let email = document.getElementById("email").value.trim();
  let emailError = document.getElementById("emailError");
  // Clear previous error messages
  emailError.textContent = "";
  fetch("http://localhost:8080/api/verify-email", {
    method: "POST",
    headers: { "Content-Type": "application/json" },
    body: JSON.stringify({ email: email })
  })
    .then(response => response.json())
    .then(data => {
      if (data.status === "error") {
        emailError.textContent = data.message;
      } else {
        // Hide email section and show password reset
        document.getElementById("emailSection").style.display = "none";
        document.getElementById("resetPasswordSection").style.display = "block";
      }
    })
    .catch(error => {
      console.error("Error:", error);
      emailError.textContent = "Server error. Please try again.";
    });
}

function resetPassword() {
  let newPassword = document.getElementById("newPassword").value.trim();
  let confirmNewPassword = document.getElementById("confirmNewPassword").value.trim();
  let passwordError = document.getElementById("passwordError");
  let confirmPasswordError = document.getElementById("confirmPasswordError");
  // Password validation pattern
  let passwordPattern = /^(?=.*[A-Z])(?=.*[a-z])(?=.*\d)(?=.*[@$!%*?&])[A-Za-z\d@$!%*?&]{8,}$/;
  // Validate new password
  if (!passwordPattern.test(newPassword)) {
    passwordError.textContent = "Password must be 8+ chars, 1 uppercase, 1 lowercase, 1 number & 1 special char.";
    return;
  }
  // Confirm password match
  if (newPassword !== confirmNewPassword) {
    confirmPasswordError.textContent = "Passwords do not match!";
    return;
  }
  // Send the new password to the server for updating
  let email = document.getElementById("email").value.trim();
  fetch("http://localhost:8080/api/reset-password", {
    method: "POST",
    headers: { "Content-Type": "application/json" },
    body: JSON.stringify({
      email: email,
      newPassword: newPassword
    })
  })
    .then(response => response.json())
    .then(data => {
      if (data.status === "success") {
        // Password reset successful
        // Show success message
      } else {
        // Password reset failed
        // Show error message
      }
    });
}
```

```

        .then(data => {
            if (data.status === "success") {
                alert("Password updated successfully! Redirecting to login...");
                window.location.href = "signin.html"; // Redirect to login page
            } else {
                alert(data.message);
            }
        }).catch(error => {
            console.error("Error:", error);
            alert("Server error. Please try again.");
        });
    </script>
    <style>
        * {
            margin: 0;
            padding: 0;
            box-sizing: border-box;
            font-family: "poppins", sans-serif;
        }
        body {
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
            background: white;
        }
        .container {
            display: flex;
            width: 100%;
            max-width: 1700px;
            height: 700px;
            border-radius: 8px;
            overflow: hidden;
            box-shadow: 0px 0px 10px rgba(0, 0, 0, 0.1);
            background-color: #fafafa;
            background-image: linear-gradient(135deg, #f5f5f5 25%, transparent 25%),
                linear-gradient(225deg, #f5f5f5 25%, transparent 25%),
                linear-gradient(45deg, #f5f5f5 25%, transparent 25%),
                linear-gradient(315deg, #f5f5f5 25%, transparent 25%);
            background-position: 10px 0, 10px 0, 0 0, 0 0;
            background-size: 20px 20px;
            background-repeat: repeat;
        }
        .image-section {
            width: 65%;
            flex: 1;
            background: url("pics/forex\ flaticon.gif ") no-repeat center / 100% 100%;
            margin-top: 100px;
            margin-bottom: 100px;
            margin-left: 80px;
            border-bottom-left-radius: 5%;
            border-top-left-radius: 5%;
            box-shadow: -5px 0 10px -2px rgba(0, 0, 0, 0.2);
        }
        .form-section {
            width: 50%;
            display: flex;
            justify-content: center;
            align-items: center;
            backdrop-filter: blur(10px);
            margin-top: 100px;
            margin-bottom: 100px;
            margin-right: 80px;
            height: 100%;
            background: white;
            padding: 80px;
            text-align: center;
            border-top-right-radius: 5%;
            border-bottom-right-radius: 5%;
            box-shadow: 5px 0 4px -2px rgba(0, 0, 0, 0.3);
        }
        h2 {
            padding-bottom: 10px;
            text-align: center;
            font-size: 30px;
            font-weight: bold;
            color: black;
            text-shadow: 0px 4px 10px rgba(0, 0, 0, 0.2);
            font-family: 'Franklin Gothic Medium', 'Arial Narrow', Arial, sans-serif;
        }
        p {
            text-align: center;
            font-size: 20px;
            color: black;
            margin-bottom: 20px;
            text-align: center;
            margin-bottom: 10px;
            font-weight: bold;
            font-size: 20px;
        }
        .heading {
            padding-bottom: 50px;
        }
        .input-group {
            margin-bottom: 15px;
            text-align: left;
        }
        label {
            font-family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida Sans Unicode', Geneva, Verdana, sans-serif;
            font-weight: bold;
            display: block;
            margin-bottom: 5px;
            font-size: 14px;
            color: black;
            letter-spacing: 1px;
        }
        input {
            font-family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;
            width: 100%;
            padding: 12px 15px;
            border: 2px solid #ccc;
            border-radius: 8px;
            font-size: 16px;
            transition: 0.3 ease;
        }
        .input-group input:hover {
            border-color: #667eea;
            box-shadow: 0px 0px 10px rgba(102, 126, 234, 0.5);
            transform: scale(1.02);
        }
        .error {
            color: red;
            font-size: 12px;
        }
        .btn {
            width: 100%;
            background: linear-gradient(to right, #667eea, #764ba2);
            color: black;
            padding: 14px;
            border: none;
            border-radius: 8px;
            font-size: 16px;
            cursor: pointer;
            transition: all 0.3s ease-in-out;
            font-weight: bold;
            letter-spacing: 1px;
            box-shadow: 0px 5px 15px rgba(102, 126, 234, 0.3);
        }
        .btn:hover {
            background: linear-gradient(to right, #764ba2, #667eea);
        }
    </style>

```

```

        transform: scale(1.05);@        box-shadow: 0px 8px
20px rgba(102, 126, 234, 0.5);@    }@ #resetPasswordSection {@
display: none;@    }@    @ .account{@        padding-top: 20px;@        text-
align: center;@    }@    .account h5{@        font-size: 16px;@    }@    .signin{@
        text-decoration: none;@        @        font-weight: bold;@        font-
size: 16px@    }@    @ .signin :hover{@        text-decoration: underline;@
        transform: scale(1.05);@    @    }@    .text-overlay h2 {@position:
absolute;@top: 6%;@left: 28%;@transform: translate(-50%, -50%);@text-align:
center;@padding: 20px;@border-radius: 10px;@}@text-overlay h2{@color: black;@font-
weight: bold;@font-size: 23px;@text-shadow: 0 0 1px #121414, 0 0 1px #181b1b, 0
0 1px #0e0f0f;@animation: fadeInOut 2.5s ease-in-out infinite;@display:
inline-block;@}@text-overlay p {@position: absolute;@top: 91%;@left: 28%;@
transform: translate(-50%, -50%);@text-align: center;@padding: 20px;@border-
radius: 10px;@}@text-overlay p{@color: black;@font-weight: bold;@font-size: 21px;@
text-shadow: 0 0 1px #161718, 0 0 1px #0f1011, 0 0 1px #171a1b;@animation:
fadeInOut 2.5s ease-in-out infinite;@display: inline-block;@}@keyframes
fadeInOut {@0% {@opacity: 0;@}@50% {@opacity: 1;@}@100% {@
opacity: 0;@}@}@/style>@

```

demo\target\classes\static\homepage.html

```
<!DOCTYPE html>ðhtml lang="en">ðhead>ð <meta charset="UTF-8">ð <meta
name="viewport" content="width=device-width, initial-scale=1.0">ð
<title>TradePro - Advanced Trading Platform</title>ð <style>ð  * {ð
margin: 0;ð    padding: 0;ð    box-sizing: border-box;ð    font-family:
'Segoe UI',Tahoma, Geneva, Verdana, sans-serif;ð    }ð    ð body {ð
background-color: #f0f2f5;ð    color: #333;ð    }ð    ð .navbar {ð
display: flex;ð    justify-content: space-between;ð    align-items: center;ð
background-color: #0a2647;ð    padding: 1rem 2rem;ð    color: white;ð
box-shadow: 0 2px 4px rgba(0,0,0,0.1);ð    }ð    ð .logo {ð    font-
size: 1.5rem;ð    font-weight: bold;ð    color: white;ð    text-
decoration: none;ð    }ð    ð .logo span {ð    color: #38bdf8;ð    }ð    ð
.nav-links {ð    display: flex;ð    gap: 1.5rem;ð    }ð    ð .nav-links
a {ð    color: white;ð    text-decoration: none;ð    font-weight: 500;ð
transition: color 0.3s ease;ð    }ð    ð .nav-links a:hover {ð
color: #38bdf8;ð    }ð    ð .auth-buttons {ð    display: flex;ð    gap:
1rem;ð    }ð    ð .auth-buttons button {ð    padding: 0.5rem 1rem;ð
border-radius: 4px;ð    font-weight: 500;ð    cursor: pointer;ð
transition: all 0.3s ease;ð    }ð    ð .login-btn {ð    background:
transparent;ð    border: 1px solid white;ð    color: white;ð    }ð    ð
.login-btn:hover {ð    background: rgba(255, 255, 255, 0.1);ð    }ð    ð
.signup-btn {ð    background: #38bdf8;ð    border: none;ð    color:
#0a2647;ð    }ð    ð .signup-btn:hover {ð    background: #0ea5e9;ð    }ð    ð
.hero {ð    padding: 2rem;ð    background: #0a2647;ð    color: white;ð
text-align: center;ð    padding-bottom: 4rem;ð    }ð    ð .hero h1 {ð
font-size: 2.5rem;ð    margin-bottom: 1rem;ð    }ð    ð .hero p {ð
font-size: 1.1rem;ð    max-width: 800px;ð    margin: 0 auto;ð    margin-
bottom: 2rem;ð    color: #cbd5e1;ð    }ð    ð .get-started {ð    padding:
0.75rem 1.5rem;ð    background: #38bdf8;ð    color: #0a2647;ð    border:
none;ð    border-radius: 4px;ð    font-weight: 600;ð    font-size: 1rem;ð
cursor: pointer;ð    transition: background 0.3s ease;ð    }ð    ð
.get-started:hover {ð    background: #0ea5e9;ð    }ð    ð .dashboard {ð
max-width: 1200px;ð    margin: -2rem auto 2rem;ð    padding: 1rem;ð
display: grid;ð    grid-template-columns: repeat(12, 1fr);ð    grid-
gap: 1rem;ð    }ð    ð .card {ð    background: white;ð    border-radius:
8px;ð    box-shadow: 0 4px 6px rgba(0,0,0,0.1);ð    padding: 1.5rem;ð
transition: transform 0.3s ease, box-shadow 0.3s ease;ð    }ð    ð
.card:hover {ð    transform: translateY(-5px);ð    box-shadow: 0 8px
15px rgba(0,0,0,0.1);ð    }ð    ð .price-chart {ð    grid-column: span 8;ð
}ð    ð .market-summary {ð    grid-column: span 4;ð    }ð    ð .volume-
chart, .technical-indicators {ð    grid-column: span 6;ð    }ð    ð .order-
book, .recent-trades {ð    grid-column: span 6;ð    }ð    ð .news-feed {ð
grid-column: span 12;ð    }ð    ð .card h2 {ð    margin-bottom: 1rem;ð
font-size: 1.25rem;ð    color: #0a2647;ð    display: flex;ð
justify-content: space-between;ð    align-items: center;ð    }ð    ð .card
h2 .icon {ð    color: #64748b;ð    font-size: 1rem;ð    cursor: pointer;ð
}ð    ð .card h2 .icon:hover {ð    color: #0a2647;ð    }ð    ð .table {ð
width: 100%;ð    border-collapse: collapse;ð    }ð    ð .table
th, .table td {ð    padding: 0.75rem;ð    text-align: left;ð    border-
bottom: 1px solid #e2e8f0;ð    }ð    ð .table th {ð    font-weight: 600;ð
color: #64748b;ð    }ð    ð .up {ð    color: #10b981;ð    }ð    ð
.down {ð    color: #ef4444;ð    }ð    ð .news-item {ð    display: flex;ð
gap: 1rem;ð    margin-bottom: 1rem;ð    padding-bottom: 1rem;ð
border-bottom: 1px solid #e2e8f0;ð    }ð    ð .news-item:last-child {ð
margin-bottom: 0;ð    padding-bottom: 0;ð    border-bottom: none;ð    }ð    ð
.news-image {ð    width: 80px;ð    height: 80px;ð    background-color:
#e2e8f0;ð    border-radius: 4px;ð    overflow: hidden;ð    }ð    ð .news-
content h3 {ð    font-size: 1rem;ð    margin-bottom: 0.5rem;ð    }ð    ð
.news-content p {ð    font-size: 0.875rem;ð    color: #64748b;ð
margin-bottom: 0.5rem;ð
```

```

    } @media (max-width: 768px) {
      .news-content .timestamp { font-size: 0.75rem; color: #94a3b8; }
      .footer { background: #0f172a; color: white; padding: 3rem 2rem; }
      .footer-content { max-width: 1200px; margin: 0 auto; display: grid; grid-template-columns: repeat(4, 1fr); gap: 2rem; }
      .footer-section h3 { margin-bottom: 1rem; font-size: 1.1rem; }
      .footer-section a { display: block; color: #cbd5e1; text-decoration: none; margin-bottom: 0.5rem; transition: color 0.3s ease; }
      .footer-section a:hover { color: #38bdf8; }
      .copyright { text-align: center; margin-top: 2rem; padding-top: 2rem; border-top: 1px solid #1e293b; color: #94a3b8; font-size: 0.875rem; }
    }
    @media (max-width: 768px) {
      .dashboard { grid-template-columns: 1fr; }
      .price-chart, .market-summary, .volume-chart, .technical-indicators, .order-book, .recent-trades, .news-feed { grid-column: span 1; }
    }
    .footer-content { grid-template-columns: repeat(2, 1fr); }
    @media (max-width: 480px) {
      .nav-links { display: none; }
      .footer-content { grid-template-columns: 1fr; }
    }
  
```

</style></head><body>
 <nav class="navbar">
 Trade
 <div class="nav-links">
 Dashboard
 Markets
 Trading
 Portfolio
 Analytics
 </div>
 <div class="auth-buttons">
 <button class="login-btn">Login</button>
 <button class="signup-btn">Sign Up</button>
 </div>
 </nav>
 <section class="hero">
 <h1>Professional Trading at Your Fingertips</h1>
 <p>Access real-time market data, advanced charting tools, and insightful analytics to make informed trading decisions.</p>
 <button class="get-started">Get Started Now</button>
 </section>
 <div class="dashboard">
 <div class="card price-chart">
 <h2>BTC/USD Price Chart</h2>
 <div id="price-chart-container" style="height: 320px;"></div>
 </div>
 <div class="card market-summary">
 <h2>Market Summary</h2>
 <table class="table">
 <thead>
 <tr>
 <th>Symbol</th>
 <th>Price</th>
 <th>24h Change</th>
 </tr>
 </thead>
 <tbody>
 <tr>
 <td>BTC/USD</td>
 <td>48,236.75</td>
 <td class="up">+2.34%</td>
 </tr>
 <tr>
 <td>ETH/USD</td>
 <td>3,152.89</td>
 <td class="up">+1.78%</td>
 </tr>
 <tr>
 <td>SOL/USD</td>
 <td>103.42</td>
 <td class="down">-0.86%</td>
 </tr>
 <tr>
 <td>XRP/USD</td>
 <td>0.5724</td>
 <td class="up">+0.43%</td>
 </tr>
 <tr>
 <td>DOT/USD</td>
 <td>6.87</td>
 <td class="down">-1.21%</td>
 </tr>
 </tbody>
 </table>
 </div>
 <div class="card volume-chart">
 <h2>Trading Volume</h2>
 <div id="volume-chart-container" style="height: 250px;"></div>
 </div>
 <div class="card technical-indicators">
 <h2>Technical Indicators</h2>
 <div id="indicators-chart-container" style="height: 250px;"></div>
 </div>
 <div class="card order-book">
 <h2>Order Book</h2>
 <div id="order-book-chart" style="height: 250px;"></div>
 </div>
 <div class="card recent-trades">
 <h2>Recent Trades</h2>
 <table class="table">
 <thead>
 <tr>
 <th>Amount</th>
 <th>Time</th>
 <th>Price</th>
 </tr>
 </thead>
 <tbody>
 <tr>
 <td>14:32:45</td>
 <td class="up">48,236.75</td>
 <td>0.0546</td>
 </tr>
 <tr>
 <td>14:32:35</td>
 <td class="down">48,231.20</td>
 <td>0.1250</td>
 </tr>
 <tr>
 <td>14:31:58</td>
 <td class="down">48,228.35</td>
 <td></td>
 </tr>
 </tbody>
 </table>
 </div>
 </div>

```

<td>0.0387</td>
</tr>
<tr>
<td>14:31:42</td>
<td>
class="up">48,240.60</td>
<td>0.2145</td>
</tr>
<tr>
<td>14:31:23</td>
<td class="up">48,245.15</td>
<td>0.0833</td>
</tr>
</tbody>
</table>
</div>
<div class="footer">
<div class="footer-section">
<h3>TradePro</h3>
<a href="#">About Us</a>
<a href="#">Careers</a>
<a href="#">Press</a>
<a href="#">Blog</a>
</div>
<div class="footer-section">
<h3>Products</h3>
<a href="#">Exchange</a>
<a href="#">Institutional Services</a>
<a href="#">API</a>
<a href="#">Mobile App</a>
</div>
<div class="footer-section">
<h3>Support</h3>
<a href="#">Help Center</a>
<a href="#">Contact Us</a>
<a href="#">FAQ</a>
<a href="#">Security</a>
</div>
<div class="footer-section">
<h3>Legal</h3>
<a href="#">Terms of Service</a>
<a href="#">Privacy Policy</a>
<a href="#">Risk Disclosure</a>
<a href="#">Cookie Policy</a>
</div>
</div>
<div class="copyright">
&copy; 2025 TradePro. All rights reserved.
</div>
</footer>
<script src="https://cdn.jsdelivr.net/npm/chart.js/3.9.1/chart.min.js"></script>
// Price Chart
const priceCtx = document.createElement('canvas');
document.getElementById('price-chart-container').appendChild(priceCtx);
const priceLabels = Array.from({length: 30}, (_, i) => i + 1);
const priceData = [46720, 46850, 47100, 46900, 47300, 47200, 47500, 47450, 47800, 47750, 47900, 47850, 48000, 48150, 48050, 48200, 48100, 48350, 48250, 48400, 48300, 48150, 48250, 48350, 48237, 48300, 48150, 48237, 48150, 48236];
new Chart(priceCtx, {
type: 'line',
data: {
labels: priceLabels,
datasets: [{
label: 'BTC/USD Price',
data: priceData,
borderColor: '#38bdf8',
backgroundColor: 'rgba(56, 189, 248, 0.1)',
tension: 0.4,
fill: true,
pointRadius: 0,
borderWidth: 2
}]
},
options: {
responsive: true,
maintainAspectRatio: false,
plugins: {
legend: {
display: false
}
},
tooltip: {
mode: 'index',
intersect: false
},
scales: {
x: {
display: true
},
y: {
display: true,
grid: {
display: false
}
}
},
grid: {
display: false
},
color: 'rgba(0, 0, 0, 0.05)'
});
// Volume Chart
const volumeCtx = document.createElement('canvas');
document.getElementById('volume-chart-container').appendChild(volumeCtx);
const volumeLabels = ['Jan', 'Feb', 'Mar', 'Apr', 'May', 'Jun', 'Jul', 'Aug', 'Sep', 'Oct', 'Nov', 'Dec'];
const volumeData = [850, 750, 820, 780, 880, 930, 920, 1020, 980, 925, 970, 1050];
new Chart(volumeCtx, {
type: 'bar',
data: {
labels: volumeLabels,
datasets: [{
label: 'Trading Volume (millions USD)',
data: volumeData,
borderRadius: 4,
backgroundColor: '#38bdf8'
}]
},
options: {
responsive: true,
maintainAspectRatio: false,
plugins: {
legend: {
display: false
}
},
scales: {
x: {
display: true
},
y: {
display: true,
grid: {
display: false
}
},
grid: {
display: false
},
color: 'rgba(0, 0, 0, 0.05)'
});
// Technical Indicators Chart
const indicatorsCtx = document.createElement('canvas');
document.getElementById('indicators-chart-container').appendChild(indicatorsCtx);
const indicatorLabels = Array.from({length: 20}, (_, i) => i + 1);
const priceDataShort = priceData.slice(-20);
const smaData = [];
const emaData = [];
// Calculate SMA and EMA
for (let i = 0; i < priceDataShort.length; i++) {
if (i >= 4) {

```



```

        // Simple 5-day moving average
        const sma =
(priceDataShort[i] + priceDataShort[i-1] + priceDataShort[i-2] +
priceDataShort[i-3] + priceDataShort[i-4]) / 5;
        smaData.push(sma);
    } else {
        smaData.push(null);
    }
    // Simple EMA
    approximation
    if (i === 0) {
        emaData.push(priceDataShort[0]);
    } else {
        const k = 2 / (10 + 1); // 10-day EMA factor
        const ema = priceDataShort[i] * k + emaData[i-1] * (1-k);
        emaData.push(ema);
    }
    new Chart(indicatorsCtx, {
        type:
'line',
        data: {
            labels: indicatorLabels,
            datasets: [
                {
                    label: 'Price',
                    data: priceDataShort,
                    borderColor: '#38bdf8',
                    borderWidth: 2,
                    tension: 0.1,
                    pointRadius: 0
                },
                {
                    label: 'SMA (5)',
                    data: smaData,
                    borderColor: '#ef4444',
                    borderWidth: 2,
                    tension: 0.1,
                    pointRadius: 0
                },
                {
                    label: 'EMA (10)',
                    data: emaData,
                    borderColor: '#10b981',
                    borderWidth: 2,
                    tension: 0.1,
                    pointRadius: 0
                }
            ],
            options: {
                responsive: true,
                maintainAspectRatio:
false,
                plugins: {
                    tooltip: {
                        mode: 'index',
                        intersect: false,
                        display: false
                    },
                    scales: {
                        x: {
                            grid: {
                                display: false
                            }
                        },
                        y: {
                            grid: {
                                color: 'rgba(0, 0, 0, 0.05)'
                            }
                        }
                    }
                }
            }
        },
        // Order Book Chart
        const orderBookCtx = document.createElement('canvas');
        document.getElementById('order-book-chart').appendChild(orderBookCtx);
        const askPrices = [48240, 48250, 48260, 48270, 48280, 48290, 48300, 48310,
48320, 48330];
        const askVolumes = [1.2, 2.5, 3.8, 4.2, 5.5, 6.1, 7.4, 8.2,
9.5, 10.8];
        const bidPrices = [48230, 48220, 48210, 48200, 48190,
48180, 48170, 48160, 48150, 48140];
        const bidVolumes = [1.5, 2.8, 3.6,
4.5, 5.3, 6.7, 7.9, 8.6, 9.3, 10.5];
        new Chart(orderBookCtx, {
            type: 'bar',
            data: {
                labels:
[...bidPrices.reverse(), ...askPrices],
                datasets: [
                    {
                        label: 'Bids',
                        data:
[...bidVolumes.reverse(), ...Array(askPrices.length).fill(null)],
                        backgroundColor: '#10b981'
                    },
                    {
                        label: 'Asks',
                        data:
[...Array(bidPrices.length).fill(null), ...askVolumes],
                        backgroundColor: '#ef4444'
                    }
                ],
                options: {
                    responsive: true,
                    maintainAspectRatio: false,
                    indexAxis: 'x',
                    plugins: {
                        tooltip: {
                            callbacks: {
                                title: function(tooltipItems) {
                                    return 'R' +
tooltipItems[0].label;
                                },
                                scales: {
                                    x: {
                                        stacked: true,
                                        display: false,
                                        ticks: {
                                            callback: function(value, index) {
                                                const allLabels =
[...bidPrices.reverse(), ...askPrices];
                                                // Show fewer ticks
                                                for readability
                                                if (index === 0 || index === 9 || index === 10
|| index === 19) {
                                                    return 'R' + allLabels[index];
                                                }
                                                return '';
                                            }
                                        },
                                        y: {
                                            stacked: true,
                                            grid: {
                                                color: 'rgba(0, 0, 0, 0.05)'
                                            },
                                            display: true,
                                            text: 'Volume'
                                        }
                                    }
                                }
                            }
                        }
                    }
                }
            }
        });
    </script>
</body>
</html>

```

demo\target\classes\static\manageUsers.html

```
<!-- manageUsers.html (already correct!) --><div class="dashboard-header"><div class="dashboard-title">Manage Users</div><div class="breadcrumb">Home > Manage Users</div></div><div class="data-card fade-in"><h2>User List</h2><div id="userLoading">Loading users...</div><table class="session-table" id="userTable" style="display: none;"><thead><tr><th>User ID</th><th>User Name</th><th>Role</th><th>Email</th><th>Phone</th></tr></thead><tbody id="userTableBody"><!-- User rows will be inserted here --></tbody></table><script>async function loadUsers() {const loadingDiv = document.getElementById("userLoading");const table = document.getElementById("userTable");const tbody = document.getElementById("userTableBody");loadingDiv.style.display = "block";table.style.display = "none";try {const response = await fetch("http://localhost:8080/api/all-users");if (!response.ok) throw new Error("Failed to fetch users");const users = await response.json();tbody.innerHTML = "";users.forEach(user => {const row = document.createElement("tr");row.innerHTML = `<td>${user.userId}</td><td>${user.userName}</td><td>${user.phone}</td><td>${user.email}</td><td>${user.role}</td>`;tbody.appendChild(row);});loadingDiv.style.display = "none";table.style.display = "table";} catch (error) {console.error("Error loading users:", error);loadingDiv.innerText = "Error loading user data.";}document.addEventListener("DOMContentLoaded", loadUsers);</script><style>* {margin: 0;padding: 0;box-sizing: border-box;font-family: 'Segoe UI', sans-serif;}body {display: flex;background: #f5f7fa;height: 100vh;overflow: hidden;}header {position: fixed;top: 0;left: 0;width: 100%;height: 60px;background: #2e3a59;color: white;display: flex;align-items: center;padding: 0 20px;z-index: 1000;box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);}user-role {font-size: 12px;color: #888;margin-top: 2px;}hamburger {background: none;border: none;color: white;font-size: 22px;cursor: pointer;margin-right: 15px;}header-title {font-size: 20px;flex: 1;}user-info-mini {display: flex;align-items: center;font-size: 16px;font-weight: 600;justify-content: center;gap: 10px;}user-info-mini .name {text-align: center;}sidebar {position: fixed;top: 60px;left: 0;width: 240px;height: calc(100% - 60px);background: #1c2536;color: #fff;transition: transform 0.3s ease-in-out;overflow-y: auto;z-index: 999;}sidebar.active {transform: translateX(-100%);}sidebar-header {padding: 20px;border-bottom: 1px solid #333;}user-info {display: flex;align-items: center;}user-avatar {background: #f39c12;color: white;font-weight: bold;width: 45px;height: 45px;border-radius: 50%;display: flex;align-items: center;justify-content: center;margin-right: 15px;font-size: 20px;animation: popIn 0.4s ease;}user-name {font-size: 16px;font-weight: bold;}user-status {font-size: 13px;color: lightgreen;}nav-menu {list-style: none;padding: 10px 0;}nav-menu li a {display: flex;align-items: center;padding: 12px 20px;color: #ccc;text-decoration: none;transition: background 0.3s;}nav-menu li a:hover,nav-menu li a.active {background: #34495e;color: #fff;}nav-menu i {margin-right: 12px;}logout-btn {width: 90%;margin: 20px auto;display: flex;align-items: center;justify-content: center;background: #e74c3c;color: white;border: none;padding: 12px;border-radius: 6px;font-size: 16px;cursor: pointer;transition: background 0.3s;}logout-btn:hover {background: #c0392b;}main-content {margin-left: 240px;padding: 80px 30px 30px 30px;width: 100%;overflow-y: auto;transition: margin-left 0.3s ease-in-out;}main-content.expanded {margin-left: 0;}dashboard-header {margin-bottom: 20px;animation: fadeInUp 0.5s ease;}dashboard-title {font-size: 28px;margin-bottom: 5px;}breadcrumb {color: #888;}
```

```

}data-card {background: white;padding: 25px;border-radius: 10px;box-
shadow: 0 3px 10px rgba(0,0,0,0.05);animation: fadeIn 1s ease;#loading {#
padding: 10px;color: #333;#session-table {width: 100%;border-collapse:
collapse;margin-top: 15px;#session-table th,#session-table td {padding: 12px;#
border: 1px solid #ddd;text-align: center;#session-table th {background:
#2e3a59;color: white;#session-table td {font-size: 15px;##* Notification
Icon */notification-icon {cursor: pointer;font-size: 22px;position: relative;#
}notificationCount {background: red;color: white;padding: 2px 6px;border-
radius: 50%;position: absolute;#op: -8px;#ight: -8px;font-size: 12px;##*
Notifications */notification-panel {display: none;position: fixed;#op: 50px;#
right: 20px;background: white;border-radius: 8px;#idth: 320px;box-shadow: 0px
4px 10px rgba(0, 0, 0, 0.2);padding: 15px;#notification-panel ul {list-
style: none;padding: 0;#notification-panel ul li {padding: 10px;border-
bottom: 1px solid #ddd;cursor: pointer;##keyframes fadeIn {from {opacity: 0;
transform: translateY(10px);}to {opacity: 1; transform: translateY(0);}##
@keyframes fadeInUp {from {opacity: 0; transform: translateY(20px);}to
{opacity: 1; transform: translateY(0);}##keyframes popIn {0% { transform:
scale(0); opacity: 0; }100% { transform: scale(1); opacity: 1; }##*
Responsive */@media (max-width: 768px) {#sidebar {# transform:
translateX(-100%);#sidebar.active {# transform: translateX(0);#main-
content {# margin-left: 0;# padding: 70px 20px;##/style>#/style>#

```

demo\target\classes\static\modules\manageUsers.html

```
<!-- manageUsers.html (already correct!) --><div class="dashboard-header"><div class="dashboard-title">Manage Users</div><div class="breadcrumb">Home > Manage Users</div></div><div class="data-card fade-in"><h2>User List</h2><div id="userLoading">Loading users...</div><table class="session-table" id="userTable" style="display: none;"><thead><tr><th>User ID</th><th>User Name</th><th>Role</th><th>Email</th><th>Phone</th></tr></thead><tbody id="userTableBody"><!-- User rows will be inserted here --></tbody></table><script>async function loadUsers() {const loadingDiv = document.getElementById("userLoading");const table = document.getElementById("userTable");const tbody = document.getElementById("userTableBody");loadingDiv.style.display = "block";table.style.display = "none";try {const response = await fetch("http://localhost:8080/api/all-users");if (!response.ok) throw new Error("Failed to fetch users");const users = await response.json();tbody.innerHTML = "";users.forEach(user => {const row = document.createElement("tr");row.innerHTML = `<td>${user.userId}</td><td>${user.userName}</td><td>${user.phone}</td><td>${user.email}</td><td>${user.role}</td>`;tbody.appendChild(row);});loadingDiv.style.display = "none";table.style.display = "table";} catch (error) {console.error("Error loading users:", error);loadingDiv.innerText = "Error loading user data.";}document.addEventListener("DOMContentLoaded", loadUsers);</script><style>* {margin: 0;padding: 0;box-sizing: border-box;font-family: 'Segoe UI', sans-serif;}body {display: flex;background: #f5f7fa;height: 100vh;overflow: hidden;}header {position: fixed;top: 0;left: 0;width: 100%;height: 60px;background: #2e3a59;color: white;display: flex;align-items: center;padding: 0 20px;z-index: 1000;box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);}user-role {font-size: 12px;color: #888;margin-top: 2px;}hamburger {background: none;border: none;color: white;font-size: 22px;cursor: pointer;margin-right: 15px;}header-title {font-size: 20px;flex: 1;}user-info-mini {display: flex;align-items: center;font-size: 16px;font-weight: 600;justify-content: center;gap: 10px;}user-info-mini .name {text-align: center;}sidebar {position: fixed;top: 60px;left: 0;width: 240px;height: calc(100% - 60px);background: #1c2536;color: #fff;transition: transform 0.3s ease-in-out;overflow-y: auto;z-index: 999;}sidebar.active {transform: translateX(-100%);}sidebar-header {padding: 20px;border-bottom: 1px solid #333;}user-info {display: flex;align-items: center;}user-avatar {background: #f39c12;color: white;font-weight: bold;width: 45px;height: 45px;border-radius: 50%;display: flex;align-items: center;justify-content: center;margin-right: 15px;font-size: 20px;animation: popIn 0.4s ease;}user-name {font-size: 16px;font-weight: bold;}user-status {font-size: 13px;color: lightgreen;}nav-menu {list-style: none;padding: 10px 0;}nav-menu li a {display: flex;align-items: center;padding: 12px 20px;color: #ccc;text-decoration: none;transition: background 0.3s;}nav-menu li a:active {background: #34495e;color: #fff;}nav-menu i {margin-right: 12px;}logout-btn {width: 90%;margin: 20px auto;display: flex;align-items: center;justify-content: center;background: #e74c3c;color: white;border: none;padding: 12px;border-radius: 6px;font-size: 16px;cursor: pointer;transition: background 0.3s;}logout-btn:hover {background: #c0392b;}main-content {margin-left: 240px;padding: 80px 30px 30px 30px;width: 100%;overflow-y: auto;transition: margin-left 0.3s ease-in-out;}main-content.expanded {margin-left: 0;}dashboard-header {margin-bottom: 20px;animation: fadeInUp 0.5s ease;}dashboard-title {font-size: 28px;margin-bottom: 5px;}breadcrumb {color: #888;}
```

```

}data-card {background: white;padding: 25px;border-radius: 10px;box-
shadow: 0 3px 10px rgba(0,0,0,0.05);animation: fadeIn 1s ease;#loading {#
padding: 10px;color: #333;#session-table {width: 100%;border-collapse:
collapse;margin-top: 15px;#session-table th,#session-table td {padding: 12px;#
border: 1px solid #ddd;text-align: center;#session-table th {background:
#2e3a59;color: white;#session-table td {font-size: 15px;##* Notification
Icon */notification-icon {cursor: pointer;font-size: 22px;position: relative;#
}notificationCount {background: red;color: white;padding: 2px 6px;border-
radius: 50%;position: absolute;#top: -8px;#right: -8px;font-size: 12px;##*
Notifications */notification-panel {display: none;position: fixed;#top: 50px;#
right: 20px;background: white;border-radius: 8px;#width: 320px;box-shadow: 0px
4px 10px rgba(0, 0, 0, 0.2);padding: 15px;#notification-panel ul {list-
style: none;padding: 0;#notification-panel ul li {padding: 10px;border-
bottom: 1px solid #ddd;cursor: pointer;##keyframes fadeIn {from {opacity: 0;
transform: translateY(10px);}to {opacity: 1; transform: translateY(0);}##
@keyframes fadeInUp {from {opacity: 0; transform: translateY(20px);}to
{opacity: 1; transform: translateY(0);}##keyframes popIn {0% { transform:
scale(0); opacity: 0; }100% { transform: scale(1); opacity: 1; }##*
Responsive */@media (max-width: 768px) {#sidebar {# transform:
translateX(-100%);#sidebar.active {# transform: translateX(0);#main-
content {# margin-left: 0;# padding: 70px 20px;##/style>#/style>#

```

demo\target\classes\static\signin.html

```

<title>Signin Page</title><div class="container"><div class="text-overlay"></div><div class="image-section"></div><div class="form-section"><div class="form-box"><h2>Welcome Back!</h2><p>Login to Continue</p><div class="input-group"><label for="userid">Username</label><input type="text" id="userid" name="userid" placeholder="Enter Username, Email, or Phone" required><div class="error" id="useridError"></div></div><div class="input-group"><label for="password">Password</label><input type="password" id="password" name="password" placeholder="Enter Password" required><div class="error" id="passwordError"></div></div><div class="remember-me"><input type="checkbox" id="rememberMe" /><label for="rememberMe">Remember Me</label><h5><a href="forgot.html" class="forgot">Forgot Password?</a></h5></div><button type="submit" class="btn">Sign In</button><div id="result"></div><h5>Don't have an account? <a class="signup" href="signup.html">Sign Up</a></h5></form></div></div></div><script>document.getElementById("signinForm").addEventListener("submit", async function (event) {event.preventDefault(); // Prevent default form submissionlet enteredValue = document.getElementById("userid").value.trim();let enteredPassword = document.getElementById("password").value.trim();let resultDiv = document.getElementById("result"); // Clear previous errorsdocument.getElementById("useridError").textContent = "";document.getElementById("passwordError").textContent = "";resultDiv.textContent = "";if (enteredValue === "" || enteredPassword === "") {resultDiv.textContent = "Please fill in all fields.";resultDiv.style.color = "red";return;}let response = await fetch("http://localhost:8080/api/signin", {method: "POST", headers: {"Content-Type": "application/json"}, body: JSON.stringify({userid: enteredValue, password: enteredPassword})});let result = await response.json();if (result.message === "Login successful") {resultDiv.textContent = "Login successful! Redirecting...";resultDiv.style.color = "green"; // Store data in sessionStoragesessionStorage.setItem("userId", result.userId);sessionStorage.setItem("userName", result.userName);localStorage.setItem("userId", result.userId);localStorage.setItem("userName", result.userName);localStorage.setItem("loggedInUser", enteredValue);setTimeout(() => {window.location.href = "user_dashboard3.html"; // Redirect to dashboard}, 1500);} else {resultDiv.textContent = result.message;resultDiv.style.color = "red";}catch (error) {console.error("Error:", error);resultDiv.textContent = "Server error. Please try again later.";resultDiv.style.color = "red";} }) * {margin: 0; padding: 0; box-sizing: border-box; font-family: "Poppins", sans-serif;} body {display: flex; justify-content: center; align-items: center; height: 100vh; background: #ffff; } .container {display: flex; width: 100%; max-width: 1700px; height: 700px; box-shadow: 0px 4px 10px rgba(0, 0, 0, 0.1);}

```

```

border-radius: 15px;@ overflow: hidden;@ @
background-color: #f0f5ff;@ background-image: @
radial-gradient(circle at 100% 150%, #f0f5ff 24%, #c8d7ff 25%, #c8d7ff 28%,
#f0f5ff 29%, #f0f5ff 36%, #c8d7ff 36%, #c8d7ff 40%, transparent 40%,
transparent),@ radial-gradient(circle at 0 150%, #f0f5ff 24%,
#c8d7ff 25%, #c8d7ff 28%, #f0f5ff 29%, #f0f5ff 36%, #c8d7ff 36%, #c8d7ff 40%,
transparent 40%, transparent),@ radial-gradient(circle at 50%
100%, #c8d7ff 10%, #f0f5ff 11%, #f0f5ff 23%, #c8d7ff 24%, #c8d7ff 30%,
#f0f5ff 31%, #f0f5ff 43%, #c8d7ff 44%, #c8d7ff 50%, #f0f5ff 51%, #f0f5ff 63%,
#c8d7ff 64%, #c8d7ff 71%, transparent 71%, transparent),@
radial-gradient(circle at 100% 50%, #c8d7ff 5%, #f0f5ff 6%, #f0f5ff 15%,
#c8d7ff 16%, #c8d7ff 20%, #f0f5ff 21%, #f0f5ff 30%, #c8d7ff 31%, #c8d7ff 35%,
#f0f5ff 36%, #f0f5ff 45%, #c8d7ff 46%, #c8d7ff 49%, transparent 50%,
transparent),@ radial-gradient(circle at 0% 50%, #c8d7ff 5%,
#f0f5ff 6%, #f0f5ff 15%, #c8d7ff 16%, #c8d7ff 20%, #f0f5ff 21%, #f0f5ff 30%,
#c8d7ff 31%, #c8d7ff 35%, #f0f5ff 36%, #f0f5ff 45%, #c8d7ff 46%, #c8d7ff 49%,
transparent 50%, transparent);@ background-size: 100px 50px;@ @
@@@ $$$ @ @ @.image-section {@ width: 700px;@ @
flex: 1;@ @ background: url("pics/unnamed\ flaticon.gif") no-
repeat center / auto ;@ @ margin-top: 100px;@ margin-bottom:
100px;@ margin-left: 80px;@ border-bottom-left-radius: 5%;@
border-top-left-radius: 5%;@ box-shadow: -5px 0 10px -2px rgba(0, 0, 0,
0.4);@ }@ @.form-section {@ width: 50%;@ @ display: flex;@
align-items: center;@ justify-content: center;@ @
backdrop-filter: blur(10px);@ margin-top: 100px;@ margin-
bottom: 100px;@ margin-right: 80px;@ @ }@ @.form-box {@ @
background: white;@ padding: 80px;@ width: 100%;@ height:
100%;@ @ @ text-align: center;@ border-top-right-radius:
5%;@ border-bottom-right-radius: 5%;@ box-shadow: 5px 0 4px -2px
rgba(0, 0, 0, 0.3);@ @ }@ @ h2 {@ font-family: 'Times New Roman',
Times, serif;@ text-align: center;@ font-size: 40px;@ font-
weight: bold;@ @ color: black;@ text-shadow: 0px 4px 10px
rgba(0, 0, 0, 0.1);@ @ }@ @ p {@ font-family: 'Times New Roman',
Times, serif;@ text-align: center;@ font-size: 20px;@ font-
weight: bold;@ color: black;@ }@ @ h4 {@ font-family: 'Franklin
Gothic Medium', 'Arial Narrow', Arial, sans-serif;@ color: black;@
font-size: 19px;@ font-weight: bold;@ padding-bottom:
20px;@ @ }@ @ h5 {@ color: black;@ font-weight: bold;@
font-size: 15px;@ }@ @.signup {@ text-decoration: none;@ color:
red;@ }@ @.signup :hover {@ text-decoration: underline;@
transform: scale(1.05);@ color: red;@ }@ @.input-group {@ @
margin-bottom: 15px;@ text-align: left;@ }@ @ label {@ font-
family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida Sans
Unicode', Geneva, Verdana, sans-serif;@ padding-bottom: 10px;@
font-weight: bold;@ letter-spacing: 1px;@ display: block;@ @
font-size: 16px;@ color: black;@ }@ @ input {@ font-
family: 'Gill Sans', 'Gill Sans MT', Calibri, 'Trebuchet MS', sans-serif;@
width: 100%;@ padding: 12px 15px;@ border: none;@
border-radius: 6px;@ font-size: 16px;@ border: 2px solid #ccc;@
color: black;@ }@ @.input-type {@ color:
black;@ }@ @.input-group input:hover {@ border-color: #667eea;@
box-shadow: 0px 0px 10px rgba(102, 126, 234, 0.5);@ transform:
scale(1.02);@ }@ @ input:focus {@ outline: none;@ transform:
scale(1.05);@ box-shadow: 0px 0px 10px rgba(255, 255, 0.4);@ }@ @
.error {@ color: red;@ font-size: 14px;@ margin-top:
5px;@ }@ @.remember-me {@ display: flex;@ @ align-
items: center;@ @ }@ @ input[type=checkbox] {@ -webkit-
appearance: checkbox;@ -moz-appearance: checkbox;@ appearance:
checkbox;@ display: inline-block;@ width: auto;@ }@ @.remember-
me input {@

```

```

        margin-right: 8px;@        cursor: pointer;@        left: 0;@
    @ }@ .btn {@        width: 100%;@        background: linear-
gradient(to right, #667eea, #764ba2);@        color: white;@        padding:
14px;@        border: none;@        border-radius: 8px;@        font-size: 18px;@
        cursor: pointer;@        transition: all 0.3s ease-in-out;@        font-
weight: bold;@        text-transform: uppercase;@        letter-spacing: 1px;@
        box-shadow: 0px 5px 15px rgba(102, 126, 234, 0.3);@ }@ @
    @ @ .btn:hover {@        background: linear-gradient(to right,
#764ba2, #667eea);@        transform: scale(1.05);@        box-shadow: 0px 8px
20px rgba(102, 126, 234, 0.5);@ }@ #result {@        font-weight: bold;@
        margin-top: 15px;@        color: #00ff99;@ }@ .forgot {@
padding-left: 230px;@        text-decoration: none;@        color: blue;@ }@
.forgot :hover{@        background: #ff4e88;@        transform:
scale(1.05);@ }@ @ @ @media (max-width: 768px) {@        .container {@
        flex-direction: column;@        height: auto;@        }@
        .image-section {@        height: 200px;@        }@        .form-
section {@        height: auto;@        padding: 20px;@        }@
        .form-box {@        width: 90%;@        }@ }@ .text-overlay h2
{position: absolute;@top: 6%;@left: 25%;@transform: translate(-50%, -50%);@text-
align: center;@color: black;@padding: 20px;@border-radius: 10px;@}@text-overlay
h2{font-weight: bold;@font-size: 24px;@text-shadow: 0 0 1px #000000, 0 0 1px
#0e1213, 0 0 1px #131616;@animation: glowFade 2.5s ease-in-out infinite;@
display: inline-block;@}@text-overlay p {position: absolute;@top: 94%;@left: 26%;@
transform: translate(-50%, -50%);@text-align: center;@color: black;@padding:
20px;@border-radius: 10px;@}@text-overlay p{font-weight: bold;@font-size: 20px;@
text-shadow: 0 0 1px #080a0a, 0 0 1px #101214, 0 0 1px #121416;@animation:
glowFade 2.5s ease-in-out infinite;@display: inline-block;@}@keyframes
glowFade {@0%, 100% {@opacity: 0.8;@text-shadow: 0 0 5px #ffffff;@}@
50% {@opacity: 1;@text-shadow: 0 0 15px #00ffff;@}@}@/style>@

```


demo\target\classes\static\signin2.html

```
<!DOCTYPE html>ðhtml lang="en">ðhead>ð <meta charset="UTF-8">ð <meta
name="viewport" content="width=device-width, initial-scale=1.0">ð
<title>Creative Authentication System</title>ð <style>ð :root {ð
  --primary: #4776E6;ð --secondary: #8E54E9;ð --
light: #f8f9fa;ð --dark: #343a40;ð --success: #28a745;ð
  --error: #dc3545;ð --transition: all 0.4s ease;ð }ð
  * {ð box-sizing: border-box;ð margin: 0;ð
padding: 0;ð font-family: 'Segoe UI', Tahoma, Geneva,
Verdana, sans-serif;ð }ð body, html {ð height: 100%;ð
overflow: hidden;ð background-color: var(--light);ð
}ð .navbar {ð position: fixed;ð display:
flex;ð justify-content: space-between;ð padding: 15px
30px;ð background: linear-gradient(135deg, var(--primary), var(--
secondary));ð border-radius: 30px;ð top: 20px;ð
left: 50%;ð transform: translateX(-50%);ð width: 90%;ð
max-width: 600px;ð z-index: 1000;ð box-
shadow: 0 10px 30px rgba(0, 0, 0, 0.1);ð }ð .nav-item {ð
color: white;ð text-align: center;ð padding:
10px 20px;ð text-decoration: none;ð font-weight: 500;ð
border-radius: 20px;ð transition: var(--transition);ð
}ð .nav-item:hover {ð background-color: rgba(255,
255, 255, 0.2);ð transform: translateY(-3px);ð }ð .auth-
container {ð position: relative;ð width: 100%;ð
height: 100vh;ð overflow: hidden;ð display: flex;ð
align-items: center;ð justify-content: center;ð
background: linear-gradient(135deg, #f5f7fa, #c3cfe2);ð }ð
.form-section {ð position: relative;ð width:
100%;ð height: 100%;ð transition: transform 0.8s cubic-
bezier(0.68, -0.55, 0.265, 1.55);ð }ð .form-section {ð
position: absolute;ð width: 100%;ð height: 100%;ð
display: flex;ð flex-direction: column;ð
justify-content: center;ð align-items: center;ð
transition: var(--transition);ð opacity: 0;ð visibility:
hidden;ð }ð .form-section.active {ð opacity: 1;ð
visibility: visible;ð }ð .form-card {ð
width: 90%;ð max-width: 450px;ð padding: 40px;ð
background: white;ð border-radius: 20px;ð box-shadow: 0
15px 35px rgba(0, 0, 0, 0.1);ð transform: translateY(50px);ð
opacity: 0;ð transition: var(--transition);ð }ð
.form-section.active .form-card {ð transform: translateY(0);ð
opacity: 1;ð }ð .form-header {ð text-align:
center;ð margin-bottom: 30px;ð }ð .form-title {ð
font-size: 28px;ð color: var(--dark);ð margin-
bottom: 10px;ð font-weight: 600;ð }ð .form-subtitle {ð
color: #6c757d;ð font-size: 16px;ð }ð .form-
group {ð margin-bottom: 25px;ð position: relative;ð
}ð .form-group label {ð position: absolute;ð
top: 0;ð left: 0;ð font-size: 16px;ð
color: #6c757d;ð padding: 18px 15px;ð pointer-
events: none;ð transition: var(--transition);ð }ð .form-
group input {ð width: 100%;ð padding: 18px 15px;ð
font-size: 16px;ð border: none;ð border-
bottom: 2px solid #e9ecef;ð background-color: transparent;ð
outline: none;ð transition: var(--transition);ð }ð
.form-group input:focus,ð .form-group input:valid {ð
border-bottom-color: var(--primary);ð }ð .form-group input:focus
+ label,ð .form-group input:valid + label {ð top: -25px;ð
left: 0;ð font-size: 14px;ð color: var(--
primary);ð
```

```

    }@ .error-message {@ color: var(--error);@
    font-size: 14px;@ margin-top: 5px;@ display:
none;@ }@ .submit-btn {@ width: 100%;@
padding: 15px;@ background: linear-gradient(135deg, var(--primary),
var(--secondary));@ color: white;@ border: none;@
border-radius: 30px;@ font-size: 16px;@ font-
weight: 600;@ cursor: pointer;@ transition: var(--
transition);@ box-shadow: 0 5px 15px rgba(0, 0, 0, 0.1);@ }@
.submit-btn:hover {@ transform: translateY(-3px);@
box-shadow: 0 8px 25px rgba(0, 0, 0, 0.15);@ }@ .form-
footer {@ text-align: center;@ margin-top: 25px;@
font-size: 15px;@ color: #6c757d;@ }@ .form-
link {@ color: var(--primary);@ text-decoration: none;@
font-weight: 600;@ transition: var(--transition);@
}@ .form-link:hover {@ color: var(--secondary);@
}@ .bg-shapes {@ position: absolute;@
width: 100%;@ height: 100%;@ overflow: hidden;@
z-index: -1;@ }@ .shape {@ position: absolute;@
border-radius: 50%;@ background: linear-
gradient(135deg, rgba(71, 118, 230, 0.1), rgba(142, 84, 233, 0.1));@
animation: float 10s infinite ease-in-out;@ }@
.shape:nth-child(1) {@ width: 300px;@ height:
300px;@ top: -100px;@ right: -100px;@ animation-
delay: 0s;@ }@ .shape:nth-child(2) {@ width: 200px;@
height: 200px;@ bottom: 50px;@ left: 30px;@
animation-delay: 2s;@ }@ .shape:nth-child(3) {@
width: 150px;@ height: 150px;@ bottom: 150px;@
right: 100px;@ animation-delay: 4s;@ }@
.shape:nth-child(4) {@ width: 100px;@ height:
100px;@ top: 120px;@ left: 150px;@ animation-
delay: 6s;@ }@ @keyframes float {@ 0% {@
transform: translate(0, 0) rotate(0deg);@ }@ 50% {@
transform: translate(20px, 20px) rotate(10deg);@ }@
100% {@ transform: translate(0, 0) rotate(0deg);@
}@ }@ .success-message {@ display: none;@
position: fixed;@ top: 20px;@ left: 50%;@
transform: translateX(-50%);@ background-color: var(--
success);@ color: white;@ padding: 15px 25px;@
border-radius: 30px;@ box-shadow: 0 5px 15px rgba(0, 0, 0, 0.2);@
z-index: 1000;@ animation: slideDown 0.5s forwards;@
}@ @keyframes slideDown {@ from {@ top:
-50px;@ opacity: 0;@ }@ to {@ top:
20px;@ opacity: 1;@ }@ }@ .progress-
bar {@ position: absolute;@ bottom: 0;@ left: 0;@
height: 5px;@ background: linear-gradient(90deg, var(--
primary), var(--secondary));@ transition: width 0.5s ease;@ }@
.indicator-container {@ display: flex;@ justify-
content: center;@ margin-top: 20px;@ }@ .form-indicator
{@ width: 10px;@ height: 10px;@ border-radius:
50%;@ background-color: #e9ecef;@ margin: 0 5px;@
transition: var(--transition);@ }@ .form-
indicator.active {@ background: linear-gradient(135deg, var(--
primary), var(--secondary));@ transform: scale(1.3);@ }@
/* Social Login Buttons */@ .social-login {@ display:
flex;@ justify-content: center;@ margin: 20px 0;@
gap: 15px;@ }@ .social-btn {@ display: flex;@
justify-content: center;@ align-items: center;@
width: 50px;@ height: 50px;@ border-radius:
50%;@ background-color: white;@ border: 1px solid #e9ecef;@
cursor: pointer;@

```

```

        transition: var(--transition);
        box-shadow: 0 3px 10px rgba(0, 0, 0, 0.1);
    }
    .social-btn:hover {
        transform: translateY(-3px);
        box-shadow: 0 5px 15px rgba(0, 0, 0, 0.15);
    }
    .social-icon {
        width: 24px;
        height: 24px;
    }
    .divider {
        display: flex;
        align-items: center;
        margin: 25px 0;
    }
    .divider-line {
        flex-grow: 1;
        height: 1px;
        background-color: #e9ecef;
    }
    .divider-text {
        margin: 0 15px;
        color: #6c757d;
        font-size: 14px;
    }
</style></head><body>
    <!-- Success Message -->
    <div class="success-message" id="success-message">
        <!-- Background Shapes -->
        <div class="bg-shapes">
            <div class="shape"></div>
            <div class="shape"></div>
            <div class="shape"></div>
            <div class="shape"></div>
        </div>
        <!-- Forms Container -->
        <div class="forms-container">
            <!-- Sign In Form -->
            <div class="form-section" id="signin-section">
                <div class="form-card">
                    <div class="form-header">
                        <h2 class="form-title">Welcome Back</h2>
                        <p class="form-subtitle">Sign in to continue your journey</p>
                    </div>
                    <div class="divider">
                        <div class="divider-line"></div>
                        <span class="divider-text">OR</span>
                        <div class="divider-line"></div>
                    </div>
                    <form id="signin-form">
                        <div class="form-group">
                            <input type="email" id="signin-email" required>
                            <label for="signin-email">Email Address</label>
                            <div class="error-message" id="signin-email-error">Please enter a valid email address</div>
                        </div>
                        <div class="form-group">
                            <input type="password" id="signin-password" required>
                            <label for="signin-password">Password</label>
                            <div class="error-message" id="signin-password-error">Password must be at least 6 characters</div>
                        </div>
                        <button type="submit" class="submit-btn">Sign In</button>
                    </form>
                    <div class="form-footer">
                        <a href="#" class="form-link" id="signin-to-forgot">Forgot Password?</a>
                        <div class="form-footer">
                            Don't have an account? <a href="#" class="form-link" id="signin-to-signup">Sign Up</a>
                        </div>
                    </div>
                    <div class="indicator-container">
                        <div class="form-indicator active"></div>
                        <div class="form-indicator"></div>
                    </div>
                </div>
            </div>
            <!-- Sign Up Form -->
            <div class="form-section" id="signup-section">
                <div class="form-card">
                    <div class="form-header">
                        <h2 class="form-title">Create Account</h2>
                    </div>
                    <form id="signup-form">
                        <div class="form-group">
                            <input type="text" id="signup-name" required>
                            <label for="signup-name">Full Name</label>
                            <div class="error-message" id="signup-name-error">Please enter your name</div>
                        </div>
                        <div class="form-group">
                            <input type="email" id="signup-email" required>
                            <label for="signup-email">Email Address</label>
                            <div class="error-message" id="signup-email-error">Please enter a valid email address</div>
                        </div>
                    </form>
                </div>
            </div>
        </div>
    </div>
</body>
</html>

```

```

</div>ð
        <div class="form-group">ð
<input type="password" id="signup-password" required>ð
        <label for="signup-password">Password</label>ð
        <div class="error-message" id="signup-password-
error">Password must be at least 8 characters with letters and numbers</div>ð
        </div>ð        <div class="form-group">ð
        <input type="password" id="signup-confirm"
required>ð
        <label for="signup-confirm">Confirm
Password</label>ð        <div class="error-message"
id="signup-confirm-error">Passwords do not match</div>ð
</div>ð        <button type="submit" class="submit-btn">Create
Account</button>ð        </form>ð        <div
class="form-footer">ð        Already have an account? <a
href="#" class="form-link" id="signup-to-signin">Sign In</a>ð
        </div>ð        <div class="indicator-
container">ð        <div class="form-indicator"></div>ð
        <div class="form-indicator active"></div>ð
        <div class="form-indicator"></div>ð
</div>ð        </div>ð        </div>ð        <!-- Forgot
Password Form -->ð        <div class="form-section" id="forgot-section">ð
        <div class="form-card">ð        <div class="form-
header">ð        <h2 class="form-title">Reset Password</h2>ð
        <p class="form-subtitle">We'll send you a reset link</
p>ð        </div>ð        <form id="forgot-form">ð
        <div class="form-group">ð
<input type="email" id="forgot-email" required>ð
<label for="forgot-email">Email Address</label>ð
<div class="error-message" id="forgot-email-error">Please enter a valid email
address</div>ð        </div>ð        <button
type="submit" class="submit-btn">Send Reset Link</button>ð
</form>ð        <div class="form-footer">ð
Remember your password? <a href="#" class="form-link" id="forgot-to-
signin">Sign In</a>ð        </div>ð        <div
class="indicator-container">ð        <div class="form-
indicator"></div>ð        <div class="form-indicator"></div>ð
        <div class="form-indicator active"></div>ð
        </div>ð        </div>ð        </div>ð
<!-- Progress Bar -->ð        <div class="progress-bar" id="progress-
bar"></div>ð        </div>ð        </div>ð        <script>ð        // Form Elementsð
        const signinSection = document.getElementById('signin-section');ð
        const signupSection = document.getElementById('signup-section');ð
        const forgotSection = document.getElementById('forgot-section');ð
        const progressBar = document.getElementById('progress-bar');ð
const successMessage = document.getElementById('success-message');ð        ð
        // Navigationð        document.getElementById('signin-to-
signup').addEventListener('click', function(e) {ð
e.preventDefault();ð        slideToForm('signup');ð        });ð        ð
        document.getElementById('signup-to-signin').addEventListener('click',
function(e) {ð        e.preventDefault();ð        slideToForm('signin');ð
        });ð        ð        document.getElementById('signin-to-
forgot').addEventListener('click', function(e) {ð
e.preventDefault();ð        slideToForm('forgot');ð        });ð        ð
        document.getElementById('forgot-to-signin').addEventListener('click',
function(e) {ð        e.preventDefault();ð        slideToForm('signin');ð
        });ð        ð        // Form Switching Functionð        function
slideToForm(formName) {ð        // Remove active class from all sectionsð
        signinSection.classList.remove('active');ð
        signupSection.classList.remove('active');ð
        forgotSection.classList.remove('active');ð

```

```

                                ð // Update
indicatorsð const indicators = document.querySelectorAll('.form-
indicator');ð indicators.forEach(ind =>
ind.classList.remove('active'));ð // Set active form
and update progress barð if (formName === 'signin') {ð
    signinSection.classList.add('active');ð
indicators[0].classList.add('active');ð progressBar.style.width
= '33.3%';ð } else if (formName === 'signup') {ð
signupSection.classList.add('active');ð
indicators[1].classList.add('active');ð progressBar.style.width
= '66.6%';ð } else if (formName === 'forgot') {ð
forgotSection.classList.add('active');ð
indicators[2].classList.add('active');ð progressBar.style.width
= '100%';ð }ð // Initializedð
window.addEventListener('load', function() {ð slideToForm('signin');ð
    });ð // Form Validationsð
document.getElementById('signin-form').addEventListener('submit', function(e)
{ð e.preventDefault();ð let isValid = true;ð
    ð // Email validationð const email =
document.getElementById('signin-email').value;ð if (!
validateEmail(email)) {ð document.getElementById('signin-email-
error').style.display = 'block';ð isValid = false;ð }
else {ð document.getElementById('signin-email-
error').style.display = 'none';ð }ð //
Password validationð const password =
document.getElementById('signin-password').value;ð if
(password.length < 6) {ð document.getElementById('signin-
password-error').style.display = 'block';ð isValid = false;ð
    } else {ð document.getElementById('signin-password-
error').style.display = 'none';ð }ð if
(isValid) {ð showSuccessMessage('Signed in successfully!');ð
    // Here you would typically send the form data to your serverð
    }ð });ð document.getElementById('signup-
form').addEventListener('submit', function(e) {ð e.preventDefault();ð
    ð let isValid = true;ð // Name
validationð const name = document.getElementById('signup-
name').value;ð if (name.trim() === '') {ð
document.getElementById('signup-name-error').style.display = 'block';ð
    isValid = false;ð } else {ð
document.getElementById('signup-name-error').style.display = 'none';ð
    }ð // Email validationð const
email = document.getElementById('signup-email').value;ð if (!
validateEmail(email)) {ð document.getElementById('signup-email-
error').style.display = 'block';ð isValid = false;ð }
else {ð document.getElementById('signup-email-
error').style.display = 'none';ð }ð //
Password validationð const password =
document.getElementById('signup-password').value;ð if
(password.length < 8 || ![a-zA-Z]/.test(password) || ![
0-9]/.test(password)) {ð document.getElementById('signup-
password-error').style.display = 'block';ð isValid = false;ð
    } else {ð document.getElementById('signup-password-
error').style.display = 'none';ð }ð //
Password confirmationð const confirm =
document.getElementById('signup-confirm').value;ð if (confirm !==
password) {ð document.getElementById('signup-confirm-
error').style.display = 'block';ð isValid = false;ð }
else {ð document.getElementById('signup-confirm-
error').style.display = 'none';ð }ð if
(isValid) {ð

```

```

        showSuccessMessage('Account created
successfully!');ð
        // Here you would typically send the form
data to your serverð
        setTimeout(() => {ð
slideToForm('signin');ð
        }, 2000);ð
        })ð
        document.getElementById('forgot-
form').addEventListener('submit', function(e) {ð
        e.preventDefault();ð
        let isValid = true;ð
        // Email
validationð
        const email = document.getElementById('forgot-
email').value;ð
        if (!validateEmail(email)) {ð
document.getElementById('forgot-email-error').style.display = 'block';ð
        isValid = false;ð
        } else {ð
document.getElementById('forgot-email-error').style.display = 'none';ð
        }ð
        if (isValid) {ð
showSuccessMessage('Password reset link sent! Check your email.');

```

demo\target\classes\static\signin3.html

```
<!DOCTYPE html>ðhtml lang="en"ðheadð <meta charset="UTF-8" />ð <meta
name="viewport" content="width=device-width, initial-scale=1.0" />ð
<title>Signin/Signup Page</title>ð <style>ð * {ð margin: 0;ð
padding: 0;ð box-sizing: border-box;ð font-family: "Poppins", sans-
serif;ð }ð body {ð display: flex;ð justify-content: center;ð
align-items: center;ð ð background: #fff;ð }ð .container {ð
display: flex;ð width: 100%;ð max-width: 1400px;ð height: 850px;ð
background: lightgoldenrodyellow;ð box-shadow: 0px 4px 10px rgba(0,
0, 0, 0.3);ð border-radius: 15px;ð overflow: hidden;ð }ð .image-
section {ð width: 50%;ð background: url("pics/art-rachen-sM4r-swmcoY-
unsplash-Photoroom.png") no-repeat center center/cover;ð }ð .form-section
{ð height: 850px;ð width: 50%;ð background: white;ð ð
display: flex;ð align-items: center;ð justify-content: center;ð }ð
.form-box {ð width: 100%;ð max-width: 400px;ð display: none;ð
}ð .form-box.active {ð display: block;ð }ð h2 {ð font-size:
32px;ð font-weight: bold;ð color: black;ð margin-bottom: 20px;ð
text-align: center;ð }ð .input-group {ð margin-bottom: 15px;ð
text-align: left;ð }ð label {ð font-weight: bold;ð font-
size: 16px;ð color: black;ð display: block;ð margin-bottom: 5px;ð
}ð input {ð width: 100%;ð padding: 12px 15px;ð font-size:
16px;ð border: 2px solid #ccc;ð border-radius: 6px;ð }ð
input:hover {ð border-color: #667eea;ð box-shadow: 0px 0px 10px
rgba(102, 126, 234, 0.5);ð transform: scale(1.02);ð }ð input:focus {ð
outline: none;ð transform: scale(1.05);ð box-shadow: 0px 0px
10px rgba(255, 255, 255, 0.4);ð }ð .error {ð color: red;ð font-
size: 14px;ð margin-top: 5px;ð }ð .remember-me {ð display: flex;ð
align-items: center;ð ð }ð input[type=checkbox]{ð -
webkit-appearance: checkbox;ð -moz-appearance: checkbox;ð
appearance: checkbox;ð display: inline-block;ð width: auto;ð }ð
.remember-me input {ð margin-right: 8px;ð cursor: pointer;ð
left: 0;ð ð }ð .btn {ð width: 100%;ð background:
linear-gradient(to right, #667eea, #764ba2);ð color: white;ð padding:
14px;ð border: none;ð border-radius: 8px;ð font-size: 18px;ð
cursor: pointer;ð font-weight: bold;ð text-transform: uppercase;ð
letter-spacing: 1px;ð transition: all 0.3s ease-in-out;ð margin-top:
10px;ð }ð .btn:hover {ð background: linear-gradient(to right,
#764ba2, #667eea);ð transform: scale(1.05);ð box-shadow: 0px 8px 20px
rgba(102, 126, 234, 0.5);ð }ð .switch {ð text-align: center;ð
margin-top: 15px;ð color: blue;ð cursor: pointer;ð text-
decoration: underline;ð }ð .forgot {ð font-size: 14px;ð text-
decoration: none;ð color: #333;ð }ð .forgot:hover {ð text-
decoration: underline;ð }ð .referral-section {ð text-align: left;ð
margin-bottom: 10px;ð }ð @media (max-width: 768px) {ð
.container {ð flex-direction: column;ð height: auto;ð }ð
.image-section {ð height: 200px;ð width: 100%;ð }ð
.form-section {ð width: 100%;ð padding: 30px;ð }ð
.form-box {ð max-width: 100%;ð }ð ð }ð </style>ð/head>ð
<body>ð <div class="container">ð <div class="image-section"></div>ð <div
class="form-section">ð <div class="form-box active" id="signInBox">ð
<h2>Sign In</h2>ð <form id="signInForm">ð <div
class="input-group">ð <label for="userid">Username</label>ð
<input type="text" id="userid" name="userid" placeholder="Enter
Username, Email, or Phone" required>ð <div class="error"
id="useridError"></div>ð </div>ð <div class="input-group">ð
<label for="signinPassword">Password</label>ð <input
type="password" id="signinPassword" name="password" placeholder="Enter
Password" required>ð <div class="error" id="passwordError"></div>ð
</div>ð <div class="remember-me">ð <div>ð
<input type="checkbox" id="rememberMe">ð
```

```

                                <label
for="rememberMe">Remember Me</label>ð                                </div>ð                                <a href="#"
class="forgot">Forgot Password?</a>ð                                </div>ð                                <button
type="submit" class="btn">Sign In</button>ð                                <div class="switch"
onclick="showForm('signup')">Don't have an account? Sign Up</div>ð                                </
form>ð                                </div>ð                                <div class="form-box" id="signupBox">ð                                <h2>Sign
Up</h2>ð                                <form id="signupForm">ð                                <div class="input-group">ð
                                <label for="userName">Username</label>ð                                <input
type="text" id="userName" name="username" required placeholder="User Name">ð
                                <div class="error" id="userIdError"></div>ð                                </div>ð
                                <div class="input-group">ð                                <label for="email">Email</
label>ð                                <input type="email" id="email" name="email" required
placeholder="Email">ð                                <div class="error" id="emailError"></div>ð
                                </div>ð                                <div class="input-group">ð                                <label
for="phone">Phone Number</label>ð                                <input type="tel" id="phone"
name="phone" required placeholder="Phone Number">ð                                <div
class="error" id="phoneError"></div>ð                                </div>ð                                <div
class="input-group">ð                                <label for="tradingId">Trading View ID</label>ð
                                <input type="text" id="tradingId" name="tradingId" required
placeholder="Trading View ID">ð                                </div>ð                                <div class="input-
group">ð                                <label for="signupPassword">Password</label>ð
                                <input type="password" id="signupPassword" name="password" required
placeholder="Password">ð                                <div class="error" id="passwordError"></
div>ð                                </div>ð                                <div class="input-group">ð                                <label
for="confirmPassword">Repeat Password</label>ð                                <input
type="password" id="confirmPassword" name="confirmPassword" required
placeholder="Confirm Password">ð                                <div class="error"
id="confirmPasswordError"></div>ð                                </div>ð                                <div
class="referral-section">ð                                <input type="checkbox"
id="referralCheckbox" onchange="toggleReferral()">ð                                <label
for="referralCheckbox">Have a Referral ID?</label>ð                                </div>ð
                                <div class="input-group" id="referralId" style="display: none;">ð
                                <label for="referral">Referral ID</label>ð                                <input type="text"
id="referral" name="referral" placeholder="Referral ID">ð                                </div>ð
                                <button type="submit" class="btn">Sign Up</button>ð                                <div
class="switch" onclick="showForm('signin')">Already have an account? Sign In</
div>ð                                </form>ð                                </div>ð                                </div>ð                                </div>ð                                <script>ð                                function
showForm(type) {ð                                const signin = document.getElementById("signinBox");ð
                                const signup = document.getElementById("signupBox");ð                                if (type ===
"signup") {ð                                signin.classList.remove("active");ð
                                signup.classList.add("active");ð                                } else {ð
                                signup.classList.remove("active");ð                                signin.classList.add("active");ð
                                }ð                                }ð                                function toggleReferral() {ð                                const referralInput =
document.getElementById("referralId");ð                                referralInput.style.display =
document.getElementById("referralCheckbox").checked ? "block" : "none";ð                                }ð
                                </script>ð</body>ð</html>ð

```


demo\target\classes\static\signin4.html

```
<!DOCTYPE html>ðhtml lang="en">ðhead>ð <meta charset="UTF-8">ð <meta
name="viewport" content="width=device-width, initial-scale=1.0">ð
<title>Creative Login Page</title>ð <style>ð    * {ð        margin: 0;ð
padding: 0;ð        box-sizing: border-box;ð        font-family: "Segoe UI",
Tahoma, Geneva, Verdana, sans-serif;ð    }ð    ð    body {ð        display: flex;ð
justify-content: center;ð        align-items: center;ð        height: 100vh;ð
background-color: #6c3bdf;ð        margin: 0;ð    }ð    ð    .container {ð
display: flex;ð        width: 800px;ð        height: 500px;ð        background:
white;ð        border-radius: 10px;ð        overflow: hidden;ð        box-shadow: 0
10px 25px rgba(0, 0, 0, 0.1);ð    }ð    ð    .image-section {ð        flex: 1;ð
padding: 20px;ð        display: flex;ð        justify-content: center;ð        align-
items: center;ð        position: relative;ð    }ð    ð    .form-section {ð
flex: 1;ð        padding: 40px;ð        display: flex;ð        flex-direction: column;ð
justify-content: center;ð    }ð    ð    .header {ð        display: flex;ð
justify-content: space-between;ð        align-items: center;ð        margin-bottom:
30px;ð    }ð    ð    .logo {ð        font-weight: bold;ð        font-size: 24px;ð
color: #6c3bdf;ð    }ð    ð    .signup-link {ð        color: #6c3bdf;ð        text-
decoration: none;ð        font-size: 14px;ð    }ð    ð    h1 {ð        font-size: 28px;ð
margin-bottom: 5px;ð        color: #333;ð    }ð    ð    .subtitle {ð
color: #888;ð        margin-bottom: 30px;ð        font-size: 14px;ð    }ð    ð
.input-field {ð        margin-bottom: 20px;ð        width: 100%;ð    }ð    ð
input {ð        width: 100%;ð        padding: 15px;ð        border: 1px solid #ddd;ð
border-radius: 6px;ð        font-size: 14px;ð        outline: none;ð    }ð    ð
.input-field input:focus {ð        border-color: #6c3bdf;ð    }ð    ð    .login-
btn {ð        width: 100%;ð        padding: 15px;ð        background-color: #6c3bdf;ð
color: white;ð        border: none;ð        border-radius: 30px;ð        font-
size: 16px;ð        font-weight: bold;ð        cursor: pointer;ð        margin-bottom:
20px;ð        text-transform: uppercase;ð    }ð    ð    .forgot-password {ð
text-align: right;ð        color: #888;ð        font-size: 14px;ð        text-
decoration: none;ð        margin-bottom: 20px;ð    }ð    ð    .plant {ð
position: absolute;ð        bottom: 10px;ð        left: 10px;ð    }ð</style>ð/head>ð
<body>ð<div class="container">ð    <div class="image-section">ð        ð
<div class="plant">ð        ð    </div>ð    </div>ð    ð    <div class="form-section">ð
<div class="header">ð        <div class="logo">Creative</div>ð        <a
href="#" class="signup-link">New User? Sign Up</a>ð    </div>ð    ð
<h1>Welcome Back!</h1>ð        <div class="subtitle">Login to continue</div>ð
ð        <div class="input-field">ð            <input type="email"
placeholder="username11@gmail.com">ð        </div>ð        ð        <div class="input-
field">ð            <input type="password" placeholder="Enter Password">ð        </
div>ð        ð        <button class="login-btn">LOGIN</button>ð        ð        <div
style="text-align: right;">ð            <a href="#" class="forgot-password">FORGOT
PASSWORD?</a>ð        </div>ð    </div>ð</div>ð</body>ð</html>
```

demo\target\classes\static\signup.html

```
<title>Signup Page</title>ð    <div class="container">ð    <div class="text-
overlay">ð        <h2>Start Your Trading Journey with Confidence</h2>ð
        <p>Trade Smart. Grow Fast.</p>ð    </div>ð    <div class="image-
section">ð        ð    </div>ð
        <div class="form-section">ð            <div class="form-box">ð                <h2>Sign up</
h2>ð                <form id="signupForm">ð                    <div class="input-group">ð
<label for="username">Username</label>ð                        <input type="text"
class="input-style" id="userName" name="username" required placeholder="User
Name">ð                            <div class="error" id="userIdError"></div>ð                            </div>ð
                            <div class="input-group">ð                                <label for="email">Email</label>ð
                                <input type="email" id="email" class="input-style" name="email"
required placeholder="Email">ð                                    <div class="error" id="emailError"></
div>ð                                    </div>ð                                    <div class="input-group">ð                                        <label
for="phone">Phone Number</label>ð                                            <input type="tel" id="phone"
class="input-style" name="phone" required placeholder="Phone Number">ð
                                                <div class="error" id="phoneError"></div>ð                                                </div>ð
<div class="input-group">ð                            <label for="tradingId">Trading View ID</
label>ð                                <input type="text" id="tradingId" class="input-style"
name="tradingId" required placeholder="Trading View ID">ð                                    </div>ð
<div class="input-group">ð                            <label for="password">Password</label>ð
                                <input type="password" id="password" class="input-style"
name="password" required placeholder="Password">ð                                    <div
class="error" id="passwordError"></div>ð                                    </div>ð                                    <div
class="input-group">ð                                        <label for="confirmPassword">Repeat Password</
label>ð                                            <input type="password" class="input-style"
id="confirmPassword" name="confirmPassword" required placeholder="Confirm
Password">ð                                                <div class="error" id="confirmPasswordError"></div>ð
                                                </div>ð                                                <div class="referral-section">ð                                                    <input
type="checkbox" id="referralCheckbox">ð                                                        <label
for="referralCheckbox">Have a Referral ID?</label>ð                                                            </div>ð                                                            <div
class="input-group" id="referralId" style="display: none;">ð                                                                <label
for="referral">Referral ID</label>ð                                                                    <input type="text" class="input-
style" id="referral" name="referral" placeholder="Referral ID">ð                                                                    </div>ð
                                                                <button type="submit" class="btn">Sign Up</button>ð                                                                </form>ð                                                                <div
class="account">ð                                                                    <h4>Already have an account? <a class="signin"
href="signin.html">Sign in</a></h4>ð                                                                    </div>ð                                                                </div>ð                                                                </div>ð                                                                </div>ð
<script>ð        // Show/Hide Referral ID inputð
document.getElementById("referralCheckbox").addEventListener("change",
function () {ð            let referralInput =
document.getElementById("referralId");ð            referralInput.style.display
= this.checked ? "block" : "none";ð            });ð
document.getElementById("signupForm").addEventListener("submit", async
function (event) {ð            event.preventDefault(); // Prevent default form
submissionð            let userName =
document.getElementById("userName").value.trim();ð            let email =
document.getElementById("email").value.trim();ð            let phone =
document.getElementById("phone").value.trim();ð            let tradingId =
document.getElementById("tradingId").value.trim();ð            let password =
document.getElementById("password").value.trim();ð            let
confirmPassword = document.getElementById("confirmPassword").value.trim();ð
            let referral = document.getElementById("referral").value.trim()
|| null;ð            // Validation Patternsð            let usernamePattern = /
^(?!\\s)(?!.*\\s{2,})[a-zA-Z0-9 ]+(?!\\s)$/;ð            let emailPattern = /^[a-
zA-Z0-9._%+-]+@[a-zA-Z0-9.-]+\\.[a-zA-Z]{2,4}$/;ð            let phonePattern
= /^[6-9]\\d{9}$/;ð            let passwordPattern = /^(?=.*[A-Z])(?=.*[a-z])(?
=.*\\d)(?=.*[@$!%*?&])[A-Za-z\\d$!%*?&]{8,}$/;ð            let isValid = true;ð
            // Clear previous errorsð
document.getElementById("userIdError").textContent = "";ð
document.getElementById("emailError").textContent = "";ð
```

```

document.getElementById("phoneError").textContent = "";
document.getElementById("passwordError").textContent = "";
document.getElementById("confirmPasswordError").textContent = "";
// Validate username
if (!usernamePattern.test(userName)) {
document.getElementById("userIdError").textContent = "Invalid username!";
isValid = false;
} // Validate email
if (!emailPattern.test(email)) {
document.getElementById("emailError").textContent = "Invalid Email Format!";
isValid = false;
} // Validate phone
number
if (!phonePattern.test(phone)) {
document.getElementById("phoneError").textContent = "Invalid Phone Number!";
isValid = false;
} // Validate password
if (!passwordPattern.test(password)) {
document.getElementById("passwordError").textContent = "Password must be 8+
characters with 1 uppercase, 1 lowercase, 1 number & 1 special character.";
isValid = false;
} // Confirm password
match
if (password !== confirmPassword) {
document.getElementById("confirmPasswordError").textContent = "Passwords do
not match!";
isValid = false;
} if (!
isValid) return; // Send data to Spring Boot Backend
let requestData = {
  username: userName,
  email: email,
  phone: phone,
  tradingId: tradingId,
  password: password,
  referralId: referralId
};
try {
let response = await
fetch("http://localhost:8080/api/signup", {
  method: "POST",
  headers: { "Content-Type": "application/json" },
  body: JSON.stringify(requestData)
});
let data = await response.json();
if (response.ok) {
// ' Show user details
in alert
alert(`Signup Successful! \n\nUsername: ${data.userName}
\nUser ID: ${data.userId} \nReferral Code: ${data.referralCode}`);
window.location.href = "signin.html";
} else {
document.getElementById("emailError").textContent =
data.message || "Signup failed!";
} catch (error) {
console.error("Error:", error);
alert("An
error occurred while signing up. Please try again.");
}
}
</script>
* {
margin: 0;
padding: 0;
box-sizing: border-box;
font-family: "Poppins", sans-serif;
body {
height: 150vh;
display: flex;
justify-content: center;
align-items: center;
background: white;
.container {
display: flex;
width: 100%;
max-width: 1500px;
height: 950px;
box-shadow: 0px 10px 30px rgba(0, 0, 0, 0.1);
border-radius: 15px;
overflow: hidden;
background-color: #f5f5f5;
background-image: linear-gradient(30deg, #ecec 12%, transparent 12.5%, transparent 87%, #ecec 87.5%, #ecec),
linear-gradient(150deg, #ecec 12%, transparent 12.5%, transparent 87%, #ecec 87.5%, #ecec),
linear-gradient(30deg, #ecec 12%, transparent 12.5%, transparent 87%, #ecec 87.5%, #ecec),
linear-gradient(150deg, #ecec 12%, transparent 12.5%, transparent 87%, #ecec 87.5%, #ecec),
linear-gradient(60deg, #ddd 25%, transparent 25.5%, transparent 75%, #ddd 75%, #ddd),
linear-gradient(60deg, #ddd 25%, transparent 25.5%, transparent 75%, #ddd 75%, #ddd);
background-size: 40px 70px;
background-position: 0 0, 0 0, 20px 35px, 20px 35px, 0 0, 20px 35px;
}
.image-section {
flex: 1;
width: 65%;
background: white;

```

```

margin-top: 60px;@margin-left: 60px;@
margin-bottom: 30px;@border-bottom-left-radius: 5%;@
border-top-left-radius: 5%;@box-shadow: -5px 0 10px -2px rgba(0,
0, 0, 0.4);@.image{@height: 420px;@
width: 48%;@justify-items: center;@position: absolute;@
top: 70%;@left: 28%;@transform:
translate(-50%, -50%);@.form-section{@
width: 45%;@display: flex;@align-items: center;@
justify-content: center;@background: #fff;@
backdrop-filter: blur(10px);@margin-top: 60px;@margin-
right: 60px;@margin-bottom: 30px;@border-top-right-
radius: 5%;@border-bottom-right-radius: 5%;@box-shadow: 5px 0
4px -2px rgba(0, 0, 0, 0.3);@.form-box{@padding: 50px;@
background: rgba(255, 255, 255, 0.2);@padding: 50px;@
width: 100%;@height: 100%;@justify-content: center;@
text-align: center;@color:
white;@border-top-right-radius: 5%;@border-bottom-right-
radius: 5%;@h2{@font-family:
'Times New Roman', Times, serif;@text-align: center;@
font-size: 40px;@font-weight: bold;@margin-bottom: 20px;@
color: black;@text-shadow: 0px 4px 10px rgba(0, 0, 0,
0.1);@h4{@font-family: 'Franklin
Gothic Medium', 'Arial Narrow', Arial, sans-serif;@color: black;@
font-size: 19px;@font-weight: bold;@padding-
bottom: 10px;@.input-group{@margin-
bottom: 15px;@text-align: left;@label{@
font-family: 'Lucida Sans', 'Lucida Sans Regular', 'Lucida Grande', 'Lucida
Sans Unicode', Geneva, Verdana, sans-serif;@font-weight: bold;@
display: block;@margin-bottom: 5px;@font-
size: 14px;@color: black;@letter-spacing: 1px;@}
input{@font-family: 'Gill Sans', 'Gill Sans MT', Calibri,
'Trebuchet MS', sans-serif;@width: 100%;@padding: 12px
15px;@border: 2px solid #ccc;@border-radius: 8px;@
font-size: 16px;@transition: 0.3 ease;@
.input-group input:hover{@border-color:
#667eea;@box-shadow: 0px 0px 10px rgba(102, 126, 234, 0.5);@
transform: scale(1.02);@.error{@color: red;@font-size: 12px;@
margin-top: 5px;@.referral-section{@
display: flex;@align-items: center;@gap: 10px;@
margin-bottom: 10px;@}
input[type=checkbox]{-webkit-appearance: checkbox;@
moz-appearance: checkbox;@appearance: checkbox;@
display: inline-block;@width: auto;@.btn{@
width: 100%;@background: linear-gradient(to right,
#667eea, #764ba2);@color: white;@padding: 14px;@
border: none;@border-radius: 8px;@font-
size: 18px;@cursor: pointer;@transition: all 0.3s ease-
in-out;@font-weight: bold;@letter-spacing: 1px;@
box-shadow: 0px 5px 15px rgba(102, 126, 234, 0.3);@
.btn:hover{@background: linear-gradient(to
right, #764ba2, #667eea);@transform: scale(1.05);@box-
shadow: 0px 8px 20px rgba(102, 126, 234, 0.5);@}
#referralId{@display: none;@.account h4{@
padding-top: 10px;@font-size: 16px;@.signin{@
text-decoration: none;@font-size: 16px;@}
.signin :hover{@text-decoration: underline;@color:
#667eea;@transition: scale(1.02);@}
Responsive */@@media (max-width: 900px) {@.main-container{@
flex-direction: column;@

```

```

width: 90%;
height: auto;
}
.container {
width: 100%;
}
.container {
height: 300px;
}
.overlay {
position: absolute;
top: 23%;
left: 27%;
transform:
translate(-50%, -50%);
text-align: center;
color: white;
padding:
20px;
border-radius: 10px;
}
text-overlay h2 {
color: black;
font-
weight: bold;
font-size: 27px;
text-shadow: 0 0 1px #141616, 0 0 1px
#131314, 0 0 1px #121414;
overflow: hidden;
opacity: 0;
transform:
translateX(30px);
animation: fadeSlideUp 1.2s ease forwards;
}
.text-
overlay p {
color: black;
font-weight: bold;
font-size: 22px;
text-
shadow: 0 0 1px #101111, 0 0 1px #000000, 0 0 1px #16191b;
opacity: 0;
transform: translateX(30px);
animation: fadeSlideUp 1.2s ease forwards;
}
@keyframes fadeSlideUp {
to {
opacity: 1;
transform: translateY(0);
}
}
</style>

```

demo\target\classes\static\user_dashboard.html

```
<!DOCTYPE html>ðhtml lang="en">ðhead>ð  <meta charset="UTF-8">ð  <meta
name="viewport" content="width=device-width, initial-scale=1.0">ð
<title>Trading Homepage</title>ð  <link rel="stylesheet" href="styles.css">ð
  <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>ð  <style>ð
    body {ð              margin: 0;ð              font-family: Arial, sans-
serif;ð              background: #0d0d0d;ð              color: #fff;ð
overflow-x: hidden;ð              scroll-behavior: smooth;ð              }ð
header {ð              display: flex;ð              justify-content: space-between;ð
              align-items: center;ð              padding: 20px;ð
background: rgba(0, 0, 0, 0.8);ð              position: fixed;ð              width:
100%;ð              top: 0;ð              left: 0;ð              animation: fadeIn 1s
ease-in-out;ð              z-index: 1000;ð              }ð              .logo {ð
font-size: 24px;ð              font-weight: bold;ð              color: #ff9800;ð
              }ð              nav ul {ð              list-style: none;ð              display:
flex;ð              gap: 20px;ð              }ð              nav ul li a {ð              text-
decoration: none;ð              color: white;ð              transition: color 0.3s;ð
              }ð              nav ul li a:hover {ð              color: #ff9800;ð              }ð
              .hero {ð              height: 100vh;ð              display: flex;ð
              align-items: center;ð              justify-content: center;ð
              background: url('trading-bg.jpg') no-repeat center center/cover;ð
              text-align: center;ð              animation: zoomIn 1.5s ease-in-out;ð
              }ð              .hero-content {ð              max-width: 600px;ð
animation: slideUp 1.5s ease-in-out;ð              }ð              .hero h1 {ð
font-size: 3rem;ð              margin-bottom: 10px;ð              text-transform:
uppercase;ð              }ð              .hero button {ð              padding: 10px 20px;ð
              font-size: 1.2rem;ð              border: none;ð              cursor:
pointer;ð              background: #ff9800;ð              color: black;ð
transition: background 0.3s;ð              }ð              .hero button:hover {ð
background: #e68900;ð              }ð              .features, .testimonials, .trading-graph
{ð              display: flex;ð              flex-direction: column;ð
align-items: center;ð              text-align: center;ð              padding: 80px
20px;ð              background: linear-gradient(to bottom, #1a1a1a, #0d0d0d);ð
              animation: fadeIn 2s ease-in-out;ð              }ð              .feature {ð
              max-width: 300px;ð              margin: 20px;ð              }ð
              .testimonial {ð              margin: 20px auto;ð              max-width:
400px;ð              font-style: italic;ð              }ð              @keyframes fadeIn {ð
              from { opacity: 0; }ð              to { opacity: 1; }ð              }ð
              @keyframes slideUp {ð              from { transform: translateY(50px);
opacity: 0; }ð              to { transform: translateY(0); opacity: 1; }ð              }ð
              @keyframes zoomIn {ð              from { transform: scale(0.8); opacity:
0; }ð              to { transform: scale(1); opacity: 1; }ð              }ð  </style>ð/
head>ðbody>ð  <header>ð    <div class="logo">TradingX</div>ð    <nav>ð
      <ul>ð        <li><a href="#hero">Home</a></li>ð
        <li><a href="#features">Markets</a></li>ð
        <li><a href="#testimonials">Insights</a></li>ð        <li><a
href="#trading-graph">Live Charts</a></li>ð        <li><a
href="#about">About</a></li>ð        <li><a href="#contact">Contact</
a></li>ð      </ul>ð    </nav>ð  </header>ð  ð  <section id="hero"
class="hero">ð    <div class="hero-content">ð      <h1>Trade Smarter,
Trade Faster</h1>ð      <p>Harness AI-driven trading strategies for
market dominance.</p>ð      <button>Get Started</button>ð    </div>ð
  </section>ð  ð  <section id="features" class="features">ð    <div
class="feature">ð      <h2>Live Market Analysis</h2>ð      <p>Stay
ahead with real-time charts and analytics.</p>ð    </div>ð    <div
class="feature">ð      <h2>AI-Powered Strategies</h2>ð      <p>Leverage machine learning for precise trading decisions.</p>ð    </div>ð
    <div class="feature">ð      <h2>Secure Transactions</h2>ð      <p>Experience top-tier security for all your trades.</p>ð    </div>ð
  </section>ð</body>ð</html>
```

```

        </section>Đ    <section id="testimonials" class="testimonials">Đ
        <h2>What Traders Say</h2>Đ        <div class="testimonial">Đ
        <p>"TradingX revolutionized my trading game!"</p>Đ        <span>- Alex T.</
        span>Đ        </div>Đ        <div class="testimonial">Đ        <p>"Best AI
        tools for market analysis hands down."</p>Đ        <span>- Sarah K.</span>Đ
        </div>Đ    </section>Đ    <section id="trading-graph" class="trading-
        graph">Đ        <h2>Live Trading Graph</h2>Đ        <canvas id="tradingChart"></
        canvas>Đ    </section>Đ    Đ    <script>Đ        document.querySelectorAll('nav
        ul li a').forEach(anchor => {Đ        anchor.addEventListener('click',
        function(event) {Đ        event.preventDefault();Đ        const
        targetId = this.getAttribute('href').substring(1);Đ        const
        targetElement = document.getElementById(targetId);Đ        if
        (targetElement) {Đ        window.scrollTo({Đ
        top: targetElement.offsetTop - 50,Đ        behavior: 'smooth'Đ
        });Đ        });Đ        });Đ        Đ
    </script>Đ</body>Đ</html>

```

demo\target\classes\static\User_dashboard2.html

```
<!DOCTYPE html>ðhtml lang="en">ðhead>ð  <meta charset="UTF-8">ð  <meta
name="viewport" content="width=device-width, initial-scale=1.0">ð  <title>
Trading Platform</title>ð  <style>ð    /* Reset and Base Styles */ð
* {ð      margin: 0;ð      padding: 0;ð      box-sizing:
border-box;ð      font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-
serif;ð    }ð    ð    body {ð      background-color: #0a0e17;ð
      color: #e6e6e6;ð      overflow-x: hidden;ð    }ð    ð
    /* Header Styles */ð    header {ð      background: linear-
gradient(135deg, #131927, #1d2533);ð      padding: 1.5rem 0;ð
position: fixed;ð      width: 100%;ð      top: 0;ð      z-
index: 1000;ð      box-shadow: 0 4px 20px rgba(0, 0, 0, 0.3);ð
animation: headerSlideDown 0.8s ease-out;ð    }ð    ð    .header-
container {ð      max-width: 1200px;ð      margin: 0 auto;ð
      display: flex;ð      justify-content: space-between;ð
      align-items: center;ð      padding: 0 2rem;ð    }ð    ð
    .logo {ð      display: flex;ð      align-items: center;ð
    }ð    ð    .logo h1 {ð      color: #4deeea;ð
font-size: 1.8rem;ð      margin-left: 1rem;ð      font-weight: 700;ð
      animation: logoPulse 2s infinite alternate;ð    }ð    ð
    .logo-icon {ð      width: 40px;ð      height: 40px;ð
      background: linear-gradient(45deg, #4deeea, #4d8eea);ð
border-radius: 8px;ð      display: flex;ð      align-items: center;ð
      justify-content: center;ð      animation: logoRotate 5s
linear infinite;ð    }ð    ð    .logo-icon::before {ð
content: "¿";ð      color: #131927;ð      font-size: 24px;ð
      font-weight: bold;ð    }ð    ð    ð    nav ul {ð
display: flex;ð      list-style: none;ð    }ð    ð    nav ul li
{ð      margin-left: 2rem;ð      position: relative;ð    }ð
    ð    nav ul li a {ð      color: #e6e6e6;ð      text-
decoration: none;ð      font-size: 1rem;ð      font-weight: 500;ð
      transition: all 0.3s ease;ð      padding: 0.5rem 0;ð    }ð
    ð    nav ul li a:hover {ð      color: #4deeea;ð    }ð
    ð    nav ul li a::after {ð      content: '';ð
position: absolute;ð      bottom: -5px;ð      left: 0;ð
width: 0%;ð      height: 2px;ð      background-color: #4deeea;ð
      transition: width 0.3s ease;ð    }ð    ð    nav ul li
a:hover::after {ð      width: 100%;ð    }ð    ð    .cta-btn {ð
      background: linear-gradient(45deg, #4deeea, #4d8eea);ð
color: #131927;ð      border: none;ð      padding: 0.8rem 1.5rem;ð
      border-radius: 30px;ð      font-weight: 600;ð
cursor: pointer;ð      transition: all 0.3s ease;ð      animation:
ctaPulse 2s infinite alternate;ð    }ð    ð    .cta-btn:hover {ð
      transform: translateY(-3px);ð      box-shadow: 0 10px 20px
rgba(77, 238, 234, 0.3);ð    }ð    ð    /* Hero Section */ð
    .hero {ð      height: 100vh;ð      display: flex;ð
      align-items: center;ð      background: url("/api/
placeholder/1200/800") center/cover;ð      background-blend-mode:
overlay;ð      background-color: rgba(13, 17, 28, 0.85);ð
padding-top: 80px;ð      position: relative;ð      overflow: hidden;ð
    }ð    ð    .hero::before {ð      content: '';ð
position: absolute;ð      top: 0;ð      left: 0;ð      width:
100%;ð      height: 100%;ð      background: linear-gradient(135deg,
rgba(13, 17, 28, 0.9), rgba(13, 17, 28, 0.6));ð      z-index: 1;ð
    }ð    ð    .hero-content {ð      max-width: 1200px;ð
margin: 0 auto;ð      padding: 0 2rem;ð      display: flex;ð
      justify-content: space-between;ð      align-items: center;ð
      width: 100%;ð      position: relative;ð      z-index: 2;ð
    }ð    ð    .hero-text {ð      flex: 0 0 50%;ð
animation: heroTextSlide 1s ease-out;ð
```



```

        }@ @ .hero-text h2 {@
font-size: 3.5rem;@ font-weight: 700;@ margin-
bottom: 1.5rem;@ line-height: 1.2;@ color: #ffffff;@
}@ @ .hero-text h2 span {@ background: linear-
gradient(45deg, #4deeea, #4d8eea);@ -webkit-background-clip: text;@
background-clip: text;@ color: transparent;@
position: relative;@ display: inline-block;@ }@ @
.hero-text p {@ font-size: 1.1rem;@ line-height:
1.6;@ margin-bottom: 2rem;@ color: #bbbbbb;@ }@
@ @ .hero-btns {@ display: flex;@ gap: 1rem;@
}@ @ .hero-btns .cta-btn.secondary {@
background: transparent;@ border: 2px solid #4deeea;@
color: #4deeea;@ }@ @ .hero-btns .cta-btn.secondary:hover {@
background: rgba(77, 238, 234, 0.1);@ }@ @
.hero-image {@ flex: 0 0 45%;@ animation:
floatUpDown 4s ease-in-out infinite, heroImageFadeIn 1.5s ease-out;@ }@
@ @ .hero-image img {@ width: 100%;@ border-
radius: 10px;@ box-shadow: 0 20px 40px rgba(0, 0, 0, 0.4);@ }@
@ /* Features Section */@ .features {@ padding:
6rem 2rem;@ background: linear-gradient(0deg, #0d111c, #131927);@
}@ @ .section-header {@ text-align: center;@
margin-bottom: 4rem;@ }@ @ .section-header h3 {@
font-size: 2.5rem;@ margin-bottom: 1rem;@
position: relative;@ display: inline-block;@ color:
#ffffff;@ }@ @ .section-header h3::after {@
content: '';@ position: absolute;@ bottom: -10px;@
left: 50%;@ transform: translateX(-50%);@
width: 80px;@ height: 3px;@ background: linear-
gradient(45deg, #4deeea, #4d8eea);@ }@ @ .section-header p {@
max-width: 700px;@ margin: 0 auto;@ color:
#bbbbbb;@ font-size: 1.1rem;@ line-height: 1.6;@ }@
@ @ .features-container {@ max-width: 1200px;@
margin: 0 auto;@ display: grid;@ grid-
template-columns: repeat(auto-fit, minmax(300px, 1fr));@ gap: 2rem;@
}@ @ .feature-card {@ background: linear-
gradient(145deg, #131927, #1d2533);@ border-radius: 15px;@
padding: 2rem;@ box-shadow: 0 10px 30px rgba(0, 0, 0,
0.2);@ transition: all 0.4s ease;@ position: relative;@
overflow: hidden;@ z-index: 1;@ }@ @
.feature-card:hover {@ transform: translateY(-10px);@
box-shadow: 0 15px 40px rgba(77, 238, 234, 0.15);@ }@ @
.feature-card::before {@ content: '';@ position:
absolute;@ top: 0;@ left: 0;@ width: 4px;@
height: 0;@ background: linear-gradient(45deg, #4deeea,
#4d8eea);@ transition: height 0.4s ease;@ z-index: -1;@
}@ @ .feature-card:hover::before {@ height:
100%;@ }@ @ .feature-icon {@ width: 60px;@
height: 60px;@ border-radius: 50%;@
background: rgba(77, 238, 234, 0.1);@ display: flex;@
align-items: center;@ justify-content: center;@ margin-
bottom: 1.5rem;@ font-size: 1.8rem;@ color: #4deeea;@
}@ @ .feature-card h4 {@ font-size: 1.5rem;@
margin-bottom: 1rem;@ color: #ffffff;@ }@ @
.feature-card p {@ color: #bbbbbb;@ line-height:
1.6;@ }@ @ /* Trading Charts Section */@ .trading-
charts {@ padding: 6rem 2rem;@ background: #0a0e17;@
position: relative;@ overflow: hidden;@ }@ @
.trading-charts::before {@ content: '';@
position: absolute;@ top: 0;@ left: 0;@ width:
100%;@

```

```

        height: 100%;
        background: url("/api/placeholder/1200/800") center/cover;
        opacity: 0.05;
        z-index: 0;
    }
    .charts-container {
        max-width: 1200px;
        margin: 0 auto;
        position: relative;
        z-index: 1;
    }
    .chart-tabs {
        display: flex;
        justify-content: center;
        gap: 1rem;
        margin-bottom: 2rem;
    }
    .chart-tab {
        padding: 0.8rem 1.5rem;
        background: linear-gradient(145deg, #131927, #1d2533);
        border: none;
        border-radius: 30px;
        color: #bbbbbb;
        cursor: pointer;
        transition: all 0.3s ease;
    }
    .chart-tab.active, .chart-tab:hover {
        background: linear-gradient(45deg, #4deeea, #4d8eea);
        color: #131927;
        font-weight: 600;
    }
    .chart-display {
        background: linear-gradient(145deg, #131927, #1d2533);
        border-radius: 15px;
        padding: 2rem;
        box-shadow: 0 10px 30px rgba(0, 0, 0, 0.2);
        height: 500px;
        position: relative;
        overflow: hidden;
        animation: chartAnimate 1s ease-out;
    }
    .chart-display img {
        width: 100%;
        height: 100%;
        object-fit: cover;
        border-radius: 10px;
    }
    .chart-stats {
        display: grid;
        grid-template-columns: repeat(4, 1fr);
        gap: 1.5rem;
        margin-top: 2rem;
    }
    .stat-card {
        background: linear-gradient(145deg, #131927, #1d2533);
        border-radius: 10px;
        padding: 1.5rem;
        text-align: center;
        box-shadow: 0 5px 15px rgba(0, 0, 0, 0.2);
        animation: fadeInUp 0.8s ease-out;
    }
    .stat-card h5 {
        font-size: 1.1rem;
        color: #bbbbbb;
        margin-bottom: 0.8rem;
    }
    .stat-card .value {
        font-size: 1.8rem;
        font-weight: 700;
        color: #ffffff;
    }
    .stat-card .trend {
        display: flex;
        align-items: center;
        justify-content: center;
        margin-top: 0.5rem;
        font-size: 0.9rem;
    }
    .stat-card .trend.up {
        color: #4dfa7b;
    }
    .stat-card .trend.down {
        color: #fa4d4d;
    }
    /* Testimonials Section */
    .testimonials {
        padding: 6rem 2rem;
        background: linear-gradient(0deg, #131927, #0d111c);
    }
    .testimonials-container {
        max-width: 1200px;
        margin: 0 auto;
    }
    .testimonial-slider {
        display: flex;
        overflow-x: auto;
        scroll-snap-type: x mandatory;
        -ms-overflow-style: none;
        gap: 2rem;
        padding: 1rem 0;
    }
    .testimonial-slider::-webkit-scrollbar {
        display: none;
    }
    .testimonial-card {
        flex: 0 0 350px;
        scroll-snap-align: start;
        background: linear-gradient(145deg, #131927, #1d2533);
        border-radius: 15px;
        padding: 2rem;
        box-shadow: 0 10px 30px rgba(0, 0, 0, 0.2);
        position: relative;
        animation: fadeInRight 1s ease-out;
    }
    .testimonial-card::before {
        position: absolute;
        top: 10px;
        left: 20px;
        font-size: 6rem;
        color: rgba(77, 238, 234, 0.1);
        line-height: 1;
    }
    .testimonial-content {
        position: relative;
        z-index: 1;
        font-style: italic;
        color: #bbbbbb;
        line-height: 1.6;
        margin-bottom: 1.5rem;
    }
    .testimonial-author {
        display: flex;
        align-items: center;
        width: 50px;
        height: 50px;
        border-radius: 50%;
        overflow: hidden;
        margin-right: 1rem;
    }

```

```

        .author-avatar img {
            width: 100%;
            height: 100%;
            object-fit: cover;
        }
        .author-info h5 {
            font-size: 1.1rem;
            color: #ffffff;
            margin-bottom: 0.2rem;
        }
        .author-info p {
            color: #bbbbbb;
            font-size: 0.9rem;
        }
        /* CTA Section */
        .cta-section {
            padding: 5rem 2rem;
            background: linear-gradient(135deg, #0a0e17, #131927);
            text-align: center;
            position: relative;
            overflow: hidden;
        }
        .cta-section::before {
            content: '';
            position: absolute;
            top: -100px;
            right: -100px;
            width: 300px;
            height: 300px;
            border-radius: 50%;
            background: rgba(77, 238, 234, 0.05);
            z-index: 0;
            animation: floatAroundLarge 15s linear infinite;
        }
        .cta-section::after {
            content: '';
            position: absolute;
            bottom: -80px;
            left: -80px;
            width: 200px;
            height: 200px;
            border-radius: 50%;
            background: rgba(77, 142, 234, 0.05);
            z-index: 0;
            animation: floatAroundSmall 10s linear infinite reverse;
        }
        .cta-container {
            max-width: 800px;
            margin: 0 auto;
            position: relative;
            z-index: 1;
        }
        .cta-container h3 {
            font-size: 2.5rem;
            margin-bottom: 1.5rem;
            color: #ffffff;
        }
        .cta-container p {
            font-size: 1.1rem;
            line-height: 1.6;
            color: #bbbbbb;
            margin-bottom: 2rem;
        }
        .cta-form {
            display: flex;
            max-width: 500px;
            margin: 0 auto;
            position: relative;
        }
        .cta-input {
            flex: 1;
            height: 60px;
            border-radius: 30px;
            border: none;
            padding: 0 2rem;
            background: linear-gradient(145deg, #131927, #1d2533);
            color: #ffffff;
            font-size: 1rem;
            box-shadow: 0 5px 15px rgba(0, 0, 0, 0.2);
        }
        .cta-input:focus {
            box-shadow: 0 5px 20px rgba(77, 238, 234, 0.2);
            outline: none;
        }
        .cta-btn-submit {
            position: absolute;
            right: 5px;
            top: 5px;
            height: 50px;
            border-radius: 25px;
            background: linear-gradient(45deg, #4deeea, #4d8eea);
            color: #131927;
            border: none;
            padding: 0 2rem;
            font-weight: 600;
            cursor: pointer;
            transition: all 0.3s ease;
        }
        .cta-btn-submit:hover {
            transform: translateX(3px);
            box-shadow: 0 5px 15px rgba(77, 238, 234, 0.3);
        }
        /* Footer */
        footer {
            background: #0a0e17;
            padding: 4rem 2rem 2rem;
            color: #bbbbbb;
        }
        .footer-container {
            max-width: 1200px;
            margin: 0 auto;
        }
        .footer-top {
            display: grid;
            grid-template-columns: repeat(auto-fit, minmax(200px, 1fr));
            gap: 2rem;
            margin-bottom: 3rem;
        }
        .footer-logo {
            grid-column: 1 / span 2;
        }
        .footer-logo .logo {
            margin-bottom: 1.5rem;
        }
        .footer-about {
            max-width: 300px;
            line-height: 1.6;
            font-size: 0.95rem;
        }
        .footer-links h4 {
            font-size: 1.2rem;
            color: #ffffff;
            margin-bottom: 1.5rem;
            position: relative;
            display: inline-block;
        }
        .footer-links h4::after {
            content: '';
            position: absolute;
            bottom: -8px;
            left: 0;
            width: 30px;
            height: 2px;
            background: linear-gradient(45deg, #4deeea, #4d8eea);
        }
        .footer-links ul {
            list-style: none;
            margin-bottom: 0.8rem;
        }
        .footer-links ul li {
            margin-bottom: 0.8rem;
        }
        .footer-links ul li a {
            color: #bbbbbb;
            text-decoration: none;
        }

```

```

        font-size: 0.95rem;D        }D        D        .footer-links ul li
a:hover {D        color: #4deeea;D        padding-left: 5px;D        }D
        D        .footer-bottom {D        border-top: 1px solid rgba(255,
255, 255, 0.1);D        padding-top: 2rem;D        display: flex;D
        justify-content: space-between;D        align-items: center;D
        flex-wrap: wrap;D        gap: 1rem;D        }D        D
        .footer-bottom p {D        font-size: 0.9rem;D        }D        D
        .social-links {D        display: flex;D        gap: 1rem;D
        }D        D        .social-link {D        width: 40px;D
height: 40px;D        border-radius: 50%;D        background: linear-
gradient(145deg, #131927, #1d2533);D        display: flex;D
align-items: center;D        justify-content: center;D        color:
#bbbbbb;D        text-decoration: none;D        transition: all 0.3s
ease;D        font-size: 1.2rem;D        }D        D        .social-
link:hover {D        background: linear-gradient(45deg, #4deeea, #4d8eea);D
        color: #131927;D        transform: translateY(-3px);D        }D
        D        /* Animations */D        @keyframes headerSlideDown {D
        from {D        transform: translateY(-100%);D
        opacity: 0;D        }D        to {D
transform: translateY(0);D        opacity: 1;D        }D        }D
        D        @keyframes logoPulse {D        from {D
transform: scale(1);D        }D        to {D        transform:
scale(1.05);D        }D        }D        D        @keyframes logoRotate {D
        from {D        transform: rotate(0deg);D        }D
        to {D        transform: rotate(360deg);D        }D
        }D        D        @keyframes ctaPulse {D        from {D
        box-shadow: 0 0 0 rgba(77, 238, 234, 0);D        }D
        to {D        box-shadow: 0 0 15px rgba(77, 238, 234, 0.5);D
        }D        }D        D        @keyframes heroTextSlide {D
from {D        transform: translateX(-50px);D        opacity: 0;D
        }D        to {D        transform: translateX(0);D
        opacity: 1;D        }D        }D        D        @keyframes
heroImageFadeIn {D        from {D        transform:
translateX(50px);D        opacity: 0;D        }D        to {D
        transform: translateX(0);D        opacity: 1;D
        }D        }D        D        @keyframes floatUpDown {D        0%,
100% {D        transform: translateY(0);D        }D        50% {D
        transform: translateY(-15px);D        }D        }D        D
        @keyframes chartAnimate {D        from {D        opacity: 0;D
        transform: scale(0.98);D        }D        to {D
        opacity: 1;D        transform: scale(1);D        }D
        }D        D        @keyframes fadeInUp {D        from {D
        opacity: 0;D        transform: translateY(20px);D
        }D        to {D        opacity: 1;D
        }D        }D        D        @keyframes
fadeInRight {D        from {D        opacity: 0;D
transform: translateX(-20px);D        }D        to {D
        opacity: 1;D        transform: translateX(0);D        }D        }D
        D        @keyframes floatAroundLarge {D        0% {D
transform: translate(0, 0);D        }D        25% {D
transform: translate(-50px, 50px);D        }D        50% {D
transform: translate(-100px, 0);D        }D        75% {D
transform: translate(-50px, -50px);D        }D        100% {D
transform: translate(0, 0);D        }D        }D        D
        @keyframes floatAroundSmall {D        0% {D
transform: translate(0, 0);D        }D        25% {D
transform: translate(30px, -30px);D        }D        50% {D
transform: translate(60px, 0);D        }D        75% {D
transform: translate(30px, 30px);D        }D

```

```

100% {
  transform: translate(0, 0);
  /* Responsive Styles */
  @media (max-width: 992px) {
    .hero-content {
      flex-direction: column;
      text-align: center;
    }
    .hero-btns {
      justify-content: center;
    }
    .chart-stats {
      grid-template-columns: repeat(2, 1fr);
    }
    @media (max-width: 768px) {
      .hero-text h2 {
        font-size: 2.5rem;
      }
      .header-container {
        flex-direction: column;
        gap: 1rem;
        flex-wrap: wrap;
      }
      nav ul {
        justify-content: center;
      }
      nav ul li {
        margin: 0 1rem;
      }
      .footer-logo {
        grid-column: 1 / -1;
      }
      @media (max-width: 576px) {
        .hero-text h2 {
          font-size: 2rem;
        }
        .cta-form {
          flex-direction: column;
        }
        .cta-btn-submit {
          position: static;
          width: 100%;
        }
        .chart-stats {
          height: 50px;
        }
        .chart-stats {
          grid-template-columns: 1fr;
        }
        .footer-bottom {
          flex-direction: column;
          text-align: center;
        }
        .social-links {
          justify-content: center;
        }
      }
    }
  }
}
</style></head><body>
<!-- Header Section -->
<header>
  <div class="header-container">
    <div class="logo">
      <h1>TradePro</h1>
    </div>
    <nav>
      <ul>
        <li><a href="#home">Home</a></li>
        <li><a href="#features">Features</a></li>
        <li><a href="#markets">Markets</a></li>
        <li><a href="#pricing">Pricing</a></li>
        <li><a href="#about">About</a></li>
      </ul>
    </nav>
    <button class="cta-btn">Start Trading</button>
  </div>
</header>
<!-- Hero Section -->
<section class="hero" id="home">
  <div class="hero-content">
    <div class="hero-text">
      <h2>Trade <span>Smarter</span>, Not Harder</h2>
      <p>Experience a new generation of trading with TradePro's advanced platform. Real-time analysis, AI-powered insights, and commission-free trades to maximize your profits.</p>
    </div>
    <div class="hero-btns">
      <button class="cta-btn">Get Started</button>
      <button class="cta-btn secondary">Learn More</button>
    </div>
  </div>
  
</div>
</section>
<!-- Features Section -->
<section class="features" id="features">
  <div class="section-header">
    <h3>Powerful Features</h3>
  </div>
  <p>Discover the tools that make TradePro the preferred choice of professional traders around the world.</p>
  <div class="features-container">
    <div class="feature-icon"><img alt="Real-Time Analytics icon" /></div>
    <div class="feature-card">
      <h4>Real-Time Analytics</h4>
      <p>Access comprehensive market analysis and real-time data from global markets to make informed trading decisions.</p>
    </div>
    <div class="feature-card">
      <div class="feature-icon"><img alt="AI Trading Assistant icon" /></div>
      <h4>AI Trading Assistant</h4>
      <p>Our advanced AI analyzes patterns and trends to suggest optimal trading strategies tailored to your risk profile.</p>
    </div>
  </div>
</section>

```

```


#### Multi-Platform Access



Trade on the go with our seamless experience across desktop, web, and
mobile applications with real-time synchronization.



#### Bank-Level Security



Your assets are protected with industry-leading encryption, two-factor
authentication, and regular security audits.



#### Advanced Charting



Over 100+
technical indicators and drawing tools to analyze market trends and make
precise predictions.



#### Commission-Free



Trade stocks, ETFs, and
cryptocurrencies without any commission fees. Only pay for premium features
you choose.


```

demo\target\classes\static\user_dashboard3.html

```
<!DOCTYPE html>ðhtml lang="en">ðhead>ð    <meta charset="UTF-8">ð
<title>User Dashboard</title>ð    ð    <link rel="stylesheet" href="https://
cdnjs.cloudflare.com/ajax/libs/font-awesome/6.5.0/css/all.min.css">ð
<script src="https://cdn.jsdelivr.net/npm/sockjs-client@1/dist/
sockjs.min.js"></script>ð <script src="https://cdn.jsdelivr.net/npm/
stompjs@2.3.3/lib/stomp.min.js"></script>ð <link rel="stylesheet"
href="user_dashboard3.css">ð/head>ðbody>ð    <header class="header">ð
<button class="hamburger" id="menuToggle">ð                <i class="fas fa-
bars"></i>ð                </button>ð                <div class="logo"></div>ð                <div class="header-
title">User Management System</div>ð                <div class="user-info-mini">ð
ð                <div class="user-data">ð                <span
id="headerUserName" class="username">Loading...</span>ð                <span
id="headerUserRole" class="name" style="font-size: 12px; color: black;
margin-left: 8px;">(Role)</span>ð                </div>ð                </div>ð    </header>ð
ð    <aside class="sidebar" id="sidebar">ð                <div class="sidebar-
header">ð                <div class="user-info">ð                <div class="user-
avatar" id="userAvatar">U</div>ð                <div>ð                <div
class="user-name" id="sidebarUserName">Loading...</div>ð
<div class="user-role" id="sidebarUserRole">Role</div>ð
<div class="user-status" id="userStatus">Online</div>ð                </div>ð
</div>ð                <ul class="nav-menu">ð
<li><a href="#" onclick="showModule('dashboardModule')"><i class="fas fa-
tachometer-alt"></i> Dashboard</a></li>ð                <li><a href="#"
onclick="showModule('userProfileModule')"><i class="fas fa-user"></i>
Profile</a></li>ð                <li><a href="#"
onclick="showModule('settingsModule')"><i class="fas fa-cog"></i> Settings</
a></li>ð                <li><a href="#"><i class="fas fa-chart-bar"></i>
Analytics</a></li>ð                <li><a href="#"
onclick="showModule('chatModule')"><i class="fas fa-comment-sms"></i> Chat</
a></li>ð                </ul>ð                <button id="logout" class="logout-btn">ð
<i class="fas fa-sign-out-alt"></i> Logoutð                </button>ð    </
aside>ð    ð    <main class="main-content" id="mainContent">ð                <div
id="dashboardModule" class="module" style="display: block;">ð                <div
class="dashboard-header">ð                <h1 class="dashboard-title">Dashboard</
h1>ð                <div class="breadcrumb">Home &gt; Dashboard</div>ð                </div>ð
ð                <div class="data-card fade-in">ð                <h2>Session
Information</h2>ð                <div id="sessionDataContainer">ð
<div class="loading" id="sessionLoading">Loading...</div>ð
<table class="session-table" id="sessionTable" style="display: none;">ð
    <thead>ð                <tr>ð
ð                <th>User ID</th>ð
<th>Username</th>ð
<th>Sign-in Time</th>ð                <th>Logout Time</th>ð
<th>Role</th>ð
<th>Status</th>ð                </tr>ð
    </thead>ð                <tbody>ð
ð                <tr>ð
ð                <td
id="userIdDisplay"></td>ð                <td
id="userNameDisplay"></td>ð
id="signinTimeDisplay"></td>ð
id="logoutTimeDisplay"></td>ð
id="roleDisplay"></td>ð                <td id="statusDisplay"></td>ð
    </tr>ð                </tbody>ð                </div>ð
ð                </div>ð                </div>ð                <!-- USER PROFILE
MODULE -->ð                <div id="userProfileModule" class="module" style="display: none;">ð
    <div class="dashboard-header">ð                <h1 class="dashboard-title">User
Profile</h1>ð                <div class="breadcrumb">Home &gt; Profile</div>ð                </div>ð
ð
```

```

        <div class="profile-card">ð        <div class="profile-photo">ð        ð        </div>ð
<div class="profile-details">ð        <div class="profile-row">ð        <span
class="profile-label">User ID:</span>ð        <span
id="profileUserId">Loading...</span>ð        </div>ð        <div class="profile-
row">ð        <span class="profile-label">Name:</span>ð        <span
id="profileUserName">Loading...</span>ð        </div>ð        <div
class="profile-row">ð        <span class="profile-label">Email:</span>ð
        <span id="profileEmail">Loading...</span>ð        </div>ð        <div
class="profile-row">ð        <span class="profile-label">Phone:</span>ð
        <span id="profilePhone">Loading...</span>ð        </div>ð        <div
class="profile-row">ð        <span class="profile-label">Role:</span>ð
        <span id="profileRole">Loading...</span>ð        </div>ð        <div
class="profile-row">ð        <span class="profile-label">Referral Code:</
span>ð        <span id="profileReferralCode">Loading...</span>ð        </div>ð
        <div class="profile-row">ð        <span class="profile-
label">Created At:</span>ð        <span id="profileCreatedAt">Loading...</
span>ð        </div>ð        </div>ð        </div>ð        </div>ð        ð        <div id="settingsModule"
class="module" style="display: none;">ð        <div class="dashboard-header">ð
        <h1 class="dashboard-title">User Settings</h1>ð        <div
class="breadcrumb">Home > Settings</div>ð        </div>ð        ð        <!-- Username
Section -->ð        <div class="settings-section">ð        <label>Username:</label>ð
        <span id="currentUsername" class="current-value">Loading...</span>ð
        <div id="usernameEdit" class="edit-section">ð        <input
type="text" id="usernameInput" placeholder="Enter new username" />ð
        <button class="save-button" onclick="saveUsername()">Save</button>ð
        <button class="cancel-button" onclick="cancelEdit('username')">Cancel</button>ð
        </div>ð        ð        </div>ð        <!-- Phone Section -->ð        <div
class="settings-section">ð        <label>Phone:</label>ð        <span
id="currentPhone" class="current-value">Loading...</span>ð        <div
id="phoneEdit" class="edit-section">ð        <input type="tel"
id="phoneInput" placeholder="Enter new phone number" />ð        <button
class="save-button" onclick="savePhone()">Save</button>ð        <button
class="cancel-button" onclick="cancelEdit('phone')">Cancel</button>ð        </
div>ð        ð        </div>ð        <!-- Single Edit Button -->ð        <div class="settings-
section">ð        <button class="edit-button" id="editProfileBtn"
onclick="editProfile()">Edit Profile</button>ð        </div>ð        <!-- Profile
Picture -->ð        <div class="settings-section">ð        <label>Change Profile
Picture:</label>ð        ð
        <input type="file" id="uploadInput" accept="image/*" />ð        </div>ð        <div
id="settingsStatus" class="status-msg"></div>ð        <div id="successPopup">ð
        <p id="successMessage"></p>ð        <button onclick="closePopup()">OK</button>ð
        </div>ð        </div>ð        ð        ð        ð        ð        ð        ð        <div id="chatModule" class="module"
style="display: none;">ð        <h2>Chat</h2>ð        ð        <!-- Chat Box to display
messages -->ð        <div class="chat-box" id="chatBox">ð        <!-- Messages will
be displayed here -->ð        </div>ð        <!-- Notification Area (Hidden initially)
-->ð        <div id="notificationArea" style="display: none; position: fixed; top:
20px; left: 50%; transform: translateX(-50%); background-color: #4caf50;
color: white; padding: 10px 20px; border-radius: 5px; cursor: pointer;">ð
        New message received!ð        </div>ð        ð        <!-- Loading indicator -->ð        <div
id="loadingIndicator" style="display: none;">Loading...</div>ð        ð        <!--
Message Input Section -->ð        <div class="chat-input-area">ð        <input
type="text" id="chatInput" placeholder="Type your message..." />ð        <button
onclick="sendChatMessage()">Send</button>ð        </div>ð        </div>ð        ð        </main>ð        ð
        ð        ð        ð        <script src="user_dashboard3.js"></script>ð        ð        </body>ð        </html>ð

```


demo\target\classes\static\user_dashboard4.html

```
<!DOCTYPE html>ðhtml lang="en">ðhead>ð  <meta charset="UTF-8">ð  <meta
name="viewport" content="width=device-width, initial-scale=1.0">ð
<title>User Dashboard</title>ð  <style>ð      :root {ð          --primary-
color: #4361ee;ð          --secondary-color: #3f37c9;ð          --success-
color: #4caf50;ð          --danger-color: #f44336;ð          --light-color:
#f5f5f5;ð          --dark-color: #333;ð          --sidebar-width: 250px;ð
          --header-height: 60px;ð          --box-shadow: 0 2px 10px
          rgba(0, 0, 0, 0.1);ð          --transition: all 0.3s ease;ð          }ð      *
{ð          margin: 0;ð          padding: 0;ð          box-sizing: border-
box;ð          font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;ð
      }ð      body {ð          background-color: #f8f9fa;ð
color: var(--dark-color);ð          position: relative;ð          min-
height: 100vh;ð          }ð      /* Header styles */ð      .header {ð
          background-color: white;ð          height: var(--header-height);ð
          padding: 0 20px;ð          display: flex;ð          align-
items: center;ð          justify-content: space-between;ð
position: fixed;ð          top: 0;ð          left: 0;ð          right: 0;ð
          z-index: 100;ð          box-shadow: var(--box-shadow);ð          }ð
      .header-title {ð          font-size: 1.2rem;ð          font-weight:
600;ð          color: var(--primary-color);ð          }ð      .hamburger {ð
          display: none;ð          font-size: 24px;ð          cursor:
pointer;ð          color: var(--dark-color);ð          background: none;ð
          border: none;ð          }ð      /* Sidebar styles */ð
      .sidebar {ð          background-color: white;ð          width:
var(--sidebar-width);ð          height: calc(100vh - var(--header-height));ð
          position: fixed;ð          top: var(--header-height);ð
          left: 0;ð          box-shadow: var(--box-shadow);ð          padding-top:
20px;ð          transition: var(--transition);ð          z-index: 99;ð
          }ð      .sidebar.collapsed {ð          left: -250px;ð          }ð
      .sidebar-header {ð          padding: 0 20px 20px;ð          border-
bottom: 1px solid #eee;ð          margin-bottom: 20px;ð          }ð      .user-
info {ð          display: flex;ð          align-items: center;ð          }ð
      .user-avatar {ð          width: 45px;ð          height: 45px;ð
          border-radius: 50%;ð          background-color: var(--primary-
color);ð          color: white;ð          display: flex;ð          align-
items: center;ð          justify-content: center;ð          font-weight:
bold;ð          margin-right: 12px;ð          }ð      .user-name {ð
font-weight: 600;ð          font-size: 1.1rem;ð          margin-bottom: 3px;ð
          }ð      .user-status {ð          font-size: 0.85rem;ð
color: var(--success-color);ð          }ð      .nav-menu {ð          list-
style: none;ð          }ð      .nav-menu li {ð          margin-bottom: 5px;ð
          }ð      .nav-menu a {ð          display: flex;ð          align-
items: center;ð          padding: 12px 20px;ð          color: var(--dark-
color);ð          text-decoration: none;ð          transition: var(--
transition);ð          }ð      .nav-menu a:hover, .nav-menu a.active {ð
          background-color: rgba(67, 97, 238, 0.1);ð          color: var(--
primary-color);ð          border-left: 3px solid var(--primary-color);ð
          }ð      .nav-menu a i {ð          margin-right: 10px;ð
font-size: 1.2rem;ð          }ð      .logout-btn {ð          margin: 20px;ð
padding: 10px;ð          background-color: #f8f9fa;ð
color: var(--danger-color);ð          border: 1px solid #ddd;ð
border-radius: 5px;ð          cursor: pointer;ð          width: calc(100% -
40px);ð          font-weight: 600;ð          display: flex;ð
align-items: center;ð          justify-content: center;ð
transition: var(--transition);ð          }ð      .logout-btn:hover {ð
background-color: var(--danger-color);ð          color: white;ð          }ð
      /* Main content styles */ð      .main-content {ð          margin-
left: var(--sidebar-width);ð          margin-top: var(--header-height);ð
          padding: 20px;ð
```

```

        transition: var(--transition);@
        .main-content.expanded {@
            margin-left: 0;@
        }@
        .dashboard-header {@
            margin-bottom: 20px;@
        }@
        .dashboard-title {@
            font-size: 1.8rem;@
            font-weight: 600;@
            margin-bottom: 5px;@
            color: var(--dark-color);@
        }@
        .breadcrumb {@
            font-size: 0.9rem;@
            color: #666;@
            margin-bottom: 20px;@
        }@
        .data-card {@
            background-color: white;@
            border-radius: 8px;@
            box-shadow: var(--box-shadow);@
            padding: 20px;@
            margin-bottom: 20px;@
        }@
        .data-card h2 {@
            font-size: 1.2rem;@
            margin-bottom: 15px;@
            color: var(--dark-color);@
        }@
        .session-table {@
            border-bottom: 1px solid #eee;@
            padding-bottom: 10px;@
            width: 100%;@
            border-collapse: collapse;@
        }@
        .session-table th, .session-table td {@
            padding: 12px 15px;@
            text-align: left;@
            border-bottom: 1px solid #eee;@
        }@
        .session-table th {@
            font-weight: 600;@
            color: #666;@
            background-color: #f8f9fa;@
        }@
        .session-table tr:last-child td {@
            border-bottom: none;@
        }@
        .session-table .status {@
            padding: 5px 10px;@
            border-radius: 20px;@
            font-size: 0.85rem;@
            font-weight: 500;@
        }@
        .session-table .status.active {@
            background-color: rgba(76, 175, 80, 0.1);@
            color: var(--success-color);@
        }@
        .session-table .status.inactive {@
            background-color: rgba(244, 67, 54, 0.1);@
            color: var(--danger-color);@
        }@
        .loading {@
            display: flex;@
            justify-content: center;@
            align-items: center;@
            height: 100px;@
        }@
        .loading:after {@
            content: " ";@
            display: block;@
            width: 40px;@
            height: 40px;@
            border-radius: 50%;@
            border: 6px solid var(--primary-color);@
            border-color: var(--primary-color) transparent var(--primary-color) transparent;@
            animation: loader 1.2s linear infinite;@
        }@
        @keyframes loader {@
            0% {@
                transform: rotate(0deg);@
            }@
            100% {@
                transform: rotate(360deg);@
            }@
        }@
        /* Responsive styles */@
        @media screen and (max-width: 768px) {@
            .hamburger {@
                display: block;@
            }@
            .sidebar {@
                left: -250px;@
            }@
            .sidebar.active {@
                left: 0;@
            }@
            .main-content {@
                margin-left: 0;@
            }@
            .header-title {@
                margin-left: 20px;@
            }@
        }@
</style>@ <!-- Icon library for menu icons -->@
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/@fortawesome/fontawesome-free@6.4.0/css/all.min.css">@</head>@<body>@ <!-- Header -->@ <header class="header">@
    <button class="hamburger" id="menuToggle">@
    <i class="fas fa-bars"></i>@ </button>@ <div class="header-title">User Management</div>@ <div class="user-info-mini">@
        <span id="headerUserName"></span>@ </div>@ </header>@
<!-- Sidebar -->@ <aside class="sidebar" id="sidebar">@ <div class="sidebar-header">@
    <div class="user-info">@
        <div class="user-avatar" id="userAvatar"></div>@
        <div class="user-name" id="sidebarUserName">Loading...</div>@
    </div>@
    <div class="user-status" id="userStatus">Online</div>@
</div>@
</div>@ </div>@ <ul class="nav-menu">@
    <li><a href="#" class="active"><i class="fas fa-tachometer-alt"></i> Dashboard</a></li>@
    <li><a href="#"><i class="fas fa-user"></i> Profile</a></li>@
    <li><a href="#"><i class="fas fa-cog"></i> Settings</a></li>@
    <li><a href="#"><i class="fas fa-bar-chart"></i> Analytics</a></li>@
    <li><a href="#"><i class="fas fa-bell"></i> Notifications</a></li>@
    <li><a href="#"><i class="fas fa-sign-out-alt"></i> Logout</a></li>@
</ul>@
</div>@
</body>@
</html>@

```

```

<i class="fas fa-sign-out-alt" style="margin-right: 10px;"></i> Logout</button></aside><!-- Main Content -->
<main class="main-content" id="mainContent">
  <div class="dashboard-header">
    <h1 class="dashboard-title">Dashboard</h1>
    <div class="breadcrumb">Home > Dashboard</div>
    <div class="data-card">
      <h2>Session Information</h2>
      <div id="sessionDataContainer">
        <div id="sessionLoading"></div>
        <table class="session-table" id="sessionTable" style="display: none;">
          <thead>
            <tr>
              <th>Sign-in Time</th>
              <th>User ID</th>
              <th>Email</th>
              <th>Logout Time</th>
              <th>Phone</th>
              <th>Status</th>
            </tr>
          </thead>
          <tbody>
            <tr>
              <td id="userIdDisplay"></td>
              <td id="signinTimeDisplay"></td>
              <td id="logoutTimeDisplay"></td>
              <td id="emailDisplay"></td>
              <td id="statusDisplay"></td>
            </tr>
          </tbody>
        </table>
      </div>
    </div>
  </main>
  <script>
    // DOM Elements
    const menuToggle = document.getElementById('menuToggle');
    const sidebar = document.getElementById('sidebar');
    const mainContent = document.getElementById('mainContent');
    const logoutBtn = document.getElementById('logout');
    const sessionLoading = document.getElementById('sessionLoading');
    const sessionTable = document.getElementById('sessionTable');
    menuToggle.addEventListener('click', () => {
      sidebar.classList.toggle('active');
      mainContent.classList.toggle('expanded');
    });
    // Format datetime for display
    function formatDate(dateTimeStr) {
      if (!dateTimeStr) return 'N/A';
      const date = new Date(dateTimeStr);
      return Intl.DateTimeFormat('en-US', {
        year: 'numeric',
        month: 'short',
        day: 'numeric',
        hour: '2-digit',
        minute: '2-digit',
        second: '2-digit'
      }).format(date);
    }
    // Check if user is logged in
    function checkUserLoggedIn() {
      const userId = sessionStorage.getItem("userId");
      const userName = sessionStorage.getItem("userName");
      // Display user info in sidebar and header
      document.getElementById("sidebarUserName").textContent = userName;
      document.getElementById("headerUserName").textContent = userName;
      document.getElementById("userAvatar").textContent = userName.charAt(0).toUpperCase();
      return true;
    }
    // Load dashboard data
    function loadDashboardData() {
      if (!checkUserLoggedIn()) return;
      const userId = sessionStorage.getItem("userId");
      // Show loading indicator
      sessionLoading.style.display = 'flex';
      sessionTable.style.display = 'none';
      // Fetch user session data
      fetch(`http://localhost:8080/api/user_sessions/${userId}`)
        .then(response => {
          if (!response.ok) {
            throw new Error('Network response was not ok');
          }
          return response.json();
        })
        .then(data => {
          // Hide loading indicator and show table
          sessionLoading.style.display = 'none';
          sessionTable.style.display = 'table';
          // Populate table with data

```

```

document.getElementById("userIdDisplay").textContent = data.userId || "N/A";
document.getElementById("signinTimeDisplay").textContent
= formatDateTime(data.signinTime);
document.getElementById("logoutTimeDisplay").textContent = data.logoutTime ?
formatDateTime(data.logoutTime) : "Active";
document.getElementById("emailDisplay").textContent = data.email || "N/A";
document.getElementById("phoneDisplay").textContent =
data.phone || "N/A";
const statusDisplay =
document.getElementById("statusDisplay");
if (data.isActive) {
statusDisplay.innerHTML = '<span
class="status active">Online</span>';
document.getElementById("userStatus").textContent = "Online";
document.getElementById("userStatus").style.color =
"var(--success-color)";
} else {
statusDisplay.innerHTML = '<span class="status inactive">Offline</span>';
document.getElementById("userStatus").textContent =
"Offline";
document.getElementById("userStatus").style.color = "var(--danger-color)";
}
.catch(error => {
console.error("Error loading dashboard data:", error);
sessionLoading.style.display = 'none';
alert("Failed to load dashboard data. Please try again.");
});
// Handle logout
logoutBtn.addEventListener('click',
function() {
const userId = sessionStorage.getItem("userId");
if (!userId) {
alert("No active
session found. Redirecting to login page...");
window.location.href = "signin.html";
return;
}
// Disable logout button to prevent multiple clicks
logoutBtn.disabled = true;
logoutBtn.innerHTML = '<i
class="fas fa-spinner fa-spin" style="margin-right: 10px;"></i> Logging
out...';
logoutBtn.addEventListener('click',
function() {
const userId = sessionStorage.getItem("userId");
// Disable logout button to prevent multiple clicks
logoutBtn.disabled = true;
logoutBtn.innerHTML = '<i class="fas fa-spinner fa-spin" style="margin-
right: 10px;"></i> Logging out...';
// Check if we have a userId to
send to the server
if (!userId) {
console.log("No userId found in
session storage");
// Just redirect to login since there's no session
to clear server-side
sessionStorage.clear();
window.location.href = "signin.html";
return;
}
// Send
logout request to server
fetch("http://localhost:8080/api/logout", {
method: "POST",
headers: {
"Content-Type":
"application/json"
},
body: JSON.stringify({ userId: userId })
})
.then(response => {
// Always clear local session data
regardless of server response
sessionStorage.clear();
// Process the response
if (!response.ok) {
console.warn("Server returned error during logout, but continuing with client
logout");
window.location.href = "signin.html";
return
null;
// Don't try to parse JSON if response is not OK
}
return
response.json();
})
.then(data => {
if (data) {
console.log("Logout response:", data);
// Even if
server says "no active session", we still want to logout locally
if (data.message.includes("No active session")) {
console.log("Server reports no active session, but proceeding with client
logout");
}
// Always redirect to login
page
window.location.href = "signin.html";
}
}
).catch(error => {
console.error("Logout error:", error);
// Even if
there's an error, clear session and redirect
sessionStorage.clear();
window.location.href = "signin.html";
}
}
}
}

```

```

    });});
Initialize dashboard on page load
document.addEventListener('DOMContentLoaded', function() {
loadDashboardData();
    }); // Refresh dashboard data every 30
seconds
    setInterval(loadDashboardData, 30000);
</script>
/body>
/html>
```

demo\src\main\resources\static\admin_dashboard.css

```
* {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
    font-family: 'Segoe UI', sans-serif;
}
body {
    display: flex;
    background: #f5f7fa;
    height: 100vh;
    overflow: hidden;
}
header {
    position: fixed;
    top: 0;
    left: 0;
    width: 100%;
    height: 60px;
    background: white;
    color: black;
    display: flex;
    align-items: center;
    padding: 0 20px;
    z-index: 1000;
    box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);
}
img {
    width: 130px;
    height: 60px;
}
.user-role {
    font-size: 12px;
    color: black;
    margin-top: 2px;
}
.user-data {
    display: flex;
    flex-direction: column;
    align-items: center;
    /* Center align text */
}
.username {
    font-size: 19px;
    font-weight: bold;
}
hamburger {
    background: none;
    border: none;
    color: black;
    font-size: 22px;
    cursor: pointer;
    margin-right: 15px;
}
header-title {
    font-size: 20px;
    flex: 1;
}
.user-info-mini {
    display: flex;
    align-items: center;
    font-size: 16px;
    font-weight: 600;
    justify-content: center;
    gap: 20px;
}
.user-info-mini.name {
    text-align: center;
}
.sidebar {
    position: fixed;
    top: 60px;
    left: 0;
    width: 240px;
    height: calc(100% - 60px);
    background: #c6d5dd;
    color: black;
    transition: transform 0.3s ease-in-out;
    overflow-y: auto;
    z-index: 999;
}
.sidebar.active {
    transform: translateX(-100%);
}
.sidebar-header {
    padding: 20px;
    border-bottom: 1px solid black;
}
.user-info {
    display: flex;
    align-items: center;
}
.user-avatar {
    background: #f39c12;
    color: white;
    font-weight: bold;
    width: 45px;
    height: 45px;
    border-radius: 50%;
    display: flex;
    align-items: center;
    justify-content: center;
    margin-right: 15px;
    font-size: 20px;
    animation: popIn 0.4s ease;
}
.user-name {
    font-size: 16px;
    font-weight: bold;
}
.user-status {
    font-size: 13px;
    color: lightgreen;
}
.nav-menu {
    list-style: none;
    padding: 10px 0;
}
.nav-menu li a {
    display: flex;
    align-items: center;
    padding: 12px 20px;
    color: black;
    font-size: bold;
    font-weight: bold;
    text-decoration: none;
    transition: background 0.3s;
}
.nav-menu li a:hover,
.nav-menu li a.active {
    background: #34495e;
    color: #fff;
}
.nav-menu i {
    margin-right: 12px;
}
.logout-btn {
    width: 90%;
    margin: 20px auto;
    display: flex;
    align-items: center;
    justify-content: center;
    background: #e74c3c;
    color: white;
    border: none;
    padding: 12px;
    border-radius: 6px;
    font-size: 16px;
    cursor: pointer;
    transition: background 0.3s ease;
}
.logout-btn:hover {
    background: #c0392b;
}
main-content {
    margin-left: 240px;
    padding: 80px 30px 30px;
    width: 100%;
    overflow-y: auto;
    transition: margin-left 0.3s ease-in-out;
}
main-content.expanded {
    margin-left: 0;
}
dashboard-header {
    margin-bottom: 20px;
    animation: fadeInUp 0.5s ease;
}
dashboard-title {
    font-size: 28px;
    margin-bottom: 5px;
}
breadcrumb {
    color: #888;
}
data-card {
    background: white;
    padding: 25px;
    border-radius: 10px;
    box-shadow: 0 3px 10px rgba(0, 0, 0, 0.05);
    animation: fadeIn 1s ease;
}
loading {
    padding: 10px;
    color: #333;
}
session-table {
    width: 100%;
    border-collapse: collapse;
    margin-top: 15px;
}
session-table th,
session-table td {
    padding: 12px;
    border: 1px solid #ddd;
    text-align: center;
}
session-table th {
    background: #2e3a59;
    color: white;
}
session-table td {
    font-size: 15px;
}
/* Notification Icon */
notification-icon {
    cursor: pointer;
    font-size: 22px;
    position: relative;
}
notificationCount {
    background: red;
    color: white;
    padding: 2px 6px;
    border-radius: 50%;
    position: absolute;
    top: -8px;
    right: -8px;
    font-size: 12px;
}
/* Notifications */
notification-panel {
    position: absolute;
    top: 60px;
    right: 20px;
    width: 280px;
    background: #fff;
    border: 1px solid #ccc;
    box-shadow: 0 4px 10px rgba(0, 0, 0, 0.15);
    border-radius: 8px;
    padding: 15px 20px;
    z-index: 1000;
}
notification-panel ul {
    list-style: none;
    padding: 0;
}
notification-panel ul li {
    padding: 10px;
    border-bottom: 1px solid #ddd;
    cursor: pointer;
}
@keyframes fadeIn {
    from {opacity: 0; transform: translateY(10px);}
    to {opacity: 1; transform: translateY(0);}
}
@keyframes fadeInUp {
    from {opacity: 0; transform: translateY(20px);}
    to {opacity: 1; transform: translateY(0);}
```

```

        to {opacity: 1; transform: translateY(0);}
    }
}
@keyframes
popIn {
    0% { transform: scale(0); opacity: 0; }
    100% { transform: scale(1); opacity: 1; }
}
/* Responsive */
@media (max-width: 768px) {
    .sidebar {
        transform: translateX(-100%);
    }
    .sidebar.active {
        transform: translateX(0);
    }
    .main-content {
        margin-left: 0;
        padding: 70px 20px;
    }
    .user-table {
        width: 100%;
        border-collapse: collapse;
        margin-top: 15px;
    }
    .user-table th, .user-table td {
        border: 1px solid #ddd;
        padding: 10px;
        text-align: center;
    }
    .user-table th {
        background-color: #34495e;
        font-weight: bold;
        color: white;
    }
    .user-table tr:nth-child(even) {
        background-color: #f9f9f9;
    }
    .loading-message {
        font-style: italic;
        color: #666;
        margin-bottom: 10px;
    }
    #searchInput {
        padding: 10px 12px;
        font-size: 14px;
        border: 1px solid #ccc;
        border-radius: 6px;
        outline: none;
        transition: border-color 0.3s ease;
    }
    #searchInput:focus {
        border-color: #1f77b4;
    }
    #sortSelect {
        padding: 10px 12px;
        font-size: 14px;
        border: 2px solid #ccc;
        border-radius: 6px;
        outline: none;
        transition: border-color 0.3s ease;
    }
    #sortSelect:focus {
        border-color: #1f77b4;
    }
    .download-btn {
        background-color: #1f77b4;
        color: white;
        border: none;
        padding: 10px 16px;
        border-radius: 6px;
        cursor: pointer;
        font-size: 14px;
        margin-bottom: 10px;
        transition: background-color 0.3s ease;
    }
    .download-btn:hover {
        background-color: #145a86;
    }
    #allUserSortSelect {
        padding: 10px 12px;
        font-size: 14px;
        border: 1px solid #ccc;
        border-radius: 6px;
        outline: none;
        transition: border-color 0.3s ease;
    }
    #allUserSortSelect:focus {
        border-color: #1f77b4;
    }
    #allUserSearchInput {
        padding: 10px 12px;
        font-size: 14px;
        border: 1px solid #ccc;
        border-radius: 6px;
        outline: none;
        transition: border-color 0.3s ease;
    }
    #allUserSearchInput:focus {
        border-color: #1f77b4;
    }
    #chatModule {
        max-width: 800px;
        margin: 20px auto;
        font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
    }
    .chat-box {
        height: 400px;
        overflow-y: auto;
        border: 1px solid #ccc;
        border-radius: 10px;
        padding: 10px;
        background: #f9f9f9;
        display: flex;
        flex-direction: column;
        gap: 8px;
    }
    .chat-message {
        max-width: 75%;
        padding: 10px 15px;
        border-radius: 20px;
        font-size: 14px;
        word-wrap: break-word;
        display: inline-block;
        position: relative;
        animation: fadeIn 0.2s ease-in-out;
    }
    .user-message {
        background-color: #d0eaff;
        color: #000;
        align-self: flex-start;
        border-top-left-radius: 0;
    }
    .admin-message {
        background-color: #e0e0e0;
        color: #000;
        align-self: flex-end;
        border-top-right-radius: 0;
    }
    .chat-input-area {
        display: flex;
        margin-top: 10px;
        gap: 8px;
    }
    .chat-input-area input {
        flex: 1;
        padding: 10px;
        border-radius: 20px;
        border: 1px solid #ccc;
        outline: none;
        font-size: 14px;
    }
    .chat-input-area button {
        padding: 10px 20px;
        border: none;
        background-color: #007bff;
        color: white;
        border-radius: 20px;
        cursor: pointer;
        transition: background-color 0.2s;
    }
    .chat-input-area button:hover {
        background-color: #0056b3;
    }
    #loadingIndicator {
        text-align: center;
        margin: 10px 0;
        color: #888;
    }
    @keyframes fadeIn {
        from { opacity: 0; transform: translateY(5px); }
        to { opacity: 1; transform: translateY(0); }
    }
    .timestamp {
        font-size: 11px;
        color: #666;
        margin-top: 5px;
        text-align: right;
    }
    #profile-card {
        display: flex;
        padding: 20px;
        border-radius: 12px;
        box-shadow: 0 0 8px rgba(0,0,0,0.1);
        background-color: #fff;
        gap: 30px;
        align-items: center;
        margin-top: 20px;
        flex-wrap: wrap;
    }
    #profile-photo img {
        width: 120px;
        height: 120px;
        border-radius: 50%;
        object-fit: cover;
    }
    #profile-details {
        display: flex;
        flex-direction: column;
        gap: 10px;
        flex: 1;
    }
    #profile-row {
        display: flex;
        justify-content: flex-start;
        gap: 15px;
        font-size: 16px;
    }
    #profile-label {
        font-weight: bold;
        min-width: 150px;
    }
    #settingsModule {
        max-width: 800px;
        width: 100%;
        background: #fff;
        padding: 30px;
        border-radius: 12px;
        box-shadow: 0 4px 20px rgba(0, 0, 0, 0.1);
        opacity: 0;
        transform: translateY(20px);
        animation: slideIn 0.5s ease-out forwards;
    }
}

```

```

        @keyframes slideIn {
            to {
                transform: translateY(0);
                opacity: 1;
            }
            .dashboard-header {
                margin-bottom: 20px;
            }
            .dashboard-title {
                font-size: 24px;
                color: #1f2937;
                font-weight: 600;
            }
            .breadcrumb {
                font-size: 14px;
                color: #6b7280;
            }
            h2 {
                font-size: 22px;
                color: #1f2937;
                margin-bottom: 25px;
                font-weight: 500;
            }
            .settings-section {
                margin-bottom: 20px;
            }
            label {
                display: block;
                font-size: 17px;
                color: #374151;
                margin-bottom: 8px;
                font-weight: 500;
            }
            .current-value {
                font-size: 17px;
                color: #374151;
                margin-bottom: 10px;
                display: inline-block;
                padding: 5px 10px;
                background: #f9fafb;
                border-radius: 4px;
            }
            .edit-section {
                display: none;
                margin-top: 10px;
                gap: 10px;
                align-items: center;
            }
            .edit-section.active {
                display: flex;
            }
            input[type="text"],
            input[type="tel"] {
                padding: 8px 12px;
                border: 1px solid #d1d5db;
                border-radius: 6px;
                font-size: 14px;
                width: 200px;
                transition: border-color 0.3s ease, box-shadow 0.3s ease;
            }
            input[type="text"]:focus,
            input[type="tel"]:focus {
                outline: none;
                border-color: #2563eb;
                box-shadow: 0 0 0 3px rgba(37, 99, 235, 0.1);
            }
            .save-button,
            .edit-button,
            .cancel-button {
                padding: 8px 16px;
                background: #2563eb;
                color: #fff;
                border: none;
                border-radius: 6px;
                font-size: 14px;
                cursor: pointer;
                transition: transform 0.2s ease, background 0.2s ease;
            }
            .cancel-button {
                background: #ef4444; /* Red for cancel */
            }
            .cancel-button:hover {
                background: #dc2626;
            }
            .save-button:hover,
            .edit-button:hover,
            .cancel-button:hover {
                transform: scale(1.05);
            }
            .save-button:active,
            .edit-button:active,
            .cancel-button:active {
                transform: scale(0.95);
            }
            .profile-pic {
                width: 100px;
                height: 100px;
                border-radius: 50%;
                object-fit: cover;
                margin-top: 10px;
                border: 2px solid #d1d5db;
            }
            input[type="file"] {
                font-size: 14px;
            }
            #successPopup {
                display: none;
                position: fixed;
                top: 50%;
                left: 50%;
                transform: translate(-50%, -50%);
                background: #10b981;
                color: white;
                padding: 20px;
                border-radius: 8px;
                box-shadow: 0 4px 20px rgba(0, 0, 0, 0.2);
                z-index: 1000;
                opacity: 0;
                animation: popIn 0.3s ease-out forwards;
            }
            @keyframes popIn {
                from {
                    opacity: 0;
                }
                to {
                    transform: translate(-50%, -50%);
                    opacity: 1;
                }
            }
            #successPopup button {
                margin-top: 10px;
                padding: 8px 16px;
                background: #047857;
                border: none;
                border-radius: 6px;
                cursor: pointer;
            }
            #successPopup button:hover {
                background: #065f46;
            }
            .status-msg {
                font-size: 14px;
                margin-top: 15px;
                opacity: 0;
                transition: opacity 0.3s ease;
            }
            .status-msg.visible {
                opacity: 1;
            }
        }
        @media (max-width: 480px) {
            #settingsModule {
                padding: 20px;
            }
            .dashboard-title {
                font-size: 20px;
            }
            h2 {
                font-size: 18px;
            }
            input[type="text"],
            input[type="tel"] {
                width: 150px;
            }
        }

```



```
flex-direction: column; }
#successPopup {
padding: 15px;
width: 80%;
}
.edit-section {
gap: 5px;
}
```

demo\src\main\resources\static\user_dashboard3.css

```
@* {margin: 0; padding: 0; box-sizing: border-box; font-family: 'Segoe UI', sans-serif;}body {display: flex; background: #f5f7fa; height: 100vh; overflow: hidden;}header {position: fixed; top: 0; left: 0; width: 100%; height: 60px; background: white; color: black; display: flex; align-items: center; padding: 0 20px; z-index: 1000; box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);}hamburger {background: none; border: none; color: black; font-size: 22px; cursor: pointer; margin-right: 15px;}header-title {font-size: 20px; flex: 1;}user-info-mini {display: flex; align-items: center; font-size: 16px; font-weight: 600; justify-content: center; } user-info-mini .name {text-align: center;} .username {font-size: 19px; font-weight: bold;}sidebar {position: fixed; top: 60px; left: 0; width: 240px; height: calc(100% - 60px); background: #c6d5dd; color: black; transition: transform 0.3s ease-in-out; overflow-y: auto; z-index: 999;}sidebar.active {transform: translateX(-100%);}sidebar-header {padding: 20px; border-bottom: 1px solid black;}user-info {display: flex; align-items: center;}image {width: 130px; height: 60px;}user-data {display: flex; flex-direction: column; align-items: center; /* Center align text */ } user-avatar {background: #f39c12; color: white; font-weight: bold; width: 45px; height: 45px; border-radius: 50%; display: flex; align-items: center; justify-content: center; margin-right: 15px; font-size: 20px; animation: popIn 0.4s ease;}user-name {font-size: 16px; font-weight: bold;}user-status {font-size: 13px; color: lightgreen;}nav-menu {list-style: none; padding: 10px 0;} .nav-menu li a {display: flex; align-items: center; padding: 12px 20px; color: black; font-size: bold; font-weight: bold; text-decoration: none; transition: background 0.3s;} .nav-menu li a:hover, .nav-menu li a.active {background: #34495e; color: #fff;} .nav-menu i {margin-right: 12px;}logout-btn {width: 90%; margin: 20px auto; display: flex; align-items: center; justify-content: center; background: #e74c3c; color: white; border: none; padding: 12px; border-radius: 6px; font-size: 16px; cursor: pointer; transition: background 0.3s ease;}logout-btn:hover {background: #c0392b;}main-content {margin-left: 240px; padding: 80px 30px 30px; width: 100%; overflow-y: auto; transition: margin-left 0.3s ease-in-out;}main-content.expanded {margin-left: 0;}dashboard-header {margin-bottom: 20px; animation: fadeInUp 0.5s ease;}dashboard-title {font-size: 28px; margin-bottom: 5px;}breadcrumb {color: #888;}data-card {background: white; padding: 25px; border-radius: 10px; box-shadow: 0 3px 10px rgba(0,0,0,0.05); animation: fadeIn 1s ease;}loading {padding: 10px; color: #333;}session-table {width: 100%; border-collapse: collapse; margin-top: 15px;}session-table th, session-table td {padding: 12px; border: 1px solid #ddd; text-align: center;}session-table th {background: #2e3a59; color: white;}session-table td {font-size: 15px;}@keyframes fadeIn {from {opacity: 0; transform: translateY(10px);} to {opacity: 1; transform: translateY(0);}}@keyframes fadeInUp {from {opacity: 0; transform: translateY(20px);} to {opacity: 1; transform: translateY(0);}}@keyframes popIn {0% {transform: scale(0); opacity: 0;} 100% {transform: scale(1); opacity: 1;}}/* Responsive */@media (max-width: 768px) { .sidebar {transform: translateX(-100%);} .sidebar.active {transform: translateX(0);} .main-content {margin-left: 0; padding: 70px 20px;}profile-card {display: flex; padding: 20px; border-radius: 12px; box-shadow: 0 8px 0px rgba(0,0,0,0.1); background-color: #fff; gap: 30px; align-items: center; margin-top: 20px; flex-wrap: wrap;}profile-photo img {width: 120px; height: 120px; border-radius: 50%; object-fit: cover;}profile-details {
```

```

        display: flex;
        flex-direction: column;
        gap: 10px;
        flex: 1;
    }
    .profile-row {
        display: flex;
        justify-content: flex-start;
        gap: 15px;
        font-size: 16px;
    }
    .profile-label {
        font-weight: bold;
        min-width: 150px;
    }
    #chatModule {
        max-width: 800px;
        margin: 20px auto;
        font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
    }
    .chat-box {
        height: 400px;
        overflow-y: auto;
        border: 1px solid #ccc;
        border-radius: 10px;
        padding: 10px;
        background: #f9f9f9;
        display: flex;
        flex-direction: column;
        gap: 8px;
    }
    .chat-message {
        max-width: 75%;
        padding: 10px 15px;
        border-radius: 20px;
        font-size: 14px;
        word-wrap: break-word;
        display: inline-block;
        position: relative;
        animation: fadeIn 0.2s ease-in-out;
    }
    .user-message {
        background-color: #d0eaff;
        color: #000;
        align-self: flex-start;
        border-top-left-radius: 0;
    }
    .admin-message {
        background-color: #e0e0e0;
        color: #000;
        align-self: flex-end;
        border-top-right-radius: 0;
    }
    .chat-input-area {
        display: flex;
        margin-top: 10px;
        gap: 8px;
    }
    .chat-input-area input {
        flex: 1;
        padding: 10px;
        border-radius: 20px;
        border: 1px solid #ccc;
        outline: none;
        font-size: 14px;
    }
    .chat-input-area button {
        padding: 10px 20px;
        border: none;
        background-color: #007bff;
        color: white;
        border-radius: 20px;
        cursor: pointer;
        transition: background-color 0.2s;
    }
    .chat-input-area button:hover {
        background-color: #0056b3;
    }
    #loadingIndicator {
        text-align: center;
        margin: 10px 0;
        color: #888;
    }
    @keyframes fadeIn {
        from { opacity: 0; transform: translateY(5px); }
        to { opacity: 1; transform: translateY(0); }
    }
    .timestamp {
        font-size: 11px;
        color: #666;
        margin-top: 5px;
        text-align: right;
    }
    /* Base styling */
    #settingsModule {
        max-width: 800px;
        width: 100%;
        background: #fff;
        padding: 30px;
        border-radius: 12px;
        box-shadow: 0 4px 20px rgba(0, 0, 0, 0.1);
        opacity: 0;
        transform: translateY(20px);
        animation: slideIn 0.5s ease-out forwards;
    }
    @keyframes slideIn {
        to {
            transform: translateY(0);
            opacity: 1;
        }
    }
    .dashboard-header {
        margin-bottom: 20px;
    }
    .dashboard-title {
        font-size: 24px;
        color: #1f2937;
        font-weight: 600;
    }
    .breadcrumb {
        font-size: 14px;
        color: #6b7280;
    }
    h2 {
        font-size: 22px;
        color: #1f2937;
        margin-bottom: 25px;
        font-weight: 500;
    }
    .settings-section {
        margin-bottom: 20px;
    }
    label {
        font-size: 17px;
        color: #374151;
        margin-bottom: 8px;
        font-weight: 500;
    }
    .current-value {
        font-size: 17px;
        color: #374151;
        margin-bottom: 10px;
        display: inline-block;
        padding: 5px 10px;
        background: #f9fafb;
        border-radius: 4px;
    }
    .edit-section {
        display: none;
        margin-top: 10px;
        gap: 10px;
        align-items: center;
    }
    .edit-section.active {
        display: flex;
    }
    input[type="text"],
    input[type="tel"] {
        padding: 8px 12px;
        border: 1px solid #d1d5db;
        border-radius: 6px;
        font-size: 14px;
        width: 200px;
        transition: border-color 0.3s ease, box-shadow 0.3s ease;
    }
    input[type="text"]:focus,
    input[type="tel"]:focus {
        outline: none;
        border-color: #2563eb;
        box-shadow: 0 0 0 3px rgba(37, 99, 235, 0.1);
    }
    .save-button,
    .edit-button,
    .cancel-button {
        padding: 8px 16px;
        background: #2563eb;
        color: #fff;
        border: none;
        border-radius: 6px;
        font-size: 14px;
        cursor: pointer;
        transition: transform 0.2s ease, background 0.2s ease;
    }
    .cancel-button {
        background: #ef4444; /* Red for cancel */
    }
    .cancel-button:hover {
        background: #dc2626;
    }

```

```

        .save-button:hover, .edit-button:hover,
        .cancel-button:hover {
            transform: scale(1.05);
        }
        .save-button:active, .edit-button:active, .cancel-
button:active {
            transform: scale(0.95);
        }
        .profile-
pic {
            width: 100px;
            height: 100px;
            border-
radius: 50%;
            object-fit: cover;
            margin-top: 10px;
            border: 2px solid #d1d5db;
        }
        input[type="file"] {
            margin-top: 10px;
            font-size: 14px;
        }
        #successPopup {
            display: none;
            position: fixed;
            top: 50%;
            left: 50%;
            transform:
translate(-50%, -50%);
            background: #10b981;
            color:
white;
            padding: 20px;
            border-radius: 8px;
            box-shadow: 0 4px 20px rgba(0, 0, 0, 0.2);
            z-index: 1000;
            opacity: 0;
            animation: popIn 0.3s ease-out forwards;
        }
        @keyframes popIn {
            from {
                transform: translate(-50%, -60%) scale(0.8);
                opacity: 0;
            }
            to {
                transform: translate(-50%, -50%)
scale(1);
                opacity: 1;
            }
        }
        #successPopup button {
            margin-top: 10px;
            padding: 8px
16px;
            background: #047857;
            border: none;
            color: white;
            border-radius: 6px;
            cursor: pointer;
        }
        #successPopup button:hover {
            background: #065f46;
        }
        .status-msg {
            font-size: 14px;
            margin-
top: 15px;
            opacity: 0;
            transition: opacity 0.3s ease;
        }
        .status-msg.visible {
            opacity: 1;
        }
        @media (max-width: 480px) {
            #settingsModule {
                padding: 20px;
            }
            .dashboard-title {
                font-
size: 20px;
            }
            h2 {
                font-size: 18px;
            }
            input[type="text"],
            input[type="tel"] {
                width: 150px;
            }
            .edit-section {
                flex-direction: column;
                gap: 5px;
            }
            #successPopup {
                width: 80%;
                padding: 15px;
            }
        }
    
```

demo\target\classes\static\admin_dashboard.css

```
* {
    margin: 0;
    padding: 0;
    box-sizing: border-box;
    font-family: 'Segoe UI', sans-serif;
}
body {
    display: flex;
    background: #f5f7fa;
    height: 100vh;
    overflow: hidden;
}
header {
    position: fixed;
    top: 0;
    left: 0;
    width: 100%;
    height: 60px;
    background: white;
    color: black;
    display: flex;
    align-items: center;
    padding: 0 20px;
    z-index: 1000;
    box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);
}
img {
    width: 130px;
    height: 60px;
}
.user-role {
    font-size: 12px;
    color: black;
}
.user-data {
    display: flex;
    flex-direction: column;
    align-items: center;
    /* Center align text */
}
.username {
    font-size: 19px;
    font-weight: bold;
}
hamburger {
    background: none;
    border: none;
    color: black;
    font-size: 22px;
    cursor: pointer;
    margin-right: 15px;
}
header-title {
    font-size: 20px;
    flex: 1;
}
.user-info-mini {
    display: flex;
    align-items: center;
    font-size: 16px;
    font-weight: 600;
    justify-content: center;
    gap: 20px;
}
.user-info-mini .name {
    text-align: center;
}
.sidebar {
    position: fixed;
    top: 60px;
    left: 0;
    width: 240px;
    height: calc(100% - 60px);
    background: #c6d5dd;
    color: black;
    transition: transform 0.3s ease-in-out;
    overflow-y: auto;
    z-index: 999;
}
.sidebar.active {
    transform: translateX(-100%);
}
.sidebar-header {
    padding: 20px;
    border-bottom: 1px solid black;
}
.user-info {
    display: flex;
    align-items: center;
}
.user-avatar {
    background: #f39c12;
    color: white;
    font-weight: bold;
    width: 45px;
    height: 45px;
    border-radius: 50%;
    display: flex;
    align-items: center;
    justify-content: center;
}
.user-name {
    font-size: 16px;
    font-weight: bold;
}
.user-status {
    font-size: 13px;
    color: lightgreen;
}
.nav-menu {
    list-style: none;
    padding: 10px 0;
}
.nav-menu li a {
    display: flex;
    align-items: center;
    padding: 12px 20px;
    color: black;
    font-size: bold;
    font-weight: bold;
    text-decoration: none;
    transition: background 0.3s;
}
.nav-menu li a:hover,
.nav-menu li a.active {
    background: #34495e;
    color: #fff;
}
.nav-menu i {
    margin-right: 12px;
}
.logout-btn {
    width: 90%;
    margin: 20px auto;
    display: flex;
    align-items: center;
    justify-content: center;
    background: #e74c3c;
    color: white;
    border: none;
    padding: 12px;
    border-radius: 6px;
    font-size: 16px;
    cursor: pointer;
    transition: background 0.3s ease;
}
.logout-btn:hover {
    background: #c0392b;
}
main-content {
    margin-left: 240px;
    padding: 80px 30px 30px;
    width: 100%;
    overflow-y: auto;
    transition: margin-left 0.3s ease-in-out;
}
main-content.expanded {
    margin-left: 0;
}
dashboard-header {
    margin-bottom: 20px;
    animation: fadeInUp 0.5s ease;
}
dashboard-title {
    font-size: 28px;
    margin-bottom: 5px;
}
breadcrumb {
    color: #888;
}
data-card {
    background: white;
    padding: 25px;
    border-radius: 10px;
    box-shadow: 0 3px 10px rgba(0, 0, 0, 0.05);
    animation: fadeIn 1s ease;
}
loading {
    padding: 10px;
    color: #333;
}
session-table {
    width: 100%;
    border-collapse: collapse;
    margin-top: 15px;
}
session-table th,
session-table td {
    padding: 12px;
    border: 1px solid #ddd;
    text-align: center;
}
session-table th {
    background: #2e3a59;
    color: white;
}
session-table td {
    font-size: 15px;
}
/* Notification Icon */
notification-icon {
    cursor: pointer;
    font-size: 22px;
    position: relative;
}
notificationCount {
    background: red;
    color: white;
    padding: 2px 6px;
    border-radius: 50%;
    position: absolute;
    top: -8px;
    right: -8px;
    font-size: 12px;
}
/* Notifications */
notification-panel {
    position: absolute;
    top: 60px;
    right: 20px;
    width: 280px;
    background: #fff;
    border: 1px solid #ccc;
    box-shadow: 0 4px 10px rgba(0, 0, 0, 0.15);
    border-radius: 8px;
    padding: 15px 20px;
    z-index: 1000;
}
notification-panel ul {
    list-style: none;
    padding: 0;
}
notification-panel ul li {
    padding: 10px;
    border-bottom: 1px solid #ddd;
    cursor: pointer;
}
keyframes fadeIn {
    from {opacity: 0; transform: translateY(10px);}
    to {opacity: 1; transform: translateY(0);}
}
keyframes fadeInUp {
    from {opacity: 0; transform: translateY(20px);}
    to {opacity: 1; transform: translateY(0);}
}
```

```

        to {opacity: 1; transform: translateY(0);}
    }
}
@keyframes
popIn {
    0% { transform: scale(0); opacity: 0; }
    100% { transform: scale(1); opacity: 1; }
}
/* Responsive */
@media (max-width: 768px) {
    .sidebar {
        transform: translateX(-100%);
    }
    .sidebar.active {
        transform: translateX(0);
    }
    .main-content {
        margin-left: 0;
        padding: 70px 20px;
    }
    .user-table {
        width: 100%;
        border-collapse: collapse;
        margin-top: 15px;
    }
    .user-table th, .user-table td {
        border: 1px solid #ddd;
        padding: 10px;
        text-align: center;
    }
    .user-table th {
        background-color: #34495e;
        font-weight: bold;
        color: white;
    }
    .user-table tr:nth-child(even) {
        background-color: #f9f9f9;
    }
    .loading-message {
        font-style: italic;
        color: #666;
        margin-bottom: 10px;
    }
    #searchInput {
        padding: 10px 12px;
        font-size: 14px;
        border: 1px solid #ccc;
        border-radius: 6px;
        outline: none;
        transition: border-color 0.3s ease;
    }
    #searchInput:focus {
        border-color: #1f77b4;
    }
    #sortSelect {
        padding: 10px 12px;
        font-size: 14px;
        border: 2px solid #ccc;
        border-radius: 6px;
        outline: none;
        transition: border-color 0.3s ease;
    }
    #sortSelect:focus {
        border-color: #1f77b4;
    }
    .download-btn {
        background-color: #1f77b4;
        color: white;
        border: none;
        padding: 10px 16px;
        border-radius: 6px;
        cursor: pointer;
        font-size: 14px;
        margin-bottom: 10px;
        transition: background-color 0.3s ease;
    }
    .download-btn:hover {
        background-color: #145a86;
    }
    #allUserSortSelect {
        padding: 10px 12px;
        font-size: 14px;
        border: 1px solid #ccc;
        border-radius: 6px;
        outline: none;
        transition: border-color 0.3s ease;
    }
    #allUserSortSelect:focus {
        border-color: #1f77b4;
    }
    #allUserSearchInput {
        padding: 10px 12px;
        font-size: 14px;
        border: 1px solid #ccc;
        border-radius: 6px;
        outline: none;
        transition: border-color 0.3s ease;
    }
    #allUserSearchInput:focus {
        border-color: #1f77b4;
    }
    #chatModule {
        max-width: 800px;
        margin: 20px auto;
        font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
    }
    .chat-box {
        height: 400px;
        overflow-y: auto;
        border: 1px solid #ccc;
        border-radius: 10px;
        padding: 10px;
        background: #f9f9f9;
        display: flex;
        flex-direction: column;
        gap: 8px;
    }
    .chat-message {
        max-width: 75%;
        padding: 10px 15px;
        border-radius: 20px;
        font-size: 14px;
        word-wrap: break-word;
        display: inline-block;
        position: relative;
        animation: fadeIn 0.2s ease-in-out;
    }
    .user-message {
        background-color: #d0eaff;
        color: #000;
        align-self: flex-start;
        border-top-left-radius: 0;
    }
    .admin-message {
        background-color: #e0e0e0;
        color: #000;
        align-self: flex-end;
        border-top-right-radius: 0;
    }
    .chat-input-area {
        display: flex;
        margin-top: 10px;
        gap: 8px;
    }
    .chat-input-area input {
        flex: 1;
        padding: 10px;
        border-radius: 20px;
        border: 1px solid #ccc;
        outline: none;
        font-size: 14px;
    }
    .chat-input-area button {
        padding: 10px 20px;
        border: none;
        background-color: #007bff;
        color: white;
        border-radius: 20px;
        cursor: pointer;
        transition: background-color 0.2s;
    }
    .chat-input-area button:hover {
        background-color: #0056b3;
    }
    #loadingIndicator {
        text-align: center;
        margin: 10px 0;
        color: #888;
    }
    @keyframes fadeIn {
        from { opacity: 0; transform: translateY(5px); }
        to { opacity: 1; transform: translateY(0); }
    }
    .timestamp {
        font-size: 11px;
        color: #666;
        margin-top: 5px;
        text-align: right;
    }
    #profile-card {
        display: flex;
        padding: 20px;
        border-radius: 12px;
        box-shadow: 0 0 8px rgba(0,0,0,0.1);
        background-color: #fff;
        gap: 30px;
        align-items: center;
        margin-top: 20px;
        flex-wrap: wrap;
    }
    #profile-photo img {
        width: 120px;
        height: 120px;
        border-radius: 50%;
        object-fit: cover;
    }
    #profile-details {
        display: flex;
        flex-direction: column;
        gap: 10px;
        flex: 1;
    }
    #profile-row {
        display: flex;
        justify-content: flex-start;
        gap: 15px;
        font-size: 16px;
    }
    #profile-label {
        font-weight: bold;
        min-width: 150px;
    }
    #settingsModule {
        max-width: 800px;
        width: 100%;
        background: #fff;
        padding: 30px;
        border-radius: 12px;
        box-shadow: 0 4px 20px rgba(0, 0, 0, 0.1);
        opacity: 0;
        transform: translateY(20px);
        animation: slideIn 0.5s ease-out forwards;
    }
}

```

```

        @keyframes slideIn {
            to {
                transform: translateY(0);
                opacity: 1;
            }
            .dashboard-header {
                margin-bottom: 20px;
            }
            .dashboard-title {
                font-size: 24px;
                color: #1f2937;
                font-weight: 600;
            }
            .breadcrumb {
                font-size: 14px;
                color: #6b7280;
            }
            h2 {
                font-size: 22px;
                color: #1f2937;
                margin-bottom: 25px;
                font-weight: 500;
            }
            .settings-section {
                margin-bottom: 20px;
            }
            label {
                display: block;
                font-size: 17px;
                color: #374151;
                margin-bottom: 8px;
                font-weight: 500;
            }
            .current-value {
                font-size: 17px;
                color: #374151;
                margin-bottom: 10px;
                display: inline-block;
                padding: 5px 10px;
                background: #f9fafb;
                border-radius: 4px;
            }
            .edit-section {
                display: none;
                margin-top: 10px;
                gap: 10px;
                align-items: center;
            }
            .edit-section.active {
                display: flex;
            }
            input[type="text"],
            input[type="tel"] {
                padding: 8px 12px;
                border: 1px solid #d1d5db;
                border-radius: 6px;
                font-size: 14px;
                width: 200px;
                transition: border-color 0.3s ease, box-shadow 0.3s ease;
            }
            input[type="text"]:focus,
            input[type="tel"]:focus {
                outline: none;
                border-color: #2563eb;
                box-shadow: 0 0 0 3px rgba(37, 99, 235, 0.1);
            }
            .save-button,
            .edit-button,
            .cancel-button {
                padding: 8px 16px;
                background: #2563eb;
                color: #fff;
                border: none;
                border-radius: 6px;
                font-size: 14px;
                cursor: pointer;
                transition: transform 0.2s ease, background 0.2s ease;
            }
            .cancel-button {
                background: #ef4444; /* Red for cancel */
            }
            .cancel-button:hover {
                background: #dc2626;
            }
            .save-button:hover,
            .edit-button:hover,
            .cancel-button:hover {
                transform: scale(1.05);
            }
            .save-button:active,
            .edit-button:active,
            .cancel-button:active {
                transform: scale(0.95);
            }
            .profile-pic {
                width: 100px;
                height: 100px;
                border-radius: 50%;
                object-fit: cover;
                margin-top: 10px;
                border: 2px solid #d1d5db;
            }
            input[type="file"] {
                font-size: 14px;
            }
            #successPopup {
                display: none;
                position: fixed;
                top: 50%;
                left: 50%;
                transform: translate(-50%, -50%);
                background: #10b981;
                color: white;
                padding: 20px;
                border-radius: 8px;
                box-shadow: 0 4px 20px rgba(0, 0, 0, 0.2);
                z-index: 1000;
                opacity: 0;
                animation: popIn 0.3s ease-out forwards;
            }
            @keyframes popIn {
                from {
                    opacity: 0;
                }
                to {
                    transform: translate(-50%, -50%);
                    opacity: 1;
                }
            }
            #successPopup button {
                margin-top: 10px;
                padding: 8px 16px;
                background: #047857;
                border: none;
                border-radius: 6px;
                cursor: pointer;
            }
            #successPopup button:hover {
                background: #065f46;
            }
            .status-msg {
                font-size: 14px;
                margin-top: 15px;
                opacity: 0;
                transition: opacity 0.3s ease;
            }
            .status-msg.visible {
                opacity: 1;
            }
        }
        @media (max-width: 480px) {
            #settingsModule {
                padding: 20px;
            }
            .dashboard-title {
                font-size: 20px;
            }
            h2 {
                font-size: 18px;
            }
            input[type="text"],
            input[type="tel"] {
                width: 150px;
            }
        }

```

```
flex-direction: column; }
#successPopup {
padding: 15px;
width: 80%;
}
.edit-section {
gap: 5px;
}
```


demo\target\classes\static\user_dashboard3.css

```
@* {margin: 0; padding: 0; box-sizing: border-box; font-family: 'Segoe UI', sans-serif;}body {display: flex; background: #f5f7fa; height: 100vh; overflow: hidden;}header {position: fixed; top: 0; left: 0; width: 100%; height: 60px; background: white; color: black; display: flex; align-items: center; padding: 0 20px; z-index: 1000; box-shadow: 0 2px 5px rgba(0, 0, 0, 0.1);}hamburger {background: none; border: none; color: black; font-size: 22px; cursor: pointer; margin-right: 15px;}header-title {font-size: 20px; flex: 1;}user-info-mini {display: flex; align-items: center; font-size: 16px; font-weight: 600; justify-content: center; } user-info-mini .name {text-align: center;} .username {font-size: 19px; font-weight: bold;}sidebar {position: fixed; top: 60px; left: 0; width: 240px; height: calc(100% - 60px); background: #c6d5dd; color: black; transition: transform 0.3s ease-in-out; overflow-y: auto; z-index: 999;}sidebar.active {transform: translateX(-100%);}sidebar-header {padding: 20px; border-bottom: 1px solid black;}user-info {display: flex; align-items: center;}image {width: 130px; height: 60px;}user-data {display: flex; flex-direction: column; align-items: center; /* Center align text */ } user-avatar {background: #f39c12; color: white; font-weight: bold; width: 45px; height: 45px; border-radius: 50%; display: flex; align-items: center; justify-content: center; margin-right: 15px; font-size: 20px; animation: popIn 0.4s ease;}user-name {font-size: 16px; font-weight: bold;}user-status {font-size: 13px; color: lightgreen;}nav-menu {list-style: none; padding: 10px 0;}nav-menu li a {display: flex; align-items: center; padding: 12px 20px; color: black; font-size: bold; font-weight: bold; text-decoration: none; transition: background 0.3s;}nav-menu li a:hover, nav-menu li a.active {background: #34495e; color: #fff;}nav-menu i {margin-right: 12px;}logout-btn {width: 90%; margin: 20px auto; display: flex; align-items: center; justify-content: center; background: #e74c3c; color: white; border: none; padding: 12px; border-radius: 6px; font-size: 16px; cursor: pointer; transition: background 0.3s ease;}logout-btn:hover {background: #c0392b;}main-content {margin-left: 240px; padding: 80px 30px 30px; width: 100%; overflow-y: auto; transition: margin-left 0.3s ease-in-out;}main-content.expanded {margin-left: 0;}dashboard-header {margin-bottom: 20px; animation: fadeInUp 0.5s ease;}dashboard-title {font-size: 28px; margin-bottom: 5px;}breadcrumb {color: #888;}data-card {background: white; padding: 25px; border-radius: 10px; box-shadow: 0 3px 10px rgba(0,0,0,0.05); animation: fadeIn 1s ease;}loading {padding: 10px; color: #333;}session-table {width: 100%; border-collapse: collapse; margin-top: 15px;}session-table th, session-table td {padding: 12px; border: 1px solid #ddd; text-align: center;}session-table th {background: #2e3a59; color: white;}session-table td {font-size: 15px;}keyframes fadeIn {from {opacity: 0; transform: translateY(10px);} to {opacity: 1; transform: translateY(0);}}keyframes fadeInUp {from {opacity: 0; transform: translateY(20px);} to {opacity: 1; transform: translateY(0);}}keyframes popIn {0% {transform: scale(0); opacity: 0;} 100% {transform: scale(1); opacity: 1;}}/* Responsive */@media (max-width: 768px) { .sidebar {transform: translateX(-100%);} .sidebar.active {transform: translateX(0);} .main-content {margin-left: 0; padding: 70px 20px;}profile-card {display: flex; padding: 20px; border-radius: 12px; box-shadow: 0 8px 0px rgba(0,0,0,0.1); background-color: #fff; gap: 30px; align-items: center; margin-top: 20px; flex-wrap: wrap;}profile-photo img {width: 120px; height: 120px; border-radius: 50%; object-fit: cover;}profile-details {
```

```

        display: flex;
        flex-direction: column;
        gap: 10px;
        flex: 1;
    }
    .profile-row {
        display: flex;
        justify-content: flex-start;
        gap: 15px;
        font-size: 16px;
    }
    .profile-label {
        font-weight: bold;
        min-width: 150px;
    }
    #chatModule {
        max-width: 800px;
        margin: 20px auto;
        font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
    }
    .chat-box {
        height: 400px;
        overflow-y: auto;
        border: 1px solid #ccc;
        border-radius: 10px;
        padding: 10px;
        background: #f9f9f9;
        display: flex;
        flex-direction: column;
        gap: 8px;
    }
    .chat-message {
        max-width: 75%;
        padding: 10px 15px;
        border-radius: 20px;
        font-size: 14px;
        word-wrap: break-word;
        display: inline-block;
        position: relative;
        animation: fadeIn 0.2s ease-in-out;
    }
    .user-message {
        background-color: #d0eaff;
        color: #000;
        align-self: flex-start;
        border-top-left-radius: 0;
    }
    .admin-message {
        background-color: #e0e0e0;
        color: #000;
        align-self: flex-end;
        border-top-right-radius: 0;
    }
    .chat-input-area {
        display: flex;
        margin-top: 10px;
        gap: 8px;
    }
    .chat-input-area input {
        flex: 1;
        padding: 10px;
        border-radius: 20px;
        border: 1px solid #ccc;
        outline: none;
        font-size: 14px;
    }
    .chat-input-area button {
        padding: 10px 20px;
        border: none;
        background-color: #007bff;
        color: white;
        border-radius: 20px;
        cursor: pointer;
        transition: background-color 0.2s;
    }
    .chat-input-area button:hover {
        background-color: #0056b3;
    }
    #loadingIndicator {
        text-align: center;
        margin: 10px 0;
        color: #888;
    }
    @keyframes fadeIn {
        from { opacity: 0; transform: translateY(5px); }
        to { opacity: 1; transform: translateY(0); }
    }
    .timestamp {
        font-size: 11px;
        color: #666;
        margin-top: 5px;
        text-align: right;
    }
    /* Base styling */
    #settingsModule {
        max-width: 800px;
        width: 100%;
        background: #fff;
        padding: 30px;
        border-radius: 12px;
        box-shadow: 0 4px 20px rgba(0, 0, 0, 0.1);
        opacity: 0;
        transform: translateY(20px);
        animation: slideIn 0.5s ease-out forwards;
    }
    @keyframes slideIn {
        to {
            transform: translateY(0);
            opacity: 1;
        }
    }
    .dashboard-header {
        margin-bottom: 20px;
    }
    .dashboard-title {
        font-size: 24px;
        color: #1f2937;
        font-weight: 600;
    }
    .breadcrumb {
        font-size: 14px;
        color: #6b7280;
    }
    h2 {
        font-size: 22px;
        color: #1f2937;
        margin-bottom: 25px;
        font-weight: 500;
    }
    .settings-section {
        margin-bottom: 20px;
    }
    label {
        font-size: 17px;
        color: #374151;
        margin-bottom: 8px;
        font-weight: 500;
    }
    .current-value {
        font-size: 17px;
        color: #374151;
        margin-bottom: 10px;
        display: inline-block;
        padding: 5px 10px;
        background: #f9fafb;
        border-radius: 4px;
    }
    .edit-section {
        display: none;
        margin-top: 10px;
        gap: 10px;
        align-items: center;
    }
    .edit-section.active {
        display: flex;
    }
    input[type="text"],
    input[type="tel"] {
        padding: 8px 12px;
        border: 1px solid #d1d5db;
        border-radius: 6px;
        font-size: 14px;
        width: 200px;
        transition: border-color 0.3s ease, box-shadow 0.3s ease;
    }
    input[type="text"]:focus,
    input[type="tel"]:focus {
        outline: none;
        border-color: #2563eb;
        box-shadow: 0 0 0 3px rgba(37, 99, 235, 0.1);
    }
    .save-button,
    .edit-button,
    .cancel-button {
        padding: 8px 16px;
        background: #2563eb;
        color: #fff;
        border: none;
        border-radius: 6px;
        font-size: 14px;
        cursor: pointer;
        transition: transform 0.2s ease, background 0.2s ease;
    }
    .cancel-button {
        background: #ef4444; /* Red for cancel */
    }
    .cancel-button:hover {
        background: #dc2626;
    }

```

```

        .save-button:hover, .edit-button:hover,
        .cancel-button:hover {
            transform: scale(1.05);
        }
        .save-button:active, .edit-button:active, .cancel-
button:active {
            transform: scale(0.95);
        }
        .profile-
pic {
            width: 100px;
            height: 100px;
            border-
radius: 50%;
            object-fit: cover;
            margin-top: 10px;
            border: 2px solid #d1d5db;
        }
        input[type="file"] {
            margin-top: 10px;
            font-size: 14px;
        }
        #successPopup {
            display: none;
            position: fixed;
            top: 50%;
            left: 50%;
            transform:
translate(-50%, -50%);
            background: #10b981;
            color:
white;
            padding: 20px;
            border-radius: 8px;
            box-shadow: 0 4px 20px rgba(0, 0, 0, 0.2);
            z-index: 1000;
            opacity: 0;
            animation: popIn 0.3s ease-out forwards;
        }
        @keyframes popIn {
            from {
                transform: translate(-50%, -60%) scale(0.8);
                opacity: 0;
            }
            to {
                transform: translate(-50%, -50%)
scale(1);
                opacity: 1;
            }
        }
        #successPopup button {
            margin-top: 10px;
            padding: 8px
16px;
            background: #047857;
            border: none;
            color: white;
            border-radius: 6px;
            cursor: pointer;
        }
        #successPopup button:hover {
            background: #065f46;
        }
        .status-msg {
            font-size: 14px;
            margin-
top: 15px;
            opacity: 0;
            transition: opacity 0.3s ease;
        }
        .status-msg.visible {
            opacity: 1;
        }
        @media (max-width: 480px) {
            #settingsModule {
                padding: 20px;
            }
            .dashboard-title {
                font-
size: 20px;
            }
            h2 {
                font-size: 18px;
            }
            input[type="text"],
            input[type="tel"] {
                width: 150px;
            }
            .edit-section {
                flex-direction: column;
                gap: 5px;
            }
            #successPopup {
                width: 80%;
                padding: 15px;
            }
        }
    
```

.vscode\launch.json

```
{
  "configurations": [
    {
      "type": "java",
      "name": "Spring Boot-DemoApplication<demo>",
      "request": "launch",
      "cwd": "${workspaceFolder}",
      "mainClass": "com.example.demo.DemoApplication",
      "projectName": "demo",
      "args": "",
      "envFile": "${workspaceFolder}/.env"
    }
  ]
}
```

.vscode\settings.json

```
{  "liveServer.settings.port": 5501}
```

Table of Contents

[demo\src\main\resources\static\admin_dashboard.js](#)

..... [object Object]

[demo\src\main\resources\static\user_dashboard3.js](#)

..... [object Object]

[demo\target\classes\static\admin_dashboard.js](#)

..... [object Object]

[demo\target\classes\static\user_dashboard3.js](#)

..... [object Object]

[demo\src\main\resources\static\adminSignin.html](#)

..... [object Object]

[demo\src\main\resources\static\adminSignup.html](#)

..... [object Object]

[demo\src\main\resources\static\admin_dashboard.html](#)

..... [object Object]

[demo\src\main\resources\static\admin_dashboard2.html](#)

..... [object Object]

[demo\src\main\resources\static\forgot.html](#)

..... [object Object]

[demo\src\main\resources\static\homepage.html](#)

..... [object Object]

[demo\src\main\resources\static\manageUsers.html](#)

..... [object Object]

[demo\src\main\resources\static\modules\manageUsers.html](#)

..... [object Object]

[demo\src\main\resources\static\signin.html](#)

..... [object Object]

[demo\src\main\resources\static\signin2.html](#)

..... [object Object]

[demo\src\main\resources\static\signin3.html](#)

..... [object Object]

[demo\src\main\resources\static\signin4.html](#)

..... [object Object]

[demo\src\main\resources\static\signup.html](#)

..... [object Object]

[demo\src\main\resources\static\user_dashboard.html](#)

..... [object Object]

[demo\src\main\resources\static\User_dashboard2.html](#)

..... [object Object]

[demo\src\main\resources\static\user_dashboard3.html](#)

..... [object Object]

[demo\src\main\resources\static\user_dashboard4.html](#)

..... [object Object]

[demo\target\classes\static\adminSignin.html](#)

..... [object Object]

[demo\target\classes\static\adminSignup.html](#)

..... [object Object]

[demo\target\classes\static\admin_dashboard.html](#)

..... [object Object]

[demo\target\classes\static\admin_dashboard2.html](#)

..... [object Object]

[demo\target\classes\static\forgot.html](#)

..... [object Object]

[demo\target\classes\static\homepage.html](#)

..... [object Object]

[demo\target\classes\static\manageUsers.html](#)

..... [object Object]

[demo\target\classes\static\modules\manageUsers.html](#)

..... [object Object]

[demo\target\classes\static\signin.html](#)

..... [object Object]

[demo\target\classes\static\signin2.html](#)

..... [object Object]

[demo\target\classes\static\signin3.html](#)

..... [object Object]

[demo\target\classes\static\signin4.html](#)

..... [object Object]

[demo\target\classes\static\signup.html](#)

..... [object Object]

[demo\target\classes\static\user_dashboard.html](#)

..... [object Object]

[demo\target\classes\static\User_dashboard2.html](#)

..... [object Object]

[demo\target\classes\static\user_dashboard3.html](#)

..... [object Object]

[demo\target\classes\static\user_dashboard4.html](#)

..... [object Object]

[demo\src\main\resources\static\admin_dashboard.css](#)

..... [object Object]

[demo\src\main\resources\static\user_dashboard3.css](#)

..... [object Object]

[demo\target\classes\static\admin_dashboard.css](#)

..... [object Object]

[demo\target\classes\static\user_dashboard3.css](#)

..... [object Object]

[.vscode\launch.json](#)

..... [object Object]

[.vscode\settings.json](#)

..... [object Object]