FULL STACK DEVELOPMENT

CCP REPORT

Developed By:

OMAMA SAJID

SALEHA ARSHAD

CS-BLUE

NEXUS CARE

NEXUS CARE

Modules Implemented: Complaint Management + Community Announcements

1. Executive Summary:

NexusCare is a full-stack digital ecosystem designed to enhance community-based services by providing secure, scalable, and responsive web modules. The system integrates modern UI components, RESTful API architecture, and MySQL-based persistent storage to streamline user activities such as complaint reporting and community announcements.

In this project, two major modules were fully implemented:

1. **Complaint & Issue Management System**
2. **Community Announcements Module**

The platform provides authenticated user access, role-based authorization, session management, CRUD functionalities, and a simple, user-friendly interface.

1. Introduction

With the rapid growth of smart communities, there is an increasing need for integrated digital platforms that can manage community services efficiently. NexusCare is designed to address this need by providing a centralized system for residents and administrators to manage complaints, announcements, and other essential services. This project demonstrates the application of full-stack development principles, secure authentication, RESTful APIs, and responsive UI design.

2. Problem Statement & Objectives

Problem Statement

Traditional community management systems often lack integration, scalability, and proper role-based access control. This results in inefficient handling of complaints, poor communication, and security risks.

<u>Objectives</u>

- To design and develop a secure full-stack web application
- To implement role-based authentication and authorization
- To provide a structured complaint handling mechanism
- To enable administrators to publish community-wide announcements
- To ensure scalability, modularity, and maintainability

3. Scope of the Project

The scope of NexusCare includes:

- User authentication and role-based access
- Complaint & Issue Management module (CRUD)
- Community Announcements module (CRUD)
- RESTful API development
- MySQL database integration
- Responsive UI using React

4. System Overview

NexusCare follows a client-server architecture:

- **Frontend:** React Single Page Application (SPA)
- **Backend:** Flask-based RESTful APIs
- **Database:** MySQL
- **Security:** Session-based authentication and role-based authorization

User Roles:

- Admin
- Resident
- Security Staff
- Medical Assistant

5. Technology Stack & Tools

Frontend

- React.js
- HTML5, CSS3, JavaScript
- Axios
- React Router

Backend
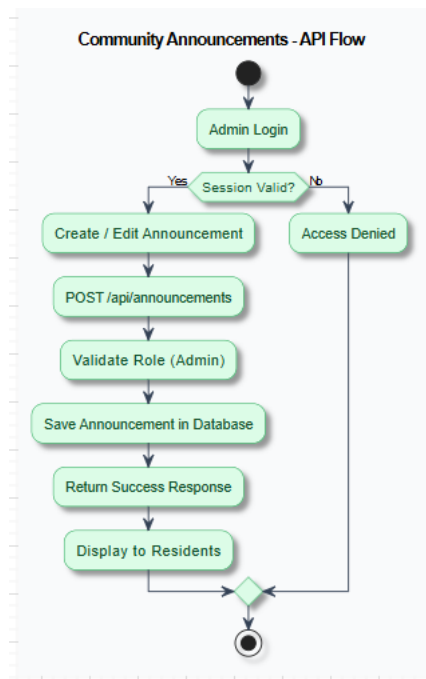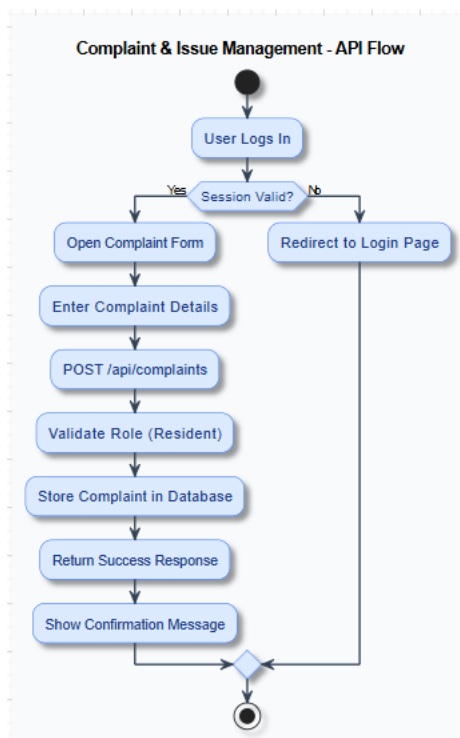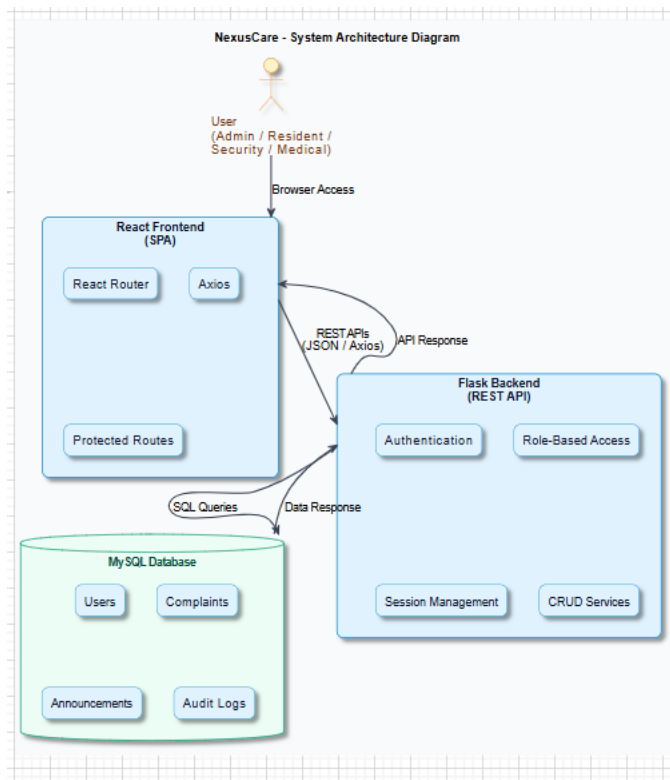
- Python (Flask)
- RESTful API design
- Session management

Database

- MySQL

6. System Architecture & Design

This system uses a layered architecture:

- Presentation Layer (React UI)
- Application Layer (Flask APIs)
- Data Layer (MySQL)

NexusCare - System Architecture Diagram

User
(Admin / Resident /
Security / Medical)

Browser Access

**React Frontend
(SPA)**

React Router   Axios

Protected Routes

REST APIs
(JSON / Axios)   API Response

**Flask Backend
(REST API)**

Authentication   Role-Based Access

Session Management   CRUD Services

SQL Queries   Data Response

MySQL Database

Users   Complaints

Announcements   Audit Logs

Complaint & Issue Management - API Flow

User Logs In

Yes → Session Valid? ← No

Open Complaint Form   Redirect to Login Page

Enter Complaint Details

POST /api/complaints

Validate Role (Resident)

Store Complaint in Database

Return Success Response

Show Confirmation Message

Community Announcements - API Flow

Admin Login

Yes → Session Valid? ← No

Create / Edit Announcement   Access Denied

POST /api/announcements

Validate Role (Admin)

Save Announcement in Database

Return Success Response

Display to Residents

---

7. Database Design

- Users
- Complaints
- Announcements
- Audit Logs

		id	title	message	created_by	created_at	updated_at	is_deleted
☐	🖉 Edit ⬚ Copy ⊝ Delete	5	Early timings	You can sign off early at 9:30pm	3	2025-12-07 18:23:29	2025-12-09 13:25:21	0
☐	🖉 Edit ⬚ Copy ⊝ Delete	6	Collection of Reports	Report all medical reports from office 1123	3	2025-12-09 13:26:09	*NULL*	0

↰　☐ Check all　　*With selected:*　🖉 Edit　⬚ Copy　⊝ Delete　🖳 Export

8. Module Implementation Details

<u>8.1 Complaint & Issue Management System</u>

Description: This module allows residents to report issues, track complaint status, and receive updates. Admins can view, update, and resolve complaints.

Features:

- Create complaint

Complaint Center

Resources Access error

There is some issue with the resource list on the management system pls check it

Submit Complaint

Title	Description	Status	Actions
Irresponsible beahviour	not behaving well	Open	Edit

Complaint submitted successfully!

Complaint Title

Complaint Description

Submit Complaint

Title	Description	Status	Actions
Irresponsible beahviour	not behaving well	Open	Edit
Resources Access error	There is some issue with the resource list on the management system pls check it	Open	Edit

- View complaints

All Complaints

ID	User ID	Title	Description	Status	Created At	Actions
12	4	Irresponsible beahviour	not behaving well	Open	09/12/2025, 6:22:11 pm	Delete
15	4	Resources Access error	There is some issue with the resource list on the management system pls check it	Open	18/12/2025, 11:23:10 pm	Delete

- Update status

- Soft delete

- Audit logging

Complaint Audit Logs

ID	User ID	Action	User	Resource	Resource ID	Old Value	New Value	Timestamp
100	3	DELETE	omama	Complaint	12	{"description":"not behaving well","title":"Irresponsible beahviour"}	No new value	18/12/2025, 11:25:02 pm
99	3	LOGIN	omama	User	Not Applicable	No previous value	No new value	18/12/2025, 11:23:39 pm
98	4	LOGOUT	Saleha	User	Not Applicable	No previous value	No new value	18/12/2025, 11:23:30 pm
97	4	CREATE	Saleha	Complaint	15	No previous value	{"description":"There is some issue with the resource list on the management system pls check it","title":"Resources Access error"}	18/12/2025, 11:23:10 pm

8.2 Community Announcements Module

Description: This module enables administrators to publish announcements that are visible to all residents.
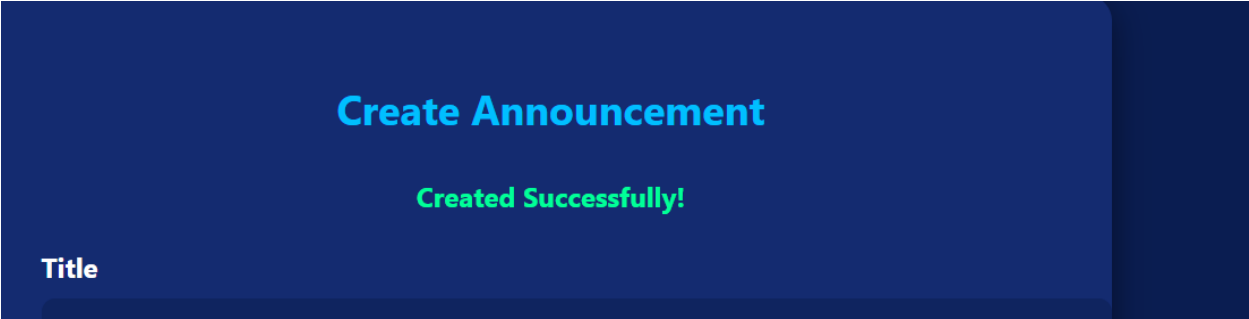
Features:

- Create announcement (by admin):

Create Announcement

Title

Early Sign-OFF

Message

All members can leave early today
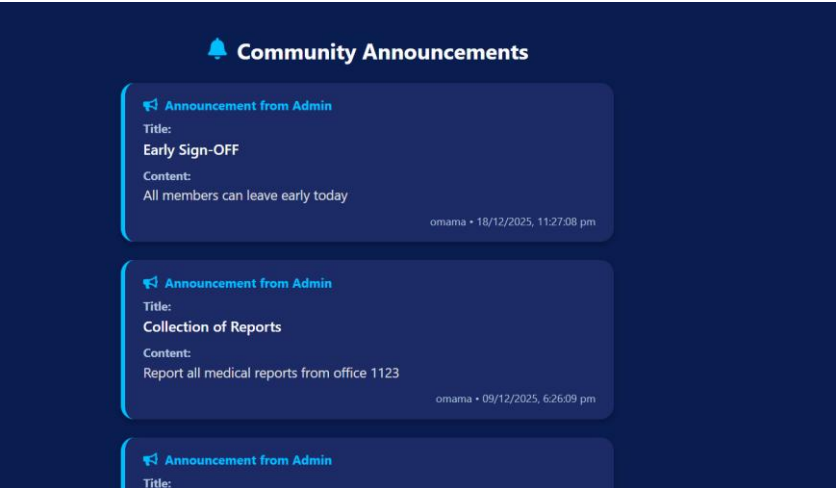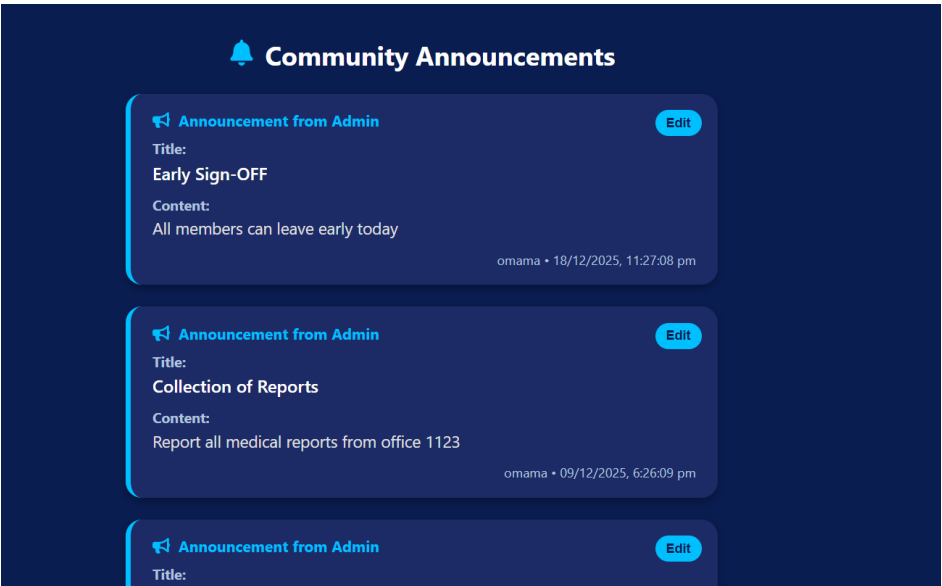
Create

- View announcements

(User View):

(Admin) view

- Edit announcements

Edit Announcement

Title

Collection of Reports

Message

Report all medical reports from office 1123

Save Changes

Edit Announcement

Title

Collection of Reports

Message

Report all medical reports from Ground floor 123

Save Changes

localhost:3000 says

Updated successfully!

OK

Edit Announcement

Title

Collection of Reports

Message

Report all medical reports from Ground floor 123

📢 Announcement from Admin   Edit
Title:
Collection of Reports

Content:
Report all medical reports from Ground floor 123

omama • 09/12/2025, 6:26:09 pm
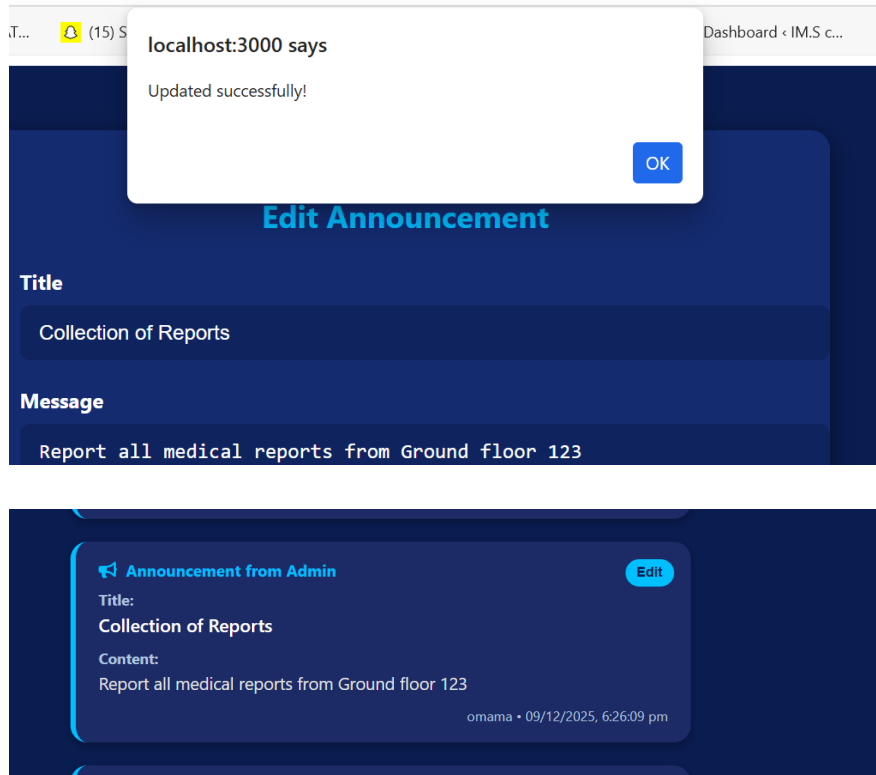
9. Security & Session Management

- Password hashing

```
if user and bcrypt.checkpw(password.encode('utf-8'), user[1].encode('utf-8')):
    session['user_id'], session['role'] = user[0], user[2]
    session.permanent = True
    insert_audit_log(user[0], 'LOGIN', 'User', resource_id=None)
    current_app.logger.info(f"User logged in: {email} with role {user[2]}")
    return jsonify({'message': 'Login successful', 'role': user[2]})

current_app.logger.warning(f"Failed login attempt for email: {email}")
return jsonify({'error': 'Invalid credentials'}), 401
```

password_hash	role
$2b$12$ljdAeXUQ10j6GG5tkGXn/Owy3hii69btbKbeV4te6fV...	Resident
$2b$12$E5tPejoaQ7mUdzZQCwzS3OJjHmWUvB8MDTUTiCg9vMX...	Admin
$2b$12$5ozgFXVv3Exgg9VBATuwLO0Wju4leCcNsaO98xI6j0F...	Medical

- Session-based authentication

```python
validate_session():
if 'user_id' in session:
    current_app.logger.info(f"Session validated for user: {session['user_id']}")
    return jsonify({'logged_in': True, 'user_id': session['user_id'], 'role': session['role']})

current_app.logger.warning("Unauthorized session validation attempt")
return jsonify({'logged_in': False}), 401
```

```javascript
const checkSession = async () => {
  try {
    const res = await axios.get("http://localhost:5000/validate-session", {
      withCredentials: true,
    });

    if (res.status === 200 && res.data.logged_in) {
      const sessionUser = { email: res.data.user_id, role: res.data.role };
      setUser(sessionUser);
      localStorage.setItem("user", JSON.stringify(sessionUser));
    } else {
      logout();
    }
  } catch (err) {
    console.error("Session check failed:", err);
    logout();
  } finally {
    setLoading(false);
  }
};
```


- Role-based authorization

```jsx
<Route
  path="/complaints"
  element={
    <ProtectedRoute allowedRoles={['Resident', 'Medical', 'Security']}>
      <ComplaintPage />
    </ProtectedRoute>
  }
/>
<Route
  path="/complaints/admin"
  element={
    <ProtectedRoute allowedRoles={['Admin']}>
      <AdminComplaints />
    </ProtectedRoute>
  }
/>
<Route
  path="/complaints/audit"
  element={
    <ProtectedRoute allowedRoles={['Admin']}>
      <AuditLogList resourceType="Complaint" />
    </ProtectedRoute>
  }
/>
```

localhost:3000/dashboard

operating-system-l...   SCHOLARSHIPS.AT...   (15) S   Dashboard ‹ IM.S c...

localhost:3000 says

Access Denied: Only admins can download logs

OK

Welcome to NexusCare

Your Role: Medical

Server Logs
Download latest server log file
CRITICAL

Download
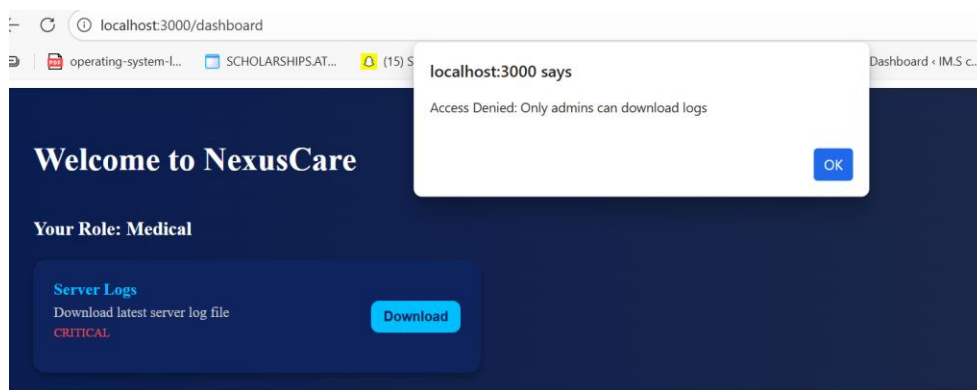
- Protected routes

```jsx
function ProtectedRoute({ children, allowedRoles }) {
  const { user, loading } = useAuth();

  if (loading) {
    return <div>Checking session...</div>;
  }

  if (!user) {
    return <Navigate to="/login" replace />;
  }
  if (allowedRoles && !allowedRoles.includes(user.role)) {
    return <Navigate to="/unauthorized" replace />;
  }
  return children;
}

export default ProtectedRoute;
```

Forbidden Route:
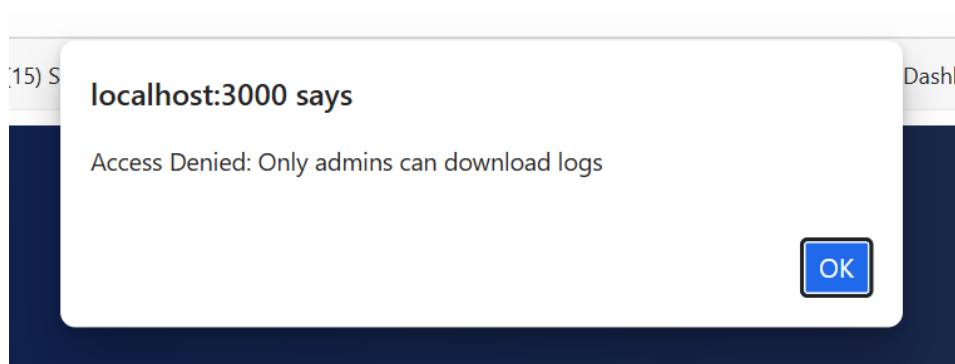
Allowed Routes:
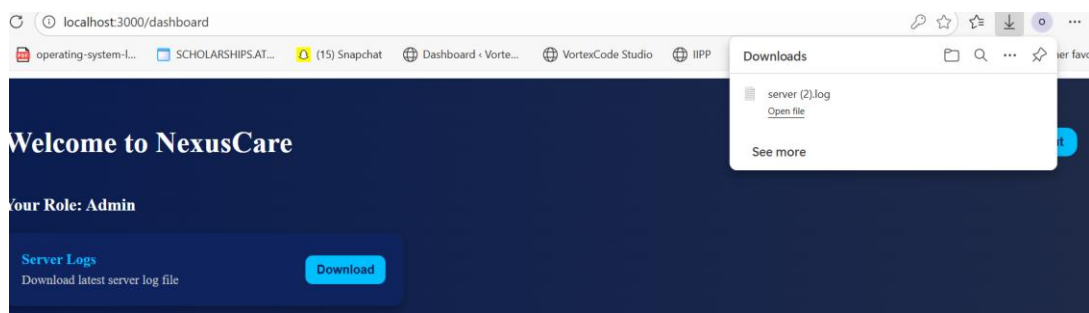
- Unauthorized access handling
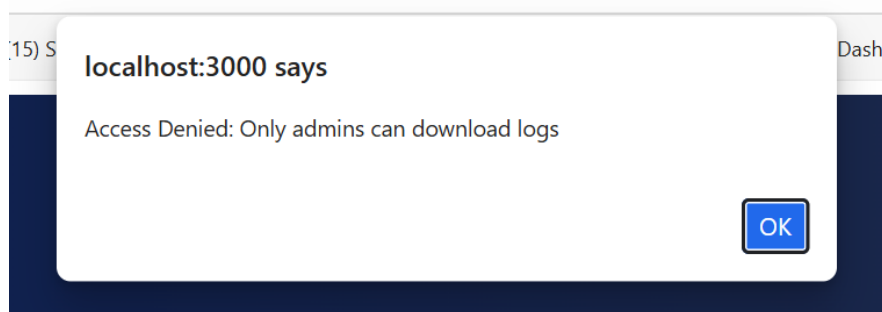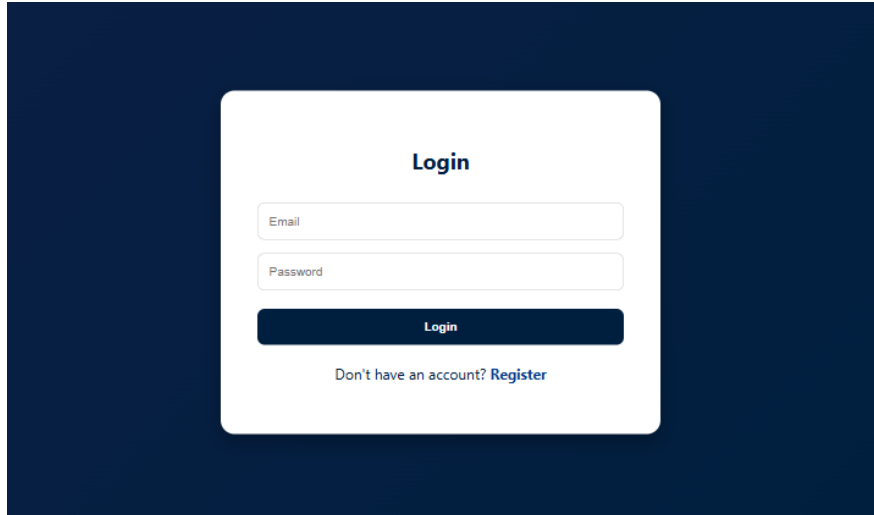

```
const downloadLogs = async () => {
  if (role !== "Admin") {
    alert("Access Denied: Only admins can download logs");
    return;
  }
  setLogDownloading(true);
  try {
    const res = await axios.get("http://localhost:5000/admin/download-logs", {
      responseType: "blob",
      withCredentials: true,
    });
    const url = window.URL.createObjectURL(new Blob([res.data]));
    const link = document.createElement("a");
```
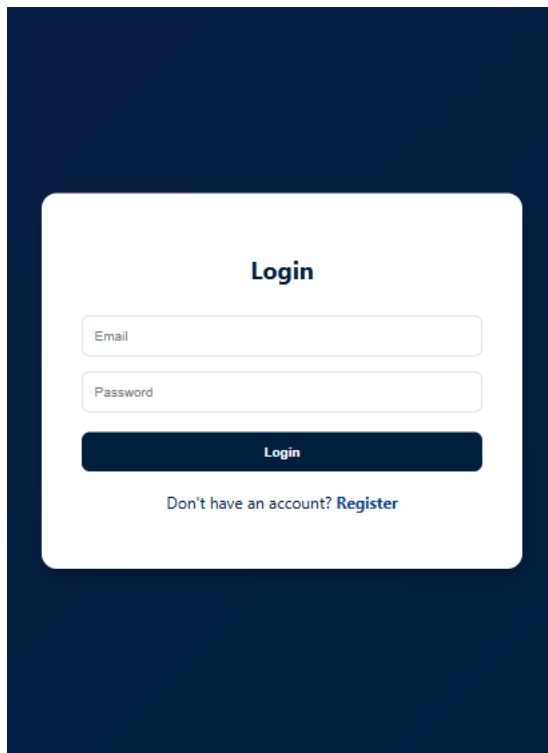
10. User Interface & UX Design

- <u>Responsive design</u>

Welcome to NexusCare

Your Role: Medical

Server Logs
Download latest server log file
CRITICAL
Download

📄 **Register Complaint**

Click here to create a new complaint.

📢 **Announcements**

Click here to view community announcements.

⚙ **Profile & Settings**

Update your profile or password.

Monthly Complaints Trend

Complaints Trend

Complaint Center

Complaint Title

Complaint Description

Submit Complaint

Title	Description	Status	Actions
Resources Access error	There is some issue with the resource list on the management system pls check it	Open	Edit

- <u>Role-based navigation:</u>

Register

User

user@gmail.com

••••••

Resident ⌄

Resident
Admin
Medical Assistant
Security Staff

Register

Welcome to NexusCare

Your Role: Medical

Server Logs
Download latest server log file
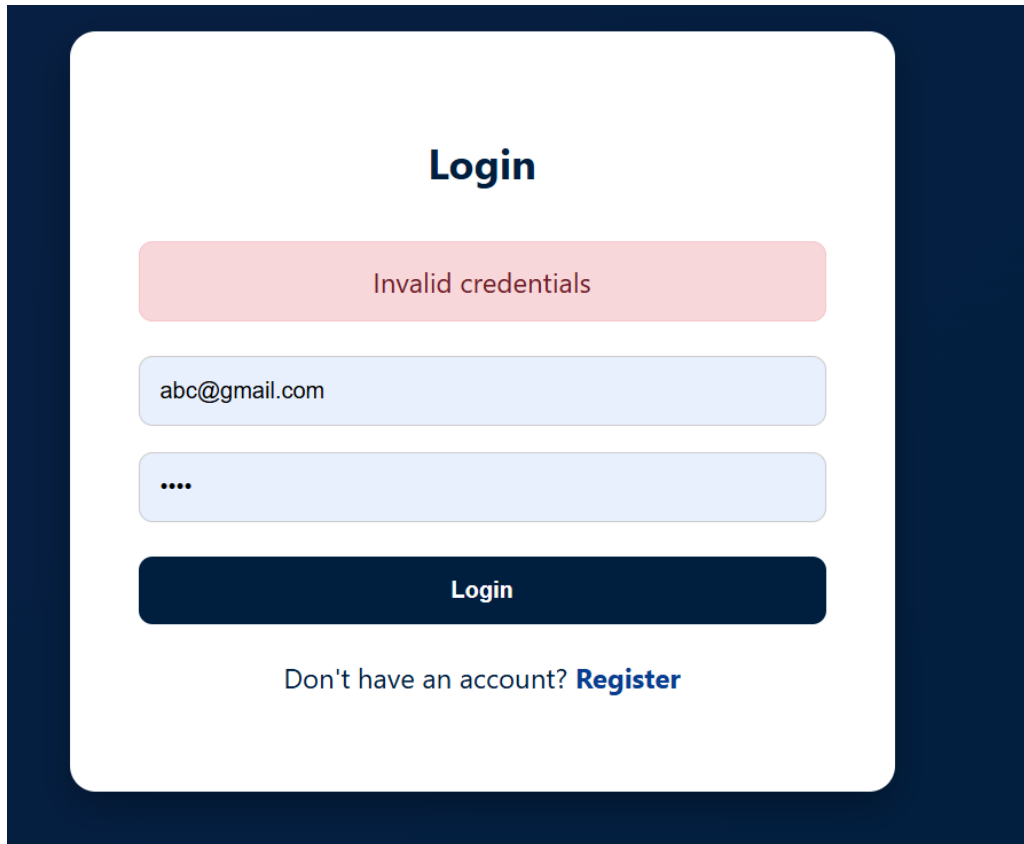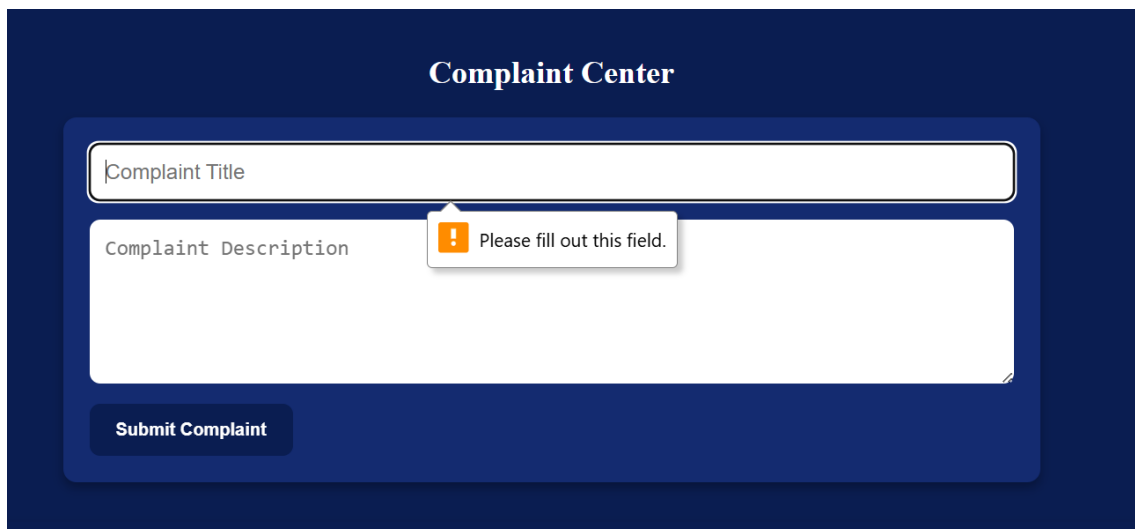CRITICAL

Download

📄 **Register Complaint**

Click here to create a new complaint.

📣 **Announcements**

Click here to view community announcements.

- Error messages & loaders

11. Testing & Validation

Functional Testing:

- Login
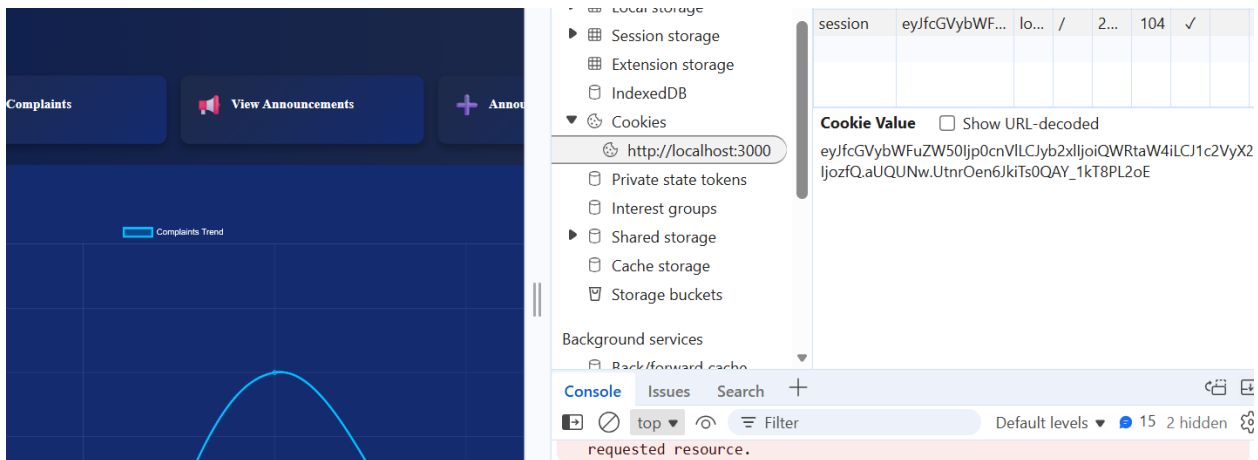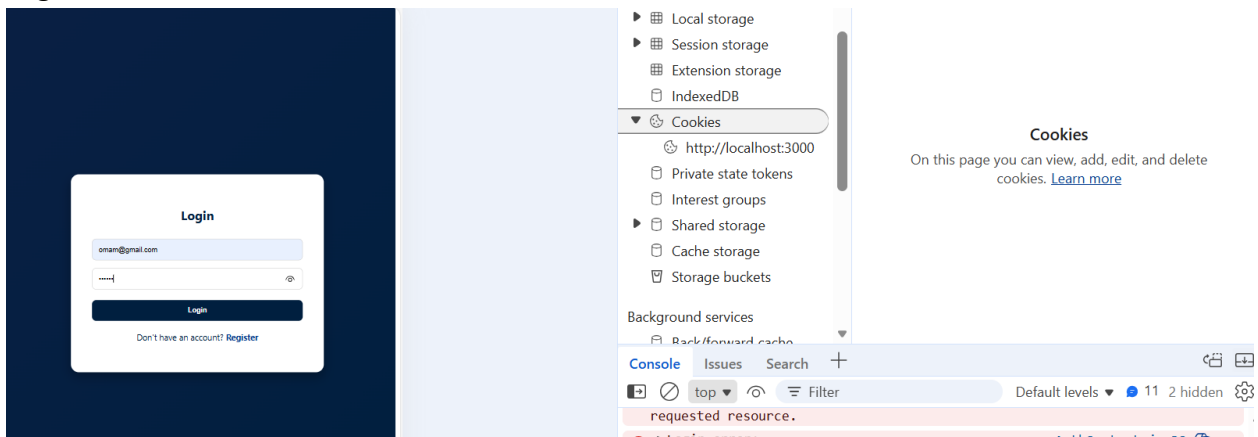

```python
@auth_bp.route('/login', methods=['POST'])
def login():
    data = request.get_json()
    email, password = data.get('email'), data.get('password')
    cursor = mysql.connection.cursor()
    cursor.execute("SELECT id,password_hash,role FROM users WHERE email=%s", (email,))
    user = cursor.fetchone()

    if user and bcrypt.checkpw(password.encode('utf-8'), user[1].encode('utf-8')):
        session['user_id'], session['role'] = user[0], user[2]
        session.permanent = True
        insert_audit_log(user[0], 'LOGIN', 'User', resource_id=None)
        current_app.logger.info(f"User logged in: {email} with role {user[2]}")
        return jsonify({'message': 'Login successful', 'role': user[2]})

    current_app.logger.warning(f"Failed login attempt for email: {email}")
    return jsonify({'error': 'Invalid credentials'}), 401
```
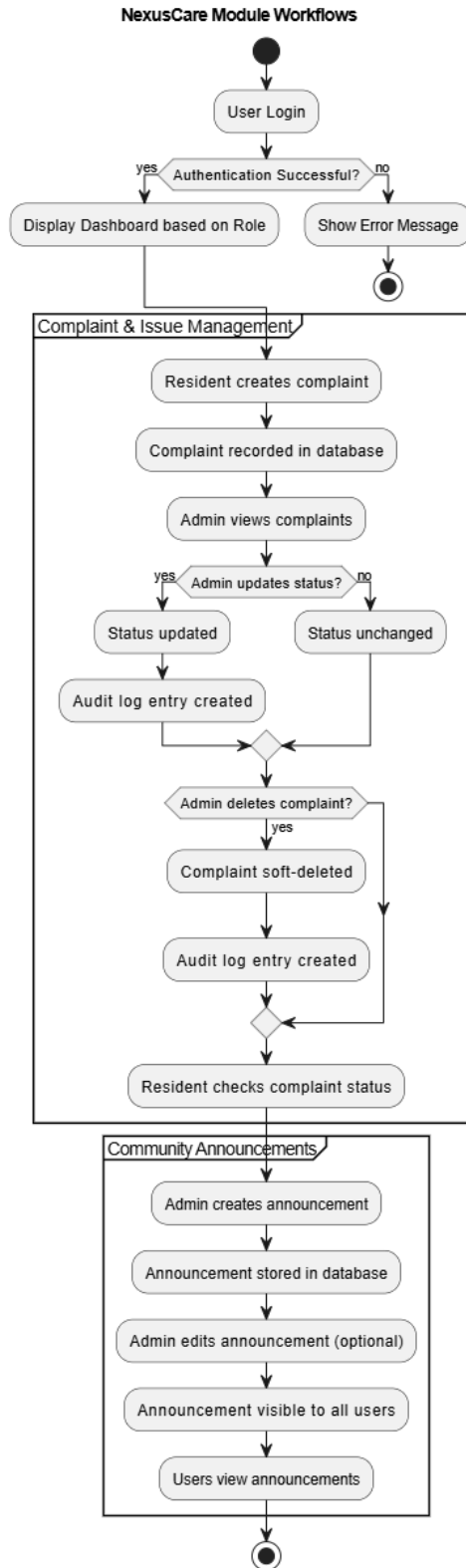
- Logout
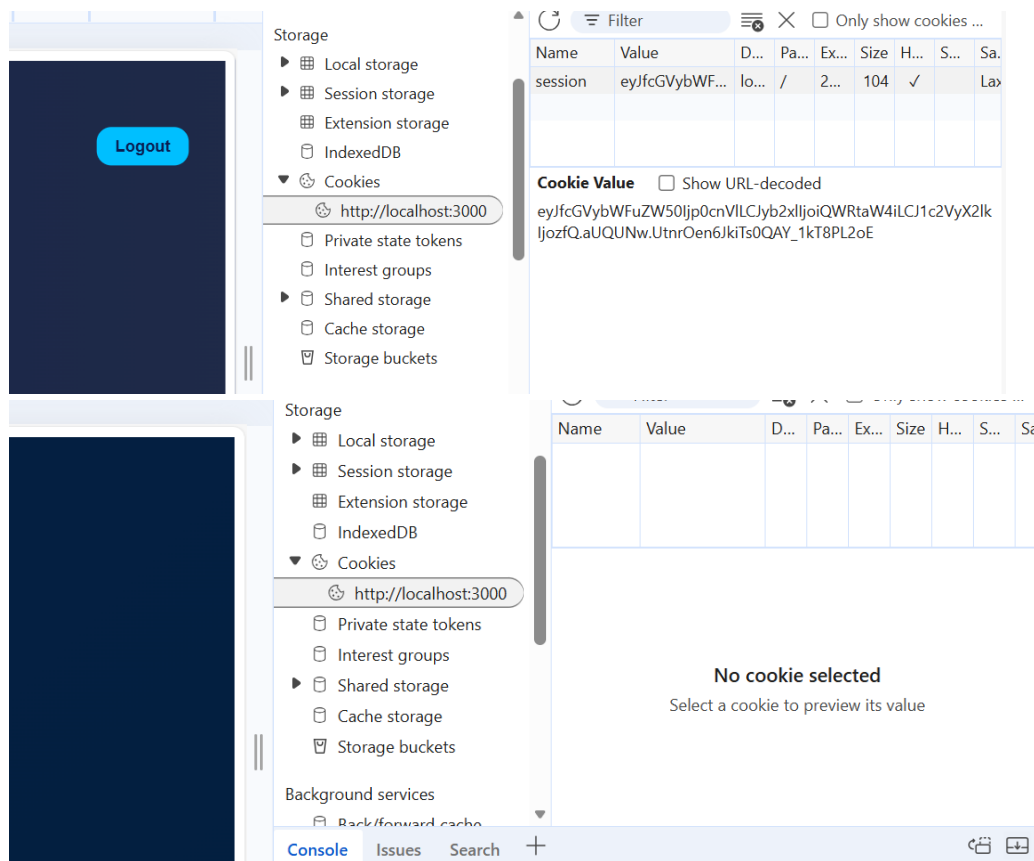

```python
@auth_bp.route('/logout', methods=['POST'])
def logout():
    user_id = session.get('user_id')
    if user_id:
        insert_audit_log(user_id, 'LOGOUT', 'User', resource_id=None)
        current_app.logger.info(f"User logged out: {user_id}")
    session.clear()
    return jsonify({'message': 'Logged out successfully'})
```
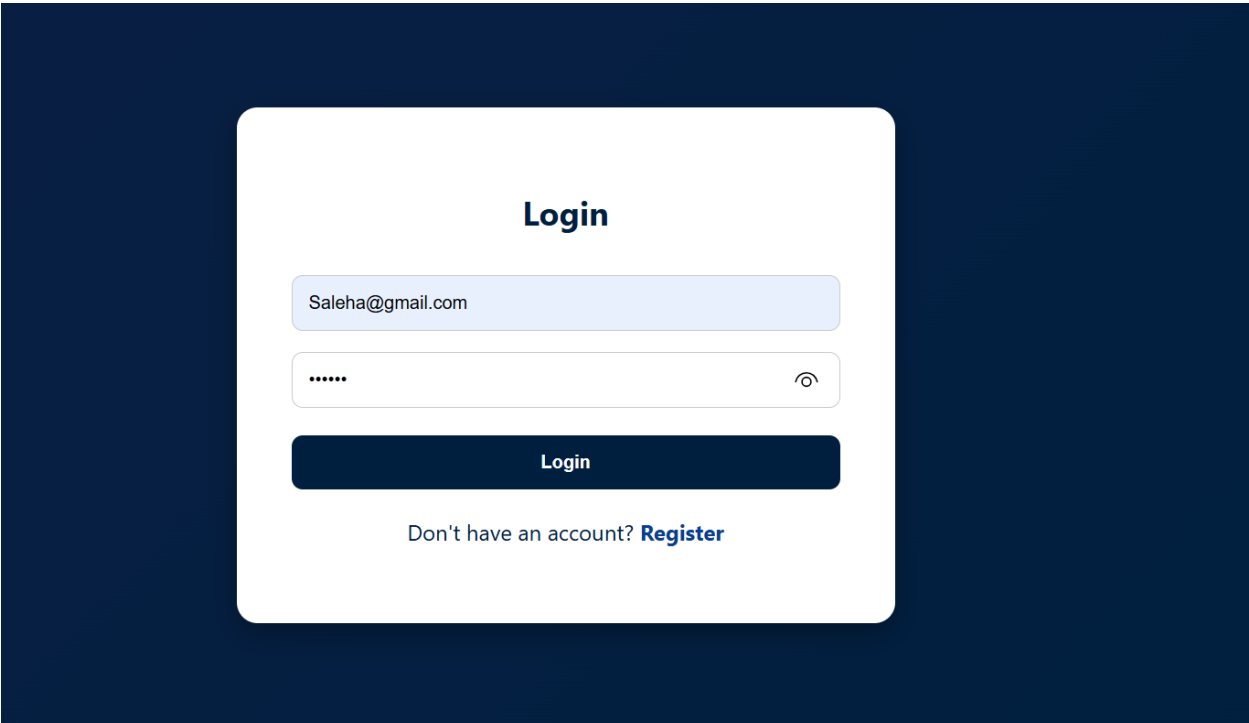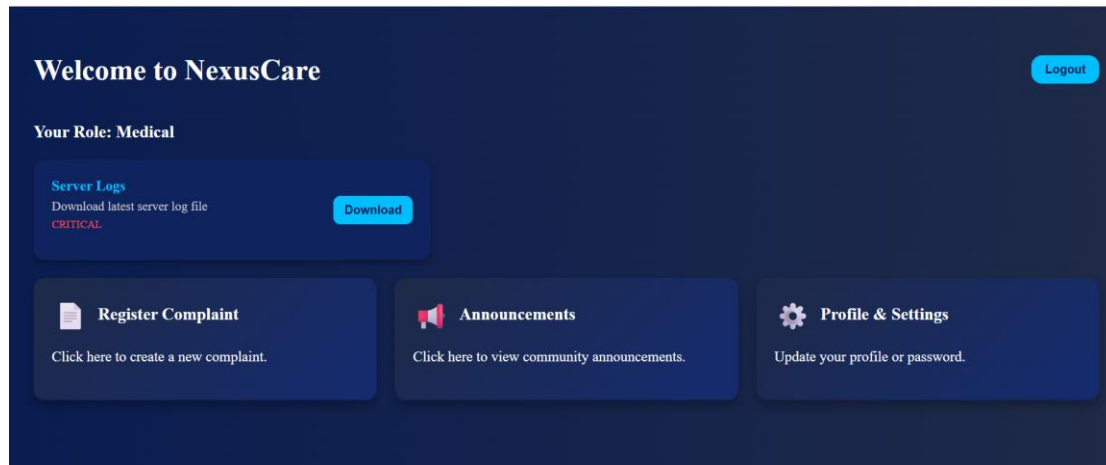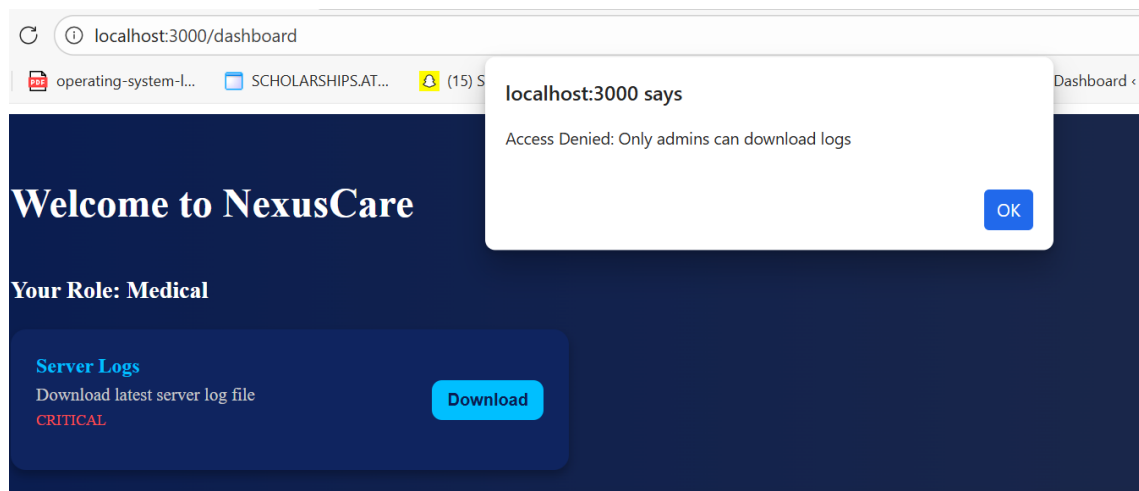
- Module workflows

NexusCare Module Workflows

Visual Walkthrough of NexusCare:

MEDICAL VIEW (NON-ADMIN VIEW):

Accessing server logs:

Complaint Creation:

Complaint View:

Announcement View:

ADMIN VIEW:

Welcome to NexusCare

Logout

Your Role: Admin

Server Logs
Download latest server log file
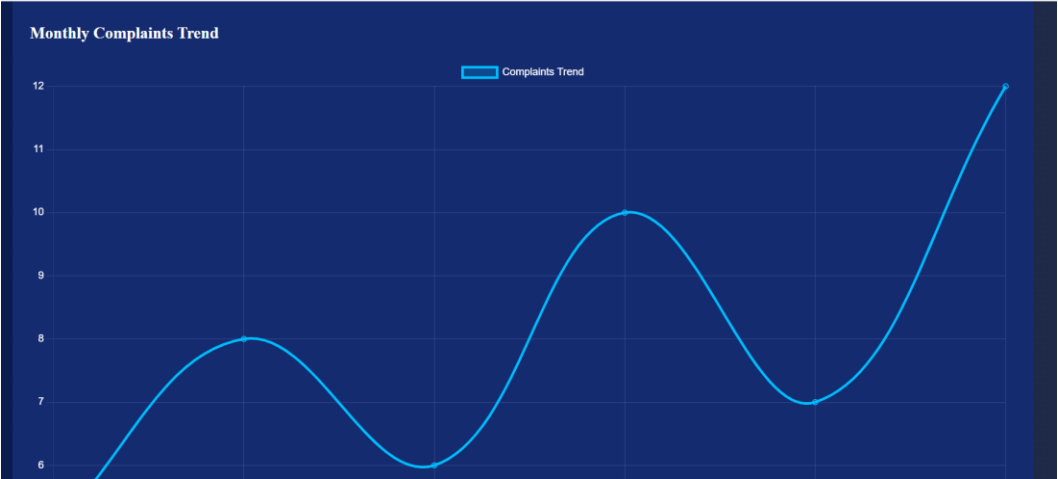
Download

📒 **View Audit Logs**

📄 **View Complaints**

📢 **View Announcements**

➕ **Announce to team**

Monthly Complaints Trend

Complaints Trend

12

AUDIT LOGS:

ID	User ID	Action	User	Resource	Resource ID	Old Value	New Value	Timestamp
137	3	LOGIN	omama	User	Not Applicable	No previous value	No new value	19/12/2025, 1:00:22 am
136	4	LOGOUT	Saleha	User	Not Applicable	No previous value	No new value	19/12/2025, 12:59:44 am
135	4	CREATE	Saleha	Complaint	17	No previous value	{"description":"Client does not cooperate with patients","title":"Misbehave"}	19/12/2025, 12:58:55 am
134	4	LOGIN	Saleha	User	Not Applicable	No previous value	No new value	19/12/2025, 12:57:12 am
133	3	LOGOUT	omama	User	Not Applicable	No previous value	No new value	19/12/2025, 12:49:42 am
132	3	LOGIN	omama	User	Not Applicable	No previous value	No new value	19/12/2025, 12:48:21 am
131	3	LOGOUT	User		Not			19/12/2025,

Accessing Server Logs:

Viewing & processing Complaints:

All Complaints

ID	User ID	Title	Description	Status	Created At	Actions
17	4	Misbehave	Client does not cooperate with patients	Open	19/12/2025, 12:58:55 am	Delete
15	4	Resources Access error	There is some issue with the resource list on the management system pls check it	Open	18/12/2025, 11:23:10 pm	Delete

All Complaints

ID	User ID	Title	Description	Status	Created At	Actions
17	4	Misbehave	Client does not cooperate with patients	Open	19/12/2025, 12:58:55 am	Delete
15	4	Resources Access error	There is some issue with the resource list on the management system pls check it	Open / In Progress / Resolved	18/12/2025, 11:23:10 pm	Delete

Annoucement View & Update:

🔔 Community Announcements

📣 **Announcement from Admin**   Edit

Title:
Early Sign-OFF

Content:
All members can leave early today

omama • 18/12/2025, 11:27:08 pm

📣 **Announcement from Admin**   Edit

Title:
Collection of Reports

Content:
Report all medical reports from Ground floor 123

omama • 09/12/2025, 6:26:09 pm
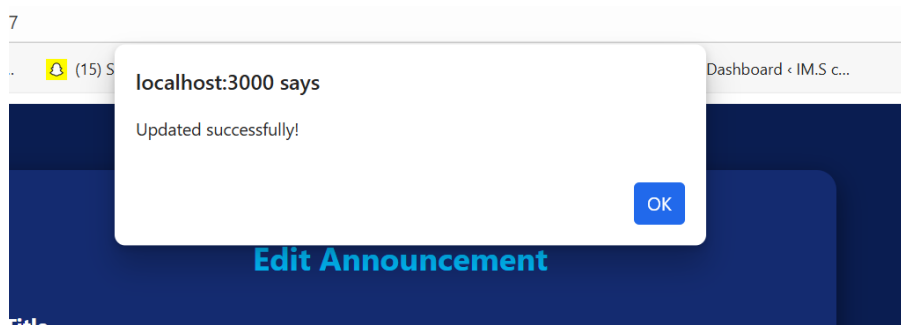
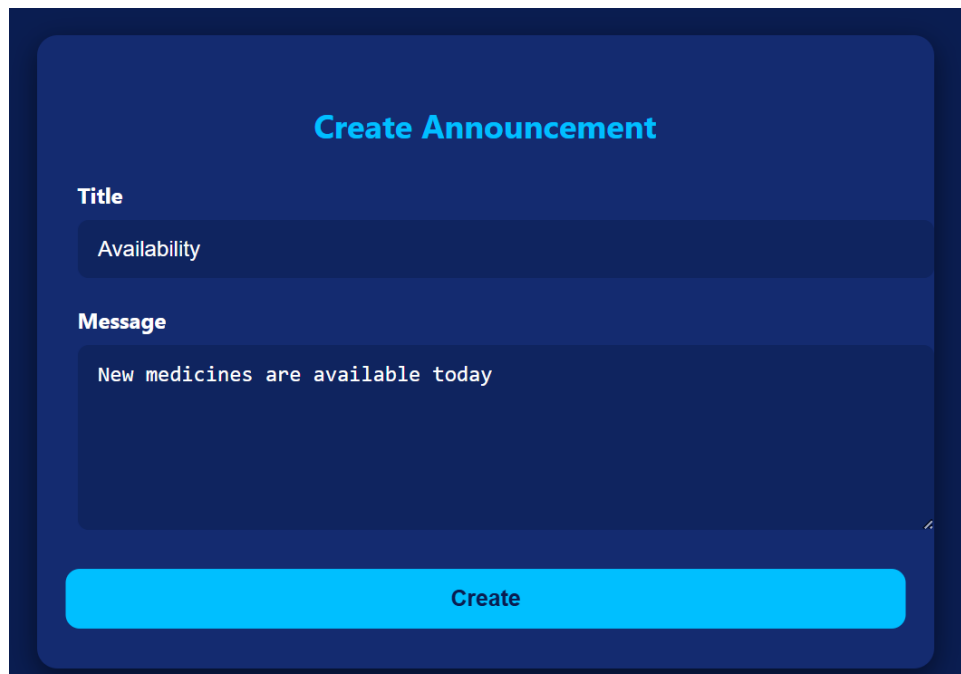📣 **Announcement from Admin**   Edit

Title:

Edit Announcement

Title

Early Sign-OFF

Message

All members can leave early today but make sure to mark attendance

Save Changes

7

🔔 (15) S   Dashboard ‹ IM.S c...

localhost:3000 says

Updated successfully!

OK

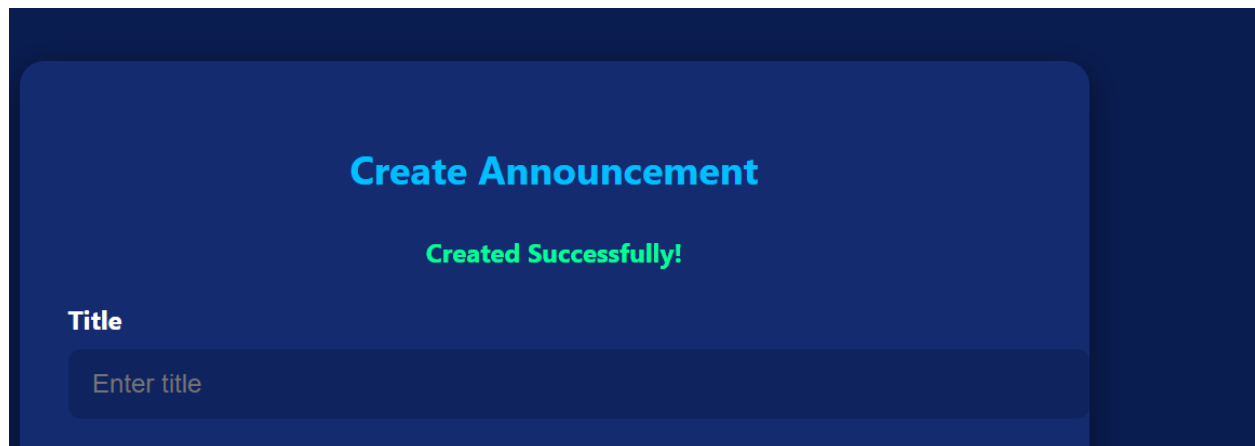## Edit Announcement

Announcement Creation:

Create Announcement

Title

Availability

Message

New medicines are available today

Create

Conclusion

NexusCare is a complete and efficient system for managing community services. It successfully handles complaints and announcements, providing residents and administrators with an easy-to-use and secure platform. The system ensures that complaints are tracked and resolved properly, and that important community announcements reach everyone quickly. With role-based access and session management, NexusCare maintains security while allowing different users to perform their tasks smoothly. Overall, the project demonstrates the practical use of modern web technologies to improve community management, making the process faster, more organized, and more reliable.