

# Inclinic System - Issues Report (Updated)

## Issue 1: Password + Security Question (Forgot Password)

- Code file name: LoginCallback.jsx
- Code file path: src/pages/LoginCallback.jsx
- What does the code do?: Handles Google OAuth login callback and sets user role in localStorage.
- Entire code: LoginCallback.jsx code
  - import React, { useEffect } from 'react';
  - import { useSearchParams, useNavigate } from 'react-router-dom';
  - 
  - function LoginCallback() {
  - const [searchParams] = useSearchParams();
  - const navigate = useNavigate();
  - 
  - useEffect(() => {
  - // Suppose the Django backend redirects here with ?userRole=FIELD\_REP or brand manager or admin
  - const userRole = searchParams.get('userRole');
  - if (userRole) {
  - localStorage.setItem('userRole', userRole);
  - }
  - navigate('/');
  - }, [searchParams, navigate]);
  - 
  - return (
  - <div style={{ padding: '20px' }}>
  - <h2>Processing login...</h2>
  - </div>
  - );
  - }
  - 
  - export default LoginCallback;

## **Inclinic System - Issues Report (Updated)**

- Error: No security question or recovery flow implemented.
- Description of Error: Users have no way to recover account using security questions.

### **Issue 2: Campaign Creation**

- Code file name: CampaignForm.jsx
- Code file path: inclinic-frontend/src/pages/Admin/Campaigns/CampaignForm.jsx
- What does the code do?: Submits a new campaign using brand name and date range to the backend.
- Entire code: Not implemented yet.
- Error: No extraction or storage of the returned Campaign ID after submission.
- Description of Error: Backend submission happens, but the frontend fails to capture the returned campaign ID.

### **Issue 3: CSV Upload (Field Rep Registration)**

- Code file name: upload\_reps.py (or inside views.py)
- Code file path: inclinic-backend/inclinic/views/upload\_reps.py
- What does the code do?: Handles CSV uploads for field reps and stores them in the database.
- Entire code: Not implemented yet.
- Error: No validation based on the type of CSV template.
- Description of Error: Accepts any CSV structure without validating column headers or expected format.

### **Issue 4: Collateral Content ID Generation (React Frontend)**

- Code file name: GenerateContentId.jsx
- Code file path: src/pages/Admin/Content/GenerateContentId.jsx
- What does the code do?: Collects content type (PDF or video) and sends it to the backend to generate a content ID.
- Entire code: Not implemented yet.
- Error: Component does not exist.
- Description of Error: This functionality is referenced but not implemented in the codebase.

### **Issue 5: WhatsApp Collateral Sharing (React Frontend)**

- Code file name: SendCollateral.jsx

## **Inclinic System - Issues Report (Updated)**

- Code file path: Not clearly defined, likely under a sharing or content directory.
- What does the code do?: Initiates content sharing with doctor via WhatsApp.
- Entire code: Not implemented yet.
- Error: Backend route not implemented.
- Description of Error: Frontend calls WhatsApp API route, but corresponding Django endpoint is missing.

### **Issue 6: Doctor Side Verification and Collateral View (React Frontend)**

- Code file names: DoctorLogin.jsx, DoctorCollaterals.jsx
- Code file path: src/pages/Doctor/DoctorLogin.jsx, src/pages/Doctor/DoctorCollaterals.jsx
- What does the code do?: Allows doctors to verify themselves and view/download collaterals.
- Entire code: Not implemented yet.
- Error: Components do not exist.
- Description of Error: Doctor verification and content access UI is completely missing.

### **Issue 7: Dashboard Data Integration**

- Code file names: AdminDashboard.jsx, BrandManagerDashboard.jsx, FieldRepDashboard.jsx, Navbar.jsx, LoginCallback.jsx
- Code file path: src/components/, src/pages/
- What does the code do?: Provides role-based dashboards.
- Error: Dashboards lack dynamic integration with backend.
- Description of Error: Dashboards have static content or fail to fetch role-based data.

### **Issue 8: Superuser Setup & reCAPTCHA Issue**

- Code file names: N/A (Command-line for superuser setup), forms.py
- Code file path: inclinic-backend/manage.py (run command), inclinic-backend/mainapp/forms.py
- What does the code do?: Superuser - Creates admin user. reCAPTCHA - Validates user actions via Google

## **Inclinic System - Issues Report (Updated)**

reCAPTCHA v3.

- Error: Superuser not yet created. reCAPTCHA not functioning correctly.
- Description of Error: Admin access unavailable until a superuser is manually created. Google reCAPTCHA fails due to domain mismatch.