

JusticeFlow - Feature-by-Feature Documentation

Detailed Feature Breakdown for Documentation

FEATURE 1: ADMIN DASHBOARD - SYSTEM CONFIGURATION

Purpose

Centralized control panel for managing all system entities (police stations, hospitals, courts) and user accounts.

Sub-Features

1.1 Police Station Management

What it does: Create, edit, and delete police station records

Fields:

- Station ID (auto-generated, e.g., POL-1234)
- Station Name
- Location
- Contact Number (10 digits, validated)

How it works:

1. Click "Add New Station" button
2. Fill in form fields
3. System validates phone number format
4. Generates unique ID automatically
5. Saves to localStorage (`justice_stations`)
6. Data persists across sessions

Edit Function:

- Click edit icon on any row
- Pre-fills form with existing data
- Update and save
- Changes reflect immediately

Delete Function:

- Click trash icon
- Confirmation dialog appears
- Permanent deletion from database

Technical Detail:

```
typescript
// Auto-save mechanism
useEffect(() => {
  localStorage.setItem('justice_stations', JSON.stringify(stations));
}, [stations]);

// ID generation
const generateID = (prefix: string) => `${prefix}-${Math.floor(1000 + Math.random() * 9000)}`;
```

1.2 Hospital Management

Identical functionality to Police Stations but for medical facilities

Fields:

- Hospital ID (HOS-XXXX)
- Hospital Name
- Location
- Contact Number

Why it's needed: JMO dashboard filters patients based on hospital name, so accurate hospital records are essential.

1.3 Court Management

Purpose: Maintain registry of courts for case assignment

Fields:

- Court ID (CRT-XXXX)
- Court Name
- Location
- Contact Number

Integration:

- Police officers select from this list when registering cases
 - Judges can only access cases assigned to their court
-

1.4 User Account Management

Purpose: Create login accounts for all system users

Supported Roles:

- Police Officer
- JMO (Medical Officer)
- Attorney
- Judge / Magistrate
- Admin

Fields:

- User ID (auto-generated, USR-XXXX)
- Full Name
- Email (becomes username)
- NIC Number (12 digits)
- Role (dropdown)
- Appointed Place (dynamic based on role)
- Contact Number (10 digits)
- Password (plain text for demo; production would hash)

Dynamic Appointed Place:

- **Police** → Select from police stations
- **JMO** → Select from hospitals
- **Judge** → Select from courts
- **Attorney** → Auto-set to "Bar Association / Private"

Password Handling:

- For new users: Required

- For editing: Leave empty to keep current password
- Updates saved to `justiceflow_users_final` in localStorage

Security Feature:

```
typescript
// When editing, preserve old password if field is empty
if (editingId) {
  const existingUser = users.find(u => u.id === editingId);
  if (!formData.password && existingUser) {
    finalPassword = existingUser.password; // Keep old password
  }
}
```

User Creation Alert: After creating a user, system shows popup with:

- Email address
 - Password So admin can share credentials with the user.
-

FEATURE 2: POLICE DASHBOARD - INCIDENT MANAGEMENT

Purpose

Allow police officers to register criminal incidents, manage case evidence, and track case progress.

Sub-Features

2.1 Station-Based Case Filtering

What it does: Officers only see cases from their assigned station

How it works:

- User's `appointedPlace` is read from login session
- Cases filtered by `station` field
- Example: OIC Perera (Mount Lavinia HQ) cannot see Dehiwala Station cases

Code:

```
typescript
```

```
const assignedStation = currentUser?.appointedPlace || 'General Headquarters';

const filteredCases = cases.filter(c => {
  const matchStation = c.station === assignedStation;
  return matchStation && matchSearch && matchDate && matchCourt;
});
```

2.2 New Incident Registration

Form Sections:

A. Basic Information

- Incident Date (cannot be future date)
- Venue / Location
- Summary / Description
- Reporting As (Victim or Witness)

B. Court Assignment ★ CRITICAL FIELD

- Dropdown populated from admin-created courts
- Required field
- Determines which judge can access the case

C. Evidence Upload

- Multi-file upload support
- Accepts images, PDFs, documents
- Converts files to Base64 for storage
- Visual preview grid
- Remove files before submission

Technical:

```
typescript
```

```

const fileToBase64 = (file: File): Promise<{name, type, data}> => {
  return new Promise((resolve, reject) => {
    const reader = new FileReader();
    reader.readAsDataURL(file);
    reader.onload = () => resolve({
      name: file.name,
      type: file.type,
      data: reader.result as string
    });
  });
}

```

D. Add Parties

- Name
- NIC (12 digits, validated)
- Role (Victim / Suspect / Witness)
- Statement Summary
- Hospitalization Checkbox
 - If checked: Select hospital from dropdown
 - This triggers JMO workflow

Multiple Parties:

- Can add unlimited parties
- Each appears as a card in the summary
- Can delete before submission

E. Submission

- Validates all required fields
- Generates unique Case ID (CRIM-XXXX)
- Saves to database via `executeSQLQuery('INSERT', newCase)`
- Form resets after success

Data Structure Created:

typescript

```
{  
    id: "CRIM-1234",  
    date: "2023-10-24",  
    venue: "Borella Junction",  
    desc: "Traffic accident...",  
    reporterRole: "Victim",  
    evidence: [...],  
    parties: [...],  
    status: "Pending",  
    victimName: "John Doe", // First victim or first party  
    jmoRequired: true, // If any party is hospitalized  
    assignedCourt: "Magistrate Court Fort",  
    station: "Mount Lavinia HQ",  
    officer: "OIC Perera"  
}
```

2.3 Case Viewing & Editing

Features:

- Click "Info" icon to open full case modal
- View-only mode by default
- Click "EDIT" to enable editing

Editable Fields:

- Venue
- Description
- Reporter Role
- Evidence (can add new files)

Non-Editable:

- Case ID
- Date
- Parties (cannot modify after creation)

Status Update:

- Dropdown: Pending → Under Investigation → Closed
- Cannot change if Judge has updated status

Save Changes:

- Updates database immediately
 - Refreshes table view
-

2.4 Advanced Filtering

Search Bar:

- Case ID
- Victim Name
- Venue

Filters:

- Court Assignment (dropdown)
- Start Date
- End Date

Combined Logic: All filters work together (AND operation).

2.5 Password Change Feature

Steps:

1. Click profile icon → "Change Password"
2. Enter current password
3. Enter new password (must meet requirements)
4. Confirm new password
5. System validates and updates

Validation Rules:

- Minimum 6 characters
- At least 1 uppercase letter
- At least 1 lowercase letter
- At least 1 special character (!@#\$%^&*)

Security:

- Verifies current password before allowing change
- Updates `(justiceflow_users_final)` in localStorage
- Forces logout after successful change

Password Visibility Toggle: Eye icon to show/hide

FEATURE 3: JMO DASHBOARD - MEDICAL EXAMINATION

Purpose

Medical officers examine hospitalized patients and submit official reports that become court evidence.

Sub-Features

3.1 Hospital-Based Patient Filtering

Logic:

- Hardcoded hospital: `(CURRENT_HOSPITAL = "National Hospital Colombo")`
- Flattens all cases into individual patient records
- Only shows patients where:
 - `(isHospitalized = true)`
 - `(hospitalName = CURRENT_HOSPITAL)`

Why Flattening? Cases can have multiple parties. We need to show each hospitalized person as a separate row.

Code:

```
typescript
```

```

allCases.forEach((c) => {
  c.parties.forEach((p, index) => {
    if (p.isHospitalized && p.hospitalName === CURRENT_HOSPITAL) {
      records.push({
        uniqueId: `${c.id}-${index}`,
        caseData: c,
        partyData: p,
        partyIndex: index
      });
    }
  });
});

```

3.2 Patient Search & Filters

Search Bar:

- Patient Name
- NIC Number
- Case ID

Date Range Filter:

- Filter by reported date
- Cannot select future dates

Table Columns:

- Case Ref (linked to original case)
 - Patient Name
 - Reported Date
 - Last Update (shows when JMO last submitted report)
 - Condition (visual status indicator)
 - Status (Examined / Action Required)
-

3.3 Medical Examination Workspace

Opens when: Click "EXAMINE" button

Left Panel - Case Info:

- Patient name and NIC
- Incident summary from police

Right Panel - Medical Form:

A. Current Condition Dropdown:

- In Hospital - Stable
- In Hospital - Critical
- Discharged
- Deceased

B. Medical Observations:

- Large text area for detailed findings
- Example: "Patient sustained multiple fractures to left arm. X-ray shows clean break at radius. Recommended surgery."

C. Document Upload:

- Select Document Type First (dropdown):
 - Medical Report
 - X-Ray Scan
 - Lab Result
 - Forensic Image
- Then click "ADD" to upload file
- File size limit: 500KB
- Preview with delete option

Submission:

- Click "SUBMIT OFFICIAL REPORT"
- Updates `(parties[index].medicalReport)` in case object
- Saves via `executeSQLQuery('UPDATE', caseToUpdate)`

Report Structure:

```
typescript
```

```
{  
  status: "In Hospital - Critical",  
  notes: "Medical observations...",  
  documents: [  
    {  
      name: "xray.jpg",  
      type: "X-Ray Scan",  
      date: "2023-10-25",  
      fileData: "data:image/jpeg;base64,..."  
    }  
  ],  
  updatedDate: "2023-10-25",  
  officer: "Dr. Perera (JMO)"  
}
```

3.4 Visual Status Indicators

Condition Column Colors:

- Critical → Red background
- Deceased → Black background
- Discharged → Blue background
- Stable → Green background
- Pending → Gray background

Status Column:

- Examined → Green pill badge
- Action Required → Red pill badge with pulse animation

Last Update Column:

- Shows date with clock icon if report exists
 - Shows "No updates" if never examined
-

FEATURE 4: ATTORNEY DASHBOARD - CASE REVIEW

Purpose

Lawyers can securely access case files by searching with exact Case ID. They have read-only access to all

evidence.

Sub-Features

4.1 Search-Based Access

Why search-only?

- Privacy: Lawyers should only access cases they're working on
- Security: No case listing prevents browsing

Search Box:

- Large centered input
- Placeholder: "Enter Case ID (e.g., JMO-1234)"
- Submit on Enter or button click

Validation:

- Case ID must exist in database
 - Error: "Restricted: Case ID not found or access denied"
-

4.2 Case Overview Panel

Left Column - Case Info:

- Case ID (large, highlighted)
- Victim Name
- Status
- Judicial Remarks (if judge has added any)

Judicial Remarks Box:

- Dark background with quote styling
 - Shows verdict/comments from judge
 - Placeholder if no remarks yet
-

4.3 JMO Medical Report View

Right Column - Medical Evidence:

- Medical Observations (in styled quote box)
- Evidence Attachments (grid of documents)

Document Cards:

- Shows file name
 - Shows document type (X-Ray, Lab Result, etc.)
 - Click to preview
-

4.4 Document Preview Modal

Features:

- Full-screen overlay
- Header with file name and case ID
- Close button (X)

Content Viewer:

- **Images:** Display with `` tag
- **PDFs:** Embedded with `<object>` tag
- **Fallback:** Download link if preview fails

Code:

```
typescript
{previewDoc.type.includes('image') ? (
  <img src={previewDoc.fileData} alt="Evidence" />
) : (
  <object data={previewDoc.fileData} type="application/pdf">
    <a href={previewDoc.fileData} download>Download File</a>
  </object>
)}
```

4.5 Read-Only Restrictions

What Attorneys CANNOT Do:

- Edit case details

- Add evidence
- Change status
- Submit reports
- See other cases without knowing ID

Visual Indicator: "Read Only Mode" badge at top

FEATURE 5: JUDGE DASHBOARD - JUDICIAL RULINGS

Purpose

Judges search for cases assigned to their court, review evidence, and issue rulings.

Sub-Features

5.1 Court Jurisdiction Check

Hardcoded Court: `ASSIGNED_COURT = "Magistrate Court Fort"`

Security Check:

```
typescript
if (found.assignedCourt !== ASSIGNED_COURT) {
    setErrorMsg("ACCESS DENIED: This case belongs to another court.");
    return;
}
```

Why this matters: Prevents judges from accessing cases outside their jurisdiction.

5.2 Case Search

Search Box:

- Enter Case ID
- System checks:
 1. Does case exist?
 2. Is it assigned to this court?
- Displays error if either check fails



5.3 Case Workspace Layout

Left Side (2/3 width) - Read-Only Data:

A. Case Header Card:

- Case ID (large)
- Date and Venue
- Status badge (top-right corner)
- Incident summary (quoted text)

B. Parties & Evidence Grid:

- Left: List of involved parties (name, role, NIC)
- Right: Police evidence files

C. Court Proceedings History:

- Timeline of all judicial actions
- Shows date, action, details, next hearing date

Right Side (1/3 width) - Action Panel: Dark background, sticky position

5.4 Judicial Ruling Form

A. Status Dropdown:

- Pending
- Adjourned (Next Hearing)
- Referred to Higher Court
- Case Dismissed
- Closed

B. Dynamic Conditional Fields:

If "Adjourned" selected:

- Shows "Next Hearing Date" calendar input
- Required field
- Saves to `courtHistory[].nextDate`

If "Case Dismissed" or "Referred to Higher Court" selected:

- Shows "Reason for Order" text area
- Required field
- Example: "Insufficient evidence to proceed"

C. Hearing Notes:

- Always visible
- Optional field
- General comments for court record

D. Upload Court Document:

- Button to attach orders, notices, etc.
 - (Currently UI-only, can be extended)
-

5.5 Save Ruling

What happens when judge clicks "UPDATE RECORD":

1. Validates required fields
2. Creates court history entry:

```
typescript
{
  date: "2023-10-26",
  action: "Ruling: Adjourned",
  details: "Case postponed pending medical reports",
  nextDate: "2023-11-10"
}
```

3. Appends to existing [courtHistory] array
4. Updates case status
5. Saves to database
6. Shows success alert: "Court Order Saved & Police Notified"
7. Refreshes view

Effect on Other Users:

- Police see updated status in their dashboard
 - Case status cannot be changed back to Pending by police
-

FEATURE 6: LOGIN SYSTEM

Sub-Features

6.1 Two-Step Login Flow

Step 1: Role Selection

- Visual cards for each role
- Icons and descriptions
- Click to proceed

Step 2: Credentials

- Pre-fills email based on role
 - Password input with visibility toggle
 - Forgot password link
-

6.2 Authentication Logic

Priority 1: Check localStorage (Updated Passwords)

```
typescript
const dbUser = users.find(u => u.email === email);
if (dbUser && dbUser.password === password) {
  onLogin(dbUser);
  return;
}
```

Priority 2: Fallback to System Accounts

```
typescript
```

```
const systemUser = SYSTEM_ACCOUNTS.find(u => u.email === email && u.password === password);
if (systemUser) {
  // Auto-save to localStorage for future
  localStorage.setItem('justiceflow_users_final', JSON.stringify([...users, systemUser]));
  onLogin(systemUser);
}
```

Why Two Levels?

- Initial login uses hardcoded accounts
 - After password change, checks localStorage
 - Prevents old password from working after update
-

6.3 Session Management

On Login:

- Saves full user object to `sessionStorage`
- Redirects to role-specific dashboard

On Page Refresh:

- Checks `sessionStorage` for active user
- Restores session without re-login

On Logout:

- Clears `sessionStorage`
 - Returns to login page
-

FEATURE 7: DATABASE SYSTEM

Purpose

Centralized data management using `localStorage` as a simple database.

Functions

7.1 SELECT (Read All)

```
executeSQLQuery('SELECT')
```

```
// Returns: array of all cases
```

7.2 INSERT (Create New)

```
typescript
executeSQLQuery('INSERT', newCase)
// Adds to beginning of array (newest first)
// Saves to localStorage
```

7.3 UPDATE (Modify Existing)

```
typescript
executeSQLQuery('UPDATE', updatedCase)
// Finds case by ID
// Replaces entire object
// Saves to localStorage
```

Initial Seed Data

```
typescript
const INITIAL_DATA = [
{
  id: "CASE-2023-001",
  date: "2023-10-24",
  desc: "Traffic accident...",
  status: "Open",
  assignedCourt: "Magistrate Court Fort",
  parties: [...]
}
];
```

Loaded on first run if localStorage is empty.

FEATURE 8: UI/UX FEATURES

8.1 Color Coding System

- **Police:** Blue theme
- **JMO:** Green/Emerald theme
- **Attorney:** Amber/Gold theme
- **Judge:** Purple/Amber theme

- **Admin:** Red/Slate theme

8.2 Icons (Lucide React)

- Gavel → Justice/Courts
- Shield → Police
- Stethoscope → Medical
- User → Attorneys
- Settings → Admin
- FileText → Documents
- Upload → File uploads
- Eye → Preview
- Edit → Modify
- Trash → Delete

8.3 Animations

- Fade-in on modal open
- Pulse on urgent status badges
- Hover effects on buttons
- Scale on button press

8.4 Responsive Design

- Grid layouts for cards
- Flex containers for headers
- Mobile-friendly (md: breakpoints)
- Sticky headers

8.5 Form Validation

- Visual red borders on error
 - Alert messages
 - Disabled submit until valid
 - Character limits (e.g., NIC = 12)
-

TESTING CHECKLIST

Admin

- Create police station
- Create hospital
- Create court
- Register user for each role
- Edit user without changing password
- Delete organization

Police

- Register incident
- Add multiple parties
- Mark party as hospitalized
- Upload evidence
- Assign to court
- Edit case
- Filter cases
- Change password

JMO

- View hospitalized patients
- Search patient
- Update condition
- Add medical notes
- Upload X-ray
- Submit report
- Verify last update date

Attorney

- Search by case ID
- View case details
- Preview evidence
- Attempt edit (blocked)

Judge

- Search case

- Verify jurisdiction
 - Adjourn case
 - Set next hearing
 - Dismiss case
 - View court history
-

CONCLUSION

This system integrates **5 user roles, 30+ features, and 6 major workflows** into a cohesive digital evidence management platform.

Each feature is designed to reflect real-world legal processes while maintaining simplicity and usability.