

I've created a comprehensive web UI for the proof repository operations team. Here are the key features:

## Core Functionality

### Dashboard Overview

- **Real-time sidebar navigation** with badge counters for pending items
- **Pending Reviews:** Shows documents awaiting verification
- **Validation Issues:** Critical issues requiring immediate attention
- **My Tasks:** Assigned maintenance tasks
- **System Health:** Overall repository status

### Proof Verification Workflow

#### Document Review Cards

- **Document metadata:** Reference, discovery date, type
- **Entity references:** Shows which MDM systems contain related entities
- **Extracted relationships:** Preview of relationship assertions from document analysis
- **Status indicators:** Visual status badges (pending, verified, disputed)

#### Interactive Verification Process

- **5-step verification workflow** with mandatory checkboxes:
  1. Document Authentication (hash, signatures, tampering)
  2. Authority Verification (issuing authority, jurisdiction, dates)
  3. Content Analysis (entities, percentages, clarity)
  4. Consistency Check (conflicts, logical validation)
  5. Final Verification (notes and completion)

### Relationship Establishment

- **Modal interface** for creating cross-system relationships
- **Entity search** with auto-complete for source/target entities
- **Relationship type selection** (owns, controls, operates\_on\_behalf\_of, etc.)
- **Attribute specification** (ownership %, authority scope, dates)
- **Validation** ensures all required fields are completed

## User Experience Features

### Operational Efficiency

- **Auto-save progress** every 30 seconds during verification
- **Keyboard shortcuts** (Ctrl+R for refresh, Esc to close modals)
- **Bulk actions** for processing multiple documents
- **Real-time updates** with badge count changes
- **Export functionality** for compliance reporting

## Safety and Compliance

- **Confirmation dialogs** for destructive actions (dispute, delete)
- **Audit trail** logging for all user actions
- **Required verification notes** before completion
- **Comprehensive validation** of relationship data

## Responsive Design

- **Mobile-friendly** layout that adapts to different screen sizes
- **Sticky sidebar** for easy navigation
- **Progressive disclosure** (verification panel shows when needed)
- **Loading indicators** for better user feedback

## Integration Points

### Backend API Integration

The UI is designed to integrate with REST APIs for:

- **Document retrieval:** `GET /api/proofs/{id}/document`
- **Verification updates:** `POST /api/proofs/{id}/verify`
- **Relationship creation:** `POST /api/relationships`
- **Entity search:** `GET /api/entities/search?q={query}`
- **Health monitoring:** `GET /api/health`

### Real-time Features

- **WebSocket connections** for real-time updates
- **Push notifications** for urgent issues
- **Collaborative editing** prevention (show when others are working on same proof)

## Security Considerations

### Access Control

- **Role-based permissions** (different capabilities for junior/senior operators)
- **Document access logging** for audit compliance
- **Session management** with timeout warnings
- **Secure document viewing** (no download/print restrictions if needed)

This UI provides operations staff with a **professional, efficient interface** for maintaining the proof repository while ensuring **regulatory compliance** and **audit readiness**. The guided workflow reduces errors and ensures consistent verification quality across all operators.