

Community App

Audit Log

**Contents**

[1 Overview and Justification 3](#_Toc204610833)

[1.1 Overview 3](#_Toc204610834)

[1.2 Justification 3](#_Toc204610835)

[2 Scenarios 3](#_Toc204610836)

[3 Business Requirements 4](#_Toc204610837)

[4 Functional Requirements 4](#_Toc204610838)

[5 Dependencies 4](#_Toc204610839)

[6 Checklist 4](#_Toc204610840)

[7 Change and Review history 5](#_Toc204610841)

**Quick Spec – Audit Log**

Author: Tahta Alfionita (tahta@slerate.com)

This document is intended to be used as part of the SLERATE spec Development and Review Process.

Confidentiality Notice

This document is an internal working document of SLERATE. It contains trade secrets and other proprietary information, which are the confidential property of SLERATE.  Neither this document nor the information contained within is to be distributed—in whole or part— outside of the SLERATE team by any means without prior authorization, including but not limited to printed, magnetic, electronic or verbal forms.

1. Overview

The Audit Log feature provides a detailed, secure, and accessible record of significant actions performed within the Admin Panel of the community app. It supports monitoring, accountability, troubleshooting, and compliance by capturing who did what, when, where, and the outcome of each action.

1. Scenarios

|  |  |
| --- | --- |
| **Scenario Name** | **Viewing Audit Log Entries** |
| Background | Manjit wants to see what another role does in admin panel to prevent any suspicious activity |
| Objective | Enable authorized users to efficiently search, filter, and view detailed audit log entries reflecting all significant admin panel actions. |
| Persona(s) | Manjit – Manager |
| Scenario Steps | 1. Manjit logs into the Admin Panel. 2. Manjit navigates to the “Audit Log” section. 3. System displays a default view showing the latest audit log entries in a tabular format (timestamp, user, action, target, result). 4. Manjit utilizes filter options (date range, user, action type, result) to narrow down entries. 5. System dynamically updates the displayed entries according to the selected filters. 6. Manjit selects a specific log entry to review additional details. 7. System presents a detail view/modal with expanded metadata (e.g., affected object, full request/response payload, IP address, device). 8. Error Handling:    1. If filters return no entries, display: “No audit log entries found for the selected criteria.”    2. If the user is not authorized, system blocks access and displays: “You do not have permission to view audit logs.”    3. If log data fails to load (e.g., server error), system informs: “Unable to load audit logs. Please try again later.” |

|  |  |
| --- | --- |
| **Scenario Name** | **Recording a User Role Change** |
| Background | Manjit wants to change role for content creator to be admin, or change the person |
| Objective | Ensure every modification to user roles is automatically logged with complete context and outcome for future review. |
| Persona(s) | Manjit – Manager |
| Scenario Steps | 1. Manjit initiates a role update (e.g., upgrading a Content Creator to Admin) in the Admin Panel. 2. System processes the role change request. 3. After a successful update, system creates an audit log entry capturing:    1. Unique Audit ID    2. Timestamp of change    3. User performing the action (ID, username)    4. Affected user (ID, username)    5. Previous role and new role    6. Action result (success)    7. IP address and device info (if available) 4. Entry is instantly available in the Audit Log interface. 5. Error Handling:    1. If the role change fails (e.g., insufficient permissions, system error), system creates an audit entry with result “failure” and error details.    2. If an unauthorized user attempts to change a role, deny the action and log the unauthorized attempt as a security event. |

|  |  |
| --- | --- |
| **Scenario Name** | **Exporting Audit Logs** |
| Background | Manjit needs to export audit logs for further analysis or reporting purposes, in cases such as monthly performance reviews, suspicious activity investigations, or official audits. |
| Objective | Allow authorized users to export selected audit logs in standard file formats, preserving all necessary information and respecting applied filters. This might be required to analyze logs externally for regulatory audits, incident reviews, or reporting. |
| Persona(s) | Manjit – Manager |
| Scenario Steps | 1. Manjit navigates to the “Audit Log” section and applies filters to focus on the required subset (e.g., by date, action, user). 2. User clicks the “Export” button and selects output format (CSV or JSON). 3. System generates the export file containing all displayed (filtered) audit log entries with full details. 4. The file download prompt appears; user saves the file locally. 5. Error Handling:    1. If there are no entries matching the filters, system disables export or prompts: “No data available to export.”    2. If export process fails (e.g., server error), system displays: “Export failed. Please try again later.”    3. Exported data is validated for completeness and format correctness before download. |

|  |  |
| --- | --- |
| **Scenario Name** | **Handling Unauthorized Access Attempts** |
| Background | Manjit needs visibility into any unauthorized access attempts. Whether from failed logins, blocked role-based actions, or suspicious access from unusual IP addresses, these events must be recorded in audit logs. This allows Manjit to detect potential threats early, investigate abnormal behavior, and take corrective action when needed. |
| Objective | Audit logs must capture both successful and failed access attempts to identify potential misuse, attacks, or unauthorized activities. |
| Persona(s) | Manjit – Manager |
| Scenario Steps | 1. A user attempts to access a restricted feature, or a login attempt is made with invalid credentials. 2. System denies the access or login and displays the appropriate error message to the user. 3. System records an audit log entry with:    1. Unique Audit ID    2. Timestamp    3. Provided username (if any)    4. Type of attempt (login, unauthorized resource access)    5. Request source (IP address, device)    6. Result (failure)    7. Error message or system response 4. Manjit can review all such events in the Audit Log with filtering options for “Failure” or “Unauthorized Access.” 5. Error Handling:    1. If log entries cannot be written (e.g., database error), system generates an alert or writes to a separate emergency log. |

1. Business Requirements

|  |  |  |
| --- | --- | --- |
|  | **Business Requirement** | **Priority** |
| 1 | Audit logs must capture all significant user activities in the Admin Panel affecting accounts, content, and settings. | High |
| 2 | Logs must detail the user who performed the action, timestamp, action type, target object, and result (success/failure). | High |
| 3 | Audit log data must be tamper-resistant and securely stored. | High |
| 4 | Authorized users must be able to view audit logs through a dedicated interface with filtering and search capabilities. | High |
| 5 | The interface must allow viewing detailed information about each logged event, including metadata like IP address and relevant request or response data. | Medium |
| 6 | Logs must record authorization decisions and changes to user roles and permissions. | High |
| 7 | Audit logs should support export functionality for offline analysis or regulatory needs. | Medium |
| 8 | System should archive or purge old logs according to retention policies configurable by administrators. | Medium |

1. Functional Requirements

|  |  |  |
| --- | --- | --- |
|  | **Functional Requirement** | **Business Requirement(s)** |
| 1 | Automatically record all admin actions: create, update, delete on users, roles, content, settings, and configuration changes. | 1 |
| 2 | Each audit log entry must contain at minimum: unique Audit ID, timestamp, user ID/username, action type, target entity/object, action result (success/failure), IP address, and platform/device info if available. | 2 |
| 3 | Provide a secure storage mechanism to protect logs from tampering and unauthorized access. | 3 |
| 4 | Implement a dedicated "Audit Log" section accessible only to authorized roles (e.g., Admin, Security Officer). | 4 |
| 5 | Support filtering logs by date/time range, user, action type, object type, and success/failure status. | 4 |
| 6 | Display detailed log information for each event in a modal or expanded view, including metadata and contextual request/response payloads where applicable. | 5 |
| 7 | Log changes to permission assignments and authorization decision points clearly. | 6 |
| 8 | Allow exporting filtered audit logs in common formats (CSV, JSON). | 7 |
| 9 | Provide configurable log retention and archival policies with automatic purging compliant with internal and external regulations. | 8 |

1. Dependencies

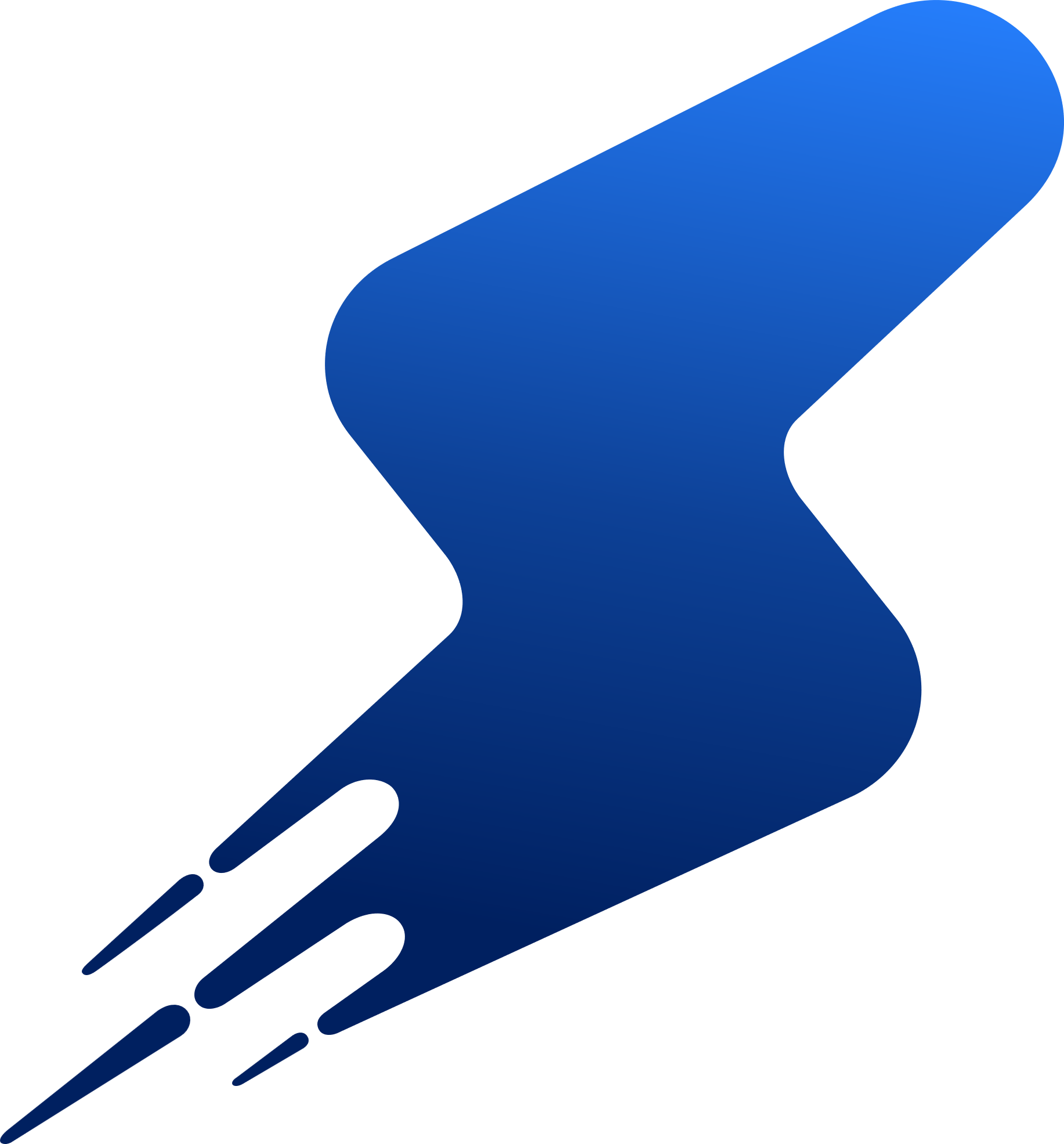
|  |  |  |  |
| --- | --- | --- | --- |
|  | **Team/Contact** | **Type** | **Description** |
| 1 | Backend Developer | Technical | Implements the logging system, defines log structure, and stores activity data |
| 2 | Frontend Developer | Technical | Builds UI in Admin Panel to view, filter, and search audit logs |

1. Checklist

|  |  |  |
| --- | --- | --- |
|  | **Checklist Item** | **Response** |
| 1 | Has this functional spec been reviewed by Dev? | No |
| 2 | Has this functional spec been reviewed by UI? | No |
| 3 | Has this functional spec been reviewed by Manager? | No |
| 4 | All critical admin and manager actions (e.g., approve, reject, role change) are logged | Pending |
| 5 | Each log entry includes user ID, action type, timestamp, and target entity (e.g., member ID) | Pending |
| 6 | Admin Panel includes an interface to view and filter logs by date, user, role, or action | Pending |
| 7 | Search functionality is implemented to quickly find specific logs | Pending |
| 8 | Logs are read-only and cannot be modified or deleted via the UI | Pending |
| 9 | Sensitive information (e.g., passwords) is excluded from logs | Pending |
| 10 | Only users with Admin or Manager roles can view audit logs | Pending |
| 11 | System automatically logs failed login attempts and unauthorized access attempts | Pending |
| 12 | Logs are stored securely and backed up regularly | Pending |
| 13 | Audit log storage follows retention policy (e.g., keep logs for 12 months unless archived/deleted) | Pending |

1. Change and Review history

| **Date** | **Author, editor, or Reviewer** | **Description of change(s)/Comments on review** |
| --- | --- | --- |
| 4/8/2025 | Tahta (Author) | Create this quick specs document |
|  |  |  |
|  |  |  |



SLERATE

August 2025

Copyright © SLERATE

www.SLERATE.com