AG RUNBOOK — Admin Dashboard

Purpose:

This runbook defines a safe, UI-only implementation for the Admin Dashboard in a multi-tenant SaaS. The dashboard provides high-level visibility into tenant security and user state without exposing raw data or introducing new backend behavior.

Global Constraints:

- Do not modify backend APIs or business logic
- Do not introduce write operations
- Do not bypass tenant isolation
- Do not aggregate cross-tenant data
- Do not expose PII beyond counts and summaries
- Assume Audit, Users, Roles, and Sessions APIs already exist

Dashboard Objectives:

- Give admins situational awareness
- Surface security-relevant signals
- Provide navigation entry points
- Avoid operational coupling

Allowed Widgets (Read-only):

1. User Overview

- Total users (ACTIVE / PENDING / DISABLED)
- Derived from users list API
- No user details shown

2. Session Overview

- Active session count
- Recently revoked sessions (last 24h)
- No IP or device details shown

3. Security Activity Summary

- Count of audit events in last 24h / 7d
- Breakdown by category:
- Auth
- Sessions
- Roles
- Data sourced from audit events API

4. Administrative Actions

- Recent admin actions (last 10)
- Action + timestamp + actor email
- Read-only summary view

5. System Status (Optional)

- Auth system: operational

- Audit logging: operational
- Tenant enforcement: operational
- Static indicators only (no health checks)

Step 1 — Dashboard Route:

Route:

- /admin/dashboard

Behavior:

- Load widgets independently
- Fail gracefully if one widget fails
- Never block page render

Step 2 — Data Fetching Rules:

- All data must be tenant-scoped
- Use existing APIs only
- No backend aggregation endpoints
- Perform aggregation in UI layer

Step 3 — UX Rules:

- No charts required (counts preferred)
- Click-through links to:
- Users
- Audit Viewer
- Roles
- No inline actions

Step 4 — Error & Empty States:

- Show “Data unavailable” for failed widgets
- Do not display raw error messages
- Empty state copy must be neutral

Audit Awareness:

- Dashboard must not generate audit events
- Dashboard reads do not require logging
- All write actions occur on dedicated pages

Success Criteria:

- Admins get quick situational awareness
- No sensitive data exposure
- Tenant isolation preserved
- Dashboard is non-critical and safe

Out of Scope:

- Cross-tenant dashboards
- Real-time updates
- Alerting

- SLA / uptime reporting
- System metrics