**Panel**

You got it. Here’s the **Admin Panel spec in English**—clean, bullet-style, ready to implement.

**1) Ready-to-use Messages (copy)**

**Pending (after submit)**

* Title: *Your account is under review*
* Body: *Thanks for signing up. Your profile is being reviewed (usually within 12 hours). If we need edits, we’ll send you notes.*

**Request Fix**

* Title: *We need a quick fix*
* Body: *Please update: {notes}. Then tap “Resubmit for review.”*

**Approved**

* Title: *Your account is live*
* Body: *You’re approved. Enjoy the app!*

**Rejected**

* Title: *We can’t approve your account*
* Body: *Reason: {reason}. You can reapply after fixing the issues.*

**Ban**

* Title: *Temporary account suspension*
* Body: *Your account is suspended until {ban\_until} for: {reason}.*

**Employee Invite**

* Title: *Admin Console invitation*
* Body: *An account with role* ***{role}*** *was created for you. Sign in at {link} and enable* ***MFA*** *on first login.*

**2) Roles (RBAC)**

* **Admin** – full control.
* **Supervisor (Moderator)** – approvals/reports/users/bans (no logs/alerts/incidents/settings/roles).
* **SOC Analyst** – logs/alerts/incidents only (no approvals/reports/users).
* **Viewer (optional)** – read-only dashboard/reports.

*MFA required for Admin/Supervisor/SOC.*

**3) Navigation (auto-filtered by role)**

* **Dashboard**
* **Approvals**
* **Reports**
* **Users**
* **Banned**
* **Logs**
* **Alerts**
* **Incidents**
* **Employees**
* **Support Inbox**
* **Archive**
* **Settings**
* **Roles & Access**
* **Audit Log**

**Sidebar visibility example**

{

"admin": ["Dashboard","Approvals","Reports","Users","Banned","Logs","Alerts","Incidents","Employees","Support Inbox","Archive","Settings","Roles & Access","Audit Log"],

"supervisor": ["Dashboard","Approvals","Reports","Users","Banned","Support Inbox"],

"soc": ["Dashboard","Logs","Alerts","Incidents","Support Inbox"],

"viewer": ["Dashboard","Reports"]

}

**4) Pages — who sees what & actions**

**Dashboard (Admin + read for Supervisor/SOC)**

* **Shows:** Pending count, approvals today, open reports, DAU/MAU, login success rate.
* **Actions:** Date range, deep links.

**Approvals (Admin + Supervisor)**

* **Shows:** pending queue + full profile card (answers/photos).
* **Actions:** **Approve**, **Reject (reason)**, **Request-Fix (notes)**.
* **Filters:** gender, city, date.

**Reports (Admin + Supervisor)**

* **Shows:** reports on **chat/profile** with severity & evidence.
* **Actions:** **Warn**, **Ban (24h/7d/30d/Perm)**, **Dismiss**.
* **Filters:** type, severity, status (open/closed).

**Users (Admin + Supervisor)**

* **Shows:** search, status (active/pending/banned), last login.
* **Actions:** **Ban/Unban**, **Edit**, **Request-Fix**.

**Banned (Admin + Supervisor)**

* **Shows:** user, ban type, ban\_until, reason, by/at.
* **Actions:** **Unban**, **Extend Ban**, **View History**.

**Logs (Admin + SOC)**

* **Shows:** auth\_events (login\_success/fail, device, ip\_hash).
* **Actions:** **Filter**, **Export CSV**, **Link to Incident**.

**Alerts (Admin + SOC)**

* **Shows:** brute\_force, shared\_ip, new\_device.
* **Actions:** **Acknowledge**, **Create Incident**.

**Incidents (Admin + SOC)**

* **Shows:** open/monitoring/closed incidents + timeline.
* **Actions:** **Assign**, **Add Note**, **Close**.

**Employees (Admin only)**

* **Shows:** name, email, role, status (active/disabled), MFA, last login.
* **Actions:** **Add Employee**, **Change Role**, **Disable/Enable**, **Reset MFA**.

**Support Inbox (Admin + Supervisor + SOC)**

* **Shows:** tickets (topic, priority, status, assignee).
* **Actions:** **Assign**, **Reply**, **Close**, **Convert → Report / Incident**.

**Archive (Admin only)**

* **Shows:** archived **Users / Conversations / Reports / Employees / Media refs**.
* **Actions:** **View**, **Restore (restricted)**, **Purge** (after retention).

**Settings (Admin only)**

* **Actions:** save policies (alert thresholds, default ban durations, review SLAs, message templates).

**Roles & Access (Admin only)**

* **Actions:** add/change/remove staff roles.

**Audit Log (Admin only)**

* **Shows:** who did what, when; filter & export.

**5) Data model (collections) — essentials**

**users**

{ "uid":"...", "status":"draft|pending|active|rejected|banned",

"ban\_until":"2025-11-30T00:00:00Z", "created\_at":"...", "last\_login\_at":"..." }

**approvals\_queue**

{ "user\_id":"...", "submitted\_at":"...", "notes":"..." }

**reports**

{ "report\_id":"...", "type":"chat|profile", "severity":"low|med|high",

"reporter\_id":"...", "target\_id":"...", "evidence":["..."],

"status":"open|closed", "created\_at":"..." }

**moderation\_actions**

{ "user\_id":"...", "action":"warn|ban\_24h|ban\_7d|ban\_30d|ban\_perm",

"reason":"...", "by\_admin":"uid", "at":"..." }

**bans** (optional separate)

{ "user\_id":"...", "type":"24h|7d|30d|perm", "ban\_until":"...", "reason":"...", "by":"uid", "at":"..." }

**auth\_events**

{ "event":"login\_success|login\_fail|signup", "user\_id":"...",

"timestamp":"...", "ip\_hash":"...", "device":"ios|android|web", "reason":"..." }

**alerts**

{ "alert\_id":"...", "kind":"brute\_force|shared\_ip|new\_device",

"details":{ "user\_id":"...","ip\_hash":"..." },

"status":"open|ack|closed", "created\_at":"..." }

**incidents**

{ "incident\_id":"...", "source\_alert":"alert\_id", "assignee":"uid|null",

"notes":[{ "by":"uid","text":"...","at":"..." }],

"status":"open|monitoring|closed", "created\_at":"..." }

**employees**

{ "uid":"...", "name":"...", "email":"...", "role":"admin|supervisor|soc|viewer",

"status":"active|disabled", "mfa\_required":true, "created\_at":"...", "last\_login\_at":"..." }

**archive**

{ "archive\_id":"...", "type":"user|conversation|report|employee|media",

"original\_id":"...", "payload":{...}, "deleted\_by":"uid",

"deleted\_at":"...", "retention\_until":"..." }

**support\_tickets / support\_messages**

{ "ticket\_id":"...", "user\_id":"...", "topic":"login|billing|abuse|other",

"priority":"low|med|high", "status":"open|pending|solved",

"assignee":"uid|null", "created\_at":"...", "updated\_at":"..." }

{ "ticket\_id":"...", "from":"user|staff:uid", "text":"...",

"attachments":[...], "at":"..." }

**6) Cloud Functions (API signatures)**

// Dashboard & Settings

getDashboardStats({ from: string, to: string })

updateSettings({ policyKey: string, value: any })

// Approvals

listPendingApprovals({ filters?: any, page?: number })

approveUser({ userId: string })

rejectUser({ userId: string, reason: string })

requestFix({ userId: string, notes: string })

// Reports & Moderation

listReports({ type?: "chat"|"profile", severity?: "low"|"med"|"high", status?: "open"|"closed", page?: number })

resolveReport({ reportId: string, action: "warn"|"ban\_24h"|"ban\_7d"|"ban\_30d"|"ban\_perm"|"dismiss", reason?: string })

// Users & Bans

searchUsers({ q?: string, status?: "draft"|"pending"|"active"|"rejected"|"banned", page?: number })

banUser({ userId: string, until?: string, reason: string })

unbanUser({ userId: string })

requestProfileFix({ userId: string, notes: string })

// Logs & Alerts & Incidents (SOC)

queryAuthEvents({ event?: "login\_success"|"login\_fail"|"signup", userId?: string, ipHash?: string, from?: string, to?: string, page?: number })

listAlerts({ status?: "open"|"ack"|"closed", page?: number })

ackAlert({ alertId: string })

createIncidentFromAlert({ alertId: string })

listIncidents({ status?: "open"|"monitoring"|"closed", page?: number })

updateIncident({ incidentId: string, assignee?: string, note?: string, status?: "open"|"monitoring"|"closed" })

// Employees (Admin)

listEmployees()

createEmployee({ email: string, name: string, role: "admin"|"supervisor"|"soc"|"viewer" })

setEmployeeRole({ uid: string, role: "admin"|"supervisor"|"soc"|"viewer" })

setEmployeeStatus({ uid: string, status: "active"|"disabled" })

resetEmployeeMFA({ uid: string })

// Support Inbox

listTickets({ status?: "open"|"pending"|"solved", priority?: "low"|"med"|"high", assignee?: string, page?: number })

getTicket({ ticketId: string })

replyTicket({ ticketId: string, message: string })

assignTicket({ ticketId: string, assignee: string })

closeTicket({ ticketId: string })

createReportFromTicket({ ticketId: string })

createIncidentFromTicket({ ticketId: string })

// Archive

listArchive({ kind: "user"|"conversation"|"report"|"employee"|"media", from?: string, to?: string, page?: number })

viewArchiveItem({ archiveId: string })

restoreFromArchive({ archiveId: string }) // restricted

purgeArchive({ archiveId: string }) // post-retention

**7) Security rules (principles)**

* No direct client reads on **reports / moderation\_actions / auth\_events / alerts / incidents / employees / archive**. Access only via Functions that check **role**.
* Enforce **MFA** for Admin/Supervisor/SOC; block **employees.status = "disabled"**.
* Log all admin actions to **audit\_log**.
* Add schedulers for alerts (e.g., ≥5 login\_fail/10m, ≥3 signups/hour from same IP).

If you want, I can also provide a **minimal Next.js sidebar component** that hides/shows items by role, plus a **sample Firestore security rules snippet** you can paste.