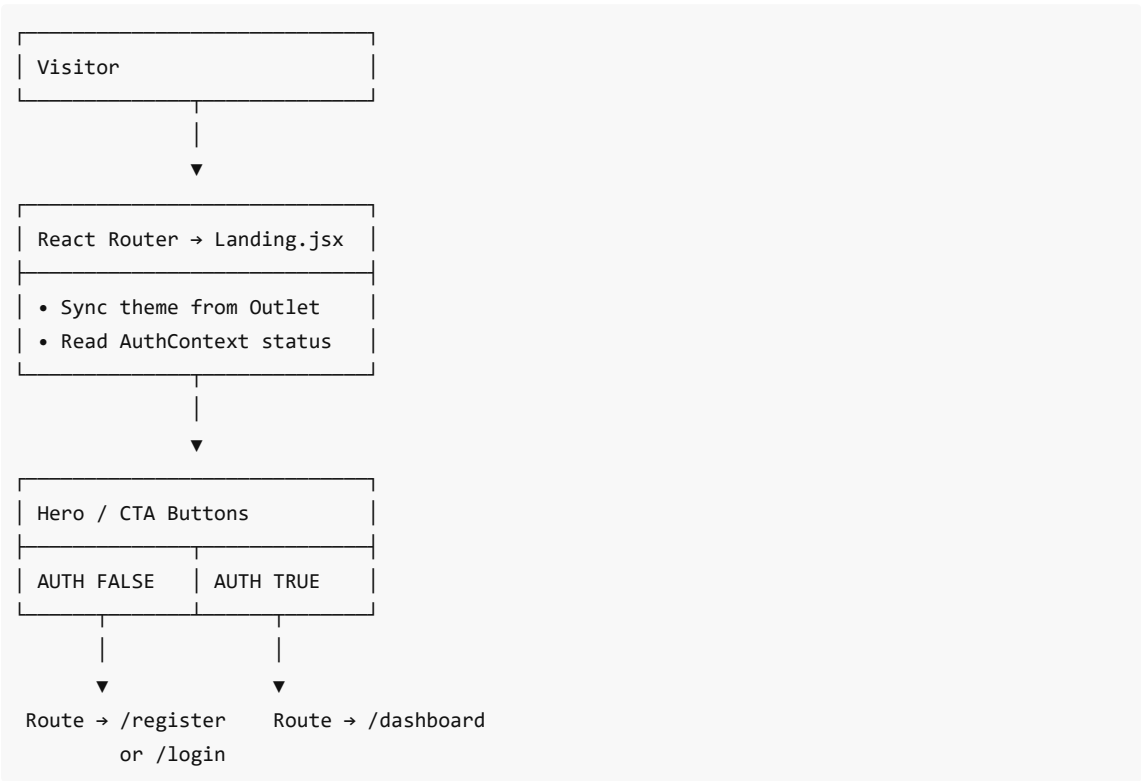


Secure UPI – Page Workflows

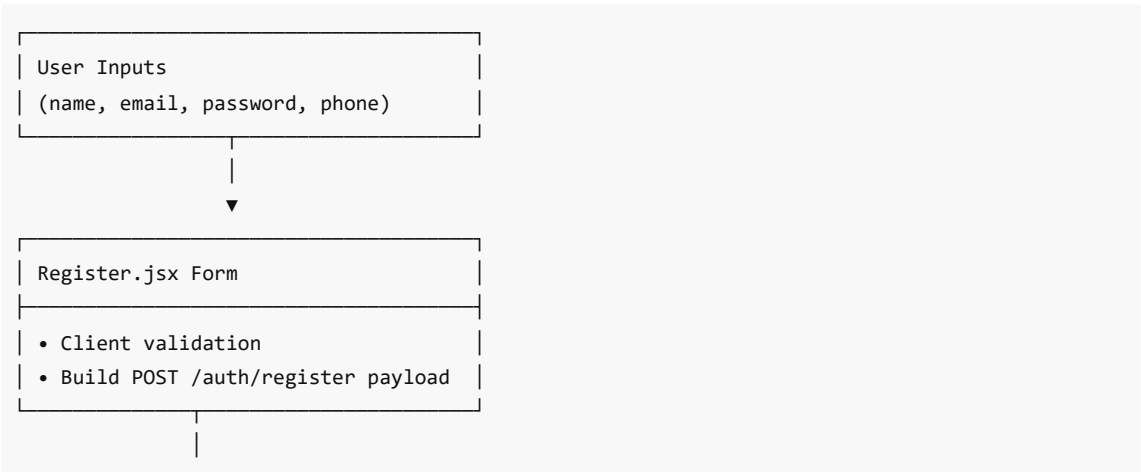
The following workflows mirror the schematic style from your PPT reference. Each diagram shows how events move from user input, through the React page, into backend routes (backend/routes/*), supporting services, and finally to MongoDB or the ML services.

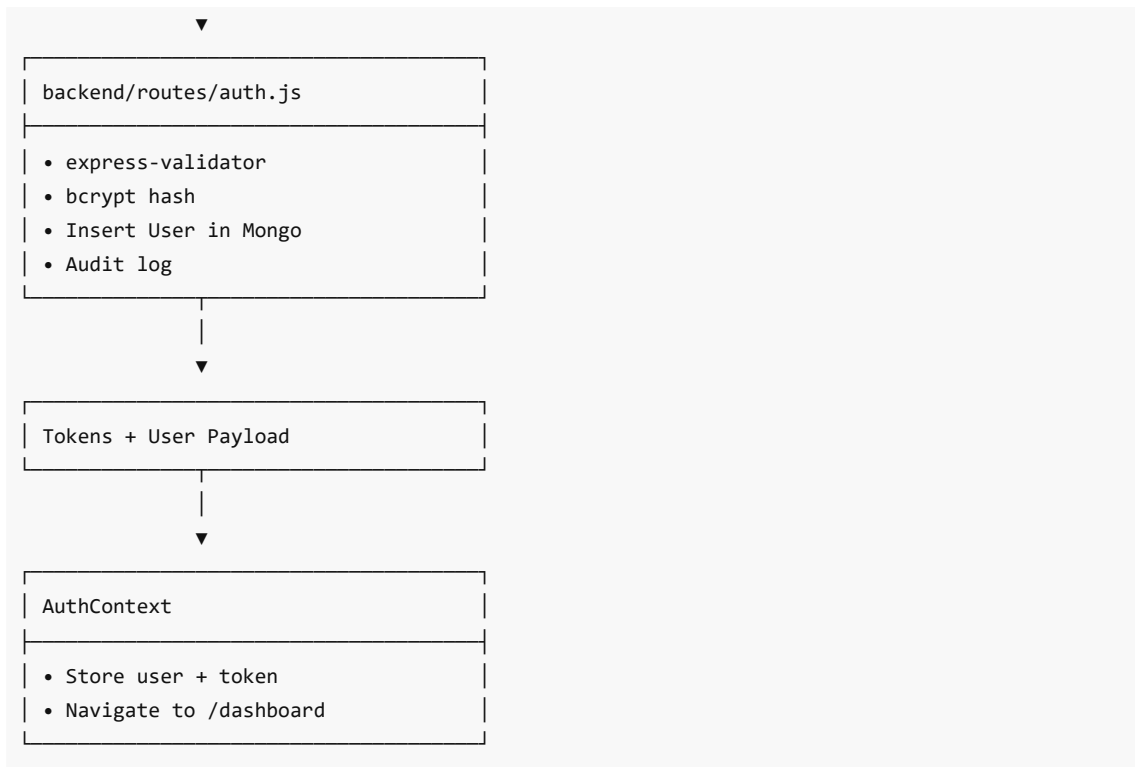
Landing (Landing.jsx)

- Greets authenticated and guest users, syncs theme via AuthContext .
- CTA buttons simply route users toward Register/Login or Dashboard, so no backend call happens here.



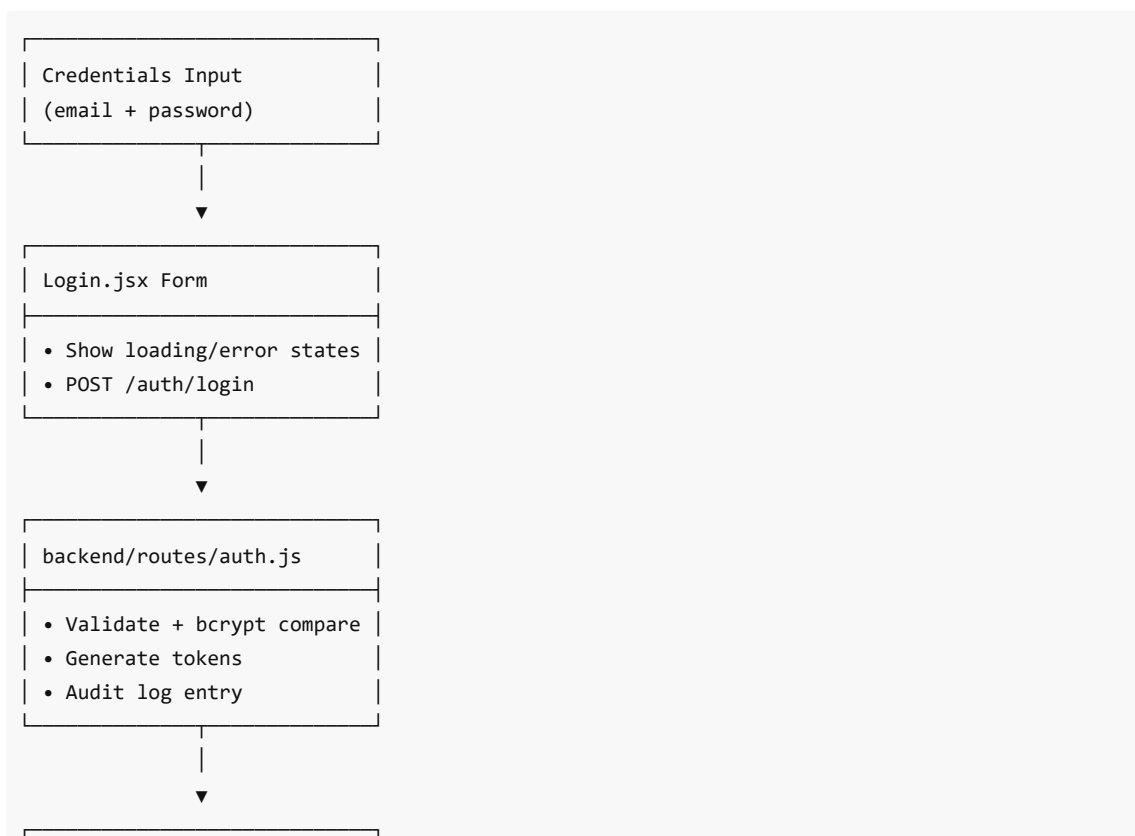
Register (Register.jsx → /auth/register)

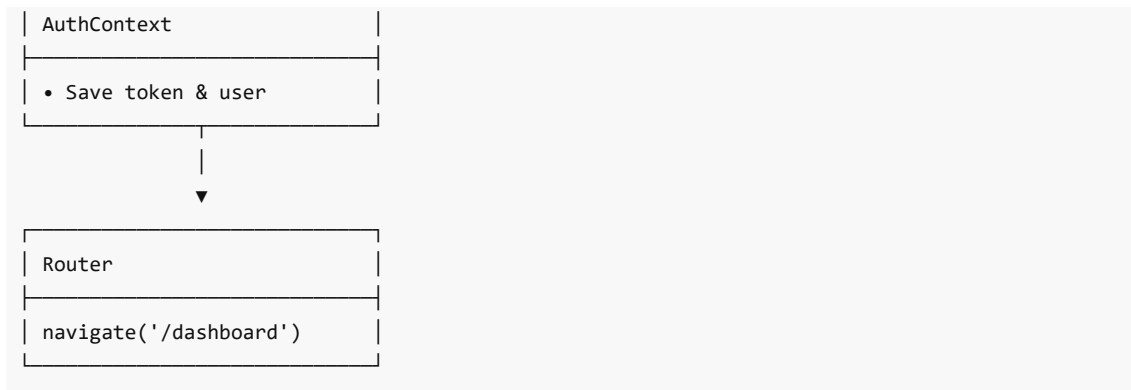




- Failure path: validation error or duplicate email returns `400` , surfaced near the form.

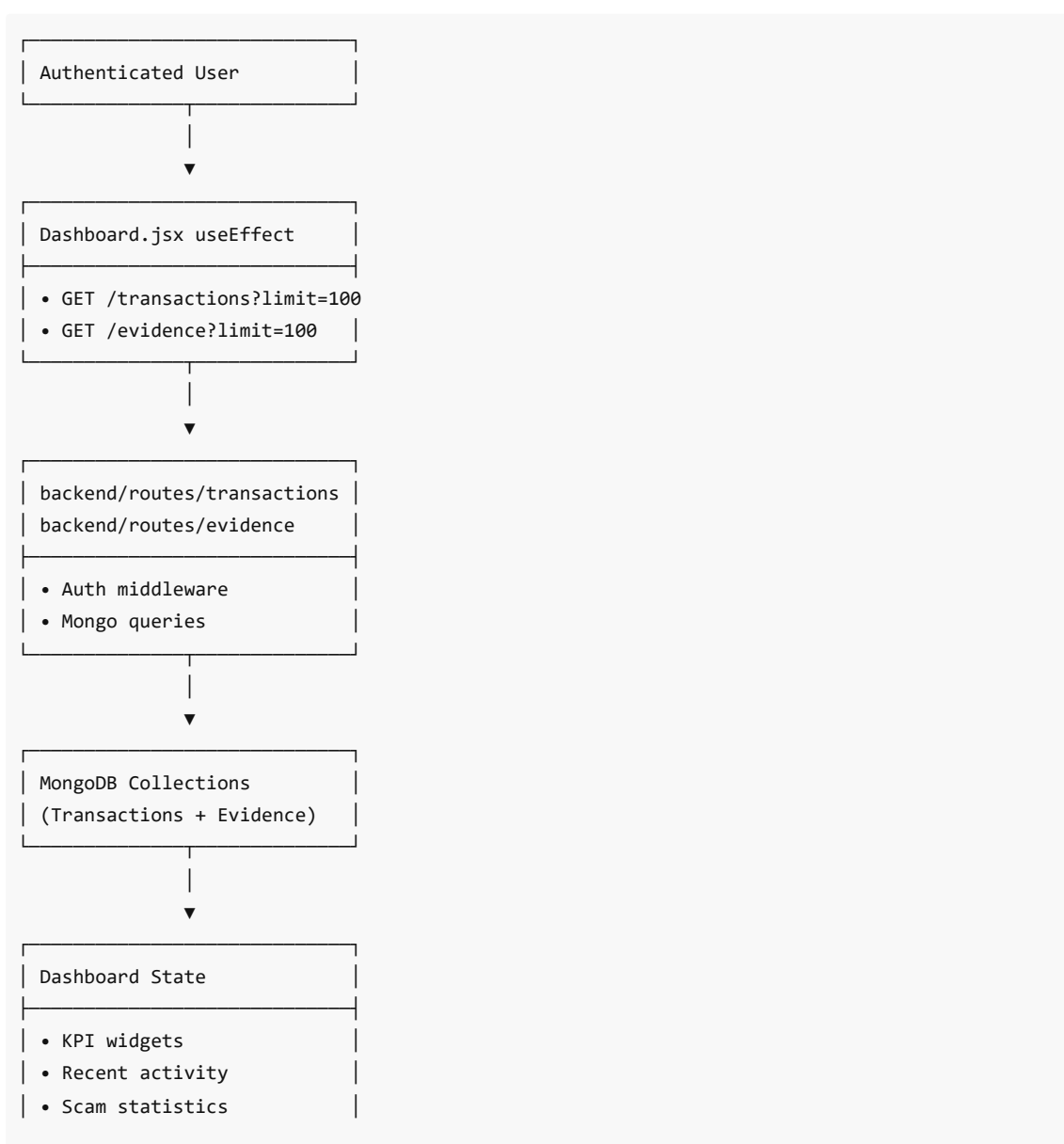
Login (Login.jsx → /auth/login)





Errors return `[ERROR]` banner with backend message (e.g., invalid credentials).

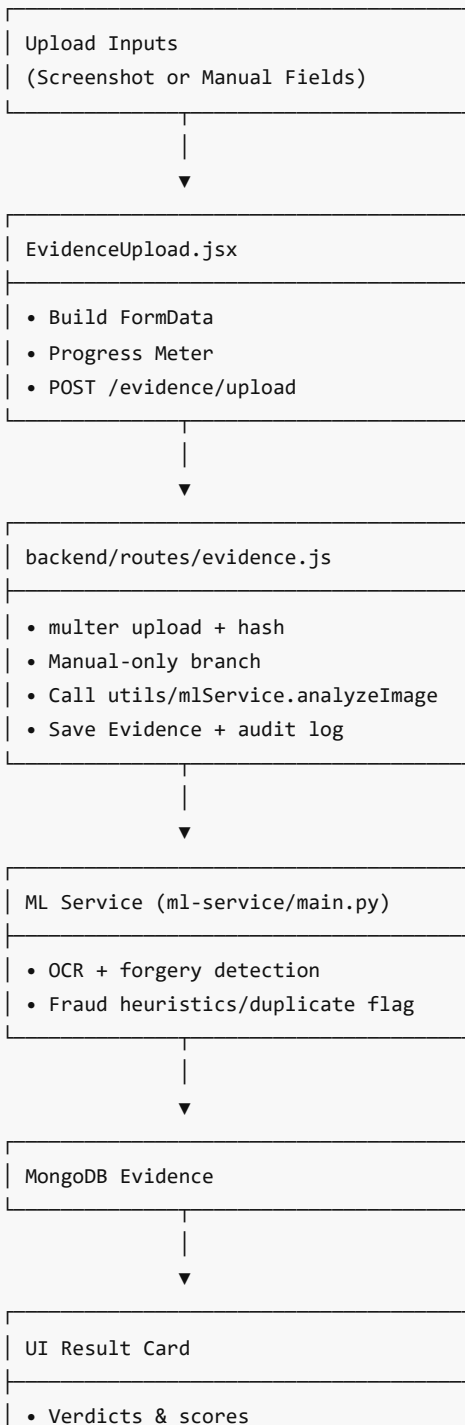
Dashboard (`Dashboard.jsx`)



- Feature cards

- Fallback timers animate progress and terminal text even if data is delayed.

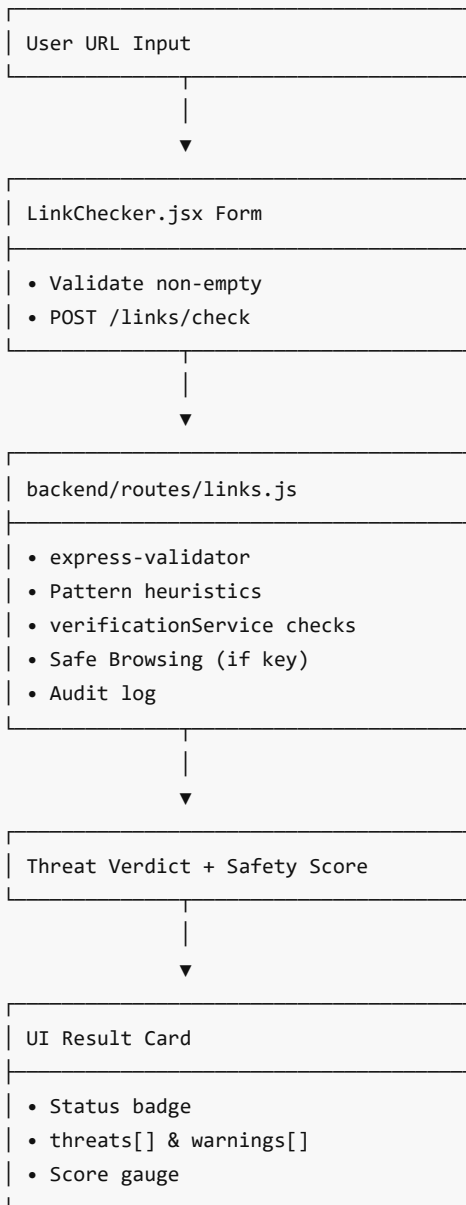
Evidence Upload (`EvidenceUpload.jsx` → `/evidence/upload`)



- Extracted data & indicators

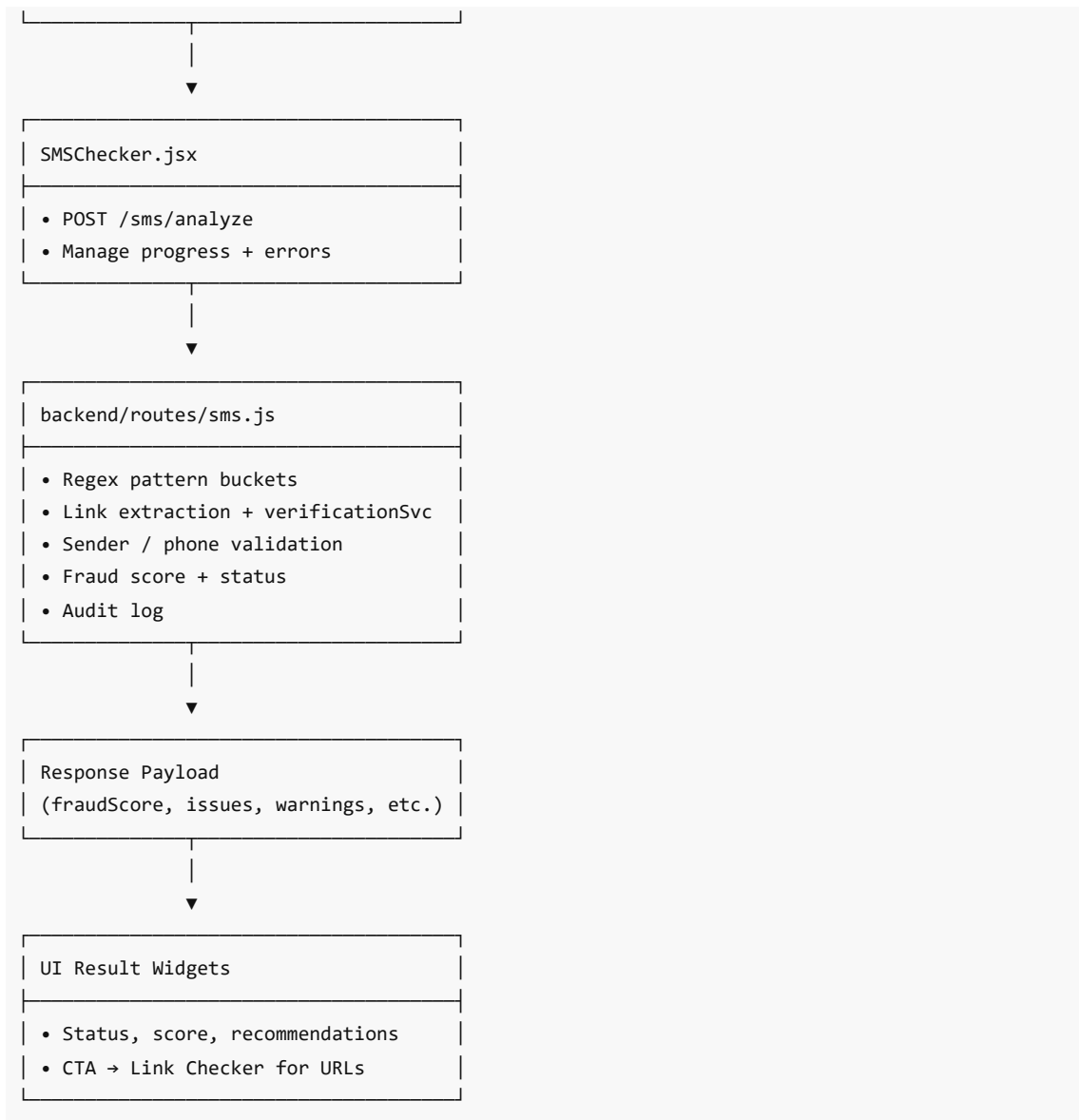
Errors (Multer limits, validation, ML failures) are surfaced via `[ERROR]` banners.

Link Checker (`LinkChecker.jsx` → `/links/check`)

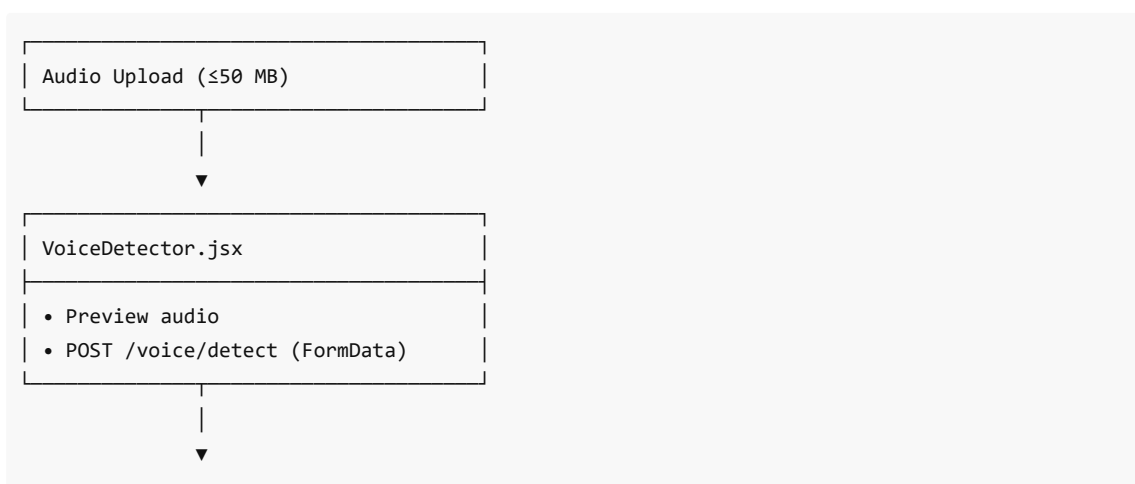


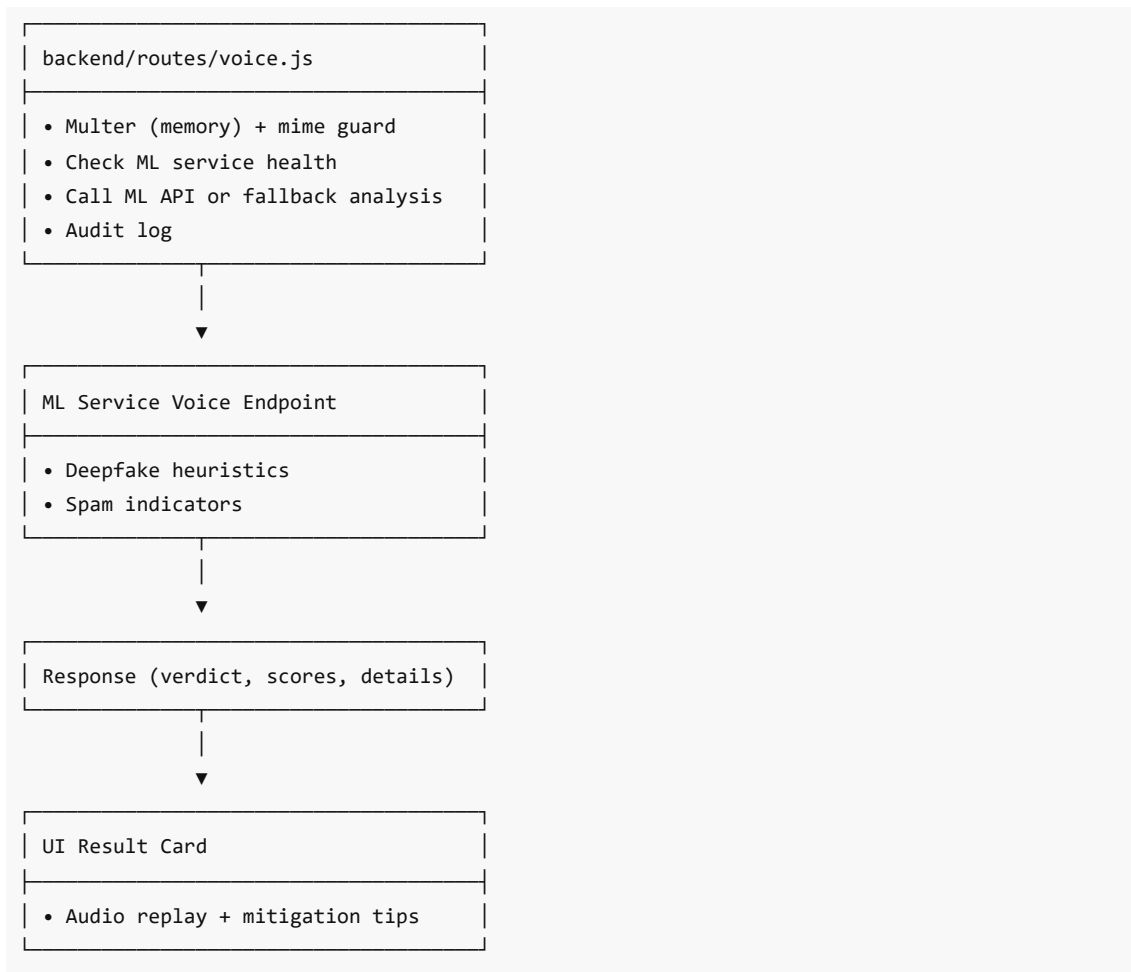
SMS Analyzer (`SMSChecker.jsx` → `/sms/analyze`)

SMS Input
(text + optional sender/phone)



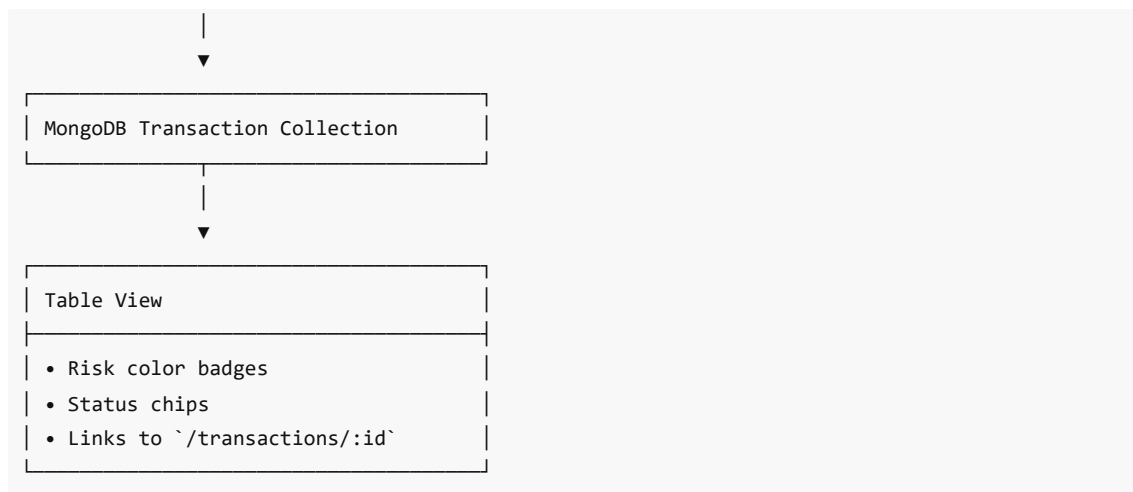
Voice Analyzer (VoiceDetector.jsx → /voice/detect)



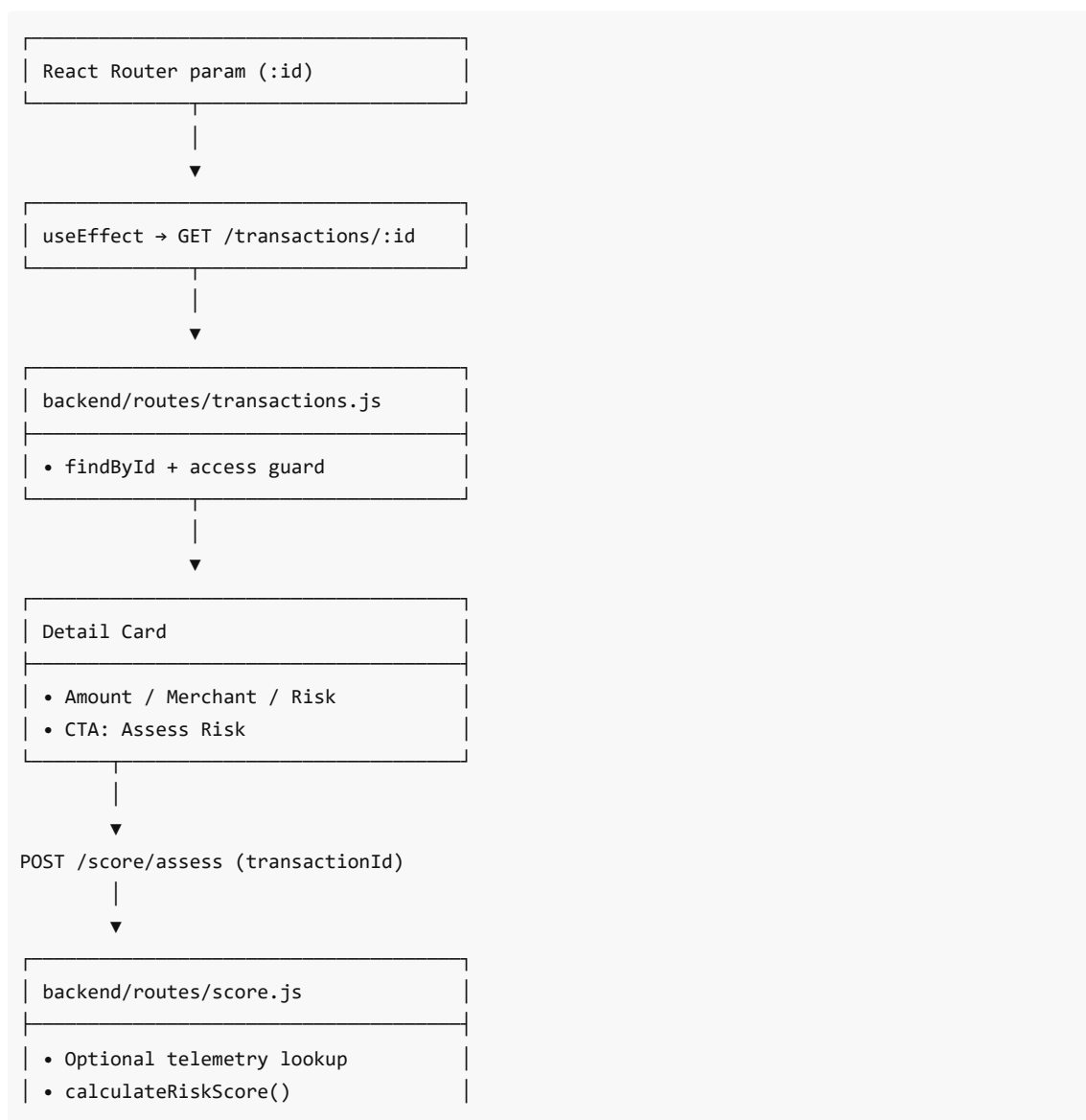


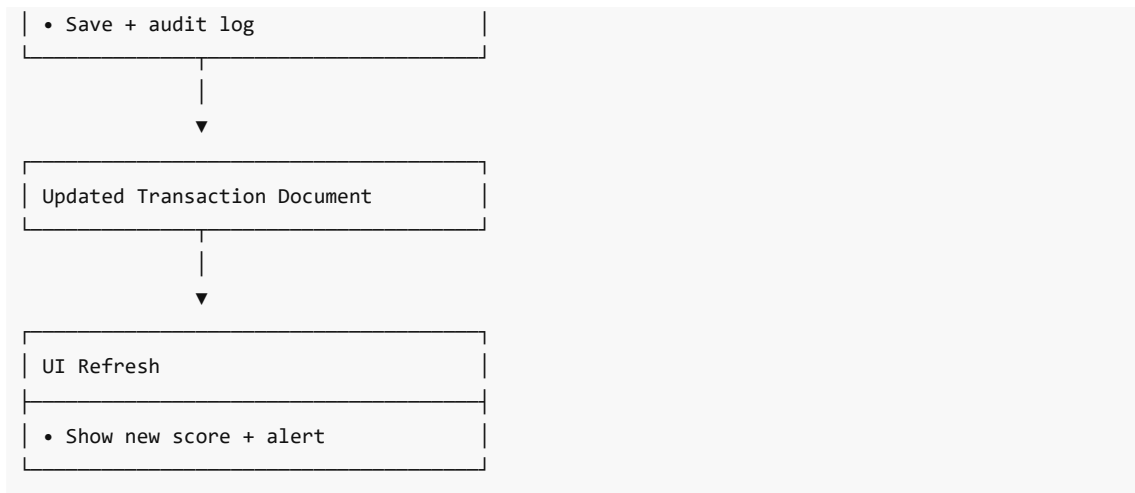
Transactions List (Transactions.jsx)



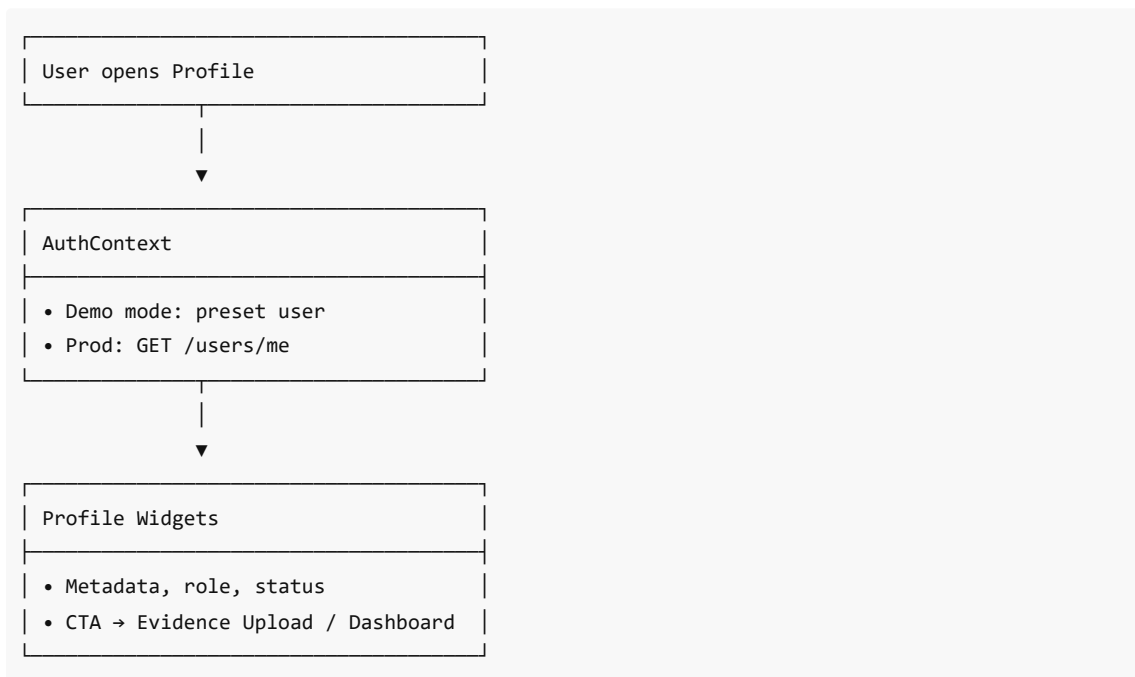


Transaction Detail (TransactionDetail.jsx)



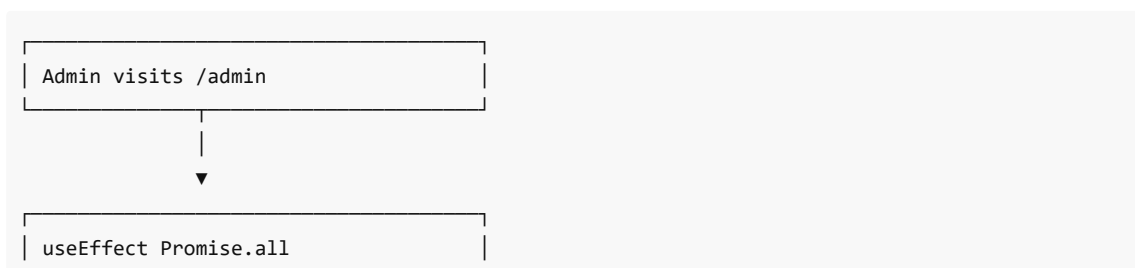


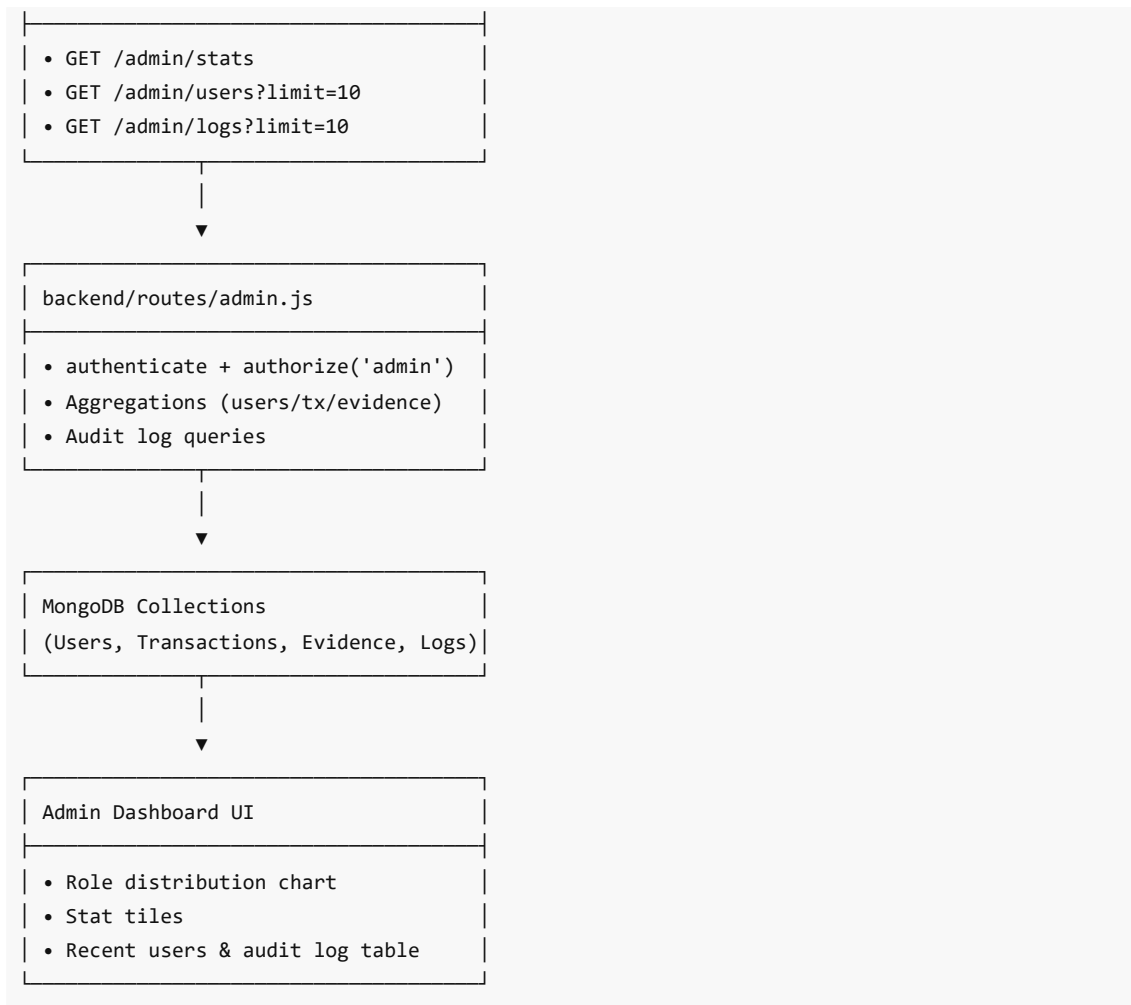
Profile (Profile.jsx)



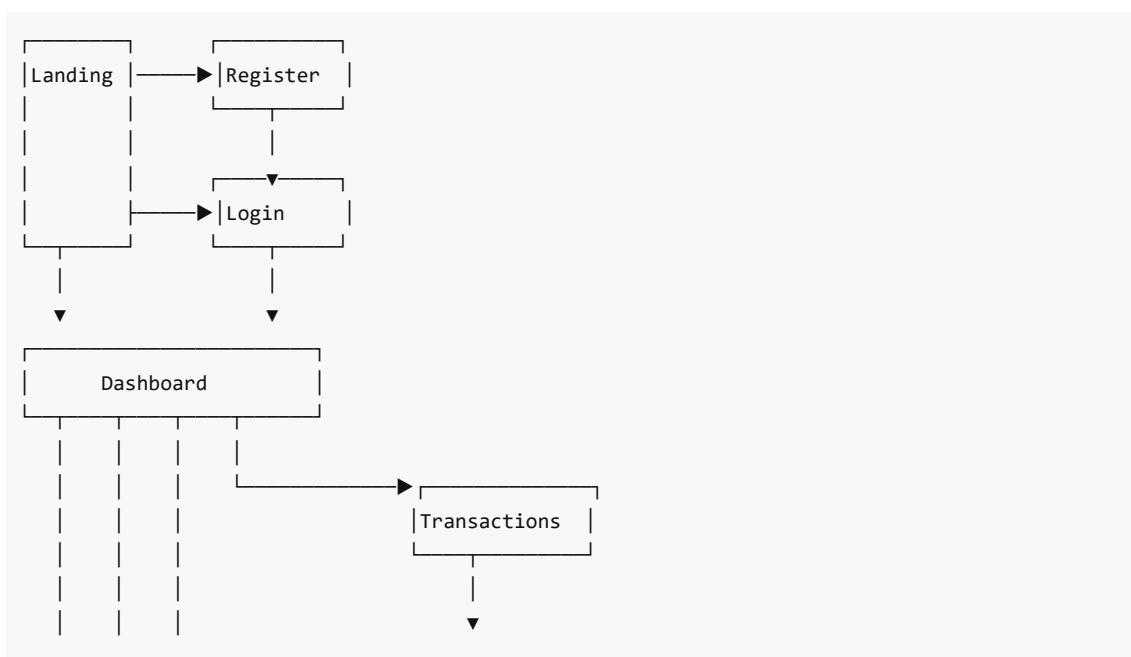
For editable deployments, `PUT /users/me` (in `backend/routes/users.js`) would mirror the update flow, logging audit entries.

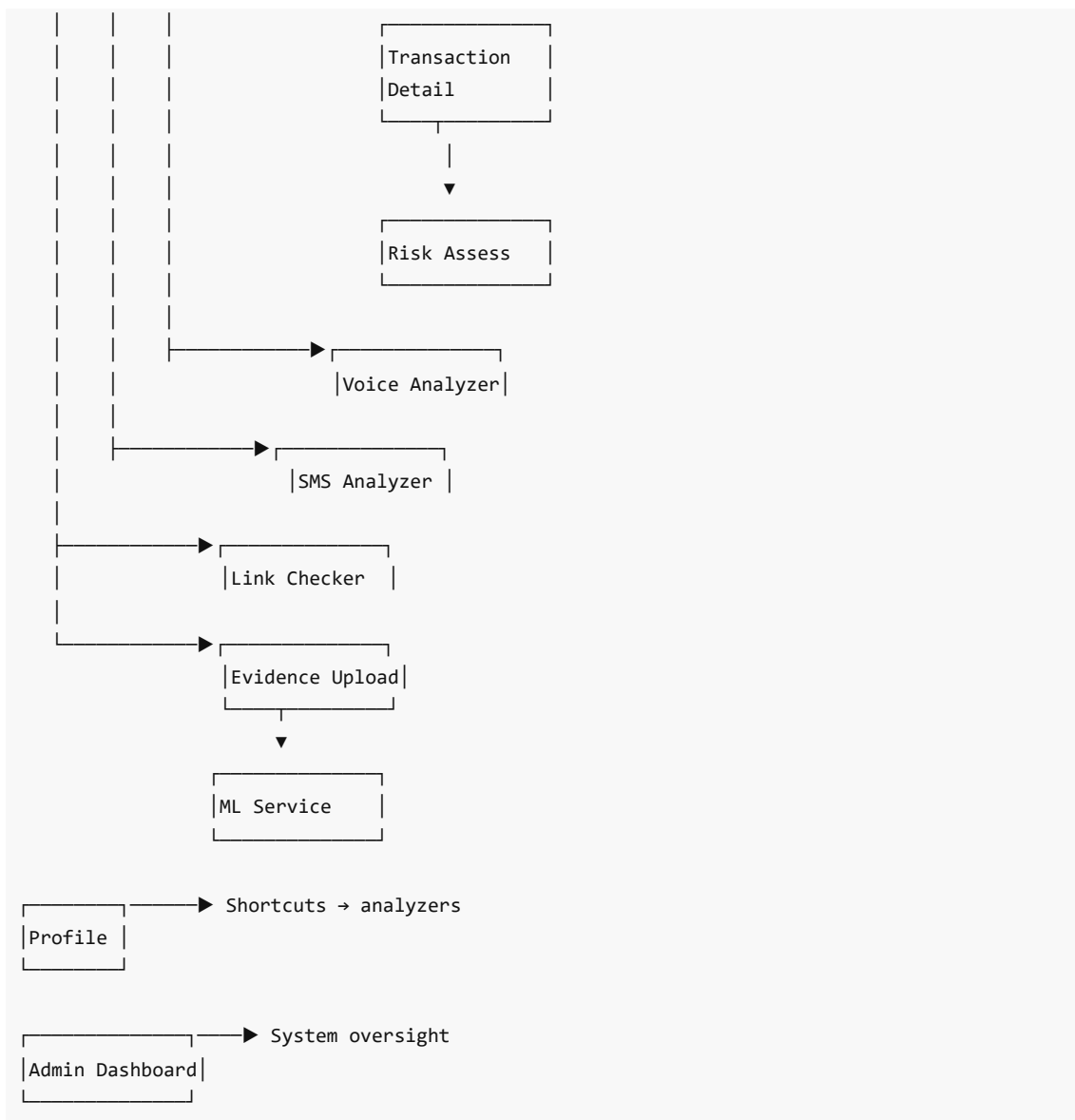
Admin Dashboard (AdminDashboard.jsx)





Page Interlinking Overview





Use this document as the canonical reference when updating slides or onboarding new contributors; every arrow references concrete files and endpoints, so you can quickly show how UI intentions are enforced by backend logic and persisted in MongoDB.