Secure UPI – End-to-End Workflow

SECURE UPI JOURNEY

From First Click to Fraud Verdict

A single, color-coded workflow showing the UI states, backend routes, and machine-learning services that power every feature—register, login, dashboard data, fraud analyzers, admin tools, and audit trails.

UI View · Backend Route / Service · Decision / Connector

1. Entry & Authentication

LANDING

CTA Hub

- Visitors choose **Register** or **Login**
- Existing session → auto-route to Dashboard

REGISTER

POST /auth/register

Collects name, email, password, phone and hits `backend/routes/auth.js`.

- bcrypt hashing
- Audit log entry
- MongoDB `User` insert

LOGIN

POST /auth/login

Validates credentials, issues JWT, records audit log.

AUTHCONTEXT

Persist Session

Stores token + user profile and shares across the routed layout.

2. Dashboard Bootstrap

LAYOUT / DASHBOARD

Token-Aware Fetch

Immediately loads stats via:

- `GET /transactions?limit=100`
- `GET /evidence?limit=100`

BACKEND

transactions.js / evidence.js

Each route authenticates, queries MongoDB collections, and returns:

- Recent transactions with risk scores
- Evidence verdicts and metadata

3. Feature Modules (Fan-out from Dashboard)

EVIDENCE UPLOAD

POST /evidence/upload

- FormData (image + manual details)
- `backend/routes/evidence.js` streams to ML service
- OCR, forgery, fraud indicators → MongoDB

LINK CHECKER

POST /links/check

- Regex heuristics + suspicious TLDs
- `verificationService` official whitelist
- Google Safe Browsing (optional)

SMS ANALYZER

POST /sms/analyze

Pattern buckets + verification lookups return fraud score, warnings, CTA to Link Checker.

VOICE ANALYZER

POST /voice/detect

- Multer (audio)
- Health check to ML service
- Deepfake/spam verdict + fallback analyzer

TRANSACTIONS LIST

GET /transactions

Shows paginated history with risk badges; each row links to detail view.

PROFILE

GET /users/me

Displays user metadata; demo mode uses static context.

ADMIN DASHBOARD

GET /admin/*

`authorize('admin')` protects routes for stats, users, audit logs.

4. ML & Storage Layer

SERVICES & PERSISTENCE

Always-on foundation

- **ML Service (Python/FastAPI):** OCR, forgery, deepfake, voice analysis, transaction risk.
- **MongoDB:** Users, Transactions, Evidence, Merchants, DeviceTelemetry, AuditLog.
- **Audit Logger:** `createAuditLog` invoked by auth, uploads, verifications, admin actions.

5. Transaction Drill-down & Risk Loop

> Detail View
>
> **GET /transactions/:id** → shows merchant, risk, timeline + button to reassess.

REASSESS CTA

POST /score/assess

Payload: transactionId (+ optional deviceId). Invokes `backend/routes/score.js`.

SCORE.JS

calculateRiskScore()

- Fetch Transaction + DeviceTelemetry
- Compute risk, update doc, add recommendation

DASHBOARD REFRESH

UI Update

New risk propagates to Dashboard KPIs, tables, and timeline instantly.

6. Journey Summary

1. **Visitors** arrive on a branded landing page and choose authentication flows.
2. **Auth routes** (`/auth/register` , `/auth/login`) validate, hash, log, and persist to MongoDB, returning JWTs consumed by `AuthContext` .
3. **Dashboard** fan-out fetches `/transactions` + `/evidence` to populate KPIs and build links into deep-dive modules.
4. **Feature modules** call their respective routes (`/evidence/upload` , `/links/check` , `/sms/analyze` , `/voice/detect` , `/transactions` , `/users/me` , `/admin/*`), each backed by Express routers, verification services, and ML endpoints.
5. **ML & database layers** supply OCR, deepfake, spam detection, and persistent evidence/risk audit trails.
6. **Transactions detail** loops into `/score/assess` , ensuring risk recalculations feed back into the dashboard ecosystem.

This colorized workflow mirrors a product handoff diagram, making it easy to drop into presentations or reports and still retain the precise backend touchpoints.***