FRD (Functional Requirements Document) for EasyInsure

**1. Introduction**

* Purpose: Replace manual processes with automated, data-driven workflows for claims, policies, and compliance.
* Scope:
  + In-scope: Policy lifecycle, AI claims, partner dashboards, self-service.
  + Out-of-scope: Physical branches, IoT driving alerts, reinsurance.

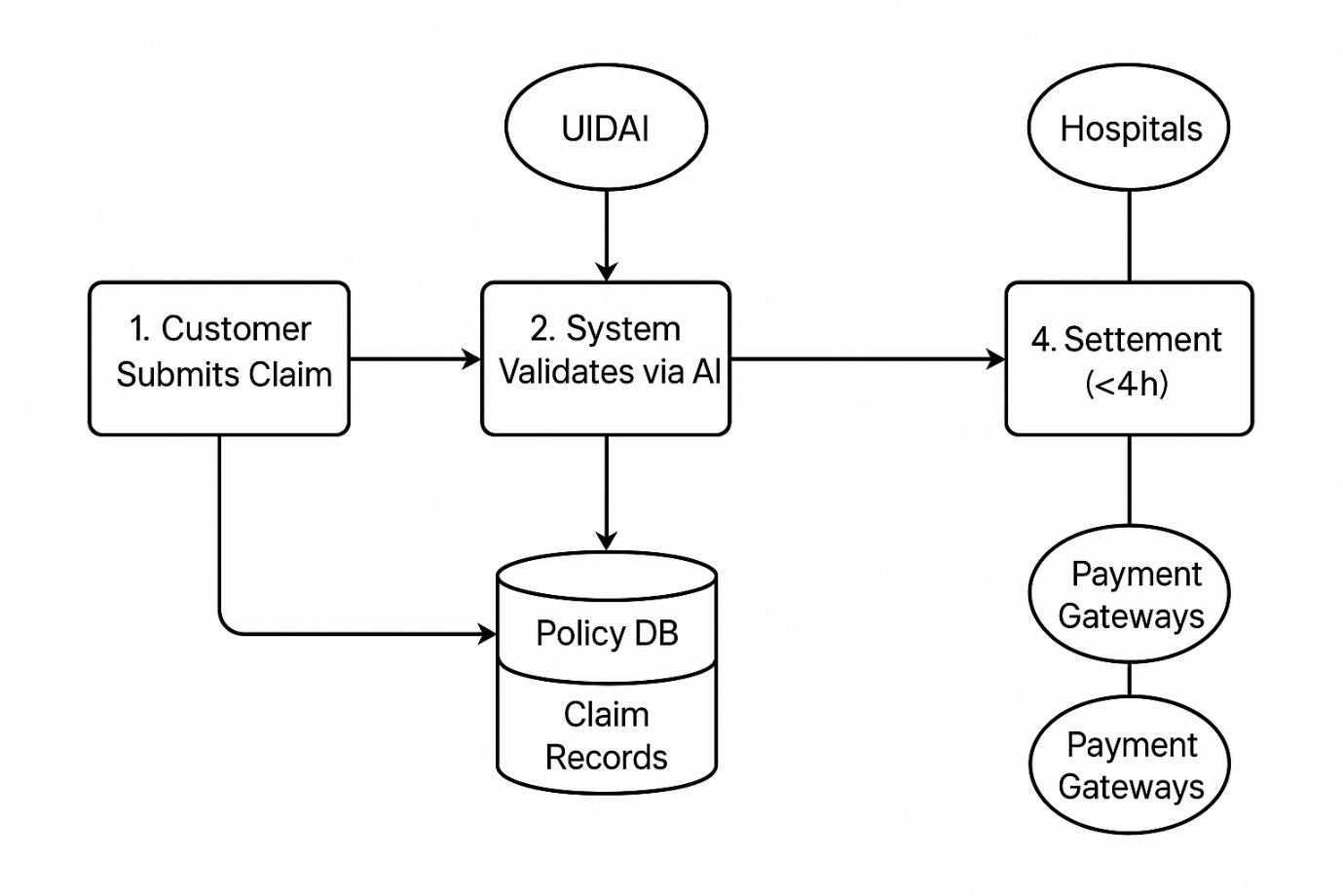
**2. System Overview**

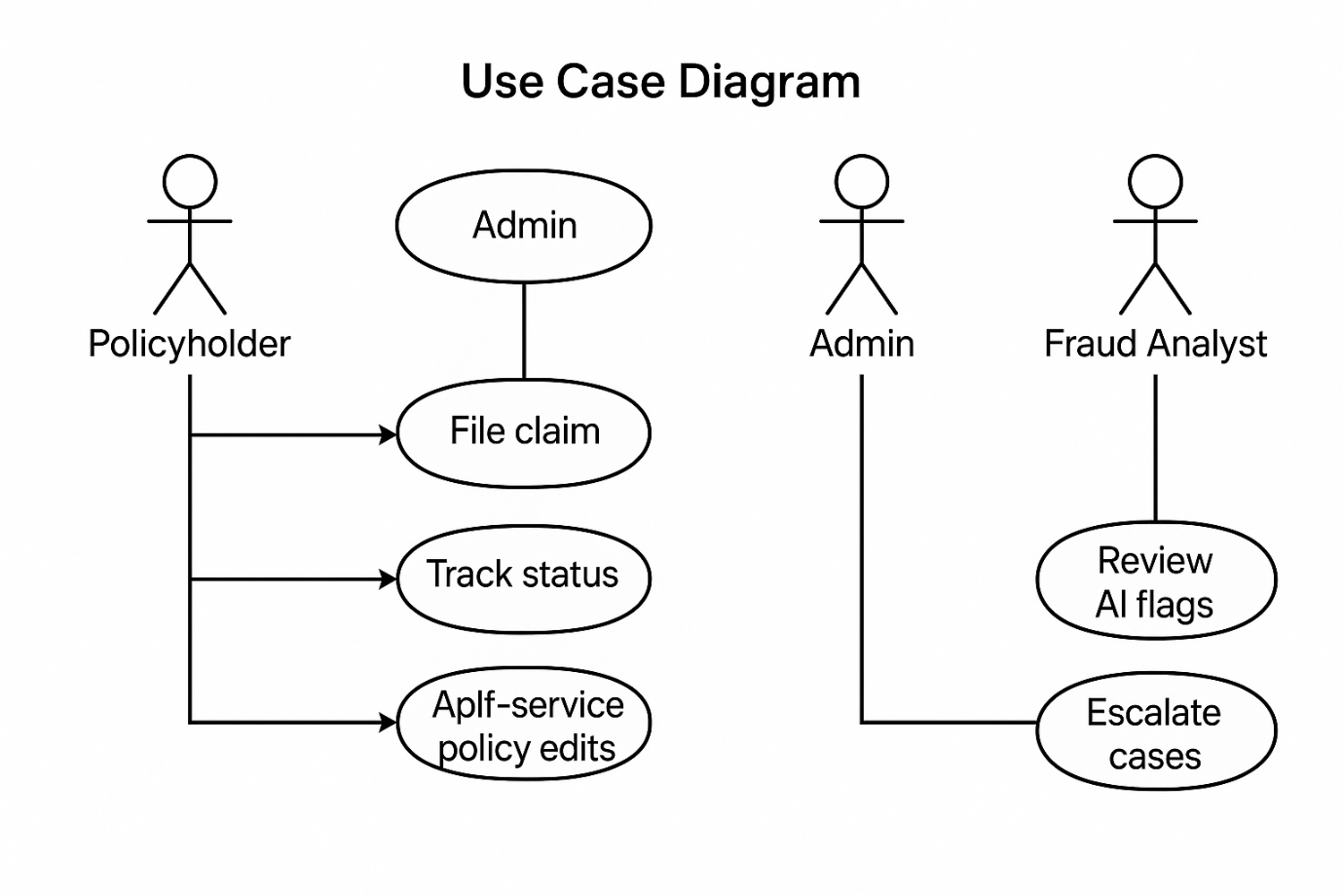
* System Context:
  + Actors: Policyholders, Admins, Garages, Hospitals, IRDAI.
  + Integrations: Aadhaar/UPI APIs, Google Maps, AWS Rekognition.
* Architecture:
  + Frontend: Mobile/Web app (React).
  + Backend: Microservices (Node.js/Python).
  + Database: Encrypted AWS RDS.

**3. Functional Requirements**

* Policy Management:
  + Dynamic quotes (<5 sec), Paytm/UPI auto-renewals, WhatsApp reminders.
* Claims Processing:
  + Mobile photo → OCR → Auto-form fill → Fraud check (image hashing).
* Admin Portal:
  + SLA dashboards, fraud heatmaps, TAT monitoring.

**4. Data Flow Diagram**

**5. Use Case Diagram**



* Policyholder: File claim, Track status, Self-service policy edits.
* Admin: Monitor TAT, Approve/reject claims, Manage partners.
* Fraud Analyst: Review AI flags, Escalate cases.

**6. Non-Functional Requirements**

* Performance: 99.9% uptime, 500K concurrent users.
* Security: AES-256 encryption, RBAC, PCI DSS compliance.
* Usability: Multilingual UI (Hindi, Tamil, English), intuitive dashboards.

**7. Acceptance Criteria**

* Claim submission: OCR auto-fills 90% fields in <10 sec.
* Fraud detection: Alerts for >90% image similarity matches.
* Payment: 100% settlements in <4h post-approval.