Directive Squad

Team Role Distribution (4 Members)

Goutham — Authentication & Routing

Tasks

- Setup **json-server-auth**

- Create `db.json` users

- Implement **Login & Register UI**

- Handle **JWT storage**

- Create **Auth Guard**

- Create **HTTP Interceptor**

- Role-based redirection (admin / agent / customer)

Poojitha — Policies & Agent Dashboard

Tasks

- Create **Policy Catalog UI**

- CRUD operations on policies

- Filter & search policies

- Policy details page

- Admin-only policy management

- JSON Server endpoints for policies

- Agent dashboard

Srineel — Customer dashboard & Claims

Tasks

- Customer dashboard
- Claim filing form
- Claim tracking page
- Claim CRUD (create/update status)
- Link customers to policies
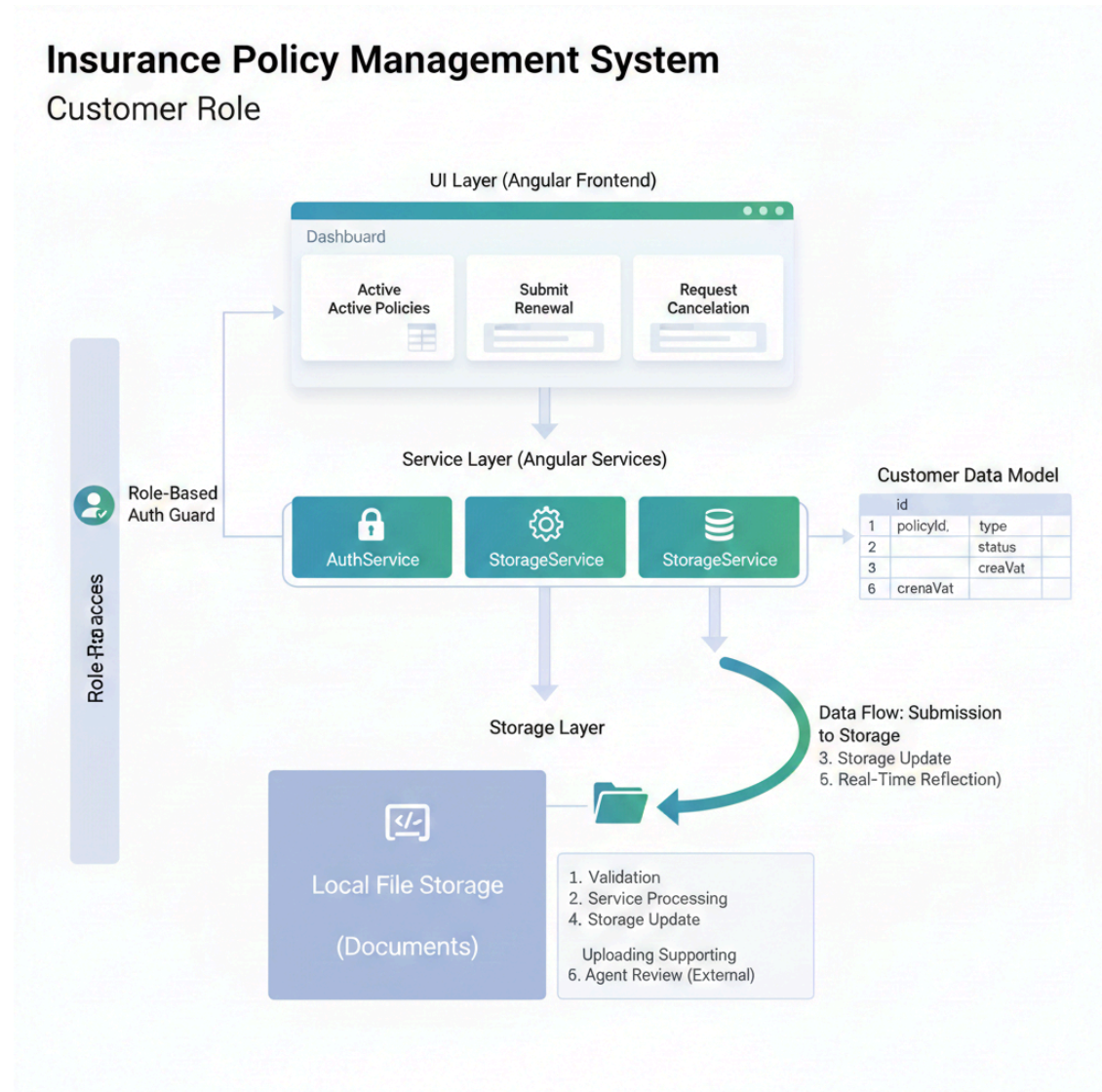- JSON Server claims APIs

Sanjay — Admin dashboard & Reports

Tasks

- Admin dashboard UI
- Show statistics (counts)
- Agent management
- Claims approval page
- Simple reports (tables/charts)
- Dashboard cards (Tailwind)

1. Customer Role Architecture

- **Purpose**: Designed for end-users to manage their personal insurance portfolios.
- **Core Features**: Enables viewing active policies, requesting renewals, and initiating cancellations.
- **Data Interaction**: Customers upload supporting documents and track request statuses in real time.
- **Workflow Ownership**: The customer is the primary owner of Renewal and Cancellation requests.
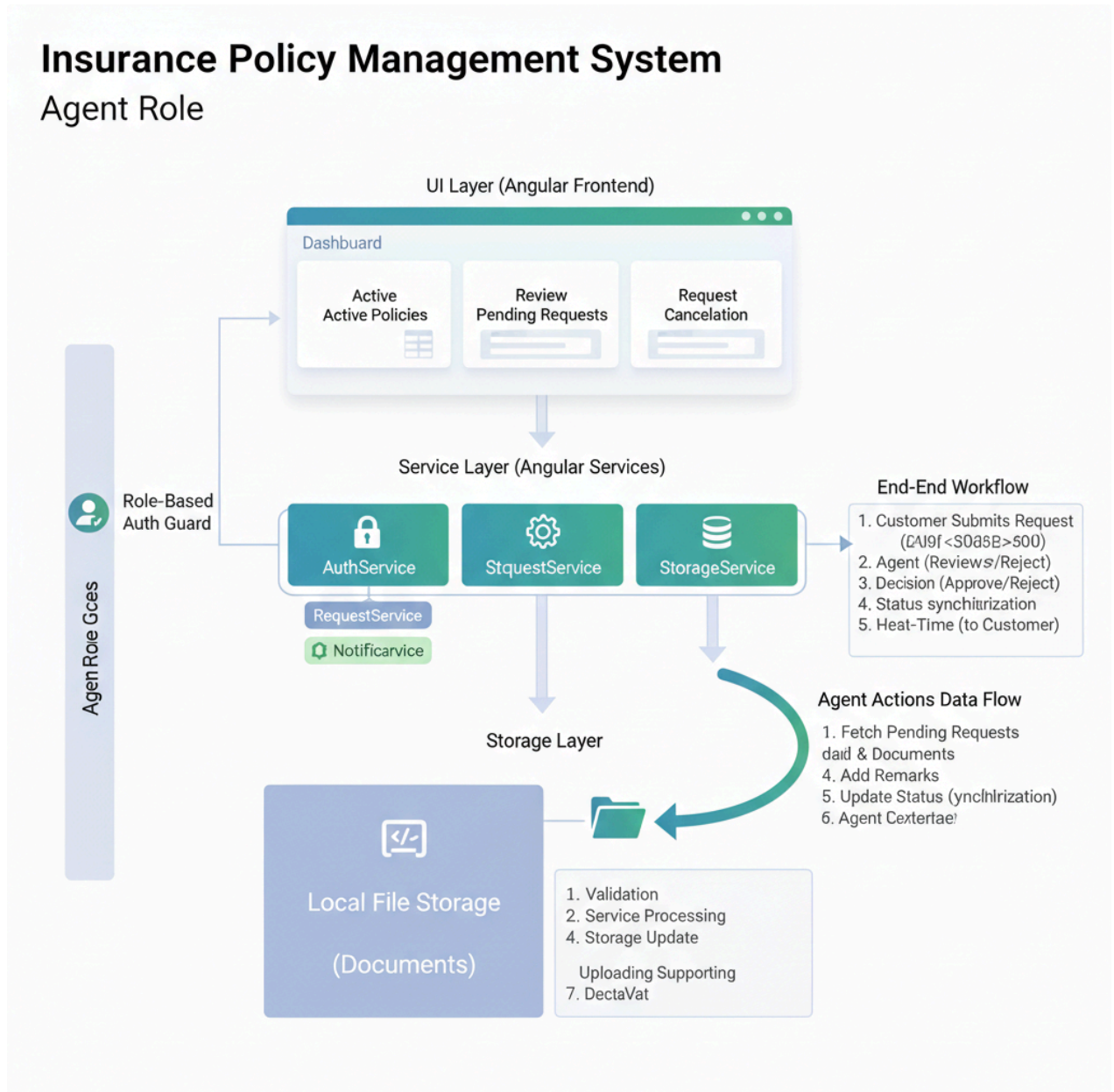

Insurance Policy Management System
Customer Role

2. Agent Role Architecture

- **Purpose**: Serves as the intermediate approver for customer-initiated workflows.
- **Core Features**: Provides tools to review incoming requests and approve or reject endorsements.
- **Data Interaction**: Agents add remarks or comments to requests, which are persisted in the service layer.
- **Workflow Ownership**: The agent owns the approval status and endorsement review process.

Insurance Policy Management System
Agent Role

UI Layer (Angular Frontend)

Dashbuard

Active
Active Policies

Review
Pending Requests

Request
Cancelation

Service Layer (Angular Services)

Role-Based
Auth Guard

AuthService

StquestService

StorageService

RequestService

Notificarvice

End-End Workflow
1. Customer Submits Request (ϹΛ9ſ<SOa5Ɓ>500)
2. Agent (Reviews/Reject)
3. Decision (Approve/Reject)
4. Status synchinurization
5. Heat-Time (to Customer)

Agen Roıe Gces

Storage Layer

Agent Actions Data Flow
1. Fetch Pending Requests daıd & Documents
4. Add Remarks
5. Update Status (ynclhlrization)
6. Agent Cextertae

Local File Storage

(Documents)

1. Validation
2. Service Processing
4. Storage Update

Uploading Supporting
7. DectaVat

3.Admin Role Architecture

- **Purpose**: Acts as the system owner with full control over policy management and user roles.
- **Core Features**: Handles the full CRUD (Create, Read, Update, Delete) lifecycle of insurance policies.
- **Data Interaction**: Manages the master policy records and user-role mappings stored in the storage layer.
- **Workflow Ownership**: Admins update master policies following approved endorsements and oversee all system requests.