Innovation partner H2S HACK2SKILL

Media partner YOURSTORY

AI for Bharat Hackathon

Powered by aws

Team Name : Empower Panchayat

Team Leader Name : Anshul Jain

Problem Statement : PS03 - AI for Rural Innovation & Sustainable Systems

Brief about the Idea:

WHAT

eGramSabha is an AI-powered platform by Empower Panchayat that digitises Gram Sabha governance -- democratizing AI to ensure marginalized rural citizens receive transparent, accountable local governance.

Aligned with Article 40 and the 73rd Constitutional Amendment, it addresses real challenges panchayats face: inadequate administrative support, insufficient training, and lack of digital infrastructure -- by bringing AI-driven competence building directly to the Gram Panchayat level.

It combines face recognition auth, multilingual voice input, LLM-powered meeting intelligence, and smart agenda creation across three role-based portals (Citizen, Official, Admin) -- strengthening community participation and enabling transparent resource administration.

INNOVATION

Configurable AI Provider Architecture

- Strategy Pattern lets deployments switch AI backends
- Supports multiple cloud & open-source AI services
- Change one env variable -- no code changes needed

Purpose-Built for Rural India

- Face auth for citizens who cannot type passwords
- Voice-first issue creation in native languages
- Low-bandwidth optimized, async AI processing

Open Source (Apache 2.0)

- Supports Digital India and e-Panchayat initiatives

Solution Overview:

How is it different?

- No existing platform combines face-auth + voice input + AI meeting intelligence
- Face recognition replaces passwords -- an accessibility breakthrough for low-literacy users
- Multiple STT providers with configurable fallback options
- LLM transcription uses expert rural development prompts for domain accuracy

How does it solve the problem?

- Citizens speak issues in native language -- AI transcribes, corrects, translates automatically
- AI clusters similar issues into structured multilingual agendas (significant effort reduction)
- Meeting recordings become formal Minutes in multiple languages simultaneously
- Complete lifecycle: Reported → Agenda → Discussed → Resolved

USP

- Configurable AI providers -- switch backends with one env variable
- Purpose-built for rural India: face auth, voice-first, multilingual
- Open-source under Apache 2.0 license, cloud-agnostic deployment

Key Features:

Face Recognition Auth

- Client-side biometric face recognition
- Liveness detection (anti-spoofing)
- High-accuracy face matching engine

Multilingual Voice Input

- Citizens speak in native language
- Multiple STT provider support
- LLM correction with domain prompts

AI Meeting Intelligence

- Auto MOM in multiple languages from audio
- Smart agenda clusters similar issues
- Async processing with scheduled pipelines

Three Role-Based Portals

- Citizen: face login, voice issues, RSVP
- Official: meetings, agenda, MOM, export
- Admin: onboarding, bulk import, config

Configurable AI Providers

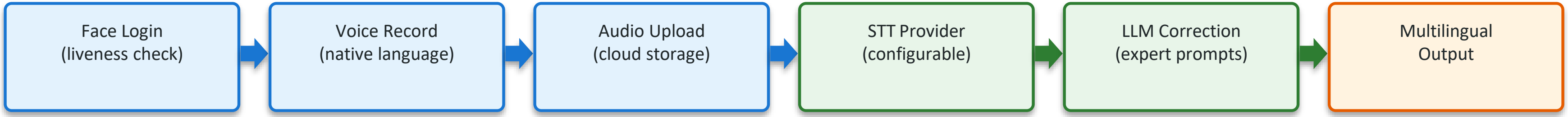
- Strategy Pattern + Factory design
- LLM: Configurable cloud providers
- STT: Configurable speech providers

Complete Governance Workflow

- Issue lifecycle tracking end-to-end
- Role-based access control with data isolation
- PDF/CSV export with Hindi + letterhead

Process Flow -- AI Pipeline:

CITIZEN ISSUE CREATION FLOW



Face Login (liveness check)

Voice Record (native language)

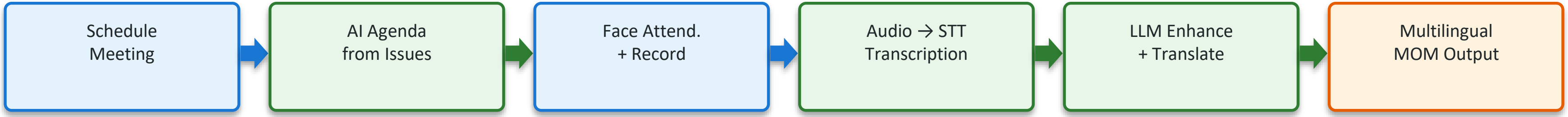
Audio Upload (cloud storage)

STT Provider (configurable)

LLM Correction (expert prompts)

Multilingual Output

MEETING INTELLIGENCE FLOW



Schedule Meeting

AI Agenda from Issues

Face Attend. + Record

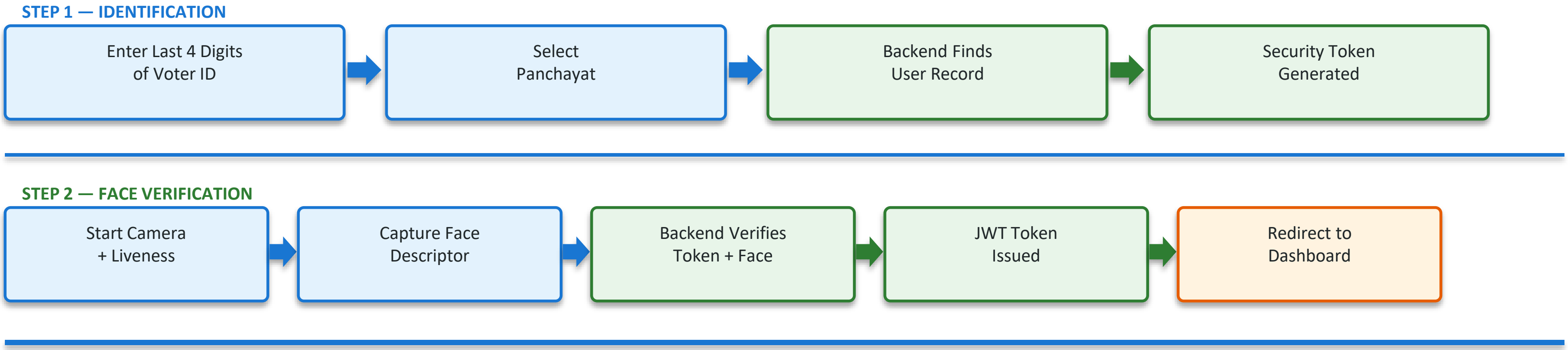
Audio → STT Transcription

LLM Enhance + Translate

Multilingual MOM Output

Async Processing Architecture: All AI operations run asynchronously with database-backed state tracking. Each request returns immediately with a request_id and polling URLs. The pipeline supports: progress checkpoints at defined milestones • automatic retry with configurable attempts • resume from last successful stage • TTL-based cleanup • Scheduled jobs orchestrate the full pipeline (transcription polling, agenda generation, translation at configurable intervals).

Citizen Authentication Flow:



STEP 1 — IDENTIFICATION

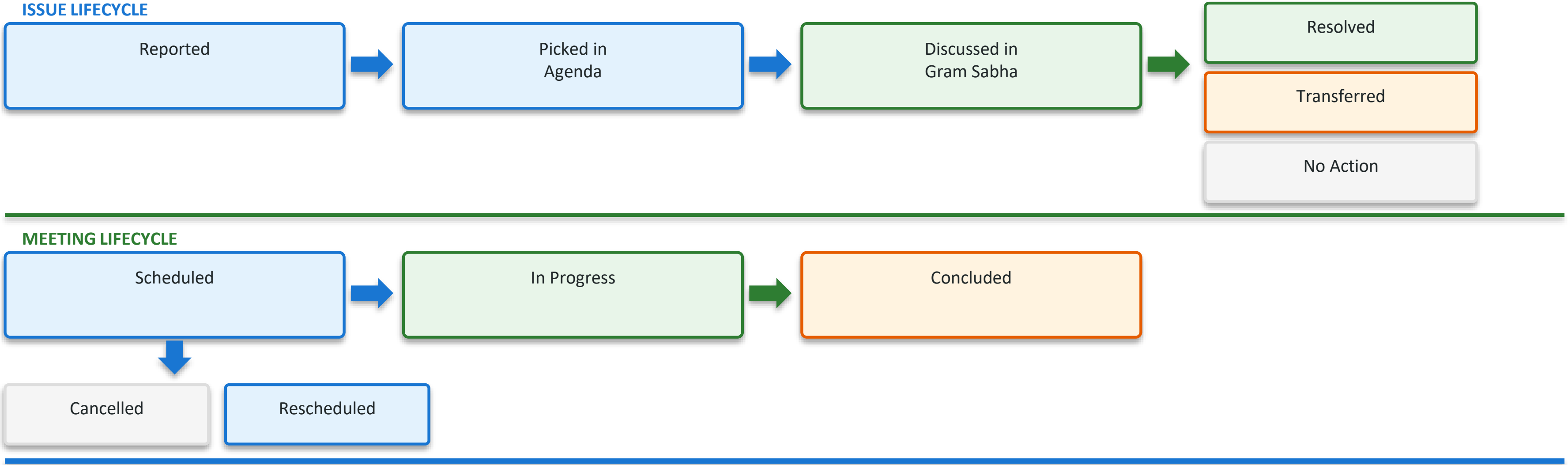
Enter Last 4 Digits of Voter ID → Select Panchayat → Backend Finds User Record → Security Token Generated

STEP 2 — FACE VERIFICATION

Start Camera + Liveness → Capture Face Descriptor → Backend Verifies Token + Face → JWT Token Issued → Redirect to Dashboard

Security: No face images stored on server -- only mathematical descriptors for privacy. Liveness detection (blink + head movement) prevents photo/video spoofing. Time-limited security tokens with separate JWT secrets per user type. All communication over HTTPS with encrypted data in transit.

Issue & Meeting Lifecycles:



ISSUE LIFECYCLE

Reported → Picked in Agenda → Discussed in Gram Sabha → Resolved / Transferred / No Action

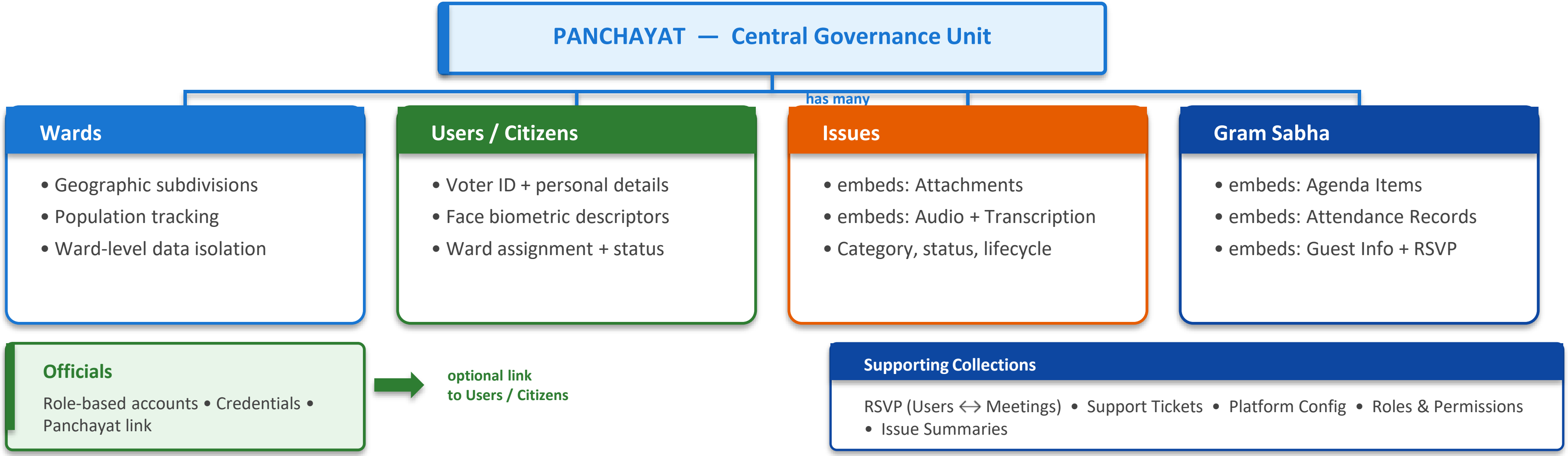
MEETING LIFECYCLE

Scheduled → In Progress → Concluded

Scheduled ↓ Cancelled / Rescheduled

Issue Categories: Culture & Nature • Infrastructure • Earning Opportunities • Basic Amenities • Social Welfare Schemes • Other | Meeting Features: RSVP management • Face-based attendance • Quorum tracking • AI agenda generation • Multilingual MOM export

Data Architecture & Relationships:



PANCHAYAT — Central Governance Unit

has many

Wards
- Geographic subdivisions
- Population tracking
- Ward-level data isolation

Users / Citizens
- Voter ID + personal details
- Face biometric descriptors
- Ward assignment + status

Issues
- embeds: Attachments
- embeds: Audio + Transcription
- Category, status, lifecycle

Gram Sabha
- embeds: Agenda Items
- embeds: Attendance Records
- embeds: Guest Info + RSVP

Officials

Role-based accounts • Credentials • Panchayat link

optional link
to Users / Citizens

Supporting Collections

RSVP (Users ↔ Meetings) • Support Tickets • Platform Config • Roles & Permissions • Issue Summaries

Storage: MongoDB document database with file storage for binary data (face images, audio recordings, letterheads). Separate async tracking database for AI request lifecycle. Compound + text indexes for optimized query patterns.

Application Portals:

Citizen Portal

- **Face Login:** liveness detection verified
- **Voice Issues:** record audio in native language
- **Issue Tracking:** real-time status dashboard
- **Meeting RSVP:** confirm attendance + history
- **Multilingual:** bilingual UI (English + Hindi)

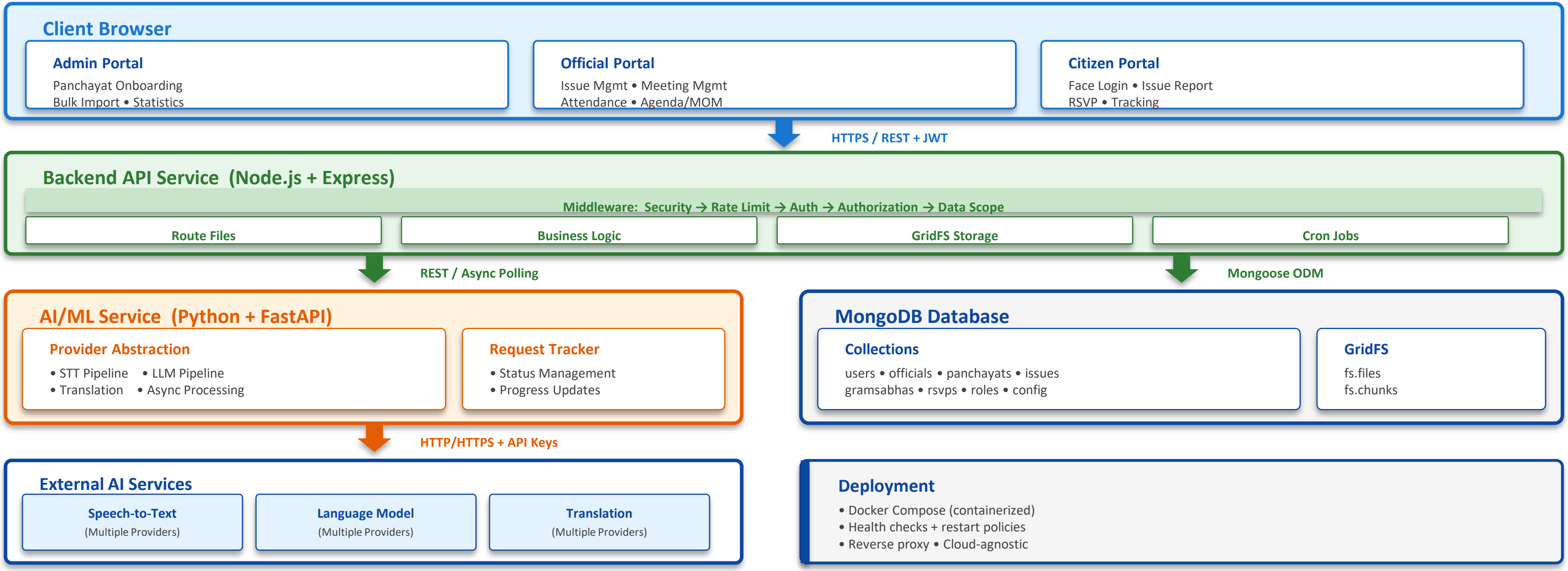
Official Portal

- **Issue Mgmt:** status workflow Reported → Resolved
- **Meeting Scheduler:** RSVP tracking + quorum mgmt
- **AI Agenda:** auto-generated from citizen issues
- **AI MOM:** multilingual minutes from recordings
- **Export:** PDF/CSV with letterhead + Hindi fonts

Admin Portal

- **Onboarding:** panchayat setup with LGD codes + wards
- **Bulk Import:** CSV citizen registration + dedup
- **Officials:** manage accounts + RBAC role assignment
- **Config:** face thresholds, liveness, platform settings
- **Analytics:** system-wide registration + usage statistics

System Architecture:



Client Browser

Admin Portal
Panchayat Onboarding
Bulk Import • Statistics

Official Portal
Issue Mgmt • Meeting Mgmt
Attendance • Agenda/MOM

Citizen Portal
Face Login • Issue Report
RSVP • Tracking

HTTPS / REST + JWT

Backend API Service (Node.js + Express)

Middleware: Security → Rate Limit → Auth → Authorization → Data Scope

Route Files | Business Logic | GridFS Storage | Cron Jobs

REST / Async Polling

Mongoose ODM

AI/ML Service (Python + FastAPI)

Provider Abstraction
- STT Pipeline
- LLM Pipeline
- Translation
- Async Processing

Request Tracker
- Status Management
- Progress Updates

MongoDB Database

Collections
users • officials • panchayats • issues
gramsabhas • rsvps • roles • config

GridFS
fs.files
fs.chunks

HTTP/HTTPS + API Keys

External AI Services

Speech-to-Text
(Multiple Providers)

Language Model
(Multiple Providers)

Translation
(Multiple Providers)

Deployment

- Docker Compose (containerized)
- Health checks + restart policies
- Reverse proxy • Cloud-agnostic

Technologies Used:

Frontend

- React + Material-UI (MUI)
- Face recognition engine
- PDF generation, internationalization
- Client-side biometric processing
- Bilingual UI (English + Hindi)

Backend

- Node.js + Express + MongoDB ODM
- MongoDB with file storage
- JWT authentication + secure hashing
- Auto-generated API documentation
- Scheduled job processing

AI / ML Service

- Python + FastAPI
- Async MongoDB driver
- Multiple STT providers (Indian languages)
- Multiple LLM providers (configurable)
- Audio processing pipeline

Cloud Services + Infra

- Cloud LLM APIs (configurable provider)
- Cloud STT services (Indian languages)
- Cloud translation services (optional)
- Docker Compose + Reverse Proxy
- Cloud SDKs for provider integration

Estimated Implementation Cost:

eGramSabha is fully open-source (Apache 2.0) -- Zero licensing cost

Infrastructure

- Cloud VM instance for hosting
- MongoDB free tier available for pilot
- Docker deployment minimizes ops cost

AI Services (Pay-per-use)

- Open-source models with free tiers
- Regional STT provider programs
- Cloud LLM: pay-per-use token pricing
- Cloud STT: pay-per-use audio pricing

Cost Optimization

- Switch providers via env variable
- Free-tier for pilots, cloud at scale
- Mix-and-match STT + LLM providers

Target: Serve panchayats across India with infrastructure that adapts to budget constraints through configurable AI provider selection.

Impact, Scalability & Innovation:

Rural Impact

- Gram Panchayats across India
- Millions of rural citizen beneficiaries
- Face auth removes literacy barrier
- Voice-first in native languages
- Supports Digital India initiative

Scalability

- Stateless services scale horizontally
- Cloud-agnostic, runs on any platform
- Separate DB for transactional + AI
- Configurable cloud providers
- Database replica set + sharding ready

AI Innovation

- Multiple STT provider pipeline
- Expert rural development LLM prompts
- Smart issue-to-agenda clustering
- Multilingual MOM from single recording
- Strategy + Factory provider pattern

Innovation partner H2S HACK2SKILL | Media partner YOURSTORY

AI for Bharat Hackathon

Powered by aws

Thank You