## “JalSanchay” - AI-Powered Rainwater Harvesting Assistant

### Core Innovation Stack

## 1. Technical Architecture

### Frontend Technologies

Mobile App:  
- Flutter (Cross-platform iOS/Android)  
- React Native Web for PWA  
- Offline-first architecture with SQLite  
  
Web Application:  
- Next.js 15 with App Router  
- Tailwind CSS + Shadcn UI  
- Progressive Web App (PWA) capabilities

### Backend Infrastructure

Microservices Architecture:  
- Node.js with Express/Fastify  
- Python FastAPI for ML services  
- GraphQL Gateway (Apollo Server)  
- Redis for caching  
- PostgreSQL with PostGIS extension

### AI/ML Pipeline

- TensorFlow/PyTorch for predictive models  
- Computer Vision for roof area detection  
- XGBoost for cost-benefit analysis  
- Prophet for rainfall prediction  
- Scikit-learn for feasibility scoring

### GIS & Mapping

- Google Earth Engine API  
- OpenStreetMap integration  
- Mapbox for visualization  
- QGIS Server for spatial analysis  
- Satellite imagery processing (Sentinel-2)

## 2. Key Differentiating Features

### A. AI-Powered Roof Detection

# Automatic roof area calculation using satellite imagery  
- Upload address → Fetch satellite image  
- CNN model detects roof boundaries  
- Calculate exact area and slope  
- Identify roof material type

### B. Smart Recommendation Engine

Features:  
- Multi-criteria decision analysis (MCDA)  
- Personalized structure recommendations  
- Budget-optimized solutions  
- Maintenance schedule generator  
- ROI calculator with payback period

### C. Real-time Data Integration

- IMD weather API integration  
- CGWB groundwater database  
- Live rainfall monitoring  
- Soil permeability data  
- Local water table information

### D. Gamification & Community Features

- Water warrior leaderboard  
- Achievement badges  
- Community water savings tracker  
- Neighborhood challenges  
- Social sharing of impact

## 3. Implementation Roadmap

### Phase 1: Core MVP (Week 1-2)

Features:  
 - Basic assessment calculator  
 - Location-based rainfall data  
 - Simple recommendation engine  
 - Cost estimation module  
   
Tech Stack:  
 - React PWA  
 - Node.js backend  
 - PostgreSQL database  
 - Basic ML models

### Phase 2: Advanced Features (Week 3-4)

Features:  
 - AI roof detection  
 - AR visualization  
 - Community features  
 - Multi-language support  
   
Additions:  
 - Computer vision integration  
 - AR.js implementation  
 - Firebase for real-time features  
 - i18n localization

### Phase 3: Scale & Polish (Week 5-6)

Features:  
 - Government dashboard  
 - Analytics portal  
 - API marketplace  
 - Contractor marketplace  
   
Infrastructure:  
 - Kubernetes deployment  
 - Auto-scaling setup  
 - CDN integration  
 - Performance optimization

## 4. Unique Selling Points for SIH

### A. Augmented Reality Visualization

// AR feature to visualize RTRWH structures  
- Point camera at roof/ground  
- See 3D model of recommended structure  
- Interactive placement and sizing  
- Real-time cost updates

### B. Blockchain-based Impact Tracking

// Smart contracts for:  
- Verified water savings certificates  
- Carbon credit calculation  
- Transparent fund allocation  
- Community incentive distribution

### C. IoT Integration Ready

# Smart monitoring system  
- Water level sensors integration  
- Quality monitoring devices  
- Automated alerts  
- Predictive maintenance

### D. Government Dashboard

interface AdminDashboard {  
 - District-wise adoption metrics  
 - Total water conserved  
 - Investment analysis  
 - Policy impact assessment  
 - Subsidy distribution tracking  
}

## 5. Database Schema Design

-- Core tables structure  
CREATE TABLE users (  
 id UUID PRIMARY KEY,  
 location GEOGRAPHY(POINT, 4326),  
 roof\_area DECIMAL,  
 household\_size INTEGER  
);  
  
CREATE TABLE assessments (  
 id UUID PRIMARY KEY,  
 user\_id UUID REFERENCES users(id),  
 feasibility\_score DECIMAL,  
 recommended\_structures JSONB,  
 cost\_estimate DECIMAL,  
 water\_potential DECIMAL  
);  
  
CREATE TABLE rainfall\_data (  
 location GEOGRAPHY(POINT, 4326),  
 date DATE,  
 precipitation DECIMAL,  
 PRIMARY KEY (location, date)  
);

## 6. API Architecture

REST APIs:  
 /api/v1/assessment:  
 POST: Create new assessment  
 GET: Retrieve assessment results  
   
 /api/v1/rainfall/{location}:  
 GET: Historical rainfall data  
   
 /api/v1/structures:  
 GET: Available RTRWH structures  
   
 /api/v1/calculate:  
 POST: Calculate harvesting potential  
  
GraphQL Schema:  
 type Assessment {  
 id: ID!  
 feasibilityScore: Float!  
 waterPotential: Float!  
 recommendations: [Structure!]!  
 costBenefit: CostAnalysis!  
 }

## 7. Scalability Strategy

### Technical Scalability

Infrastructure:  
 - Microservices with Docker containers  
 - Kubernetes orchestration  
 - Horizontal pod autoscaling  
 - Database sharding by region  
 - Redis clustering for caching  
   
Performance:  
 - CDN for static assets (CloudFlare)  
 - Image optimization pipeline  
 - Lazy loading implementation  
 - Service worker caching  
 - GraphQL query optimization

### Business Scalability

- White-label solution for states  
- API licensing for third parties  
- Integration with Jal Jeevan Mission  
- Partnership with NGOs  
- Corporate CSR integration

## 8. Monetization & Sustainability

Revenue Streams = {  
 1. "Freemium Model": "Basic assessment free, premium features paid",  
 2. "Government Contracts": "B2G licensing to state governments",  
 3. "API Access": "Paid API for developers and businesses",  
 4. "Contractor Marketplace": "Commission on contractor connections",  
 5. "Premium Analytics": "Advanced insights for corporations",  
 6. "Consultation Services": "Expert advice and implementation support"  
}

## 9. Impact Metrics & Analytics

impact\_metrics = {  
 "water\_saved": "Million liters tracked",  
 "co2\_reduced": "Carbon footprint reduction",  
 "money\_saved": "Household savings calculator",  
 "adoption\_rate": "District-wise penetration",  
 "job\_creation": "Local contractor employment",  
 "groundwater\_improvement": "Water table rise tracking"  
}

## 10. Security & Compliance

Security Features:  
 - OAuth 2.0 authentication  
 - JWT token management  
 - Data encryption at rest (AES-256)  
 - HTTPS enforcement  
 - Rate limiting  
 - Input validation & sanitization  
   
Compliance:  
 - GDPR compliant  
 - India's Data Protection Bill ready  
 - ISO 27001 standards  
 - WCAG 2.1 accessibility

## 11. Deployment Strategy

# CI/CD Pipeline  
GitHub Actions → Build → Test → Docker → Deploy  
  
# Infrastructure as Code  
Terraform for AWS/Azure/GCP deployment  
Ansible for configuration management  
  
# Monitoring  
Prometheus + Grafana for metrics  
ELK stack for logging  
Sentry for error tracking

## 12. Demo Scenario for Hackathon

### Live Demo Flow:

1. **Problem Statement** - Water crisis visualization
2. **User Journey** - Simple 3-step assessment
3. **AI Magic** - Live roof detection demo
4. **AR Wow Factor** - Visualize structure on stage
5. **Impact Dashboard** - Show potential national impact
6. **Scalability Proof** - Load testing demonstration

## 13. Team Composition Recommendation

1. Full-Stack Developer (Team Lead)  
2. ML/AI Engineer (Computer Vision)  
3. Mobile Developer (Flutter/React Native)  
4. GIS Specialist (Mapping & Spatial Analysis)  
5. UI/UX Designer (User Experience)  
6. Domain Expert (Water Conservation)

## 14. Winning Pitch Elements

### Opening Hook:

“Every year, India receives 4,000 BCM of rainfall, but we capture less than 8%. Our solution can help capture 20% just from rooftops!”

### Problem-Solution Fit:

* Clear problem identification
* Quantifiable impact
* Scalable solution
* Government alignment

### Technical Excellence:

* Live demos work flawlessly
* Code quality and documentation
* Open-source contributions
* API documentation ready

### Business Viability:

* Clear revenue model
* Government integration plan
* Sustainability roadmap
* Partnership strategy

## Repository Structure

JalSanchay/  
├── frontend/  
│ ├── web/  
│ ├── mobile/  
│ └── ar-module/  
├── backend/  
│ ├── api-gateway/  
│ ├── assessment-service/  
│ ├── ml-service/  
│ └── gis-service/  
├── ml-models/  
│ ├── roof-detection/  
│ ├── rainfall-prediction/  
│ └── feasibility-scoring/  
├── infrastructure/  
│ ├── docker/  
│ ├── kubernetes/  
│ └── terraform/  
├── docs/  
│ ├── API.md  
│ ├── DEPLOYMENT.md  
│ └── USER\_GUIDE.md  
└── README.md

This comprehensive solution addresses all aspects of the problem statement while incorporating cutting-edge technologies and innovative features that will make your team stand out in the SIH hackathon. The combination of practical utility, technical excellence, and social impact makes this a winning proposition.