

# CleanWaterAI - Predicting Water Contamination Risk in Kenyan Water Points - Product Requirements Document (PRD)

---

## 1. Overview

Millions of people in rural communities globally, particularly in regions like Kenya, lack access to safe drinking water. Traditional water quality testing is reactive, expensive, and logistically limited. The goal of this project is to build a proactive, AI-powered prediction system that assesses water contamination risk using multi-modal data. The output is surfaced via an interactive dashboard for stakeholders like NGOs, water authorities, and local governments to drive early intervention and improve public health.

---

## 2. Goals and Objectives

### 2.1 Core Objectives (Expanded)

#### 1. **Build a Unified, AI-Driven Contamination Risk Prediction System:**

- Leverage chemical (GEMS), environmental (GEE + WPDx), and unstructured textual data to provide early warning signals of water contamination.
- Enable automated risk classification (Safe/Unsafe) for any water point in the region.

#### 2. **Empower Local Decision-Makers with Actionable Insights:**

- Provide interactive geospatial visualizations, model explanations, and intervention recommendations through a Streamlit dashboard.
- Allow authorities and NGOs to prioritize field visits, testing, chlorination, and education campaigns.

#### 3. **Fuse Multiple Data Streams to Improve Model Robustness:**

- Combine structured environmental and chemical data with contextual field reports.
- Ensure high-coverage predictions even when one data source is sparse or missing.

#### 4. **Deliver Transparency and Trust in AI Models:**

- Use interpretability tools (SHAP, LIME, PDP) to visualize why a water point is flagged as unsafe.
- Make feature importance clear to domain experts and non-technical stakeholders.

#### 5. **Automate Monitoring and Model Improvement:**

- Enable periodic updates using live data pipelines (e.g., CHIRPS rainfall or new WPDx reports).
- Allow model retraining using fresh contamination labels as new ground truth data is collected.

Each objective is directly tied to a public health need, ensuring that the system is not just technically sound but also impactful in real-world scenarios.

---

## 3. Data Sources

### 3.1 GEMS (Chemical Water Quality Data)

- Source: GEMStat (<https://gemstat.org/data-gemstat/global-water-quality/>)
- Variables:
  - **Dissolved Oxygen (DO)**
  - **Nitrogen (Nitrate, Nitrite, Ammonia)**
  - **Phosphorus**
  - **pH**
- Domain Understanding:
  - Low DO values indicate insufficient oxygen, harmful to ecosystems and indicative of contamination.
  - High nitrogen and phosphorus levels signify fertilizer runoff or sewage, contributing to eutrophication.
  - pH extremes (outside 6.5-8.5) indicate corrosiveness or biological unsuitability.
- Feature Engineering:
  - `do_risk = 1 if DO < 4 mg/L else 0`
  - `n_risk = normalized_score(N_total)` (log or z-scaled)
  - `p_risk = normalized_score(P_total)`
  - `ph_deviation = abs(pH - 7)`
  - Final `chemical_risk_score = weighted_sum([do_risk, n_risk, p_risk, ph_deviation])`

### 3.2 Environmental & Infrastructure Data (WPDX + GEE)

- **WPDX:**
  - Lat/Lon
  - Water source type (borehole, spring, etc.)
  - Status (functional/non-functional)
  - Management type
  - Install year
- **GEE Layers:**
  - **Rainfall** (CHIRPS - past 30 days)
  - **Land Surface Temperature** (MODIS LST - 8-day average)
  - **Vegetation (NDVI)** - Proxy for land use/agricultural activity
- Feature Engineering:
  - `rainfall_30d_sum`
  - `lst_mean`
  - `ndvi_avg`

- `years_since_installation`
- `proximity_to_latrines_or_farms` (toxic risk proxy)
- Risk Calculation:
  - Environmental risk is modeled with greater weighting due to broader coverage and temporal resolution.
  - `environmental_risk_score = model(XGBoost or RF on engineered features)`

### 3.3 NLP-derived Text Features (Field Reports, Logs)

- Sources: NGO reports, technician logs, community SMS/self-reports
  - Extraction:
    - **Keywords:** diarrhea, stomach, smell, cloudy, broken, rust
    - **TF-IDF vectorization**
    - **Sentiment score** (TextBlob or transformer-based, fine-tuned)
  - NLP Feature Engineering:
    - `mentions_smell_flag, mentions_illness_flag`
    - `report_sentiment_score`
    - `nlp_risk_score = sigmoid(weighted_sum([flags + sentiment]))`
  - NLP plays a supporting role, as text is sparse. Weight in ensemble: ~10%.
- 

## 4. Modeling Approach

### 4.1 Model Structure (CRISP-DM aligned)

- Data Understanding -> Feature Engineering -> Model Selection -> Evaluation -> Deployment

### 4.2 Individual Models

#### A. GEMS Chemical Model

- **Target:** Safe/Unsafe based on thresholds
- **Algorithm:** Logistic Regression (baseline), Random Forest (final)
- **Features:** DO, N, P, pH-deviation
- **Eval:** F1 Score, ROC-AUC

#### B. Environmental Model (WPDx + GEE)

- **Target:** Safe/Unsafe based on status\_id and historic contamination zones
- **Algorithm:** XGBoost (high performance), Random Forest (interpretability)
- **Features:** rainfall, NDVI, LST, years\_since\_install, source type, location
- **Eval:** ROC-AUC > 0.85, F1 > 0.80

#### C. NLP Signal Model

- **Target:** Contamination-flag probability
- **Algorithm:** TF-IDF + Logistic Regression | Optional: BERT embeddings + shallow MLP
- **Features:** Mention flags, keyword scores, sentiment polarity
- **Eval:** Precision @ high risk threshold, recall on true contaminated classes

## 4.3 Ensemble Strategy

- **Weighted Voting Ensemble:**
  - GEMS chemical model: 25%
  - Environmental model: 65%
  - NLP model: 10%
- Final Prediction:

```
final_risk_score = 0.65 * env_score + 0.25 * chem_score + 0.10 * nlp_score  
label = 'Contaminated' if final_risk_score >= 0.5 else 'Safe'
```

- **Interpretability:** SHAP for tree models, LIME for NLP, PDPs to visualize risk vs. feature values

---

## 5. Deployment and Application (Streamlit Dashboard)

### 5.1 Streamlit Dashboard Features (Descriptive Mapping to Objectives)

Each feature in the Streamlit UI is intentionally designed to fulfill our goals and objectives, ensuring that stakeholders can easily access and act on the information provided:

#### 1. Location-Based Contamination Risk Search

- **Feature:** Users can input coordinates or select a village/location from a dropdown.
- **How it supports the objectives:**
  - Tied to Objective #1 and #2: Enables hyper-local predictions and planning.
  - Allows use even in areas without dense sensor coverage.

#### 2. Risk Classification Output (Safe/Contaminated)

- **Feature:** Outputs binary classification based on the fused model.
- **How it supports the objectives:**
  - Tied to Objective #1: Clear communication of water safety status.
  - Uses combined strength of three models.

#### 3. Interactive Feature Contribution Table

- **Feature:** Shows which features most influenced a prediction using SHAP values.

- **How it supports the objectives:**

- Tied to Objective #4: Builds model trust and helps identify interventions (e.g., nearby latrines, high nitrogen).

#### 4. Geospatial Risk Heatmaps

- **Feature:** Map overlay of predicted risk across regions or villages.

- **How it supports the objectives:**

- Tied to Objective #2 and #3: Visual planning tool for stakeholders to see hotspots and act proactively.

#### 5. Ask Questions About Observations

- **Feature:** Accepts logs or field notes in text form as questions to parse contamination signals to retrain and answer.

- **How it supports the objectives:**

- Tied to Objective #3: Makes use of sparse, subjective field data to boost weak signal detection.

#### 6. Intervention Recommendations Panel

- **Feature:** Based on dominant contributing factors, suggests testing, chlorination, or hygiene education.

- **How it supports the objectives:**

- Tied to Objective #2: Enables immediate action.
- Rooted in model logic, not generic advice.

#### 5.2 Post-deployment Stakeholder Use Cases

- **NGO Alerts(Objective #2):**

- Unsafe flags trigger notifications to relevant authorities.

- **On-Demand Custom Reports Generation (Objective #2 & Monetization):**

- This feature serves dual purposes: informing evidence-based policy decisions for authorities and creating a revenue stream through specialized research deliverables.
- Stakeholders can request niche queries like correlation analysis between installation year and failure rates, or nitrogen contamination patterns near agricultural zones.
- Reports include visualizations, statistical summaries, and actionable recommendations tailored to the specific query.

- **Data-Backed Action and Resource Allocation (Objective #2):**




- Rank risky villages by predicted contamination probability -> Dispatch limited testing kits, mobile health teams, and awareness campaigns.

- Combine NLP-derived signals (mentions of illness, odor complaints) with environmental risk scores -> Ensure early intervention timing and type.

5.3 Model Feedback Loop

- **Label Incorporation (Objective #5):**
  - Real-world water tests from NGOs and local health teams feed back into the dataset.
  - Regular retraining ensures that the model adapts to seasonal patterns or new contamination dynamics.
- **Live Monitoring (Objective #5):**
  - Use scheduled GEE and WPDx fetches to keep environmental and infrastructure data current.
- **Performance Monitoring:**
  - Auto-track precision, recall, and F1 score of predictions against verified contamination outcomes.

6. File & Folder Structure Summary (GitHub)

cleanwaterai/	
├─  app/	# Streamlit application & live
services	
└─ streamlit_app.py	# Main dashboard interface
└─ trigger_ingestion.py	# Live data ingestion scheduler
└─ trigger_predictions.py	# Real-time prediction engine
└─ trigger_alerts.py	# Alert notification system
└─ trigger_reports.py	# Minimal queried CSVs for
specific purposes	
└─  data/	# Data storage hierarchy
└─ raw/	# Original source data
└─ wpd_x_kenya.csv	# Water Point Data Exchange (22K
points)	
└─ ndvi_scaled.csv	# Satellite environmental data
└─ mercury.csv	# GEMS mercury contamination
data	
└─ zinc.csv	# GEMS zinc contamination data
└─ processed/	# Cleaned, merged datasets
└─ environmental.csv	# Processed environmental
features	
└─ gems.csv	# Processed GEMS water quality
data	
└─ nlp.csv	# Processed text analysis data
└─  scripts/	# Core data pipeline scripts
└─ extract_data.py	# WPDx API data extraction
└─ merge_data.py	# Data integration & joining
└─ prepare_data.py	# Feature engineering pipeline
└─ train_xgb.py	# XGBoost model training

├─ train_nlp.py	# NLP model for text analysis
├─ ingest_live_wpdx.py	# Live WPDx updates
├─ ingest_live_gee.py	# Live satellite data
├─ ingest_live_nlp.py	# Live text processing
├─ deploy.py	# Model deployment utilities
├─ notebooks/	# Jupyter analysis notebooks
(CRISP-DM flow)	
├─ 01_data_extraction.ipynb	# Data Understanding
├─ 02_data_cleaning.ipynb	# Data Preparation
├─ 03_data_merging.ipynb	# Data Integration
├─ 04_data_preparation.ipynb	# Feature Engineering
├─ 05_xgboost_model_training.ipynb	# Modeling
├─ 06_nlp_model_training.ipynb	# Text Analytics
├─ 07_model_evaluation.ipynb	# Model Assessment
├─ 08_model_interpretability.ipynb	# Model Insights
├─ 09_deployment.ipynb	# Deployment Strategy
├─ 10_monitoring_and_maintenance.ipynb	# Production Monitoring
├─ models/	# Trained model artifacts
├─ nlp.pkl	# NLP model for text
classification	
├─ reports/	# Documentation & presentations
├─ prd.pdf	# Product Requirements Document
├─ presentation.pdf	# Project presentation
├─ report.pdf	# Technical report
├─ main.py	# Main orchestration starting
point	
├─ requirements.txt	# Python dependencies
├─ pyproject.toml	# Build configuration
├─ setup.py	# Package installation

## 7. Conclusion

This PRD outlines a complete pipeline to predict and monitor rural water contamination risks using an ensemble of AI models fed by chemicals, environmental, and text-derived signals. Every feature, from model training to interactive geospatial dashboards, is designed to map directly to a public health objective.

By weighting models appropriately and deploying them via a user-friendly interface, we bridge the gap between technical AI capabilities and lifesaving field action. With data pipelines, explainable models, and real-time alerts, this solution not only predicts contamination but empowers local actors to intervene before crises escalate.

The groundwork is done—what remains is to finalize the modeling notebooks, run evaluations, and feed outputs to Streamlit for impact.

"Safe water should not wait for a crisis. Predict it, prevent it."