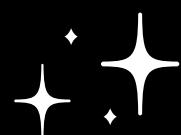
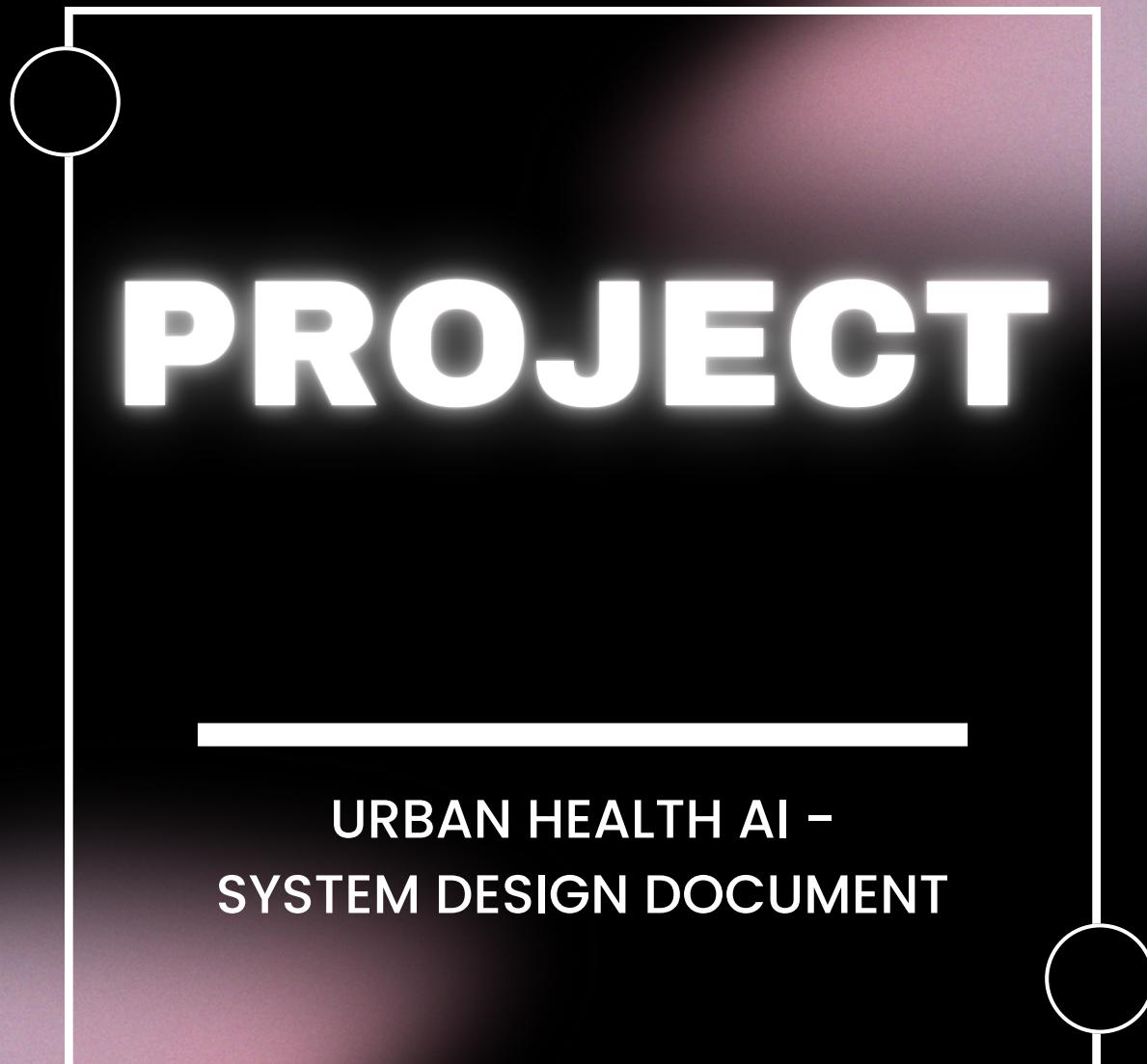


MD ABDUL RAZAQ



**"I DESIGNED AN END-TO-END, PRODUCTION-AWARE ML SYSTEM THAT INGESTS UNRELIABLE EXTERNAL DATA, ENFORCES FEATURE CONSISTENCY, PREDICTS URBAN HEALTH RISK, AND DELIVERS INSIGHTS THROUGH A CLOUD-DEPLOYED INTERACTIVE DASHBOARD."**

# **URBAN HEALTH AI — SYSTEM DESIGN EXPLANATION**

## **SYSTEM OVERVIEW**

**- URBAN HEALTH AI IS A DATA-DRIVEN DECISION SYSTEM THAT CONVERTS ENVIRONMENTAL DATA INTO HEALTH RISK INTELLIGENCE.**

**THE SYSTEM HAS FIVE MAJOR LAYERS:**

- 1. DATA INGESTION**
- 2. DATA PROCESSING & FEATURE ENGINEERING**
- 3. MACHINE LEARNING LAYER**
- 4. VISUALIZATION & INTERACTION LAYER**
- 5. DEPLOYMENT & OPERATIONS**

**EACH LAYER IS LOOSELY COUPLED, SO FAILURES IN ONE LAYER DO NOT BREAK THE ENTIRE SYSTEM.**

## DATA INGESTION LAYER

### PURPOSE

- COLLECT RAW POLLUTION AND WEATHER DATA FROM EXTERNAL SOURCES.

### SOURCES

- AIR POLLUTION: OPENAQ API
- WEATHER: OPENWEATHER API

### REAL-WORLD CHALLENGES

- APIs RETURN INCONSISTENT SCHEMAS
- FREQUENT 401 / 404 / DEPRECATED ENDPOINTS
- MISSING OR NULL FIELDS

### DESIGN DECISIONS

- VALIDATE HTTP STATUS CODES BEFORE PROCESSING
- DEFENSIVE JSON PARSING (CHECK KEYS BEFORE ACCESS)
- GRACEFULFallback WHEN DATA IS INCOMPLETE

### WHY THIS DESIGN?

- EXTERNAL APIs ARE UNRELIABLE IN PRODUCTION.
- FAILING FAST OR CRASHING PIPELINES IS UNACCEPTABLE.

## DATA PROCESSING & FEATURE ENGINEERING LAYER

### PURPOSE

- TRANSFORM RAW API RESPONSES INTO CLEAN, ML-READY STRUCTURED DATA.

### PROCESSING STEPS

- NORMALIZE COLUMN NAMES
- HANDLE MISSING VALUES
- AGGREGATE POLLUTION METRICS BY CITY
- COMPUTE AQI USING PM2.5 & PM10 STANDARDS
- MERGE POLLUTION AND WEATHER DATASETS

### STORAGE STRATEGY

- RAW DATA → DATA/RAW/
- PROCESSED DATA → DATA/PROCESSED/
- ML-READY DATA → ML\_READY.CSV

### WHY BATCH PROCESSING?

- REDUCES API USAGE COST
- IMPROVES REPRODUCIBILITY
- DECOUPLES INGESTION FROM INFERENCE

## MACHINE LEARNING LAYER

### OBJECTIVE

- PREDICT HEALTH RISK CATEGORY (LOW / MODERATE / HIGH) FOR A CITY.

### MODEL CHOICE

- RANDOM FOREST CLASSIFIER

### WHY RANDOM FOREST?

- HANDLES NON-LINEAR RELATIONSHIPS WELL
- ROBUST TO NOISY ENVIRONMENTAL DATA
- REQUIRES MINIMAL FEATURE SCALING
- WORKS EFFICIENTLY ON CPU (CLOUD-FRIENDLY)

### FEATURE CONTRACT ENFORCEMENT

- MODEL FEATURES LOCKED USING:
- MODEL.FEATURE\_NAMES\_IN\_
- INFERENCE INPUTS STRICTLY ALIGNED WITH TRAINING SCHEMA

### WHY THIS MATTERS?

- FEATURE MISMATCH IS ONE OF THE MOST COMMON ML PRODUCTION FAILURES.
- THIS DESIGN GUARANTEES STABLE PREDICTIONS ACROSS ENVIRONMENTS.

## VISUALIZATION & INTERACTION LAYER

 PURPOSE  
CONVERT MODEL OUTPUTS INTO HUMAN-UNDERSTANDABLE INSIGHTS.

### DASHBOARD CAPABILITIES

- CITY-WISE POLLUTION METRICS
- AQI VISUALIZATION
- TIME-BASED TREND ANALYSIS
- INTERACTIVE INDIA MAP (GEOSPATIAL AQI)
- MANUAL "WHAT-IF" HEALTH RISK PREDICTION

### MAPPING

- IMPLEMENTED USING PYDECK
- COLOR-CODED AQI SEVERITY
- TOOLTIP-BASED CONTEXTUAL INSIGHTS

### DESIGN PHILOSOPHY

- COMPANIES DON'T HIRE FOR CSVS.
- THEY HIRE FOR INSIGHTS AND DECISION CLARITY.

## DEPLOYMENT & OPERATIONS LAYER

### PLATFORM

- STREAMLIT CLOUD

### DEPLOYMENT CHALLENGES

- MISSING DEPENDENCIES
- LOCAL VS CLOUD ENVIRONMENT MISMATCH
- SECURE HANDLING OF API KEYS

### SOLUTIONS

- EXPLICIT DEPENDENCY MANAGEMENT (REQUIREMENTS.TXT)
- ENVIRONMENT VARIABLES FOR SECRETS
- GITHUB-BASED CI DEPLOYMENT FLOW

### SECURITY

- NO SECRETS COMMITTED TO GIT
- API KEYS ISOLATED FROM CODEBASE

## SCALABILITY CONSIDERATIONS

### CURRENT SCALE

- 5 CITIES
- BATCH-BASED UPDATES
- SINGLE ML MODEL

### FUTURE SCALING PATH

- ADD MORE CITIES WITHOUT CODE CHANGES
- REPLACE CSV WITH POSTGRESQL / BIGQUERY
- INTRODUCE SCHEDULED INGESTION JOBS
- ENABLE MODEL RETRAINING PIPELINES

### WHY THIS DESIGN SCALES

- MODULAR COMPONENTS ALLOW HORIZONTAL GROWTH WITHOUT ARCHITECTURAL REWRITES.

## RELIABILITY & FAULT TOLERANCE

- API FAILURES DO NOT CRASH PIPELINE
- MISSING DATA HANDLED GRACEFULLY
- PREDICTION SCHEMA ENFORCED
- DASHBOARD REMAINS OPERATIONAL EVEN WITH PARTIAL DATA

THIS ENSURES HIGH SYSTEM AVAILABILITY, EVEN UNDER IMPERFECT CONDITIONS.

## END-TO-END FLOW SUMMARY-



## 🔍 WHY THIS SYSTEM DESIGN IS STRONG

THIS SYSTEM DEMONSTRATES:

- REAL-WORLD DATA ENGINEERING
- ML PRODUCTION AWARENESS
- DEFENSIVE PROGRAMMING
- CLEAR SEPARATION OF CONCERNS
- BUSINESS-FOCUSSED INSIGHTS

IN SHORT:

NOT JUST A MODEL – A COMPLETE SYSTEM.