

UVCeed Disease & Outbreak Awareness Architecture (v1)

Objective: Enable the UVCeed app to proactively alert users when communicable disease activity or foodborne illness outbreaks increase in their area, so they can increase disinfection frequency and reduce exposure risk. All alerts are advisory and based on public health surveillance data.

Design Principles

- Reliability over live querying
- Server-side normalization and scoring
- ZIP-based UX with county-level aggregation
- Commercial-safe use of public data
- Extensible to new diseases and regions

Primary Data Sources

Communicable Diseases (CDC/HHS)

- Wastewater Surveillance (COVID, Flu, RSV)
- Influenza (FluView / ILINet)
- RSV (RSV-NET, NREVSS)
- CDC Current Multistate Outbreaks

Foodborne Illness

- FDA openFDA Food Enforcement API
- FDA Recalls & Safety Alerts (website fallback)
- USDA FSIS Recalls (API + website fallback)

Geolocation Strategy

- User provides ZIP code (or coarse location)
- ZIP mapped to county FIPS via HUD/FCC datasets
- County used as the canonical geographic unit for risk scoring

System Architecture Overview

Public Health Data Sources		UVCeed Backend		UVCeed App
CDC / FDA / USDA (APIs & Websites)	→	Ingestion Jobs (Normalize & Store)	→	ZIP-based Risk Alerts & Disinfection Guidance
		Risk Scoring Engine (County-Level)		Push Notifications & In-App Advisories

Application API

The mobile app queries a single UVCeed-managed endpoint such as:

GET /v1/outbreaks?zip=XXXXXX

The response includes county, state, last update time, disease-specific risk levels, an overall risk score, and a human-readable disinfection advisory.

Why Backend Aggregation (Key Decision)

- Protects API keys and avoids rate limits
- Shields users from upstream outages
- Enables push notifications and proactive alerts
- Allows risk logic to evolve without app updates