

# Event Finder — Concise High-Level Architecture

## Event Finder — High-Level Architecture (Concise)

### 1) Purpose

Provide a lightweight system for event discovery: ingest events, store them in a simple catalog, enable fast search, and return user-friendly summaries via a web page.

### 2) Core Capabilities

- Ingestion: Gather and normalize events from trusted sources.
- Catalog: Maintain structured, deduplicated events with source metadata.
- Search: Keyword matching, basic relevance (recency), future filters.
- Response: Generate readable narratives and/or simple cards.
- Web UI: Single search box with results display, mobile-friendly.

### 3) Interaction Flow

Sources → Ingestion → Catalog → Search → Response → Web UI

### 4) Cross-Cutting Concerns

- Data quality: normalization, deduplication, fallback handling.
- Security: approved sources only, safe configs (no public debug).
- Observability: logs for ingestion/search, error monitoring.
- Accessibility: semantic HTML, keyboard navigation.

### 5) Roadmap

Phase 0: MVP with keyword search and narrative responses.

Phase 1: Persistent catalog, basic filters (date, location).

Phase 2: Relevance tuning, multi-source ingestion, simple dashboards.

Phase 3: Geo-awareness, rich cards, TLS & rate-limiting, admin tools.

### 6) Success Metrics

- Query success rate
- Time-to-first-result
- Data freshness
- Zero-result recovery
- Ingestion error rate