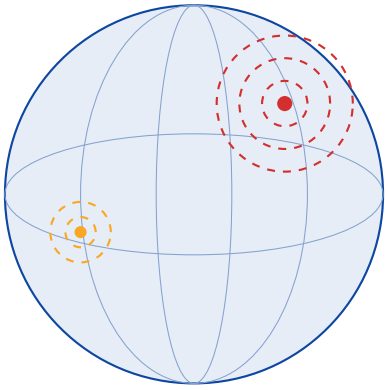
ShockGlobe

Global Event-Driven Market Shock Visualizer



Comprehensive Project Plan

Hacklytics 2026

Team: UpStackLabs

Date: February 20–22, 2026

Version: 1.0

Powered by Globe.GL • Actian VectorAI DB • Sphinx AI

Contents

1 Hacklytics 2026 Overview **3**
1.1 Track Selection: Finance . . . 3
1.2 Sponsor Challenge Submissions . . . 3
1.2.1 Best Use of Actian VectorAI DB . . . 3
1.2.2 Most Unique Application of Sphinx AI . . . 4
1.3 Team Constraints . . . 4

2 Submission Requirements **5**
2.1 Required Deliverables . . . 5
2.2 Judging Format . . . 5
2.3 Judging Criteria . . . 6

3 Executive Summary **7**
3.1 The Problem . . . 7
3.2 The Solution . . . 7

4 Use Cases & Scenarios **8**
4.1 Primary Use Case: Venezuelan Invasion Scenario . . . 8
4.2 Additional Use Cases . . . 8

5 Architecture Overview **9**
5.1 High-Level System Architecture . . . 9
5.2 Data Flow . . . 9

6 Core Algorithms & Models **11**
6.1 Shock Score Calculation . . . 11
6.2 Surprise Factor . . . 11
6.3 Interlinkedness Score . . . 11

7 Technology Stack **12**
7.1 Frontend . . . 12
7.2 Backend & Intelligence . . . 12
7.3 Data Sources . . . 12

8 Globe Visualization Design **13**
8.1 Visual Elements . . . 13
8.1.1 Heatmap Layer . . . 13
8.1.2 Event Markers . . . 13
8.1.3 Connection Arcs . . . 13
8.1.4 Stock Particles . . . 13
8.2 Interactive Features . . . 14

9 Actian VectorAI DB Integration **15**
9.1 Setup & Access . . . 15
9.2 Role in ShockGlobe . . . 15
9.3 Schema Design . . . 15
9.4 Query Patterns . . . 15

10 Sphinx AI Integration **17**
10.1 ML Model Pipeline . . . 17
10.2 Model Architecture . . . 17

11 Frontend Component Architecture **19**
11.1 Component Hierarchy . . . 19
11.2 State Management (Zustand) . . . 20

12 API Design **21**
12.1 REST Endpoints . . . 21
12.2 WebSocket Channels . . . 21

13 Data Pipeline & Ingestion **22**
13.1 Event Ingestion Pipeline . . . 22
13.2 Market Data Pipeline . . . 22
13.3 OSINT & Alternative Data Pipeline . . . 22

14 Project Structure **23**
14.1 Repository Layout . . . 23

15 Implementation Roadmap **25**
15.1 Phase 1: Foundation — Friday Night (7:30 PM – 1:00 AM) . . . 25
15.2 Phase 2: Core Features — Saturday (9:00 AM – 7:00 PM) . . . 25
15.3 Phase 3: Intelligence & Polish — Saturday Night (7:00 PM – 3:00 AM) . . . 25
15.4 Phase 4: Submission Prep — Sunday Morning (6:00 AM – 9:00 AM) . . . 26

16 Venezuelan Invasion: Full Demo Walkthrough **27**
16.1 Scenario Setup . . . 27
16.2 Shock Propagation Visualization . . . 27
16.3 Interactive Exploration . . . 27
16.4 Pentagon Pizza Tracker Integration . . . 28

17 Performance Considerations **29**
17.1 Frontend Optimization . . . 29
17.2 Backend Optimization . . . 29

18 Risk Mitigation **29**

19 Success Metrics **30**

20 Conclusion **30**

Hacklytics 2026 Overview

Hacklytics 2026: Golden Byte is a 36-hour data science hackathon (datathon) hosted by Data Science @ GT at the Georgia Institute of Technology, Klaus Advanced Computing Center. The event runs from **Friday, February 20 to Sunday, February 22, 2026**.

Key Insight

As a **datathon**, Hacklytics challenges teams to craft data-driven solutions. ShockGlobe leverages generative AI, machine learning, predictive modeling, and interactive data visualization to deliver a compelling finance-track entry.

Track Selection: Finance

ShockGlobe will be submitted under the **Finance** track:

> *"Analyze market trends, predict stock movements, and build fintech solutions."*

Finance Track Prizes:

- **First Place:** Nespresso Vertuo Next w/ Milk Frother
- **Second Place:** JBL Grip Speaker
- **Third Place:** Clay Poker Set

All projects are automatically considered for **Best Overall Hack**, judged on: Scope and Technical Depth, Creativity and Originality, Impact and Relevance, Clarity and Engagement, and Soundness and Accuracy.

Best Overall Prizes:

- **First Place:** Apple 13" MacBook Air (M4)
- **Second Place:** Apple AirPods Max
- **Third Place:** Samsung 27" Odyssey OLED G5 QHD Gaming Monitor

Sponsor Challenge Submissions

Teams may submit to **unlimited sponsor challenges** in addition to their one Hacklytics track. ShockGlobe will target:

Best Use of Actian VectorAI DB

> *"Build an AI application using Actian VectorAI DB. We're looking for creative implementations that showcase vector search capabilities—examples include RAG systems, recommendation engines, semantic search applications, or real-time AI solutions."*

Access:

- Docker: `docker pull williamimoh/actian-vectorai-db:1.0b`
- GitHub: https://github.com/hackmamba-io/actian-vectorAI-db-beta

Prizes (quality-gated, per team):

- 1st Place: $500 in prepaid AmEx cards
- 2nd Place: $300 in prepaid AmEx cards
- 3rd Place: $200 in prepaid AmEx cards

Most Unique Application of Sphinx AI

> *"Build something creative using Sphinx AI and show off an unexpected or innovative use of the platform. This challenge rewards projects that push beyond typical use cases and demonstrate original ways Sphinx can reason over data and deliver insight."*

Prize (first place only): $100 Visa gift cards + Sphinx branded backpack

Team Constraints

- Maximum **4 students** per project team.
- Each team submits **ONE project** (excluding the separate Figma individual challenge).
- Each team is eligible for only **one Hacklytics prize category** (either Overall or Finance Track).

Submission Requirements

All materials must be submitted by **9:00 AM EST on Sunday, February 22, 2026**. No late submissions will be accepted.

Required Deliverables

1. **Devpost Submission** (https://hacklytics-2026.devpost.com/)
    - Select the **Finance** track
    - Select sponsor challenges: **Best Use of Actian VectorAI DB** and **Most Unique Application of Sphinx**
    - Only ONE team member submits
2. **2-Minute Demo Video**
    - Explain the project and highlight main features
    - Submit as a URL to YouTube or Google Drive (must be set to "Public")
    - Must fit within 2 minutes
3. **Project Writeup**
    - Outline ideation and development process
    - Submitted directly on Devpost
4. **Public GitHub Repository**
    - Contains all project code
    - Must include a README
    - Repository must be set to public
5. **Google Form**
    - Devpost submission link
    - Project name
    - Team members' details
    - Chosen track (Finance) and challenges (Actian, Sphinx)

Judging Format

Projects will be evaluated during a **live expo**:

- Up to **5 minutes** to present and demo
- Followed by **2 minutes** of Q&A with judges

Criterion	Description
Scope & Technical Depth	Complexity of the solution, quality of engineering, and technical ambition
Creativity & Originality	Novel approach, unique perspective, and innovative use of data/tools
Impact & Relevance	Real-world value, market applicability, and potential for meaningful use
Clarity & Engagement	Quality of presentation, demo polish, and ability to communicate the idea
Soundness & Accuracy	Validity of data analysis, model correctness, and reliability of results

Table 1: Hacklytics 2026 judging criteria

Judging Criteria

Key Insight

Demo Strategy: With only 5 minutes to present, the Venezuelan invasion scenario must be a tight, rehearsed narrative: event trigger → shock propagation on globe → surprise factor discovery → interactive drill-down. Every second counts.

Executive Summary

ShockGlobe is an interactive 3D globe-based visualization platform that connects global events—geopolitical crises, economic shifts, military movements, and policy changes—to their predicted market impacts across countries, sectors, and individual stocks.

Key Insight

The core innovation is the **Surprise Factor**: a quantitative measure of how unexpectedly a stock's price moves relative to a triggering global event. ShockGlobe makes these invisible ripple effects visible, interactive, and explorable on a 3D globe in near real-time.

The Problem

- Global events cause cascading, non-obvious market shocks that propagate across regions and sectors.
- Retail and institutional investors lack intuitive tools to visualize *how* a single event ripples through interconnected global markets.
- Existing dashboards are flat, disconnected, and fail to show geographic and sectoral propagation patterns.
- The relationship between events (e.g., a Venezuelan invasion) and affected assets (e.g., oil futures, Latin American equities, shipping stocks) is opaque.

The Solution

ShockGlobe provides:

1. **Real-time event tracking** — geopolitical, economic, military, and policy events ingested from multiple data sources.
2. **Shock propagation mapping** — each event is mapped to predicted market impacts across sectors, regions, and individual tickers.
3. **3D globe visualization** — shock intensity rendered as heatmaps, ripple animations, icons, and interactive overlays on a Globe.GL-powered interface.
4. **Surprise Factor analysis** — quantifying how unusual a stock's price movement is relative to event expectations.
5. **Event simulation** — "what-if" scenario modeling to predict potential market impacts of hypothetical future events.

Use Cases & Scenarios

Primary Use Case: Venezuelan Invasion Scenario

A hypothetical Venezuelan military conflict serves as the flagship demonstration:

Impact Category	Affected Assets	Shock Direction
Crude Oil	WTI, Brent, PDVSA-linked bonds	↑↑↑
Latin American FX	VEF, COP, BRL	↓↓
Defense Stocks	LMT, RTX, NOC, GD	↑↑
Shipping / Logistics	Tanker companies, ports	↑
Regional Equities	Caracas Stock Exchange, iShares MSCI	↓↓↓
Safe Havens	Gold, USD, US Treasuries	↑↑
Agriculture	Coffee, cocoa (supply chain)	↑

Table 2: Predicted shock propagation from a Venezuelan invasion scenario

Additional Use Cases

1. **Pentagon Pizza Tracker** — Monitor Pentagon activity patterns (food delivery surges as a proxy for crisis activity). Correlate with defense spending, military mobilization signals, and stock movements in defense sector.

2. **Satellite Movement Tracking** — Track satellite repositioning over conflict zones. Unusual satellite activity → geopolitical tension signals → defense/energy stock impacts.

3. **Military Movement Detection** — Aggregate open-source intelligence (OSINT) on troop/fleet movements. Map to regional market impacts and commodity shocks.

4. **Jobs Data Shock Analysis** — Non-farm payrolls, unemployment claims, and labor market surprises mapped to sector rotations and equity impacts.

5. **Tariff & Trade Policy Shocks** — Track sudden tariff announcements or trade deal breakdowns. Map to affected industries, currencies, and supply-chain-dependent equities.

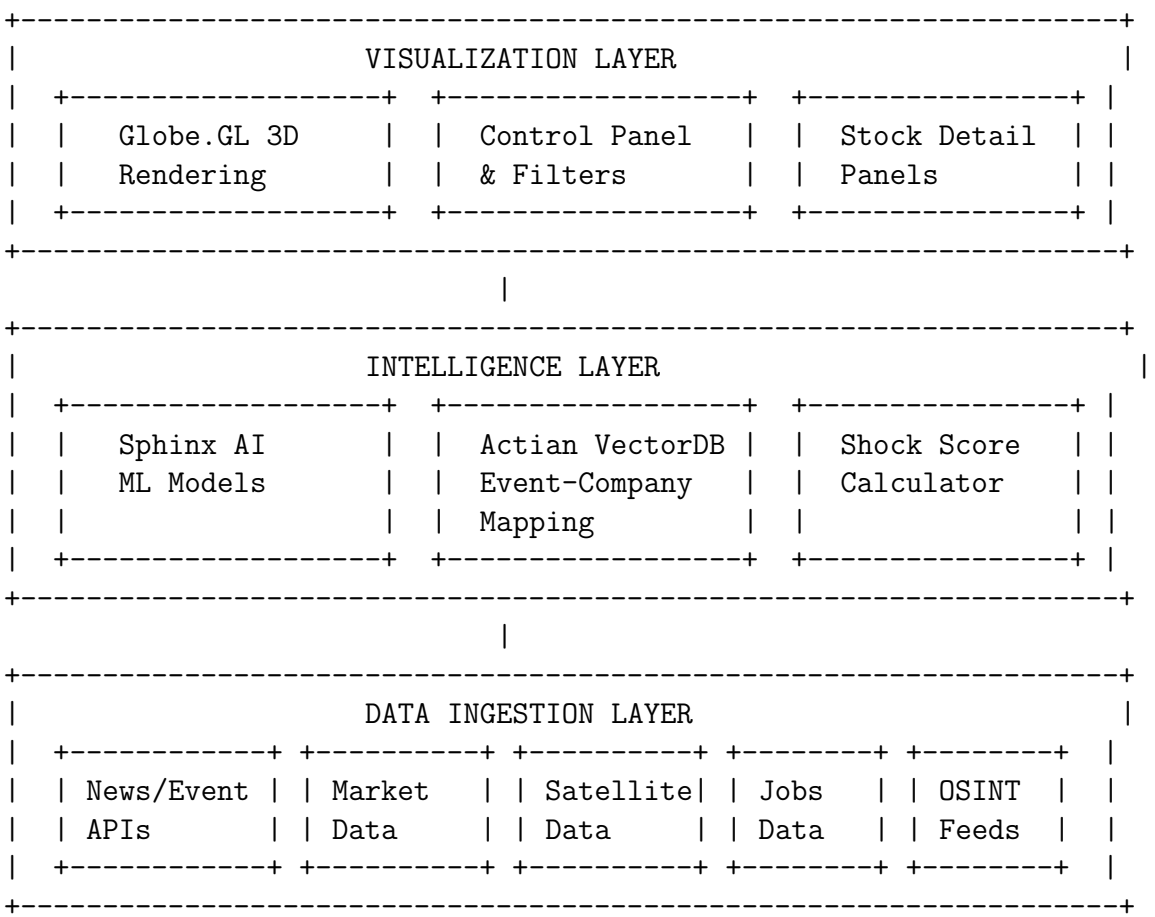
Architecture Overview

High-Level System Architecture

Technical Detail

Three-Tier Architecture:

1. **Data Ingestion Layer** — Event feeds, market data APIs, OSINT sources
2. **Intelligence Layer** — Sphinx AI for ML modeling, Actian VectorDB for semantic event-company mapping
3. **Visualization Layer** — Globe.GL 3D rendering, React frontend, interactive overlays



Data Flow

1. **Event Ingestion**: Global events are captured from news APIs, government feeds, OSINT sources, and satellite/military tracking systems.
2. **Event Embedding**: Each event is converted into a vector embedding using Sphinx AI's NLP models and stored in Actian VectorDB.

3. **Company/Sector Mapping**: VectorDB performs semantic similarity search to find companies, sectors, and regions most likely impacted by each event.

4. **Shock Score Calculation**: A composite shock score is computed for each affected entity based on historical correlation, supply chain linkage, geographic proximity, and sector sensitivity.

5. **Surprise Factor**: Actual market movements are compared against predicted shocks to generate a "surprise" metric—highlighting unexpectedly large or small reactions.

6. **Visualization**: Results are rendered on the Globe.GL 3D globe with heatmaps, connection arcs, ripple animations, and interactive drill-down panels.

Core Algorithms & Models

Shock Score Calculation

The shock score for a company c given an event e is:

$$S(c, e) = \alpha \cdot \text{sim}(\vec{v}_e, \vec{v}_c) + \beta \cdot H(c, e) + \gamma \cdot G(c, e) + \delta \cdot \text{SC}(c, e) \tag{1}$$

Where:

- $\text{sim}(\vec{v}_e, \vec{v}_c)$ = cosine similarity between event embedding and company embedding in Actian VectorDB
- $H(c, e)$ = historical sensitivity of company c to events of type e (regression coefficient from past data)
- $G(c, e)$ = geographic proximity factor (inverse distance weighting from event epicenter to company's primary operating regions)
- $\text{SC}(c, e)$ = supply chain linkage score (graph-based propagation through company supply chain network)
- $\alpha, \beta, \gamma, \delta$ = tunable weights (learned via Sphinx AI or manually set)

Surprise Factor

The surprise factor quantifies how much a stock's actual movement deviates from the model's prediction:

$$\text{Surprise}(c, e) = \frac{|\Delta P_{\text{actual}}(c) - \Delta P_{\text{predicted}}(c, e)|}{\sigma_c} \tag{2}$$

Where:

- $\Delta P_{\text{actual}}(c)$ = actual price change of stock c in the event window
- $\Delta P_{\text{predicted}}(c, e)$ = predicted price change based on shock score $S(c, e)$
- σ_c = historical volatility of stock c (normalizing factor)

Key Insight

A high surprise factor indicates the market reacted far more (or less) than expected—these are the most interesting signals for traders and analysts. Stocks with surprise $> 2\sigma$ are flagged as "shock anomalies" on the globe.

Interlinkedness Score

To measure how interconnected markets are through an event:

$$I(e) = \frac{1}{N(N-1)} \sum_{i \neq j} \text{corr}(\Delta P_i, \Delta P_j \mid e) \tag{3}$$

Where N is the number of affected assets and $\text{corr}(\Delta P_i, \Delta P_j \mid e)$ is the pairwise correlation of price changes conditioned on event e. This metric is visualized as connection arcs on the globe.

Technology Stack

Frontend

Technology	Version	Purpose
React	18+	UI framework, component architecture
TypeScript	5.x	Type safety, developer experience
Globe.GL	Latest	3D globe rendering (WebGL/Three.js)
Three.js	r150+	3D graphics primitives, custom shaders
D3.js	7.x	Data-driven charts, heatmap scales
TailwindCSS	3.x	Utility-first styling
Zustand	4.x	Lightweight state management
React Query	5.x	Server state, caching, real-time data
Vite	5.x	Build tool, HMR, fast dev server
Socket.io Client	4.x	Real-time WebSocket connections

Table 3: Frontend technology stack

Backend & Intelligence

Technology	Version	Purpose
Python / FastAPI	3.11+	API server, event processing
Actian VectorAI DB	1.0b (closed beta)	Vector similarity search, event-company mapping
Sphinx AI	Latest	ML model training, NLP embeddings, predictions
Redis	7.x	Caching, pub/sub for real-time updates
PostgreSQL	15+	Relational data storage (events, companies)
Celery	5.x	Async task queue for data ingestion

Table 4: Backend technology stack

Data Sources

Source	Data Provided
Alpha Vantage / Polygon.io	Live stock tickers, historical prices, intraday data
NewsAPI / GDELT	Global event detection, news sentiment
ACLED	Armed conflict location & event data
BLS / FRED	Jobs data, economic indicators
Sentinel Hub / Planet	Satellite imagery and movement data
ADS-B Exchange	Aircraft/military flight tracking
MarineTraffic	Naval vessel tracking
OpenStreetMap	Geographic data, infrastructure mapping

Table 5: External data sources

Globe Visualization Design

Visual Elements

Heatmap Layer

- Countries colored by aggregate shock intensity (green → yellow → red gradient).
- Opacity and saturation scale with the magnitude of predicted market impact.
- Pulsing effect on countries currently experiencing active shock propagation.

Event Markers

- **Epicenter Pins**: Large, animated markers at event origin points (e.g., Caracas for Venezuelan invasion).
- **Category Icons**: Military (crosshairs), Economic (chart), Policy (gavel), Natural Disaster (lightning).
- **Ripple Rings**: Expanding concentric circles emanating from event epicenters, visualizing shock wave propagation speed and reach.

Connection Arcs

- Curved arcs connecting event epicenters to affected markets/exchanges.
- Arc thickness $\propto$ shock intensity.
- Arc color encodes direction: red = negative impact, green = positive impact.
- Animation: arcs "flow" from event to affected region over time.

Stock Particles

- Individual stocks rendered as small glowing particles clustered around their home exchange.
- Particle size $\propto$ market cap.

- Particle color $\propto$ surprise factor (blue = expected, red = surprising).
- On hover: tooltip with ticker, price change, shock score, surprise factor.

Interactive Features

1. **Click-to-Drill**: Click any country/region to expand a side panel showing affected sectors, top shocked stocks, and historical event parallels.
2. **Time Slider**: Scrub through time to replay how shocks propagated over hours/days.
3. **Event Filter**: Toggle event categories (military, economic, policy, natural) on/off.
4. **Sector Filter**: Highlight only specific sectors (energy, defense, finance, tech, agriculture).
5. **Surprise Mode**: Toggle to highlight only stocks with surprise factor $> 2\sigma$.
6. **Simulation Mode**: Input a hypothetical event, location, and severity to see predicted shock propagation.
7. **Comparison View**: Split-screen two events to compare their shock fingerprints.

Actian VectorAI DB Integration

Setup & Access

Actian VectorAI DB is available as a closed beta for Hacklytics participants:

Technical Detail

Docker:
`docker pull williamimoh/actian-vectorai-db:1.0b`

GitHub Repository:
https://github.com/hackmamba-io/actian-vectorAI-db-beta

Role in ShockGlobe

Actian VectorAI DB serves as the semantic intelligence backbone, enabling:

1. **Event-to-Company Mapping**: Each event description is embedded into a high-dimensional vector. Companies are also embedded based on their business descriptions, sector, supply chain, and geographic footprint. VectorDB's similarity search finds the most relevant companies for any given event.
2. **Interlinkedness Discovery**: By storing company relationship vectors, VectorDB enables discovery of non-obvious connections (e.g., a rare earth mining company in Congo linked to a semiconductor firm in Taiwan through supply chain embeddings).
3. **Historical Event Matching**: When a new event occurs, VectorDB finds the most similar historical events and retrieves their actual market impacts as priors for the shock model.

Schema Design

Technical Detail

Collections:

- `events` — Event embeddings with metadata (type, location, severity, timestamp)
- `companies` — Company embeddings with metadata (ticker, sector, country, market cap)
- `supply_chains` — Supply chain relationship embeddings (company pairs, dependency strength)
- `historical_shocks` — Past event-impact pairs for training and retrieval

Query Patterns

1. **Nearest Neighbor**: Given event vector $\vec{v}_e$, find top-k most affected companies:
`SELECT * FROM companies ORDER BY cosine_similarity(embedding, ` $\vec{v}_e$ `) LIMIT k`

2. **Filtered Search**: Find affected companies within a specific sector or region:
   `WHERE sector = 'Energy' AND cosine_similarity > 0.7`

3. **Historical Retrieval**: Find past events most similar to current event for shock prediction calibration.

Sphinx AI Integration

ML Model Pipeline

Sphinx AI powers the following ML capabilities:

1. **Event Classification**
   - NLP model to classify incoming events by type (geopolitical, economic, military, policy, natural disaster).
   - Severity scoring (1–10 scale) based on event description, historical parallels, and source credibility.
2. **Shock Prediction**
   - Regression model: Event features → predicted price change per affected stock.
   - Training data: Historical events paired with subsequent market movements.
   - Features: event type, severity, geographic proximity, sector sensitivity, supply chain depth.
3. **Surprise Detection**
   - Anomaly detection model trained on residuals (actual vs. predicted price changes).
   - Flags stocks where the market reaction significantly deviates from the model's prediction.
4. **Sentiment Analysis**
   - Real-time sentiment scoring of news articles, social media, and government statements related to events.
   - Sentiment velocity (rate of change) as an early warning signal.
5. **Data Visualization Assistance**
   - Sphinx AI generates dynamic chart configurations based on the data patterns detected.
   - Auto-suggests the most informative visualization for each dataset (heatmap vs. scatter vs. time series).

Model Architecture

```
Event Text  NLP Encoder  Event Embedding  VectorDB Storage


                        Shock Prediction Model
                         (Gradient Boosted Trees
                          + Neural Network Ensemble)
```

```
Predicted          Confidence       Sector-Level
Price Change       Interval         Impact Map


Actual Price   Surprise Factor   Anomaly Alert
```

Frontend Component Architecture

Component Hierarchy

```
<App>
 <Header>
    <Logo />
    <EventSearchBar />
    <ModeToggle /> (Live / Simulation / Historical)

 <GlobeContainer>
    <Globe3D />                    // Globe.GL instance
       <HeatmapLayer />           // Country shock intensity
       <EventMarkers />           // Epicenter pins + ripples
       <ConnectionArcs />         // Event-to-market arcs
       <StockParticles />         // Individual stock dots

    <GlobeControls>
       <TimeSlider />             // Temporal navigation
       <EventFilter />            // Category toggles
       <SectorFilter />           // Sector highlights
       <SurpriseToggle />         // Surprise mode on/off

    <GlobeOverlays>
        <EventTooltip />           // Hover info for events
        <StockTooltip />           // Hover info for stocks
        <ShockLegend />            // Color scale legend

 <SidePanel>
    <EventDetailPanel />           // Selected event deep-dive
       <EventSummary />
       <AffectedSectors />
       <HistoricalParallels />

    <StockDetailPanel />           // Selected stock deep-dive
       <PriceChart />
       <ShockScoreBreakdown />
       <SurpriseAnalysis />

    <SimulationPanel />            // What-if scenario builder
        <EventBuilder />
        <SeveritySlider />
        <PredictionResults />

 <BottomTicker>
    <LiveStockTicker />            // Scrolling live prices

 <Footer />
```

State Management (Zustand)

```
GlobalStore {
  // Event State
  events: Event[]
  selectedEvent: Event | null
  eventFilters: { military, economic, policy, natural }

  // Market State
  stocks: Stock[]
  selectedStock: Stock | null
  sectorFilters: string[]

  // Globe State
  cameraPosition: { lat, lng, altitude }
  timePosition: Date
  viewMode: 'live' | 'historical' | 'simulation'
  surpriseMode: boolean

  // Simulation State
  simulatedEvent: SimEvent | null
  simulationResults: ShockResult[]

  // Actions
  setSelectedEvent(event)
  setTimePosition(date)
  toggleSurpriseMode()
  runSimulation(event)
}
```

API Design

REST Endpoints

Method	Endpoint	Description
GET	`/api/events`	List events with filters (type, date range, severity)
GET	`/api/events/{id}`	Get event details with affected companies
GET	`/api/events/{id}/shocks`	Get shock scores for all affected entities
GET	`/api/stocks`	List stocks with current shock data
GET	`/api/stocks/{ticker}`	Get stock detail with shock history
GET	`/api/stocks/{ticker}/surprise`	Get surprise factor analysis
GET	`/api/globe/heatmap`	Get country-level shock intensity data
GET	`/api/globe/arcs`	Get connection arcs data for globe
POST	`/api/simulate`	Run what-if simulation for a hypothetical event
GET	`/api/sectors`	Get sector-level aggregate shock data
GET	`/api/historical/similar`	Find similar historical events

Table 6: REST API endpoints

WebSocket Channels

Channel	Description
`ws://events/live`	Real-time event feed (new events as they are detected)
`ws://shocks/live`	Real-time shock score updates as events unfold
`ws://prices/live`	Live stock price ticker feed
`ws://surprises/alerts`	Push alerts when surprise factor exceeds threshold

Table 7: WebSocket real-time channels

Data Pipeline & Ingestion

Event Ingestion Pipeline

```
[Data Sources]


[Celery Workers]  Fetch, parse, deduplicate events


[Sphinx AI NLP]  Classify event type, extract entities,
                    score severity, generate embedding

[Actian VectorDB]  Store event embedding
                       Find similar companies (top-k)

[Shock Calculator]  Compute S(c,e) for each affected company


[Redis Pub/Sub]  Push to WebSocket channels


[PostgreSQL]  Persist event, shock scores, and metadata
```

Market Data Pipeline

1. **Live Ticker Feed**: Polygon.io WebSocket or Alpha Vantage polling for real-time prices.
2. **Historical Data**: Daily OHLCV data for backtesting and model training.
3. **Derived Metrics**: Computed in-stream—returns, volatility, moving averages, relative strength.
4. **Surprise Calculation**: On each price update, re-compute surprise factor for active events.

OSINT & Alternative Data Pipeline

1. **Satellite Tracking**: Periodic polling of satellite position APIs. Anomalous repositioning triggers event creation.
2. **Military Movement**: ADS-B flight data + MarineTraffic vessel data. Clustering algorithms detect unusual concentration patterns.
3. **Pentagon Activity**: Proxy signals (delivery data, parking lot imagery, late-night activity patterns) correlated with subsequent policy announcements.
4. **Jobs/Economic Data**: BLS and FRED API polling on release schedules. Instant shock calculation on release.

Project Structure

Repository Layout

```
hackalytics-frontend/
 public/
    index.html
    favicon.ico
    assets/
        textures/             # Globe textures (earth, night)
        icons/                # Event category icons

 src/
    main.tsx                  # App entry point
    App.tsx                   # Root component

    components/
       globe/
          Globe3D.tsx       # Globe.GL wrapper
          HeatmapLayer.tsx  # Country heatmap overlay
          EventMarkers.tsx  # Event pins + ripple effects
          ConnectionArcs.tsx# Shock propagation arcs
          StockParticles.tsx# Individual stock dots
          GlobeControls.tsx # Camera, zoom, rotation

       panels/
          EventDetail.tsx   # Event deep-dive panel
          StockDetail.tsx   # Stock deep-dive panel
          SimulationPanel.tsx # What-if builder
          SectorView.tsx    # Sector breakdown

       controls/
          TimeSlider.tsx    # Temporal navigation
          EventFilter.tsx   # Category toggles
          SectorFilter.tsx  # Sector highlights
          SearchBar.tsx     # Event/stock search

       charts/
          PriceChart.tsx    # Stock price line chart
          ShockBar.tsx      # Shock score bar chart
          SurpriseScatter.tsx # Surprise factor scatter

       common/
           Header.tsx
           Footer.tsx
           Tooltip.tsx
           LiveTicker.tsx
           LoadingSpinner.tsx
```

```
   hooks/
      useGlobe.ts          # Globe instance management
      useEvents.ts         # Event data fetching
      useStocks.ts         # Stock data fetching
      useShocks.ts         # Shock score computation
      useWebSocket.ts      # Real-time connection
      useSimulation.ts     # Simulation logic

   store/
      globalStore.ts       # Zustand global state
      eventStore.ts        # Event-specific state
      simulationStore.ts   # Simulation state

   services/
      api.ts               # REST API client
      websocket.ts         # WebSocket manager
      eventService.ts      # Event data operations
      stockService.ts      # Stock data operations

   types/
      event.ts             # Event type definitions
      stock.ts             # Stock type definitions
      shock.ts             # Shock score types
      globe.ts             # Globe visualization types

   utils/
      shockCalculator.ts   # Client-side shock math
      colorScale.ts        # D3 color scale helpers
      geoUtils.ts          # Geographic calculations
      formatters.ts        # Number/date formatting

   styles/
       globals.css           # Global styles
       tailwind.config.ts    # Tailwind configuration

package.json
tsconfig.json
vite.config.ts
.env.example
README.md
```

Implementation Roadmap

Timeline: Hacking begins Friday evening (Feb 20) after opening ceremony (∼7:30 PM) and ends Sunday morning at **9:00 AM EST** (Feb 22). That gives approximately **36 hours** of build time. The roadmap below uses wall-clock times.

Phase 1: Foundation — Friday Night (7:30 PM – 1:00 AM)

Task	Description	Owner
T1.1	Initialize React + Vite + TypeScript project	Frontend
T1.2	Set up Globe.GL with basic earth rendering	Frontend
T1.3	Implement basic camera controls and globe interaction	Frontend
T1.4	Set up FastAPI backend with project structure	Backend
T1.5	Pull Actian VectorAI DB Docker image, configure collections	Backend
T1.6	Integrate first market data API (live ticker)	Backend
T1.7	Set up Sphinx AI environment and first NLP model	ML

Table 8: Phase 1: Foundation tasks (Friday night)

Phase 2: Core Features — Saturday (9:00 AM – 7:00 PM)

Task	Description	Owner
T2.1	Build event ingestion pipeline with classification	Backend
T2.2	Implement shock score calculation engine	ML
T2.3	Implement VectorAI DB event-company mapping	Backend
T2.4	Build heatmap layer on globe	Frontend
T2.5	Build event markers with ripple animations	Frontend
T2.6	Build connection arcs with shock intensity encoding	Frontend
T2.7	Build side panel with event and stock detail views	Frontend
T2.8	Implement WebSocket real-time data push	Backend
T2.9	Build live stock ticker component	Frontend

Table 9: Phase 2: Core feature tasks (Saturday daytime)

Phase 3: Intelligence & Polish — Saturday Night (7:00 PM – 3:00 AM)

Task	Description	Owner
T3.1	Implement surprise factor calculation and visualization	ML
T3.2	Build simulation mode (what-if scenario builder)	Full Stack
T3.3	Implement time slider for historical replay	Frontend
T3.4	Add sector/category filter controls	Frontend
T3.5	Integrate OSINT data sources (satellite, military)	Backend
T3.6	Build Venezuelan invasion demo scenario with full data	All
T3.7	Performance optimization (LOD, data windowing)	Frontend
T3.8	Polish UI/UX, animations, and transitions	Frontend

Table 10: Phase 3: Intelligence and polish tasks (Saturday night)

Task	Description	Owner
T4.1	Seed database with curated demo events and shock data	Backend
T4.2	**Record 2-minute demo video** (upload to YouTube/Drive as Public)	All
T4.3	Write project README and Devpost writeup	All
T4.4	Final bug fixes and edge case handling	All
T4.5	**Submit to Devpost** (Finance track + Actian + Sphinx challenges)	Lead
T4.6	**Submit Google Form** with Devpost link, team details, track/challenges	Lead
T4.7	Ensure GitHub repository is **public** with complete README	All
T4.8	Rehearse 5-minute live demo + prepare for 2-minute Q&A	All

Table 11: Phase 4: Submission prep (Sunday morning, deadline 9:00 AM EST)

Phase 4: Submission Prep — Sunday Morning (6:00 AM – 9:00 AM)

Key Insight

Critical Deadline: All Devpost submissions, demo videos, and Google Forms must be submitted by **9:00 AM EST Sunday, February 22**. No exceptions. Budget at least 1 hour before the deadline for submission logistics.

Venezuelan Invasion: Full Demo Walkthrough

This section describes the flagship demo scenario in detail.

Scenario Setup

1. User opens ShockGlobe. The 3D globe rotates slowly, showing the current state of global markets.
2. A **breaking event** alert appears: *"Military forces mobilizing near Venezuelan border. DEFCON level elevated."*
3. The globe auto-rotates to center on Venezuela. A red pulsing epicenter marker appears on Caracas.

Shock Propagation Visualization

1. **T+0 seconds**: Red ripple rings begin expanding from Venezuela.
2. **T+2 seconds**: First connection arcs appear—thick red arcs to major oil exchanges (NYMEX, ICE).
3. **T+4 seconds**: Secondary arcs reach defense stocks (Pentagon region lights up).
4. **T+6 seconds**: Latin American currencies show yellow-to-red heatmap intensification.
5. **T+8 seconds**: Safe haven arcs (gold—London/Zurich, USD—global, Treasuries—DC) appear in green.
6. **T+10 seconds**: Supply chain arcs reach unexpected targets (coffee futures, shipping lanes, rare earth minerals).

Interactive Exploration

1. User clicks on **Saudi Arabia**—side panel shows oil sector shock, Aramco impact, OPEC response prediction.
2. User clicks on **Lockheed Martin** particle—detail panel shows +8.2% predicted, +12.1% actual, **Surprise Factor: 2.4**σ.
3. User toggles **Surprise Mode**—globe dims expected movers, highlights only anomalous reactions (e.g., a Colombian coffee company up 15% when model predicted 3%).
4. User drags **Time Slider** to T+48 hours—shows how the shock evolved, which arcs faded, which intensified.

Pentagon Pizza Tracker Integration

1. A secondary indicator shows: *"Pentagon food delivery volume: 340% above normal."*
2. Historical overlay shows that past delivery surges of >200% preceded major military operations within 72 hours in 78% of cases.
3. This data point increases the event severity score, propagating through the shock model in real-time.

Performance Considerations

Frontend Optimization

1. **Level of Detail (LOD)**: Reduce polygon count for distant globe regions. Only render stock particles for the visible hemisphere.
2. **Data Windowing**: Only fetch and render shocks for the current time window. Paginate historical data.
3. **WebGL Instancing**: Use Three.js instanced meshes for stock particles (render thousands of particles in a single draw call).
4. **Shader-Based Heatmap**: Compute heatmap colors in GPU shaders rather than CPU-side D3 scales.
5. **Debounced Updates**: Throttle WebSocket updates to 60fps max; batch state updates to prevent React re-render storms.

Backend Optimization

1. **VectorDB Indexing**: Ensure HNSW index on event and company embeddings for sub-millisecond similarity search.
2. **Redis Caching**: Cache shock scores with 30-second TTL. Serve from cache on globe load.
3. **Batch Processing**: Compute shock scores for all affected companies in parallel (vectorized NumPy operations).
4. **Connection Pooling**: Use async database connections with connection pooling for PostgreSQL and VectorDB.

Risk Mitigation

Risk	Impact	Mitigation
API rate limits	Cannot fetch live data	Pre-cache demo data; use mock data fallback
Globe.GL performance	Slow rendering with many elements	LOD, instancing, particle limits
VectorDB latency	Slow event-company mapping	Index optimization, caching
Hackathon time pressure	Incomplete features	Prioritize demo scenario; cut simulation mode if needed
Data quality	Incorrect shock predictions	Use curated demo data; clearly label as “predicted”

Table 12: Risk mitigation strategies

Success Metrics

Mapped directly to the Hacklytics 2026 judging criteria:

1. **Scope & Technical Depth**: Full three-tier architecture with Globe.GL, Actian VectorAI DB, and Sphinx AI deeply integrated. Vector similarity search, ML shock prediction, and real-time 3D visualization working end-to-end.
2. **Creativity & Originality**: Novel "Surprise Factor" metric; Pentagon pizza tracker as an alternative data signal; 3D globe shock propagation visualization—none of these exist in current market tools.
3. **Impact & Relevance**: Directly applicable to retail/institutional investors, risk managers, and geopolitical analysts. Finance track alignment with real market data.
4. **Clarity & Engagement**: 5-minute live demo rehearsed around the Venezuelan invasion narrative. Globe visualization is inherently engaging and demo-friendly.
5. **Soundness & Accuracy**: Shock scores grounded in cosine similarity, historical regression, and supply chain graph analysis. Surprise factor normalized by historical volatility.
6. **Sponsor Challenge Fit**: Actian VectorAI DB is the core semantic intelligence layer (not bolted on). Sphinx AI drives event classification, shock prediction, and data visualization—a genuinely unique application.
7. **Demo Video**: 2-minute video clearly shows event trigger → globe shock propagation → drill-down → surprise discovery.

Conclusion

ShockGlobe transforms abstract global events into tangible, interactive market intelligence. By combining Globe.GL's immersive 3D visualization with Actian VectorAI DB's semantic mapping and Sphinx AI's predictive modeling, we create a platform that makes the invisible ripple effects of global events visible, explorable, and actionable.

The Venezuelan invasion scenario demonstrates the full power of the system: from event detection to shock propagation to surprise discovery—all rendered on a beautiful, interactive 3D globe that tells the story of how global events move global markets.

Submitted under the **Finance** track at **Hacklytics 2026: Golden Byte**, with sponsor challenge submissions for **Best Use of Actian VectorAI DB** and **Most Unique Application of Sphinx AI**.

ShockGlobe: See the Shock. Understand the Market.