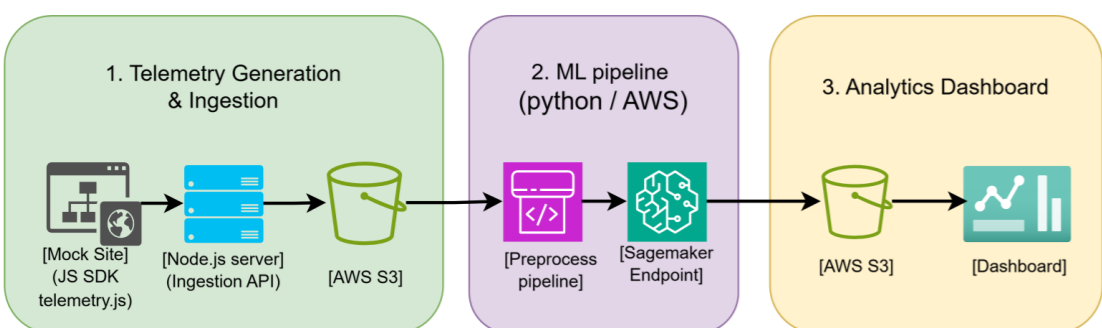


INTRODUCTION

Introduction: Vanguard aims to deliver smooth, low-cost digital experiences across web, mobile, and support channels. While most client interactions are successful, a subset of sessions - referred to as *tail experiences* - exhibit unusually high friction, including repeated errors, navigation loops, or failed workflows. These experiences disproportionately impact customer satisfaction and brand perception. This project develops a synthetic-data-driven AI prototype that detects user frustration by analyzing telemetry generated from a mock digital environment. The system integrates behavioral, system, and sentiment/support-adjacent signals to compute a frustration score and severity level for each synthetic session. The prototype is designed as a proof of concept only and does not use real Vanguard clients or production data.

What was achieved: With the help of custom JS SDK, we were able to capture user behaviours by pairing with simulation engine to generate realistic traffic. ML models were trained and hosted on AWS to automatically assign frustration score to user sessions which were then visualized using react dashboard giving developers & customer service agents real time view of exactly when and where users got stuck and were frustrated.

SYSTEM ARCHITECTURE



We make use of 3 phases for the implementation of end-to-end architecture listed as follows:

1. Phase 1: Telemetry Generation & Data Ingestion

- Mock data of different scenarios are generated and ingested to S3 bucket.

2. Phase 2: Machine learning Pipeline

- Raw data is processed and our trained ML model makes prediction on user frustration.

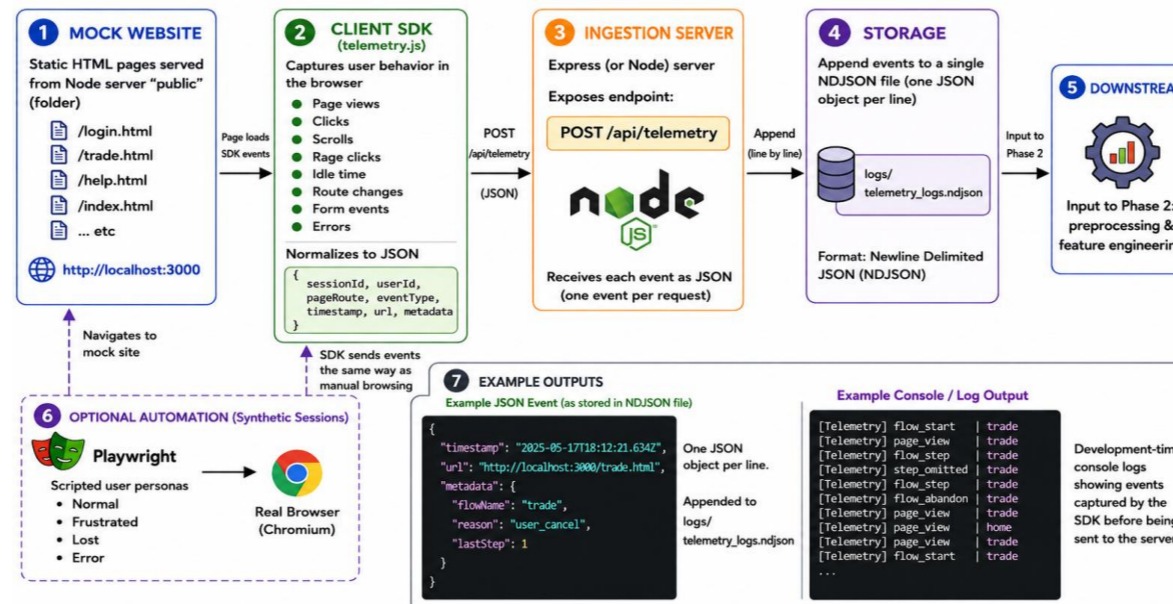
3. Phase 3: Analytics Dashboard

- Visualization of user frustration and other key metrics such as rage clicks, timeline etc

Implementation Stack



PHASE 1 | TELEMETRY & DATA GENERATION



KEY POINTS

- Simulate realistic client behaviors at scale
- Capture friction and interaction telemetry
- Build a scalable ingestion pipeline
- Prepare structured data for AI analysis

PHASE 2 | ML PIPELINE

End to End ML Pipeline from feature engineering to inference.

1. Feature Engineering

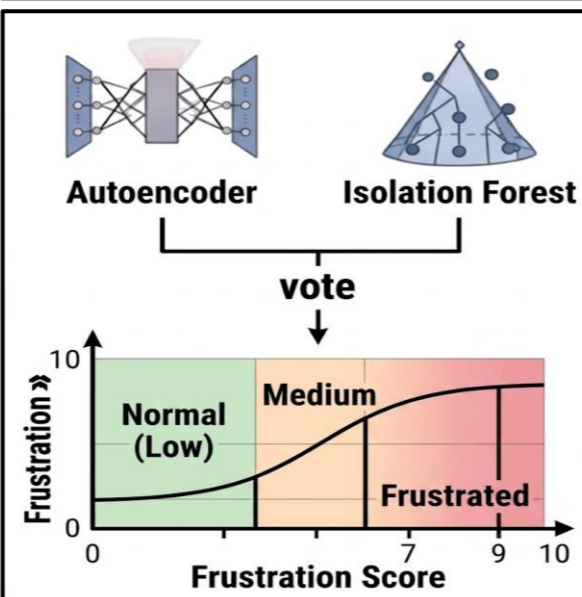
- Vectorized extraction of raw data into 13 session-level features
- Config-driven schema mapping
- Processed data is further standardized for ML model.

2. ML Ensemble

- 2-Model unsupervised Ensemble
- Autoencoder (reconstruction error)
- Isolation Forest (anomaly score)
- Ensemble scores aggregated to 0-10 scale

3. Inference

- Auto-syncs model artifacts from S3
- Automatic fallback to local artifacts if endpoint failure
- Maps scores to severity level and saves to S3



Severity	Score Range	Description
Normal	<7	Expected Behaviour
Medium	7-8.99	Elevated hesitation
High	>9	Anomalous behaviour

PHASE 3 | ANALYTICS DASHBOARD

KEY POINTS

- Telemetry data and AI prediction results are loaded into the dashboard
- Sessions are automatically classified into Low, Medium, or High severity
- Sessions Overview displays scores, severity, timestamps, and status
- Alerts view highlights high-frustration sessions that need attention
- Queue view prioritizes sessions based on severity and timestamp
- Session Detail view shows metrics, predictions, metadata, and event timeline data
- Admin view allows manual severity overrides and feedback/comments
- Dashboard automatically refreshes when new run data becomes available
- Cross-page consistency keeps session and prediction data synchronized across all views

RESULTS & FUTURE WORK

RESULTS

- Built an end-to-end prototype linking telemetry generation, ML scoring, and dashboard visualization.
- Generated realistic synthetic low-, medium-, and high-friction user sessions.
- ML ensemble assigns frustration scores and severity levels.
- Dashboard displays sessions, alerts, queue priority, and session details.
- Admin feedback and overrides support continuous model improvement.

FUTURE WORK

- Improve model accuracy with larger and more diverse datasets.
- Add real-time streaming and faster alerting.
- Expand telemetry signals for hesitation, confusion, and workflow failure.
- Integrate with support or ticketing systems.
- Strengthen privacy, security, and compliance for future production use.