Page View prompt

**Objective:**

Analyze page views to:

* Identify **the most-viewed and least-viewed pages**.
* Provide **view counts** for any requested page.
* Track **trends in user engagement** over time.
* Detect **high drop-off rates** and trigger an alert if user engagement significantly decreases before the final step.

**Scope:**

* Identify pages with **highest and lowest views**.
* Compare views between **critical steps** (e.g., from the first page to the last page).
* If the drop-off rate is significant, **trigger an alert** to indicate potential issues in user flow.
* Capture **device/browser trends** and **time-based fluctuations**.

**Approach:**

**Data Collection:**

* Extract **total page views** per page.
* Capture **session IDs, timestamps, and navigation paths**.

**Drop-off Detection & Alert:**

* Define the **drop-off percentage** as:

Drop-off Rate=(1−Views of Last PageViews of First Page)×100\text{Drop-off Rate} = \left(1 - \frac{\text{Views of Last Page}}{\text{Views of First Page}} \right) \times 100Drop-off Rate=(1−Views of First PageViews of Last Page)×100

* If the drop-off rate exceeds a set threshold (e.g., **50% or more**), an **alert notification** is triggered.

**Insights & Reporting:**

* Provide a **ranked list of pages** from most to least viewed.
* Show **view count trends** (e.g., daily/weekly fluctuations).
* Detect **significant drop-off points** and potential user journey issues.
* Suggest **UX/UI improvements** to optimize the conversion funnel.

**Expected Outcome:**

* **Clear ranking** of pages by views.
* **Time-based trend analysis** for better decision-making.
* **Automated alerts** for major drop-offs in user navigation.
* **Actionable insights** to enhance engagement and conversion rates.

Health Monitoring prompt

**Objective:**

Analyze the health of the system by monitoring user activity across different sites. The goal is to identify trends in successful vs unsuccessful user interactions, detect anomalies, and ensure system stability. Additionally, send an alert notification if the percentage of unsuccessful users exceeds a defined threshold, indicating potential system issues.

**Scope:**

* Track user activity across different sites to determine system health.
* Identify locations with the highest and lowest success rates.
* Detect anomalies in unsuccessful user interactions and correlate them with potential system issues.
* Compare user success rates across different timeframes, locations, and demographics.
* Send an alert if the percentage of unsuccessful users exceeds a predefined threshold.

**Approach:**

**Data Collection:**

* Extract user activity data from the health monitoring database.
* Capture **Sitename, Total Users, Successful Users, Unsuccessful Users, and Health Percent.**
* Record timestamps and session IDs for trend analysis.

**Data Analysis:**

1. **Calculate Health Percentage:**

Health Percent=(Successful Users /Total Users) \*100

Failure percent=(Unsuccessful Users /Total Users)\*100

Identify sites with the highest and lowest health percentages.

1. **Analyze Unsuccessful User Rate:**

Unsuccessful Rate= (Unsuccessful Users/Total Users)× 100

* Detect sites where unsuccessful rates are abnormally high.

1. **Compare Across Segments:**
   * Analyze user success trends based on time (hourly, daily, weekly).
   * Identify sites or regions with consistently low health percentages.
   * Compare trends across device types, browsers, and user demographics.
2. **Investigate Technical Issues:**
   * Identify slow response times, authentication failures, or UI errors contributing to low success rates.
   * Correlate unsuccessful user spikes with system downtime or performance issues.

**Alert Notification:**

1. **Define an alert condition:**

If  (Unsuccessful Users>Threshold), Send Alert

If the unsuccessful rate at any site exceeds a critical threshold, trigger an automated alert.

1. **Alert should include:**
   * Sitename
   * Unsuccessful Users Count
   * Health Percentage
   * Timestamp of the anomaly
2. **Alert recipients:**
   * Notify the relevant technical and support teams for immediate investigation.

**Outcome:**

* **Comprehensive insights** into system health across different sites.
* **Identification of problem areas** affecting user success.
* **Automated alerts** for early detection of critical issues.
* **Actionable recommendations** to improve system stability and user experience.