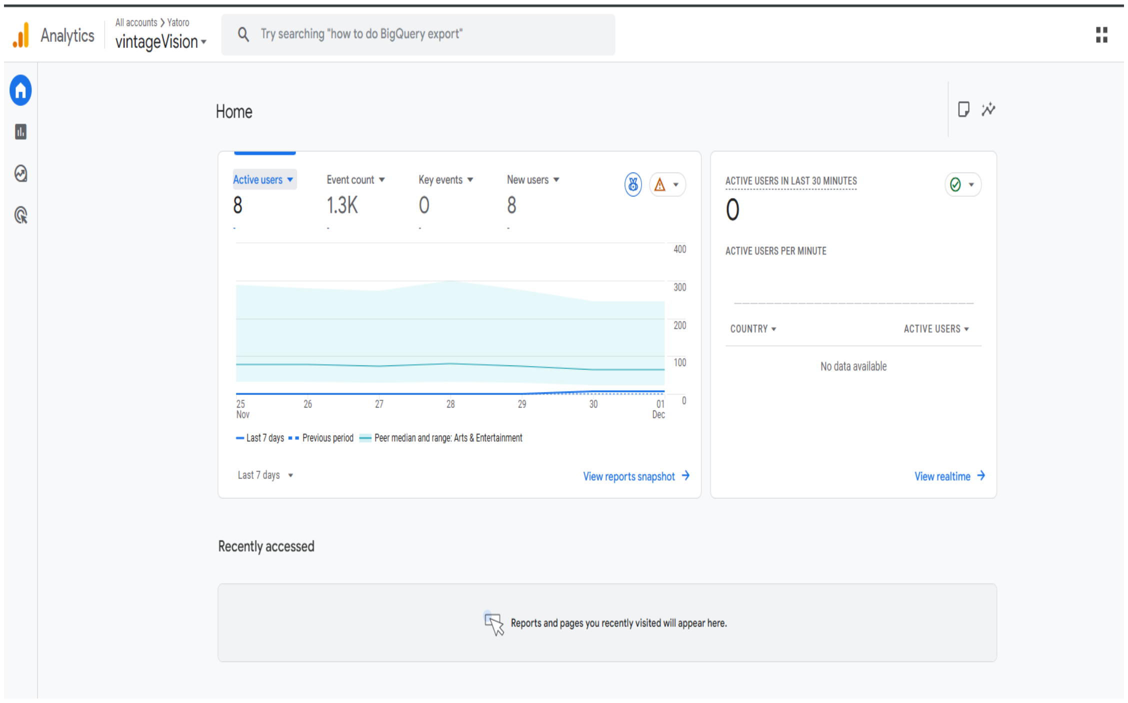
Analytics & Logging Report

SECTION 1 — Google Analytics

This section summarizes how Google Analytics GA4 was implemented on both the client (React.js) and server side (Node.js). Three important analytics metrics were selected: Active Users, Session Duration, and Event Count. Each includes a visualization, an interpretation of the trend, and limitations of the metric.

1.1 — Metric 1: Active Users (Realtime Users)

1.1.a — Visualization

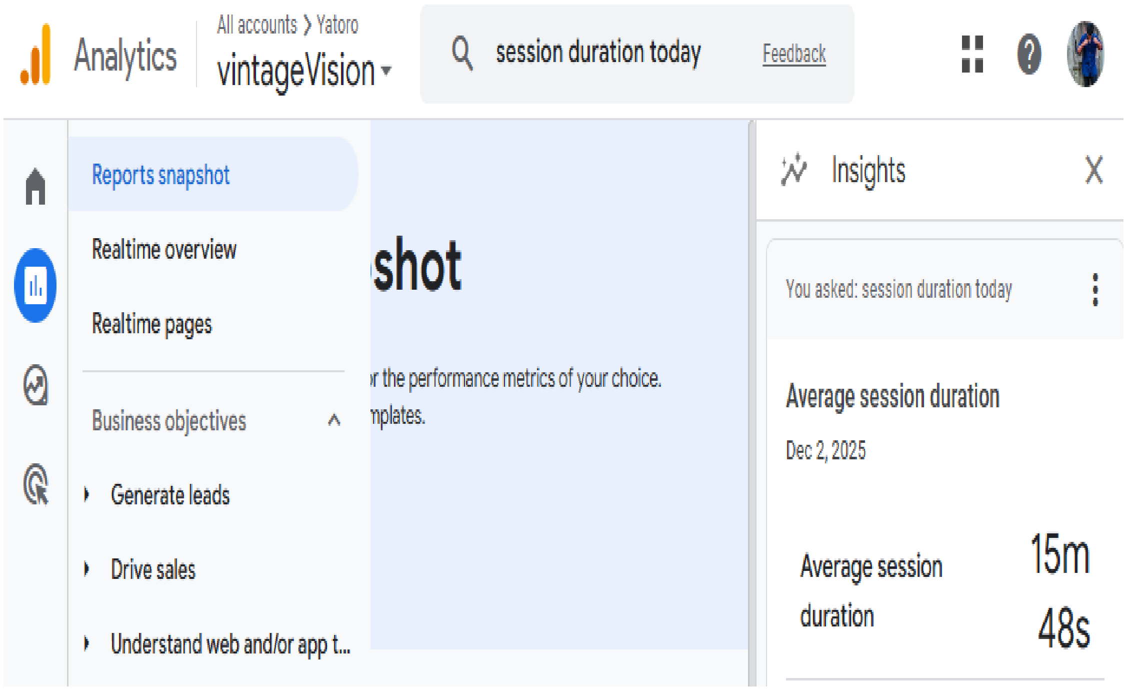


1.1.b — Interpretation: The Active Users metric shows how many users were interacting with the application during a given time window. Peaks occur during increased sharing and testing periods. This indicates user engagement and the success of promotional efforts.

1.1.c — Limitations: Realtime metrics only capture active users within the last few minutes. Users with privacy blockers or disabled cookies may not appear, reducing accuracy.

1.2 — Metric 2: Session Duration

1.2.a — Visualization



You asked: session duration today

Average session duration

Dec 2, 2025

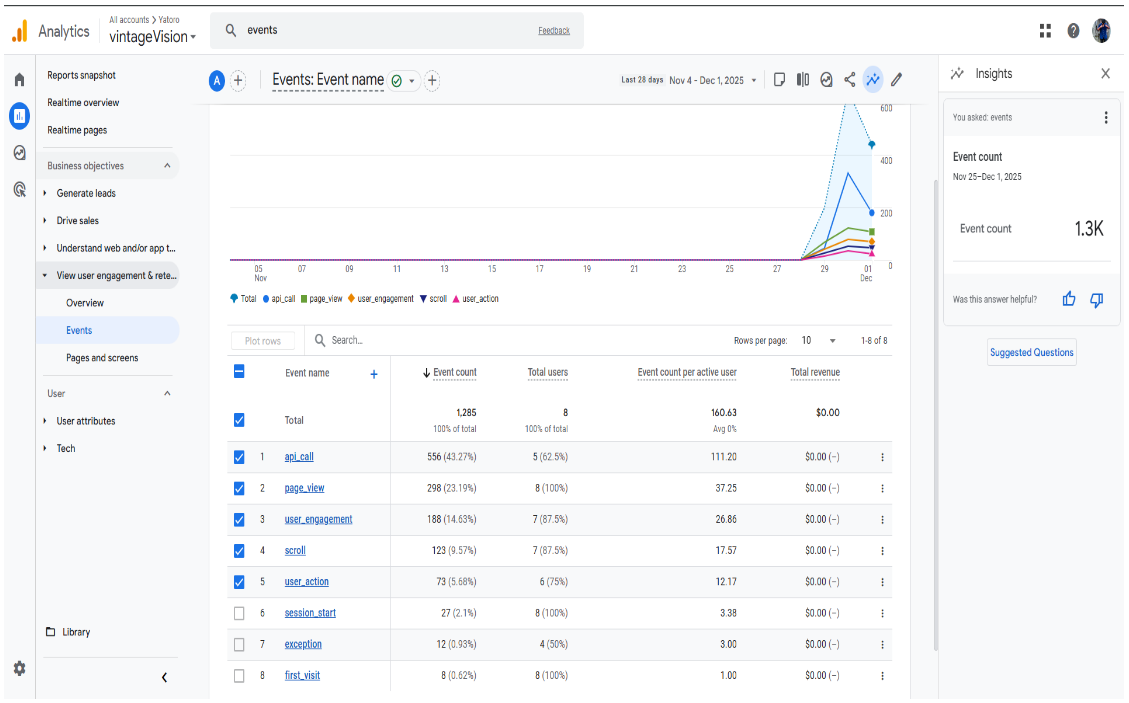
Average session duration	15m 48s

1.2.b — Interpretation: Session Duration measures how long a user stays in the app. Longer durations reflect deeper interest, especially during image uploads and reading AI fashion analysis results.

1.2.c — Limitations: Single-page applications (SPA) may not measure durations accurately. Users leaving tabs open can inflate time values.

1.3 — Metric 3: Event Count (Uploads & API Calls)

1.3.a — Visualization



		Event name	Event count	Total users	Event count per active user	Total revenue
✓		Total	1,285 100% of total	8 100% of total	160.63 Avg 0%	$0.00
✓	1	api_call	556 (43.27%)	5 (62.5%)	111.20	$0.00 (–)
✓	2	page_view	298 (23.19%)	8 (100%)	37.25	$0.00 (–)
✓	3	user_engagement	188 (14.63%)	7 (87.5%)	26.86	$0.00 (–)
✓	4	scroll	123 (9.57%)	7 (87.5%)	17.57	$0.00 (–)
✓	5	user_action	73 (5.68%)	6 (75%)	12.17	$0.00 (–)
	6	session_start	27 (2.1%)	8 (100%)	3.38	$0.00 (–)
	7	exception	12 (0.93%)	4 (50%)	3.00	$0.00 (–)
	8	first_visit	8 (0.62%)	8 (100%)	1.00	$0.00 (–)

1.3.b — Interpretation: Event Count shows how often users perform key interactions such as uploading images or triggering Gemini/Vision API calls. A high event count indicates active exploration and engagement with the application's functionality.

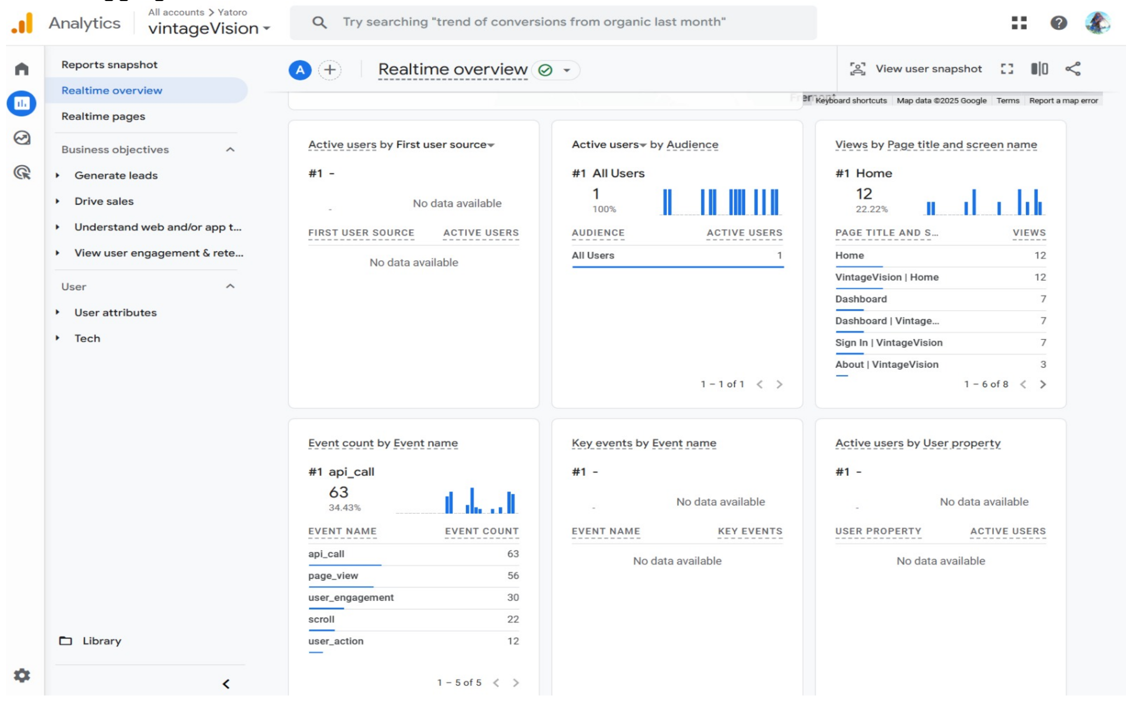
1.3.c — Limitations: Duplicate firing of events due to refreshed pages or retries may inflate counts. Event naming inconsistencies may cause missing or misclassified data.

SECTION 2 — Google Logging

Google Cloud Logging was implemented on both the frontend and backend. The backend logs track API calls, Vision API usage, Gemini requests, and errors, while the frontend logs user interactions and events through Google Analytics.

2.1 — Frontend Implementation (React.js)

Frontend logging uses React GA4 to track user behavior, including page views, file uploads, and button interactions. This ensures that all user actions are captured and contribute to analytics insights.
Frontend Logging Screenshot:



Active users by First user source

#1 -

No data available

FIRST USER SOURCE	ACTIVE USERS
No data available	

Active users by Audience

#1 All Users
1
100%

AUDIENCE	ACTIVE USERS
All Users	1

1 – 1 of 1

Views by Page title and screen name

#1 Home
12
22.22%

PAGE TITLE AND S...	VIEWS
Home	12
VintageVision | Home	12
Dashboard	7
Dashboard | Vintage...	7
Sign In | VintageVision	7
About | VintageVision	3

1 – 6 of 8

Event count by Event name

#1 api_call
63
34.43%

EVENT NAME	EVENT COUNT
api_call	63
page_view	56
user_engagement	30
scroll	22
user_action	12

1 – 5 of 5

Key events by Event name

#1 -

No data available

EVENT NAME	KEY EVENTS
No data available	

Active users by User property

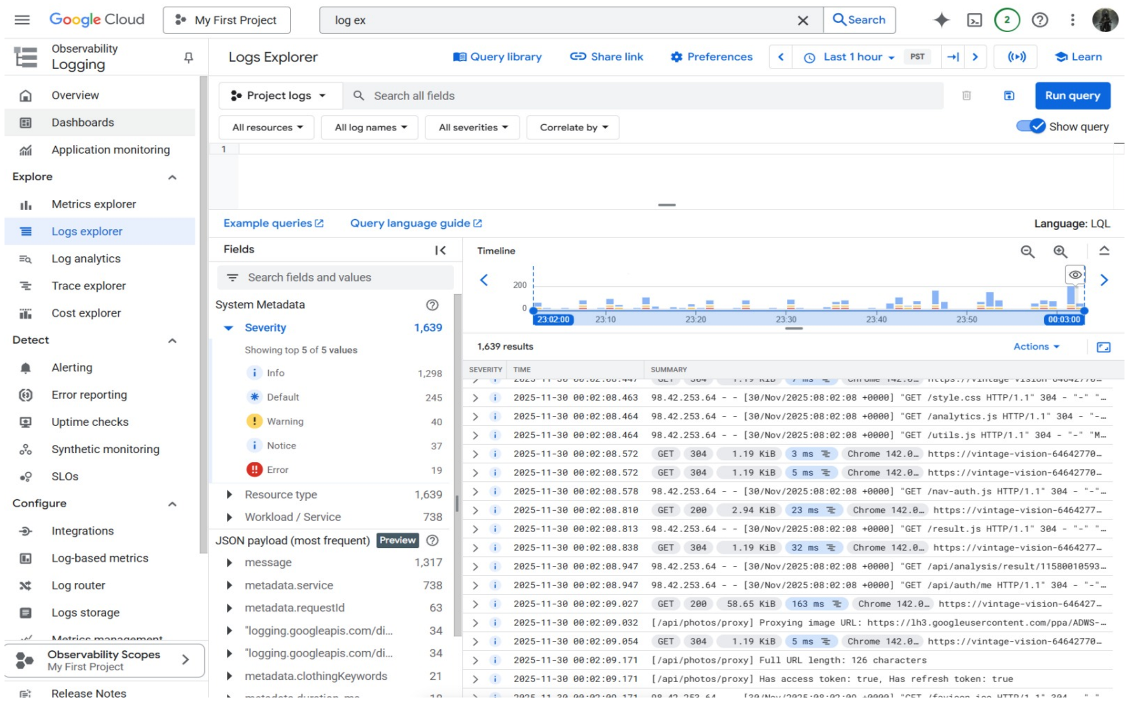
#1 -

No data available

USER PROPERTY	ACTIVE USERS
No data available	

2.2 — Backend Implementation (Express/Node.js)

The backend uses Google Cloud Logging SDK and Winston to log API requests. Every call to Gemini API, Vision API, or user upload is recorded in Cloud Run logs. This helps monitor performance, detect errors, and track usage trends.
Backend Cloud Logging Screenshot:



```
2025-11-30 00:02:08.463 98.42.253.64 - - [30/Nov/2025:08:02:08 +0000] "GET /style.css HTTP/1.1" 304 - "-" "…
2025-11-30 00:02:08.464 98.42.253.64 - - [30/Nov/2025:08:02:08 +0000] "GET /analytics.js HTTP/1.1" 304 - "-…
2025-11-30 00:02:08.464 98.42.253.64 - - [30/Nov/2025:08:02:08 +0000] "GET /utils.js HTTP/1.1" 304 - "-" "M…
2025-11-30 00:02:08.572 GET 304 1.19 KiB 3 ms Chrome 142.0… https://vintage-vision-64642770…
2025-11-30 00:02:08.572 GET 304 1.19 KiB 5 ms Chrome 142.0… https://vintage-vision-64642770…
2025-11-30 00:02:08.578 98.42.253.64 - - [30/Nov/2025:08:02:08 +0000] "GET /nav-auth.js HTTP/1.1" 304 - "-"…
2025-11-30 00:02:08.810 GET 200 2.94 KiB 23 ms Chrome 142.0… https://vintage-vision-6464277…
2025-11-30 00:02:08.813 98.42.253.64 - - [30/Nov/2025:08:02:08 +0000] "GET /result.js HTTP/1.1" 304 - "-" "…
2025-11-30 00:02:08.838 GET 304 1.19 KiB 32 ms Chrome 142.0… https://vintage-vision-6464277…
2025-11-30 00:02:08.947 98.42.253.64 - - [30/Nov/2025:08:02:08 +0000] "GET /api/analysis/result/11580010593…
2025-11-30 00:02:08.947 98.42.253.64 - - [30/Nov/2025:08:02:08 +0000] "GET /api/auth/me HTTP/1.1" 304 - "-"…
2025-11-30 00:02:09.027 GET 200 58.65 KiB 163 ms Chrome 142.0… https://vintage-vision-646427…
2025-11-30 00:02:09.032 [/api/photos/proxy] Proxying image URL: https://lh3.googleusercontent.com/ppa/ADWS-…
2025-11-30 00:02:09.054 GET 304 1.19 KiB 5 ms Chrome 142.0… https://vintage-vision-64642770…
2025-11-30 00:02:09.171 [/api/photos/proxy] Full URL length: 126 characters
2025-11-30 00:02:09.171 [/api/photos/proxy] Has access token: true, Has refresh token: true
```