

---

# CS651 - Web Systems

## Vintage Vision

Team 8 - Mesmer Alfs, Haoran Li, Sravan  
Kumar and Navin Ramani

---

---

# Overview

Web Application hosted on Google GAE to analyze user-selected photographs from their google photos account. The analysis will include:

- Era or time when it was in fashion
  - Type of fashion
  - Explanation of the style
  - Clickable search queries and
  - Shopping tips
-

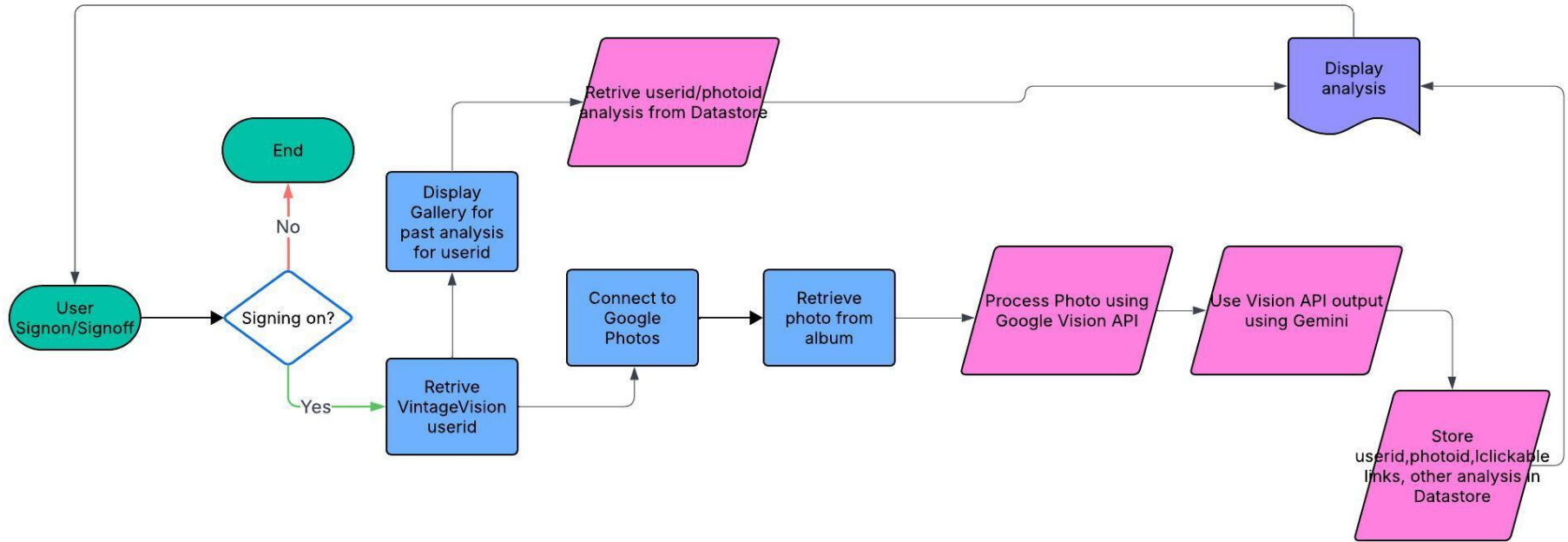
---

# Technology

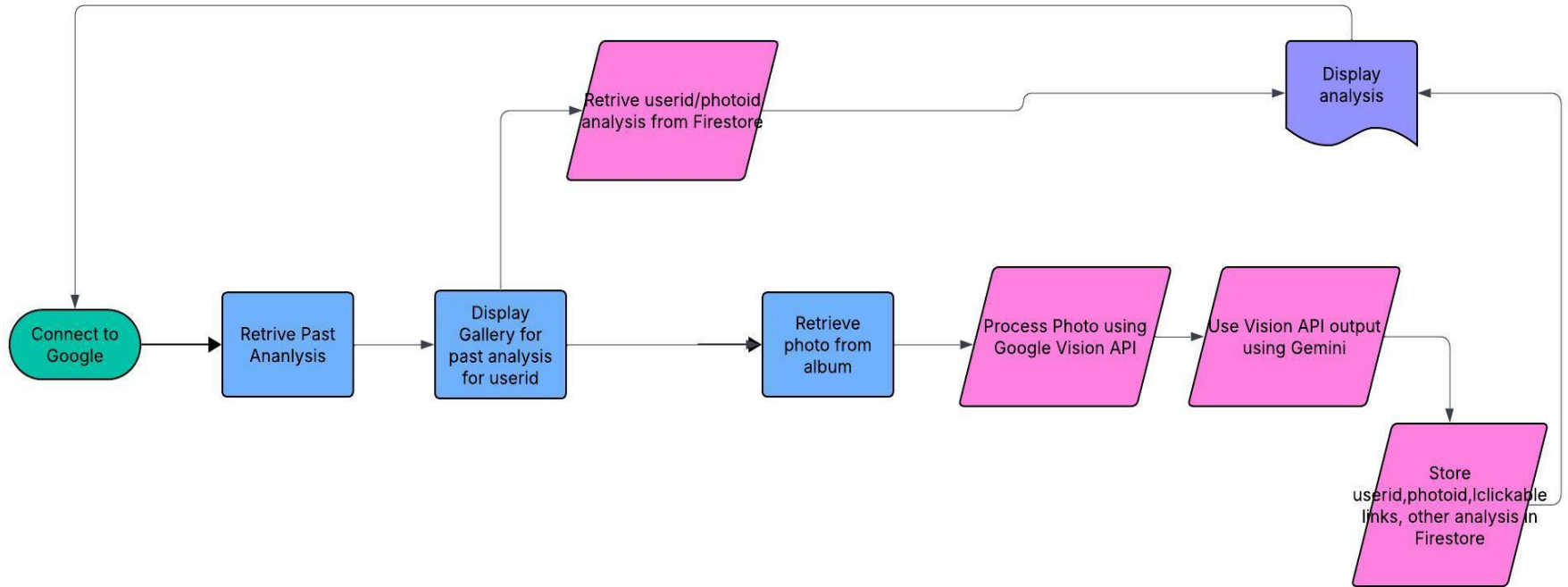
The app was built using the following technology:

- HTML, CSS, Javascript, React, [Node.Js](#), Express
  - Google OAuth 2.0
  - Google APIs including photopicker, Vision, Gemini, Firestore, Axios
  - Hosted on Google App Engine Standard Environment
-

# Initial Proposal



# Implementation



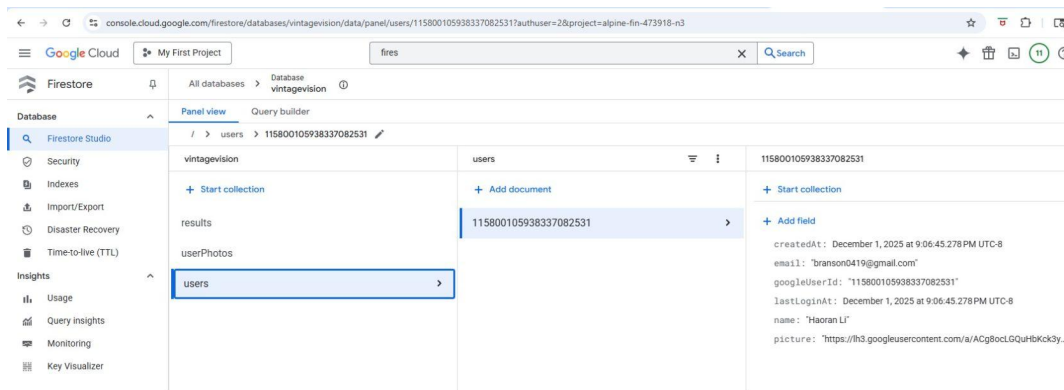
---

# Authentication Flow

Steps:

1. User signs in to their google account
  2. Google OAuth end-point gets confirmation for access to user profile and photos
  3. Http only cookies are stored and front end loads dashboard
  4. User can view previous results or pick new photos
-

# Firestore users collection



# Firestore users collection

vintagevision	users	116611520161433500772
+ Start collection	+ Add document	+ Start collection
results	115800105938337082531	+ Add field
userPhotos	116611520161433500772	createdAt: December 1, 2025 at 10:17:52.402 PM UTC-8
users		email: "vinapptester@gmail.com"
		googleUserId: "116611520161433500772"
		lastLoginAt: December 1, 2025 at 10:17:52.402 PM UTC-8
		name: "Vinapp Tester"
		picture: "https://lh3.googleusercontent.com/a/ACg8ocLmv0dCwnaS..."



# userPhotos Collection

google.com/firestore/databases/vintagevision/data/panel/userPhotos/IFEJDHFGGcZUuytrKFjK?authuser=2&project=alpine-fin-473918-n3

y First Project fires X Search

All databases > Database vintagevision ⓘ

Panel view Query builder

/ > userPhotos > IFEJDHFGGcZUuytrKFjK

vintagevision	userPhotos	IFEJDHFGGcZUuytrKFjK
+ Start collection	+ Add document	+ Start collection
results	IFEJDHFGGcZUuytrKFjK >	+ Add field
userPhotos >		baseUrl: "https://lh3.googleusercontent.com/ppa/ADWS-gdulNx0EMO..." createdAt: December 1, 2025 at 9:07:04.393 PM UTC-8 filename: "3.jpg" mimeType: "image/jpeg" photoId: "AGINBGUCpYVEGD_ZN00558s03vHLGk6sELVQI7s0FLKjca..." sessionId: "cb2bd10a-460b-44d0-ae2b-d9cd8ddb22fc" source: "google_photos_picker" status: "pending" type: "PHOTO"
users		

# results collection

Screenshot of the Google Cloud Firestore console showing the 'results' collection in the 'vintagevision' database.

The browser address bar shows the URL: `google.com/firestore/databases/vintagevision/data/panel/results/115800105938337082531_1764652043201_cq3glxh?authuser=2&project=alpine-fin-473918-n3`.

The console interface includes a search bar with 'fires' and a 'Search' button. The database is 'vintagevision'.

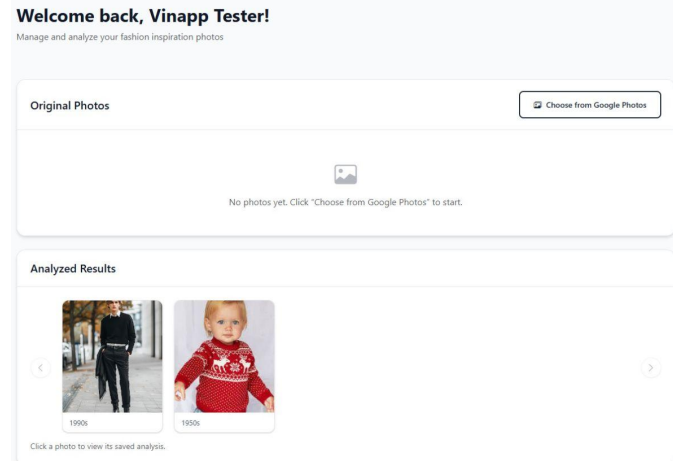
The breadcrumb path is: `/ > results > 115800105938337082531_1764652043201_cq3glxh`.

vintagevision	results	115800105938337082531_1764652043201_cq3glxh
<a href="#">+ Start collection</a>	<a href="#">+ Add document</a>	<a href="#">+ Start collection</a>
<b>results</b>	<b>115800105938337082531_1764652043201_cq3glxh</b>	<a href="#">+ Add field</a>
userPhotos		analyzedAt: December 1, 2025 at 9:07:23.201 PM UTC-8
users		baseUri: "https://lh3.googleusercontent.com/ppa/ADWS-gdulNx0E..."
		<b>geminResult</b>
		era_primary: "1970s"
		rationale: "The short length, sheer ruffled bodice, and flowy bell..."
		<b>search_queries</b>
		en

---

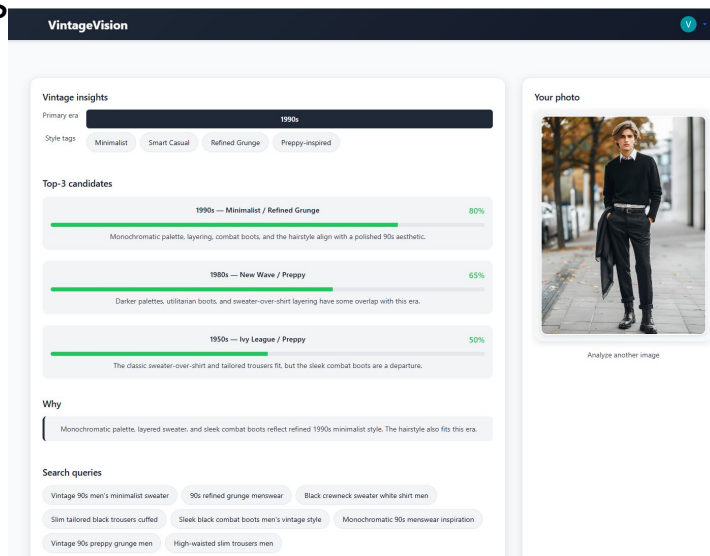
# Dashboard

After authentication, user lands on the dashboard



# Sample Analysis

## Analysis



---

# Analysis Pipeline

Vision API does image detection, prominent colors, and extracts:

- Patterns, Colors, Objects

Gemini uses this information, along with image and prompt to produce:

- JSON formatted results
-

---

# Firestore

There are three JSON object collections in Firestore for this app:

1. User - authenticated users
  2. Userphotos - userid, photo url, timestamp
  3. Results - contains Vision and Gemini results
-

---

# Caveats and Enhancements

- Caveat - Google photo urls stored in results expire after a certain amount of time leaving the dashboard with blank photo spaces for analyzed images.
  - Enhancement Needed - When user logs in, delete documents from results and userPhotos to eliminate stale urls.
-