**SUBMITTED BY: Muhammad Hasan**

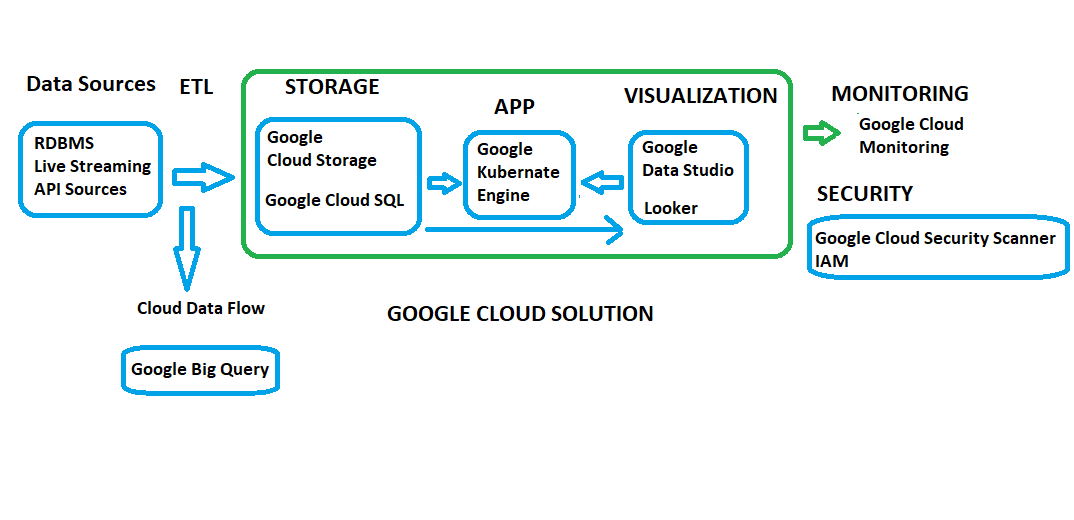
**FINAL PROJECT REPORT**

**Project Statement:**

As a client with a social media app, I require a comprehensive solution architecture for cloud-based data management. The goal is to optimize the app's data storage, processing, and retrieval capabilities, ensuring scalability, reliability, and security. The solution should leverage cloud technologies and services to enable efficient data handling, analysis, and integration with other systems. The architecture should address data governance, data privacy, and compliance requirements, while also considering performance optimization and cost-effectiveness. Ultimately, the aim is to enhance the overall user experience, streamline data workflows, and enable future growth and innovation within the app.

**Proposed Solution:**

Google Cloud Platform is used for the given problem. Please refer below architecture diagram.



**Data Types and Sources**

In Social Media APP we have to deal with Structured, Semi Structured and Non Structured data. We have user profile information, pictures, videos, live posts etc. These data can be present in data bases or can be ingested in real time.

**Data Pipeline (ETL)**

Extract, Transform, and Load (ETL) refers to a series of processes that map our data’s journey from its sources to the warehouse. The implementation of ETL involves bringing in different varieties of data from different sources, curating the data, and loading the curated data into another data source.

**Google Data Flow**

[Dataflow](https://cloud.google.com/dataflow) is a server less, fast, and cost-effective service for processing both stream and batch data. Just like Data Fusion, Google takes care of infrastructure provisioning and cluster management. If your organization deals with data that is being continuously generated together with that which has been stored over a period, Dataflow would suit your needs. It allows users to build pipelines using Apache Beam SDK together with either Python or Java. These pipelines are then deployed and executed as Dataflow jobs. Dataflow recruits virtual machines to execute the data processing. You don’t have to worry if your traffic pattern is irregular, Dataflow seamlessly auto scale to increase the number of instances when traffic spikes.

**Google Big Query**

Google BigQuery is a server less, highly scalable data warehouse that comes with a built-in query engine. The query engine is capable of running SQL queries on terabytes of data in a matter of seconds, and petabytes in only minutes. You get this performance without having to manage any infrastructure and without having to create or rebuild indexes.

We integrate Big Query in our Solution for faster query operation from data warehouse.

**Data Storage**

**Google Cloud Storage**

Google Cloud Storage is a service within the [Google Cloud Platform](https://www.techtarget.com/searchcloudcomputing/definition/Google-Cloud-Platform). It provides unified object storage for live or archived data. [Objects](https://www.techtarget.com/searchstorage/definition/object-storage) stored in Google Cloud Storage are grouped into buckets. Buckets are [containers](https://www.techtarget.com/whatis/definition/storage-container) within the cloud that can be individually assigned to storage classes. Google Cloud Storage is an enterprise public [cloud storage](https://www.techtarget.com/searchstorage/definition/cloud-storage) platform that can house large unstructured data sets.

We used Google Cloud Storage to store Non Structured data from Social Media App.

**Google cloud SQL**

Cloud SQL is a fully-managed database service that helps us set up, maintain, manage, and administer our relational databases on Google Cloud Platform.

We can use Cloud SQL with [MySQL](https://cloud.google.com/sql/docs/mysql), [PostgreSQL](https://cloud.google.com/sql/docs/postgres), or [SQL Server](https://cloud.google.com/sql/docs/sqlserver).

We used Google Cloud SQL to store structured data from our Social Media App.

**Application**

**Google Kubernetes Engine (GKE)**

Google Kubernetes Engine (GKE), a managed [Kubernetes](https://kubernetes.io/) service that you can use to deploy and operate containerized applications at scale using Google's infrastructure. GKE is a Google-managed implementation of the [Kubernetes](https://kubernetes.io/) open source container orchestration platform.

GKE is ideal if you need a platform that lets you configure the infrastructure that runs your containerized apps, such as networking, scaling, hardware, and security. GKE provides the operational power of Kubernetes while managing many of the underlying components, such as the control plane and nodes.

**How GKE works**

A GKE environment consists of *nodes*, which are [Compute Engine virtual machines (VMs)](https://cloud.google.com/compute), that are grouped together to form a *cluster*. You package your apps (also called *workloads*) into containers. You deploy sets of containers as *Pods* to your nodes. You use the Kubernetes API to interact with your workloads, including administering, scaling, and monitoring.

Kubernetes clusters have a set of management nodes called the *control plane*, which run system components such as the Kubernetes API server. In GKE, Google manages the control plane and system components for you. In Autopilot mode, which is the recommended way to run GKE, Google also manages your worker nodes. Google automatically upgrades component versions for improved stability and security, ensuring high availability, and ensuring integrity of data stored in the cluster's persistent storage.

**Modes of operation**

GKE has the Autopilot and Standard *modes of operation*, which offer you different levels of flexibility, responsibility, and control. We recommend the fully-managed [Autopilot mode](https://cloud.google.com/kubernetes-engine/docs/concepts/autopilot-overview), in which Google Cloud manages your nodes for you and provides a workload-focused, cost-optimized, production-ready experience.

**Data Visualization**

**Google Data Studio**

Google Data Studio is a web-based data visualization tool that helps users build customized dashboards and easy-to-understand reports. It helps in tracking key KPI's for customers, visualizing trends, and comparing performances over time.

**Benefits of Using Google Data Studio**

* Easier set up with simple reporting
* Creates customized and engaging reports.
* Pulls multiple data sources into a single report.
* Makes sharing and collaboration simple.
* Creates up to five customized reports that are free of cost.
* Helps the user connect to 150+ data sources.
* Monitors campaigns effectively.
* Reduces turnaround time on reporting.

### **Connect Data Sources**

### Data sources act as a link to connect a Data Studio report to a collection of underlying data.

* Each data source has unique features and pre-built connectors that help users access and connect efficiently with the Data Studio.
* Data Studio supports 500+ data sources; some of the most popular sources are:
  + Google Analytics
  + Google Ads
  + Google Search Console
  + BigQuery
  + YouTube Analytics
  + PostgreSQL
  + Search Ads 360

**Looker Studio**

Google rebranded “Google Data Studio” to “Looker Studio” after [acquiring Looker](https://cloud.google.com/blog/topics/inside-google-cloud/google-completes-looker-acquisition) earlier in February 2021. Looker and Looker Data Studio are both Google tools with a lot in common. They are both strong BI solutions that provide data analytics, integration, and visualization to assist enterprises in overcoming business challenges.

## Tell your story in data

* Visualize your data through highly configurable charts and tables.
* Easily connect to a variety of data sources.
* Share your insights with your team or with the world.
* Collaborate on reports with your team.
* Speed up your report creation process with built-in sample reports.

**Visualize your data**

Looker Studio is a free tool that turns your data into informative, easy to read, easy to share, and fully customizable dashboards and reports. Use the drag and drop report editor to:

* Tell your data story with charts, including line, bar, and pie charts, geo maps, area and bubble graphs, paginated data tables, pivot tables, and more.
* Make your reports interactive with viewer filters and date range controls. The data control turns any report into a flexible template report that anyone can use to see their own data.
* Include links and clickable images to create product catalogs, video libraries, and other hyperlinked content.
* Annotate and brand your reports with text and images.
* Apply styles and color themes that make your data stories works of data visualization art.

**Connect to your data**

With Looker Studio, you can easily report on data from a wide variety of sources, without programing. In just a few moments, you can connect to data sets such as:

* Databases, including BigQuery, MySQL, and PostgreSQL
* Google Marketing Platform products, including Google Ads, Analytics, Display & Video 360, Search Ads 360
* Google consumer products, such as Sheets, YouTube, and Search Console
* Flat files via CSV file upload and Google Cloud Storage
* Social media platforms such as Facebook, Reddit, and Twitter
* Blended data from any combination of related sources

## Share and collaborate

It's easy to share your insights with individuals, teams, or the world. Invite others to view or edit your reports, or send them links in scheduled emails. To tell your data stories as broadly as possible, you can embed your reports in other pages, such as Google Sites, blog posts, marketing articles, and annual reports.

When you share a Looker Studio file with another editor, you can work it together in real time as a team.

**Google Studio VS Looker Studio**

## Summarized Comparison Table

|  |  |  |
| --- | --- | --- |
| **Features** | **Data Studio** | **Looker** |
| Price | Free | From USD3K per month |
| Data Source Integration | SQL & non-SQL | SQL |
| Data Source Merging | Yes | Yes |
| Fully-Hosted on Cloud | Yes | Yes |
| On-Prem Deployment | No | Yes |
| Linux / Windows Deployment | No | No |
| Basic Data Modeling | Yes | Yes |
| Advanced Data Modeling | No | Yes |
| Predictive Analytics | No | Yes |
| Programming Languages Support | Yes | Yes |
| Embedded Analytics | Yes | Yes |
| Free Version | Yes | No |
| SQL Support | Yes | Yes |
| API Access | Yes | Yes |
| Custom Visualization | Yes | Yes |

**Data Monitoring**

**Google Cloud Monitoring**

This document provides an overview of Cloud Monitoring, which is one part of Google Cloud's operations suite. Cloud Monitoring is integrated with most Google Cloud services, and it automatically collects and stores performance information about those services. It can also collect system and application metrics from third-party applications. The data visualization and analysis tools provided by Cloud Monitoring help you answer important questions like the following:

* What is the load on my service?
* Is my website responding correctly?
* Is my service performing well?

Cloud Monitoring provides Google Cloud console and API support for most of its services, and the Cloud Monitoring API reference pages, such as the page [alertPolicies.list](https://cloud.google.com/monitoring/api/ref_v3/rest/v3/projects.alertPolicies/list), let you experiment with API calls directly from the reference page.

This document is intended for developers and system administrators who need to monitor the performance of a service or system.

## Monitor the load on a service

To understand the current load on a service, or to view the performance data of your service for the past month, use the [charts and dashboards](https://cloud.google.com/monitoring/dashboards) tools. You can chart and monitor any (numeric) metric data that your Google Cloud project collects, including the following:

* System metrics generated by Google Cloud services. These metrics provide information about how the service is operating. For example, Compute Engine reports more than 25 unique metrics for each virtual machine (VM) instance. For a complete list of metrics, see [Google Cloud metrics](https://cloud.google.com/monitoring/api/metrics_gcp).
* [System and application metrics](https://cloud.google.com/monitoring/api/metrics_opsagent) that the [Google Cloud operations suite agents](https://cloud.google.com/monitoring/agent) collect. These metrics provide additional information about system resources and applications running on Compute Engine instances. Optionally, you can configure the agent to collect metrics from [third-party plugins](https://cloud.google.com/monitoring/agent/ops-agent/third-party) such as Apache or Nginx web servers, or MongoDB or PostgreSQL databases.
* [User-defined metrics](https://cloud.google.com/monitoring/custom-metrics) that your service writes by using the [Cloud Monitoring API](https://cloud.google.com/monitoring/docs/apis) or by using a library like [Open Telemetry](https://opentelemetry.io/).
* [Log-based metrics](https://cloud.google.com/logging/docs/logs-based-metrics), which collect numeric information about the logs written to [Cloud Logging](https://cloud.google.com/logging/docs). Google-defined log-based metrics include counts of errors that your service detects and the total number of log entries received by your Google Cloud project. You can also define log-based metrics. For example, you might create a metric that counts the number of 404 Not Found errors for an application deployed to App Engine.

To visualize your data to see trends, identify outliers, and view other details about your data, you can use the following tools:

* [Google Cloud dashboards](https://cloud.google.com/monitoring/charts/predefined-dashboards): Cloud Monitoring automatically creates these dashboards based on the resources used by your Google Cloud project.

For example, when a Google Cloud project contains Compute Engine VM instances, dashboards for those VM instances and disks are created automatically. By using the **VM instances** dashboard, you can view details such as memory and disk usage, identify IP addresses, and identify which VMs are dropping network packets. This dashboard also displays information about your usage of the Cloud Monitoring agent and provides suggestions for instrumentation.

* [Custom dashboards](https://cloud.google.com/monitoring/charts/dashboards): You create or [install](https://cloud.google.com/monitoring/dashboards/dashboard-templates) these dashboards. Custom dashboards let you define what data you want to view and how to view that data. For example, you can display metric data, alerting policies, and logs stored in your Google Cloud project. You can display time-series data on a chart, with a gauge or scorecard, or in tabular form. Dashboards also support text widgets. You can create a custom dashboard with the [Dashboards API](https://cloud.google.com/monitoring/dashboards/api-dashboard) or with the [Google Cloud console](https://cloud.google.com/monitoring/charts/dashboards).
* Charts: You can add charts to a custom dashboard or you can use [Metrics Explorer](https://cloud.google.com/monitoring/charts/metrics-explorer), which is a charting tool designed to let you quickly chart and explore time-series data. You can [save charts created with Metrics Explorer](https://cloud.google.com/monitoring/charts/metrics-explorer#save) to a custom dashboard.

## Monitor website availability

To monitor whether a website is responding, configure an [uptime check](https://cloud.google.com/monitoring/uptime-checks). These checks periodically probe your service in a way that mimics how your customers access your service, and then they record the success and latency of the probe.

To view information about your uptime checks, Cloud Monitoring provides a dashboard that summarizes the status of each uptime check, and for each check, it provides a dashboard with detailed information. The detail view for an uptime check displays the success or failure of the response and the latency of the response, along with details about the uptime check

## Get notified when a service isn't performing well

To be notified when the performance of a service meets criteria that you define, create an [alerting policy](https://cloud.google.com/monitoring/alerts). For example, you can create an alerting policy that notifies your on-call team when the 90th percentile of the latency of HTTP 200 responses from your service exceeds 100 ms. Similarly, you can be notified when an [uptime check](https://cloud.google.com/monitoring/uptime-checks) fails.

**Data Security**

**Google Cloud Security Scanner**

Web Security Scanner identifies security vulnerabilities in your App Engine, Google Kubernetes Engine (GKE), and Compute Engine web applications. It crawls your application, following all links within the scope of your starting URLs, and attempts to exercise as many user inputs and event handlers as possible. Currently, Web Security Scanner only supports public URLs and IPs that aren't behind a firewall.

Web Security Scanner currently supports the App Engine standard environment and App Engine flexible environments, Compute Engine instances, and GKE resources.

Web Security Scanner is designed to complement your existing secure design and development processes. To avoid distracting you with false positives, Web Security Scanner errs on the side of under reporting and doesn't display low confidence alerts. It does not replace a manual security review, and it does not guarantee that your application is free from security flaws.