

## UNIT –V

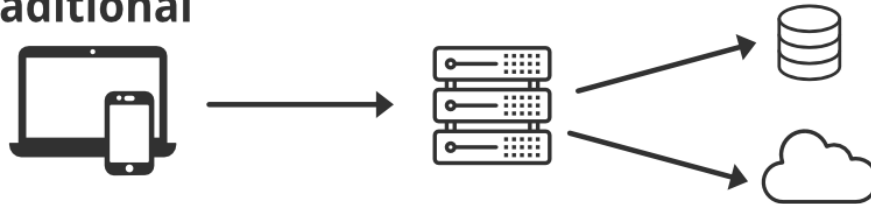
### STORAGE, INDEXING, QUERY PROCESSING, AND EMERGING TRENDS

#### Firestore - Introduction

Firestore, a product of Google, is a powerful platform that enables developers to build, manage, and scale applications with ease. It simplifies app development by offering a secure and efficient backend, eliminating the need for server-side programming. Firestore supports multiple platforms, including Android, iOS, web, and Unity, making it a versatile solution for developers.

With features like real-time cloud storage, authentication, and NoSQL database support, Firestore ensures seamless data management and synchronization. Its cloud-based infrastructure allows for faster development, enhanced security, and effortless scalability, empowering developers to focus on creating great user experiences.

#### Traditional



#### Firestore



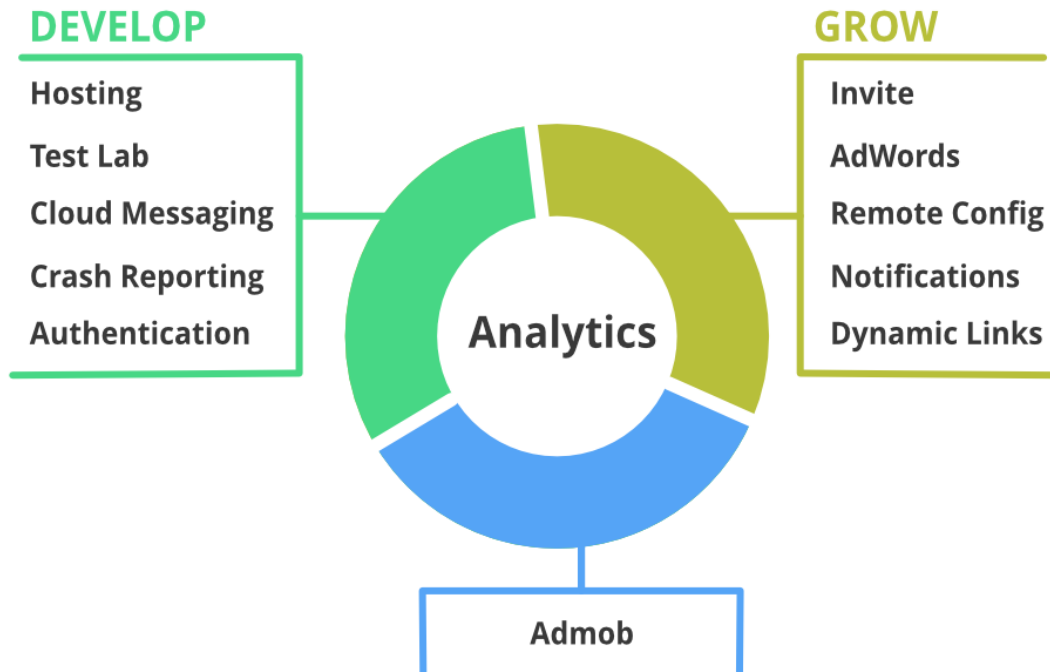
#### A brief history:

Firestore was originally launched as an online chat service provider under the name Envelope, offering real-time communication through an API for websites. However, developers began utilizing it beyond chat functionality, leveraging its capabilities to synchronize application data, such as game states, in real-time across users.

Recognizing this broader potential, James Tamplin and Andrew Lee, the founders of Envelope, separated the underlying architecture from its chat system. They refined and expanded its functionality, leading to the creation of Firestore in 2012—a platform designed to provide real-time database and backend services for modern app development.

#### Features:

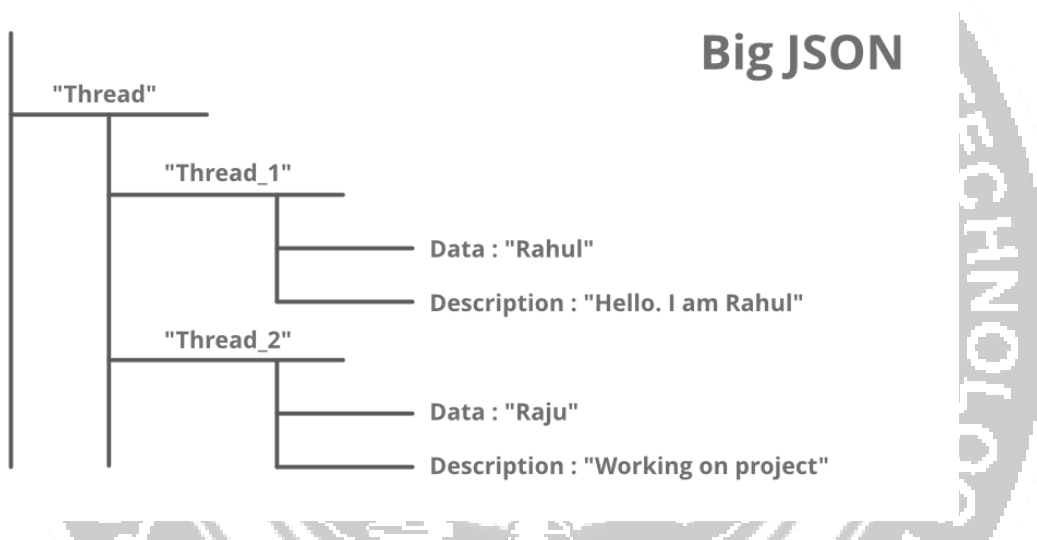
Mainly there are 3 categories in which firestore provides its services.



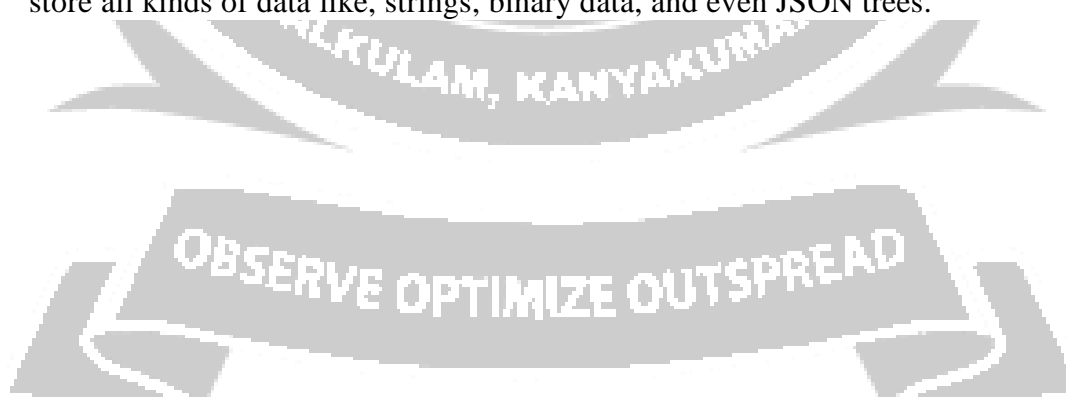
Build better applications

This feature mainly includes backend services that help developers to build and manage their applications in a better way. Services included under this feature are :

- **Realtime Database:** The Firebase Realtime Database is a cloud-based NoSQL database that manages your data at the blazing speed of milliseconds. In simplest term, it can be considered as a big JSON file.



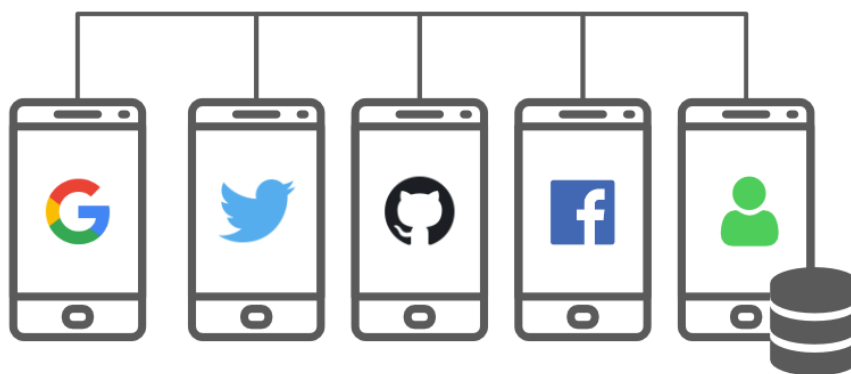
- **Cloud Firestore:** The cloud Firestore is a NoSQL document database that provides services like store, sync, and query through the application on a global scale. It stores data in the form of objects also known as Documents. It has a key-value pair and can store all kinds of data like, strings, binary data, and even JSON trees.



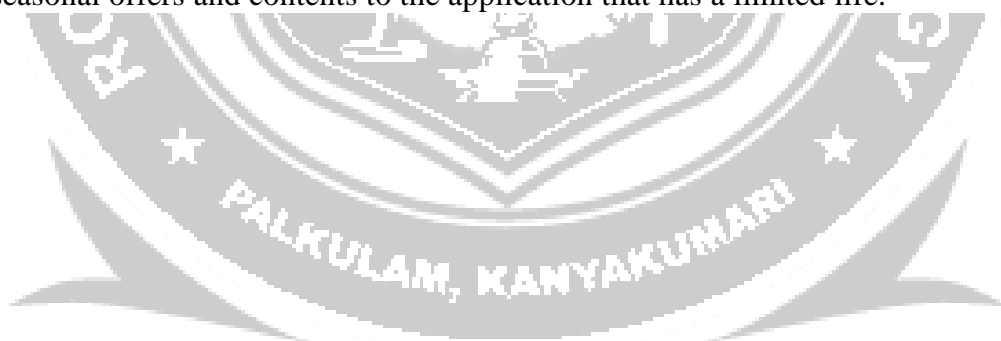


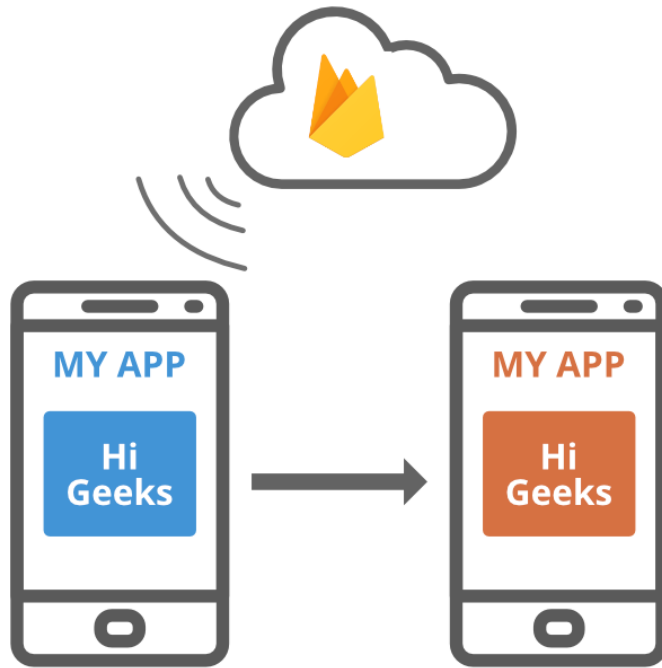
- **Authentication:** Firebase Authentication service provides easy to use UI libraries and SDKs to authenticate users to your app. It reduces the manpower and effort required to develop and maintain the user authentication service. It even handles tasks like merging accounts, which if done manually can be hectic.

## Firestore

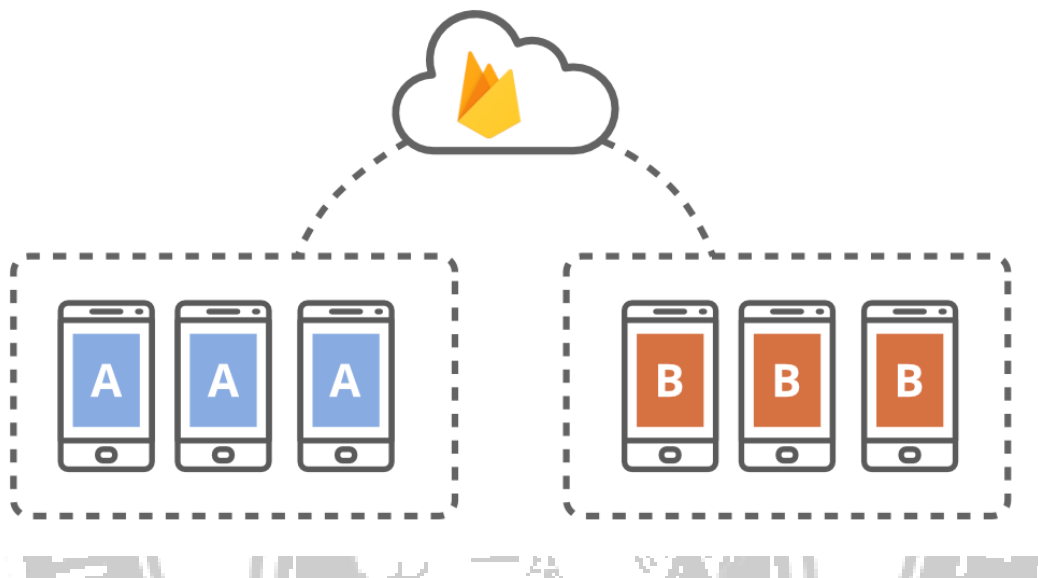


- **Remote Config:** The remote configuration service helps in publishing updates to the user immediately. The changes can range from changing components of the UI to changing the behavior of the applications. These are often used while publishing seasonal offers and contents to the application that has a limited life.





- **Hosting:** Firebase provides hosting of applications with speed and security. It can be used to host Static or Dynamic websites and microservices. It has the capability of hosting an application with a single command.
- **Firebase Cloud Messaging (FCM):** The FCM service provides a connection between the server and the application end users, which can be used to receive and send messages and notifications. These connections are reliable and battery-efficient.



Improve app quality:

Here majorly all the application performance and testing features are provided. All the features required to check and manage before launching your application officially are provided in this section. Services included are:



- **Crashlytics:** It is used to get real-time crash reports. These reports can further be used to improve the quality of the application. The most interesting part of this service is that it gives a detailed description of the crash which is easier to analyze for the developers.
- **Performance monitoring:** This service gives an insight to the performance characteristics of the applications. The performance monitoring SDK can be used to receive performance data from the application, review them, and make changes to the application accordingly through the Firebase console.
- **Test lab:** This service helps to test your applications on real and virtual devices provided by Google which are hosted on the Google Datacenters. It is a cloud-based app-testing infrastructure which supports testing the application on a wide variety of devices and device configurations
- **App Distribution:** This service is used to pre-release applications that can be tested by trusted testers. It comes in handy as decreases the time required to receive feedback from the testers.

Grow your app:

This feature provides your application analytics and features that can help you to interact with your user and make predictions that help you to grow your app. Services provided are:

- **Google analytics:** It is a Free app measurement service provided by Google that provides insight on app usage and user engagement. It serves unlimited reporting for up to 500 distinct automatic or user-defined events using the Firebase SDK.
- **Predictions:** Firebase Predictions uses machine learning to the application's analytics data, further creating dynamic user segments that are based on your user's behavior. These are automatically available to use for the application through Firebase Remote Config, the Notifications composer, Firebase In-App Messaging, and A/B Testing.
- **Dynamic Links:** Deep Links are links that directly redirects user to specific content. Firebase provides a Dynamic linking service that converts deep links into dynamic links which can directly take the user to a specified content inside the application. Dynamic links are used for converting web users to Native app users. It also increases the conversion of user-to-user sharing. In addition, it can also be used to integrate social media networks, emails, and SMS to increase user engagement inside the application.
- **A/B Testing:** It is used to optimize the application's experience by making it run smoothly, scaling the product, and performing marketing experiments.

### Pros and Cons of Using Firebase:

Below listed are the advantages and disadvantages of using a Firebase backend:

Pros:

- Firebase provides a simple and developer-friendly interface, making it **easy to integrate into apps** without backend expertise.
- Firebase offers **real-time data synchronization**, making it ideal for chat apps, multiplayer games, and collaborative tools.
- Works seamlessly on Android, iOS, web, and Unity, enabling **multi-platform app development**.
- Provides **secure authentication** via Google, Facebook, Twitter, email/password, and phone authentication without extra coding.
- Google Cloud infrastructure ensures Firebase **scales** automatically based on user demand.
- **Serverless backend execution** allows running custom logic without managing servers.
- The Spark Plan (**free tier**) offers generous limits for small projects.
- Firebase provides **secure cloud storage** and **fast web hosting** with built-in SSL.
- Offers **real-time crash reporting** and detailed **user analytics** for better app performance and insights.

Cons:

- Firebase uses Firestore (NoSQL), which **lacks complex query support like SQL** databases, making it challenging for relational data handling.
- The free tier is great for small apps, but **costs can escalate** quickly as data storage, reads/writes, and users increase.
- Firebase is a Google product, so **migrating to another backend can be complex** if you need to switch later.
- While Cloud Functions offer flexibility, Firebase **doesn't provide full control over server-side logic** compared to traditional backends.
- Firebase security rules require careful management, and **improper setup can lead to data leaks** or unauthorized access.
- Although Firestore supports **offline mode**, its functionality **may not always be as reliable** as traditional databases.
- **Queries in Firestore are limited** (e.g., no full-text search or advanced filtering like SQL).
- Since Firebase is deeply **integrated with Google Cloud**, any service downtime can impact your app.

### Companies using Firebase

Below are some reputable organizations that rely on a firebase backend for its functioning: The New York Times, Alibaba.com, Gameloft, Duolingo, Trivago, Venmo, Lyft.

#### Pricing:

There are 2 plans available. Spark plan is initially free but as your user base grows you might need to upgrade to blaze plan. Firebase asks you to pay as you go. For most developers who are just getting started and are on a learning path, they are covered by spark plan.

### Difference between Data Lake and Data Warehouse

#### Data Lake

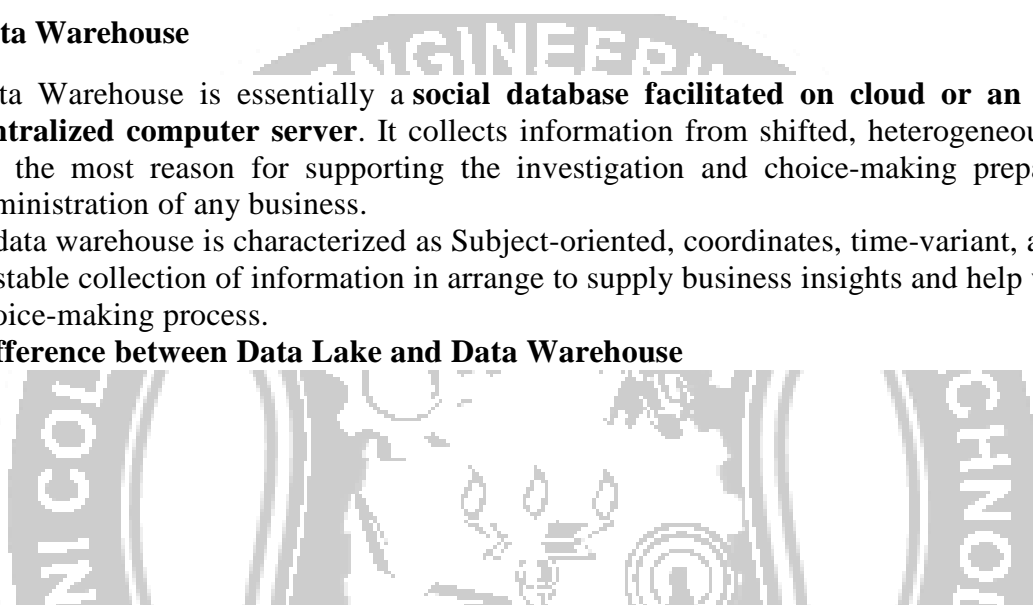
Data Lake is the **concept where all sorts of data can be landed at a low cost but exceedingly adaptable storage/zone** to be examined afterward for potential insights. It is another advancement of what ETL/DWH pros called the Landing Zone of data. Only presently we are looking at ALL sorts of information independent of construction, structure, metadata, etc. One of the thoughts behind Data Lake is that presently innovation has made it conceivable to store ALL information that a firm generates/buys (prior it would be a case where the firm HAD to select the pertinent information and store it in a structured distribution center).

#### Data Warehouse

Data Warehouse is essentially a **social database facilitated on cloud or an endeavor centralized computer server**. It collects information from shifted, heterogeneous sources for the most reason for supporting the investigation and choice-making preparation of administration of any business.

A data warehouse is characterized as Subject-oriented, coordinates, time-variant, and non-unstable collection of information in arrange to supply business insights and help within the choice-making process.

### Difference between Data Lake and Data Warehouse



Data Lake	Data Warehouse
Data is kept in its raw form in Data Lake and here all the data are kept independent of the source of the information. They are as it was changed into other shapes at whatever point required.	Data Warehouse is composed of data that are extricated from value-based and other measurement frameworks. Here the information isn't in raw shape and is continuously changed and clean.
The most target for Data Lake is Data Researchers, Big Data Engineers, and <u>Machine Learning</u> Engineers who ought to do to profound investigation to form models for commerce such as predictive modeling.	The primary target of Data Warehouse is the operational clients as this information is in organized organize and can give prepared to construct reports. So they are generally utilized for trade intelligence.
The most inputs to data Lake are all sorts of information such as organized, semi-structured, and unstructured information. This information dwells in data Lake in their unique form.	The primary inputs to Data warehouse are organized information that is coming from value-based and measurements frameworks which are at that point organized within the shape of schemas.
Comprises of raw data that will or might not be curated.	It comprises of curated data which is centralized and is prepared to be used for commerce insights and analytics purpose.
Data is not in normalized form.	Denormalized schemas
The advances that are utilized in data lakes such as <u>Hadoop</u> , Machine Learning are moderately modern as compared to the information warehouse.	Here the technology that's utilized for a data warehouse is older.
A data lake can have all sorts of information and can be utilized with keeping past, show and prospects in mind.	Data Warehouse is concerned, here most of the time is went through on analyzing different sources of the data.
Data in interior of the data lake are profoundly open and can be rapidly updated.	Data in interior of the data warehouse are more complicated and it requires more fetched to bring any changes to them, availability is additionally confined as it were authorized users.

