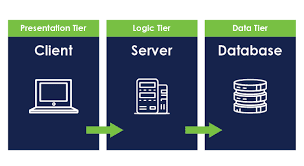
**Architecture and its types:**

A three-tier architecture is a software architecture pattern where an application is structured into three separate logical layers, each serving a specific purpose and having distinct responsibilities. These layers are typically referred to as presentation, application (or business logic), and data tiers. Here's a brief overview of each tier.



**Presentation Tier (User Interface):**

This tier is responsible for presenting information to the user and gathering user inputs.

It encompasses everything the user interacts with directly, such as web browsers, mobile apps, or desktop applications.

The main goal of this tier is to provide an interface that is intuitive, user-friendly, and responsive.

**Application Tier (Business Logic):**

Also known as the logic or middle tier, this layer contains the application logic that processes the requests, manipulates data, and performs the core functionalities of the application.

It acts as an intermediary between the presentation tier and the data tier.

Common tasks in this tier include authentication, authorization, data validation, business rules implementation, and workflow management.

**Data Tier (Data Storage):**

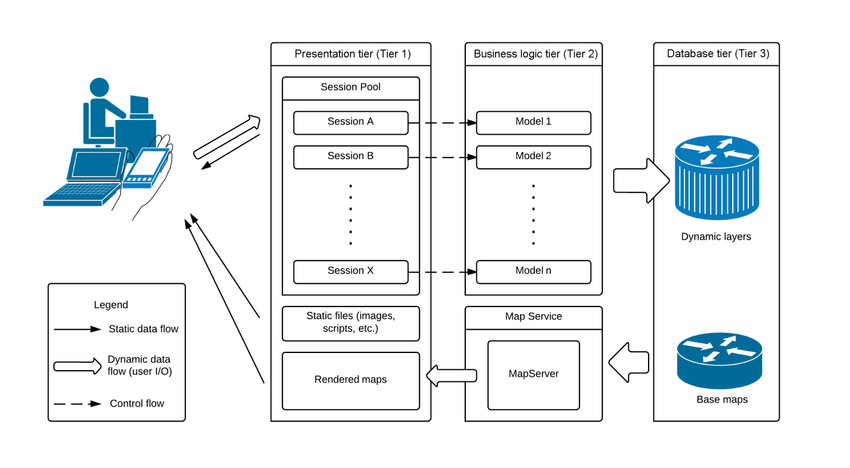
This tier is responsible for storing and managing data used by the application.

It typically includes databases or other data storage systems where information is persisted.

The data tier handles tasks such as data retrieval, storage, updating, and deletion.

Common database technologies used in this tier include SQL databases (e.g., MySQL, PostgreSQL) and NoSQL databases (e.g., MongoDB, Cassandra).

**Workflow:**



**Benefits of the three-tier architecture include:**

**Scalability:** Each tier can be scaled independently based on demand.

**Modularity:** The separation of concerns makes it easier to maintain, update, and modify different parts of the application without affecting others.

Security: By separating the layers, it's easier to implement security measures at each level.

**Reusability:** Components within each tier can be reused across different parts of the application or in other applications.

This architecture pattern is widely used in web applications, where the presentation tier often consists of web browsers or mobile devices, the application tier is implemented using web servers or application servers, and the data tier involves databases or other data storage systems.