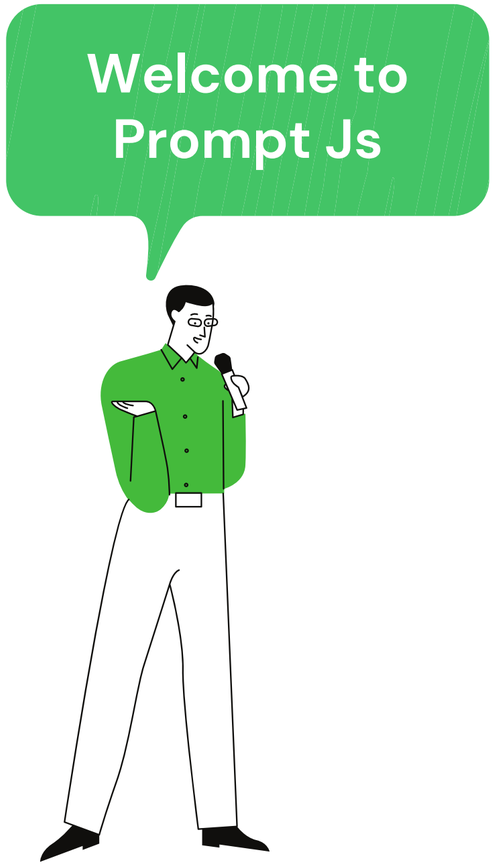
SOFTWARE

EVERYTHING ABOUT SOFTWARE



Its me santosh kunwar .
Backend Developer

skillPrompt

Today's Topics

1. TYPES OF SOFTWARES
2. SOFTWARE ARCHITECTURES
3. QNA SESSION

JR Developer

Software is like a parachute.

It doesn’t work if it’s not open.

System software

software > system software

System software is a program that manages and controls a computer's hardware and basic operations, enabling application software to run.

1. Operating software
2. Utility software
3. Device driver programs

JR Developer

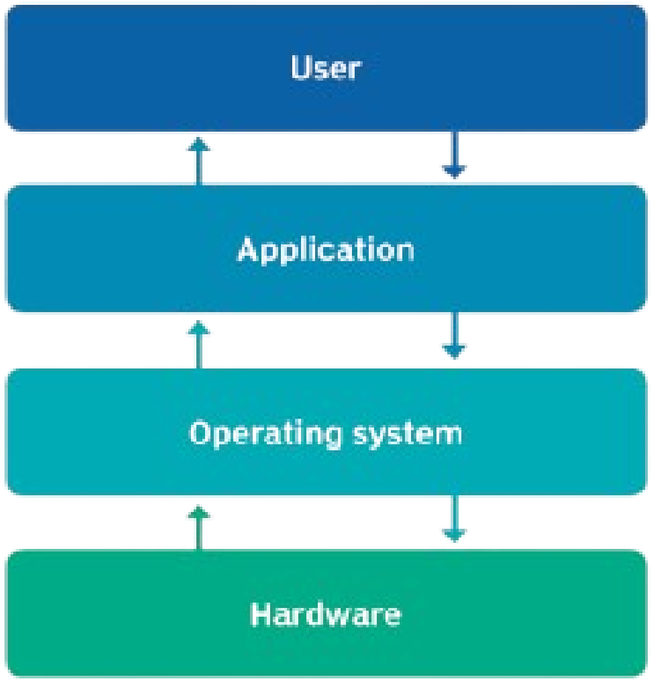
Why did the system software join a gym?

Because it wanted to improve its core strength and handle heavy loads more efficiently!

Operating software

software > system software> operating software

Operating software, or an operating system (OS), manages a computer's hardware and software resources, ensuring they work together efficiently. It allows you to run applications and perform tasks without needing to understand the computer's inner workings.



1. Stand alone operating software
   Linus , windows , Mac Os
2. Server operating software
   windows server , UNIX , Netware
3. Embedded operating software
   BlackBerry , Plam OS

Utility software

software > system software> utility software

Utility software helps maintain, manage, and protect your computer, making it run smoothly. It includes programs like antivirus, file management tools, and system cleaners.

1. **File management**
   copying, renaming, deleting, moving , sorting files and folders
2. **Disk cleanup**
   scans for and removes unwanted files
3. **Virus cleanup**
   removing any viruses contained in the memory, storage media, or incoming files

Utility software

software > system software> utility software

Utility software helps maintain, manage, and protect your computer, making it run smoothly. It includes programs like antivirus, file management tools, and system cleaners.

1. **File management**
   copying, renaming, deleting, moving , sorting files and folders
2. **Disk cleanup**
   scans for and removes unwanted files
3. **Virus cleanup**
   removing any viruses contained in the memory, storage media, or incoming files

programmers

Why did the utility software feel unappreciated?

Because everyone only notices it when something goes wrong!

Device Driver

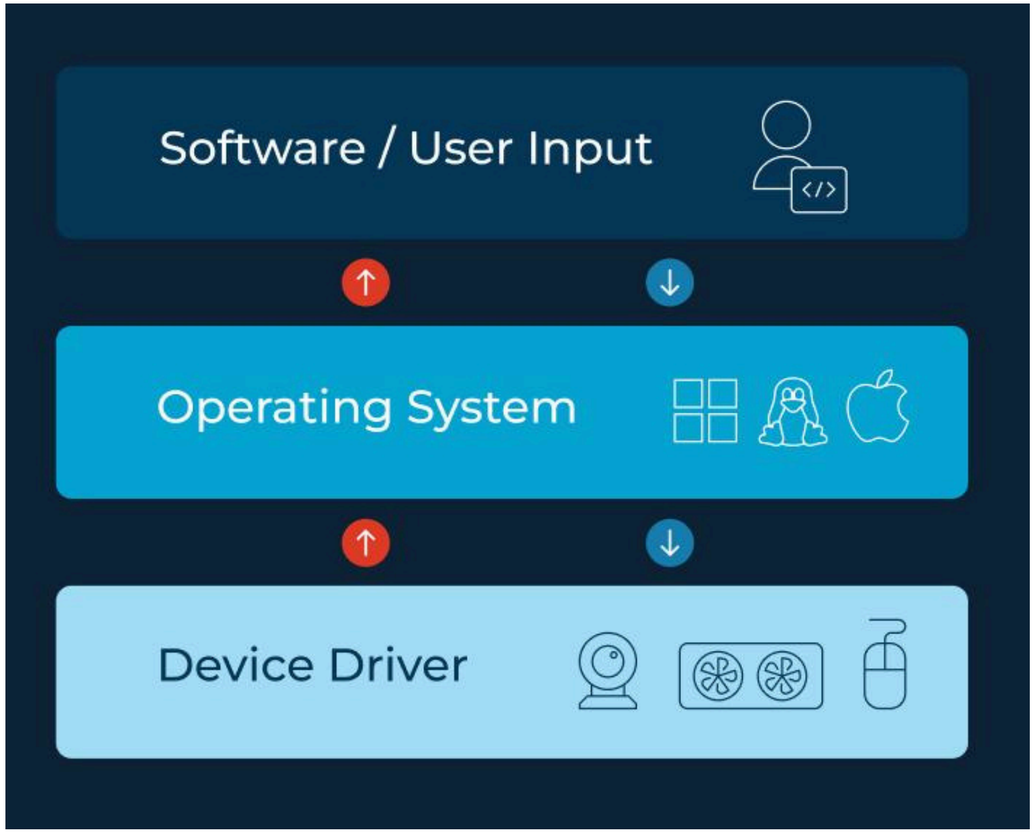
software > system software> device driver

A device driver is a small program that helps your computer communicate with hardware like printers, graphics cards, or keyboards. It ensures the hardware works correctly with your operating system.

1. **Graphics Driver**
   NVIDIA GeForce Driver, AMD Radeon Driver, Intel Graphics Driver

2. **Sound Driver**
   Realtek HD Audio Driver, Creative Sound Blaster Driver, ASUS Xonar Driver

3. **Printer Driver**
   HP Printer Driver, Canon Printer Driver, Epson Printer Driver



JR Developer

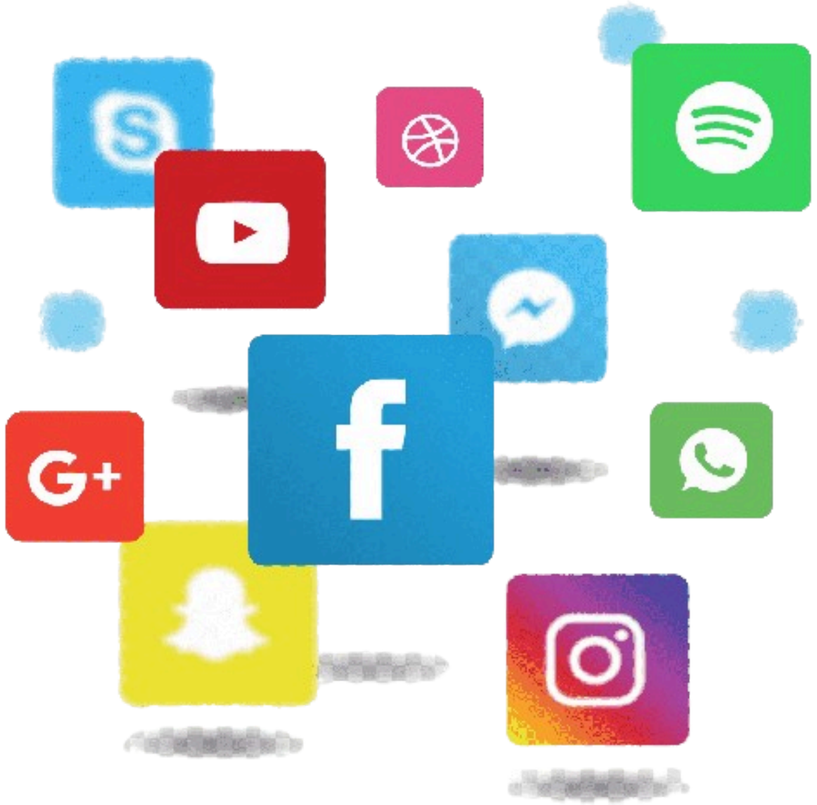
Why did the device driver break up with its keyboard?

Because they just couldn't find the right key to their relationship!

Application software

software > application software

Application software consists of programs designed for users to perform specific tasks, such as writing documents, browsing the internet, or editing photos. It's the software you use to complete everyday activities on your computer.

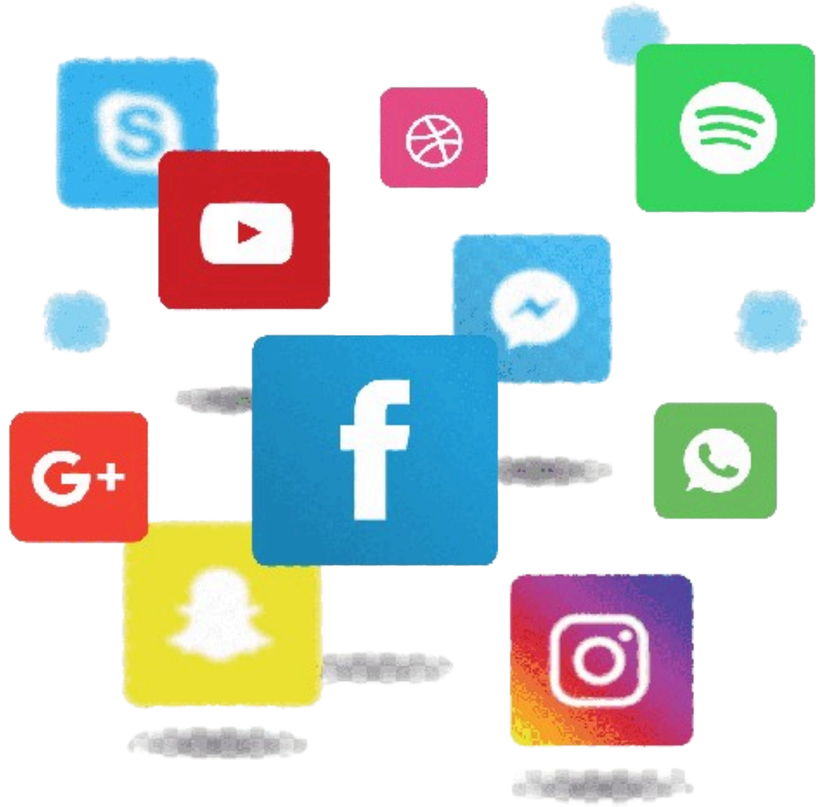


1. By funcitonality
   general purpose , special purpose
2. By Sharability and liscence
   Freeware , shareware , open source , closed source
3. By platform
   Desktop , mobile ,browser , cross platform
4. By Deployment
   On-Premises Software , Cloud-Based Software , Hybrid Software

Application software

software > application software

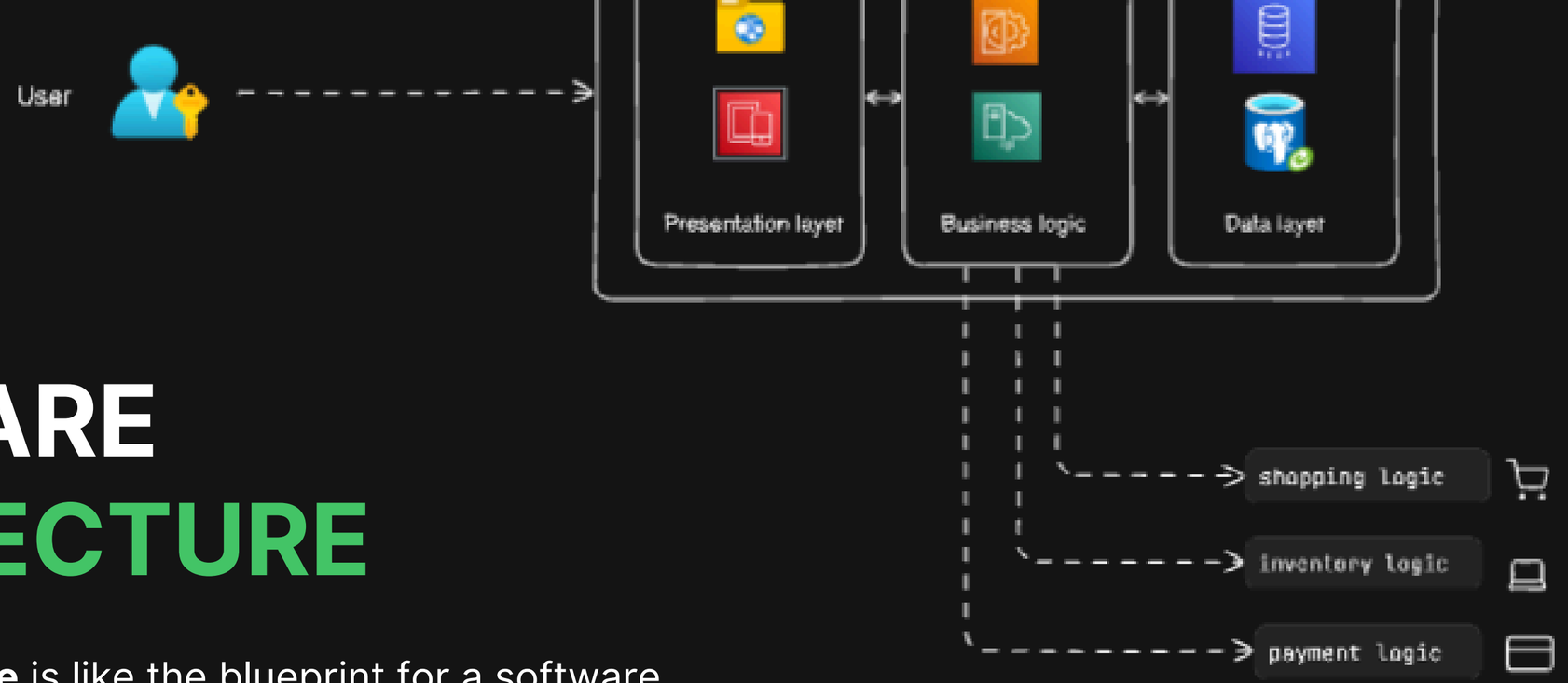
Application software consists of programs designed for users to perform specific tasks, such as writing documents, browsing the internet, or editing photos. It's the software you use to complete everyday activities on your computer.



5 **By User type**
Consumer software, Enterprise Software, Small Business Software

6 **By Cost**
Free Software, Subscription-Based Software , One-Time Purchase Software

7 **By Industry**
Healthcare Software, Financial Software, Educational Software, Retail Software



SOFTWARE ARCHITECTURE

Software architecture is like the blueprint for a software system. It defines the structure of the system, including how different parts of the software interact with each other and how they work together to achieve the overall functionality.

MICROSERVICES ARCHITECTURE

Ecommerce system architecture

JR Developer

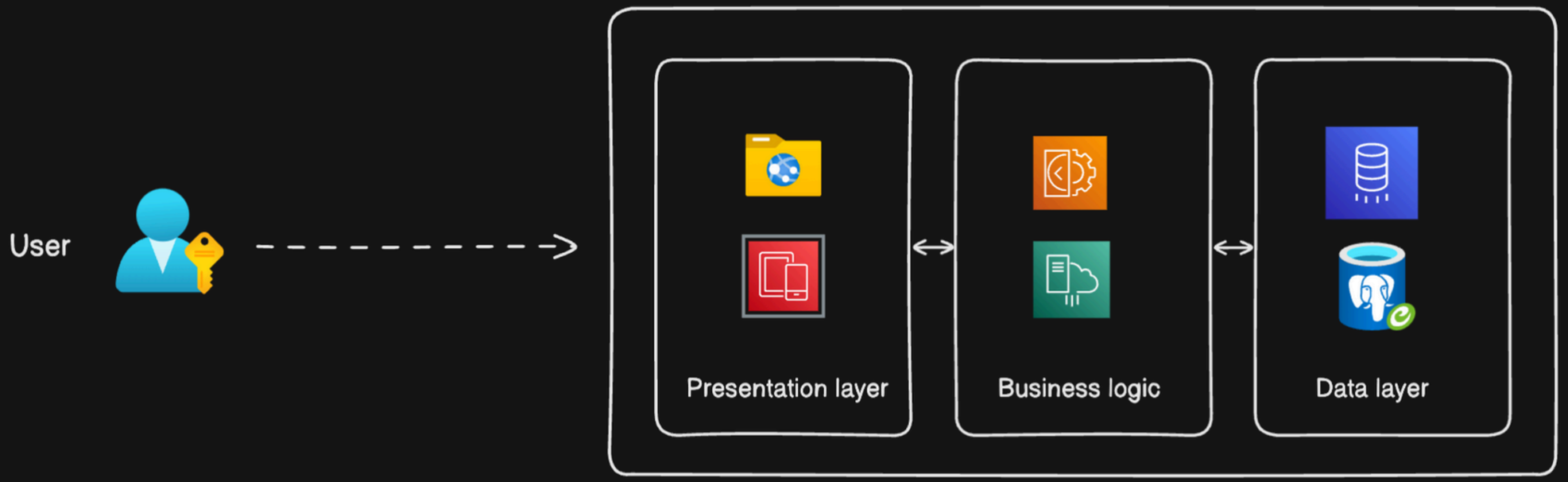
Why did the software architect always carry a ladder?

To reach the highest levels of abstraction, of course!

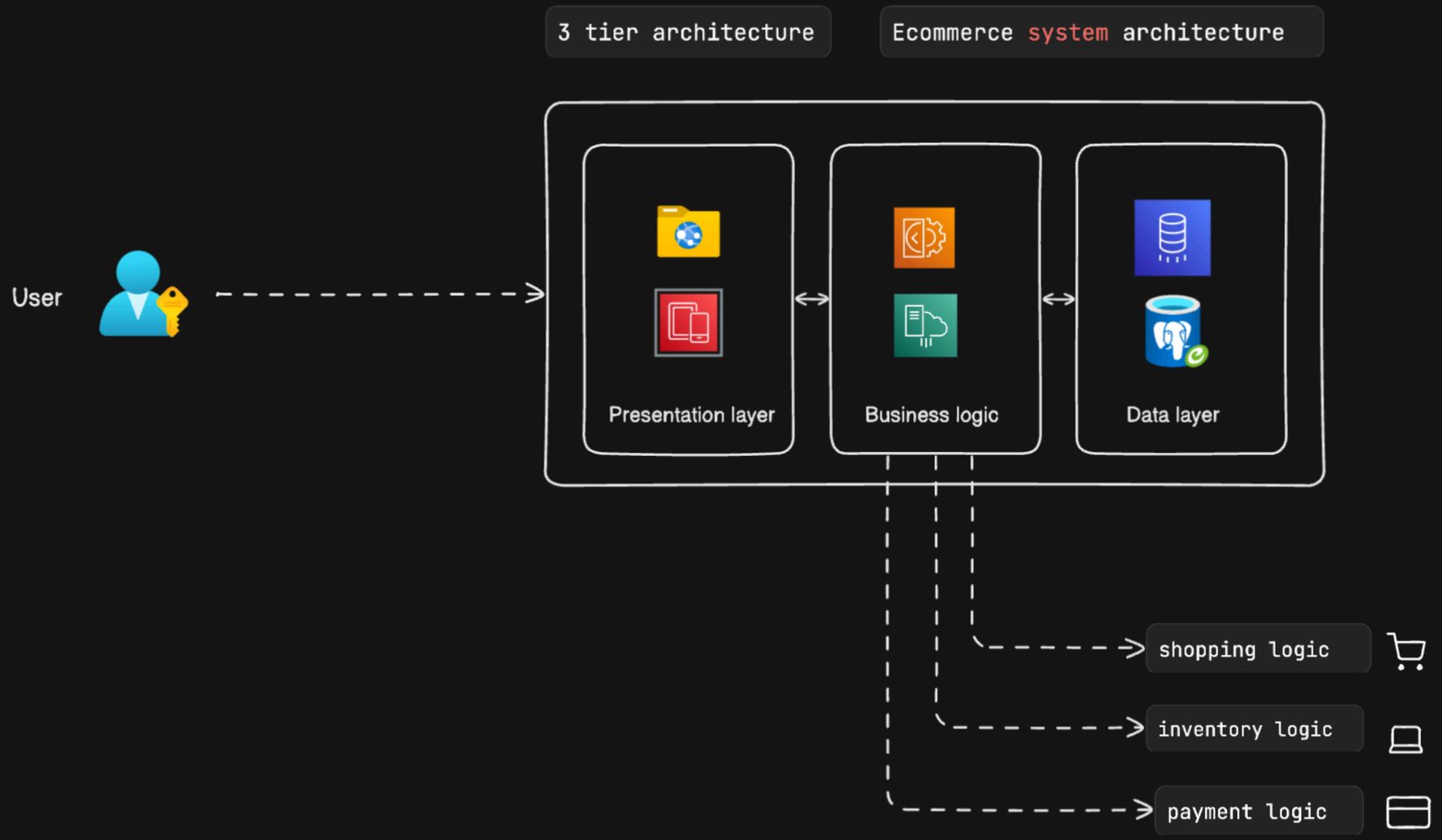
MONOLITHIC ARCHITECTURE

3 tier architecture

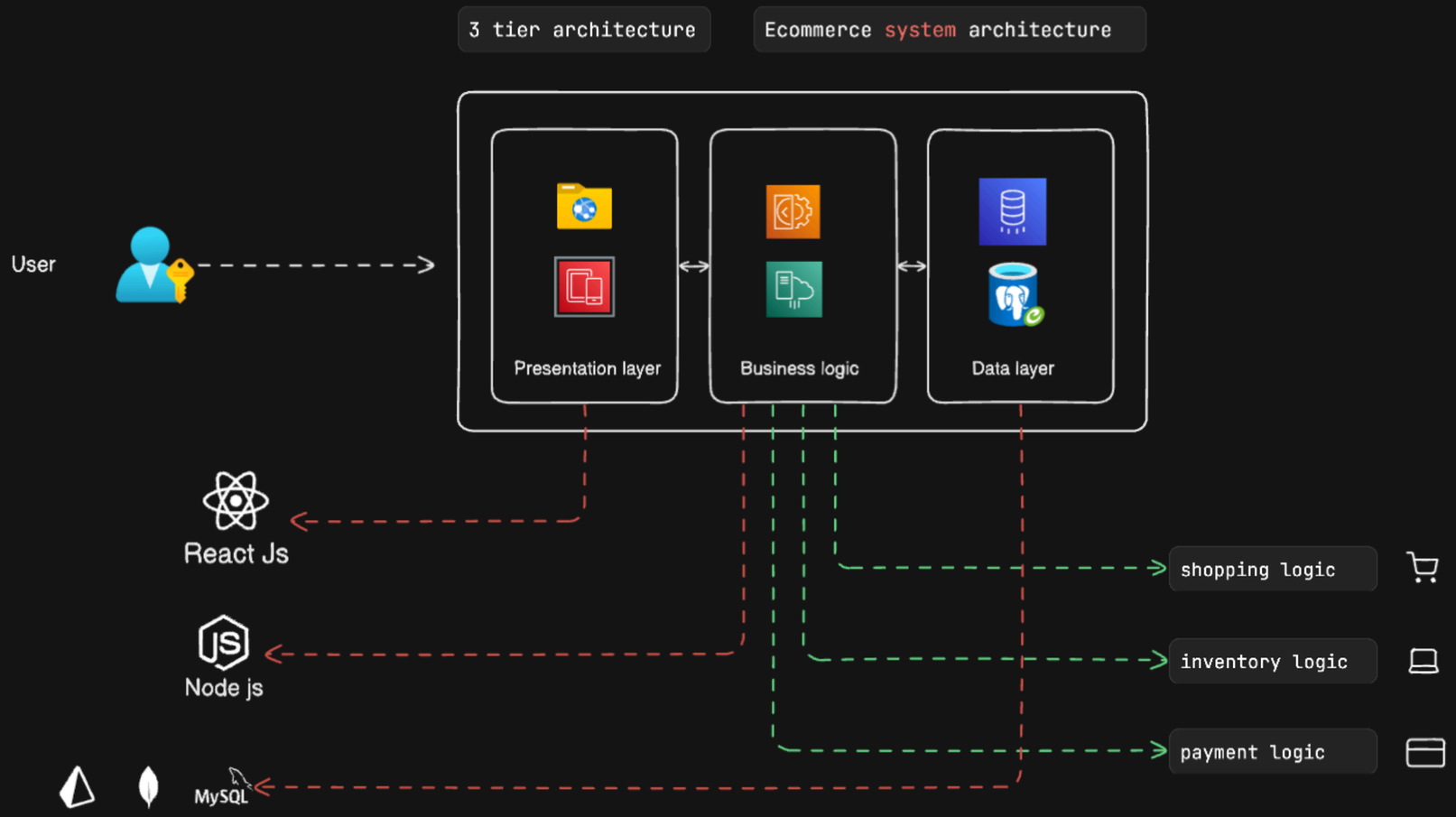
Ecommerce system architecture



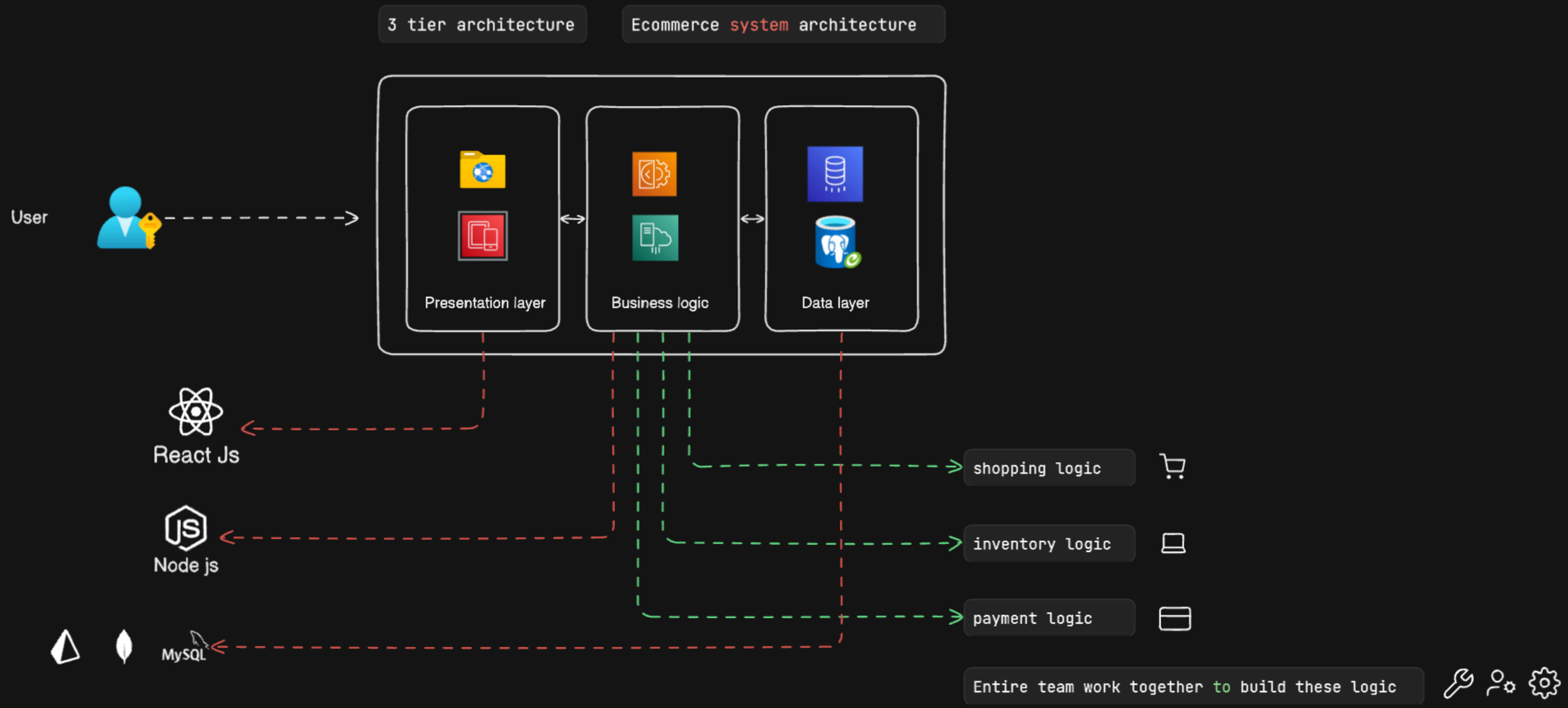
MONOLITHIC ARCHITECTURE



MONOLITHIC ARCHITECTURE

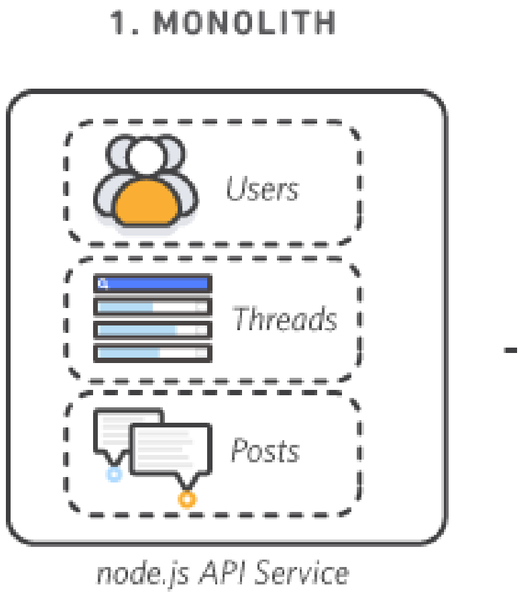


MONOLITHIC ARCHITECTURE



MONOLITH ARCHITECTURE

Monolith architecture is a software design where the entire application is built as a single, unified unit, making it easier to develop initially but harder to scale and maintain over time.



-

Single Codebase

All components of the application are developed and maintained within a single codebase, making it easier to manage and deploy.

Tight Coupling

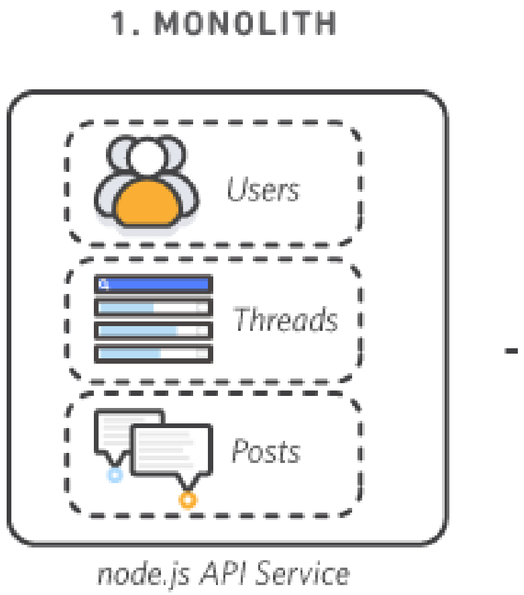
Components within the architecture are tightly integrated and interdependent, often sharing data and resources directly.

Shared Memory

Monolithic applications typically share the same memory space, allowing components to communicate efficiently without the need for network overhead

MONOLITH ARCHITECTURE

Monolith architecture is a software design where the entire application is built as a single, unified unit, making it easier to develop initially but harder to scale and maintain over time.



-

Centralized Database

Data storage is centralized within the application, typically using a single database instance for all data storage needs.

Layered Structure

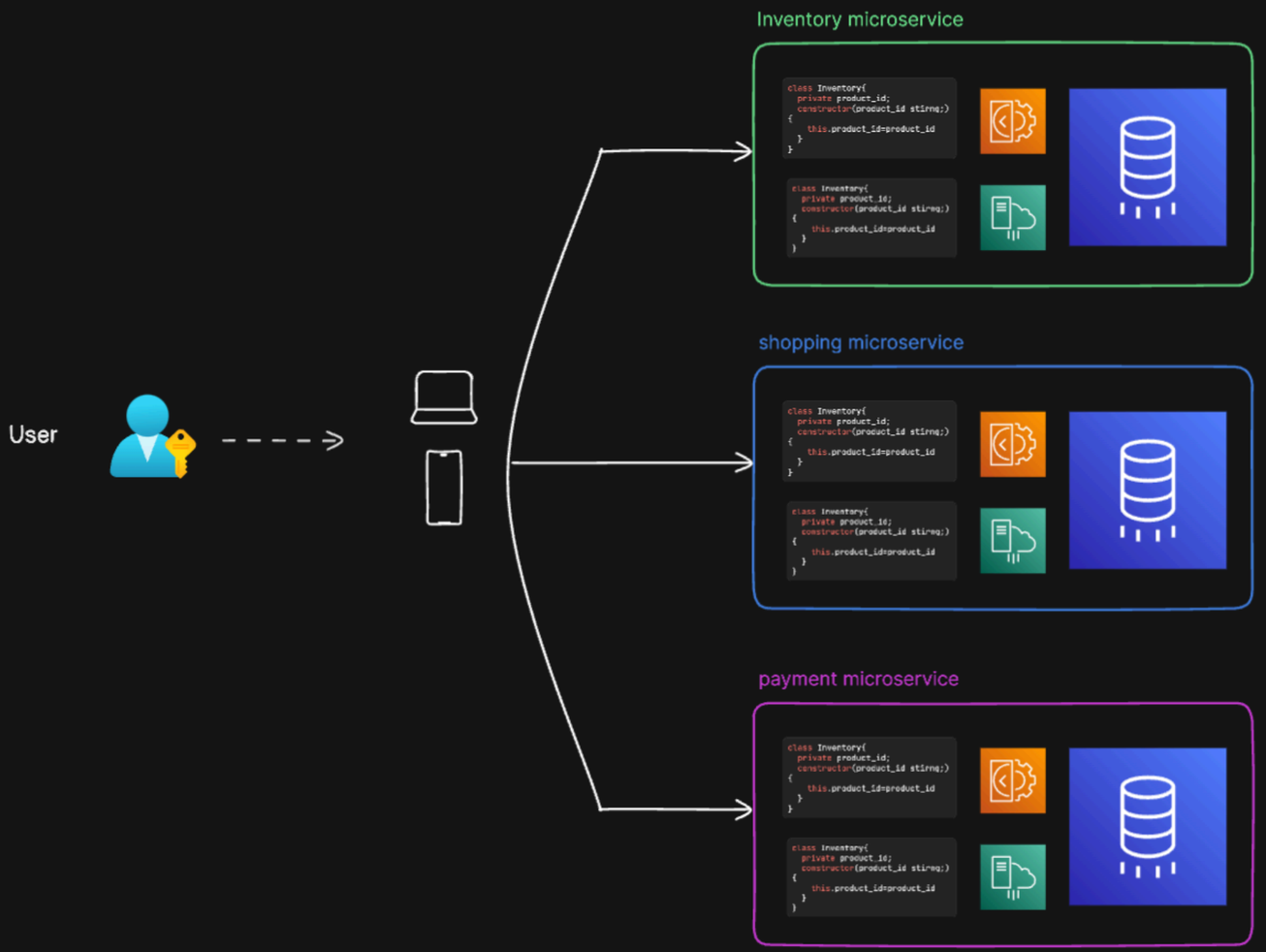
Monolithic architectures often follow a layered structure, with distinct layers for presentation, business logic, and data access.

Limited Scalablity

Scaling a monolithic application can be challenging, as the entire application must be scaled together, often resulting in inefficiencies and increased resource consumption.

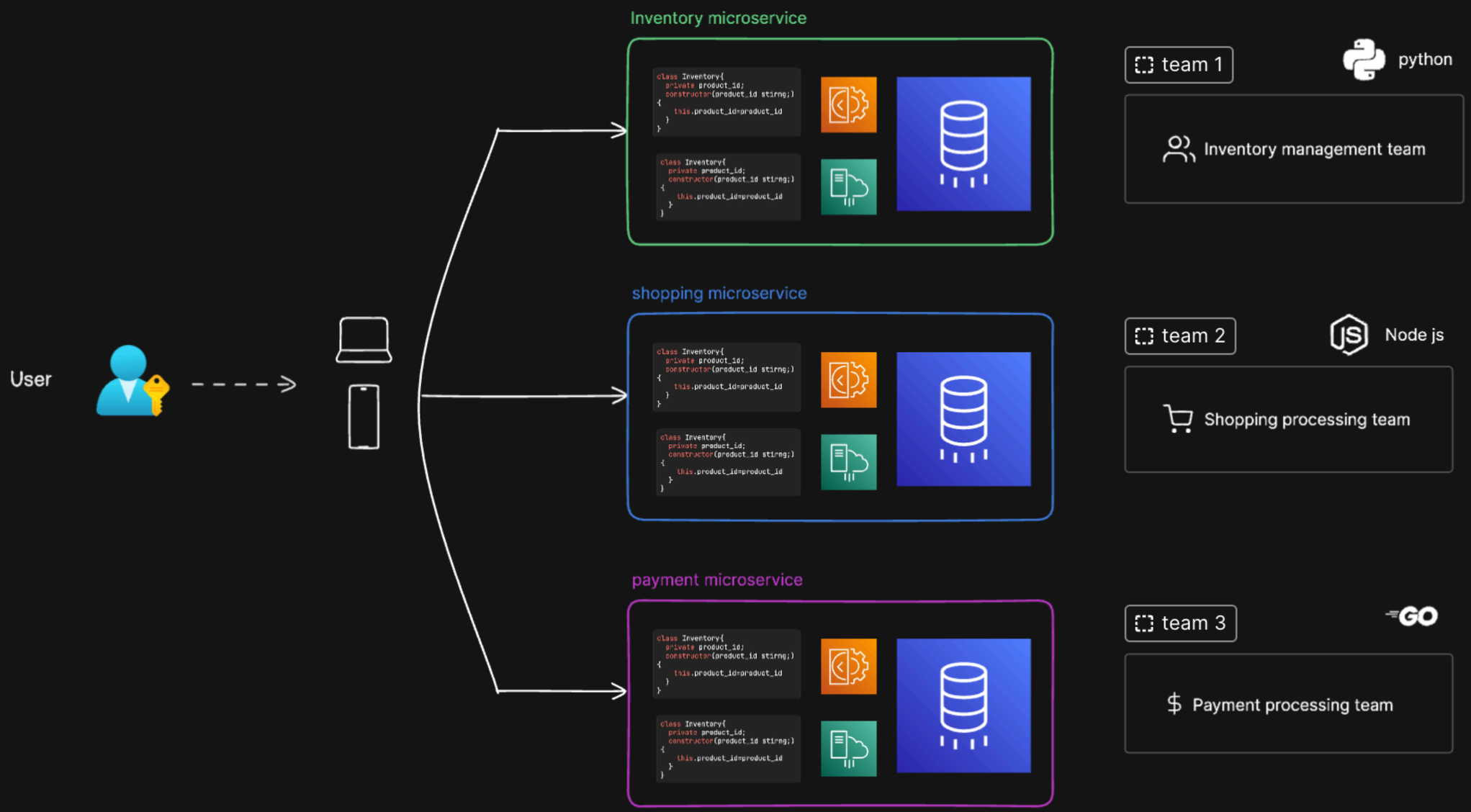
MICROSERVICES ARCHITECTURE

Ecommerce system architecture



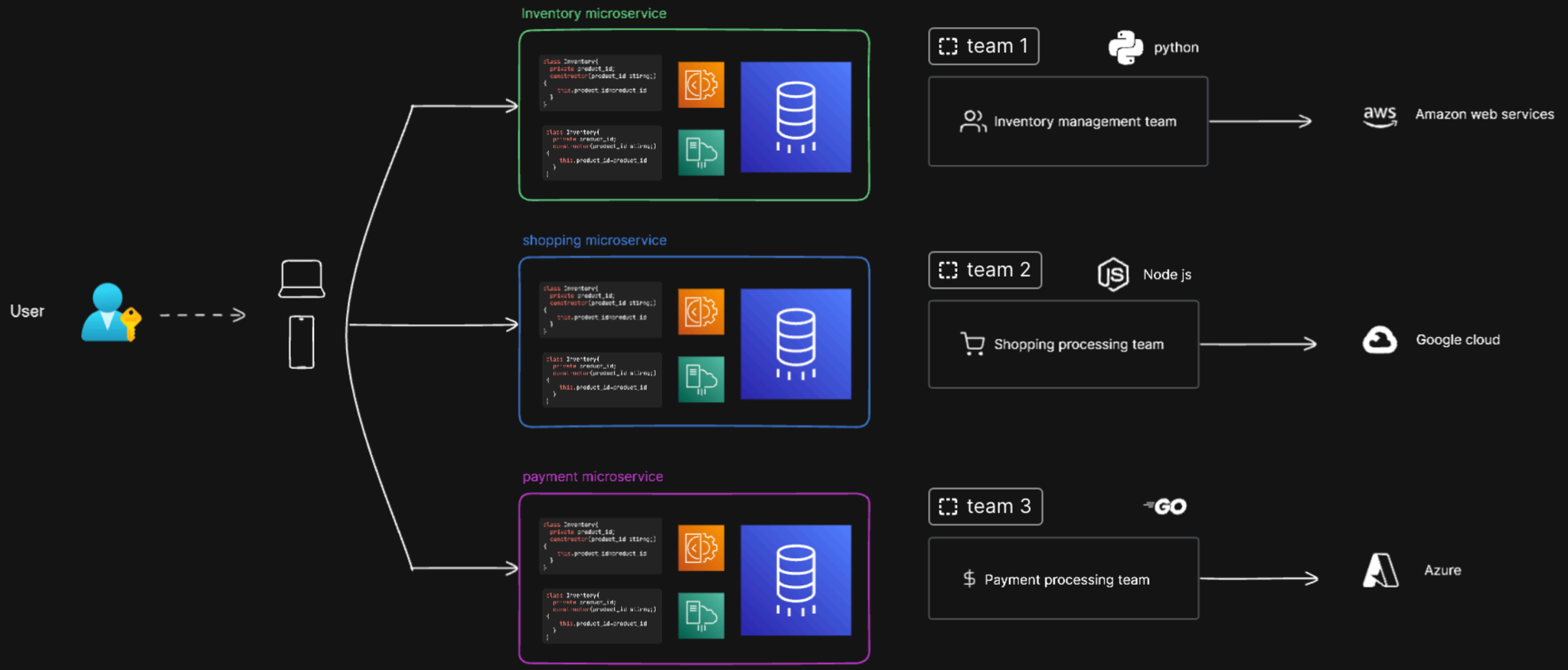
MICROSERVICES ARCHITECTURE

Ecommerce system architecture



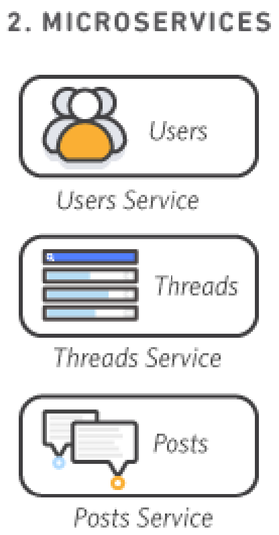
MICROSERVICES ARCHITECTURE

`Ecommerce system architecture`



MICROSERVICES ARCHITECTURE

Microservice architecture is a software design where an application is built as a collection of small, independent services that work together, making it easier to develop, scale, and maintain each part separately.



Multiple Codebases

Each microservice typically has its own independent codebase, allowing for separate development, deployment, and scaling

Loose Coupling

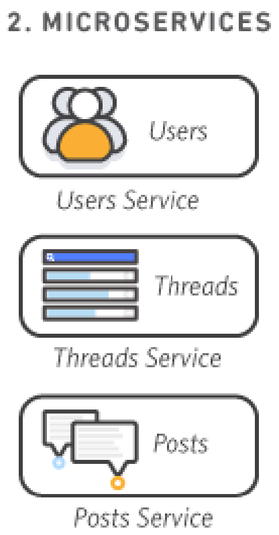
Microservices are designed to be loosely coupled, meaning they are more independent and can be developed and deployed independently without affecting other services.

Isolated Data Storage

Microservices often have their own databases or data stores, reducing the reliance on shared memory and avoiding issues of contention.

MICROSERVICES ARCHITECTURE

Microservice architecture is a software design where an application is built as a collection of small, independent services that work together, making it easier to develop, scale, and maintain each part separately.



Decentralized Database Management

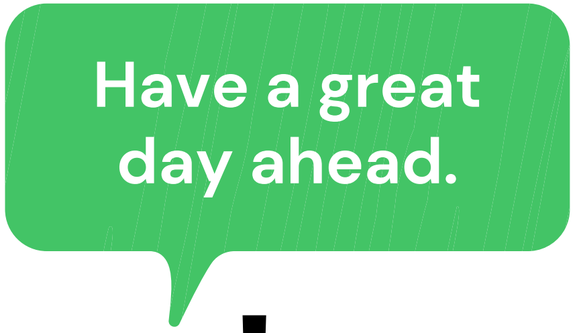
Rather than a centralized database, each microservice manages its own data storage, enabling better scalability and performance optimization for individual services.

Scalability

Microservices enable scaling individual services based on demand, contrasting with monolithic apps that scale by replicating the entire system.

Deployment Flexibility

Microservices allow independent updates, enabling faster releases and reducing risks of deployment failures across the whole application.



Thank you!

skillPrompt