Quick recap

AlgoCamp conducted a session on backend development focusing on client-server architecture and its fundamental concepts, using examples from various applications to illustrate how clients and servers interact through requests and responses. The discussion covered technical aspects of computer networking, including IP addressing, DNS, and port allocation, while explaining how machines communicate with each other. The session concluded with explanations of scaling approaches and load balancing concepts, with plans to cover cloud technologies and demos in future sessions.

Next steps

- [Students to explore the concept of DHCP protocol for understanding IP address allocation.](#)

- [Students with AWS accounts to spin up an EC2 instance, observe its IP address, shut it down, restart, and observe any changes to the IP address.](#)

- [Instructor to upload a detailed lecture on IP addressing concepts and computer network stack.](#)

- [Instructor to show a demo in the next class for students without AWS accounts on EC2 instance IP address changes.](#)

- [Instructor to continue the discussion on solving the issue of load balancer being a single point of failure in the next class.](#)

- [Instructor to address how data will be managed across multiple servers in the next class.](#)

- [Instructor to discuss cloud technologies in the next class.](#)

- [Instructor to introduce the concept of protocols in the next class.](#)

- [Students to keep an eye on Discord for the next week's schedule.](#)

- [Students to drop any additional questions in the tiered out section on Discord.](#)

Summary

Backend Development: Client-Server Architecture

AlgoCamp led a session on backend development, focusing on client-server architecture and using the case study of the now-defunct social bookmarking site Delicious. The discussion explained how users access web applications from different locations and how software can run on separate machines from the user's device, requiring a mechanism to transfer data between them. AlgoCamp emphasized that this understanding is crucial for developing web applications, particularly social media platforms, and outlined plans to explore how to build such applications in future sessions.

Client-Server Architecture Fundamentals

AlgoCamp explained the fundamental concepts of client-server architecture, defining a client as any process running on a machine capable of making requests, and a server as a process capable of accepting and processing requests to send responses. He provided examples of clients, such as the Zomato app and the Facebook app, illustrating how client processes raise requests to perform actions like searching for restaurants or liking posts. AlgoCamp emphasized that while the client is often associated with the frontend, it can also refer to any process making requests, distinguishing it from the broader concept of the frontend.

Software Architecture Fundamentals Explained

AlgoCamp explained the concepts of client, server, and producer-consumer in the context of software architecture. He used examples like HDFC Bank's loan deduction process and Google Pay's interaction with NPCI to illustrate how a client raises requests to a server, which processes and responds. He also described producer-consumer relationships, where one process creates data (producer) and another consumes it, using Zoom's recording processing as an example. AlgoCamp clarified that these concepts are fundamental to understanding how different software systems interact and process requests.

Client-Server Concepts and React

AlgoCamp explained the fundamental concepts of clients and servers, emphasizing that a server is a process running on a machine that accepts requests, processes them, and sends responses, while a client is any process capable of making requests. They discussed how a React server processes and transpiles JSX to HTML, CSS, and JavaScript before sending it to the client's browser. AlgoCamp also clarified that a client and server can run on the same machine, and explained that communication between different machines is already solved through existing solutions like HTTP. The session concluded with plans to write server-side logic for a social media application and expose a client-side interface for users.

Computer Connection Types Explained

AlgoCamp explained the two types of computer system interactions: wired and wireless connections, discussing their pros and cons. They highlighted that while wired connections offer faster speeds and lower latency, wireless connections are more practical for distant client-server interactions due to cost and scalability. AlgoCamp also explained the concept of computer networks, defining the Internet as a network of computer networks, and emphasized the need for a wireless connection in their project to facilitate interaction between client and server machines.

Client-Server Communication Basics

AlgoCamp explained the technical aspects of client-server interactions, focusing on how machines identify and communicate with each other using IP addresses and port numbers. They discussed how domain names simplify the process by mapping to IP addresses, similar to how phone directories map contact names to numbers. The session also covered how port numbers are allocated by developers to uniquely identify processes running on a machine, and addressed questions about MAC addresses and DNS servers.

Network Fundamentals and Port Allocation

AlgoCamp explained the basics of IP addressing, DNS, and port allocation. They clarified that each process can use only one port and explained the difference between dynamic and static IP addresses. AlgoCamp also discussed how DNS works and how clients and servers communicate. They emphasized the importance of understanding these concepts for developing real-world projects. AlgoCamp mentioned they would upload

additional lecture materials for further explanation and promised to cover HTTP in the next session.

Scaling Strategies for Social Media

AlgoCamp explained the concepts of vertical and horizontal scaling in the context of running a social media application. He described how vertical scaling involves upgrading a single machine's power to handle more load, but this approach has limitations and is costly. Horizontal scaling, on the other hand, involves deploying multiple machines of similar capability to distribute the load and handle more users. AlgoCamp emphasized that while vertical scaling improves the power of existing machines, horizontal scaling is more effective for handling large-scale user loads.

Load Balancing Concepts and Challenges

AlgoCamp explained the concept of load balancing, discussing how a load balancer works as an intermediary between clients and servers to distribute requests efficiently. They highlighted that while load balancers can become a single point of failure, this issue and others, such as managing static IPs and data backup, will be addressed in the next class. AlgoCamp also mentioned that they would cover cloud technologies, demos, and different types of load balancers in future sessions.