

System Design Interview PDF



You are welcome to download and read the PDF for your personal use, but please respect our intellectual property rights and do not share or publish the content elsewhere without our permission. Thank you for your understanding.

Overview of system design

System design refers to the process of designing the architecture, components, modules, interfaces, and data for a software system to satisfy specified requirements. It involves defining the system's architecture, decomposing the system into smaller components, specifying the interfaces between these components, and identifying the data structures and algorithms required to implement the system's functionality.

Overview of system design

As you prepare for your system design interview, don't let assignments and homework hold you back. With the help of a reliable homework help service, you can focus on acing your interview and landing your dream job. Get the support you need to excel in both your coursework and career.

1. *Design a Chat Application*

A chat application allows users to send and receive messages in real-time. Some key considerations for designing such an application include user authentication and authorization, message delivery and storage, and scalability. An example architecture might include a message broker like Apache Kafka for handling message delivery and a NoSQL database like Cassandra for storing messages.

2. Design a social network

A social network is a platform that connects users with friends, family, and colleagues. Key considerations for designing such a platform include user profiles, friend connections, and news feeds. An example architecture might include a graph database like Neo4j for representing user relationships and a distributed cache like Redis for storing and serving user profiles.

3. Design a file-sharing system

A file-sharing system allows users to upload and download files, as well as share them with others. Key considerations for designing such a system include user authentication and authorization, file storage and retrieval, and access controls. An example architecture might include a cloud storage service like Amazon S3 for storing files and an access control service like OAuth for managing user permissions.

4. Design a ride-sharing service

A ride-sharing service allows users to request and offer rides to other users. Key considerations for designing such a service include user authentication and authorization, ride matching and routing, and payment processing. An example architecture might include a geospatial database like MongoDB for handling ride requests and a payment processing service like Stripe for handling payments.

5. Design a search engine

A search engine allows users to search for information on the web. Key considerations for designing such an engine include indexing and ranking algorithms, query processing, and search result presentation. An example architecture might include a search engine like Elasticsearch for indexing and querying web pages and a front-end framework like React for presenting search results.

6. Design a recommendation system

A recommendation system suggests products or content to users based on their preferences and behavior. Key considerations for designing such a system include user data collection and analysis, machine learning algorithms, and recommendation presentation. An example architecture might include a data lake like Amazon S3 for storing and processing user data and a machine learning service like Amazon SageMaker for training recommendation models.

6. Design a recommendation system

A recommendation system suggests products or content to users based on their preferences and behavior. Key considerations for designing such a system include user data collection and analysis, machine learning algorithms, and recommendation presentation. An example architecture might include a data lake like Amazon S3 for storing and processing user data and a machine learning service like Amazon SageMaker for training recommendation models.

7. Design a content delivery network (CDN)

A CDN delivers content to users from servers located geographically closer to them, improving website performance and reducing latency. Key considerations for designing such a network include server selection and routing algorithms, caching policies, and load balancing. An example architecture might include a CDN provider like Cloudflare for routing requests to the closest server and caching content.

8. Design a video streaming service

A video streaming service allows users to watch videos on-demand. Key considerations for designing such a service include video encoding and transcoding, content delivery, and video player performance. An example architecture might include a video platform like Vimeo for encoding and delivering video content and a video player like Video.js for playing videos in the browser.

9. Design a booking system

A booking system allows users to reserve seats or rooms for events, flights, or hotels. Key considerations for designing such a system include seat allocation and reservation, payment processing, and inventory management. An example architecture might include a booking platform like OpenTable for managing reservations and a payment gateway like PayPal for handling payments.

10. Design a recommendation system for a retail website

A recommendation system for a retail website suggests products to users based on their browsing and purchasing history. Key considerations for designing such a system include user data collection and analysis, product similarity algorithms, and recommendation presentation. An example architecture might include a product database like MySQL for storing product data and a recommendation engine like Amazon Personalize for generating personalized recommendations.