Step 3 - Design a deep dive

Goal: dig into details of some crutial system components

When do we move to this step?

- Our the high-level design should get agreement from the interviewer(overall goals and feature scope)
- We've already known what to focus on in the deep dive(specific components)

In the context of a system design interview, this step focuses on exploring your architectural decisions and justifying them based on a theoretical design scenario. The aim here is not to create a full-blown implementation plan as you would in an actual project but to demonstrate your ability to think critically about system architecture, scalability, security, and other important aspects of system design.

Here's how you should approach this stage in an interview setting and the types of questions you might be expected to answer:

What to Do in Step 3 - Design a Deep Dive for an Interview:

- 1. **Discuss Component Design**: Elaborate on the key components of your architecture, explaining why each component exists and how they interact.
- Focus on Scalability: Describe how the system can scale, discussing both horizontal and vertical scaling strategies appropriate for the hypothetical workload.
- 3. **Detail Data Management**: Talk about the database schema, choice of database system (SQL vs. NoSQL), and how data consistency, integrity, and partitioning would be handled.
- 4. **Explain API Strategy**: Outline the key API endpoints that would be needed, their inputs and outputs, and any RESTful design principles you would adhere to.

Step 3 - Design a deep dive

- 5. **Address Security Measures**: Discuss how you would secure the system, including authentication, authorization, data encryption, and other security protocols.
- 6. **Talk About Fault Tolerance and Reliability**: Explain how the system design accommodates failures, using concepts like redundancy, replication, and health checks.
- 7. **Consider Performance Optimization**: Mention how you would use caching, load balancing, and database indexing to enhance performance.

Sample Questions to Answer During Step 3:

1. Component Integration:

 How do the various components of your system communicate? What choices did you make regarding API design, and why?

2. Data Management:

- What type of database would you choose for this application, and what factors influenced this decision?
- How would you handle transactional integrity and data consistency in your design?

3. Scalability:

- How would you ensure that the system can handle a significant increase in user traffic or data volume?
- What are some potential bottlenecks in your design, and how might you address them?

4. Security:

- What security features are essential for this application, and how would you implement them?
- How would you secure sensitive user data, especially payment information?

5. Performance:

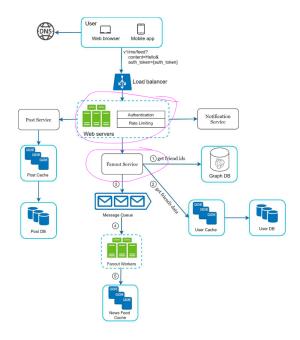
What strategies would you use to reduce latency in the system?

Step 3 - Design a deep dive

• How would caching be implemented to improve response times?

6. Reliability and Fault Tolerance:

- What strategies would you implement to ensure high availability and reliability?
- How would the system handle component failures?



Step 3 - Design a deep dive