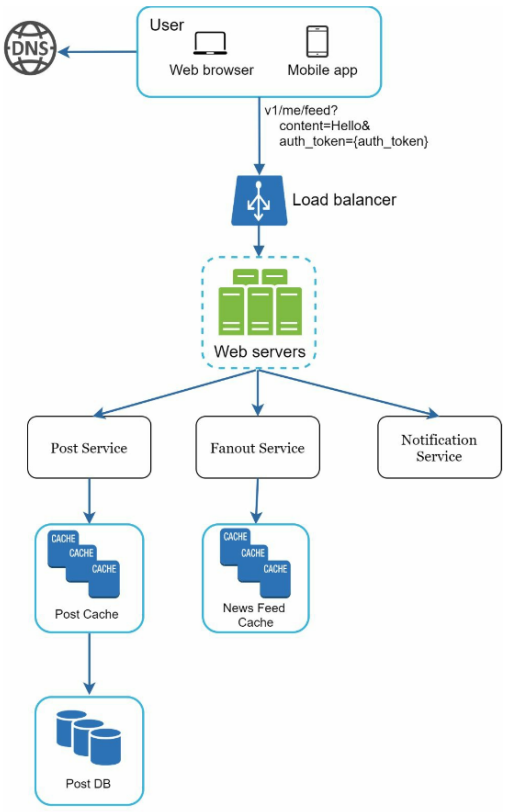
Step 2 - Propose high-level design and get buy-in

> Goal: reach an agreement with the interviewer on the design

- **Work together** with the interviewer on the design and **get feedback**
- Draw box diagrams with key components
- Do back-of-the-envelop estimation if needed
- Go through the design with use cases → discover edge cases

Example for feed publishing:



In Step 2 of the system design process for an online bookstore—proposing a high-level design and getting buy-in—it's important to lay out a blueprint that addresses all the

core requirements identified in the first step. Here's what you might ask to ensure alignment and gather necessary inputs before finalizing this high-level design:

Questions to Ask(complement to step 1) for the online bookstore problem:

1. **Purpose and Core Features:**
    - What specific features are most critical for the online bookstore (e.g., search functionality, recommendations)?
2. **User Base:**
    - Who is our target audience (e.g., general public, specific interest groups)?
    - Do we expect user traffic to vary seasonally or with specific promotions?
3. **Usage Patterns:**
    - What are the peak traffic expectations, and how often do they occur?
    - What is the expected frequency and volume of transactions per user?
4. **Data:**
    - What kinds of data will we be handling (e.g., user data, payment information, book details)?
    - How much data storage will be required initially, and what is the expected growth over time?
5. **Performance Requirements:**
    - What are the expected response times for user queries and transactions?
    - Are there specific performance metrics we need to meet for search results and checkout processes?
6. **Scalability:**
    - How will the system scale to handle thousands of concurrent users or more?
    - What strategies will we use for scaling, such as cloud services, load balancers, or database sharding?
7. **Integration:**
    - What external services will we need to integrate, such as payment gateways or social media platforms for login?

- Are there existing inventory systems or databases that need to be integrated?

8. **Availability and Reliability:**
   - What uptime is required for the bookstore to be considered reliable?
   - What redundancy plans and backup systems will be in place to ensure data integrity?

   ==========

   (Advanced, not limited to interviews)

9. **Security Requirements:**
   - What levels of security are needed for user data and financial transactions?
   - How will we comply with regulations like GDPR or PCI DSS?
10. **Budget and Resources:**
    - What is the budget for developing and maintaining the online bookstore?
    - What resources do we have available, and what might we need to acquire (e.g., technology, personnel)?
11. **Maintenance and Monitoring:**
    - What tools will we use to monitor the health of the system and user activity?
    - How will we handle updates and maintenance without affecting availability?
12. **Timeline:**
    - By when does the system need to be operational?
    - Are there key milestones, such as beta launches or marketing campaigns, that affect the timeline?

Online Bookstore High-Level Design Proposal:

Here's a simple high-level design for an online bookstore:

1. User Interface:

- **Web Frontend**: Handles user interactions such as searching, browsing, and purchasing books.
- **Mobile App**: Provides access on mobile devices with similar functionalities as the web version.

2. Backend Services:

- **User Service**: Manages user profiles, authentication, and authorization.
- **Book Management Service**: Handles book inventory, details, and metadata.
- **Shopping Cart Service**: Manages the shopping cart and session data.
- **Order Processing Service**: Handles order creation, payment processing, and order fulfillment.
- **Review and Rating Service**: Manages user reviews and book ratings.

3. Database Layer:

- **User Database**: Stores user data, preferences, and security information.
- **Book Database**: Stores book details, stock levels, and pricing.
- **Transaction Database**: Records transactions, orders, and payment details.

4. Supporting Infrastructure:

- **Search Engine**: Powers book searches with filters and sorting capabilities.
- **Payment Gateway Integration**: Manages transactions through external payment services.
- **CDN and Caching Solutions**: Ensures fast content delivery and reduces load on core servers.

5. Deployment and Operations:

- **Cloud Infrastructure**: Leverages cloud services for hosting, scaling, and backups.
- **Monitoring and Logging**: Tools to monitor system health and log activities for troubleshooting.