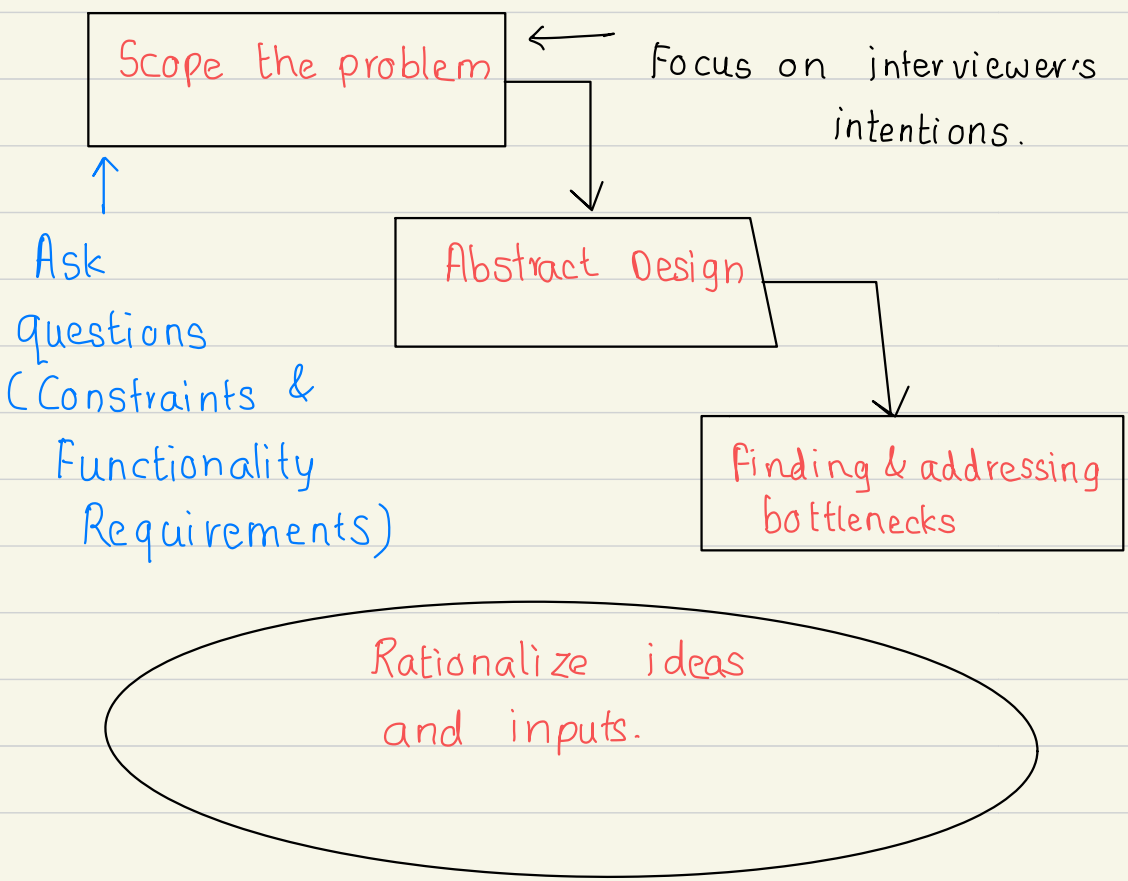
System Design Basics

1) Try to break the problem into simpler modules (Top down approach)
2) Talk about the trade-offs
(No solution is perfect)
Calculate the impact on system based on all the constraints and the end test cases.



System Design Basics (contd.)

1) Architectural pieces / resources available
2) How these resources work together
3) Utilization & Tradeoffs

- Consistent Hashing
- CAP Theorem ✓
- Load balancing ✓
- Queues
- Caching ✓
- Replication ✓
- SQL vs No-SQL ✓
- Indexes ✓
- Proxies
- Data Partitioning ~

Load Balancing

(Distributed System)

Types of distribution
- Random
- Round - robin
- Random (weights for memory & CPU cycles)

To utilize full scalability & redundancy, add 3 LB

1) User <-LB1-> Web Server
2) Web Server <-LB2-> App Server / Cache Server (Internal platform)
3) Internal platform <-LB3-> DB.