

# DETAILED ROADMAP

TO SHIFT FROM **SERVICE**-BASED  
**COMPANY** TO **TOP PRODUCT**  
**BASED COMPANY**





## **\*DISCLAIMER\***

This is an **action-oriented** document  
When you **tick all the boxes**, consider yourself  
prepared for the top techs

# MASTER **PROBLEM SOLVING** IN DSA

Start with basics, level up gradually, & cover all topics

## DATA STRUCTURE

- ☐ Arrays
- ☐ Strings
- ☐ Linked List
- ☐ Stacks
- ☐ Queues
- ☐ Binary Tree
- ☐ Binart Search Tree
- ☐ Hash Tables
- ☐ Graphs
- ☐ Heaps
- ☐ Matrix

## ALGORITHMS

- ☐ Sorting
- ☐ Greedy
- ☐ Dynamic
- ☐ Backtracking
- ☐ Divide and Conquer
- ☐ Randomized
- ☐ Hashing
- ☐ Graph Theory



**Weeks**

# BE MINDFUL OF

- ☐ Not mugging up the solutions
- ☐ Solving it yourself, or take a mentor's help
- ☐ Writing optimized solutions in terms of time and space complexity
- ☐ Checking for edge cases
- ☐ Solving the questions in a given time limit (once you are comfortable with the topic)





# BE CONFIDENT WITH **CS FUNDAMENTALS**

- ☐ Database
- ☐ Operating System
- ☐ Computer Network
- ☐ Object-Oriented  
Programming



## **BE MINDFUL OF**

---

- ☐ Making **short** and **crisp notes** to avoid confusion
- ☐ Take up **some quizzes** to assess yourself in these topics



# LEARN **SYSTEM DESIGN** **[LLD & HLD]**

The objective of **LLD** is that you as a software engineer write **good quality code** that is **extensible**, **reusable**, and **maintainable**.

## **LOW LEVEL DESIGN INCLUDES**

---

- |   |   |
|---|---|
| <input type="checkbox"/> OOP's                      | <input type="checkbox"/> Machine Coding         |
| <input type="checkbox"/> Design Patterns            | <input type="checkbox"/> Solid Principles       |
| <input type="checkbox"/> UML/Class Diagrams         | <input type="checkbox"/> Reduce Code Redundancy |
| <input type="checkbox"/> Schema Design              | <input type="checkbox"/> Efficient Memory use   |
| <input type="checkbox"/> Test Cases/Design Problems |   |



**HLD** provides a brief description of how the various **sub-systems** and **components** of the system fit together.

### **HIGH LEVEL DESIGN INCLUDES**

---

- ☐ Performance vs scalability
- ☐ Latency vs throughput
- ☐ CAP theorem
- ☐ Availability patterns
- ☐ Domain name system
- ☐ Content delivery network
- ☐ Microservices
- ☐ Architecture Design
- ☐ Distributed Systems
- ☐ Load Balancing
- ☐ Consistent Hashing
- ☐ Caching
- ☐ Transactions
- ☐ SQL



# DEVELOPMENT AND PROJECTS

- ☐ After having **hands-on** with with **required skills** start working on real-life industry-scale projects ( like the bare minimum of **Swiggy** )
- ☐ Have **2 to 3 projects** in your resume, showcasing your deep understanding of using the core **fundamental skills**
- ☐ Build **end-to-end products** from Frontend to Backend which can scale
- ☐ Try to add **impactful** and **scalable** projects creating Amazon and Uber app systems



# WORK ON **RESUME** AND **NETWORKING**

- ☐ Create an **effective resume**.  
[Click here](#) to get the complete guide
- ☐ **Be thorough** with everything mentioned in the **resume**
- ☐ Network with the **software engineers** and **recruiters** from the companies you aspire to work with



**Weeks**

# APPLY AND HAVE PATIENCE

- ☐ Apply to **20+ companies**. This increases the probability of getting shortlisted

To lessen your load, here's a compiled list of 90+ [Top product companies](#)

- ☐ Draft an **effective message** to seek referrals among your connections
- ☐ Apply to **job posts** matching **70%** of your profile
- ☐ Tailor your resume according to **job description**
- ☐ Keep a track of your **multiple job** applications



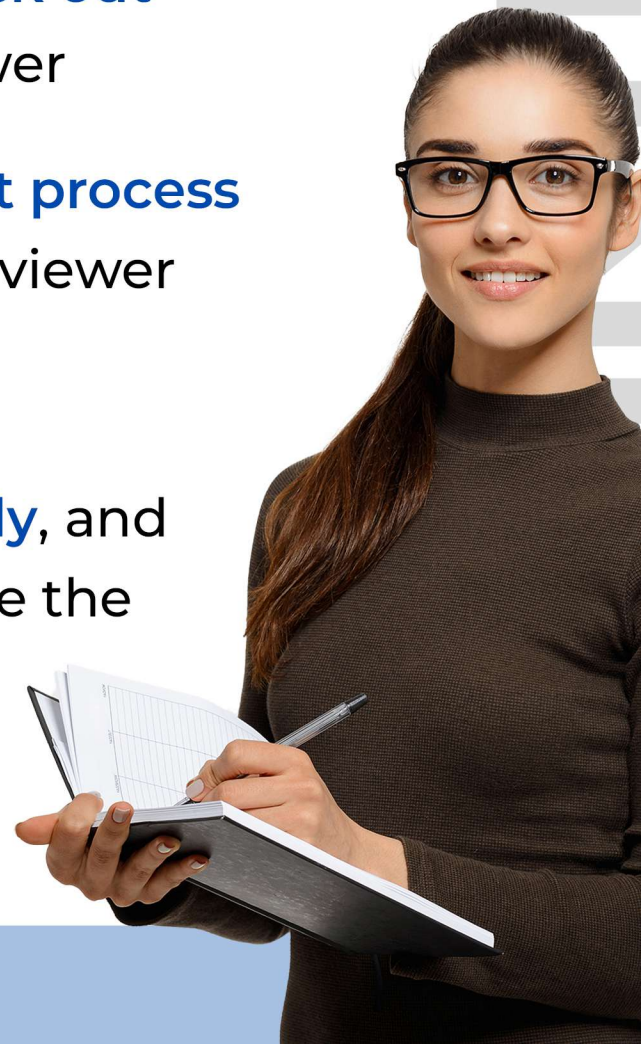


# KEEP PRACTICING MOCK INTERVIEWS

To **overcome** the notion of "I'm not ready yet."

## BE MINDFUL OF

- ☐ Don't make assumptions. **Ask out your doubts** to the interviewer
- ☐ Always discuss your **thought process** and approach with the interviewer
- ☐ Talk about the **edge cases**
- ☐ Take **feedback constructively**, and work to fill in the gaps before the actual interview



Don't get **OVERWHELMED!**  
We have got a **ONE-STOP** SOLUTION

- ✓ Well-structured curriculum for **DS+Algo**,  
**CS fundamentals**, & **System Design**  
(LLD & HLD)
- ✓ Industry **Relevant Projects**
- ✓ **1:1 Mentorship** & Mock Interviews  
with industry experts
- ✓ **Career Support:** Resume  
optimization, Community referrals

Within a span of **7 months**, you will  
**Develop Skills + Confidence + Hands-On Experience**  
to grab your dream job.




[www.bosscoderacademy.com](http://www.bosscoderacademy.com)







# WHY BOSSCODER

-  **200+** alumni placed at Top **product-based** companies
-  More than **120% hike** for every **2 out of 3** working professional
-  Avg package of **22 LPA**

The syllabus is most up-to-date and the list of problems provided covers all important topics.

Lavanya  
 Meta



Course is very well structured and streamlined to crack any MAANG company

Rahul  