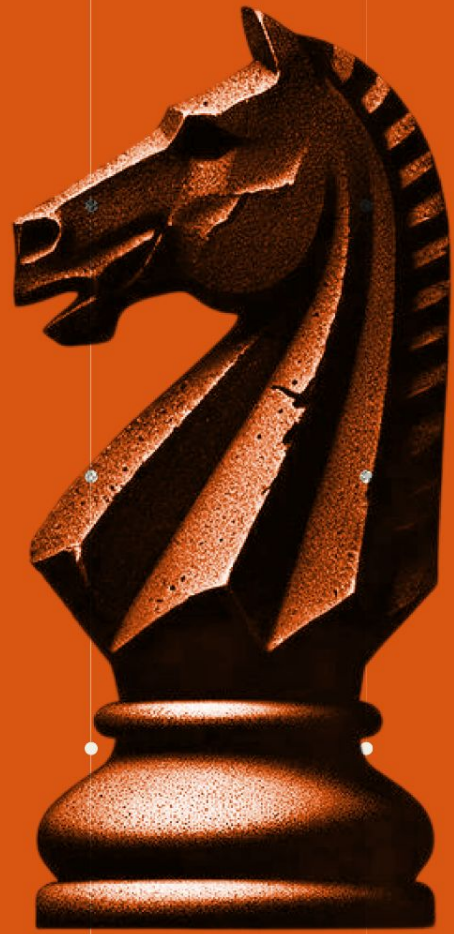
KTC

KRACK THE CASE

BUSINESS GUIDE

STRATEGY • STRUCTURE • SOLVE



Consulting & Analytics Club
IIT Guwahati

Disclaimer

All information in this guide is accurate and complete to the best of our knowledge. Any data, definitions or inspirations have been acknowledged wherever possible. Any redistribution or reproduction of part or all of the contents in any form is prohibited.

If you find some discrepancy in the material, feel free to contact us at cac@iitg.ac.in

---

© Consulting & Analytics Club, IIT Guwahati. ***All rights reserved.***
This edition was published in June 2026.

Developed by the Consulting, Product, and Design Team at Consulting & Analytics Club, IIT Guwahati.

Visit the official page of Krack the Case at iitg.ac.in/sa/caciitg/ktc/.

WE ARE C&A

About Consulting & Analytics Club, IIT Guwahati

The Consulting & Analytics Club at IIT Guwahati (C&A) aims to facilitate a seamless transition from graduate school to a successful career in management consulting, product and data analytics through awareness, skill-building, and networking. We aim to impart and inculcate our culture of learning-by-practice among students and professionals alike. To that end, C&A supports and organises a multitude of events throughout the year, including webinars, AMAs, workshops, hackathons and case challenges.

cac@iitg.ac.in /company/caciitg caciitg.com

What's new in the 2026 Edition

We're proud to present the second edition of ***Krack the Case: The Business Guide*** — a comprehensive and accessible resource built to support students aspiring for careers in consulting, product management, strategy, and other non-tech domains.

This guide is part of our flagship initiative, *Krack the Case*, and has been designed to cover every aspect of non-tech preparation. It brings together the core theoretical foundations of **consulting, product, and economics** along with practical insights such as **real interview transcripts (across multiple rounds), ideal resumes, sample cover letters**, and everything else you'll need to navigate your placement journey. This edition newly features an easy-to-follow 8-step Product Teardown methodology, a one-stop last-minute revision SQL Cheatsheet, and more.

Alongside the Business Guide, we'll also be launching a dedicated *Business Casebook*, focused purely on case interview practice and structured frameworks, supported by just the right amount of theory. Together, the Business Guide and Casebook are meant to form a complete preparation loop: *refer to the guide for concept-building, and the casebook for practical application*.

We hope this guide becomes your go to starting point — a foundational companion that simplifies theory, structures preparation, and helps you tackle every round with clarity and conviction.

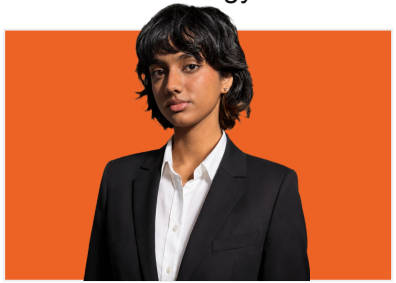
Editorial Team

The club gratefully acknowledges the invaluable contributions of the team behind the 2026 edition of the Business Guide, and extends heartfelt appreciation to the alumni of C&A whose continued guidance and legacy made this possible.

Dhairya Nisar
Head of Consulting



Keerthana Mahesh
Head of Product & Business Strategy



Tanmay N
Creatives Head



Yashvi Mehta
Institute Secretary



Prakhar Gupta
Overall Coordinator



Sanchit Pendharkar
Outreach Head



How to use the guide?

To make the most of this guide, we recommend familiarizing yourself with the structure and flow of its content. The guide is designed to support both foundational learning and deep dives across multiple non-tech domains, including consulting, product management, and analytics.

Section 1: Consulting

Start here if you're new to consulting or want to strengthen your basics. It covers everything from the roles of consultants and sectors of consulting to structured problem solving. Use this section to:

- Understand core consulting mindsets, tools, and frameworks.
- Master guesstimates, MECE breakdowns, and structured case-solving.
- Practice with solved examples and explore advanced frameworks like M&A or Private Equity.

Section 2: SQL

This section is ideal for PMs and analysts alike. It breaks down SQL concepts:

- Start with what SQL is and tackle commonly asked interview queries.
- Explore the **different types of SQL**, common **constraints**, and various types of **joins** — all explained in an intuitive, easy-to-apply manner.
- Explore how SQL intersects with product management responsibilities.
- Use the cheat sheet for quick reference and revision.

How to use the guide?

Section 3: Product

This is your go-to for understanding the PM role in-depth:

- Learn what a PM does, key career paths, and mental models like JTBD, OKRs, HEART, etc.
- Dive into simplified tech concepts: APIs, system design, frontend-backend, Git, CI/CD, and more.
- Strengthen product thinking through teardown exercises, MVPs, and product strategy frameworks.

Section 4: Building Your Profile & Interview Prep

Use this section to prepare for placements and interviews:

- Get guidance on resumes, cover letters, and impactful project building.
- Prepare for PM, consulting, and behavioral interviews with step-by-step frameworks and tips.
- Learn how to crack VP interviews, technical rounds, and case interviews confidently.

Appendix

This part contains curated resources such as useful formulae, quick-reference data points, and additional reading materials to support your revision and last-mile prep.

TABLE OF CONTENTS

Click on the headings to navigate between topics.

Section 1: Consulting

- **Consulting: The Basics and Beyond**
  - Overview — Roles of a Consultant
  - Sectors of Consulting
  - Career Paths in Consulting
  - Developing a Case Interview Mindset
  - The Pyramid Approach to Communicating Ideas
  - The Pareto Principle
  - Solving Case Interviews — Evaluating your approach
  - General Tips
- **Case Structuring Tools**
  - Issue Trees
  - MECE Approach
  - Creating Issue Trees
  - Solved Examples
- **Guesstimates**
  - What's a Guesstimate
  - General Method
  - Guesstimate Tips & Tricks
  - Guesstimate Cheat Sheet
  - Solved Examples
  - Previous Year's Guesstimates
- **Consulting Tools**
  - SWOT Analysis
  - PESTEL Analysis
  - BCG Growth-Share Matrix
  - GE-McKinsey Nine-Box Matrix
  - Balanced Scorecard
  - Value Chain Analysis
  - Benchmarking
  - Task Environment
  - Structure–Conduct–Performance Model
  - 3Cs Model
  - McKinsey 7S Model
  - 4Ps of Marketing
  - Miscellaneous
- **Case Frameworks**
  - Profitability
  - Market Entry
  - Pricing Strategy

TABLE OF CONTENTS

- Growth Strategy
- Market Sizing
- Organisational Capacity Expansion
- Mergers and Acquisitions
- Private Equity Investment

Section 2: SQL

- **Understanding SQL**
  - What is Structured Query Language
  - Constraints in SQL
  - Joins in SQL
  - FAQs
- **SQL from a PM Lens**
  - A Product Manager's Perspective on SQL
  - SQL Cheat Sheet

Section 3: Product

- **Introduction to Product Management**
  - What is a PM?
  - What does a PM do?
  - What is a PM Trying to Solve?
  - Types of Products a PM Might Encounter & How They Affect the PM Role
  - Types of Product Managers
  - The PM Career Path
  - Transitioning from Engineer to Product Manager
  - What Companies Look for – The PM of the Future
  - Top Myths About Product Management
- **Building a PM Mindset**
  - The PM Mindset
  - Five Stages of User Journey
  - The JTBD Framework
  - The CIRCLES Framework
  - The North Star Metric
  - The OKR Framework
  - The Kano Model
  - The MoSCoW Prioritization
  - The RICE Prioritization
  - The Product Life Cycle
  - The Hook Model
  - The HEART Framework
  - The AARRR Framework
  - Ansoff Matrix

TABLE OF CONTENTS

- User Research Methods
- Selecting & Interpreting Metrics

- **Tech Simplified for PMs**
  - ***Part 1: Foundations of the Web and Internet***
  - How the Internet Works
  - Client, Server, DNS, IP Addresses
  - HTTP, HTTPS, TCP/IP, SSL
  - 3-Tier Architecture (Frontend, Backend, Database)
  - How the Web Works (Client-Server Model)
  - Web Protocols & Standards
  - HTTP, REST, OAuth
  - Cookies & Sessions
  - Status Codes & Methods (GET, POST, etc.)
  - ***Part 2: Core Tech Stack & Tools***
  - Frontend Technologies (HTML, CSS, JavaScript, Frameworks)
  - Backend Technologies (Server-side logic, APIs & Middleware)
  - Databases (SQL vs NoSQL, Schemas, Redis)
  - Tech Stack Overviews (MERN, LAMP, MEAN)
  - ***Part 3: Development & Version Control***
  - RESTful APIs (Endpoints, Methods, Auth)
  - Webhooks
  - API Testing Tools – Postman
  - API Product Management
  - SDK vs API
  - CDNs
  - ***Part 4: APIs, Webhooks & Communication***
  - Git & GitHub (Versioning, Branching, Merge Conflicts)
  - Feature Flags
  - CI/CD Pipelines & DevOps Essentials
  - Jenkins, Docker, Kubernetes
  - ***Part 5: System Design & Scalability***
  - High-Level System Design (Flows, Architecture)
  - Scaling Systems (Horizontal vs Vertical)
  - Databases in Scalable Systems (Sharding, Replication, CAP)
  - Non-Functional Requirements – Latency, Availability, Security

TABLE OF CONTENTS

- ***Part 6: Tech Concepts Every PM Should Know***
- AI, ML, Deep Learning
- Generative AI
- Cloud Computing
- Blockchain
- AR/VR
- Internet of Things (IoT)
- Edge Computing

- **Mastering Product Thinking**
  - Product Design Frameworks (CIRCLES, HEART)
  - Problem-Solving Exercises
  - Architecture of Popular Apps
  - App Teardowns / Analysis
  - Building MVPs (Minimum Viable Products)

Section 4: Building Your Profile and Acing the Interviews

- **Profile Building**
  - How does one get shortlisted?
  - Resume Tips
  - Building a Non-Tech CV
  - Cover Letters
  - Guide to Amazing Presentations
  - Crafting Impactful PM Projects
  - How to Make Good PM Pitch Decks
- **Interview Preparation**
  - How to Approach a PM Interview
  - Recommended Framework for PM Interviews
  - How to Approach A Technical Interview Question
  - Common Technical PM Interview Questions
  - How to Approach A Behavioral Interview Question
  - Common Behavioral PM Interview Questions
- **Mastering the Interview**
  - Acing the PM Interviews
  - PM Interview with the VP of the Product
  - Acing the Case Interviews
  - Interview Experience & Transcripts

Appendix

Recommended Books

FOCUS

01

CONSULTING:

The Basics and Beyond



Overview — Roles of a Consultant

Consulting is more than just giving recommendations on solving the client's dilemma. Although that is the primary function of a management consultant, there are more roles linked with their expertise.



- Understand the underlying needs of the client
- Effective diagnosis possibly involving redefinition of the problem
- Recommending an action plan based on the diagnosis
- Assisting with the implementation of the action plan created
- Continuous evaluation of the recommended solutions in future

Sectors Of Consulting



Focuses on supporting private sector clients with the development of corporate, organizational or functional strategies

Strategy



Relates to services aimed at helping clients on how they utilize Information technology to optimally achieve their business goals

Technology



Provide advisory and implementation services that improve a company's internal operations and performance in the value chain

Operations



Pertains to assisting enterprises to identify, manage and mitigate IT & enterprise compliance risk

Risk/Compliance



Advises the executives of companies to improve the effectiveness of business strategies and operational processes

Management

Career Paths in Consulting (Entry to Senior Roles)



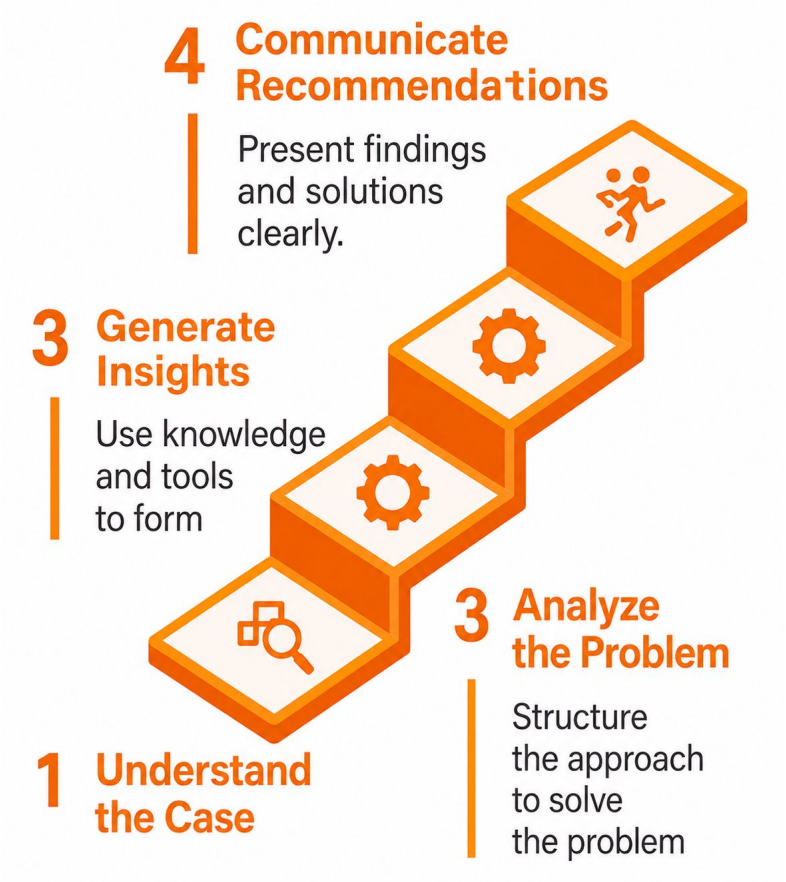
Developing a Case Interview Mindset

Practice Hypothesis-Driven Thinking

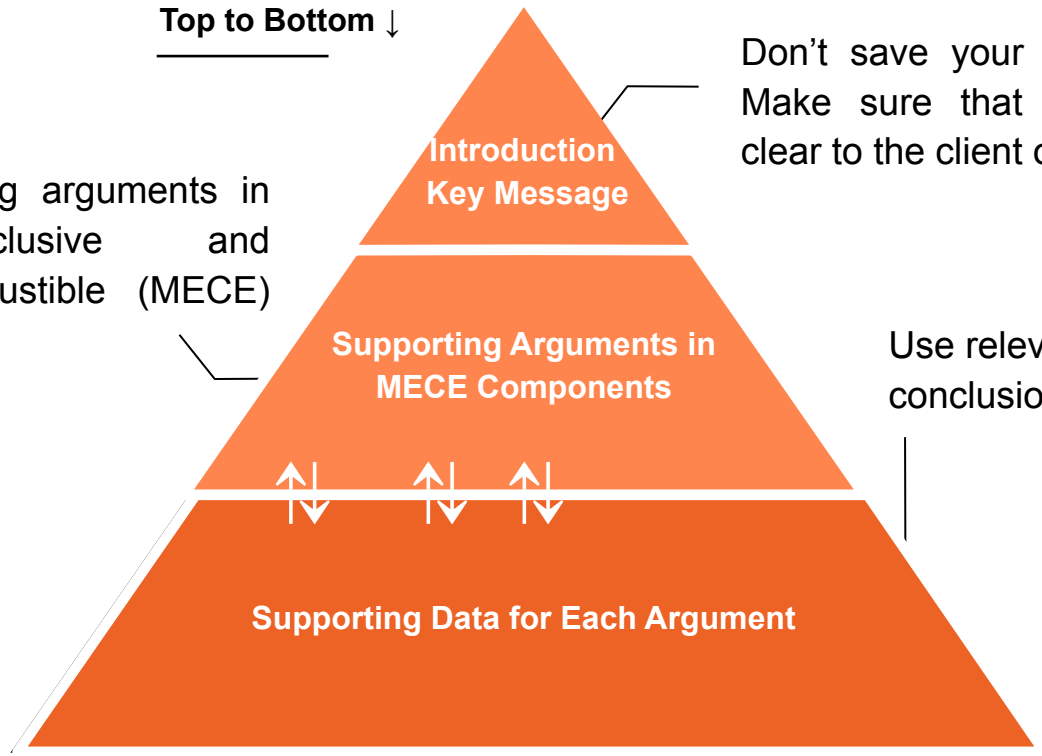
- Always have a working hypothesis and refine it as new data emerges.
- Don’t aimlessly explore—have a direction and adapt as needed

Communicate Like a Leader

- Speak clearly, prioritize insights, and summarize often.
- Think aloud to showcase your logic and invite collaboration.



The Pyramid Approach to Communicating Ideas



Don't save your best for the last. Make sure that your intention is clear to the client or interviewer.

Deliver supporting arguments in mutually exclusive and collectively exhaustible (MECE) components.

Use relevant data to derive conclusions.

The Pyramid Principle

The Pyramid Approach to Communicating Ideas

As a consultant effective communication is very important.

A powerful technique for the same is pyramid principle -

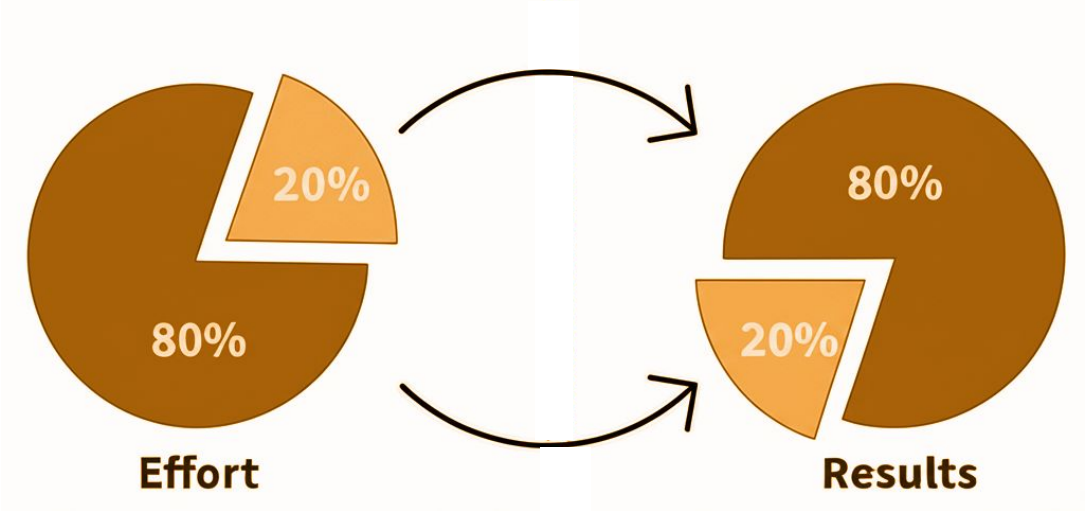
How it works

- Start with the **main conclusion or recommendation**
- Follow it with **supporting arguments**, grouped into logical categories
- Use a **top-down** structure to guide the audience through your reasoning

When to Use the Pyramid Approach:

1. Business & Consulting Communication
2. Interviews (especially Consulting & Product)
3. Team & Stakeholder Discussions
4. Writing & Presentations

The Pareto Principle



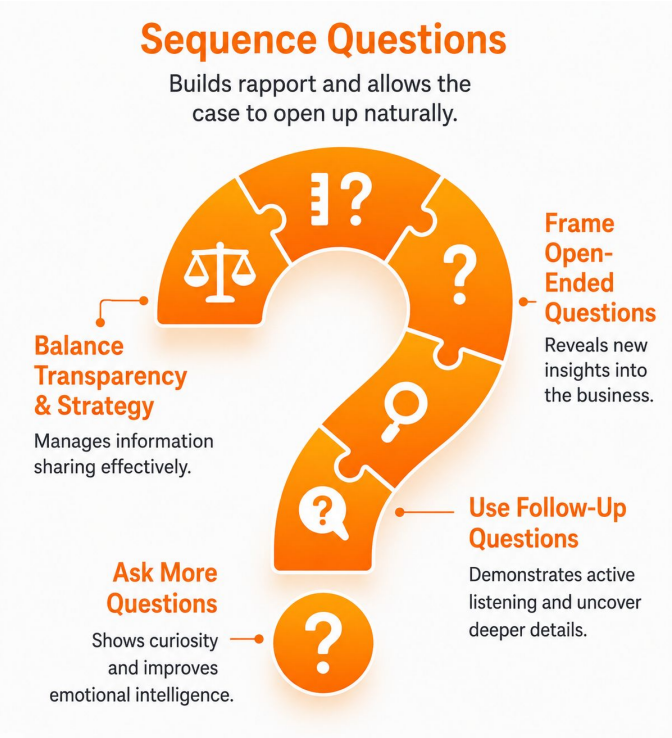
What is Pareto's Principle?

A principle stating that **80% of outcomes come from 20% of causes** or inputs

How it applies:

- In business, **80% of sales** often come from **20% of customers**.
- In productivity, **20% of tasks** yield **80% of results**.
- In quality control, **80% of problems** stem from **20% of issues**.

Solving Case Interviews — Evaluating your approach



Recommended Reads | Harvard Business Review -The Surprising Power of Question`s

THE PROBLEM

Some professionals such as litigators, journalists and even doctors, are taught to ask questions as part of their training. But few executives think about questioning as a skill that can be honed. That's a missed opportunity.

THE OPPORTUNITY

Questioning is a powerful tool, it fuels innovation and better performance, it builds trust among team members. And it can mitigate business risk by uncovering.

THE APPROACH

Favor follow-up questions, know when to keep questions open-ended, get the sequence right, use the right tone, and pay attention to group dynamics.

General Tips

- Be prepared with different tools and case frameworks. Practice a few cases beforehand and stay updated on industry news.
- Communication skills would be of utmost importance. While solving the case, **think out loud!**
- Ask clarifying questions in the beginning. Asking the right questions increases the chances of finding an accurate solution. Also, don't forget to take notes throughout the process.
- Do not make any assumptions unless necessary; even if you do, communicate with the interviewer to get them validated.
- Be calm, confident and patient with your approach. Take time to compose your thoughts — don't just start talking.
- Do not get stuck on calculations! Round off numbers, and use them quickly and efficiently.
- Whenever a framework is used, communicate the approach to solving the case rather than mentioning the exact framework.
- Don't panic — engage the interviewer in a straightforward conversation.
- Keep communicating with the interviewer to let them know of your thought process. In most cases, the interviewers will help you get back on track if they see you going in the wrong direction.

FOCUS

02

CONSULTING:
Case Structuring Tools



Issue Trees

Issue trees are considered the bread & butter of a consultant's life. Making an issue tree is essential for **mapping** the problem. This begs the question, what are issue trees?

In simple terms, it's a method of dissecting a problem to its root cause and finding probable solutions logically. An issue tree is an efficient and logical tool to answer quantitative concerns. Moreover, they aid in categorising ideas which cover all possibilities without going deep into minute details. The issue tree and each subtree are a **MECE** structure *(see the next page)*.

Types of Issue Trees



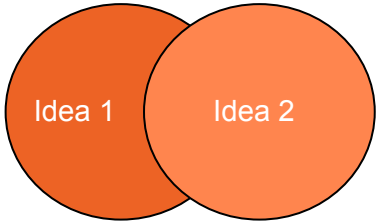
Diagnostic Trees
Answers the question 'why' and identifies and categorises all possible root causes for the problem.



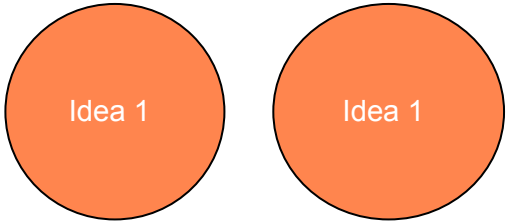
Solution Trees
Answers the question 'how' and identifies all possible alternatives to fix the problem.

MECE Approach

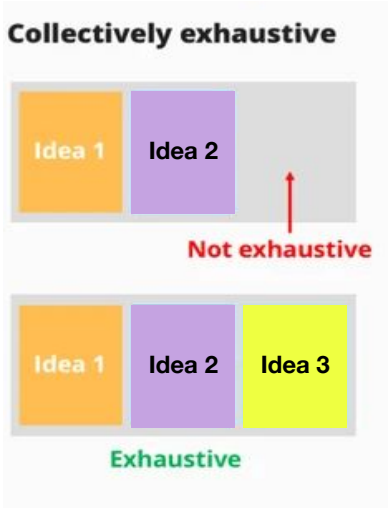
MECE stands for Mutually Exclusive and Collectively Exhaustive.

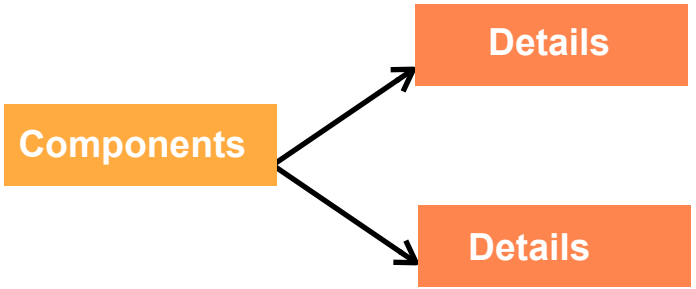


Not Exclusive



Exclusive





The components are listed vertically and progress into details as it reads to the right. While creating issue trees using the MECE approach, one should also ensure that the sub- objects that the issue is divided into belong to the same class, and one should not use too many sub-objects.

Creating Issue Trees



1. **Divide and Conquer:** Start with finding the root cause of the problem and divide it into subproblems keeping in mind the MECE approach.
2. **Lay down a structure:** Use the MECE approach to segment information. Add mini MECE layers to each subsequent layer. This will help to solve the problem efficiently at each layer.
3. **Continue until you solve the problem.**

Solved Examples

Problem Statement 1

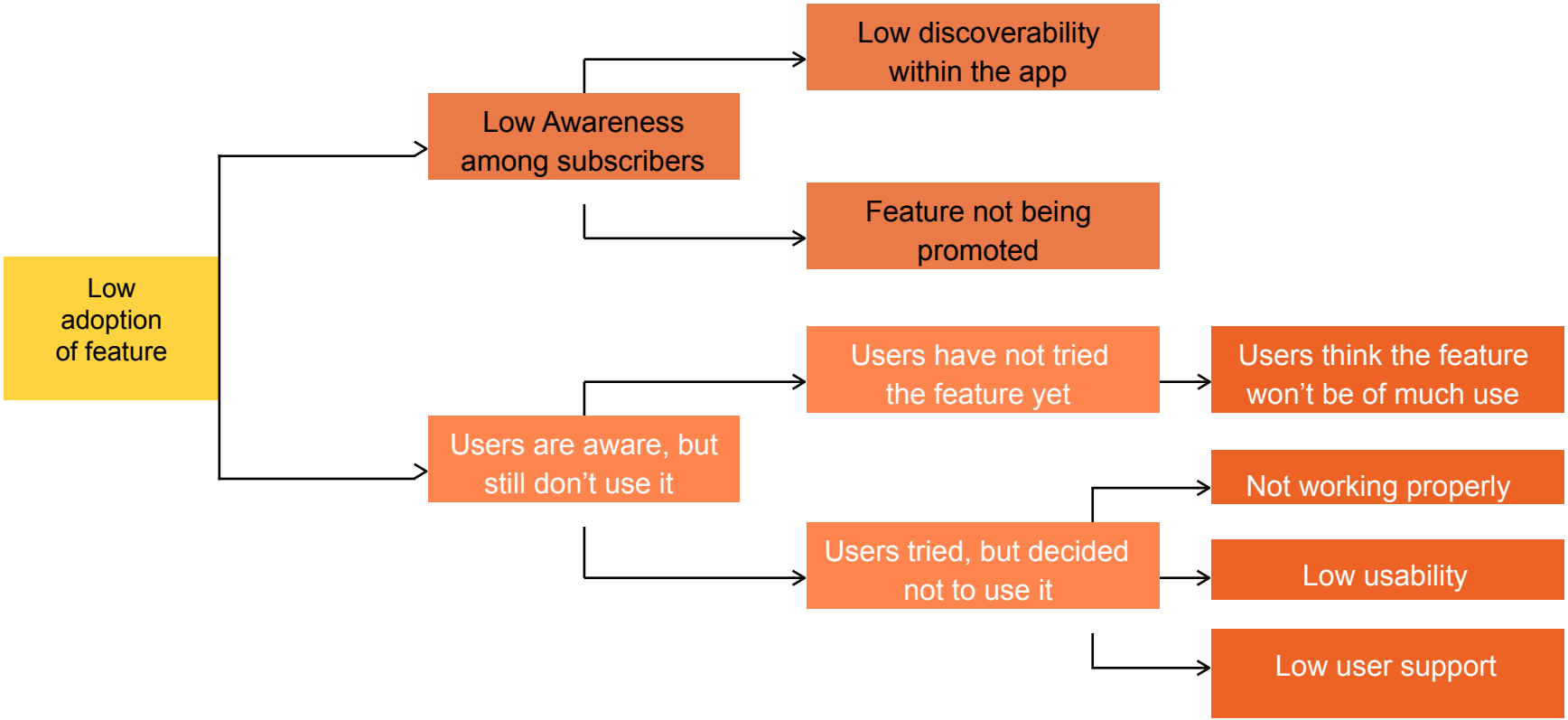
Paramount + recently updated its streaming app to include an Intelligent Recommender Feature. However, this feature is **facing low adoption among its subscribers**. Find out the reasons why.

Clarifying Questions

1. How long ago did the company update the app?
   The app was updated just over three months ago.
2. Is this issue limited to specific geography?
   No, the problem is not isolated to a particular region.
3. Do other streaming services offer this feature?

   Netflix and Disney+ provide similar features on their apps, but not on web players. In addition to our apps, we are offering this feature on all supported browsers.

Structure


Low adoption of feature
Low Awareness among subscribers
Low discoverability within the app
Feature not being promoted
Users are aware, but still don't use it
Users have not tried the feature yet
Users think the feature won't be of much use
Users tried, but decided not to use it
Not working properly
Low usability
Low user support

Problem Statement 2

Estimate if the new targeted ad structure yields more revenue than the current one of a company which sells monthly magazines in India.

Clarifying Questions

1. What is the current ad revenue structure?
   We include full-page ads in our magazine and get paid for by the company whose ad it is. Today, we sell 1M copies. Each magazine has 100 ads. Each ad costs $50 per 1000 copies.

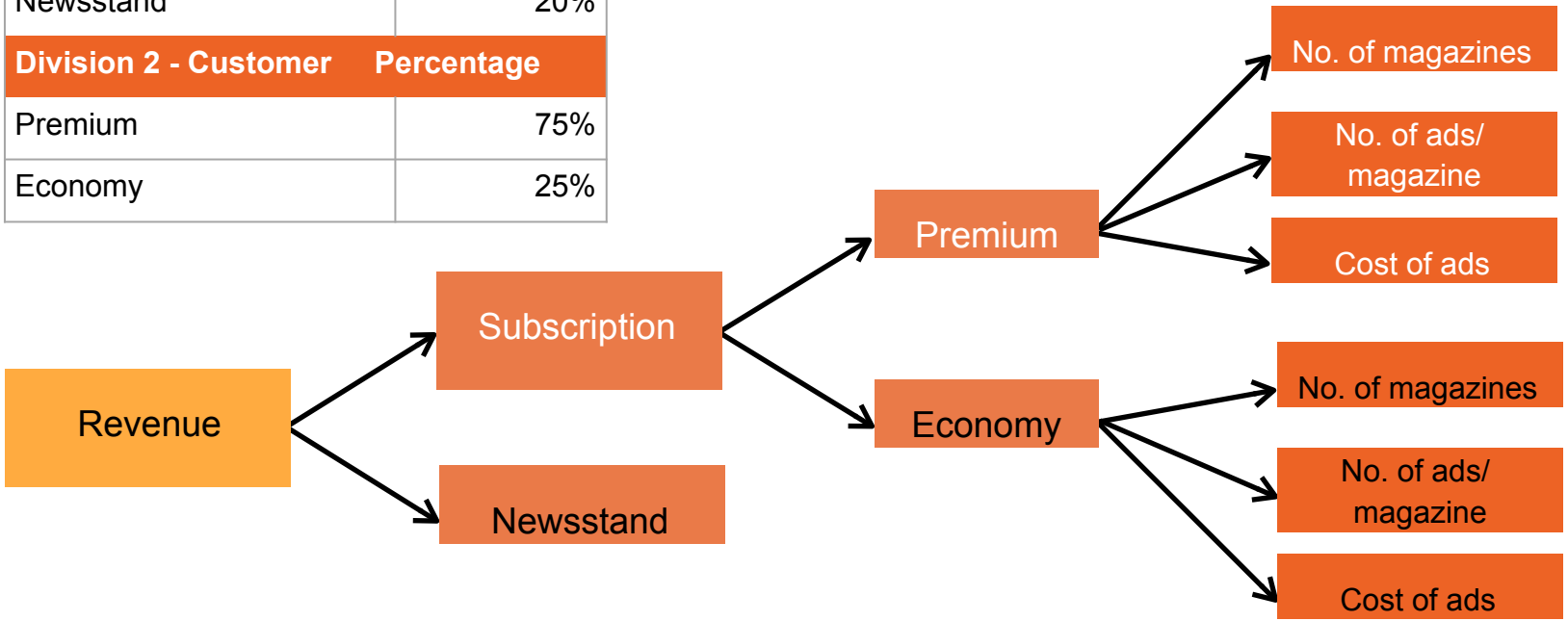
2. What exactly do you mean by targeted ads?
   We have data on our customers based on which we would send relevant ads to them.

3. What are the categories based on which you want to target ads?
   We have premium brands' ads and non-premium brands' ads today. The proposal discusses sending premium brands' ads to upper class/ premium customers and non-premium brands' ads to economy customers.

Structure

Division 1 - Revenue	Percentage
Subscription-Based	80%
Newsstand	20%
Division 2 - Customer	**Percentage**
Premium	75%
Economy	25%

Cost = No . of Magazines × No . of Ads per Magazine × Cost of ads



FOCUS

03

CONSULTING:
Guesstimates



What is a Guesstimate?



A **Guesstimate** is a blend of **"guess"** and **"estimate"**, commonly used in consulting, product management, and case interviews to **arrive at a rough, logical estimate** of a number or quantity when exact data is not available.

What It Tests:

- Logical thinking and structured problem-solving
- Comfort with numbers and mental math
- Business intuition (e.g., consumer behavior, market sizing)
- Clarity in communication

General Method (1/2)

Step 1



Clarifying the Scope

Anything that is/may seem ambiguous must be cleared right away. Ask questions! Smart questions show that you have an eye for detail. You need to identify the constraints you are working under to answer clearly and move through the case faster.

Step 2



Developing a Strategy

Think of an overall strategy that exhaustively covers the issue at hand. Try to think from different perspectives as well. Generally, three methods are used (with tailored variations) — Top-Down Method, Bottom-Up Method, and Employing a Proxy. Base your strategy on general 2-3 element formulae (such as averages, demand, and supply).

General Method (2/2)

Step 3



Break down the problem

Once you identify the constraints you must work under, break down the problem into smaller pieces (a MECE approach of-course). Do keep in mind the following factors while breaking it down into pieces — Population, Income split, Age split, Gender, and Rural- Urban split.

Step 4



Perform Math & Sanity Checks

Estimate each piece separately and multiply them to get the overall estimate. Perform sanity checks mentally wherever there is “sensitive data” and suggest sanity checks on your own to the interviewer. Making sure your answer is realistic shows that you are methodical in your approach.

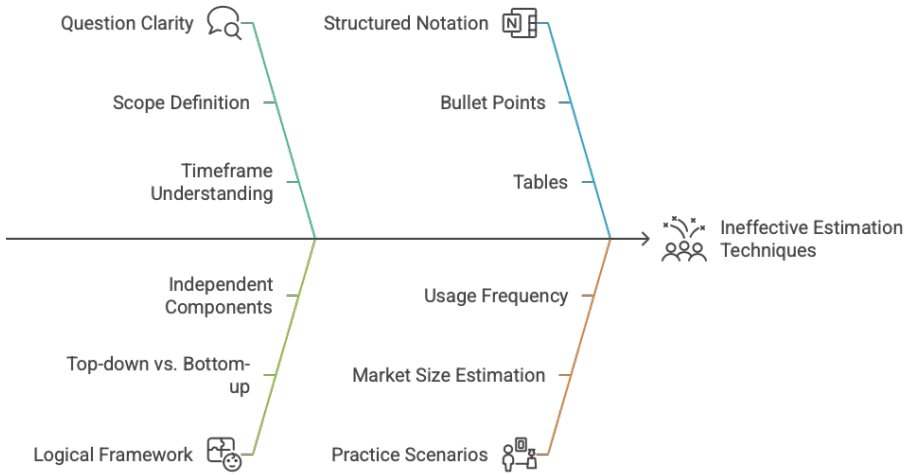
Guesstimate Tips & Tricks

1. Clarify the Question

- Don’t jump in—**ask clarifying questions**.
- Make sure you understand the scope: **"India or just urban cities?"**, **"Per day or per month?"**

2. Use a Logical Framework

- Choose a **Top-down** (start from population, break into segments) or **Bottom-up** (start with one unit, multiply up) approach.
- Break it into **independent components** like:
  - Population
  - Penetration (% users)
  - Frequency (usage/day)
  - Value per transaction



Guesstimate Tips & Tricks

3. Round Numbers Smartly

Use simple figures like **1.3 billion → 130 crore → 130**, to speed up mental math.

Approximate calculations: use **10% instead of 9.6%**, or **0.33 for 1/3**.

4. State Assumptions Clearly

Every number must come with a **justified assumption**.

“Assuming 60% of Indians are in urban areas...”

5. Sanity Check Your Answer

- After calculating, ask: **“Is this reasonable?”**
- Compare with known benchmarks or gut.

Guesstimate Tips & Tricks

6. Use Structured Notation

Break your answer down like:

- Step 1: Estimate total users
- Step 2: Estimate usage
- Step 3: Multiply for total volume

7. Practice Common Scenarios

Estimate market size for toothpaste, number of flights per day in India, water bottles sold daily.

8. Stay Calm Under Pressure

- You’re being tested for **thinking style**, not perfect numbers.
- Speak your thought process clearly and confidently.

Guesstimate Cheat Sheet

World		
Total Population	7,300 0000	7,3 billion
Gender		
Male	50%	650 billion
Female	50%	655 billion
Continents		
Asia	60%	4.3 billion
Africa	15%	1.09 billion
Europe	10%	730 million
Latin America	10%	730 million
North America	5%	255 million
Age Groups		
0–14	10%	730 million
6–14	15%	1.09 billion
15–35	40%	2.9 billion
36–60	20%	1.46 billion
Above 60	15%	730 million
Income Group		
Poor	15%	1.09 billion
Low	55%	4. billion
Middle	15%	1.02 billion
High	0,5%	365 million

India		
Total Population	1,300,000	1,3 billion
Gender		
Male	50%	650 million
Female	50%	650 million
Rural Urban Divide		
Rural	70%	910 million
Urban	30%	350 million
Major Cities		
0–14	30%	330 million
15–35	33%	450 million
36–60	25%	325 million
Above 60	5%	65 million
Marital Status		
Married/Others	50%	650 million
Never Married	50%	650 million
Income Group		
Poor	20%	260 million
Low	75%	925 million
Middle	4%	52 million
Upper Middle	1%	13 million
High	0,5%	6,5 million

Solved Examples

Problem Statement 1

Estimate the number of routers sold in India in a year.

Clarifying Questions

1. Should I consider the impact of the COVID-19 pandemic, or is scenario one during the pre- COVID times? *(Consider post-pandemic times)*
2. By router, do you mean the traditional router where I should also include a modem that comes with it or the modern one which does both the work of a router and a modem? *(Consider both)*

Notes

1. Always proceed with caution and be mentally aware of what you are doing.

2. Think of an overall strategy (You could try thinking of supply/demand/geography demographics, but these approaches involve too many variables). So, we will proceed by considering the end user demographic, i.e. residential and commercial users.
3. Since we have talked about COVID-19 and its impact, we must necessarily include the growth of the Internet over the years (be it post-COVID or pre-COVID), bringing in new users.
4. We must now consider approaches to solve both the pieces (Residential and Commercial components) by breaking them down further.
5. We must also factor in the lifetime of a router and the replenishment rate.
6. Keeping in mind all the above points, we now base our guesstimate on the following two formulae:

1. $$\text{Number of existing routers} = \frac{\text{Number of users}}{\text{Average number of people using a router}}$$

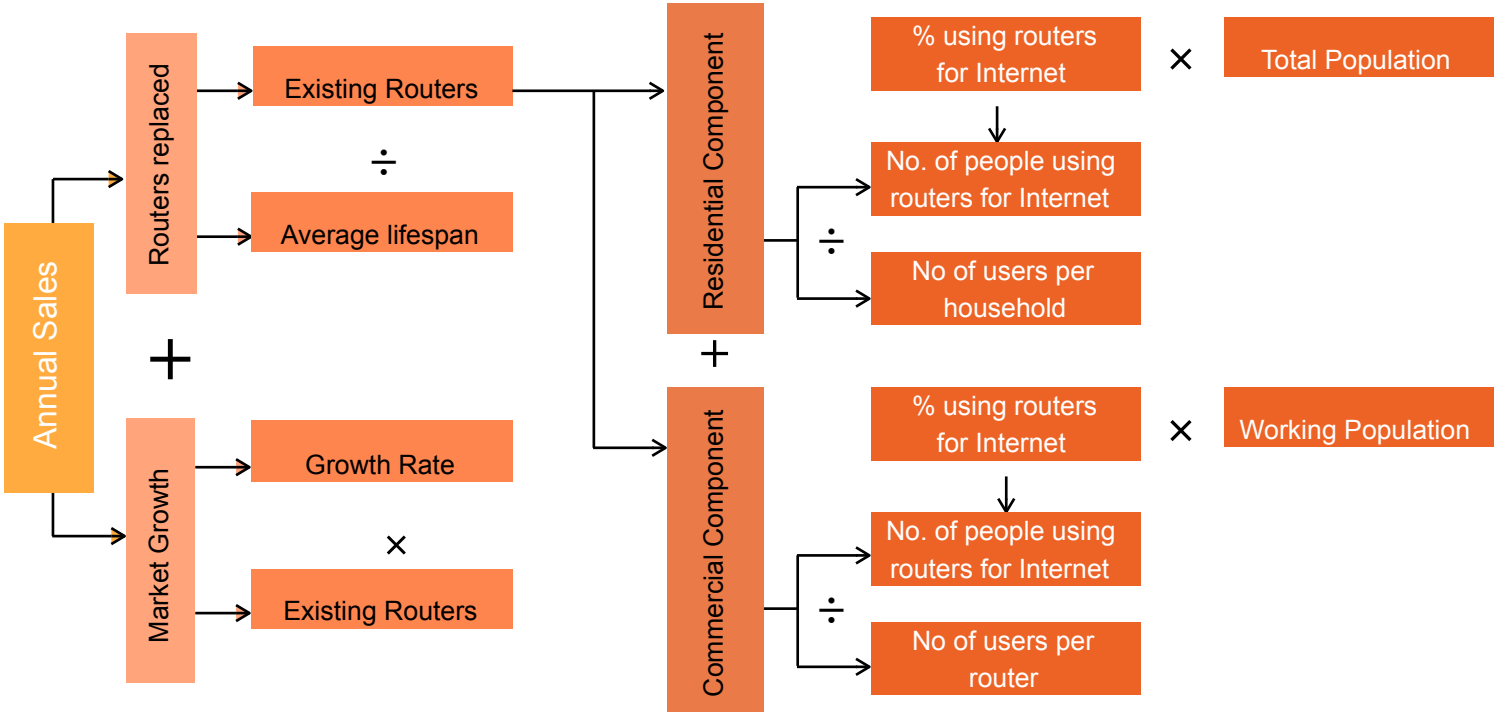
2. $$\text{Number of new routers added} = \text{Market Growth} + \text{Routers Replaced}$$

$$= (\text{Number of existing routers} \times \text{Growth rate}) + \frac{\text{Number of existing routers}}{\text{Lifetime}}$$

Considerations

1. Internet growth during the pre-COVID times was around 4%, and that during post-COVID times was about 10%.
2. The percentage of users who have access to the internet in India pre-COVID was 40%, and post-COVID was around 50%.
3. The workforce of India is around 35% of its population, and the percentage of people working in unorganised and other blue-collar jobs is about 80%. India's population can be considered to be 1.40 billion.
4. The average lifetime of a router is five years.

Structure


Annual Sales
Routers replaced
Existing Routers
÷
Average lifespan
+
Market Growth
Growth Rate
×
Existing Routers
Residential Component
% using routers for Internet
×
Total Population
No. of people using routers for Internet
÷
No of users per household
+
Commercial Component
% using routers for Internet
×
Working Population
No. of people using routers for Internet
÷
No of users per router

Calculations

Residential Component

Population having access to internet = 50% = 700 million

20% of 700 million rely on routers = 140 million

Apply Formula 1 → say if 5 people share a router →
Number of existing routers = 140/5 = 28 million

Apply Formula 2 →
Number of New Routers
= (28 x Growth rate) + (28 / Lifetime)
= (28 x 0.1) + (28 / 5) = 8.4 million

Commercial Component

Working population of India = 35% = 500 million

People working in White and Grey Collar Jobs = 20% = 100 million

No. of people with access to router = 70% of 100 million = 70 million

Apply Formula 1 → say if 10 people share a router →
Number of existing routers = 70/10 = 7 million

Apply Formula 2 →
Number of New Routers
= (7 x Growth rate) + (7 / Lifetime) =
(7 x 0.1) + (7 / 5) = 1.9 million

Total number of routers sold in India every year

= Residential Component + Commercial Component

= 8.4 + 1.9
= 10.3 million
~ 10 million

Problem Statement 2

What is the market size of sofas in India?

Clarifying Questions

1. Does market size refer to volume or value? *(Consider the value of sofas sold in a year)*
2. Are single-seated armchairs included here? *(By definition, no.)*
3. Sofas sold or produced? *(Consider only the sofas sold in India.)*

Considerations

1. Sofas are, of course, needed in homes. Other possible uses are – in offices, airports, lounges, cafeterias, and hotels. Since India has many households, the lion's share of sofas is for residential purposes. *(An adjustment may be applied to the final answer to account for the rest of the possible users – say 30%)*.

2. Since the problem requires the value of sofas *sold* in a year, it is helpful to assume that there is no bottleneck in supply. Thus, an approach from the demand side would be the most useful.

3. Consider the population of India as 1.4 billion and an average of 5 people per household.

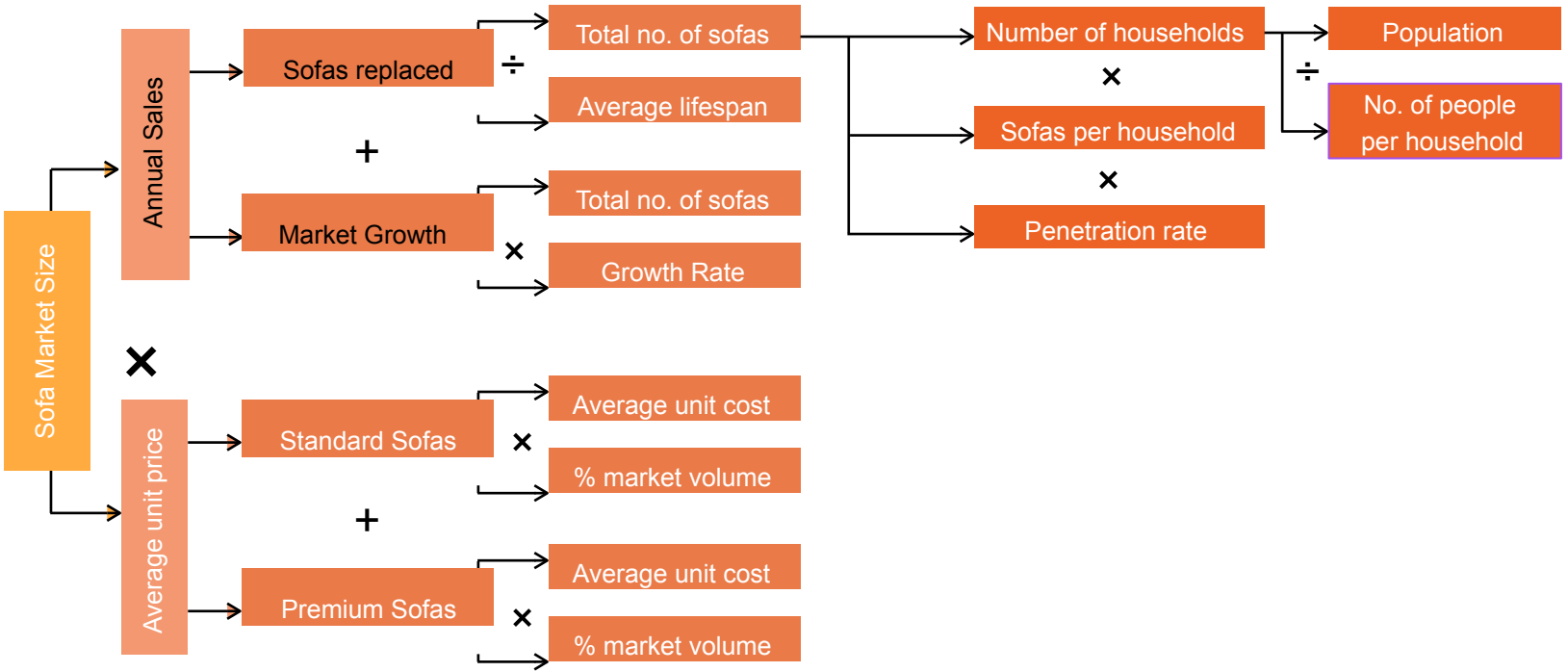
4. Higher-income households may have more than one sofa. On average, assume two sofas per household.

5. Penetration Rate: Consider the 30:70 split of urban and rural populations in India. Let's say this applies to families too. Assume 30% of urban and 1% of rural households have sofas. This amounts to approximately 10% penetration.

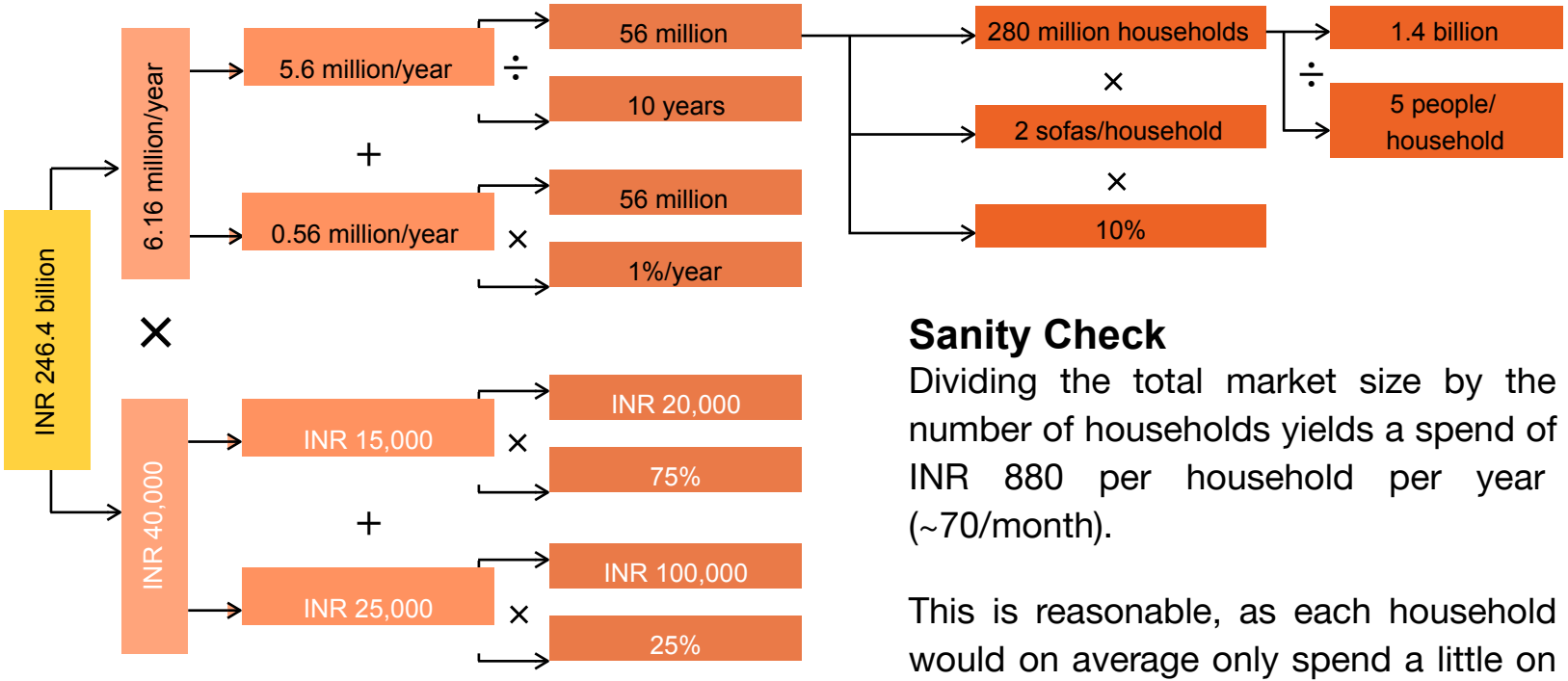
6. Assume sofas get replaced every ten years on average. Assume the sofa market grows parallel to population growth. Thus, 1% market growth.

7. Standard sofas may cost INR 20k, with a 75% market volume. Premium sofas may cost five times as much.

Structure


Sofa Market Size
Annual Sales
Sofas replaced
÷
Total no. of sofas
Average lifespan
+
Market Growth
×
Total no. of sofas
Growth Rate
×
Average unit price
Standard Sofas
×
Average unit cost
% market volume
+
Premium Sofas
×
Average unit cost
% market volume
Number of households
×
Sofas per household
×
Penetration rate
÷
Population
No. of people per household

Calculations



Sanity Check

Dividing the total market size by the number of households yields a spend of INR 880 per household per year (~70/month).

This is reasonable, as each household would on average only spend a little on sofas (a fairly long-term purchase).

Previous Year Guesstimates

McKinsey & Company	Bain & Company
1. Estimate the colored TV sales for Flipkart working during Diwali season (in India). 2. Estimate toll collection outside Bangalore people airport (in a week on average). alone. 3. Estimate the annual logistics cost for Flipkart India (in India). 4. Estimate the annual budget required for would textbooks for students in Karnataka. 5. How much change would you find on the floor of an average mall (today)? textbooks 6. How many second hand wedding dresses are Antarctica? sold in the UK each year?	1. Guesstimate the number of laptops in condition in Delhi right now. 2. Calculate the number of watches that currently own in Delhi region 3. How many cups of tea are sold in all of every week? 4. Estimate the number of flying cars that be sold in 2025! 5. Estimate the annual budget required for for students in Karnataka. 6. How many penguins can live in

Previous Year Guesstimates - Contd.

Boston Consulting Group

1. Estimate the size of the e-pharmacy market in India.
2. How many Mazda dealers are there in the US?
3. Estimate the size of chewing gum market in the US.
4. Describe how to measure the average time taken in a queue at an airport check-in counter.
5. Estimate the size of the quick commerce market in India.
6. How many hotel-sized mini bottles of shampoo and conditioner are produced each year around the world?
7. How many people in Uganda have no access to Electricity?
8. How many pay-phones are there on the Island of Manhattan?

Oliver Wyman / Marsh McLennan / Mercer

1. How many miles of road are there in the US?
2. How many people are travelling in an aeroplane in the UK right now?
3. How much gasoline does a gas station sell per day?
4. How many songs are stored on iPods in the UK?
5. How many words are there in the daily edition of The Times?
6. Are there two dogs in the world with the same number of hairs?
7. How many spikes does a hedgehog have?
8. How many taxicabs are there in New York City?

Previous Year Guesstimates - Contd.

The Big Four (Deloitte, PwC, EY, KPMG)

1. How many medium-size industries are there in Mumbai? (Deloitte)
2. How much packets of noodles are consumed in a month at a college campus? (Deloitte)
3. How many people smoke cigarettes in Hyderabad? (Deloitte)
4. Estimate the number of people who have ever lived on earth. (Deloitte)
5. Estimate the number of flights taking off in a day from Delhi Airport. (Deloitte)
6. Estimate the number of Maggi packets sold on the first day of relaunch (Deloitte)
7. How many gallons of ice cream are sold in the US every year? (PwC)
8. What is the market size of lighting fixtures in Middle-East? (PwC)
9. Calculate the capacity of a second airport at New Delhi. (PwC)
10. How many streetlights are there in New York City? (PwC)
11. What is market size of hand knitting yarn in the US? (EY)
12. How much kilograms of canned fruit is sold by Del Monte Foods in the US? (EY)
13. How much it will cost the Bank of New York Mellon to overhaul its IT System? (EY)

Previous Year Guesstimates - Contd.

Accenture	ZS Associates
1. How many basketballs are purchased by the NBA every year? 2. How much to pay for the world's only Dinosaur? 3. Estimate the number of customers Flipkart gained due to the Big Billion Days sale. 4. Estimate the number of dry cleaners in the city of Philadelphia. 5. How many airplanes flying the opposite route will you pass on the journey from John F Kennedy Airport to Los Angeles Airport? The flight takes five hours and you leave at 12:00 PM EST. 6. How much money could Continental Airlines by giving customers 1/2 a can instead of a whole can of Sprite?	1. Estimate the size of diaper market in India. 2. Estimate the number of traffic lights in Patiala 3. How many kilograms of chicken is sold in your city per annum? 4. How many cricket bats are produced in India annually? 5. What is the number of mobile phones used in the world? 6. What is the market size of blue car in your native city? 7. How many cricket balls will fit into a Boeing 747? 8. Estimate the revenues of any outlet of save Haldirams in 1 day. 9. Estimate the number of ATM machines in Delhi.

Previous Year Guesstimates - Contd.

Software Companies

1. How many beer bottles are in circulation in the U.S.? (IBM GBS)
2. In the United States, how many people are in the air right now? (IBM)
3. How many golf balls can fit into a school bus? (Google)
4. How many dogs are there in the United States? (Google)
5. What is the annual demand for sugar in the US? (Applied Predictive Technologies)
6. Can you guess the size of the grocery market in India? (Flipkart)

Other Companies

1. If I were to fill this room with pennies, how many pennies would fit in?
2. How many piercings are done in India annually? (Capgemini)
3. What is the size of the market for disposable diapers in China?
4. How many Delta Airlines planes will take off in the next hour in United States?
5. How many gas stations are in North America?
6. How to estimate ExxonMobil’s revenues from gasoline sales? (AlixPartners)
7. Estimate the annual revenue of Tropicana's orange juice division in the US?

FOCUS

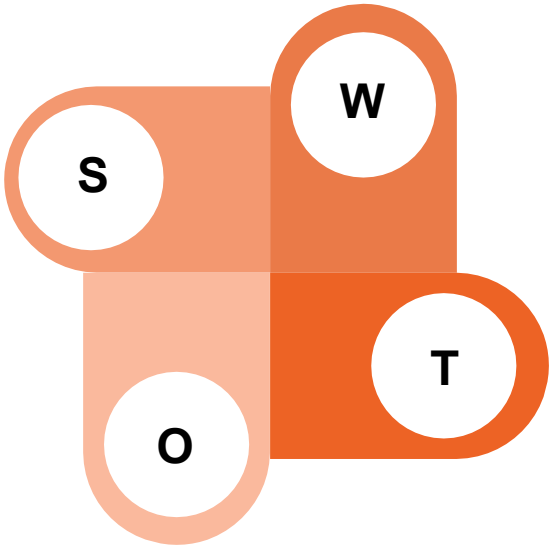
04

CONSULTING:
Consulting Tools



SWOT Analysis

SWOT is a great tool to capture the current state of your organisation; it helps set objectives for strategic planning and identify the firm's core competencies. It analyses whether or not to embark on a strategy.



Internal	STRENGTHS	WEAKNESSES
External	OPPORTUNITIES	THREATS
	Helpful	Harmful

Tips and Tricks (1/2)

Strengths	Weaknesses
1. What is our competitive advantage? 2. What resources do we have? 3. What products are performing well?	1. Where can we improve? 2. What products are underperforming? 3. Where are we lacking resources?
Opportunities	**Threats**
1. What technology can we use to improve operations? 2. Can we expand our core operations? 3. What new market segments can we explore?	1. What new regulations can threaten the operations? 2. What do our competitors do well? 3. What consumer trends threaten business?

Tips and Tricks (2/2)

Internal Factors	Weakness
1. Aspects of the company which it does/ does not do well at. 2. Qualities that make the company better/ worse than its competitors. 3. Assets/liabilities that the company holds (including financial, physical and human).	1. 1. Market scenario and the relevance of the product being sold . 2. Press/Media coverage of the brand. 3. Customer satisfaction/dissatisfaction 4. Changing trends and customer behaviour.

Practice Problem

Perform a SWOT Analysis on **Apple, Inc.** Refer to the solved example given on the next page.

SWOT Analysis Example - Apple

Strength

1. Apple has established itself as a trustable brand over time, and it's the world's most valuable company by market value.
2. Innovation: Apple keeps to deliver a set of products that provide a better UX.
3. Apple has always identified itself as a brand that focuses on Privacy as its key value.
4. Loyal Consumer Base.

Weakness

1. Expensive Product Range which acts a barrier for expanding in developing countries.
2. Closed-source components that prevents cross-device compatibility.
3. Apple Products are always known for blocking third-party repairs.



Opportunity

1. Expansion in AR, autonomous technologies.
2. Providing extensive streaming options for price sensitive markets.
3. Globalisation to reduce prices and escape higher tax regimes in countries.

Threat

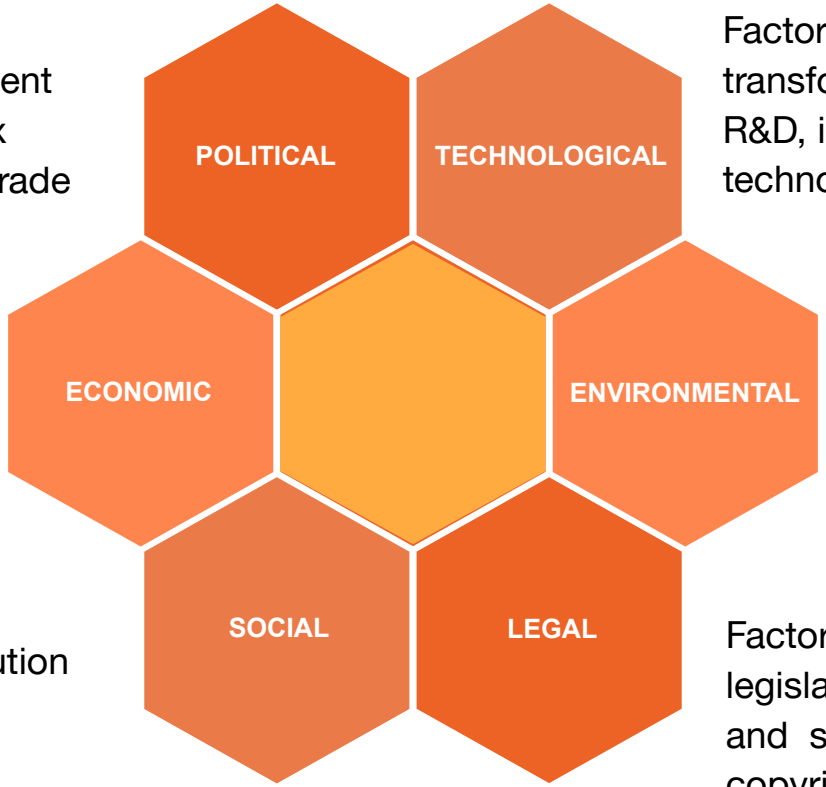
1. Governments requesting Apple to open backdoors.
2. A number of pending lawsuits- including that against its anti-competitive practices.
3. Apple, usually known for its late-moving policies, could experience disruptions in sales due to better technologies/ products being sold by its competitors.

PESTEL Analysis

Factors such as government policy, political stability, tax policy, labour law, foreign trade policy.

Factors like economic growth, interest rates, inflation, exchange rates, employment rates, and purchasing power.

Factorslike,demographics, culture,lifestyle,age,distribution and career attitudes.



Factors like digital transformation, automation, R&D, innovation, and technological awareness.

Include climate change, recycling procedures,carbon footprint,waste disposal,sustainability and net-zero targets.

Factors such as employment legislation, consumer laws, health and safety, trade regulations, copyrights and patents

What is PESTEL Analysis?

PESTEL Analysis is a tool that analyses the **external environment** of an organisation or business.

When is it best of use?

Ideal when ***entering a new market***

Useful while ***launching a new business***

Helps understand ***external risks and opportunities***

Why It Matters

Identifies ***EXTERNAL factors*** that influence the business but **cannot be controlled** by it.

These include aspects like ***climate, regulations, industry trends, inflation, political shifts,*** etc.

While the business can't change these factors, ***analyzing them provides strategic insight*** into the industry landscape

PESTEL Analysis

Political Factors

- Has to follow tax regulations and trade tariff.
- Have to consider political stability and relations between countries as to not to impact the supply of raw materials and semi finished goods carried .

Economic Factors

- Products are generally expensive.
- Have to consider import rates, exercise duty and exchange rates as they can spike up the cost of products.

Social Factors

- Increasing popularity of gaming and esports in major markets.
- Pandemic increased the demand for laptops.
- Decreasing popularity for the brand among younger age groups.

Technological Factors

- Technology is continuously developing, requiring companies to release products frequently.
- Strong competition from brands offering cheaper prices.

Environmental Factors

- Need to recycle e-waste properly
- Increasing pressure to adopt greener technology
- Leaving Carbon footprints is a concern

Legal Factors

Need to follow labor and employee safety laws.

DELL needs to follow taxation laws in the operating markets to avoid penalties.

PESTEL Analysis of Uber

POLITICAL

- Need to make its stand clear about driver's Insurance
- Have to follow minimum wage rules
- Have to deal with bans in many countries

ECONOMICAL

- Easily accessible
- Affordable fare charges
- Offers job opportunities, but pay may not be convincing

SOCIAL

- User friendly
- Quick pick up
- Gives better ride experience than taxis.

TECHNOLOGICAL

- Excellent mobile app for users
- Using social media and other electronic media well for promotion

ENVIRONMENTAL

- Fuel usage may increase
- Traffic congestion is a concern.

LEGAL

- Need to prevent ban in many countries.
- Need to follow labor and employee safety laws
- Copyright laws need to be looked at.

Porter's Five Forces

The Five Forces determine the competitive structure of an industry and its profitability. Industry structure and a company's relative position within the industry are the two fundamental drivers of profitability. The following steps have been proposed to comprehend the same:

1. Understand your industry of interest.
2. Identify attractive vs less attractive industries/markets.
3. Identify opportunities and risks.
4. How profits within an industry will be distributed
5. Extrapolate industry trends & anticipate changing trends.



Porter's Five Forces

Bargaining Power of Buyers
Price sensitivity, Number of Customers, Buyer's ability to substitute, Size of each customer order, Buyer's information, Difference between competitors.

Threat of Substitutes
Number of substitutes, Buyer propensity to substitute, Relative price performance o substitute, Perceived level of product differentiation, Switching costs

Bargaining Power of Suppliers
Number and size of suppliers, Uniqueness of each supplier's product, Ability to substitute

Rivalry among competitors
Industry growth, Quality differences, Brand loyalty, Barriers to exit, Number of competitors, Diversity of competitors

Threat of New Entrants
Switching costs, Barriers to entry, Capital requirements, Cumulative experience, Access to distribution channels.

Tips and Tricks

1. Use this model where there are at least three competitors in the market.
2. Consider the impact that government has or may have on the industry.
3. Consider the industry life cycle stage-earlier stages will be more turbulent.
4. Consider the dynamic/changing characteristics of the industry.
5. Avoid using this model for an individual firm; it is designed for use on an industry basis.

BCG Growth-Share Matrix

- **Stars** (High Market Share, High Market Growth):
  These are top-performing products in high-growth markets. They generate strong revenue and have high potential, making them ideal for continued investment.
- **Cash Cows** (High Market Share, Low Market Growth):
  These products dominate mature markets. They require minimal investment but yield steady, high profits—ideal for funding other ventures.
- **Question Marks** (Low Market Share, High Market Growth):
  These products operate in growing markets but haven’t yet captured a strong position. With the right improvements and investment, they could become future Stars.
- **Dogs** (Low Market Share, Low Market Growth):
  These underperforming products exist in shrinking markets. Since they offer limited return potential, it's often best to divest or discontinue them.

Solved Example 1 — Growth-Share Matrix of Air India

The airline industry is booming. However, all services of Air India except its cargo flight service have shrunk to a minor portion of the market. Hence, the cargo services of Air India can be considered a star, while others will fall under question marks.

Solved Example 2 — Growth-Share Matrix of Amul

Stars: Amul Ice creams, Lassi and Amul Kool are well known and have a growing market.

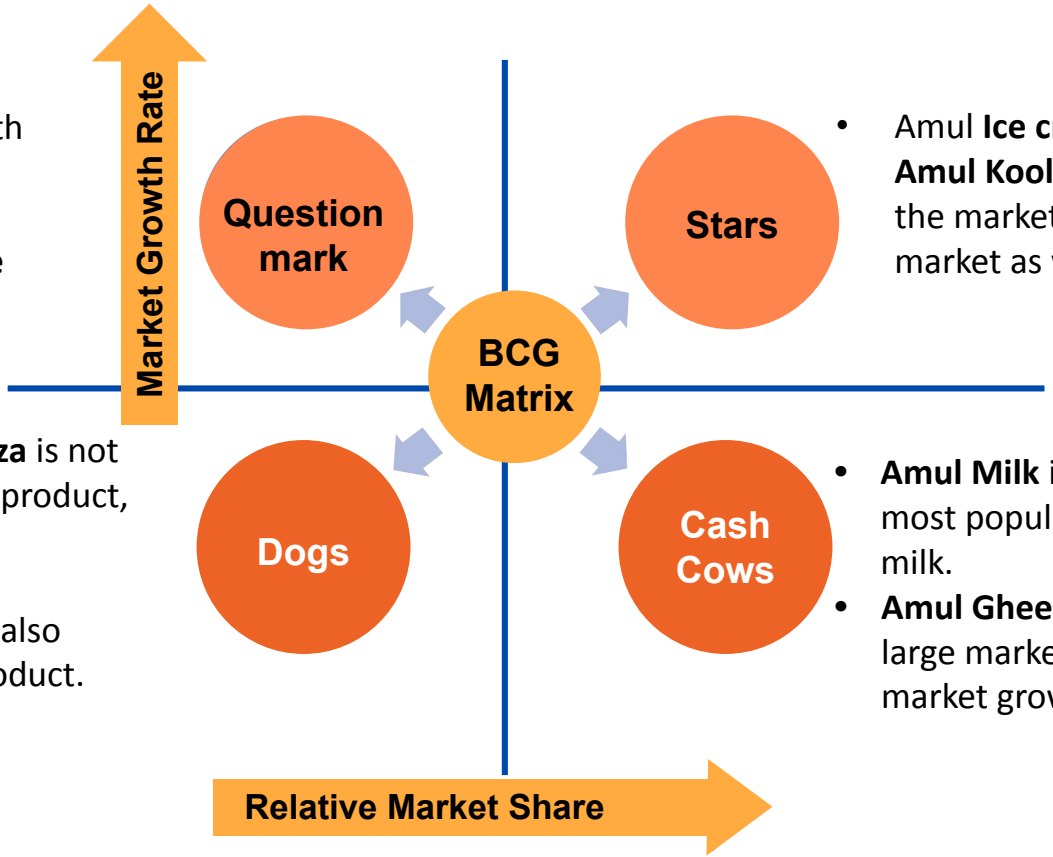
Cash Cows: Amul Milk is one of the most popular milk brands. Amul Ghee and Butter have significant market share but low market growth.

Question Marks: Amul bread is a new product with a huge possible market. Amul Pizza Base also has a growing demand.

Dogs: Amul Frozen Pizza is not a much-desired product due to better alternatives. Amul Cookies is also another such product.

BCG Matrix on Amul

- **Amul bread** is a new product with a huge possible market.
- **Amul Pizza Base** also has a growing market

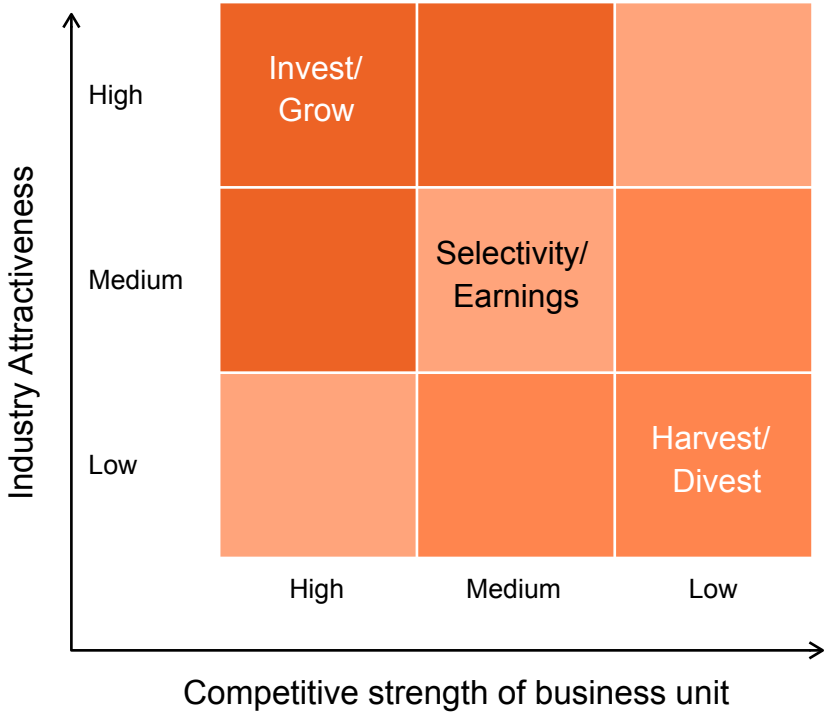


- Amul **Ice creams, Lassi** and **Amul Kool** are well known in the market and have growing market as well.

- **Amul Frozen Pizza** is not a much- desired product, due to better alternatives.
- **Amul Cookies** is also another such product.

- **Amul Milk** is one of the most popular brands of milk.
- **Amul Ghee** and **Butter** have large market share but low market growth.

GE-McKinsey Nine-Box Matrix



The GE-McKinsey nine-box matrix is a strategic tool used by multi-business corporations to prioritise investments among their business units. McKinsey developed it in the early 1970s, and it is conceptually similar to BCG's Growth-Share Matrix — but more complicated.

The business units (BUs) are evaluated and assigned a position on the matrix following a quantitative assessment of —

- the ***industry attractiveness***, and
- the ***competitive strength*** of the business unit.

GE-McKinsey Nine-Box Matrix

Industry Attractiveness

Industry attractiveness measures the ease of doing business in a market and generating profits in the long run. It is determined by factors such as:

1. Market growth rate (both short-term and long-term)
2. Size of the industry
3. Industry Structure (using Structure-Conduct-Performance Model)
4. Industry Profitability (using Porter's Five Forces Analysis)
5. Macro-environmental factors (using PESTEL)

Competitive Strength of Business Unit

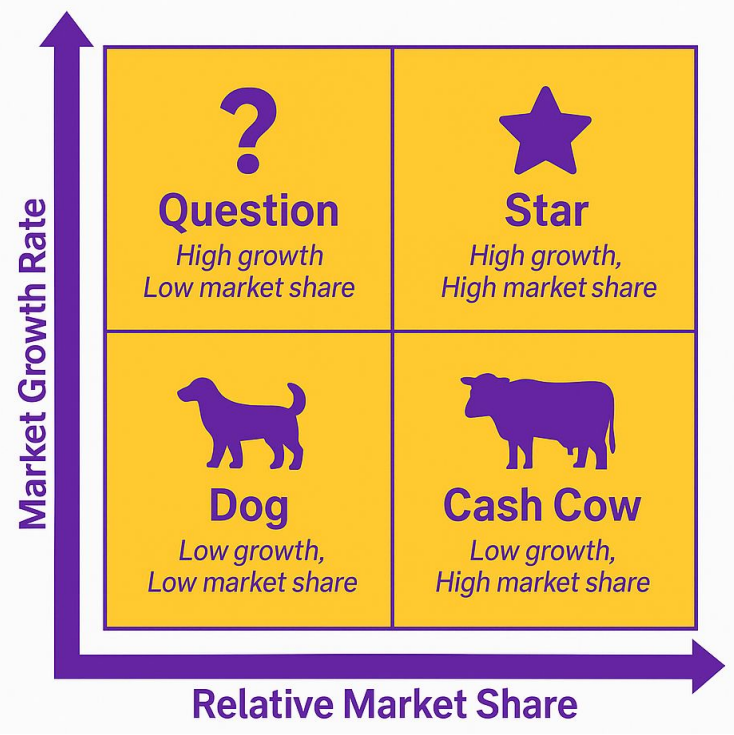
Factors used to determine business unit strength or sustainable competitive advantage include:

1. Market share and growth in market share
2. Brand Equity
3. Production capacity, Resources, Capabilities (using VRIO)
4. Value Chain Strength (using Value Chain Analysis and Benchmarking)

BCG Growth-Share Matrix

The BCG Growth-Share Matrix is a portfolio management framework developed by the Boston Consulting Group (BCG) to assist a company and show where to invest its capital to benefit the most.

The four quadrant matrix has two axes — Growth and Relative Market Share. In simpler terms, growth is the increase or decrease of demand in the market, and relative market share is how well the product is doing in the market compared to the highest competitor. Now, high or low Growth/Relative Market Share can be classified into four categories: Star, Question Mark, Cash Cow, and Pet/Dog.



GE-McKinsey Nine-Box Matrix

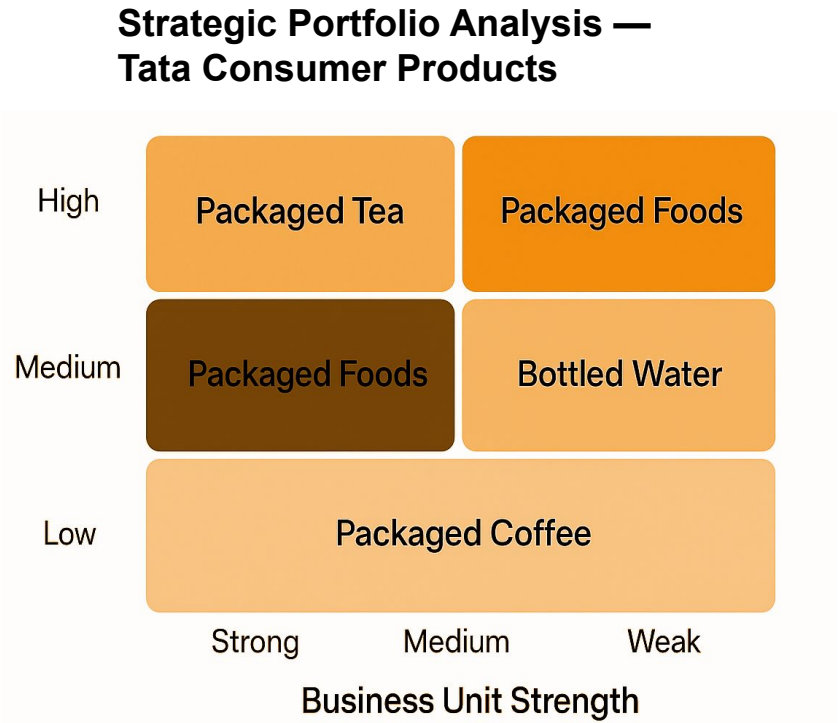
After assigning the weight (indicating importance) to each factor, the industry attractiveness score and business unit strength score are calculated using the formula given below:

$$\text{Score} = \sum_{i=1}^{N} \text{factor value}_{\,i} \times \text{factor weight}_{\,i}$$

The scores are then categorised as high, medium or low, and the business unit is plotted on the matrix. Recommendations can be made to grow, hold, or harvest a unit as discussed below:

Invest/ Grow	Selectivity/ Earnings	Harvest/ Divest
Applicable to BUs operating in a moderate to highly attractive industry with a medium to a high competitive advantage.	These BUs have low to moderate competitive position in an attractive industry or very high competitive position in a less attractive industry.	Applicable to BUs having low competitive advantage, operating in an unattractive industry or a combination of both.
Provide as much resources as the business unit needs regardless of whether it can generate itself or not.	Invest only if there is money left over investments in invest/grow business units or if they can generate cash in future.	Invest just enough to keep the units operational or divest the units by selling it to a prospective buyer.

GE-McKinsey Nine-Box Matrix



This matrix visualizes the strategic position of Tata Consumer's core business units based on two dimensions:

- **Industry Attractiveness**: Evaluated using factors like market growth, competition, and profitability.
- **Business Unit Strength**: Assessed through market share, brand equity, and operational capabilities.

Key Takeaways:

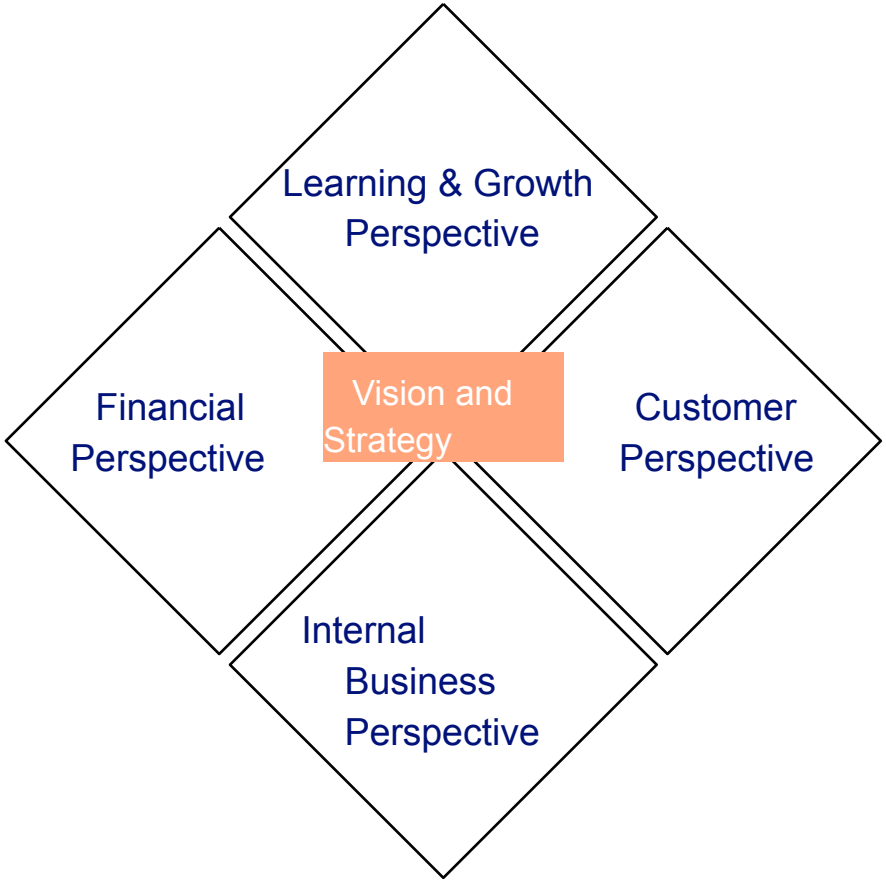
- **Packaged Tea** (e.g., Tata Tea, Tetley) is a clear leader and should be prioritized for investment and global expansion.
- **Packaged Foods** (e.g., Tata Salt, Sampann) operates in a moderately attractive market with moderate strength; selective investment and innovation are recommended.
- **Bottled Water** (e.g., Himalayan) has niche appeal and medium potential; brand strengthening and cost optimization are key.
- **Packaged Coffee** struggles in both dimensions and may warrant divestment or a focused turnaround strategy.

Balanced Scorecard

A balanced scorecard identifies and improves various internal business functions and their resulting external outcomes.

The balanced scorecard puts vision and strategy at the centre and isolates four aspects of an organisation for analysing — **Learning and Growth, Finance, Internal Processes, and Customer**.

Vital information is gathered from four business areas — strategic **objectives, KPIs, targets and initiatives/practices**.



Four Perspectives of a Balanced Scorecard — Explained

1. Learning and Growth Perspective	Determines the company's ability to improve and adapt to change
2. Financial Perspective	Analyses the company's ability to pay dividends, manage risks involved and better appeal to shareholders.
3. Customer Perspective	Determines the level of customer satisfaction with the company's products and services and how it provides value to its customers
4. Internal Business Perspective	Evaluates the company's products and services and how well it runs.

Some books also mention the **Learning and Growth Perspective** as the **Organisational Capacity Perspective**.

The third-generation balanced scorecard features a strategic linkage model connecting the dots between the objectives and a vision statement representing the result of implementing the initiatives listed in the scorecard. **[click here]**

Solved Example — Balanced Scorecard of JCPenney

Note that the targets mentioned here are representative in nature. Actual values may vary.

Objectives		KPIs	Targets	Initiatives
1. Learning and Growth Perspective	1. Optimise Supply Chain 2. Reduce Employee Turnover Rate 3. Improve Employee Skills	1. Supply Chain Efficiency Index 2. Turnover Rate 3. Training Index	1. Achieve 95% efficiency 2. Reduce by 5% 3. Achieve 85% efficiency	1. Sign long-term agreements 2. Increase job rotation 3. Start upskilling programmes
2. Financial Perspective	1. Increase Return on Investment 2. Reduce Operating Costs 3. Increase Revenue	1. ROI 2. Operating Costs 3. Revenue	1. Increase by 8% over two years 2. Reduce by 3% per year 3. Increase by 15% per year	1. Bulk purchases to achieve EoS 2. New billing system 3. Competitive Pricing Options

Solved Example — Balanced Scorecard of JCPenney

	Objectives	KPIs	Targets	Initiatives
3. Customer Perspective	1. Improve Customer Satisfaction 2. Reduce Waiting Times 3. Increase Customer Size	1. CSAT Score 2. Average Waiting Time 3. New Visitors (Monthly)	1. Achieve 85% CSAT Score 2. Reduce by 15% 3. Increase by 6% every quarter	1. Provide more choices at stores 2. Improved billing system 3. Offer incentives and coupons
4. Internal Business Perspective	1. Improve Cost Control 2. Reduce Process Cycle Times 3. Ensure supply-demand balance	1. Cost Efficiency Index 2. Average Cycle Time 3. Fill rate	1. Achieve 92% efficiency 2. Reduce by 15% 3. Achieve 98% efficiency	1. Buy raw materials in bulk 2. Increase automation 3. Use AI-enabled WMS and IMS

Value Chain Analysis

What is a value chain?

A value chain refers to all the business activities a firm undertakes to create a product from start to finish (e.g., design, production, distribution, and so on).

What do you mean by a value chain analysis?

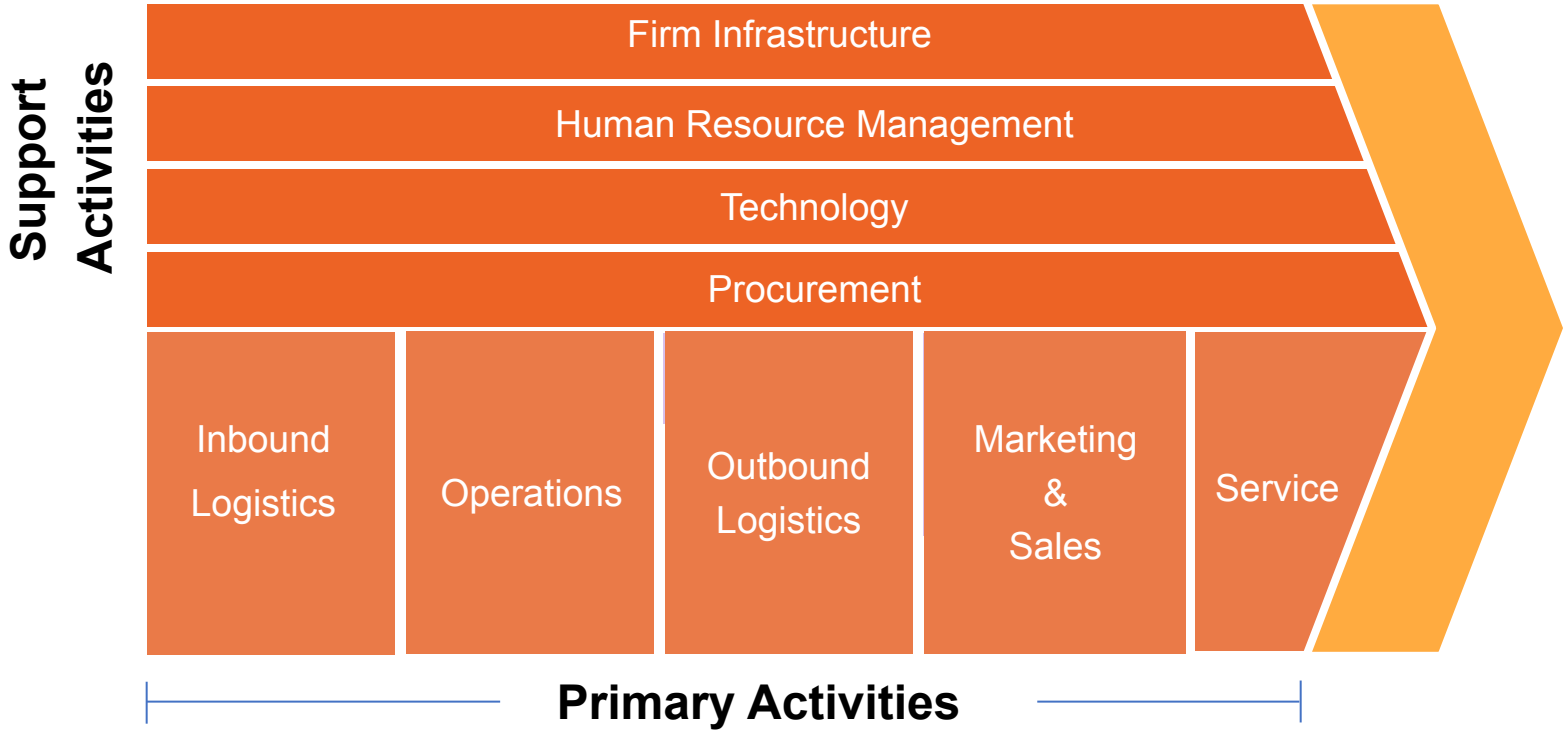
Value chain analysis is a strategic process where a firm evaluates its internal activities to identify how each contributes to the firm's competitive, cost and differentiation advantage. Its primary objective is to give businesses a visual model of these activities, allowing them to determine where they can reduce costs.

Porter's model is most commonly used for this purpose, considering several primary and support activities. The overall process is completed in five steps.

We'll now give you an overview of Porter's Value Chain Model.

Value Chain Analysis

Overview — Porter's Value Chain



Value Chain Analysis

Primary Activities vs Support Activities

Primary Activities

Inbound Logistics

Getting raw materials from suppliers

Operations

Manufacturing of the product

Outbound Logistics

Involves storage and movement of products to the end user

Marketing & Sales

Strategies to enhance visibility of product among the target audience

Service

Customer support for the product, warranty etc.

Support Activities

Firm Infrastructure

The management, financial and legal systems incorporated in the business

Human Resource Management

Managing employees and fresh recruitments

Technology

Innovation through R&D to enhance efficiency and profit

Procurement

Sourcing from suppliers for the best quality possible

Steps of Value Chain Analysis

1. Determining the business' primary and support activities	The primary and support activities include each action required in developing a product or service, from raw material to final product.
2. Analysing the value and cost of the activities	This involves identifying how each activity provides value to customers and the business as a whole, including checking for the competitive advantage (based on cost leadership or differentiation).
3. Referring to competitors' value chains	Use relevant metrics to compare your company to competitors. This practice is called competitive benchmarking.
4. Understanding a firm's customer base's perception of value	Conducting customer surveys, digging into any qualitative or quantitative data that can be pieced together, etc., to understand what its target market thinks of it.
5. Identifying opportunities to gain a competitive advantage	Undertaking minor changes at the beginning that provides high-impact results. After the easy wins are identified and actioned, the more significant challenges hindering efficiency must be tacked.

Solved Example 1 — Value Chain Analysis of Samsung

Primary Activities

Inbound Logistics

579 suppliers globally. Owns several logistics firms as its subsidiaries.

Operations

Corporate business model. In-house manufacturing.

Outbound Logistics

Integrated logistics subsidiaries, SELC & Samsung SDC transport and distribute finished products.

Marketing & Sales

Print and media advertising, public relations programs. 53 international sales bases.

Service

Regular customer satisfaction surveys, Samsung Direct Support.

Support Activities

Firm Infrastructure

Well executed financial support system and corporate governance.

Human Resource Management

Regular training and critical resources to employees. Competitive salaries and added benefits

Technology

Heavy investment in R&D (more than 20 trillion KRW)

Procurement

2400+ suppliers globally. International Procurement Centre manages supplier relationships

Solved Example 2 — Value Chain Analysis of Nike

Primary Activities

Inbound Logistics

Supply chain based on outsourcing and diversification.

Operations

600 factories and 1048 retail stores worldwide.

Outbound Logistics

Selling products to wholesalers, D2C sales through retail outlets and own website.

Marketing & Sales

Print and media advertising, promo events and promotion through celebrities.

Service

Excellent pre-sale and after sales services, special store hours, store-to-door services.

Support Activities

Firm Infrastructure

High focus on quality management, efficient legal matter handling, state-of-the-art offices, procurement facilities and distribution outlets.

Human Resource Management

Promotes a work culture based on diversity and inclusion.

Technology

Integration of AR technology into stores, Reuse of wastes generated in Nike factories.

Procurement

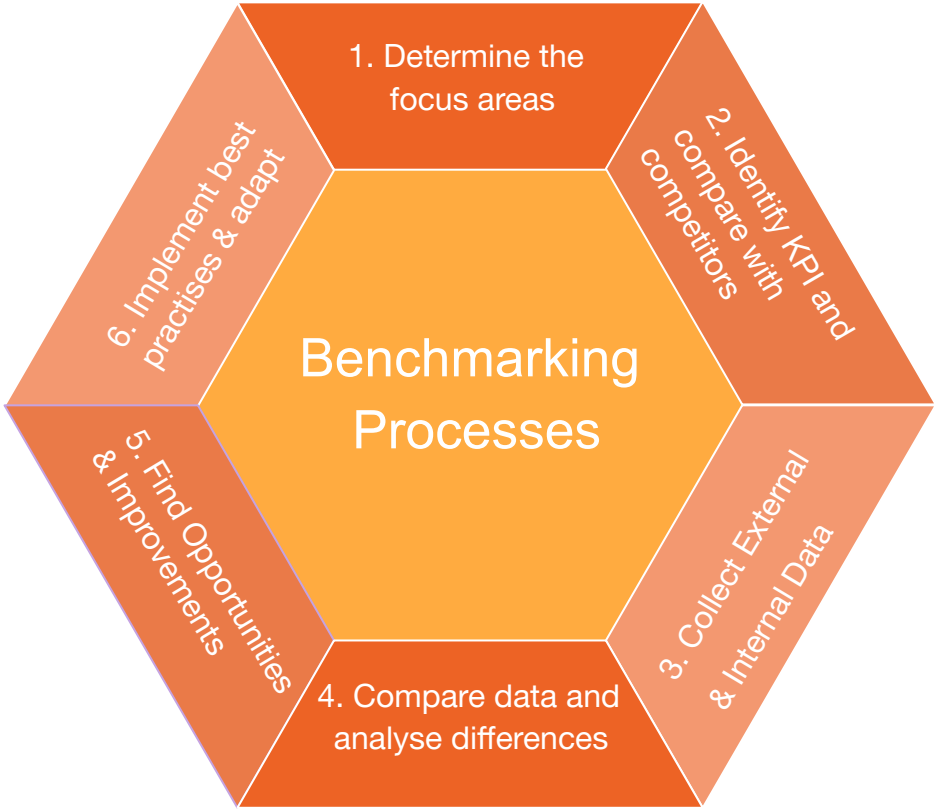
A global procurement team manages the procurement process to ensure supply of high quality raw materials.

Benchmarking

Benchmarking involves comparing a company's metrics with that of industry competitors (often industry leaders) or to those of innovative companies in related industries.

This strategic tool aims to identify the industry's best practices and incorporate similar activities into other companies.

In some cases, comparisons can be made internally between operations of businesses owned by the same firm.



Benchmarking Processes — Explained

1. Determining the focus areas	In the first step, the problem areas are identified. Usually, these are business operations with scope for improvement.
2. Identifying KPIs and competitors to compare data	The next step is to decide on the performance metrics for benchmarking. Companies operating within the industry or internal areas with better performance rates are chosen to compare data.
3. Collecting external and internal data	Now, data is collected on performance and company practices. In most cases, the data collected includes revenues, profit margins, production costs, process cycle times, etc.
4. Comparing data and analysing differences	The two sets of metrics are compared side-by-side and the gaps are identified. The relevance of these differences is analysed.
5. Finding opportunities of improvement	The competitor's best practices are identified, and an action plan is formulated.
6. Implement best practices and adapt	Finally, companies can incorporate these best practices after tweaking them to their needs. Benchmarking should be repeated at regular intervals to track metrics changes or adapt to other changes.

Solved Example on Benchmarking — Logistics Sector

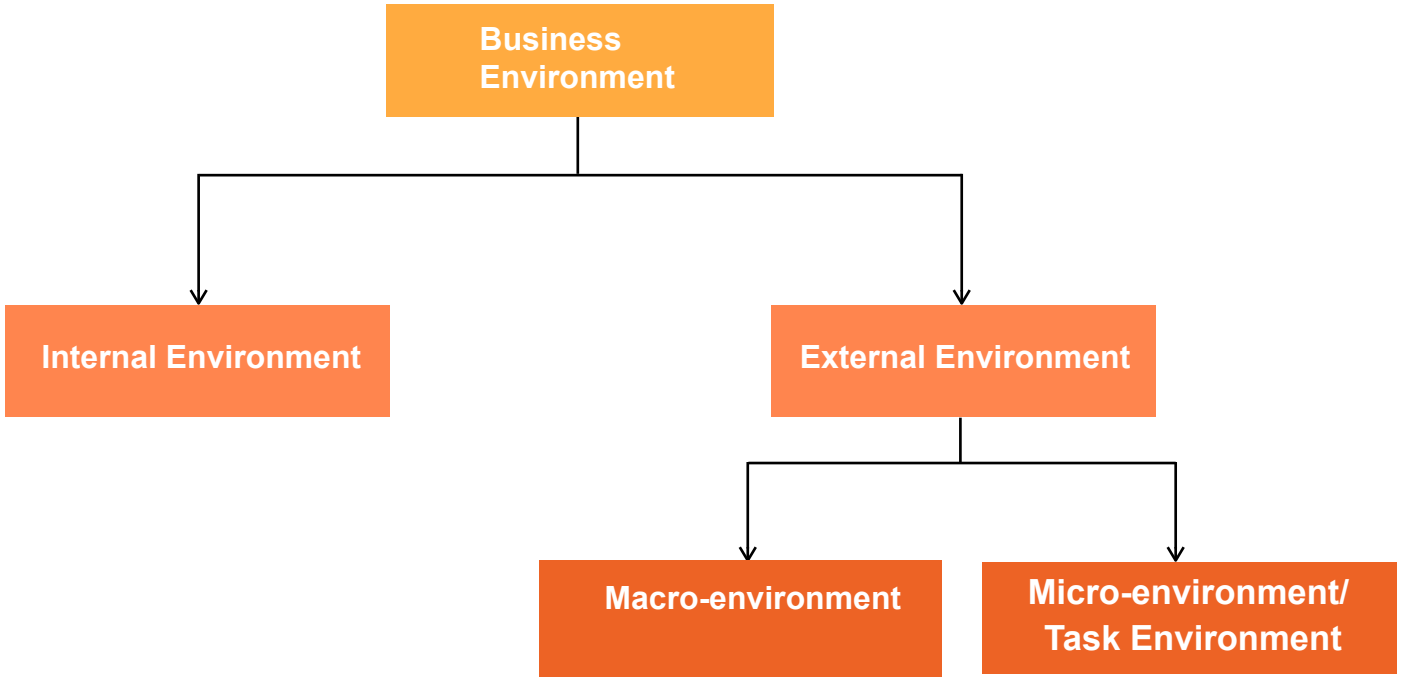
Problem Statement

Attoship is a global logistics company that provides package delivery services. The company saw flattening revenue growth during H2-2021, while the worldwide market experienced growth of 12% during the same time. Recommend a strategy to increase revenues by 15% in a year.

Approach

1. The primary step is to find all **operations/processes that add value to the supply chain** (warehousing, transportation, distribution, value-added services, packaging).
2. The next step is to identify the performance metrics — in this case, **on-time performance, package costs, employee turnover rate, customer satisfaction rate and storage cost per product shipped**.
3. A competitor, preferably the market leader (e.g. DHL), can now be chosen to compare data.
4. After relevant data is collected, the **differences can be analysed,** and an **action plan can be suggested**.

Task Environment



Task Environment

Business Environment: The aggregate of all the forces, factors and institutions which are external to and beyond the control of an individual business enterprise but significantly influence the enterprise's functioning and growth.

Internal Environment: Several elements inside an organisation act as strengths and may also become a reason for weaknesses of the organisation. These internal factors have an impact on organisational decisions.

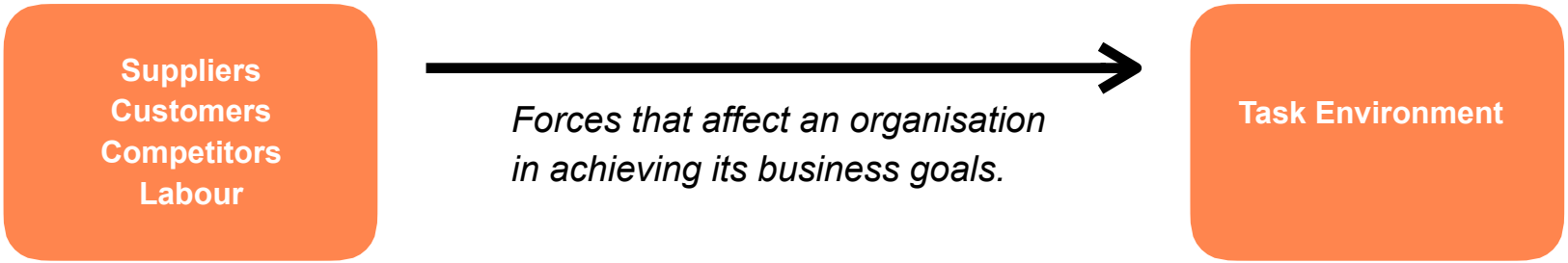
External Environment: This includes factors existing outside the business enterprise that offer opportunities and create threats for the business. These include factors such as economic, socio- cultural, legal, demographic etc. These factors are beyond the control of the company. There are two types of external environments — **macro** and **micro**.

The ***macro-environment*** refers to the broader condition of an economy as opposed to specific markets. It consists of ***demographics, economic, natural, technological and political factors***.

Task Environment

The **micro-environment** includes many factors in the company's immediate environment and has a bearing on the company's performance. These can be **suppliers, customers, competitors and labour**.

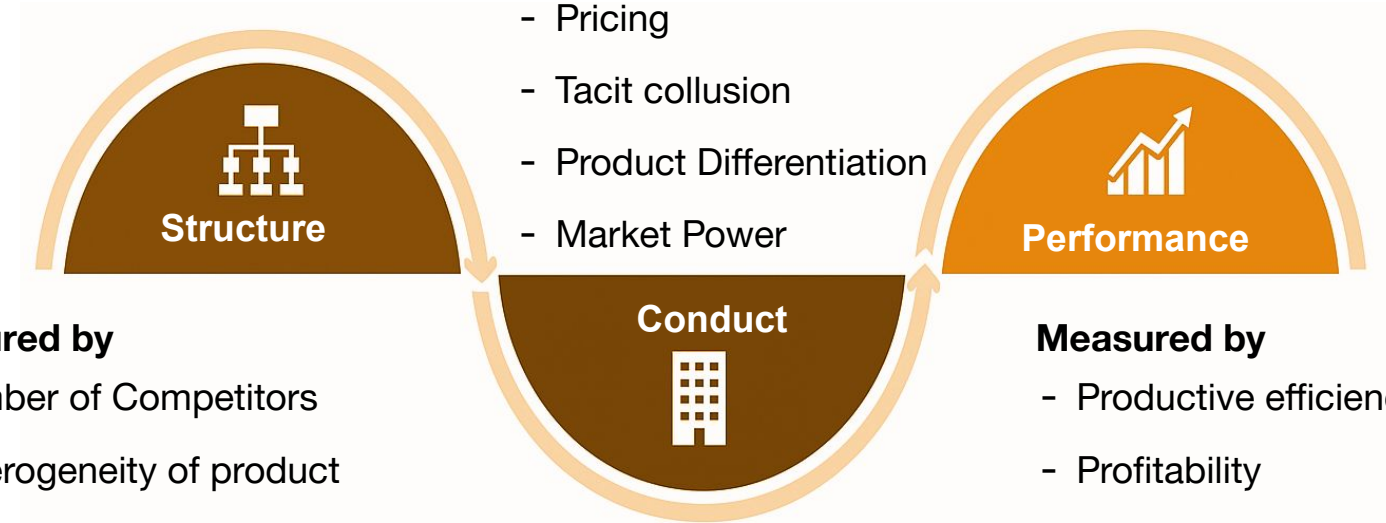
The **micro-environment** is also referred to as the **Task Environment**.



The task environment of an organisation is the environment which directly affects the organisation in attaining business goals. These can be analysed better using Porter's 5 Forces Analysis.

SCP Paradigm

The structure-conduct-performance (SCP)paradigm states that market structure would determine firm conduct, which, in turn, would determine performance.



Measured by

- Number of Competitors
- Heterogeneity of product
- Supply-Demand Match
- Cost of Entry and Exit

Measured by

- Pricing
- Tacit collusion
- Product Differentiation
- Market Power

Measured by

- Productive efficiency
- Profitability
- Product Quality
- Resource Allocation

SCP Paradigm

Use Cases

The structure-conduct-performance (SCP) paradigm can be used in the following scenarios:

- To analyse a non-changing industry.
- To justify consolidation in an industry.
- To investigate the effects of an external shock on one sector (like talent drain, shifting global demand, pandemics, data attacks and changing consumer demand).
- It studies whether structure drives performance and also influences conduct.
- To analyse the effects of a more attractive industry structure on the industry's performance.

The SCP paradigm cannot be used alone to generate relevant insights and is often used with other consulting tools.

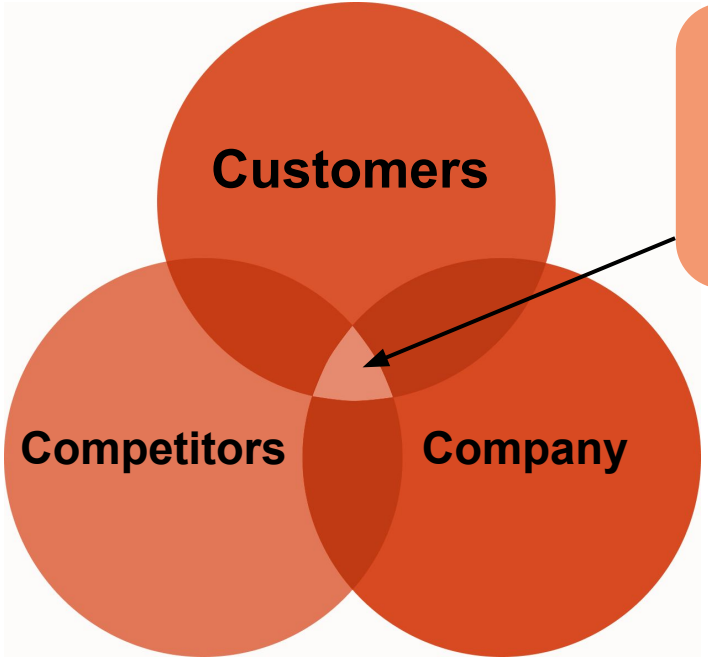
Practice Problem

Analyse the impact of adverse weather and natural disasters at the workplace on the performance of Texas Instruments.

3Cs Model

Customers

- Who are the key customers?
- What do they value?
- Why will they choose our product?
- What do they think of our product?



Strategy (A SWEET POT)
Strategy is the balance between what we offer, what the customers want and the threats from our competition.

Competitors

- What are they offering that we can do better?
- How are they making money/profits?
- What are their strategies and weaknesses?
- What should we do to beat them?

Company

- What are our strengths and weaknesses?
- What cost advantages or operational strengths exist?
- What is our unique selling point (USP)?

3Cs Model

Customer Analysis

1. Customers are the most crucial of the three elements since a firm cannot exist without them, or there would be no opportunity to achieve a competitive edge. The best strategies are often found by putting oneself in the customer's shoes.
2. The best approach to determine how to appeal to your target market is to conduct thorough consumer research.
3. Demographic information is crucial to this research. This information can be found through surveys and interviews with the target market.
4. Understanding your target market and what they want can help you develop marketing techniques more likely to succeed once implemented.

Competitor Analysis

1. After analysing the customers, a company needs to understand how it compares to its rivals.
2. The first step would be finding out the main competitors in the industry.
3. Then one should find their business model, revenue streams, go-to marketing strategy and USP.

4. One can do this by analysing their website using SEO tools, reading their newsletters and reports, visiting their stores or online payment check-out and contacting them as a potential customer.
5. Also, you should find out the brand image and penetration of the competitors.

Company Analysis

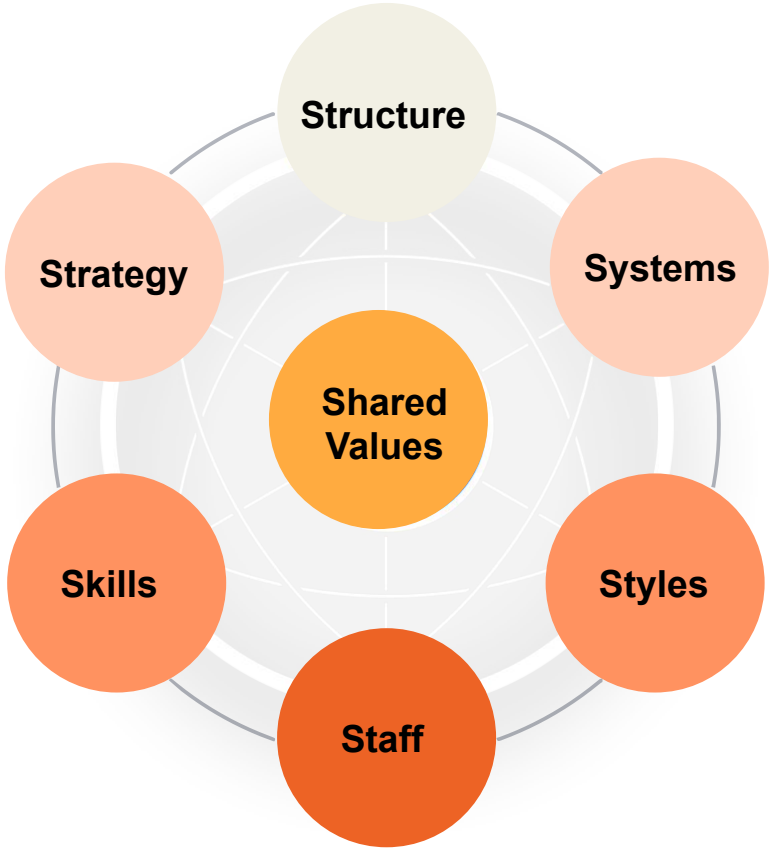
1. Last but not least, one should analyse the client company.
2. You must focus on cost structure and cost-effectiveness in production and marketing. These are key to ensuring more cash in your hands.
3. You should find out what marketing tactics have previously been successful for them and what hasn't. It would help if you approached this, once more, from the client's perspective.
4. We must now analyse our brand image and presence, compare it with the competitors, and see ways to gain an advantage.

McKinsey 7S Model

The McKinsey 7S Framework is a management model that identifies seven internal elements of an organisation that must be aligned for success.

It is used to assess performance following a merger or other restructuring to identify areas that need improvement. The model suggests that the risk of resistance to change can be significantly reduced by incorporating all seven elements from the start.

This model ensures that an organisation's culture is not negatively affected by implementing necessary changes.



Solved Example — 3Cs Model of Burger King



Customer

1. The average Burger King customer is a middle-high income male in the age bracket of 15-40 years.
2. Its unique and innovative marketing strategies show its focus on customers.
3. It offers one of the most reasonably priced and largest variety of products which is loved by the people.



Competitors

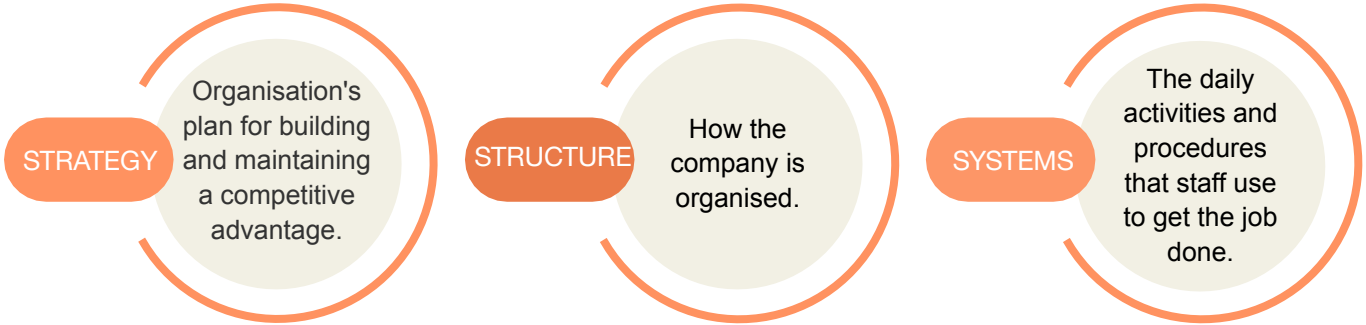
1. As McDonald's is the market leader, BK’s major strategy has been to attract McD’s customers over to them.
2. The one penny whooper played a very important role in this. They built their brand image as one which promotes fresh foods using the stale whooper campaign against the longevity of McDonald's burgers.



Company

1. Burger King's mission is to offer reasonably priced quality food served quickly in attractive, clean surroundings.
2. It's based on a franchisee model.
3. It focuses on a cost leadership and differentiation business model, thus catering to a more significant no. of people.

The Seven Interdependent Elements - Explained



Hard Elements



Soft Elements

How to use the McKinsey 7S Model?

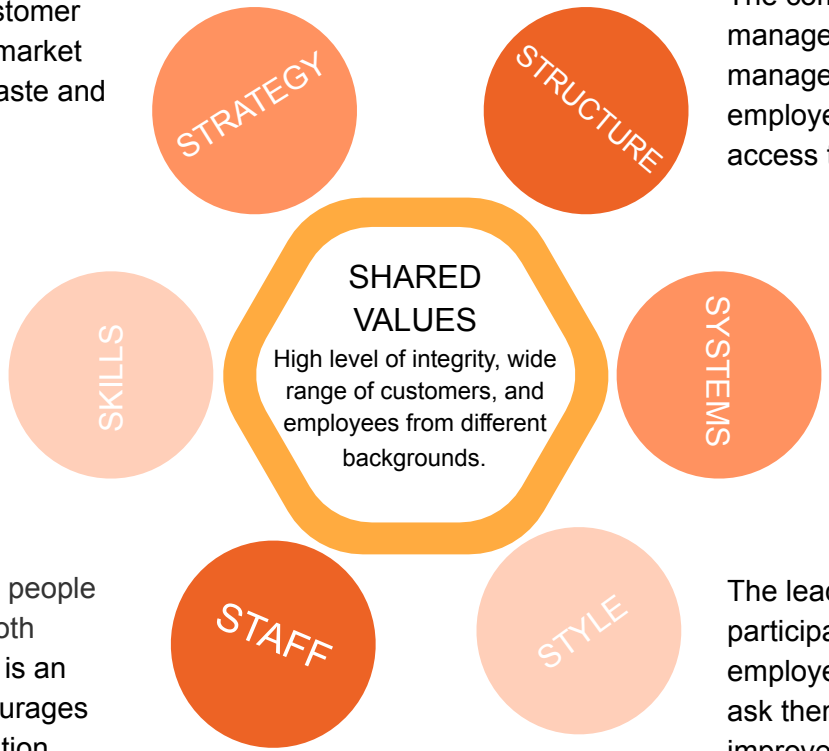
1	Start with your shared values: do they align with your organisational structure, strategy, and operations? If not, what can be changed?
2	Then look at the hard elements — strategy, structure, and systems. How well do they align with each other? Identify areas of improvement.
3	Next, consider the soft elements — shared values, skills, style, and staff. Are they consistent with the hard elements? Do they align with one another? If not, what can be changed?
4	You'll need to employ an iterative (often time-consuming) approach of making adjustments and then re-analysing how it influences other elements and their alignment.

Solved Example — McKinsey 7S Model of McDonald's

Cost leadership, High speed of customer services, aggressive international market expansion, the universality of the taste and cleanliness of restaurants.

Mcdonald's pays special attention to skills, training, and workshops are regularly held to ensure that staff is able to provide flawless service to customers. HR is one of its core competitive advantages.

McDonald’s employs over 200,000 people worldwide and the hiring is done both internally and externally. The team is an inclusive one that accepts, encourages diversity, and works in synchronisation.



The company has a flat structure and the manager of each outlet usually manages assistants and employees. All employees work as a team and have easy access to senior leadership if needed.

The systems include but are not limited to employee recruitment and selection, transaction processing, team development and orientation and customer relationship management system.

The leadership style at McDonald’s is participative. Seniors engage with employees from different levels and ask them to share their feedback to improve strategy and operations.

Overview — 4Ps of Marketing

To market a product successfully, the 4 Ps are essentially important. The 4 Ps are comprised of four core elements:

1. Product
2. Price
3. Place
4. Promotion

The 4Ps are often referred to as marketing mix. However, this tool got extended to the 7Ps with the insertion of People, Process and Physical evidence. The marketing mix strategy is not static. One must adjust and refine it based on customer needs and product growth.



Product

Marketers first have to analyse who needs the product and why. They have to define the quality of the product in all areas like ease of availability, attractiveness, effect on the environment, etc. Being first in introducing a category can unexpectedly boost sales. For example, Apple was the first company to launch a touch-screen smartphone which could play music, browse, and make phone calls. Its sales touched the two billion mark in the year 2021.

Price

Price is the amount the marketer sets up for the product and the customer is willing to pay. Marketers must set the price, keeping production costs and profit margin in mind. The price of a product must be accurate and perceive its value. The supply costs, seasonal discounts, competitor prices, and retail markup are the essential considerations for putting up the product's price. Marketers also have an option to increase the price and create artificial scarcity to increase the demand for their products. Giving appropriate discounts at the time needed also will draw customers, but it provides a look that the product is less desirable.

Place

The place is where the product is kept for sale. It may either be online or offline. The critical consideration for offline stores is to keep the product where it seems more people will buy. Selecting luxurious malls and stores will give the product an expensive and better quality look.

Promotion

Promotion is the way to communicate and advertise the product to customers. Giving ads on social media, banners and flexes on roadsides and malls will attract customers' eyes.

When to use 4Ps of Marketing?

This tool could be of use while launching a product in the market. At the time, analysing the existing development and sales will clearly show the areas of interest. This tool can optimise sales. Although it seems the 4Ps are mutually exclusive in their way, they overlap inevitably when the product sale begins to grow in the market.

Solved Example — Amazon

Product: Amazon started to keep various products ranging from electronics, clothing, home appliances & decoration, and accessories. It expanded its product range to Amazon Fresh for groceries, Kindle for reading e-books, Amazon Prime Video for movies, TV shows and web series, etc.

Price: Amazon gives an option to search for items below a specific price, providing ease to its customers. It also announces sales and highlights discount percentages on the product. Amazon Prime allows customers to take a subscription plan with multiple benefits of shopping. The options of pay later and auto-pay for subscriptions draws the customers.

Place: Amazon offers all its products and services globally. It operates massive warehouses at strategic locations all over the world to ensure timely delivery at even the remotest of places.

Promotion: Amazon uses aggressive marketing campaigns to promote its brand. It uses older searches and shows advertisements on other social media platforms linked with the same account.

Applying the Concepts

Start with clarifying questions. You could follow the *3Cs framework* to gain insights about customers, competitors, and the company.

You can understand the internal and external factors affecting the company using *SWOT* and *PESTEL* analysis. One can use *Porter's Five Forces Analysis* or *Potential Industry Earnings (PIE) Model* to determine the competitive structure of an industry and its profitability.

Similarly, *BCG Growth Matrix* or *GE-McKinsey Nine-Box Matrix* can be used to restructure business units or plot the product portfolio's competitive position.

Finally, while suggesting recommendations, the interviewee can evaluate the strategic fit of the solutions by analysing cost-feasibility and cost-benefit.

1. Potential Industry Earnings (PIE) (Miscellaneous)

This can be considered a modern version of the original Five Forces Model and is used to evaluate a firm's ability to enjoy its share of the industry profits.

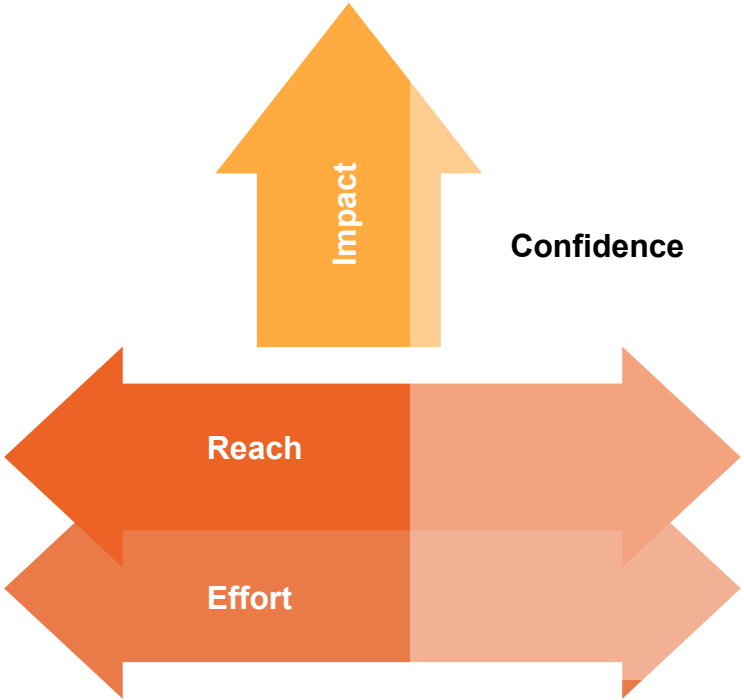
PIE = Total value added by the industry – Total cost to produce the goods

In addition to the original five elements, the PIE model considers the effect added by value disciplines, which explains how some companies can achieve and maintain market leadership despite being in competitive industries. The three value disciplines are:

1. Operational Excellence
2. Product Leadership
3. Customer Intimacy

By identifying what is most important for an industry and its customers, a strategist can make specific recommendations about the direction a company should go. Companies should adopt either of the value disciplines to enjoy success.

2. RICE Model (Miscellaneous)



RICE is a prioritisation model companies use to improve internal decision-making processes.

Each of the four factors is scored as depicted in the illustration on the left:

1. **Reach** is measured as the number of people/ events per time period.
2. The **impact** of an idea on an individual user is estimated.
3. The level of **confidence** in the estimates is factored in.
4. The **effort** is estimated as the work that one team member can do in a month.

3. VRIO Model (Miscellaneous)

VRIO stands for value, rarity, imitability, and organisation and is used to evaluate an organisation's resources and capabilities to determine its competitive advantage. You can use a decision tree to help map the outcomes of your probe, depending on whether you deem a resource as having met the criteria or not.

- ***The Question of Value:*** Can the firm exploit an opportunity or neutralise an external threat with the resource/capability?
- ***The Question of Rarity:*** Does the firm control scarce resources or capabilities or own something hard to find yet in demand?
- ***The Question of Imitability:*** Is it difficult to imitate, and will there be a significant cost disadvantage to a firm trying to obtain, develop, or duplicate the resource/capability?
- ***The Question of Organization:*** Is the firm organised, ready, and able to capitalise on resources and capabilities? Has the firm managed to capture value?

FOCUS

05

CONSULTING:
Consulting Frameworks



Profits vs Profitability

Profits are merely a difference between revenues and cost, while Profitability refers to profit as a proportion of sales.

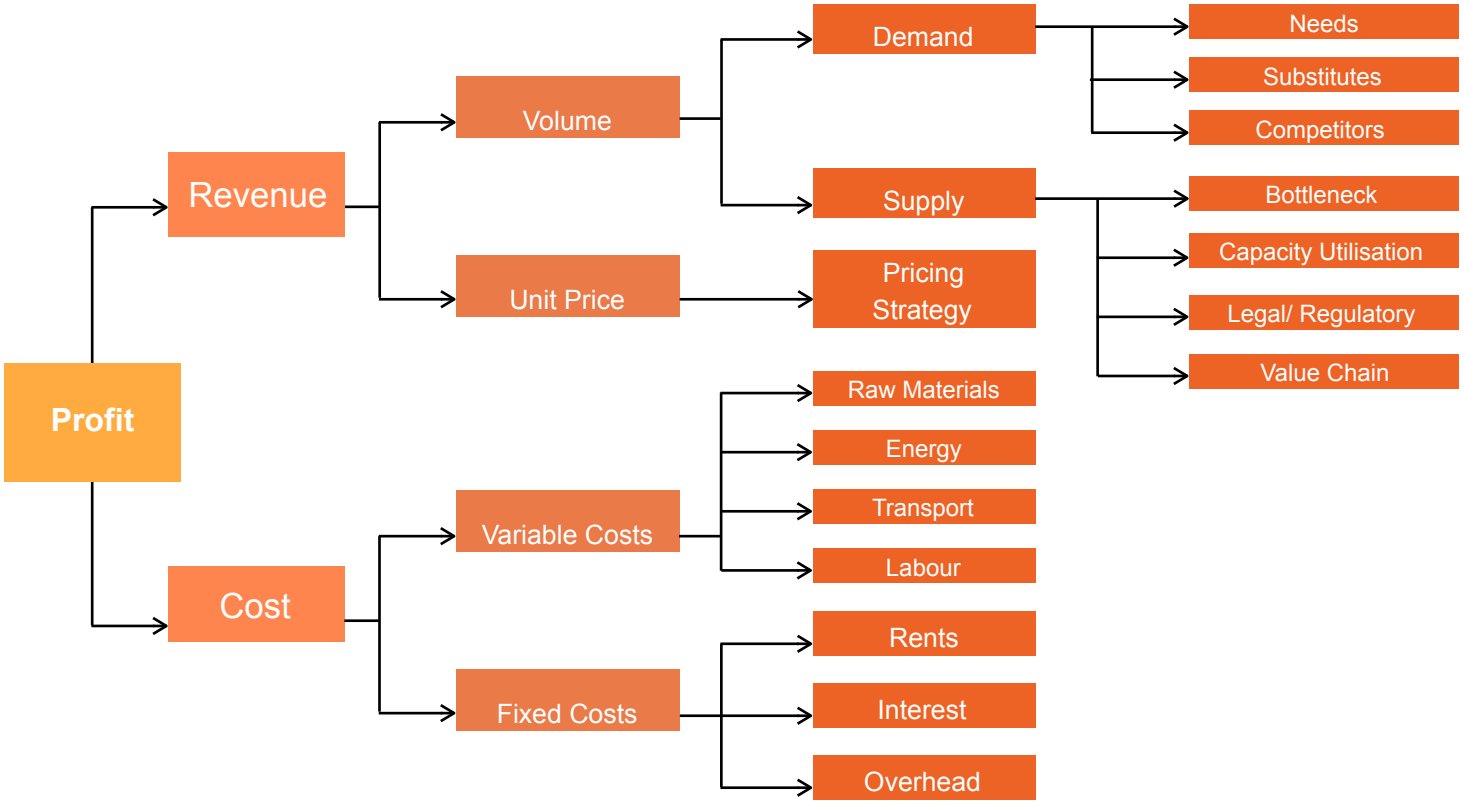
These cases are generally focused on drilling down to uncover the lever driving a change in the business. The 4 main steps in the process are given below:

1. Get comfortable with your levers
2. Drill down to the components
3. De-average & customise the components
4. Provide a recommendation

When to use the Profitability Framework?

- Growth Strategies
- Creative cost Benefit
- Declining Profits
- Weighing Scenarios

Framework Overview — Profitability



Problem Statement

Our client is a healthcare technology company that recently launched a hand-held device, the sales for which have not been picking up. Identify the reasons and suggest solutions.

Transcript

Candidate	Interviewer
Before I begin to analyse the case, I would like to ask a few clarifying questions. I would want to understand the client's business. What kind of hand-held device does the company sell?	This is a blood pressure measuring machine; it's a typical machine with a screen and a strap attached to it so that the strap is wrapped around the palm and the screen gives the reading.
How long have they been operating and where?	15 years; they have been operating in various Asian countries and is headquartered in New Delhi.
What customer segment are they targeting for the product, and what channel are being used?	They target customers aged 40 or above and use pharmacies and hospitals as channels.

Case Transcript — Healthcare Technology

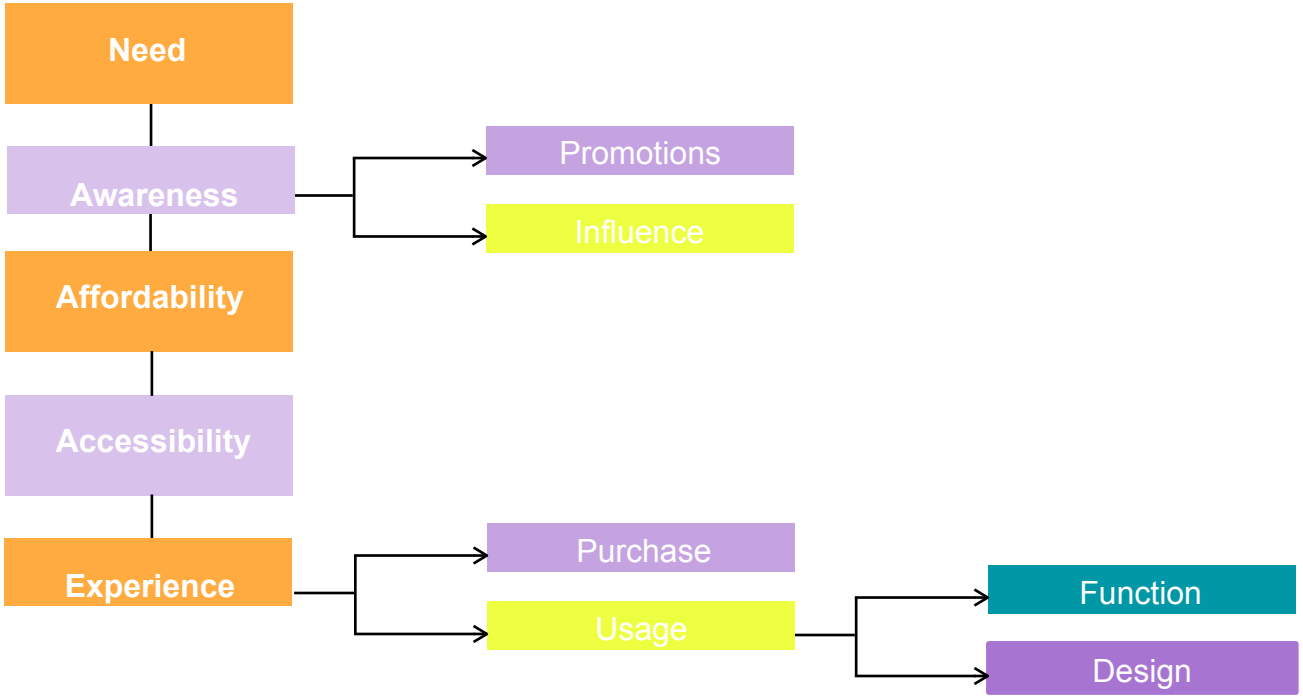
Candidate	Interviewer
Have both the channels not seen a rise?	No, pharmacies have somehow been meeting the targets, however hospital is the one channel that contributed to the biggest decline.
Could you elaborate a bit like in hospitals, do you mean the chemist shop in the hospital, or is there a direct contact?	So, whenever you go to a doctor, they recommend some machines, so by hospitals, we mean the doctors who recommend the devices. The chemist shops which are there in the hospitals are included in the pharmacy.
So those are not meeting the targets?	No, the doctor recommendations are not meeting the targets set.
Are there any supply-side issues that I need to look into, i.e., are the products reaching the market or is it a demand-side problem?	No, I think it is a demand-side problem. We have been approaching the product quite well, the supplies are there but the demand is not there.
So, on a broad level, I have laid down the approach. Is there any particular bucket that I should dive deep into?	Let's focus on awareness.

Candidate	Interviewer
So, I see a significant price difference there. In 3 years, the doctor would be getting Rs 100 per patient, and here the doctor is getting Rs 700 in a year. So this is a probable reason why the doctors are not pushing the product. There could be other reasons that could relate to the perception of your brand, however that won't be the case here. Another issue can be the product itself. Is that a possibility?	Let's dive into them.
With the product, that would bring us to the experience. There, I could talk about the user experience. So, broadly if I think about the product, there are two significant aspects - one is the product's function and the second is the design. In design, we will talk about its form as well. So, functionally is your product fool proof and provide an accurate measure of BP?	If the strap is bound correctly, it will be accurate. However, if it isn't then there will be certain inaccuracies.

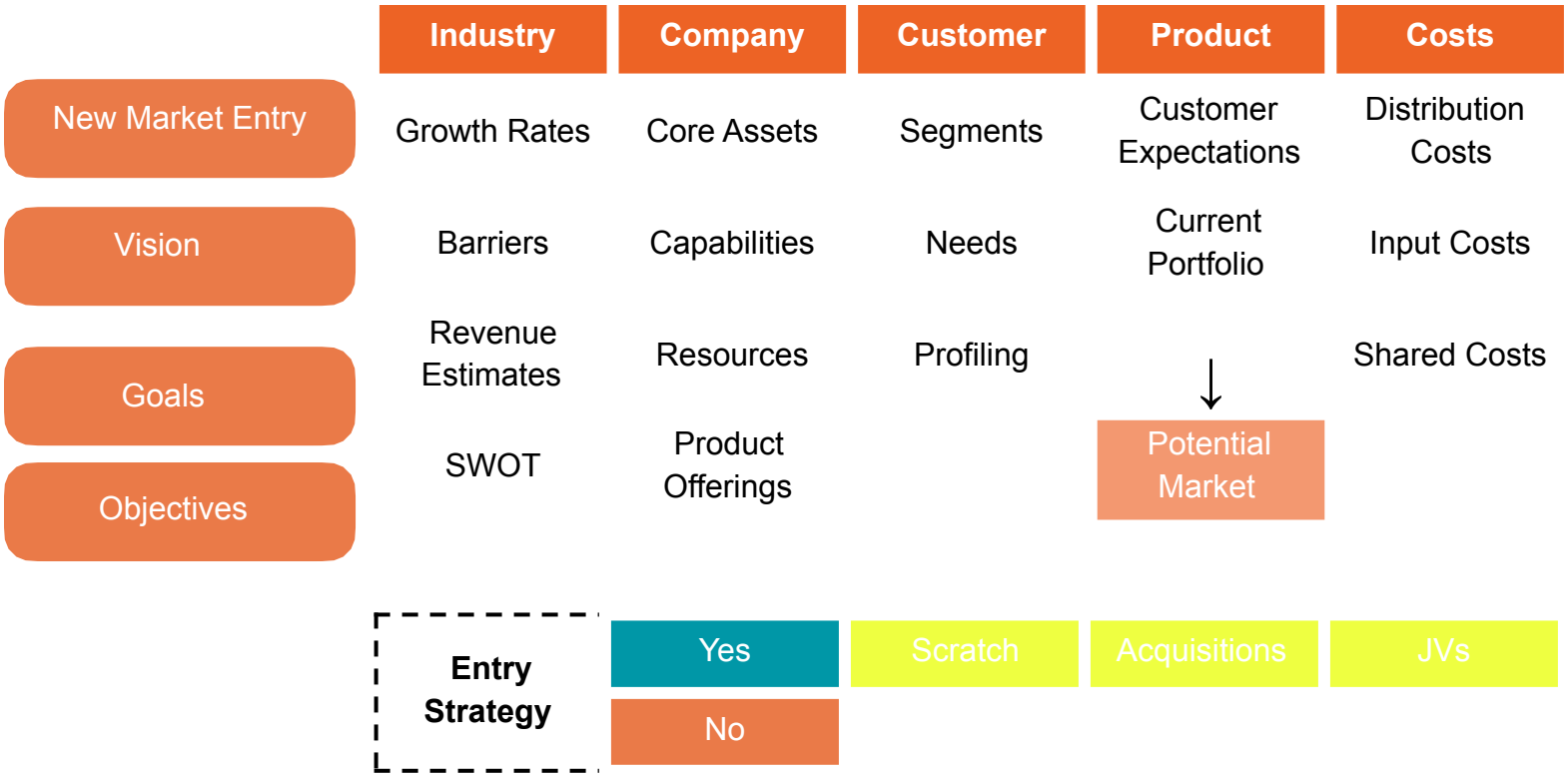
Case Transcript — Healthcare Technology

Candidate	Interviewer
You mentioned the strap which brings us to the design. So does the strap get attached every time properly or is there a possible glitch?	We have experienced certain complaints about the strap getting loose after 12 months of usage and hence the readings turn inaccurate.
So we have an issue with the product design as well?	We might have.
So far, I identified two problems — one caused by concerns among doctors and the second caused by its design.	Good. Now, how would you propose solving these issues?
To solve the influencer issue, we can actually increase the commission for doctors; issues with design can be solved by improving the quality of straps of device so that get tied properly and easily.	Great! Let's conclude the case here.

Structure


Need
Awareness
Promotions
Influence
Affordability
Accessibility
Experience
Purchase
Usage
Function
Design

Framework Overview — Market Entry



Framework — Market Entry

The magnitude and importance of entry decisions — encompass everything from geographic expansion to new products to diversification efforts — calling for a detailed analysis. Before one firm can enter a market, it is essential to estimate the economic and operational feasibility of the business in that market.

To remove any bias from our analysis of entry decisions, we should target the five core issues — value proposition and capabilities, the market size, competition, market share and revenue, and costs involved.

Key Questions to ask yourselves

1. What is my distinctive value proposition, and do I have the necessary capabilities and competencies (including talent and experience in other related markets) to succeed?
2. What geographic area will I serve, what are the existing conditions, and how much demand will be there in the market?
3. What costs and economies of scale/scope will I face?
4. What sales do I expect to see for my offering?

Predictors of Success

Predictor	Description
Size of Entry wrt Minimum Efficient Scale (MES)	Companies that are closer to an industry's MES upon entry are more likely to succeed. Entering below MES and then quickly scaling up is more viable when testing a market.
Relatedness of the Market Entered	The more related the market is to a company's current portfolio of products/ services, the greater the chance of success, but properly measuring how related is crucial.
Complementary Assets	Complementary assets such as marketing and distribution, are often more important factors for success than core assets. One should perform a value chain analysis to understand the problems and risks involved.
Industry-life-cycle stage	The life cycle stage of an industry is easily determined and greatly influences opportunities for success. Companies entering early in an industry's life cycle have greater odds for success than those entering near the shakeout
Degree of Technological Innovation	When a high level of superior information is necessary to innovate, incumbents have a major advantage over new entrants.

Solved Case — Launching Electric Trucks in India

Problem Statement

Our client, Paccar, is a Dutch premier commercial vehicle manufacturer. In the wake of the Government of India's newly announced Production-Linked Incentive scheme and reduced tax schemes for EVs, the company plans to introduce its range of electric trucks and buses in India. You have been hired to find out if they should introduce these vehicles in India.

Notes — Production-Linked Incentive Scheme

1. The PLI scheme provides an incentive of Rs. 26,058 crores, which will be delivered over five years, starting in 2021.
2. Manufacturers would have to commit to setting up a manufacturing facility with a minimum capacity of 5GWh and ensure a minimum 60% domestic value addition within five years.
3. Subsequently, the battery or cell manufacturer will have to achieve a domestic value addition of 25%, show a minimum investment of ₹225 crores per GWh within two years, and raise it to 60% domestic value addition within five years.

Approach — Structure

Customer	Needs longer range	Conservative	Opt for cheaper ICE versions	Small % receptive towards EVs
Product	Current portfolio include electric buses and trucks with a driving range of 200-300 km on a single charge.			
Company	Offers a wide product range consistent with global R&D divisions			Substantial market presence in Europe.
	Net. Annual Revenue : $ 18.7 Bn.	Net. Annual Income : $ 1.4 Bn.		
	Company enjoys consistent profitability, a strong balance sheet, and excellent cash flow.			
Industry & Costs	CAGR: 48%	3 Major / Existing Players, 4-5 Potential Entrants		High Sunk Costs
	The Government has assigned $486 million as incentives for the deployment of 7,090 electric buses, with subsidies amounting to $69,000 per bus.			
	Initial Investments as high as INR 13,500 Crores	Proper infrastructure for support is missing		

Approach — SWOT Analysis

<table>
<tr><th>Strengths</th><th>Weaknesses</th><th>Opportunities</th><th>Threats</th></tr>
<tr><td>Technical Expertise</td><td>Lack of ACC Manufacturing Capacity in India</td><td>High Dependency on Public Transport</td><td rowspan="2">Existing players are locally grown and established within the sector, having less operating costs</td></tr>
<tr><td>Environmentally Sustainable Portfolio</td><td rowspan="2">High Price of EVs (30 - 60% extra)</td><td>Rapidly progressing urbanisation across India</td></tr>
<tr><td>Excellent Cash Flow</td><td>GOI’s newly announced scheme to install charging infrastructure worth INR 20000 Cr.</td><td>High use of fossil fuels</td></tr>
<tr><td>Less substitutes</td><td>Cannot be used for Long Distance Transport</td><td>PLIs</td><td>High Sunk Costs and Low Demand</td></tr>
</table>

You can use ***SWOT, PESTEL*** or other consulting tools, as per your needs and requirements.

Recommendation

It is recommended for the company to look for possibilities to enter into a JV with existing players / potential entrants so that the company would be able to utilise existing infrastructure/ share costs. They shouldn't enter the market on their own immediately, given the existing conditions - low demand/sales for EVs and high sunk costs.

Framework Overview — Pricing

What is a Pricing Case ?

Pricing cases demand a consultant to come up with an optimum price for products or services based on several aspects like demographics, product features, market environment and purchasing power by employing various approaches to decide the final price of a new product like a market entry or reprice its older product which is a rare case.

Factors influencing Pricing

Total Costs	Market Conditions	Strategic Position
Fixed Costs, Variable Costs	Price War, Life Cycle of a Product, Seasonality and Availability	Cost Advantage, Benefit Advantage

Unique Selling Point	Government Policies and Regulations	Changing Consumer Behaviour and Expectations

Pricing Strategies

Cost-based Pricing

This is comparatively a simple method as it needs only the calculation of various costs involved in the product. Nowadays, the cost-based approach is considered insufficient, but it helps to understand the cost structure of the product. This strategy will typically give the lowest price or lower limit.

Cost Based Pricing can be done using Cost Plus Pricing and Break Even pricing.

Competitive Pricing

Using a pricing strategy based on competition that is present in the market.

- Lower Prices: Companies which capitalise on economies of scale set prices that are not profitable to attract new customers to sell more profitable goods and services.
- Higher Prices: Pricing the products higher than competitors in the market is usually employed by well- established brands.
- Equal Pricing: Companies price product equal to that of the competitors' and offer unique experiences.

Value-based Pricing

Involves setting the price of a product or service based on the value it brings to the customer. This is a function of the target segment; the wider the target segment gets, the lower the aspirational value will be.

It is essential to consider alternative and substitute products' prices before generating the final price of the product. This strategy helps us to determine the maximum price of a product.

Tips and Tricks



Supply vs Demand tradeoff approximation by graphical approach is suggested.



Check whether time value of money is considered.



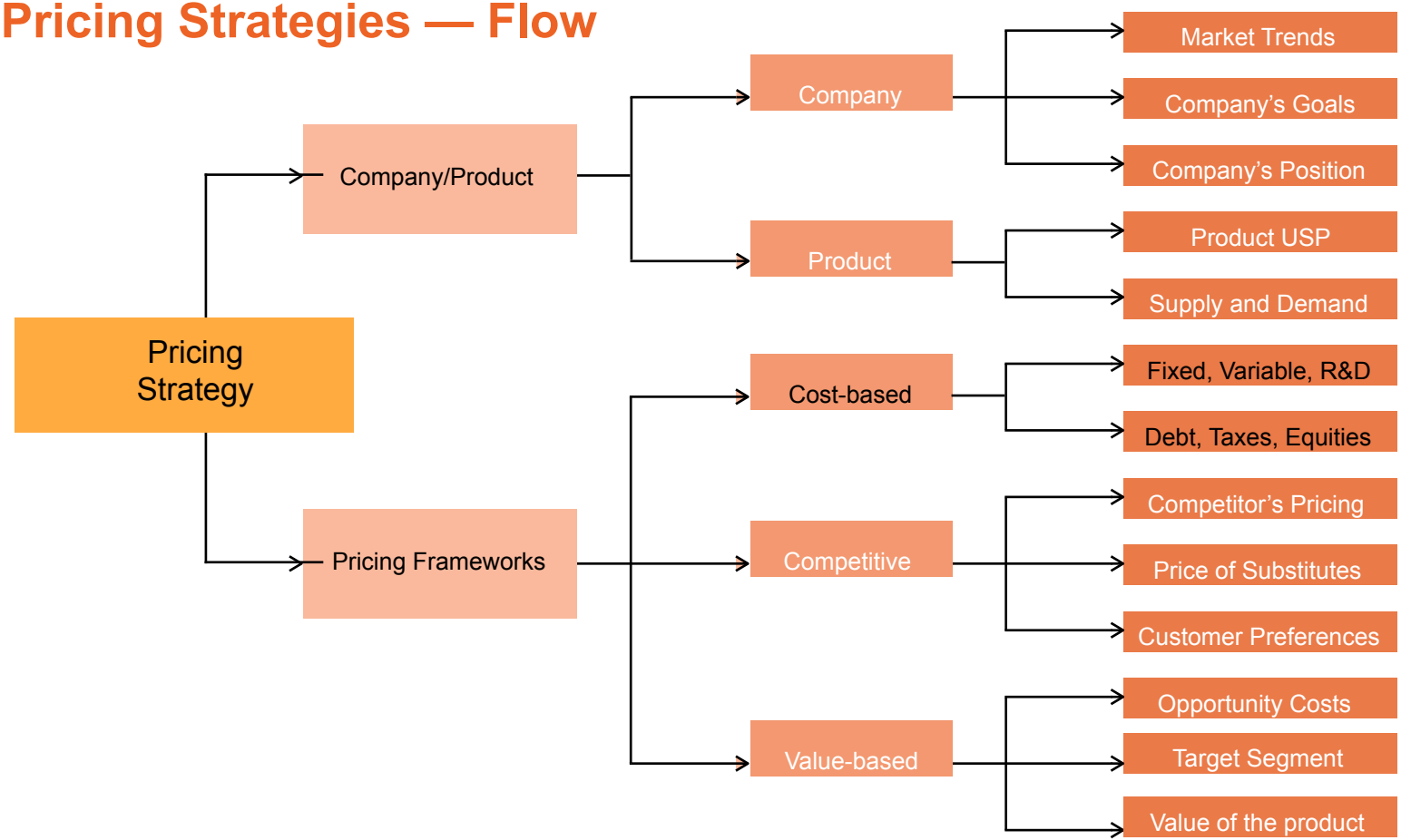
Calculating market size can be helpful.



In many cases, pricing problems can be 'masked market size' problems. Finally

deliver a recommendation.

Pricing Strategies — Flow



Case Transcript — Drone Taxi Service

Problem Statement

Our client is an airline company trying to launch a drone taxi service between Mumbai (Bandra Station) and the newly built Navi Mumbai airport. How would you price this service? Is it a viable business?

Transcript

Candidate	Interviewer
Are they operating anywhere?	Yes, they currently operate in Dubai, Los Vegas, Tokyo and Seoul.
Since this is a new product launch in India, I would like to see if the services are feasible in the Indian market. Any specific reason for choosing India?	Growth Opportunities in India, traffic congestion and a booming economy.

Case Transcript — Drone Taxi Service

Candidate	Interviewer
No existing players as well. Plus points would be time saved and luxury.	Indeed! Other forms of transport take 3 hours to commute in peak traffic hours, but this service can complete a one-way trip in 30 minutes.
Sounds good! Nearly 85% of the time was saved. Here we can go with two strategies. a) Cost Based b) Value-Based. Since there are no direct competitors, I wish not to take the competitor-based approach.	Well! We can start with Cost-based approach.
Are you planning to buy the drones or rent them?	We are collaborating with a famous drone taxi manufacturer and will use the drones on rental basis.
How many drones are we going to use? Will they be manned or unmanned?	Initially, there will only be one drone and it will be unmanned.

Case Transcript — Drone Taxi Service

Candidate	Interviewer
What are the other expenses — like maintenance, fuel, employee salaries, airport charges and rental charges of drone?	Rent of drone is 25 lakh/month, airport charges are 2 lakh/month, and employee salaries and overhead costs are 5 lakh per month. The drone is powered by hydrogen cells which cost 50,000/hr of flying.
How many passengers can travel per trip?	Eight.
Let's consider peak hours 9-11 AM morning and 5-8 PM in the evening. Total service time is 5 hr/day and 25 working days per month. It turns out to be 5 round trips per day.	Nice! Go ahead. Also, assume 100% occupancy rate at all times.
We can have a total of 125 trips/month. It will cost 3,78,000/day. If 100% occupancy is considered, we can provide service to 80 passengers/day. So, a one- way ticket should cost around 4800 rupees.	

Case Transcript — Drone Taxi Service

Candidate	Interviewer
On the other hand, it would cost only a maximum of 800 rupees by road. So the difference is 4000 rupees. Though the service is expensive, the rich can afford it, and the services are to be restricted to busy areas and at peak hours. It can be expanded with a similar strategy in other metro cities as well.	Well Done! We can conclude our case here.

Key Takeaways

A basic approach to pricing cases consists of the following questions:

1. ***USP:*** What is the product's unique selling point?
2. ***Cost Price:*** What is the cost of production?
3. ***Competitor Analysis:*** How have the competitors priced their products?
4. ***Value:*** What are the benefits of this product to the customer?

Framework Overview — Growth Strategy

A scenario involving a growth plan or revenue increase is typical in your consulting interviews.

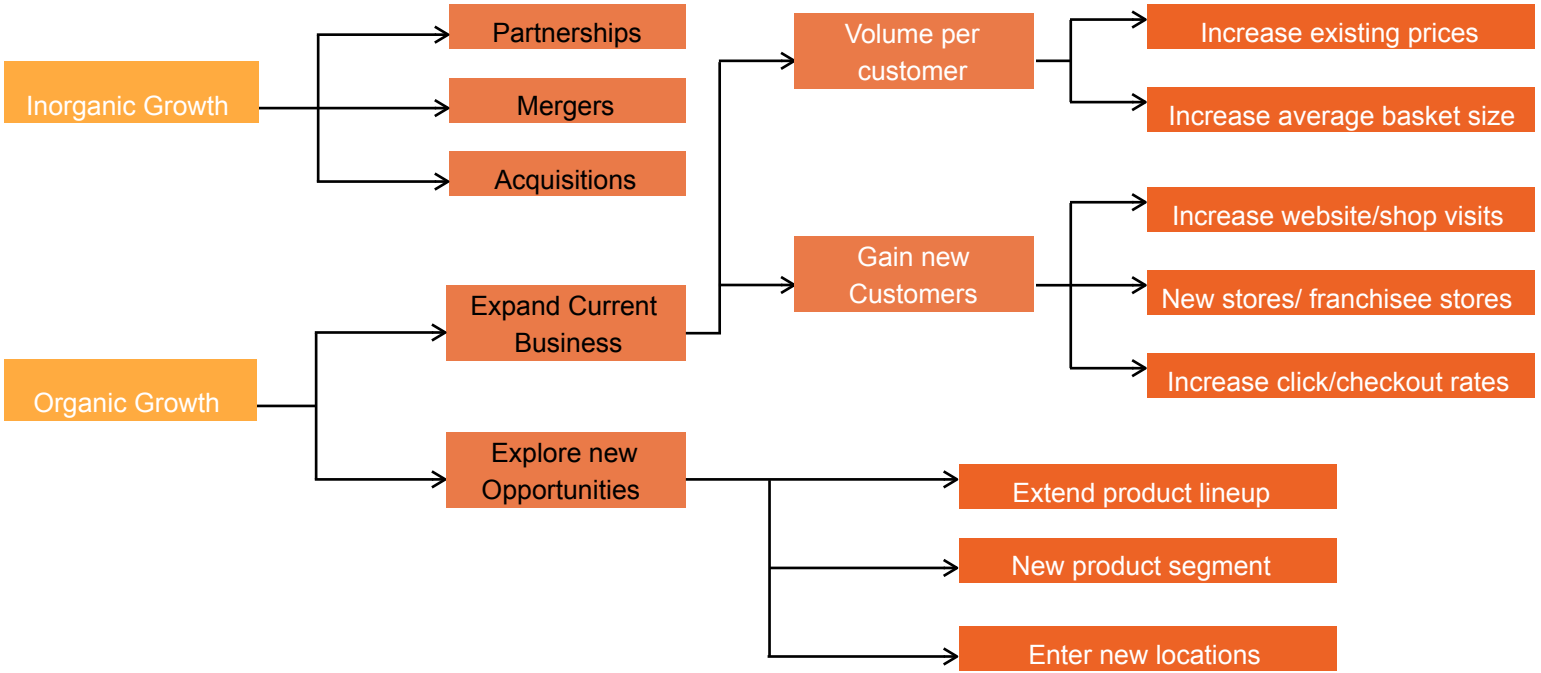
Case studies involving situations of revenue growth may be handled by changing two key factors that affect growth rates. These factors are mainly customers and orders/billings.

There are two most common types of growth strategies: Organic and Inorganic growth.

Organic growth, fuelled by increasing output or engaging in internal activities, is the most prevalent sort of growth that businesses attempt. In other words, the company is developing due to its resources and initiatives.

On the other side, growth fuelled through partnerships, joint ventures, or acquisitions is known as ***Inorganic growth.***

Structure — Growth Strategy



Key Takeaways

The following questions are to be answered in a Growth Strategy case.

1. What's the industry average growth?
2. What are current and target growth rates?
3. Amount of cash available/cash flow and time.
4. Our existing strengths in the targeted segment
5. What is the competition doing right and wrong?
6. What method are we looking into (organic/inorganic) and scale?
7. Price elasticity and sensitivity of the market
8. People's new demands/preferences
9. Are there any threats, hurdles or barriers to our plans?
10. What will be the expected return in terms of Revenue and Profits?

Case Transcript — Pharma Sector

Problem Statement

Your client is major drug manufacturer. Its business was booming during the COVID times, but its revenue and market share have fallen in the last quarter. It has approached you to help to grow its business and reverse its declining trend.

Transcript

Candidate	Interviewer
I'd like to clarify a few things before I analyse the case. I want to comprehend the client's position in its industry. What position does it have in the value chain?	Our client is one of the leading pharmaceutical companies in India. The company focuses on developing new formulations and producing a large variety of drugs and a wide range of pharmaceutical products.

Case Transcript — Pharma Sector

Candidate	Interviewer
Is the problem industry-wide or specific to our client?	Only our client is facing the problem as there is an overall growth trend in the industry of about 5-7%. However, since the reduction in COVID-19, the client is facing a revenue drop.
I believe the revenue reduction is mainly because our client was producing a large variety of drugs used to control or treat COVID. Is that correct?	Yes, that is correct. However, we are looking for ways to improve growth and regain market share. Could you suggest some ways for the same?
We can look forward to both organic and inorganic growth streams. We can look forward to the acquisition of some regional players doing well, and this would lead to mutual growth. On the other hand, we can focus on reaching more customers, increase our offering variety, or expand business in areas with low presence.	I would like you to focus more on our current capacities.

Case Transcript — Pharma Sector

Candidate	Interviewer
The issue can be related to either supply or demand. I would consider factors like production capacity, accessibility, and product mix while considering the supply side. I would consider price, rivals, and product quality from the demand perspective. So, would you like me to focus on the supply or demand first?	I think the issue is majorly based on the supply, so let's take that first.
In that case, let's focus on our existing product mix and operational efficiency. What are the major product categories we offer?	The client is operating at almost 95% efficiency, which is the industry standard and makes mainly prescription drugs and is a minor player in OTC drugs and is involved in mass production of these drugs. The drugs are sold mostly through hospitals, chemists and pharmacy stores. It also has an export business of drugs to the EU, USA and Africa (mainly South Africa).

Case Transcript — Pharma Sector

Candidate	Interviewer
Since we don’t offer many OTC drugs, I feel that we can look into that segment as it's one of the fastest growing segments with higher than industry average growth rates. Also, that will help increase our online presence as most people buy OTCs from e-retailers and prescription drugs from stores. We can use our vast manufacturing setup to gain a foothold in that segment and establish cost leadership.	
We can also look towards selling in bulk to insurance companies and suppliers as I feel users would be more than happy to buy our products due to our lower cost structure. Also, I wanted to know about other areas where you would like me to focus, like exports or R&D.	The client is satisfied with its current R&D developments. However, we can look into exports and foreign markets.

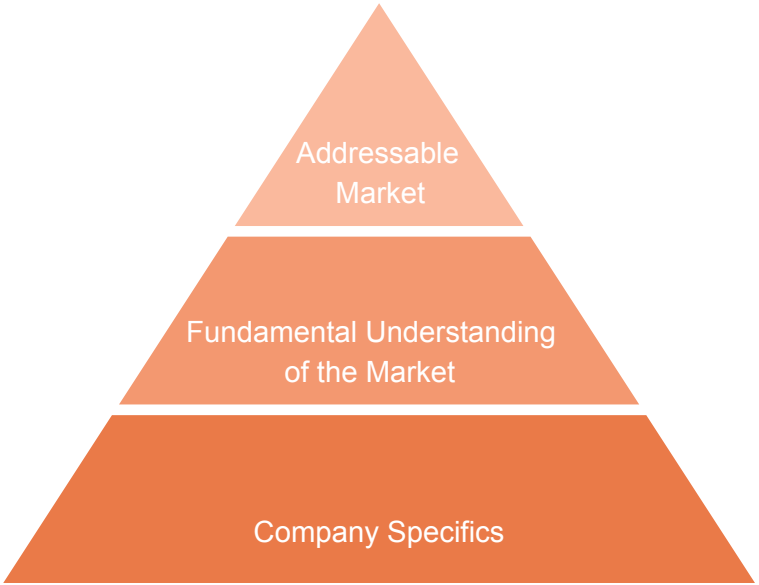
Case Transcript — Pharma Sector

Candidate	Interviewer
Which are the key countries we export to and manufacture in?	We mainly manufacture in India due to cheaper labour and production costs. However, we supply to the EU, South Africa, USA and Canada.
We can also look to set up a manufacturing plant in South Africa if possible, as it would help to lower costs in the long run and allow us to expand our business in that continent; it may also help to reduce costs to sell to EU and North America.	Finally, I want you to conclude with three critical recommendations for our client.
First, we should expand our OTC (over-the-counter) drugs business and increase our presence in e-pharmacies. Secondly, we can look to strike a deal with insurance companies to provide our drugs to their clients. Finally, we should look to strengthen our position in South Africa and set up a plant there if possible to help increase our global business.	That was excellent. That is all from my side. Thank you.

Top Down Approach or 'Chain Ratio Method'

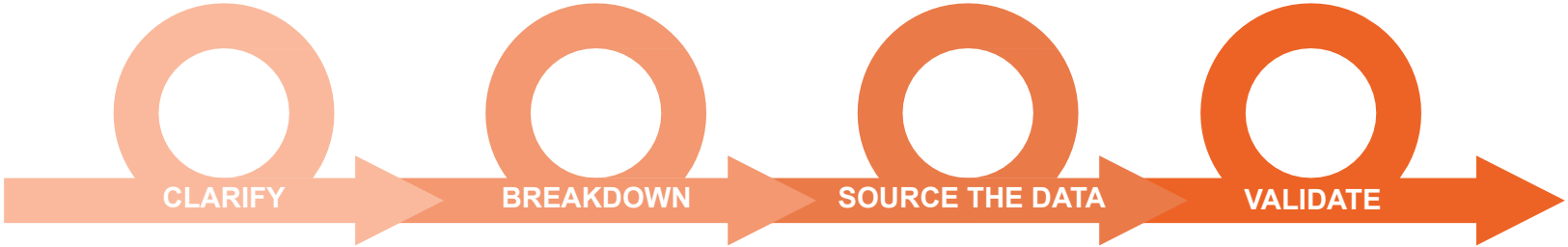
Starting with a comprehensive market size estimate, the addressable market is estimated and then reduced to the target market segments using assumptions and data. A firm must first identify the customer segments it intends to reach and then assess its size and growth.

Let's take the example of determining the market size for toothbrushes. In a top-down market sizing framework, one would start with the Indian population and then move on to estimate what percentage of the population uses a toothbrush. Next, they would count how many toothbrushes the average person goes through in a year and the cost per toothbrush. Multiplying all these figures helps estimate the market size.



Framework Overview — Market Sizing

Market size refers to the maximum number of sales or customers a business can see, often measured over a year. Knowing the potential market size before launching a new product line or line of business can help understand if it's a worthwhile investment of time and money. Typically, any market sizing exercise follows a four-step pattern:



Clarify the question by defining the product/ service measured and by conceptually breaking down the market by sub-products, applications, customer segments, geographies, etc.

Build your market sizing model by **breaking the problem** into smaller pieces, and **map out** your calculations. The two principal methods are **Top-Down Approach** and the **Bottom-Up Approach**.

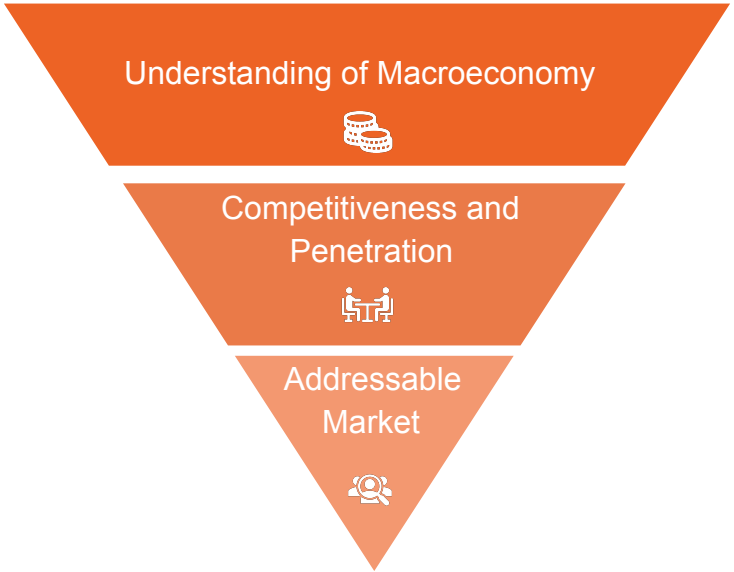
Gather the data to resolve each of the smaller pieces and combine them into a single outcome. Sourcing the data is always a mix of secondary research, primary research, and analysis.

The final step is to **validate your outcome** using product definitions and segmentation, examining your calculations, sources and assumptions, as well as run sanity tests.

Bottom Up Approach or 'Counting Noses'

To do a bottom-up analysis, you start with the basic units of your business (your product, price, customers) and estimate how large you can scale those units. A firm must first identify the customer segments it intends to reach and then calculate its size and growth.

Let's take the example of a retailer selling fruits. How big is the market for this retailer? A Bottom-Up market sizing needs to start with the price of fruits, then move on to calculating the number of consumers he can reach and the average amount of money a consumer spends on fruits. We can also consider expanding to other cities depending on the available capital and operating plan. This gives us the size of the market for this retailer.



Case Transcript — Luxury Jewellery Segment

Problem Statement

The client is an American brand in the business of making luxury jewellery. The client wants to expand to the Indian market. You have been hired to find out the estimated size of the target market in India.

Transcript

Candidate	Interviewer
I want to be sure I comprehend all the essential details of the client's circumstances. They are American luxury jewellery and speciality retailer and wish to enter the Indian Market. We must assess the market and determine the size of the target market in India.	That’s right. Now that you've understood the situation well, how do you propose going about the solution?

Case Transcript — Luxury Jewellery Segment

Candidate	Interviewer
Given that this is a brand-new product launch, I'd like to organise my discussion around the product's characteristics (creation and customisation for the Indian market) before moving on to the market size.	This sounds fine. Also, this product is not entirely new; it has already been introduced in other countries.
To start with, can you tell me something more about the product? How is it different?	The biggest strength of our client is its global luxury brand image. Another strength is their transparency in providing diamond information which consecutively ensures high-quality diamonds and other stones.
That is good. It gives us the advantage of positioning our product as superior and with a loyal customer base. I want to conduct a competitive analysis next so we can decide the price before determining the overall market size.	The first competitor in Indian Market is TBZ- The Original, which has a reputation for high-calibre craftsmanship. The second and strongest competitor is Tanishq having a solid reputation due to its parent company being the world-famous TATA Group. Some other competitors include Senco and Malabar Jewels which have a solid customer base among senior women.

Case Transcript — Luxury Jewellery Segment

Candidate	Interviewer
Some factors I would like to consider while estimating are the purchase frequency and willingness to pay (for a premium brand).	That is fair. So what approach would be better for this problem while making the estimations?
The top-down approach. We start by considering the population of India. And it would be fair to assume that most of our consumer base will be the female adult population of India. Based on the frequency of purchase, we can fairly divide them into frequent buyers, casual/occasional buyers, and those who would not buy at all. (See calculation).	Good. What do you think will drive the market share for the client?
The Indian cohort may not relate to paying a massive price for a brand. They have never experienced this due to their trust in traditional jewellers, passed on from generation to generation.	

Case Transcript — Luxury Jewellery Segment

Candidate	Interviewer
What other countries has the product been introduced to? How accepting have the people in those nations been of it?	The other countries have had good reception due to better standards and cost of living. The product has been quite a success. The percentage of upper-class people is more in those nations than in India.
So, our client in India will focus on gaining market share and increasing sales by targeting the elite class who can afford to pay for the exclusive jewellery. To estimate the pricing, it is fair to assume that this addressable market will be highly willing to pay. We should be able to calculate the optimal profit case by considering the trade-off in sales volume vs price for various price points. The solution will also be influenced to an extent by the growth rates of the different target segments overall, like the movement to the upper class from the middle class.	Yes… that is good. Let's assume we did this and came up with a price X; now, try estimating the market size with this assumption.

Case Transcript — Luxury Jewellery Segment

Candidate	Interviewer
However, the globalised brand image of the client and the fact that it is the first of its kind in terms of western jewellery in India will play a significant factor in attracting a young audience. Down the line, our client can focus on understanding Indian sensibilities and match Indian taste in their products to appeal to a broader audience.	Very Interesting, What about a short-term strategy?
As a short-term strategy, the client can focus on building a personalised digital omnichannel presence as most of the audience spend some time surfing through the web, curating and comparing designs before making a purchase.	Good, I think we have covered the different aspects of the case. Thank You!

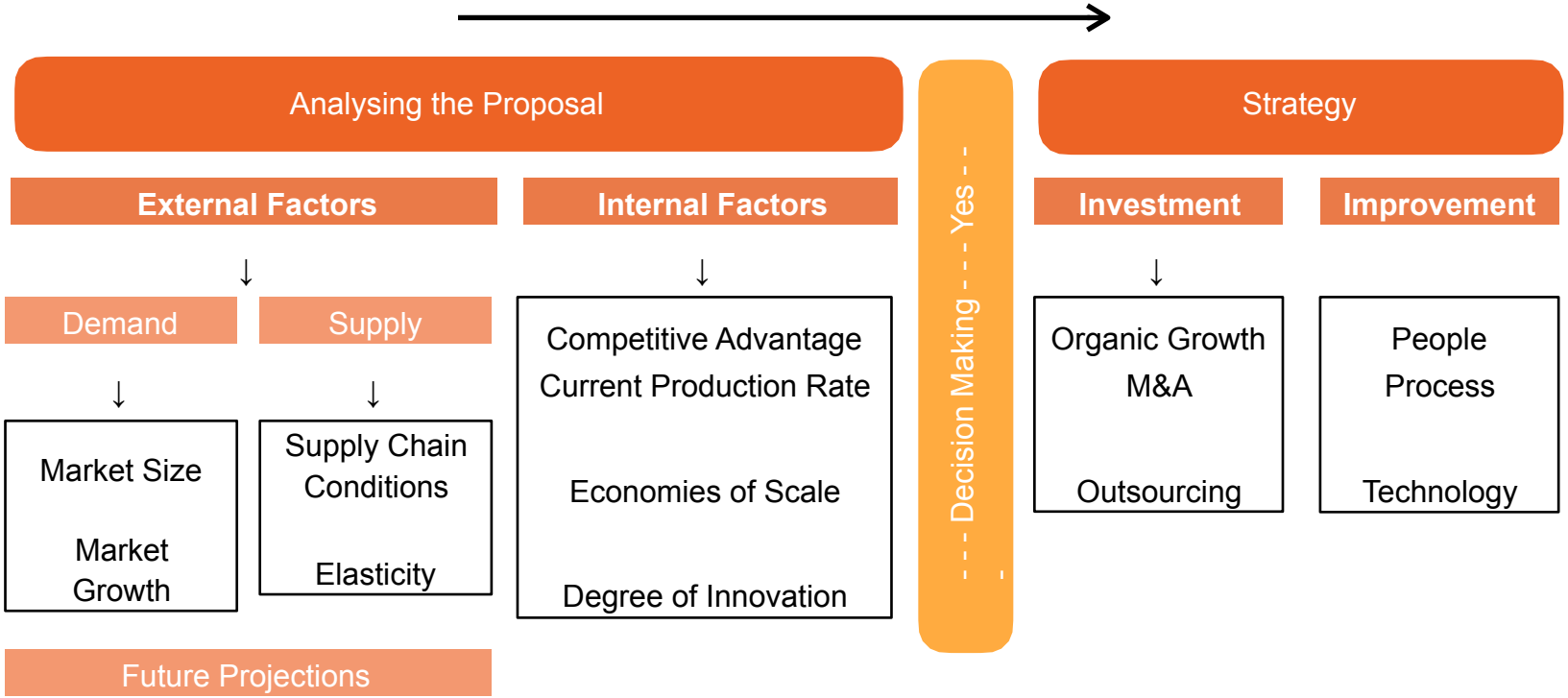
Considerations and Calculations

The current population of India is 138 crores, of which 60 crores (approx.) are adult females.
The Population of India is categorised by the standard of Living (out of 60 Cr): 10% elite, 20% upper middle class, 30% middle class, and 40% lower middle class.

	Would not buy at all	Casual Buyers	Frequent Buyers
% female population	Lower middle class + 70% middle class	30% middle class + Upper middle class	Elite class
Population size	24 Cr + 12.6 Cr	5.4 Cr + 12 Cr	6 Cr
Frequency and quantity of purchase	N/A	2 items/year	Every two months, 1 item/ purchase = 6 items/year
Total consumption	0	2*(12.0 Cr + 5.4 Cr) = 34.8 Cr	6*(6Cr) = 36 Cr

Total: 34.8 Cr + 36 Cr = ~71 Cr. Now, for avg. price per unit X, the market size will be **71(X) Cr**.

Framework Overview — Organisational Capacity Expansion



Framework Overview — Mergers and Acquisitions?

What are Mergers and Acquisitions?

Mergers and acquisitions (M&A) refer to transactions between companies converging in some form. Although the terms are used interchangeably, they come with different legal meanings. In a merger, two companies of similar size combine to form a new single entity. An acquisition is when a larger company absorbs a smaller company's business.

Other transactions fall under the M&A umbrella, such as consolidations (where a new company is created by combining core businesses and abandoning old corporate structures) and tender offers (one company offers to purchase the outstanding stock of the other firm at a specific price rather than the market price). There are also asset acquisitions and management-lead buyouts (like the 2013 acquisition of Dell by its founder) — however, these topics won't be discussed much in this guide.



M&A deals can be friendly or hostile, depending on the approval of the target company's board.

Reasons for Mergers and Acquisitions

Unlocking Synergies

Cost synergies are created due to economies of scale, while revenue synergies are created by cross-selling, increasing market share, or higher prices.



Higher Growth

Inorganic growth through M&A is usually a faster way for a company to achieve higher revenues than growing organically.



Stronger Market Power

In a horizontal merger, the resulting entity will attain a higher market share. In vertical M&As, the company will gain more control of its supply chain.



Diversification

Acquiring a target in a non-cyclical sector enables a company to diversify and reduce the risk of facing significant losses in an industry slowdown.



Tax Benefits

Acquiring the company with the tax losses enables the acquirer to use the tax losses to lower its tax liability.



Types of Mergers

Horizontal	Vertical	Conglomeration
A horizontal merger happens between companies operating in similar industries that may or may not be direct competitors.	A vertical merger occurs between a company and its supplier or a customer along its supply chain, aiming to move up or down along its supply chain.	This transaction is usually done for diversification and between companies in unrelated industries.

Congeneric	Market Extension	Product Extension
A congeneric merger occurs between companies with a common customer base but serves them differently.	Occurs between two companies that sell the same products, but in different markets.	Occurs between two companies selling different but related products in the same market.

Case Transcript — Luxury Jewellery Segment

Problem Statement

The client is an American brand in the business of making luxury jewellery. The client wants to expand to the Indian market. You have been hired to find out the estimated size of the target market in India.

Transcript

Candidate	Interviewer
I want to be sure I comprehend all the essential details of the client's circumstances. They are American luxury jewellery and speciality retailer and wish to enter the Indian Market. We must assess the market and determine the size of the target market in India.	That’s right. Now that you've understood the situation well, how do you propose going about the solution?

Case Transcript — Luxury Jewellery Segment

Candidate	Interviewer
Given that this is a brand-new product launch, I'd like to organise my discussion around the product's characteristics (creation and customisation for the Indian market) before moving on to the market size.	This sounds fine. Also, this product is not entirely new; it has already been introduced in other countries.
To start with, can you tell me something more about the product? How is it different?	The biggest strength of our client is its global luxury brand image. Another strength is their transparency in providing diamond information which consecutively ensures high-quality diamonds and other stones.
That is good. It gives us the advantage of positioning our product as superior and with a loyal customer base. I want to conduct a competitive analysis next so we can decide the price before determining the overall market size.	The first competitor in Indian Market is TBZ- The Original, which has a reputation for high-calibre craftsmanship. The second and strongest competitor is Tanishq having a solid reputation due to its parent company being the world-famous TATA Group. Some other competitors include Senco and Malabar Jewels which have a solid customer base among senior women.

Case Transcript — Luxury Jewellery Segment

Candidate	Interviewer
What other countries has the product been introduced to? How accepting have the people in those nations been of it?	The other countries have had good reception due to better standards and cost of living. The product has been quite a success. The percentage of upper-class people is more in those nations than in India.
So, our client in India will focus on gaining market share and increasing sales by targeting the elite class who can afford to pay for the exclusive jewellery. To estimate the pricing, it is fair to assume that this addressable market will be highly willing to pay. We should be able to calculate the optimal profit case by considering the trade-off in sales volume vs price for various price points. The solution will also be influenced to an extent by the growth rates of the different target segments overall, like the movement to the upper class from the middle class.	Yes… that is good. Let's assume we did this and came up with a price X; now, try estimating the market size with this assumption.

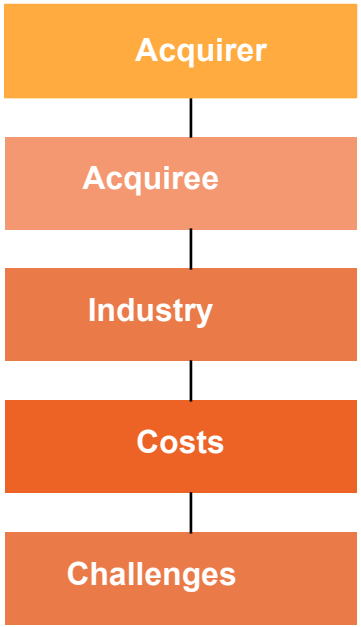
Case Transcript — Luxury Jewellery Segment

Candidate	Interviewer
Some factors I would like to consider while estimating are the purchase frequency and willingness to pay (for a premium brand).	That is fair. So what approach would be better for this problem while making the estimations?
The top-down approach. We start by considering the population of India. And it would be fair to assume that most of our consumer base will be the female adult population of India. Based on the frequency of purchase, we can fairly divide them into frequent buyers, casual/occasional buyers, and those who would not buy at all. (See calculation).	Good. What do you think will drive the market share for the client?
The Indian cohort may not relate to paying a massive price for a brand. They have never experienced this due to their trust in traditional jewellers, passed on from generation to generation.	

Case Transcript — Luxury Jewellery Segment

Candidate	Interviewer
However, the globalised brand image of the client and the fact that it is the first of its kind in terms of western jewellery in India will play a significant factor in attracting a young audience. Down the line, our client can focus on understanding Indian sensibilities and match Indian taste in their products to appeal to a broader audience.	Very Interesting, What about a short-term strategy?
As a short-term strategy, the client can focus on building a personalised digital omnichannel presence as most of the audience spend some time surfing through the web, curating and comparing designs before making a purchase.	Good, I think we have covered the different aspects of the case. Thank You!

Structure



How is the acquirer performing?
What are the trends in the profit margin of the client?

What are the acquiree's financials?
What is its USP?

Is the industry growing/attractive?
What are the supply chain conditions overall?

What are the costs involved in the acquisition?
Can the client sustainably handle itself after the acquisition?

What are the possible challenges the acquirer could face after the acquisition?
What are the risks regarding adaptability and financial and operational feasibility?

Framework Overview — Private Equity Investment

Private Equity (PE) Investment comprises funds and investors directly investing in private companies or engaging in buyouts of public companies, effectively delisting the public equity.

The following aspects should be considered during a PE investment:

PE Firm Characteristics	Industry Attractiveness	Target Specific Metrics
Fund Size	Market Size	Business Model
Fund Style	Growth Rate	Valuation
Portfolio	Barriers to Entry and Exit	Management Capability
NPV / IRR	Competition	Portfolio
Exit Period	Customers	Profitability
	Supply Chain Conditions	
	Elasticity	

The scores can be assigned to each in the same way implemented with the **Nine-Box Model**.

Practice Problem — Investment in Sustainable Packaging Solutions

Problem Statement

With a growing demand for sustainable solutions, Capital Green is looking for investment opportunities in recycled packaging businesses. One such company, StaxPak UK, has developed a new cost-effective technology that reduces packaging costs by up to 25%. However, it needs additional funding of $200 million to start production facilities for mass-market adoption of this technology. Should Capital Green invest in the company? Identify probable risks while assessing.

Initial Considerations

StaxPak UK Statistics		Industry Statistics	
Total Assets	GBP 1.235 Bn	CAGR (2021-27)	8%
Current Liabilities FY22	GBP 0.475 Bn	Key Markets	APAC, EMEA
Income	GBP 0.065 Bn	Application	Dairy and meat, Personal Care

Transcript

Candidate	Interviewer
So our client, CapitalGreen, wants to know whether they should pursue this investment opportunity in StaxPak, which has developed a new cost-effective technology to reduce costs by 25%. Am I right?	Yes, you are. How will you approach this case?
I'd like to ask a few clarifying questions. Is it safe to assume that the company operates nationwide and that domestic business is the primary arm of its business?	That is a fair observation; however, StaxPak also sells its products in other parts of Europe and recently started exporting the material to the US.
Also, what are the existing supply chain conditions, and what model has StaxPak adopted to sell its products? Is it similar to what other companies have adopted?	Like other companies in the industry, StaxPak operates via a direct-to-business model, and companies often place the order two months prior to meet their needs. But how is that relevant here?

Transcript

Candidate	Interviewer
Since the packaging industry heavily depends on its distribution channels, any optimisation would positively impact the cost of the material sold. Also, has the working capital of the company increased year-over-year?	Although the working capital has increased, the current liabilities have too. What's your take on that?
This could be mainly due to the supply-demand imbalance and rising input costs. It could also arise due to additional expenses encountered while exporting to the US and the COVID-19 pandemic.	Yes, the costs have increased. Why don't you start by identifying strategies to solve these issues and make suggestions for the client?
StaxPak could expand to the APAC and middle-eastern regions as they've been identified as the largest markets for their products and also offers cheap labour and supply.	Are there other avenues the company should focus to reduce costs?

Transcript

Candidate	Interviewer
It should also analyse its operations in the US to check if the market is operationally feasible. It could also use predictive technologies to reduce process cycle times (between ordering and delivery). Shall I proceed to analyse whether this investment opportunity would be profitable for the client?	Now that you have listed out strategies to resolve the major concerns, you may proceed to drafting the recommendation for Capital Green.
Are there any major legal barriers in the UK or Asia to set up a new production facility? Also shall I take that the private equity firm would have an exit period not less than five years?	There are no legal barriers that would prevent setting up a new production facility. Also, you may assume Capital Green is not looking to exit the business before five years.
Given that the market is growing with a CAGR of 8%, Capital Green will be able to receive its investment and make profits of over 20% in not more than 4 years. If StaxPak builds its new facility in Asia, it could also pay dividends at a much faster rate.	Good. That would be all.

Transcript

Candidate	Interviewer
It should also analyse its operations in the US to check if the market is operationally feasible. It could also use predictive technologies to reduce process cycle times (between ordering and delivery). Shall I proceed to analyse whether this investment opportunity would be profitable for the client?	Now that you have listed out strategies to resolve the major concerns, you may proceed to drafting the recommendation for Capital Green.
Are there any major legal barriers in the UK or Asia to set up a new production facility? Also shall I take that the private equity firm would have an exit period not less than five years?	There are no legal barriers that would prevent setting up a new production facility. Also, you may assume Capital Green is not looking to exit the business before five years.
Given that the market is growing with a CAGR of 8%, Capital Green will be able to receive its investment and make profits of over 20% in not more than 4 years. If StaxPak builds its new facility in Asia, it could also pay dividends at a much faster rate.	Good. That would be all.



FOCUS

01

SQL:

Understanding SQL

What is Structured Query Language

SQL, also known as Structured Query Language, is a powerful tool to search through large amounts of data and return specific information for analysis

There are many different sql dialects - PostgreSQL, MySQL, SQLLite , SQL Server,Oracle SQL.

SQL Syntax:

SQL is written using statements, which consist of keywords like SELECT, FROM, WHERE, etc., to perform operations on the data. SQL keywords are not case-sensitive, but it's common to write them in uppercase (e.g., SELECT, FROM).

SQL Data Types

Databases store data in different formats. The most common data types are:
- INT (Integer): For whole numbers.
- VARCHAR(n) or TEXT: For storing text data.
- DATE: For dates.
- DECIMAL: For precise decimal values, often used in financial calculations.



Constraints in SQL

Constraints are rules applied to columns or tables to enforce data integrity, consistency, and accuracy

NOT NULL

- Ensures that a column **cannot have a NULL value**.

```
CREATE TABLE Students (
    StudentID INT NOT NULL,
    Name VARCHAR(100) NOT NULL
);
```

PRIMARY KEY

- Uniquely identifies each record in a table.
- Combines **UNIQUE** + **NOT NULL**.
- Only **one primary key** per table (can be composite across multiple columns)

```
CREATE TABLE Students (
    StudentID INT PRIMARY KEY,
    Name VARCHAR(100)
);
```

Constraints in SQL

UNIQUE

- Ensures all values in a column are **different**.
- Allows **only one NULL** (varies by SQL dialect).

```sql
CREATE TABLE Employees (
    Email VARCHAR(100) UNIQUE
);
```

FOREIGN KEY

- Establishes a relationship between two tables.
- The foreign key in one table references the **primary key** in another.
- Enforces **referential integrity**

```sql
CREATE TABLE Orders (
    OrderID INT PRIMARY KEY,
    CustomerID INT,
    FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)
);
```

Constraints in SQL

CHECK

- Limits the **range of values** in a column using a Boolean condition

```
CREATE TABLE Products (
    Price DECIMAL(10, 2),
    CHECK (Price > 0)
);
```

DEFAULT

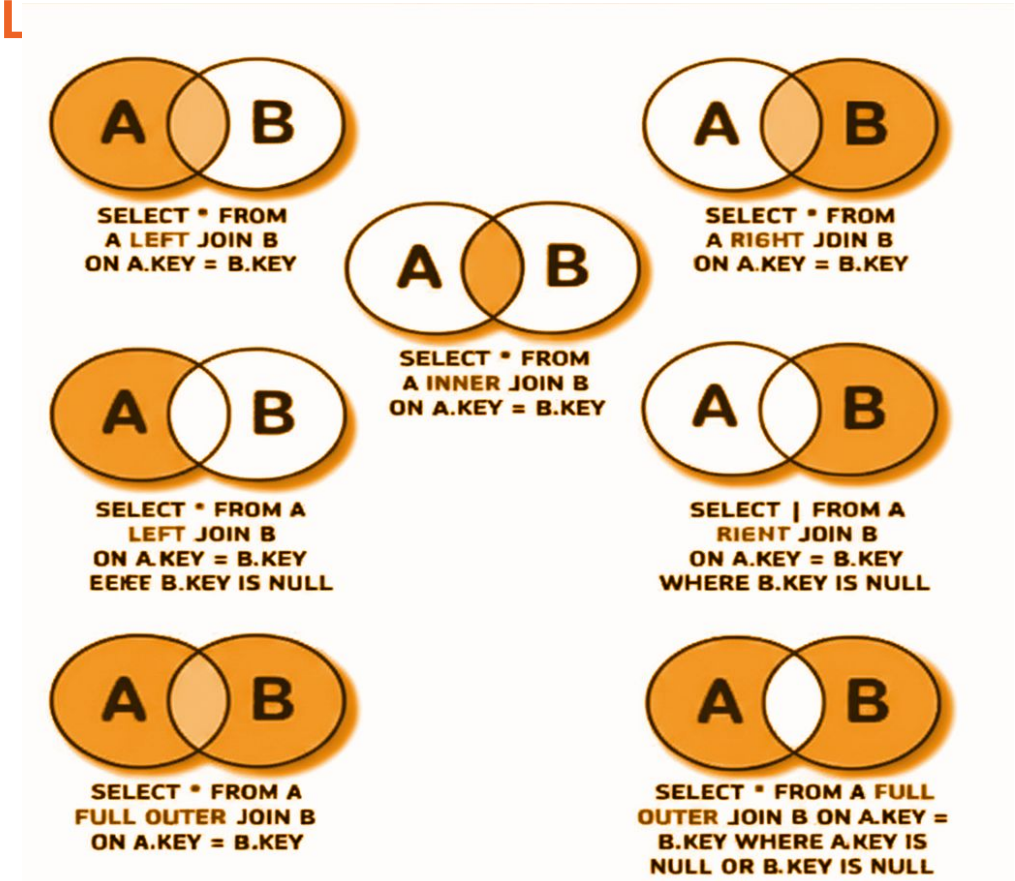
- Sets a **default value** for a column if no value is provided

```
CREATE TABLE Users (
    Status VARCHAR(10) DEFAULT 'active'
);
```

INDEX (Not a constraint, but often included)

- Improves query performance but does **not enforce** data rules

Joins in SQL



Read this for more:https://datalemur.com/blog/sql-joins-infographics

RIGHT JOIN (RIGHT OUTER JOIN)

- Returns **all rows from the right table (B)**, plus matches from the left (A).
- If no match, left side = **NULL**.
- *"Keep all right rows, match if you can."*

FULL OUTER JOIN

- Returns **all rows from both tables**, matched or not.
- Fills unmatched parts with **NULLs**.
- *"Everything from both tables."*

INNER JOIN

- Returns **only matching rows** from both tables.
- *"Give me the overlap."*

LEFT JOIN (LEFT OUTER JOIN)

Returns **all rows from the left table (A)**, plus matches from the right (B).

- If no match, right side = **NULL**.
- *"Keep all left rows, match if you can."*

What are tables and fields in SQL?

A table is an organized set of related data stored in a tabular form, i.e., in rows and columns. A field is another term for a column of a table.

What is an SQL query, and what types of queries do you know?

A query is a piece of code written in SQL to access or modify data from a database.

There are two types of SQL queries: **select** and **action** queries. The first ones are used to retrieve the necessary data (this also includes limiting, grouping, ordering the data, extracting the data from multiple tables, etc.), while the second ones are used to create, add, delete, update, rename the data, etc.

What is a subquery?

Also called an inner query, a query placed inside another query, or an outer query. A subquery may occur in the clauses such as `SELECT`, `FROM`, `WHERE`, `UPDATE`, etc.

What types of SQL subqueries do you know?

- **Single-row** – returns at most one row.
- **Multi-row** – returns at least two rows.
- **Multi-column** – returns at least two columns.
- **Correlated** – a subquery related to the information from the outer query.
- **Nested** – a subquery inside another subquery.

What SQL constraints do you know?

- `DEFAULT` – provides a default value for a column.
- `UNIQUE` – allows only unique values.
- `NOT NULL` – allows only non-null values.
- `PRIMARY KEY` – allows only unique and strictly non-null values (`NOT NULL` and `UNIQUE`).
- `FOREIGN KEY` – provides shared keys between two or more tables.

What is a join?

A clause used to combine and retrieve records from two or multiple tables. SQL tables can be joined based on the relationship between the columns of those tables. Check out our SQL joins **tutorial** for more context, plus our dedicated guide to **SQL joins interview questions**.

What types of joins do you know?

- `(INNER) JOIN` – returns only those records that satisfy a defined join condition in both (or all) tables. It's a default SQL join.
- `LEFT (OUTER) JOIN` – returns all records from the left table and those records from the right table that satisfy a defined join condition.
- `RIGHT (OUTER) JOIN` – returns all records from the right table and those records from the left table that satisfy a defined join condition.
- `FULL (OUTER) JOIN` – returns all records from both (or all) tables. It can be considered as a combination of left and right joins.

What is a unique key?

A column (or multiple columns) of a table to which the `UNIQUE` constraint was imposed to ensure unique values in that column, including a possible `NULL` value (the only one).

What is a foreign key?

A column (or multiple columns) of a table to which the `FOREIGN KEY` constraint was imposed to link this column to the primary key in another table (or several tables). The purpose of foreign keys is to keep connected various tables of a database.

What is an index?

A special data structure related to a database table and used for storing its important parts and enabling faster data search and retrieval. Indexes are especially efficient for large databases, where they significantly enhance query performance

Some common / Basic questions

What types of indexes do you know?

- **Unique index** – doesn't allow duplicates in a table column and hence helps maintain data integrity.
- **Clustered index** – defines the physical order of records of a database table and performs data searching based on the key values. A table can have only one clustered index.
- **Non-clustered index** – keeps the order of the table records that don't match the physical order of the actual data on the disk. It means that the data is stored in one place and a non-clustered index – in another one. A table can have multiple non-clustered indexes.

What is a schema?

A collection of database structural elements such as tables, stored procedures, indexes, functions, and triggers. It shows the overall database architecture, specifies the relationships between various objects of a database, and defines different access permissions for them. Read our **database schema guide** for a deeper understanding

Recommended reads: https://iam.highonproduct.com/modules/preparing-for-interviews/sql-for-pm-interviews



FOCUS

02

SQL:

SQL From a PM Lens

A Product Manager's Perspective on SQL

The power of data

It's painful, it's brutal and it's a little sadistic. But it's wonderful.
Data to product managers is like ammunition to an assassin; nothing gives us more pleasure than gently annihilating a loud, forceful stakeholder with an incisive bullet of data which obliterates their stance in a few seconds.

As a Product Manager, you're not expected to be a data engineer or write complex queries daily. However, SQL becomes an invaluable tool for **data-driven decision-making**, especially when collaborating with data teams or validating insights yourself

Tips for PMs Using SQL

- **Start simple**: Learn to ask the right question before writing the query.
- **Automate dashboards**: Once validated, convert your queries into dashboards using tools like Metabase, Looker,Tableau.
- **Validate hypotheses**: Use SQL to test assumptions quickly before taking them to engineering.

SQL Cheat Sheets

SELECT

The `SELECT` statement is used to select data from a database.

SYNTAX:

```
SELECT column1, column2, ...
FROM table_name;
```

SELECT DISTINCT

The `SELECT DISTINCT` statement is used to return only distinct (unique) values.

SYNTAX:

```
SELECT DISTINCT column1, column2, ...
FROM table_name;
```

WHERE

The `WHERE` clause is used to filter records.

SYNTAX:

```
SELECT column1, column2, ...
FROM table_name
WHERE condition;
```

ORDER BY

The `ORDER BY` keyword sorts the result-set in ascending order (ASC) by default.

SYNTAX:

```
SELECT column1, column2, ...
FROM table_name
ORDER BY column1, column2, ... ASC|DESC;
```

AND

The `AND` operator is used to filter records based on more than one condition.

SYNTAX:

```
SELECT column1, column2, ...
FROM table_name
WHERE condition1 AND condition2 AND condition3 ...;
```

OR

The `OR` operator is used to filter records based on more than one condition.

SYNTAX:

```
SELECT column1, column2, ...
FROM table_name
WHERE condition1 OR condition2 OR condition3 ...;
```

NOT

The `NOT` operator is used in the `WHERE` clause to return all records that DO NOT match the specified criteria.

SYNTAX:

```
SELECT column1, column2, ...
FROM table_name
WHERE NOT condition;
```

INSERT INTO

The `INSERT INTO` statement is used to insert new records in a table.

SYNTAX:

```
INSERT INTO table_name (column1, column2, column3, ...)
VALUES (value1, value2, value3, ...);
```

UPDATE

The `UPDATE` statement is used to update or modify one or more records in a table.

SYNTAX:

```
UPDATE table_name
SET column1 = value1, column2 = value2, ...
WHERE condition;
```

SQL Cheat Sheets

DELETE

The `DELETE` statement is used to delete existing records in a table.

SYNTAX:
```
DELETE FROM table_name WHERE condition;
```

SELECT TOP

The `SELECT TOP` clause is used to limit the number of records to return.

SYNTAX:
```
SELECT TOP number|percent column_name(s)
FROM table_name
WHERE condition;
```

MIN

The `MIN()` function returns the smallest value of the selected column.

SYNTAX:
```
SELECT MIN(column_name)
FROM table_name
WHERE condition;
```

MAX

The `MAX()` function returns the largest value of the selected column.

SYNTAX:
```
SELECT MAX(column_name)
FROM table_name
WHERE condition;
```

COUNT

The `COUNT()` function returns the number of rows that matches a specified criterion.

SYNTAX:
```
SELECT COUNT([DISTINCT] column_name | *)
FROM table_name
WHERE condition;
```

SUM

The `SUM()` function is used to calculate the total sum of values within a numeric column.

SYNTAX:
```
SELECT SUM(column_name)
FROM table_name
WHERE condition;
```

AVG

The `AVG()` function returns the average value of a numeric column.

SYNTAX:
```
SELECT AVG(column_name)
FROM table_name
WHERE condition;
```

LIKE

The `LIKE` operator is used in a `WHERE` clause to search for a specified pattern within a column's text data.

SYNTAX:
```
SELECT column1, column2, ...
FROM table_name
WHERE columnN LIKE pattern;
```

IN

The `IN` operator functions as a shorthand for multiple `OR` conditions

SYNTAX:
```
SELECT column_name(s)
FROM table_name
WHERE column_name IN (value1, value2, ...);
```

SQL Cheat Sheets

BETWEEN

The `BETWEEN` operator is used in the `WHERE` clause to select values within a specified range.

SYNTAX:

```
SELECT column_name(s)
FROM table_name
WHERE column_name BETWEEN value1 AND value2;
```

ALIASES

An alias is created with the `AS` keyword, and is often used to make a column name more readable.

SYNTAX:

```
SELECT column_name AS alias_name
FROM table_name;
(OR)
SELECT column_name(s)
FROM table_name AS alias_name;
```

INNER JOIN

The `INNER JOIN` returns only rows that have **matching values** in both tables.

SYNTAX:

```
SELECT column_name(s)
FROM table1
INNER JOIN table2
ON table1.column_name = table2.column_name;
```

LEFT JOIN

The `LEFT JOIN` returns **all rows from the left table** (table1), and only the matched rows from the right table (table2).

SYNTAX:

```
SELECT column_name(s)
FROM table1
LEFT JOIN table2
ON table1.column_name = table2.column_name;
```

RIGHT JOIN

The `RIGHT JOIN` returns **all rows from the right table** (table2), and only the matched rows from the left table (table1).

SYNTAX:

```
SELECT column_name(s)
FROM table1
RIGHT JOIN table2
ON table1.column_name = table2.column_name;
```

FULL JOIN

The `FULL JOIN` returns **all rows** when there is a match in either the left or right table.

SYNTAX:

```
SELECT column_name(s)
FROM table1
FULL JOIN table2
ON table1.column_name = table2.column_name
WHERE condition;
```

SELF JOIN

A self join is a regular join, but the table is joined with itself.

SYNTAX:

```
SELECT column_name(s)
FROM table1 T1, table1 T2
WHERE condition;
```

UNION

The `UNION` operator is used to combine the result-set of two or more `SELECT` statements, does not include any duplicates..

SYNTAX:

```
SELECT column_name(s) FROM table1
UNION
SELECT column_name(s) FROM table2;
```

UNION ALL

The `UNION ALL` operator is used to combine the result-set of two or more `SELECT` statements, including any duplicates..

SYNTAX:

```
SELECT column_name(s) FROM table1
UNION ALL
SELECT column_name(s) FROM table2;
```

SQL Cheat Sheets

GROUP BY

The `GROUP BY` statement is used to group rows that have same values into summary rows.
Used in conjunction with aggregate functions.

SYNTAX:

```
SELECT column1, aggregate_function(column2), ...
FROM table_name
WHERE condition
GROUP BY column1, column3
ORDER BY column_name;
```

HAVING

The `HAVING` clause is used to filter the results of a `GROUP BY` query based on aggregate functions.

SYNTAX:

```
SELECT column1, aggregate_function(column2),...
FROM table_name
WHERE condition
GROUP BY column1, column3
HAVING condition
ORDER BY column_name;
```

EXISTS

The `EXISTS` operator is used in a `WHERE` clause to check whether a subquery returns any rows.

SYNTAX:

```
SELECT column_name(s)
FROM table1
WHERE EXISTS (subquery);
```

ANY

The `ANY` operator is used to compare a value to every value returned by a subquery.

SYNTAX:

```
SELECT column_name(s)
FROM table1
WHERE column_name operator ANY (subquery);
```

ALL

The `ALL` operator is used to compare a value to every value returned by a subquery.

SYNTAX:

```
SELECT column_name(s)
FROM table1
WHERE column_name operator ALL (subquery);
```

SELECT INTO

The `SELECT INTO` statement is used to create a new table and fill it with data from an existing table.

SYNTAX:

```
SELECT * INTO newtable [IN external_db]
FROM sourcetable
WHERE condition;
```

INSERT INTO SELECT

The `INSERT INTO SELECT` statement is used to copy data from an existing table and insert it into another existing table.

SYNTAX:

```
INSERT INTO target_table
SELECT * FROM source_table
WHERE condition;
```

CASE

The `CASE` expression is used to define different results based on specified conditions in an SQL statement. It stops on first match.

SYNTAX:

```
CASE
  WHEN condition1 THEN result1
  WHEN condition2 THEN result2
  WHEN conditionN THEN resultN
  ELSE default_result
END;
```

FOCUS

01

PRODUCT:
Introduction to Product Management

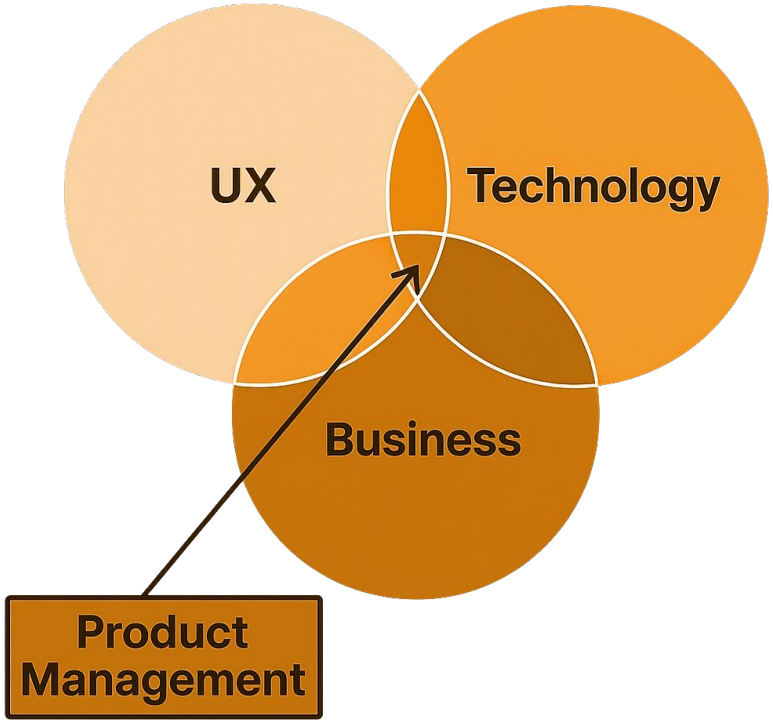


What is a PM ?

- A **Product Manager (PM)** ensures that a team ships a great product owning everything from vision to execution.
- Often called the **mini-CEO** of the product, a PM sets direction, defines success, and makes key decisions. But unlike a CEO, PMs have **no direct authority**. They lead by **influencing** engineers, designers, and stakeholders through clarity, research, and collaboration.
- PMs sit at the **intersection of tech, business, and design**, acting as the **voice of the customer**. They translate user needs into product goals, shape features, and ensure cohesive, usable solutions.
- It’s a **collaborative role**, working across engineering, design, QA, marketing, support, and leadership knowing when to involve others and filling gaps when needed.

What does a PM do?

A Project Manager is responsible for a project's planning, procurement, execution and completion. The project manager is in charge of the entire project and handles everything involved, such as the project scope, the project team, and the resources assigned to the project.



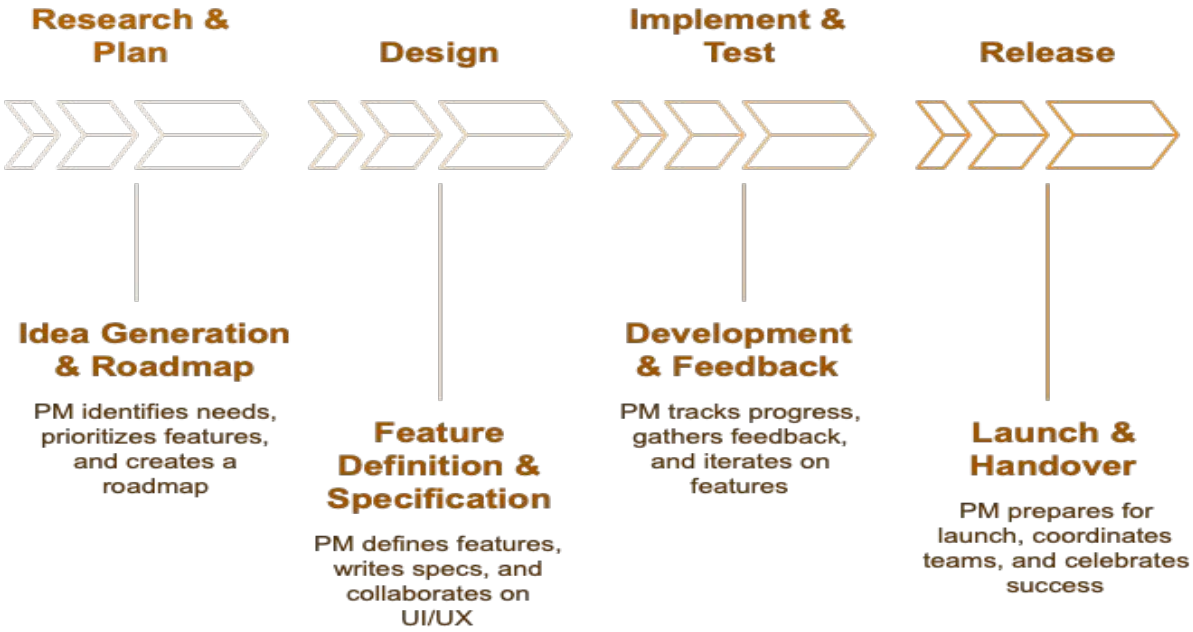
What does a PM do?

The tasks of a Product Manager typically include:

- Planning what work needs to be done, when and who's going to do it,
- Looking at the risks involved in a particular project and managing these risks,
- Making sure the work is done to a suitable standard,
- Motivating the team of people involved in the project,
- Coordinating work done by different people,
- Making sure the project is running on time and to budget,
- Dealing with changes to the project as and when necessary, and
- Making sure the project delivers the expected outcomes and benefits.

What is PM Trying to Solve ?

The day to day work of a PM varies across the product lifecycle. At the start, you're figuring out *what* to build. In the middle, you're ensuring progress. Toward the end, you're preparing for launch.



What is PM Trying to Solve ?

Research & Planning

- Involves identifying *what to build next* — sourced from user feedback, sales, marketing, tech trends, or big-picture vision.
- PMs define product goals, create a roadmap, and prioritize features based on user needs, business value, and team input.
- PMs also align with stakeholders (top-down in some orgs like Microsoft/Amazon, bottom-up in orgs like Google/Facebook).
- Success criteria are defined early through OKRs and measurable goals.

Design

- Once direction is set, PMs define features and functionality.
- On some teams, PMs write detailed specs; on others, they co-create with designers using whiteboards or mock-ups.
- Role can vary from deeply involved in UX to minimal input if design-led (e.g. Apple).
- Important to clarify the design/PM split during interviews.

What is PM Trying to Solve ?

Implementation & Testing

- PMs track progress, unblock teams, and adjust scope as needed.
- Help resolve dependencies and reprioritize features when needed.
- Actively collect early feedback via:
  - **Dogfooding** (using your own product internally)
  - **Usability testing** (observing real users try prototypes)
  - **Experiments/A-B testing** (measuring impact through data)
- Based on feedback, iterate while managing trade-offs and bugs to keep the product on track for release.

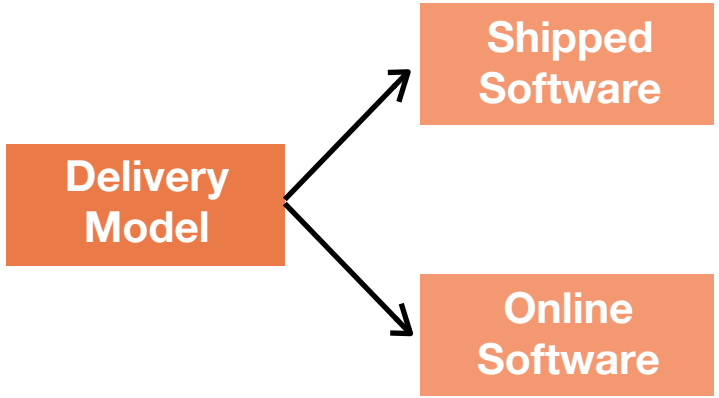
Release

- Coordinate final launch checklist and stakeholder approvals.
- Ensure customer support, ops, or manufacturing teams are ready.
- Handle last-minute issues and lead the go-live.
- After launch, share results, celebrate with the team, and gather data for iteration or handoff.

Types of Products a PM Might Encounter & How They Affect the Role

By Delivery Model

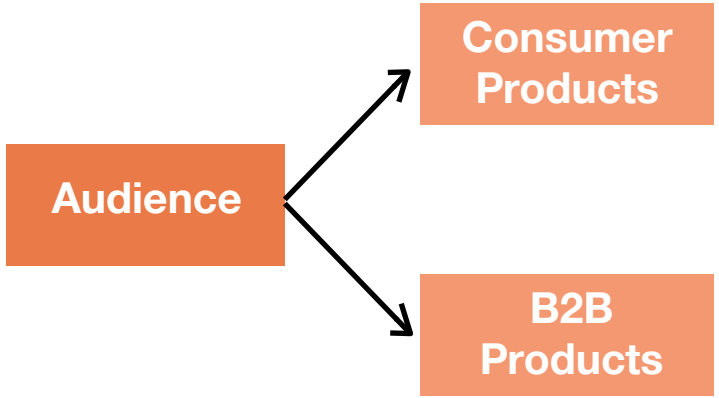
- **Shipped Software**
  (e.g. mobile apps, DVD software)
  → Requires precision. Updates are hard post-launch, so planning, specs, and testing are critical.
  → PMs focus more on project management, timelines, and cross-team coordination.
- **Online Software**
  (e.g. SaaS, websites)
  → Fast-paced and iterative. Frequent launches, A/B testing, and real-time metrics guide decisions.
  → PMs must be data-savvy and comfortable with rapid experimentation.



Types of Products a PM Might Encounter & How They Affect the Role

By Audience

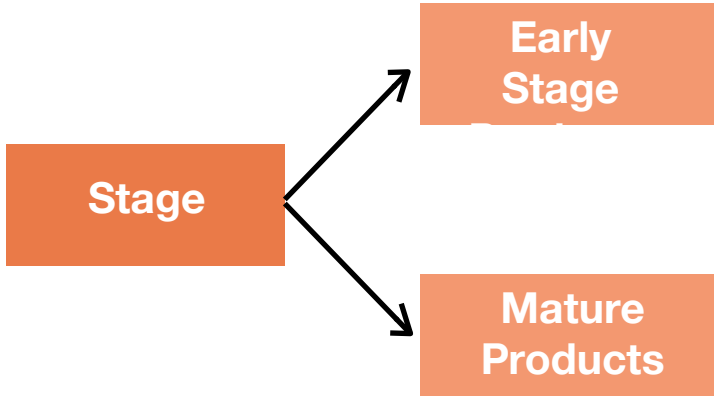
- **Consumer Products**
  (e.g. social media, music apps)
  → PMs guide rather than dictate , everyone has opinions. Success is metric-driven (engagement, retention).
  → Great for PMs with strong product sense and storytelling skills.
- **B2B Products**
  (e.g. ad platforms, productivity tools)
  → Users are professionals; PMs bridge customer needs and business outcomes.
  → PMs focus on research, stakeholder input, and revenue-driven roadmaps.



Types of Products a PM Might Encounter & How They Affect the Role

By Stage

- **Early-Stage Products**
  (e.g. MVPs, new launches)
  → PMs move fast, ship lean, and learn quickly. The focus is finding product-market fit.
  → Ideal for PMs who like ambiguity, speed, and 0→1 thinking.
- **Mature Products**
  (e.g. market leaders)
  → Focus shifts to optimization, retention, and scalable improvements.
  → PMs need to balance innovation with stability, and avoid getting stuck in incrementalism.



Types of Product Managers : Getting Started in PM

1. Associate Product Manager (APM)

- An Associate Product Manager is an entry-level role and a great way to break into Product. Working in an assistant-like position with other Product Managers, an APM is given a variety of smaller tasks to assist with product development.
- Some big companies like Facebook and Google run well-known APM programs specifically to help people break into Product.

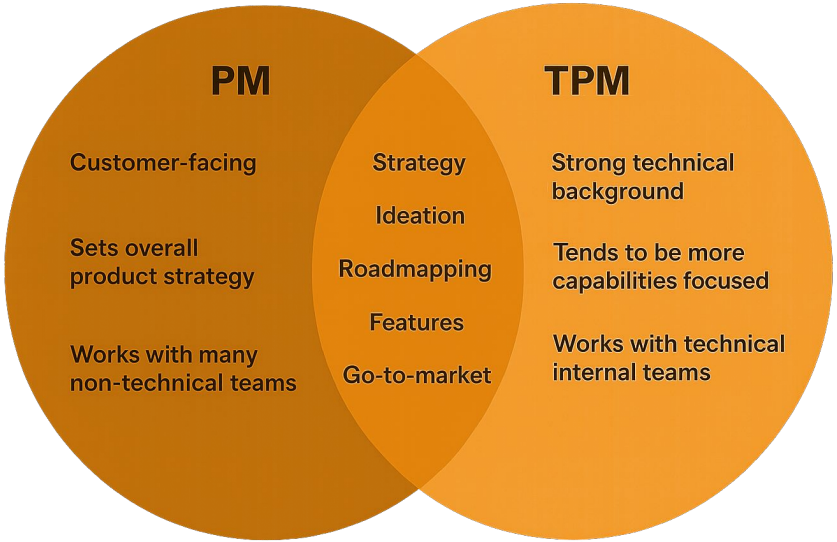
2. Product Manager

- Product Managers have full ownership of a product or a significant feature within a larger product. They are at the core of the product management positions, driving the development, launch, and continuous improvement of their products.
- Product Managers can usually expect to work alongside other team members and not above them. The key to wielding their ‘power’ is to influence without authority. Earning the trust of those on the product team is the best way to get things done.

Types of Product Managers : Specializing within PM

3. Technical Product Manager

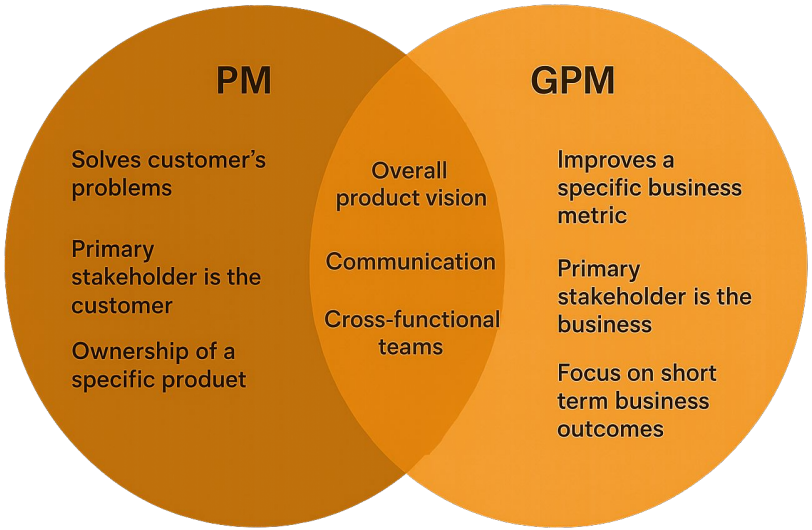
- Technical Product Managers (TPMs) typically have a background in engineering or software development and manage products that require deeper technical expertise such as APIs, infrastructure tools, or data platforms.
- They focus on defining technical requirements, working closely with engineering teams, and bridging the gap between complex systems and business value. While their responsibilities largely mirror those of a general PM, TPMs often take a more hands on role in technical execution, which may mean less involvement in areas like marketing or customer outreach.



Types of Product Managers : Specializing within PM

4. Growth Product Manager

- Growth PMs focus less on the life of a product and more on driving user acquisition, retention, and monetization. They dedicate their time to improving certain business metrics. While all Product Managers keep their goals in line with those of the business as a whole, this will be a Growth Product Manager's primary focus.
- They employ data-driven approaches to optimize the product for market growth and user engagement. By owning a metric rather than an entire product, a Growth Product Manager designs and runs a series of experiments, working on a micro rather than a macro level.



Types of Product Managers : Specializing within PM

5. Digital Product Manager

- It operate exclusively within the digital realm, dealing with software applications, online platforms, and digital services. While not as techy as Technical PMs, they have a solid understanding of software development processes and the latest technological trends.
- DPMs play a crucial role in identifying opportunities and Product Discovery. They make sure the Product Vision is clearly defined and oversee a digital product throughout its lifecycle.

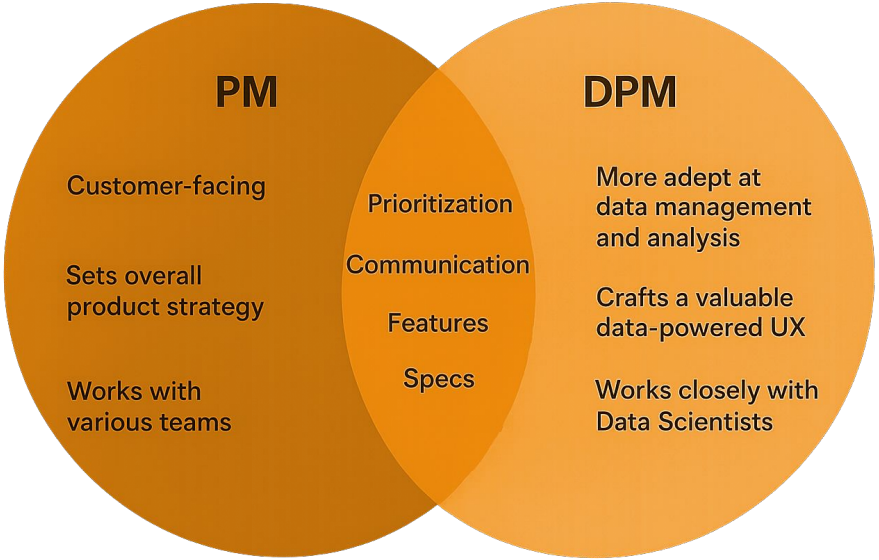
6. UX/Product Design Manager

- UX/Product Design Managers focus on the user experience and design aspects of product management. They ensure that products are not only functional but also intuitive and engaging for users.
- Product Design managers lead design teams, define UX/UI standards, conduct user research, and collaborate with product managers and engineers to integrate user-centered design into the product development process.

Types of Product Managers : Specializing within PM

7. Data Product Manager

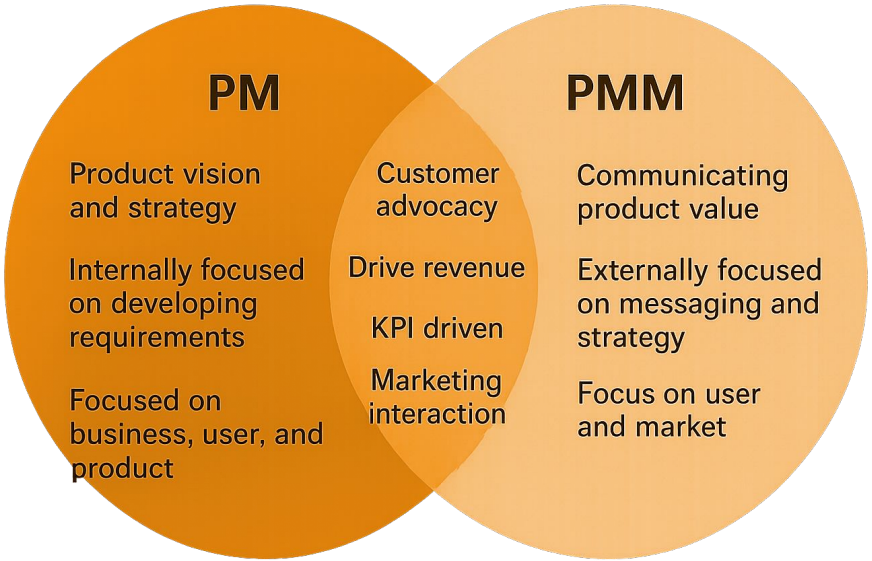
- A Data Product Manager will be more adept at data management and analysis. They manage data-driven products or features, such as analytics platforms, AI models, or business intelligence tools. Data PMs spend their days ensuring data quality and accessibility, and collaborating with data scientists and analysts to turn data into actionable insights for users.
- Data Product Management is a great career choice for anyone who loves working with numbers. Neither data nor product is disappearing any time soon!



Types of Product Managers : Specializing within PM

8. Product Marketing Manager

- PMMs bridge the gap between product development and marketing strategy. They play a key role in positioning, launching, and driving demand for the product. Unlike core PMs, PMMs are less involved in product development and spend limited time with engineers on bugs or roadmaps.
- Their focus lies in crafting go-to-market strategies, creating case studies, writing content, organizing press briefings, and leading the marketing team. As the voice of the customer, they conduct research and run focus groups to shape messaging and product-market fit. It's an ideal role for marketing professionals looking to break into product roles.



Types of Product Managers : Specializing within PM

9. AI Product Manager

- AI Product Managers focus on products powered by artificial intelligence and machine learning. They define the vision for AI-driven features, identify AI/ML opportunities, set product requirements, and collaborate with data scientists and engineers—while ensuring ethical AI use.
- As AI becomes central to many industries, these PMs are key to shaping the future of product development. With the rise of AI products, more PMs are gaining AI proficiency to build solutions using LLMs and ML.

10. Staff Product Manager

- Staff Product Managers are seasoned professionals who contribute significantly to product strategy and mentorship within product teams. They often work horizontally to handle complex problems and serve as advisors. Often individual contributors, the decisions of Staff PMs will guide investments of money and time, higher risks but also higher rewards, and product impact.

Types of Product Managers : Specializing within PM

11. Senior Product Manager

- A **Senior Product Manager** oversees a group of products and may lead other Product Managers. At this level, the focus shifts to **long-term strategy**, setting a vision for where the team should be in the next 3–5 years and aligning directors, stakeholders, and teams around that vision.
- The role becomes more **strategic and stakeholder-facing**, with less hands-on involvement in day-to-day product development. Senior PMs also play a key role in **recruitment and team building**.

12. Product Lead

- Product Leads are primarily responsible for new products within a company. When large companies launch something brand new, they need experienced and trustworthy professionals to lead the way.
- They spend most of their time liaising between different departments and communicating with senior management. Their main responsibility is to ensure that the final product is delivered on time and within budget, which means schedules and targets are their lifeblood!

Types of Product Managers : Specializing within PM

13. Group Product Manager

- A step above Lead Product Manager, Group Product Managers hold senior leadership roles, often overseeing entire product lines or categories. Some Group Product Managers prefer to lean more towards individual contribution; conducting research, focusing on strategy, and generally being more hands-on with development. Others prefer to have a zoomed-out approach and focus more on managing the teams.

14. Director of Product Management

- Directors of Product Management are high-level executives who oversee all product management activities within an organization or business unit. It is a cross-functional leadership role that involves overseeing the work of Senior Product Managers and the teams they lead. They identify potential areas for the growth of a product within its market and have control over recruitment.

Types of Product Managers : Specializing within PM

15. Principal Product Manager

- Principal Product Managers are highly experienced leaders, often focusing on strategic innovation and guiding the company's most critical product initiatives. They are individual contributors who are responsible for leading strategic product initiatives related to top-priority products.

16. Head of Product or VP of Product

- Head of Product and VP of Product are senior roles responsible for leading the entire product management function. They define the long-term product vision and strategy, ensuring alignment with business goals. These leaders coordinate across teams to drive execution and deliver impactful results. They play a crucial role in shaping the company’s product direction, team structure, and overall product culture. By guiding product managers and influencing cross-functional efforts, they ensure that the product portfolio evolves effectively and competitively in the market.

Types of Product Managers : Executive Level PM

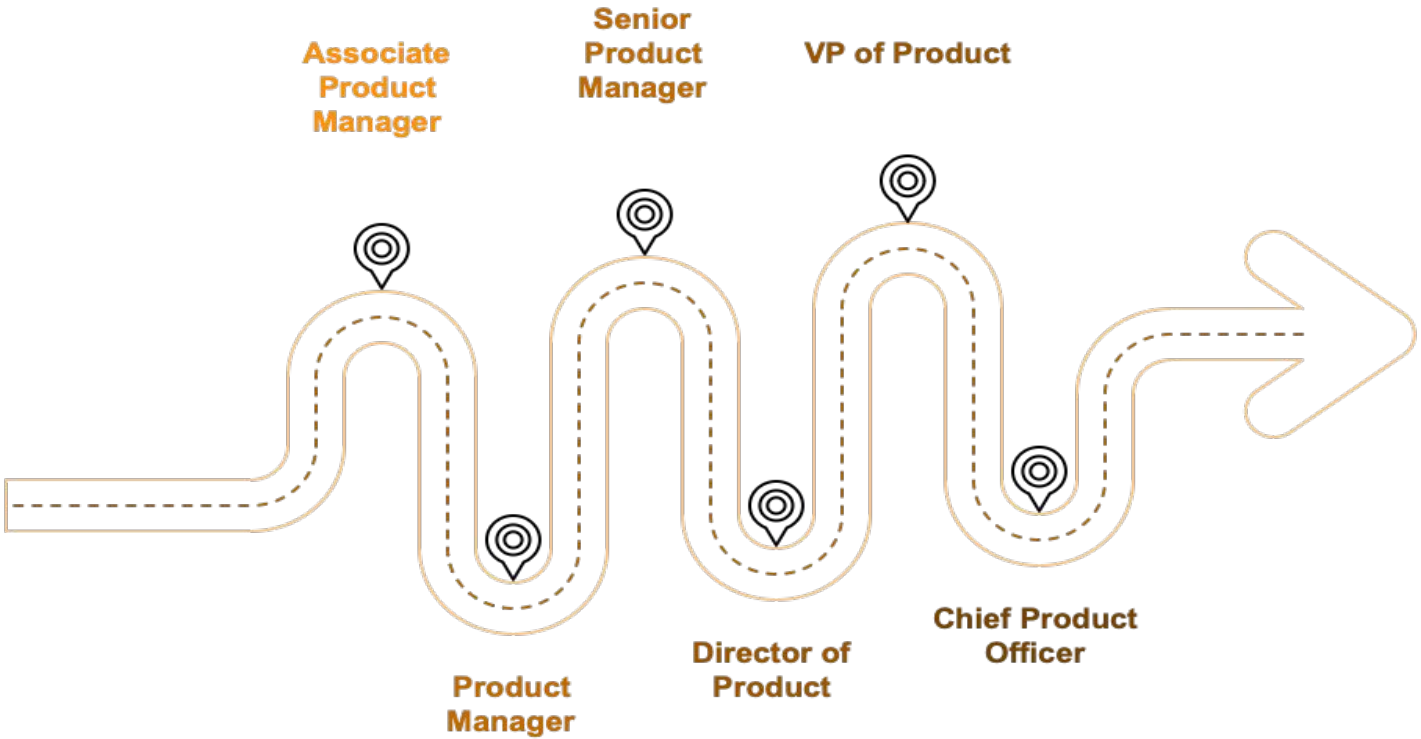
17. Senior Vice President (SVP) of Product

- SVPs of Product Management hold an elevated position within the company, often involved in broader business strategy beyond product management. They are responsible for setting strategic priorities for the product and the business, driving cross-functional alignment, and contributing to the executive management of the company.

18. Chief Product Officer (CPO)

- The Chief Product Officer is the top executive role within the product management hierarchy, responsible for the overall product vision, strategy, and execution across all products. CPOs hold a seat at the decision-making table, representing product and product-led growth amongst other key stakeholders.

Product Manager Career Path



Product Manager Career Path

1. Associate Product Manager (APM)

An entry level PM role, ideal for gaining hands on experience in product development and strategy by working closely with senior PMs and cross-functional teams.

Key Responsibilities

- Prioritize tasks based on defined guidelines
- Collaborate with UX, engineering, and product teams
- Align customer needs with business goals
- Communicate product status to stakeholders

2. Product Manager (PM)

Central to driving the product lifecycle from planning to execution.

Key Responsibilities

- Understand market and customer sentiment
- Analyze competition and spot growth opportunities
- Maintain clear communication across teams

Product Manager Career Path

3. Senior Product Manager (Sr. PM)

A seasoned product leader managing complex portfolios and strategic planning.

Key Responsibilities

- Own product strategy aligned with company goals
- Lead and mentor product teams
- Define and communicate product roadmaps
- Collaborate cross-functionally
- Monitor performance and iterate

4. Director of Product

Oversees product teams and connects product vision with business strategy.

Key Responsibilities

- Define and align overall product strategy
- Lead and scale PM teams
- Own roadmap prioritization
- Drive cross-functional alignment
- Lead market analysis and innovation

5. VP of Product

A senior executive shaping company-wide product strategy in collaboration with leadership.

Key Responsibilities

- Set long-term product vision aligned with business goals
- Build and mentor high-performing PM teams
- Conduct market analysis to identify growth opportunities
- Own strategic product roadmap
- Ensure seamless cross-functional execution

6. Chief Product Officer (CPO)

The highest-ranking product executive, responsible for overall product vision and alignment with company goals.

Key Responsibilities

- Define company-wide product strategy
- Oversee end-to-end product development
- Lead market and competitive analysis
- Manage product lifecycle decisions
- Communicate with executives, investors, and customers

Transitioning from Engineer to Product Manager

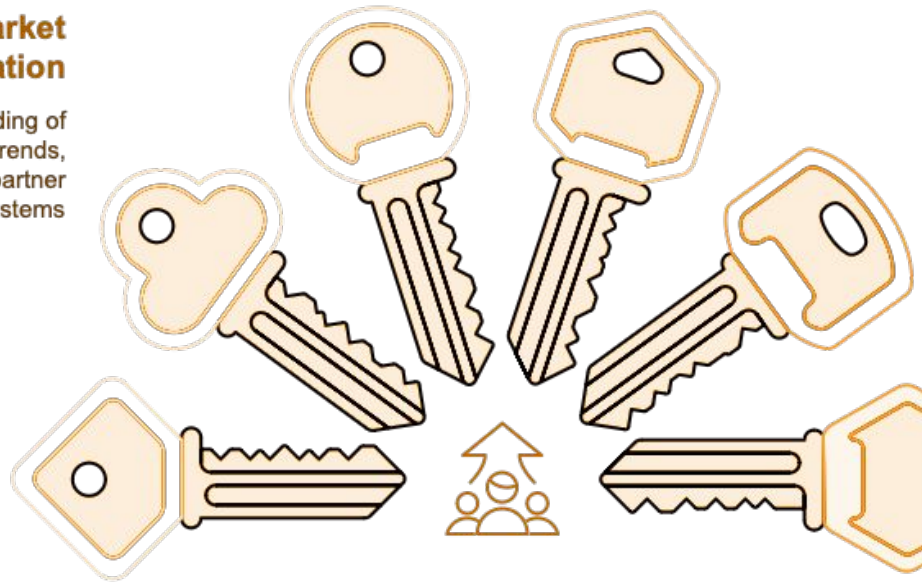
Engineers are well-positioned to transition into product management because they understand what it takes to build a product and the impact of trade-offs. Many have worked with PMs they admire or wish to improve upon.

- ***Customer Focus:*** The most critical skill to develop is customer focus. This means deeply understanding the end user's needs, not just generating ideas. Engineers can develop this by joining customer calls, reading support tickets, and conducting internal user studies. Writing user scenarios and practicing customer-centric communication helps build this mindset. Speak in terms of user benefits, not just technical features.
- ***Think Big:*** Strategic thinking is vital. Engineers often focus on feasibility, but PMs must start with a big vision. Think about how a product could change the world, tie benefits to human needs, and write press-release-style plans. Balance practicality with visionary leadership.
- ***Master Persuasive Communication:*** PMs must be credible and persuasive. While data matters, so does conviction and the ability to rally a team. Build trust by being right often, but also know when to rely on data and when to communicate belief and vision.

Transitioning from Engineer to Product Manager

- ***Expect Role Shifts:*** PM work is less tangible. Instead of coding, you achieve wins through coordination, influence, and decision-making. You may become a focus of criticism and need to handle feedback constructively. Time constraints mean you won't get to perfect everything—prioritization is key.
- ***Find Opportunities Within Your Team:*** Many engineers shift into PM roles within their existing teams. Let your interest be known and talk to potential hiring managers. Review your feedback for signs of PM aptitude or gaps. Start doing PM work informally—help with specs, product decisions, or coordination.
- ***Leverage Your Engineering Background:*** Apply to specialized PM roles related to your technical expertise, such as platform or developer tools. Your domain knowledge can make you stand out.
- ***Consider Business School (Optional):*** An MBA can help build business acumen, open networking doors, and connect you with PM recruiters. But the cost is high and the ROI uncertain if PM is your only goal. If you're already deeply in engineering and want a fresh start, an MBA might be worth it. Ensure the PM roles you're targeting align with your strengths and goals.

What Companies look for - The PM of the Future



Business Acumen

Comfort with strategy, pricing, GTM plans, and tracking KPIs and ROI

Technical Depth

Understanding tech trends, architecture, roadmaps, and product lifecycle

Market Orientation

Deep understanding of market trends, competition, and partner ecosystems

Soft Skills

Ability to lead teams, communicate across functions, and influence change

Customer Experience Grounding

Ability to design seamless, user-centric journeys across touchpoints

Organizational Enablers

Fostering innovation by enabling teams, driving experimentation, and scaling processes

What Companies look for - The PM of the Future

- The role of a **Product Manager** is undergoing a significant transformation, from coordinating feature delivery to becoming a **strategic connector** between **customer needs** and **business value**. In a world driven by rapid technological innovation and increasingly informed consumers, PMs are expected to bring a strong understanding of **data**, **emerging technologies**, and **user behavior** into every product decision. The future PM will not only define what to build but also why it matters, backing decisions with **real insights** and a **deep awareness** of **evolving market dynamics.**
- As organizations place greater emphasis on **customer-centricity**, PMs are now spending a substantial part of their time on external-facing activities, engaging with **customers**, **aligning with C-suite leadership**, and **shaping long-term vision**. This expanded scope reflects the growing influence of the role, prompting leading global universities to introduce Product Management as a formal specialization.
- As a result, the PM of tomorrow is expected to be a **cross-functional leader**, **blending empathy**, **strategic thinking**, and **technical fluency** to drive impact at scale.

Top Myths about Product Manager

1. Product Managers are Project Managers

Project managers handle coordination and timelines. PMs identify problems, decide what to solve, and ensure the team builds effective solutions. Product sense is key.

2. Product Managers are in Marketing

Though some companies use the title in marketing, PMs at tech firms (Google, Amazon, etc.) sit within engineering. Marketing gets users in; PMs define what happens after.

3. You can’t become a PM right out of college

Despite the "manager" title, many companies (Google, Microsoft) hire PMs directly from college. Passion, intellect, and customer focus can outweigh experience.

4. Product Managers just write specs

Specs are a communication tool, not the job itself. PMs ensure successful execution , through whiteboards, conversations, and follow-ups.

5. Product Managers just set up meetings

PMs synthesize stakeholder input and make decisions. They reduce meetings by representing team interests and ensuring clear communication.

6. PMs should build exactly what customers ask for

PMs dig deeper into customer behavior to uncover hidden needs. Like Oxo's angled measuring cup — insight comes from observation, not just requests.

7. Product Managers set the dates

Engineers set realistic timelines. PMs negotiate scope and prioritize to meet deadlines — not force faster coding.

8. Product Managers are the boss

PMs lead without authority. They earn trust through clarity, research, and alignment — not control. Designers and engineers own their domains.

FOCUS

02

PRODUCT:
Building The PM Mindset



The PM Mindset

The essence of the PM mindset is rooted in a deep commitment to understanding and serving the customer. Great PMs go beyond simply reacting to user feedback; they proactively seek out hidden needs and champion the customer's voice throughout the organization. Their focus extends beyond individual features to the entire user journey, ensuring that every touchpoint adds value and delivers a seamless experience.

PMs are agile and adaptable, viewing change as an opportunity to innovate and stay ahead in dynamic markets. Balancing strategic vision with tactical execution, PMs set long-term goals while breaking them into actionable steps, skillfully prioritizing and guiding their teams to deliver meaningful outcomes. This holistic, customer-centric, and adaptable approach defines the core of the PM mindset.



Key Principles of the PM Mindset

- Insatiable Curiosity:
  PMs are lifelong learners. They constantly ask questions, investigate new trends, and seek to understand the “why” behind user behaviors and market shifts. This curiosity drives innovation and helps uncover hidden opportunities.
- Deep, Analytical Thinking:
  Product managers analyze problems from multiple angles, using data and structured frameworks to inform decisions. They break down complex issues into manageable parts and look for root causes rather than surface symptoms.
- Incremental Improvement:
  Rather than aiming for perfection from the start, PMs focus on continuous, incremental progress. They believe in shipping early, learning fast, and iterating based on feedback. This approach accelerates learning and delivers value sooner.
- Empathy for Users and Stakeholders:
  PMs build strong relationships with both users and internal teams. They listen actively, understand different perspectives, and strive to align diverse interests towards common goals. Empathy fosters trust and leads to better products.
- Data-Informed Decision Making:
  While intuition and experience are valuable, PMs ground their decisions in evidence. They define and track key metrics, run experiments, and use both quantitative and qualitative data to guide prioritization and strategy.

Five Stages of User Journey

The user journey in product management spans five key stages: Discovery, where users first encounter the product through targeted outreach and brand presence; Consideration, as they evaluate features and compare alternatives; Acquisition/Onboarding, involving seamless sign-up and guided introductions; Engagement, where users interact with core features and form habits; and Retention/Advocacy, focusing on long-term satisfaction and turning users into promoters. Product managers play a crucial role at each stage by understanding user motivations, removing friction, and continuously iterating based on feedback and data. Leading resources emphasize mapping user emotions and pain points throughout the journey, while platforms like Amplitude stress the importance of tracking engagement and retention metrics to optimize every touchpoint and foster lasting user relationships.



The JTBD Framework

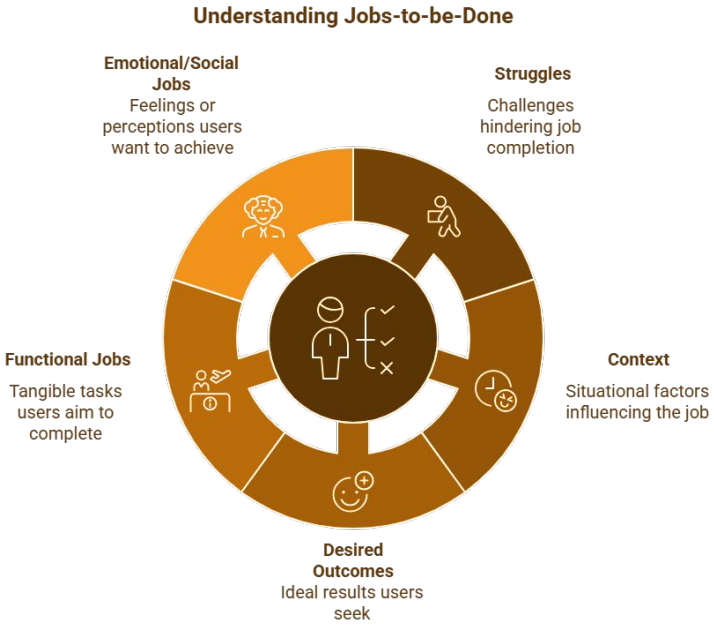
What?

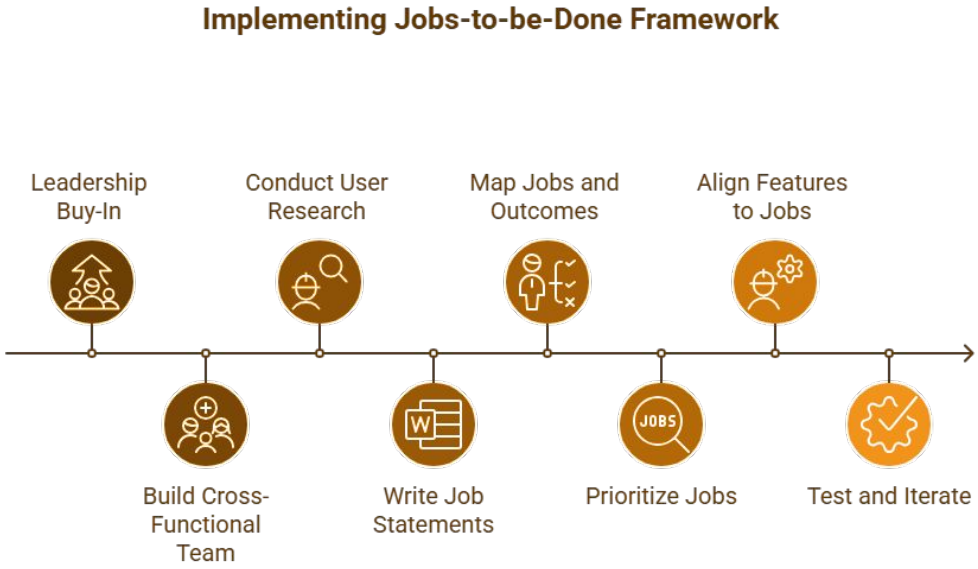
Jobs-to-be-Done (JTBD) is a product development and innovation framework focused on understanding the underlying "jobs" customers are trying to accomplish in their lives. Rather than segmenting markets by demographics or product features, JTBD asks: *What goal or problem is the customer "hiring" this product to solve?* This approach, popularized by Clayton Christensen, helps teams design products that align with real user motivations and needs, leading to more successful, customer-centric solutions.

Why?

- Customer-Centric: Shifts focus from features to outcomes users care about.
- Drives Innovation: Reveals unmet needs and opportunities for differentiation.
- Aligns Teams: Provides a common language for product, design, marketing, and support.
- Improves Product-Market Fit: Ensures solutions are built around real jobs, not assumptions.

The JTBD Framework





The JTBD Framework

Brownie Points

- Start Small: Pilot JTBD with one segment or product, then expand.
- Ask the Right Questions: Use prompts like “Help me ___,” “I need to ___,” or “Help me avoid ___” to uncover jobs.
- Look Beyond Functionality: Emotional and social jobs can be as important as functional ones.
- Integrate with Product Processes: Use JTBD insights to guide roadmap, marketing, and support.

Traditional Approach	JTBD Approach
Focuses on demographics or features	Focuses on user goals and outcomes
Asks “Who are our users?”	Asks “What are users trying to accomplish?”
Prioritizes features	Prioritizes user jobs and struggles
Risk of feature bloat	Drives purposeful, outcome-oriented design

The CIRCLES Framework

What?

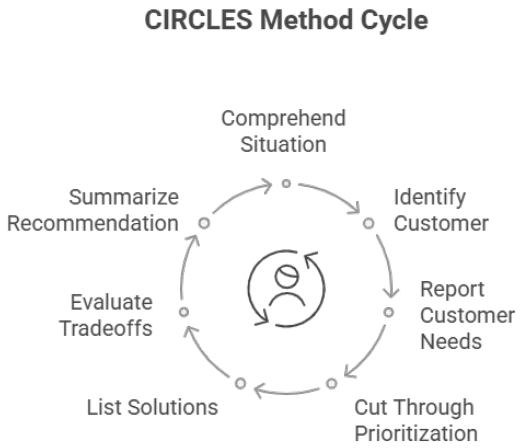
The CIRCLES Framework is a structured, step-by-step method created by Lewis C. Lin to help product managers tackle product design and problem-solving challenges in a comprehensive, user-focused, and organized way. Widely used in both real-world product management and PM interviews, CIRCLES ensures that no critical aspect of problem-solving is overlooked and that solutions are deeply rooted in user needs and business goals.

Why?

- Comprehensive: Ensures all critical aspects of product design and problem-solving are considered.
- User-Centric: Keeps the focus on real user needs rather than jumping to solutions.
- Prioritization: Helps teams decide what matters most and allocate resources effectively.
- Structured Communication: Improves clarity and persuasiveness in both interviews and stakeholder presentations.
- Reduces Risk: Minimizes the chance of overlooking important factors or making hasty decisions.

The CIRCLES Framework

C – Comprehend the Situation
Understand the problem, context, objectives, and constraints.
Ask: What's the core challenge? What does success look like?
I – Identify the Customer
Pinpoint and segment your target users.
Ask: Who are they? What are their key characteristics?
R – Report Customer Needs
Define what users want to achieve and their pain points.
Ask: What problems are they facing? What outcomes do they seek?



C – Cut Through Prioritization
Rank needs by impact, urgency, and feasibility.
Ask: Which problems matter most right now?
L – List Solutions
Brainstorm a range of creative solutions.
Ask: What ideas address the top needs?
E – Evaluate Tradeoffs
Assess pros, cons, feasibility, and risks for each solution.
Ask: What are the benefits and downsides of each option?
S – Summarize Recommendation
Present and justify your chosen solution.
Ask: Why is this the best path forward?

The CIRCLES Framework

Example

Scenario: Improving the onboarding experience for a mobile banking app.

- Comprehend the Situation: Onboarding drop-off rate is high; goal is to increase activation.
- Identify the Customer: New users aged 25–40, tech-savvy but cautious with financial apps.
- Report Customer Needs: Users want a quick, secure, and reassuring sign-up process.
- Cut Through Prioritization: Prioritize reducing friction & increasing trust over adding new features.
- List Solutions: Biometric login, progress bar, in-app chat support, video walkthrough.
- Evaluate Tradeoffs: Biometric login increases security but requires device compatibility; video walkthroughs are engaging but may slow down the process.
- Summarize Recommendation: Implement biometric login and progress bar for immediate impact, pilot video walkthroughs for future iterations.

Brownie Points

- Practice applying the framework to real and hypothetical product challenges.
- Involve cross-functional teams for richer brainstorming and evaluation.
- Use CIRCLES in interviews to structure your answers and demonstrate systematic thinking.

The North Star Metric

What?

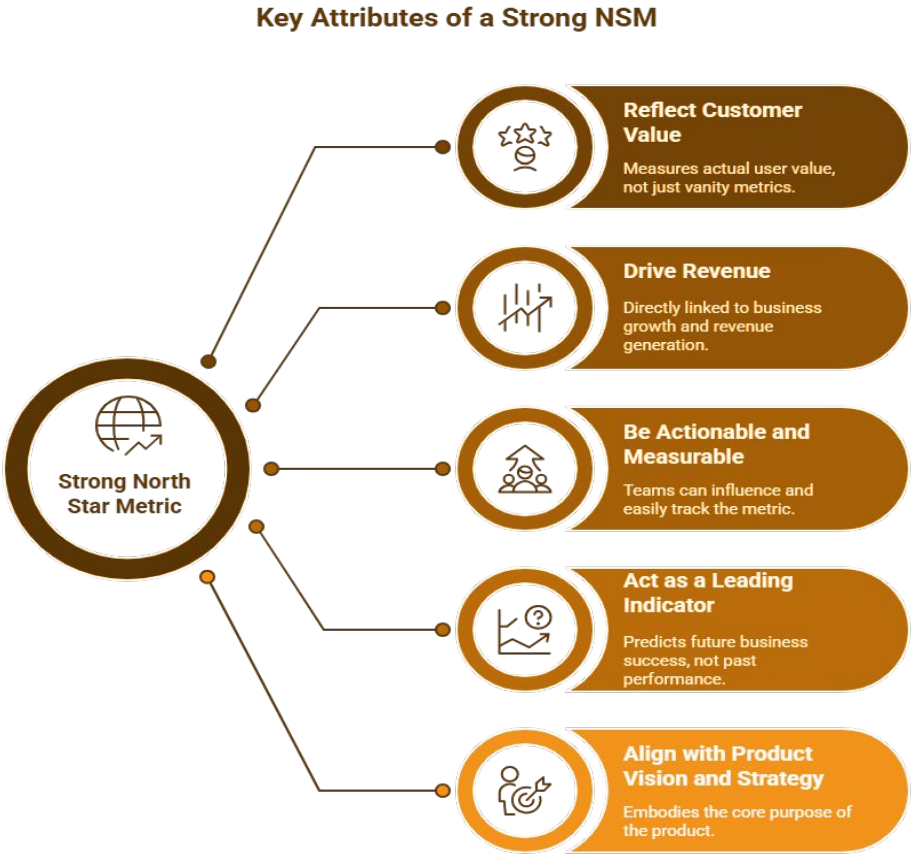
A North Star Metric (NSM) is the single most important metric that best captures the core value your product delivers to customers and serves as the guiding compass for your product strategy. This metric aligns your entire organization—across product, marketing, sales, and support—around a shared goal, ensuring that every initiative, feature, and experiment is focused on driving sustainable business growth and customer value.

Why?

- Alignment: It unites all teams around a clear, outcome-oriented goal, reducing wasted effort and conflicting priorities.
- Clarity: It provides a simple, measurable way to track progress and communicate impact to the organization.
- Accountability: It shifts focus from outputs (features shipped) to outcomes (customer and business impact).
- Focus: It helps prioritize initiatives that directly contribute to long-term growth.

The North Star Metric

Poor NSM Example: *“Number of registered users” (doesn’t reflect value or engagement)*
Good NSM Example: *“Number of successful orders delivered” (directly tied to customer value and business success).*



Defining Your North Star Metric

Step 1: Identify Your Core Value Proposition

Determine what action or outcome best represents the value your product delivers. Ask: What do our best users do that signals they're getting value?

Step 2: Map Customer Engagement to Business Growth

Analyze which user behaviors correlate with retention and revenue. Your NSM should increase as your business grows.

Step 3: Ensure It's Measurable and Actionable

Choose a metric you can track in real time and that teams can influence through product changes or campaigns.

Step 4: Choose the Right Type for Your Business Model

- E-commerce: Number of purchases per customer
- SaaS: Number of active subscriptions or messages sent
- Marketplaces: Completed transactions or bookings
- Media/Content: Total watch time or articles read

Step 5: Validate and Refine with Data

Use historical data and experiments to ensure your NSM predicts growth and customer satisfaction. Adjust if needed as your product evolves.

Using Your North Star Metric

- Align teams: Make the NSM visible and part of every team's objectives.
- Guide decisions: Prioritize projects and features that move the NSM.
- Set supporting metrics: Use secondary metrics for individual teams (e.g., activation rate for onboarding).
- Review regularly: Reassess your NSM as your product and market evolve.
- Experiment: Test changes and measure their impact on the NSM to drive continuous improvement.

Avoiding Pitfalls

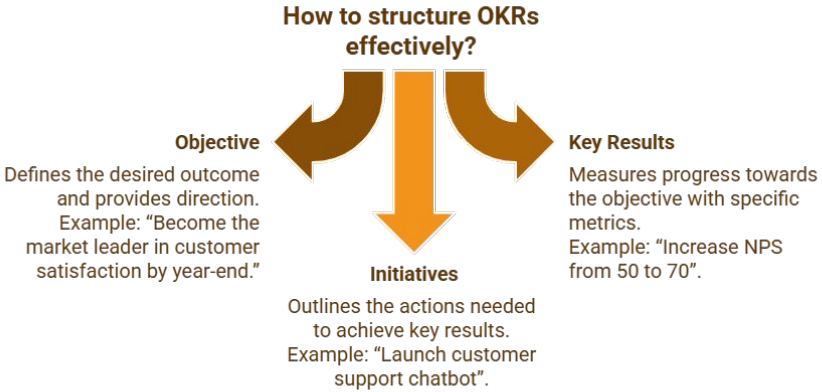
- Choosing vanity metrics (e.g., sign-ups) that don't reflect real value.
- Focusing on lagging indicators (e.g., revenue) instead of leading indicators.
- Failing to revisit and update the metric as your product grows or pivots.

The OKR Framework

Definition

OKR (Objectives and Key Results) is a widely adopted goal-setting framework that helps organizations, teams, and individuals align their efforts, prioritize work, and measure outcomes. The framework consists of three main components:

- Objective: A clear, ambitious, and qualitative goal that sets a direction.
- Key Results: Specific, measurable outcomes that indicate progress toward the objective.
- Initiatives: The projects and tasks undertaken to achieve the key results.



Effective OKRs

Setting Objectives

- Make objectives ambitious, inspiring, and aligned with company strategy.
- Limit to 3-5 objectives to maintain focus.
- Use clear, action-oriented language (e.g., “Launch,” “Achieve,” “Transform”).

Setting Key Results

- Make them outcome-focused, not activity-based.
- Ensure they are quantifiable and specific (use numbers, percentages, deadlines).
- Keep 3-5 key results per objective for clarity and focus.
- Key results should directly measure progress toward the objective.

Setting Initiatives

- Initiatives are the specific actions or projects that drive progress on key results.
- Should be within your control and clearly defined.
- If initiatives aren’t moving the needle, reassess and adapt.

Best Practices

- Alignment: Start with organizational OKRs and cascade to teams and individuals for alignment.
- Transparency: Make OKRs visible across the organization to foster accountability and collaboration.
- Regular Check-ins: Review progress frequently (weekly or bi-weekly), discuss blockers, and adjust as needed.
- Ambition & Realism: OKRs should stretch teams but remain achievable.
- Outcome over Output: Focus on results, not just activities.

Benefits

- Alignment towards goals: Ensures everyone is working toward common objectives.
- Clarity and focus: Helps prioritize what matters most.
- Transparency and accountability: Progress is visible and measurable.
- Agility: Enables quick corrections if progress stalls.
- Performance culture: Encourages teams to set and achieve ambitious targets.

The Kano Model

Definition

The Kano Model is a customer-centric framework that helps product managers prioritize features by categorizing them based on their impact on customer satisfaction and expectations. By understanding which features are essential, which drive delight, and which may even cause dissatisfaction, teams can make more strategic decisions about what to build next.

Usage

1. Identify customer needs by using surveys, interviews, and customer feedback.
2. Categorize feature and map each feature into one of the Kano categories based on customer input and analysis.
3. Prioritize Development:
    - First, ensure all Basic Needs are met.
    - Next, enhance Performance Needs to increase satisfaction.
    - Finally, add Excitement Needs to delight and differentiate.
    - Avoid investing in Indifferent or Reverse Needs.
4. Continuously gather feedback and track satisfaction to keep your product relevant and competitive, as customer expectations evolve over time.

Good Read - Medium Article

Category	Description	Working Principle	Example (Smartphone)
Basic Needs (Must-Have)	Essential features customers expect as a minimum requirement. Their absence causes dissatisfaction, but their presence doesn't increase satisfaction.	Meeting these needs prevents dissatisfaction, but exceeding them doesn't boost satisfaction.	Ability to make calls and send texts
Performance Needs (More is Better)	Features that have a linear relationship with satisfaction—the better they are, the happier the customer.	Improving these features increases satisfaction; their absence doesn't necessarily cause dissatisfaction.	Bigger screen, faster processor
Excitement Needs (Delighters)	Unexpected features that can greatly enhance satisfaction and set the product apart.	Their presence delights customers, but absence doesn't cause dissatisfaction.	Voice assistant (e.g., Siri, Google Assistant)
Indifferent Needs (Take It or Leave It)	Features that don't significantly impact satisfaction, whether present or not.	Customers are largely neutral; these features don't affect overall satisfaction.	Customization options for device color
Reverse Needs (Dissatisfaction When Present)	Features that can actually cause dissatisfaction if present, but their absence doesn't increase satisfaction.	These are often unexpected in a negative way and can turn customers off.	Pre-installed bloatware, intrusive ads

The MoSCoW Prioritization

Definition

MoSCoW is a clear, structured prioritization framework that helps teams categorize initiatives, features, or requirements into four groups:

- Must-have
- Should-have
- Could-have
- Won’t-have (this time)

It was originally developed to help Agile and software teams manage scope and expectations, but it’s now used across industries for project and product management.



The MoSCoW Prioritization

Example: For a new web app MVP:

- Must-have: User sign-up, login, password reset, core dashboard.
- Should-have: Account deletion, billing system, advanced reporting.
- Could-have: Mobile app version, theme customization.
- Won't-have: Social features, extensive integrations.

Strengths

- Clarity: Makes priorities explicit and easy for all stakeholders to understand.
- Expectation Management: Clearly communicates what will and won't be delivered, reducing scope creep.
- Adaptability: Works for both high-level roadmaps and detailed sprint planning.

Brownie Points

- Engage stakeholders early and align on definitions.
- Limit Must-haves to essentials to avoid overcommitment.
- Review and update regularly to maintain relevance in Agile environments.
- Use MoSCoW alongside objective ranking methods (e.g., weighted scoring) for even better results.

The RICE Prioritization

Definition

RICE is a data-driven prioritization framework designed to help product managers and teams objectively assess which projects, features, or initiatives should be tackled first. It stands for Reach, Impact, Confidence, and Effort—the four key factors used to score and rank ideas. Developed at Intercom, RICE helps teams make better-informed decisions, minimize personal bias, and clearly communicate the rationale behind prioritization choices.

Calculation

The RICE score is calculated using the following formula:

$$RICE\ Score = \frac{Reach \times Impact \times Confidence}{Effort}$$

Higher scores indicate higher priority.

This formula helps teams focus on initiatives that offer the most value for the least effort.

The RICE Prioritization

Factor	What It Measures	How to Estimate	Example
Reach	How many people or events will be affected by this initiative in a given time period?	Use product metrics or forecasts (e.g., number of users per month, transactions per quarter).	"This feature will reach 1,000 users per month."
Impact	How much will this initiative move the needle for each affected user or event?	Use a scale and choose the scale that best fits your business goal (e.g., conversion, delight).	"Expected to increase conversion rate by 10% (Impact = 2)."
Confidence	How sure are you about your estimates for reach and impact?	Expressed as a percentage or decimal. Use data and research to support higher confidence.	"We have strong data, so confidence is 90% (0.9)."
Effort	How much total work is required to complete the initiative?	Estimate in person-months, weeks, or another consistent unit. Lower effort is better.	"This will require 2 person-months of work."

The RICE Prioritization

Benefits

- Objectivity: Reduces bias by using quantifiable factors.
- Transparency: Makes it easy to explain and defend prioritization decisions to stakeholders.
- Efficiency: Helps teams focus on high-impact, low-effort work.
- Adaptability: Suitable for both short-term sprints and long-term product roadmaps.

Limitations

- Estimation Challenges: Requires reliable data for accurate scoring; estimates may be subjective or uncertain.
- Not One-Size-Fits-All: May need to be combined with other frameworks for complex or strategic decisions.

The Product Life Cycle

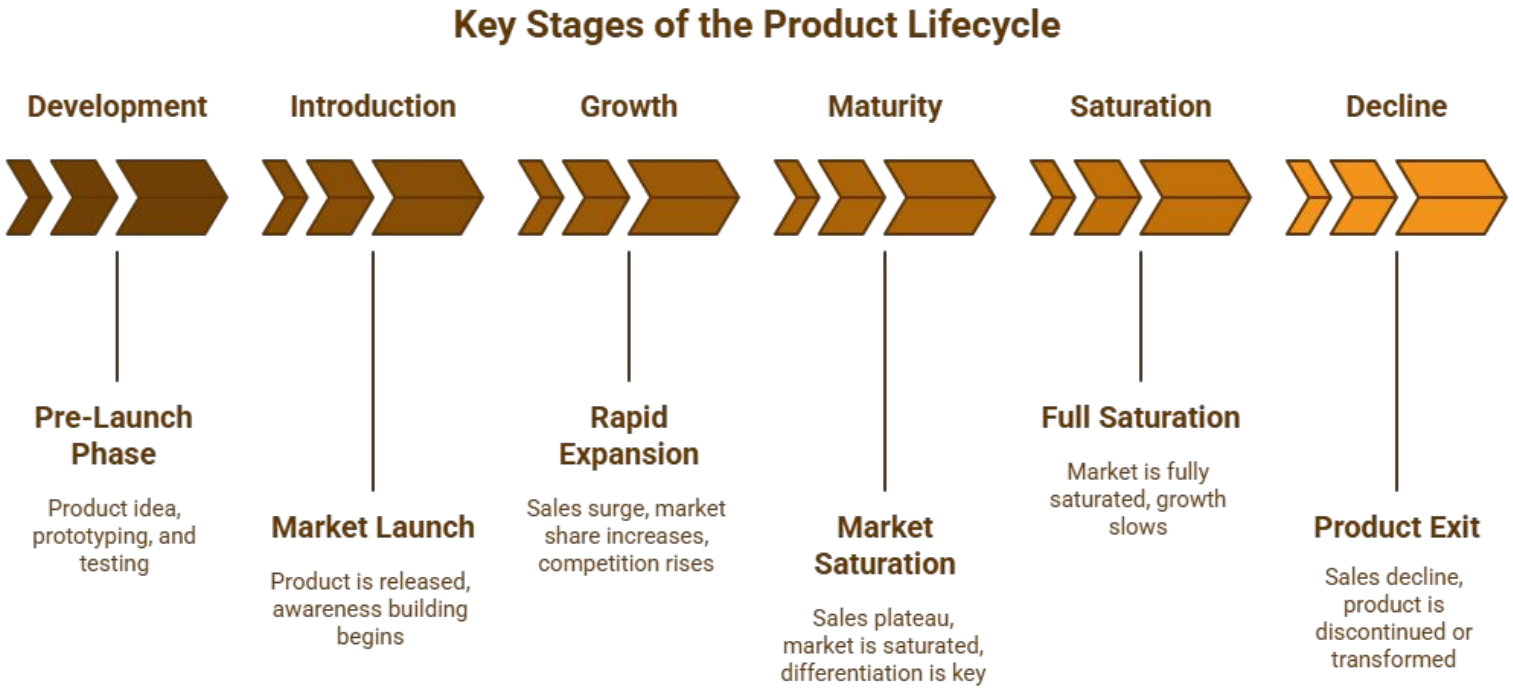
Definition

The product life cycle is the journey a product takes from its initial idea and development through its introduction, growth, maturity, and eventual decline in the market. Understanding the life cycle helps product managers and businesses make informed decisions about marketing, investment, and product strategy at every stage.

Significance

- Strategic Planning:
  Helps allocate resources and marketing spend appropriately at each stage.
- Maximizing Profit:
  Informs pricing, promotion, and feature development strategies to extend profitable phases.
- Managing Decline:
  Enables proactive decisions about product updates, repositioning, or discontinuation.

The Product Life Cycle



The Product Life Cycle

Examples

- *Oldsmobile:*
  Peaked in the 1980s, then declined and was discontinued in 2001.
- *Coca-Cola's "New Coke":*
  Rapid introduction and decline within 79 days due to poor market fit.
- *Nintendo:*
  Successfully extends the maturity phase by updating and reinventing classic products.
- *Flat-Screen TVs:*
  Currently in the mature phase, with incremental improvements and fierce competition.

The Hook Model

Definition

The Hook Model is a behavioral design framework that helps companies create products that drive user engagement through habit formation. It defines the stages that characterize the relationship between customers and engaging products, providing a systematic approach to designing and improving products that subtly influence user behavior over time. The ultimate goal is to reach a point where users automatically turn to the product without needing any external trigger—a stage known as ***the internal trigger***.

Tips

- Understand true user motivation: Research and empathize with your users to uncover what truly drives them.
- Simplify actions: Make key actions as easy and frictionless as possible.
- Deliver satisfying, variable rewards: Ensure rewards are meaningful and unpredictable to maintain engagement.
- Encourage investment: Design opportunities for users to invest in the product, increasing its personal value and creating future triggers.

Trigger

Triggers motivate users to take action and come in two forms:

- External Triggers: Environmental cues like notifications or ads.
- Internal Triggers: Emotions or thoughts (e.g., boredom, loneliness).

Products become habits by first using external triggers, then building using the internal ones.
Tip: Identify your users' true motivations.

Actions

Actions are user behaviors in response to triggers, moving them closer to their goals. Actions should be:

- Easy, attractive, matched to user motivation and ability
- According to Behavior = Motivation × Ability × Trigger equation , all three must be present.

Tip: Make actions simple and clearly beneficial.

Investment

Investment is the user's commitment—time, effort, data, or money—which increases product value and future engagement (the IKEA effect).
Examples: customizing profiles, building collections. Investments create cues that keep users coming back.
Tip: Provide multiple levels of investment to deepen commitment.

Variable Rewards

Variable rewards keep users engaged through unpredictability. Rewards can be:

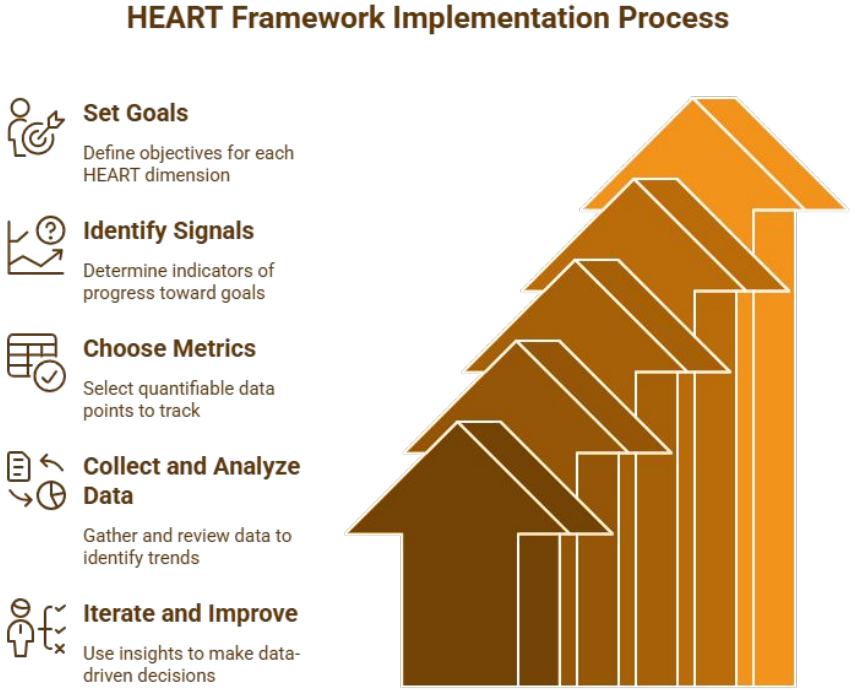
- Social (likes) or material (points)
- Self-directed (personal satisfaction)
- Unpredictable activating the brain's dopamine system

Tip: Make rewards dynamic and satisfying.

The HEART Framework

The HEART Framework is a user-centered methodology developed by Google to measure and improve the user experience (UX) of digital products. It breaks UX down into five key dimensions—Happiness, Engagement, Adoption, Retention, and Task Success—and provides a structured approach for teams to set goals, identify signals, and track metrics for each area.

- Holistic UX Measurement: Captures both emotional and behavioral aspects of user experience.
- Data-Driven Decisions: Links user feedback and analytics directly to product improvements.
- Team Alignment: Provides a shared language and structure for cross-functional teams to focus on UX goals.



The HEART Framework

Dimension	What It Measures	Example Metrics
Happiness	User attitudes, satisfaction, and sentiment	NPS, CSAT, user satisfaction surveys, reviews
Engagement	Depth and frequency of user interaction with the product	DAU/MAU, session length, actions per user
Adoption	How many users start using the product or a new feature	New user sign-ups, feature adoption rate
Retention	How many users return and continue to use the product over time	Churn rate, repeat usage, cohort retention
Task Success	How efficiently and effectively users complete key tasks within the product	Task completion rate, time on task, error rate

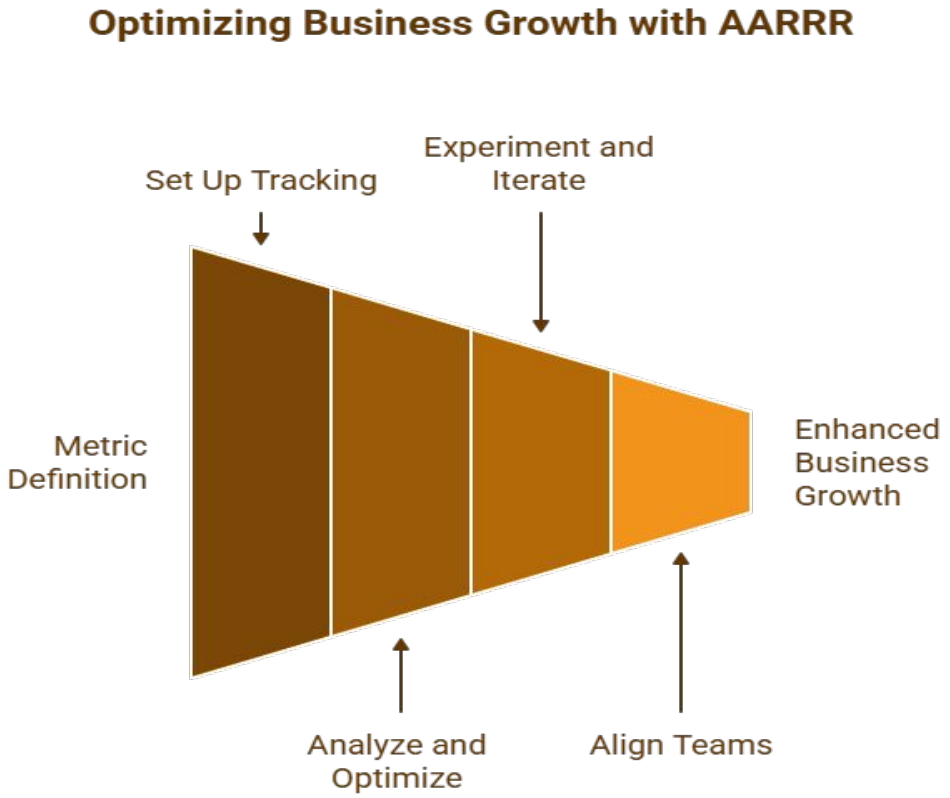
Example

For a fintech app:

- ***Happiness:*** Goal—users feel the app is secure and easy. Signal—positive app store reviews. Metric—NPS, average app rating.
- ***Engagement:*** Goal—users actively use features. Signal—more transactions per user. Metric—number of transactions per user per month.
- ***Adoption:*** Goal—new users complete onboarding. Signal—first transaction completed. Metric—onboarding completion rate.
- ***Retention:*** Goal—users return monthly. Signal—repeat logins. Metric—monthly active users, churn rate.
- ***Task Success:*** Goal—users complete payments without errors. Signal—successful payment completion. Metric—payment success rate, error rate.

The AARRR Framework

The AARRR Framework, also known as Pirate Metrics, is a simple yet powerful model for tracking and optimizing the key stages of the user journey in product-led growth businesses. Developed by Dave McClure, AARRR stands for Acquisition, Activation, Retention, Referral, and Revenue—five critical metrics that help startups and product teams understand, measure, and improve how users interact with their product at every stage.



The AARRR Framework

Stage	Key Question	What to Track	Example Metrics
Acquisition	How are people discovering our product?	Channels and tactics that bring users to your product	Website visits, signups, CAC, source
Activation	Are users having a great first experience?	Actions that indicate users are realizing product value	Onboarding completion, “aha” moments, activation rate, free-to-paid conversions
Retention	Are users coming back and continuing to engage?	How often users return and continue using the product	Retention rate, churn rate, DAU/MAU
Referral	Are users recommending the product to others?	How users share and promote your product	Referral program participation, NPS, social shares, invites sent
Revenue	Are users willing to pay and generating profit?	How the product generates income and monetizes users	MRR, ARPU, CLV, upgrades, total revenue

Example

- ***Acquisition:***
  Use SEO, paid ads, and social media to drive traffic. Measure which channels bring the most engaged users.
- ***Activation:***
  Track how many users complete onboarding or reach the “aha” moment (e.g., Facebook’s “7 friends in 10 days”).
- ***Retention:***
  Monitor how many users return after a week or month. Reduce churn by improving product stickiness.
- ***Referral:***
  Launch referral programs or sharing features. Measure invites sent, shares, or NPS.
- ***Revenue:***
  Track upgrades, purchases, or subscriptions. Optimize pricing and upsell strategies.

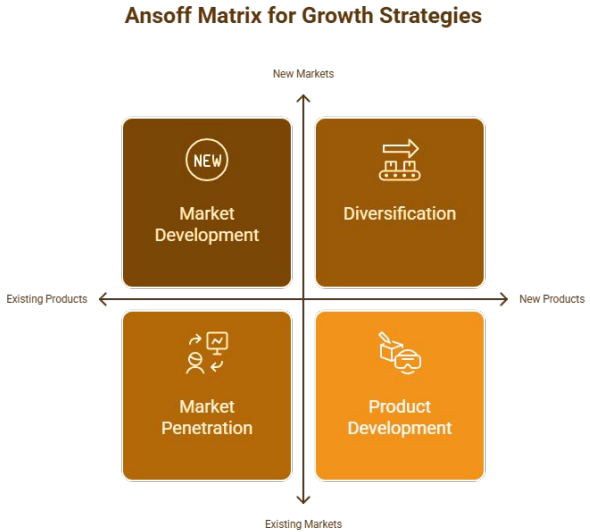
Ansoff Matrix

The Ansoff Matrix—also known as the Product/Market Expansion Grid—is a strategic planning tool that helps businesses evaluate and plan growth strategies by considering whether to focus on new or existing products and new or existing markets. The matrix is structured as a 2x2 grid, with “Products” (existing/new) on the X-axis and “Markets” (existing/new) on the Y-axis. Each quadrant represents a distinct growth strategy, each with its own risk profile.

To use the Ansoff Matrix,

1. Identify your existing products/services and current markets.
2. For each quadrant, brainstorm potential initiatives.
3. Recognize that risk increases as you move from market penetration to diversification.
4. Choose the strategy that aligns with your business goals, resources, and risk tolerance.
5. Develop action plans and track performance against objectives.

The matrix does not account for competitor reactions or provide a cost-benefit analysis.



Ansoff Matrix

Strategy	Description	Typical Actions	Example	Risk Level
Market Penetration	Selling more of existing products in existing markets.	Increase promotion, competitive pricing, loyalty programs, improve distribution.	Coca-Cola increases sales of classic Coke in the U.S. through promotions and advertising.	Lowest
Market Development	Selling existing products in new markets.	Expand geographically, target new segments, use new distribution channels.	Starbucks opens stores in new countries with its existing menu.	Moderate
Product Development	Introducing new products to existing markets.	Launch new features, product variants, or entirely new products to current customers.	Apple launches the iPad to its existing customer base familiar with iPhones and Macs.	Moderate
Diversification	Launching new products in new markets (can be related or unrelated to current business).	Acquire or develop new products for unfamiliar markets, could be via partnerships/acquisitions.	Amazon enters cloud computing (AWS) beyond its core e-commerce market.	Highest

User Research Methods

User research methods are essential for product managers to deeply understand their users and design better products. Three foundational methods include *personas, surveys, and journey mapping*.

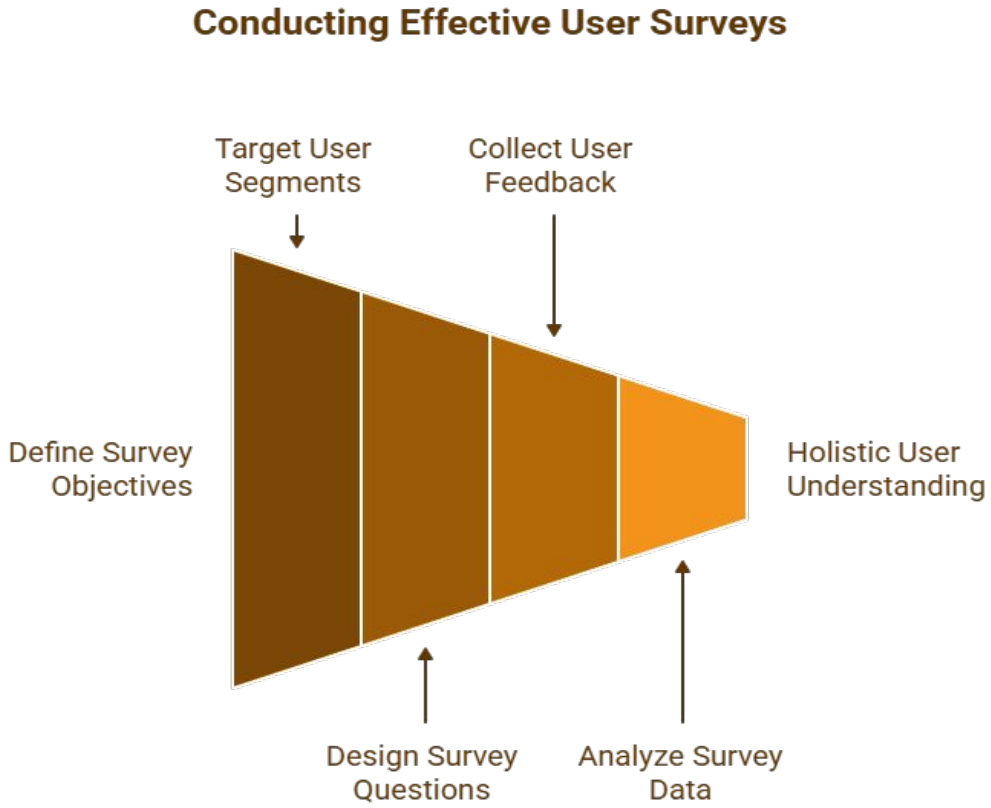
Personas are semi-fictional, data-driven representations of key user segments. Built from qualitative and quantitative research—including interviews, surveys, and observations—personas capture demographic details, motivations, pain points, goals, behaviors, and even communication styles. They help teams empathize with users, align on user needs, and make design and feature decisions that are grounded in real user contexts. Personas are not just market segments; they are detailed character studies that humanize data and keep the user at the center of product decisions.



For instance, *Duolingo* struggles to retain long-term users, especially those preparing for tests like TOEFL or IELTS, as they leave after meeting short-term goals, hurting daily active user (DAU) rates. For such a scenario, one of the personas could look like,

User Research Methods

Surveys provide quantitative insights into user preferences, pain points, and satisfaction. They are a scalable way to validate hypotheses, gather feedback on features, and measure user sentiment across a broad audience. Surveys can be structured with closed-ended questions for statistical analysis or open-ended questions for richer, qualitative feedback. Best practices include defining clear objectives, targeting the right user segments, and combining survey data with other research methods for a holistic view.



User Research Methods

Journey Mapping visualizes the full experience of a user as they interact with a product or service, highlighting each stage, touchpoint, pain point, and emotional state. Journey maps typically start with a persona and plot the user's actions, mindset, and emotions across phases such as discovery, onboarding, and engagement. This process reveals friction points and opportunities for improvement, enabling teams to prioritize enhancements that have the greatest impact on user experience. Journey mapping is most effective when informed by direct user research—such as interviews, field studies, and diary studies—and when tailored to different personas for more personalized insights.



User Journey v/s User Flow

User Journey:

A user journey provides a holistic, high-level view of a user's experience with a product or service across multiple channels and over time. It captures not only the actions but also the emotions, motivations, and pain points at each stage, helping teams understand the broader context and optimize the end-to-end experience.

User Flow:

A user flow, in contrast, is a granular, step-by-step representation of the specific actions a user takes to complete a particular task within a product. It focuses on the micro-level—such as signing up, resetting a password, or making a purchase—detailing each interaction and system response. User flows are especially valuable for identifying and removing friction in specific processes to improve usability and efficiency.

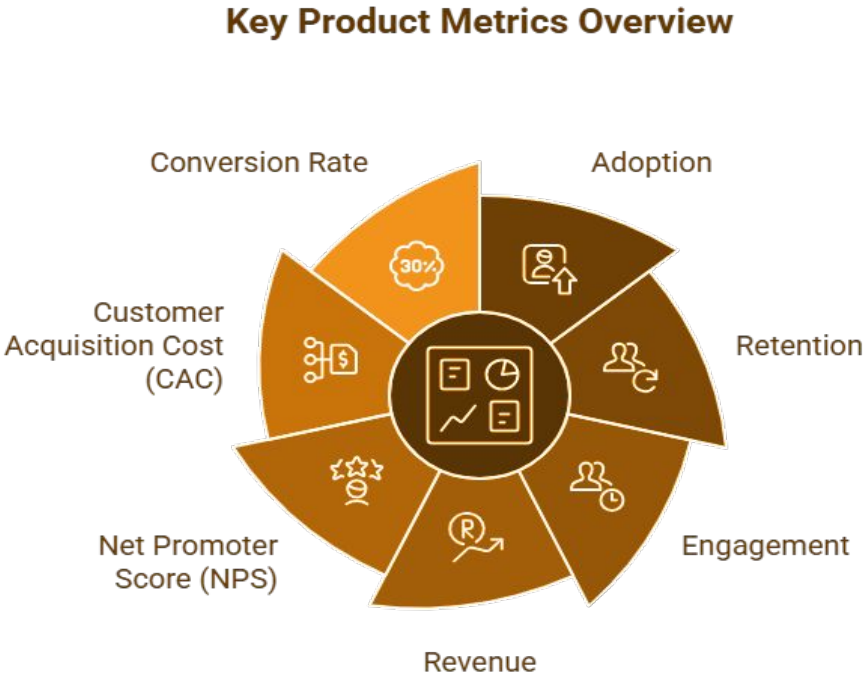
User journeys and user flows are complementary: journeys help teams understand the full experience and emotional context, while flows enable detailed optimization of key interactions.

User Journey v/s User Flow

Aspect	User Journey	User Flow
Scope	Macro: holistic, cross-channel, over time	Micro: specific task, within one product
Focus	Emotions, motivations, pain points, touchpoints	Actions, decisions, system responses
Output	Journey maps, storytelling	Flowcharts, wireframes, task diagrams
Purpose	Optimize overall experience, identify gaps	Improve task efficiency, remove friction
Example	Onboarding to advocacy for a SaaS product	Steps to reset a password in an app

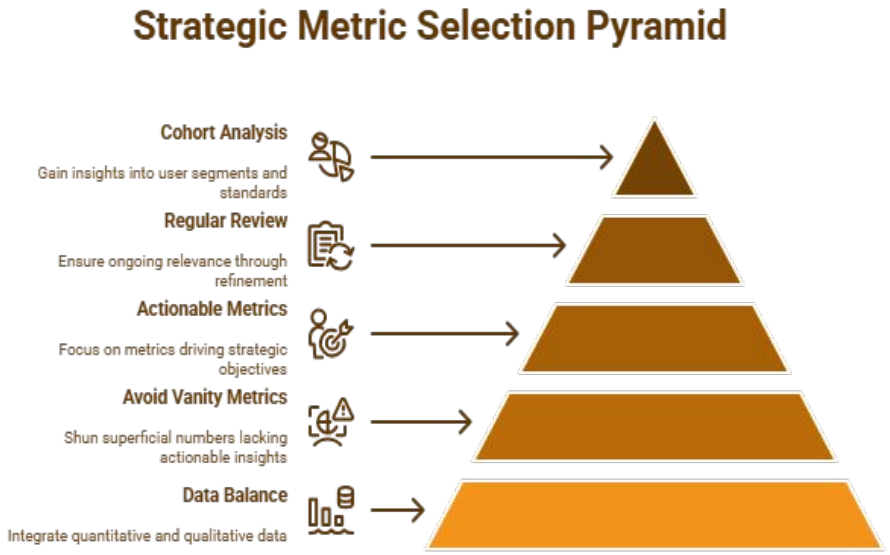
Selecting & Interpreting Metrics

Selecting and interpreting product metrics is fundamental to understanding product performance, guiding decision-making, and driving growth. Key types of metrics include ***adoption (e.g., activation rate, feature adoption), retention (e.g., user retention rate, churn rate), engagement (e.g., DAU/MAU, session length), revenue (e.g., MRR, CLV), Net Promoter Score (NPS), Customer Acquisition Cost (CAC), and conversion rate***. To choose the right metrics, product managers should align them with core business and user goals, often using the North Star Metric framework, which focuses on a single metric that best captures the product's value to customers and drives long-term success.



Selecting & Interpreting Metrics

Effective metric selection balances quantitative data (like usage analytics) with qualitative insights (such as user feedback and satisfaction surveys), providing a comprehensive view of user behavior and motivations. It is crucial to avoid vanity metrics—superficial numbers like total sign-ups or page views that look impressive but do not inform actionable improvements or reflect true product health. Instead, focus on actionable, outcome-oriented metrics that directly influence strategic objectives and can be clearly linked to business value. Regularly review and refine your metrics to ensure ongoing relevance, and use cohort analysis and benchmarking to gain deeper insights into user segments and industry standards.



Product Metrics: What, Why, How, and Examples

Metric	What It Measures	How to Use It	Example
Adoption Rate	% of new users who start using a feature or product	Track feature launches, identify onboarding friction, inform resources	40% of new users complete onboarding in their first week
Activation Rate	% of users reaching a key milestone that signals value realization	Define activation event, optimize onboarding, measure value realization	60% of sign-ups upload a document within 24 hours
Retention Rate	% of users returning to the product over a period	Assess stickiness, identify churn causes, inform retention strategies	30-day retention is 25% (1 in 4 users return after a month)
Churn Rate	% of users who stop using the product in a given period	Monitor user loss, identify at-risk segments, improve retention	Monthly churn rate is 5% (50 out of 1,000 users leave per month)
Engagement (DAU/MAU)	Frequency of user interactions (Daily/ Monthly Active Users)	Gauge usage, compare DAU/MAU for stickiness, spot trends	DAU/MAU ratio is 0.2 (20% of monthly users are active daily)
Session Length	Average time users spend per session	Assess depth of engagement, optimize flows	Average session is 7 minutes; drop-off after 3rd screen
Revenue (MRR/ARR)	Monthly or Annual Recurring Revenue	Track growth, forecast, segment by plan/cohort	MRR is $50,000; ARR is $600,000
Customer Lifetime Value (CLV)	Total revenue expected from a user over their lifetime	Inform acquisition spend, prioritize high-value segments	CLV is $400 per user; CAC should be < $100
Customer Acquisition Cost (CAC)	Cost to acquire a new customer	Monitor marketing efficiency, compare to CLV, optimize channels	CAC is $80 per user; compare to CLV
Conversion Rate	% of users moving from one stage to another (e.g., sign-up to paid)	Optimize funnels, test messaging, identify friction	10% of free trial users convert to paid
Net Promoter Score (NPS)	User willingness to recommend (scale -100 to 100)	Gauge satisfaction, benchmark, follow up with detractors	NPS is 45; follow up with detractors for feedback
Feature Adoption	% of users using a specific feature	Assess feature success, inform roadmap, identify training needs	25% use the new reporting dashboard
Cohort Retention	Retention of specific user groups over time	Segment by acquisition date, compare cohort behavior	March cohort: 35% 90-day retention vs. 20% in February

FOCUS

03



PRODUCT:

Tech Simplifield For PMs

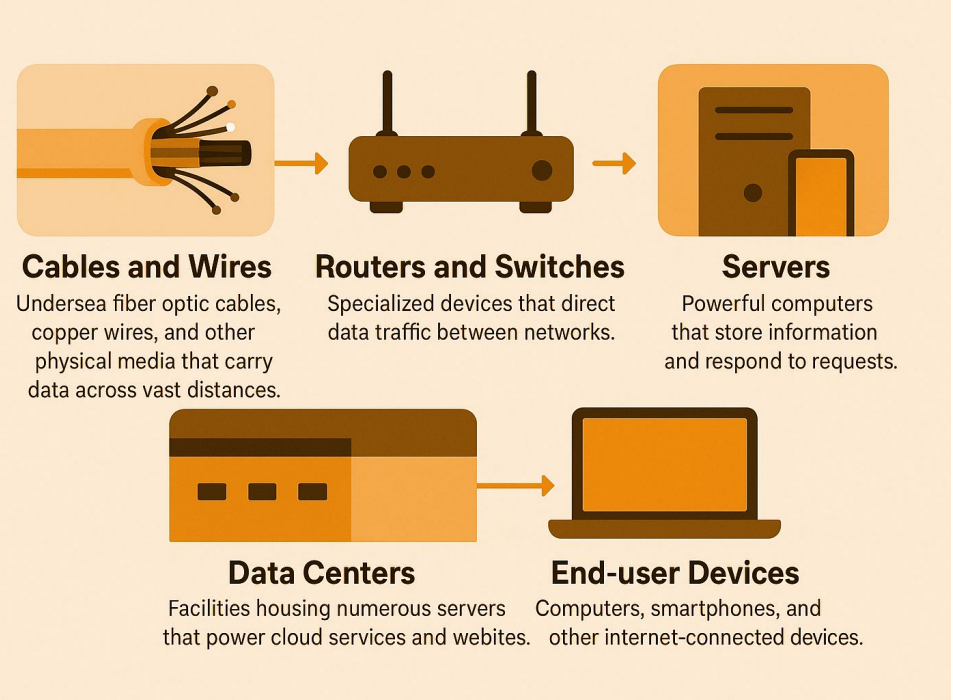


Tech for PM

***PART 1*: Foundations of the Web and Internet**

How the Internet Works

The Internet is fundamentally a global network of interconnected networks—in fact, the term "Internet" literally comes from "interconnected networks." This massive system enables billions of devices worldwide to communicate with each other, regardless of their geographic location or hardware specifications.



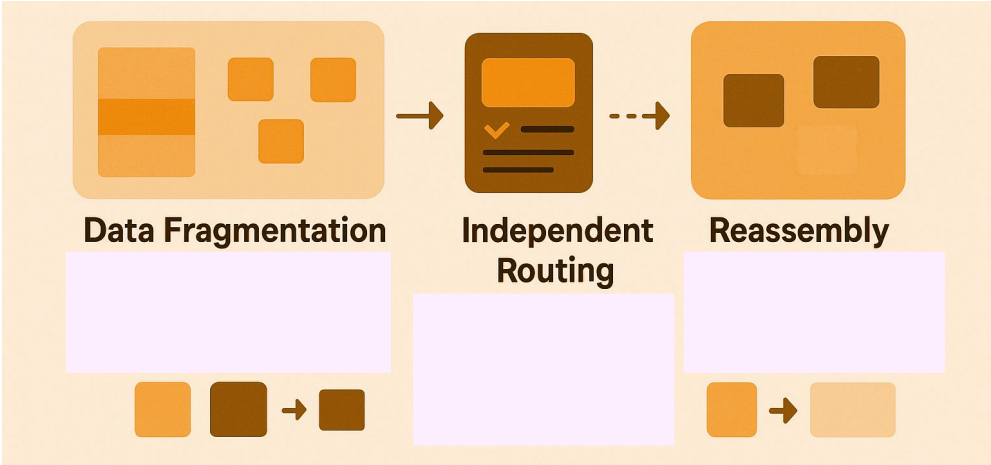
The Physical Infrastructure

At its foundation, the Internet consists of physical components that create the backbone of global connectivity

Packet Switching: The Internet's Core Mechanism

The Internet transmits information using a technique called packet switching, which is fundamental to understanding how data moves across networks:

- **Data Fragmentation:** When you send or request information over the Internet, that data is broken down into smaller units called packets.
- **Packet Structure:** Each packet contains:
  - The actual data (content)
  - A header with routing information (origin and destination addresses)
  - Instructions for reassembly.
- **Independent Routing:** Packets travel independently through the network, potentially taking different paths,
- **Reassembly:** When packets reach their destination, they're reassembled in the correct order to reconstruct the original information.



The Role of Protocols

For the Internet to function cohesively, all connected devices must **"*speak the same language.*"** This is where networking protocols come in:



- Protocols: Standardized rules that define how data is formatted, transmitted, and received across networks 1
- Protocol Suites: Collections of protocols that work together, with TCP/IP being the foundation of the modern Internet 7

Client, Server, DNS, IP Addresses

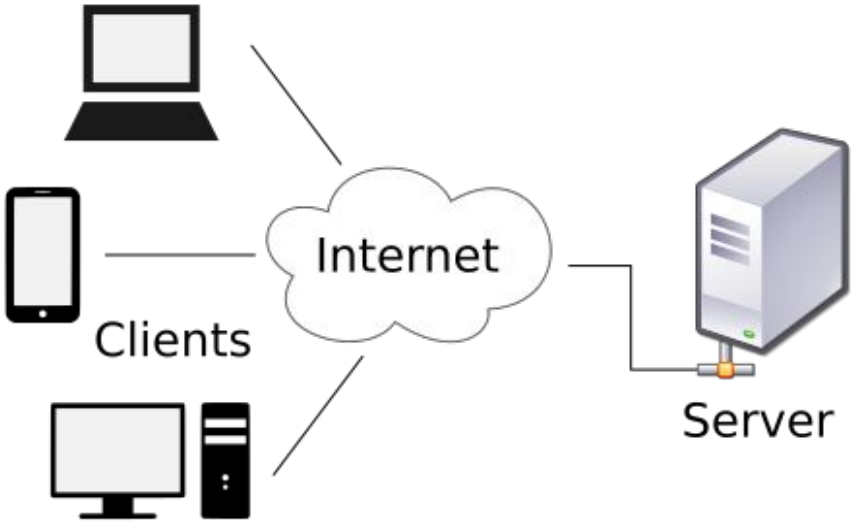
The Client-Server Model

The client-server model is the predominant paradigm for network-based applications and services on the Internet:

- Clients: End-user devices (computers, smartphones, tablets) that request information or services.
- Servers: Computers that store, process, and deliver resources or services upon request

This relationship is fundamental to most digital products:

1. A client sends a request for specific information or functionality
2. The server processes this request according to its programming
3. The server returns the appropriate response to the client
4. The client presents the information or executes the requested function.



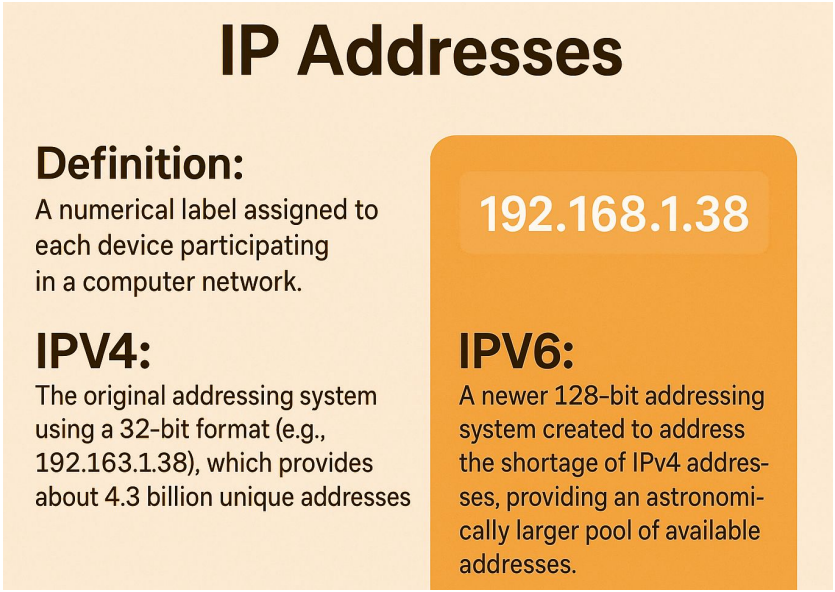
Client, Server, DNS, IP Addresses

IP Addresses: The Internet's Addressing System

Every device connected to the Internet needs a unique identifier to send and receive data. IP addresses serve this purpose:

- Definition: A numerical label assigned to each device participating in a computer network.
- IPv4: The original addressing system using a 32-bit format (e.g., 192.168.1.38), which provides about 4.3 billion unique addresses.
- IPv6: A newer 128-bit addressing system created to address the shortage of IPv4 addresses, providing an astronomically larger pool of available addresses.

IP addresses enable direct communication between devices across the Internet, but they're difficult for humans to remember.

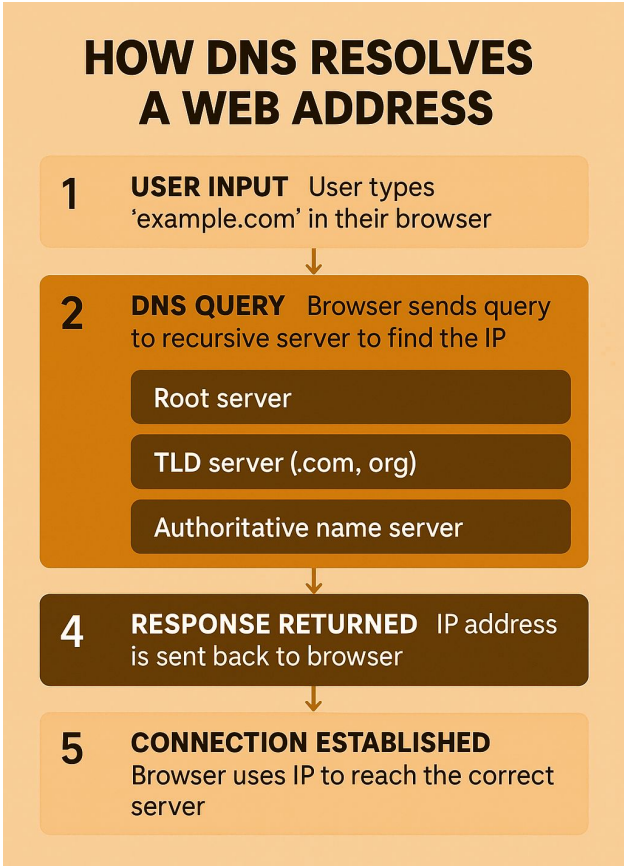


Client, Server, DNS, IP Addresses

Domain Name System (DNS): The Internet's Directory

The Domain Name System (DNS) translates human-friendly domain names into machine-readable IP addresses:

1. **User Input:** A user types a URL like "example.com" in their browser
2. **DNS Query:** The browser sends a query to DNS servers to find the corresponding IP address
3. **Hierarchical Resolution:** The query may pass through several DNS servers:
   a. Root servers
   b. Top-Level Domain (TLD) servers (e.g., .com, .org)
   c. Authoritative name servers
4. **Response:** The IP address is returned to the browser
5. **Connection:** The browser connects to the web server at that IP address



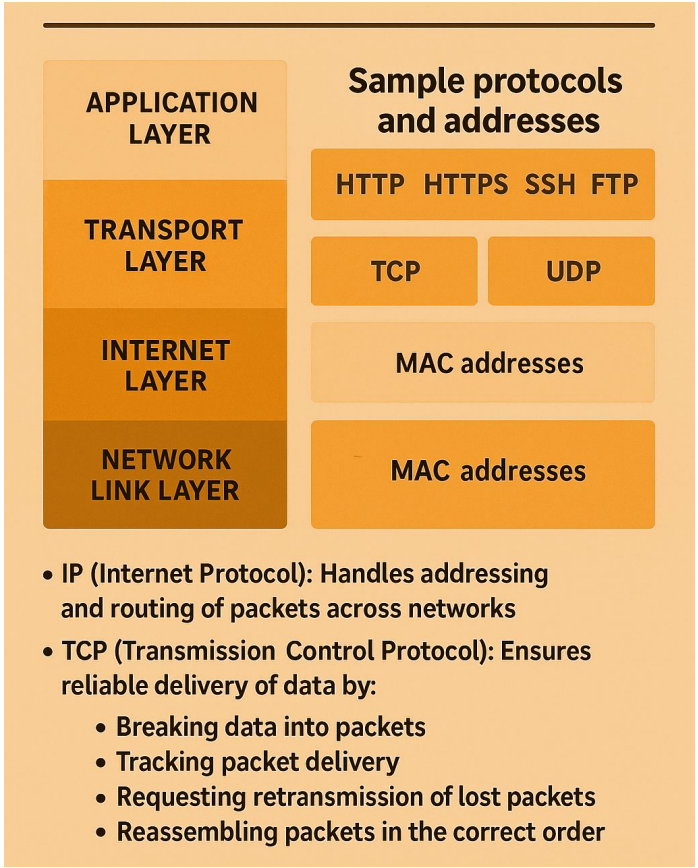
HTTP, HTTPS, TCP/IP, SSL

TCP/IP: The Foundation of Internet Communication

TCP/IP (Transmission Control Protocol/Internet Protocol) is the fundamental suite of protocols that enables internet connectivity:

- **IP (Internet Protocol):** Handles addressing and routing of packets across networks
- **TCP (Transmission Control Protocol)**: Ensures reliable delivery of data by:
- **Breaking data into packets**
- **Tracking packet delivery**
- **Requesting retransmission** of lost packets
- **Reassembling packets** in the correct order

TCP/IP works as an abstraction layer between applications and the physical network, allowing different types of networks and devices to communicate seamlessly

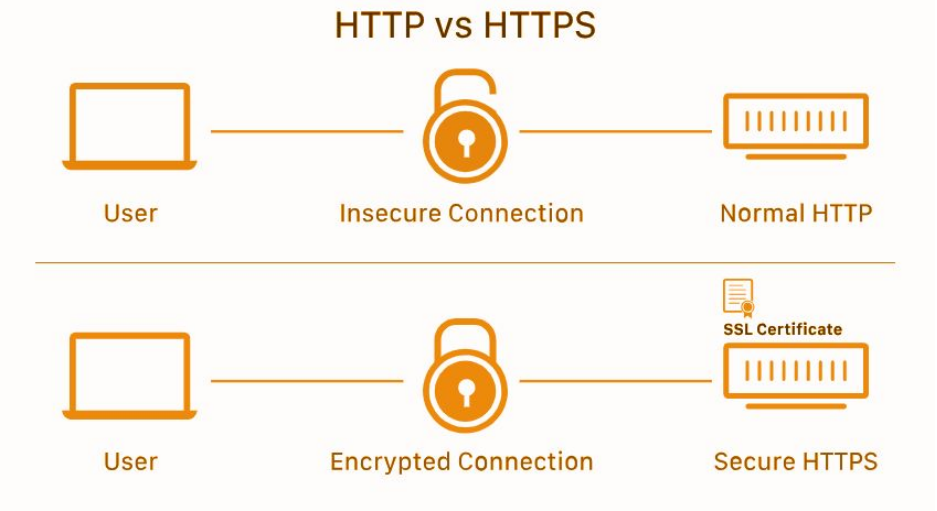


HTTP, HTTPS, TCP/IP, SSL

HTTP: The Web's Communication Protocol

HTTP (Hypertext Transfer Protocol) is the protocol that governs how web browsers and servers exchange information:

- **Request-Response Cycle:** HTTP follows a simple model where clients send requests and servers return responses
- **Stateless Protocol:** Each request-response pair is independent, with no inherent memory of previous interactions
- **Plain Text:** HTTP transmits data in plain text, making it readable but not secure



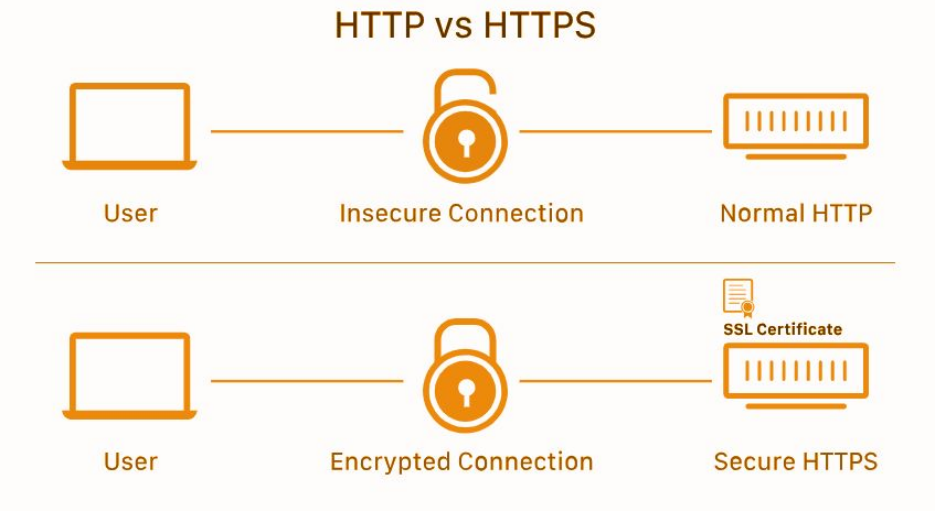
For product managers, understanding HTTP is essential for designing APIs, troubleshooting issues, and making informed decisions about web application architecture.

HTTP, HTTPS, TCP/IP, SSL

HTTPS: Securing Web Communication

HTTPS (HTTP Secure) addresses the security limitations of HTTP by encrypting all communications between browser and server:

- ***Encryption:*** All data transmitted is encrypted, preventing eavesdropping
- ***Authentication:*** Verifies the identity of the website through digital certificates
- ***Data Integrity:*** Ensures data hasn't been tampered with during transmission



The security benefits make HTTPS essential for:

- **User login and authentication**
- **E-commerce transactions**
- **Form submissions**
- **Any exchange of sensitive information**

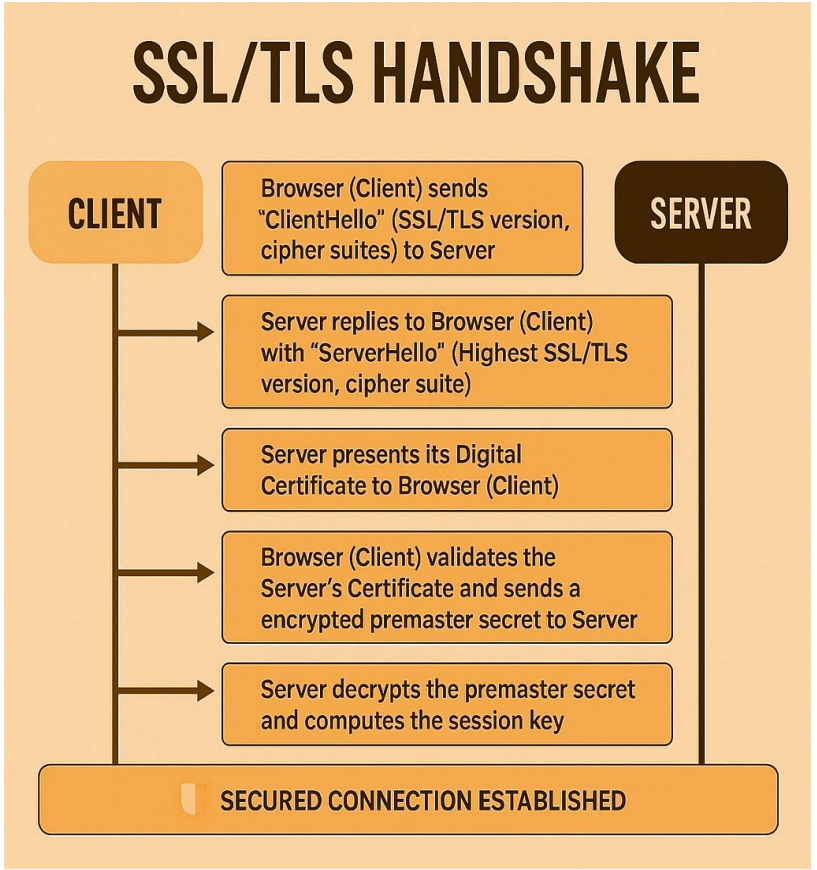
HTTP, HTTPS, TCP/IP, SSL

SSL/TLS: The Security Layer

SSL (Secure Sockets Layer) and its successor TLS (Transport Layer Security) are the cryptographic protocols that enable HTTPS:

- **Handshake Process:** Establishes a secure connection through a series of steps:
  - Client and server agree on encryption algorithms
  - Server presents its digital certificate for verification
  - Encryption keys are exchanged
  - A secure connection is established
- **Digital Certificates:** Documents that verify the identity of websites, issued by trusted Certificate Authorities (CAs).

For product managers, implementing HTTPS is no longer optional—it's a necessity for user trust, search engine ranking, and modern browser compatibility.



3-Tier Architecture (Frontend, Backend, Database)

Presentation Tier (Frontend)

The presentation tier is what users directly interact with:

- **Purpose**: Display information and collect user input
- **Technologies:** HTML, CSS, JavaScript, and frontend frameworks
- **Responsibility:** Create an intuitive, responsive user interface
- *Examples: Web pages, mobile app interfaces, desktop application GUIs*

This layer focuses exclusively on the **user experience,** keeping business logic and data management separate.

Application Tier (Backend/Middle Tier)

The application tier contains the business logic that processes information:

- **Purpose:** Process data, implement business rules, and coordinate application functions
- **Technologies:** Programming languages like Python, Java, PHP, Ruby, JavaScript (Node.js)
- **Responsibility:** Receive user input from the presentation tier, process it according to business rules, and interact with the data tier
- *Examples: API endpoints, authentication services, data validation logic*

This layer serves as the bridge between what users see and where data is stored.

3-Tier Architecture (Frontend, Backend, Database)

Data Tier (Database)

The data tier is responsible for storing and managing application data:

- **Purpose:** Store, retrieve, and maintain data integrity
- **Technologies:** Relational databases (MySQL, PostgreSQL), NoSQL databases (MongoDB, Cassandra), data warehouses
- **Responsibility:** Provide data persistence, enforce data relationships, and ensure data consistency
- *Examples: User accounts, product information, transaction records*

Benefits for Product Development

This architectural pattern offers several advantages for product development:

- **Independent Development:** Teams can work on different tiers simultaneously
- **Scalability**: Each tier can be scaled independently based on specific demands
- **Maintenance:** Changes to one tier don't necessarily affect others
- **Security:** Data is isolated from direct user access

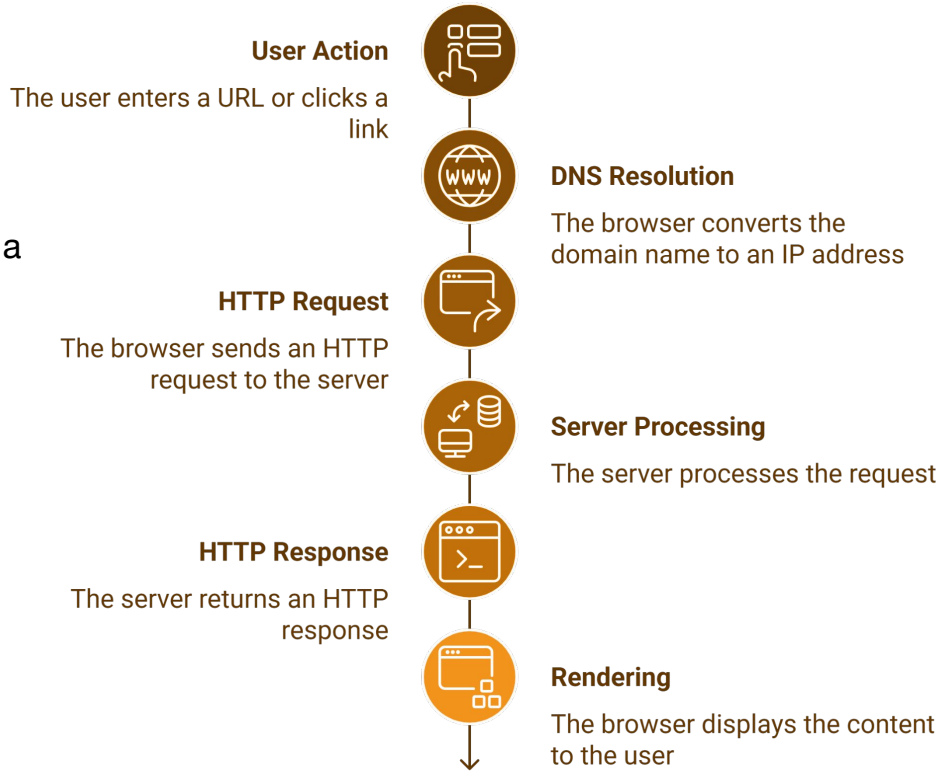
For product managers, understanding this architecture helps in planning feature development, resource allocation, and technology stack decisions.

How the Web Works (Client-Server Model)

The Web Request Lifecycle

When a user interacts with a web application, a sequence of events occurs:

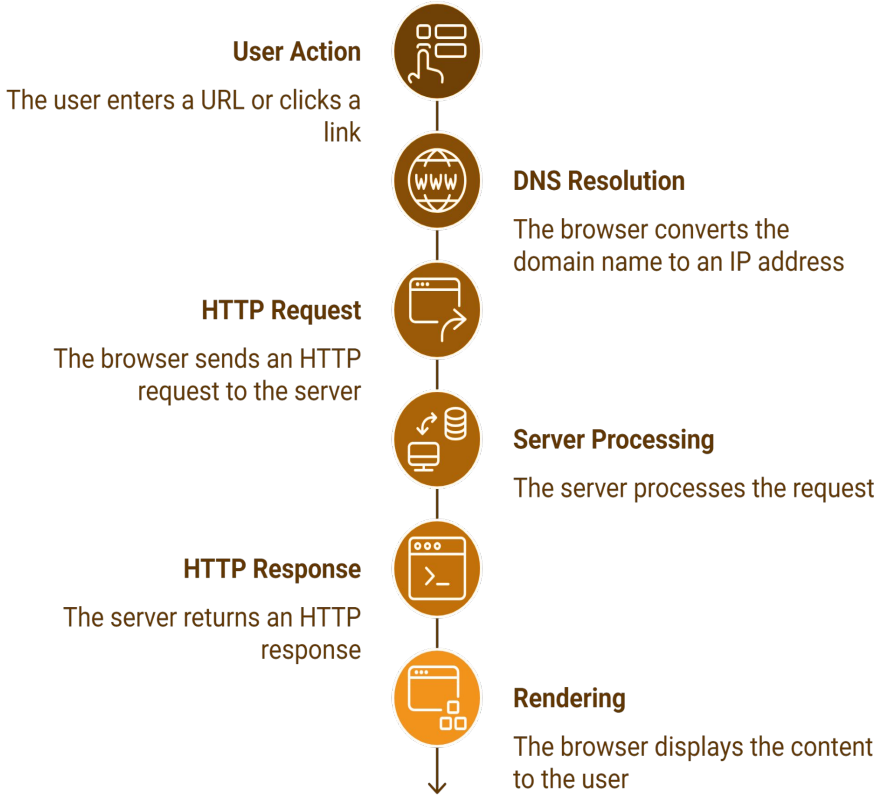
- User Action: The user enters a URL or clicks a link in their browser (the client)
- DNS Resolution: The browser uses DNS to convert the domain name to an IP address
- HTTP Request: The browser sends an HTTP request to the server at that IP address
- Server Processing: The server processes the request, which may involve:
  - Authentication
  - Database queries
  - Application logic
  - Content generation



How the Web Works (Client-Server Model)

- HTTP Response: The server returns an HTTP response containing:
  - Status code (indicating success or failure)
  - Headers (metadata about the response)
  - The requested content (HTML, JSON, images, etc.)
- Rendering: The browser processes and displays the content to the user

This entire process typically happens in milliseconds, creating the seamless experience users expect.



How the Web Works (Client-Server Model)

Static vs. Dynamic Content

Web content broadly falls into two categories:

- ***Static Content:*** Files (HTML, images, CSS, JavaScript) that are served exactly as they're stored on the server
- ***Dynamic Content:*** Generated on-demand based on:
  - User input
  - Database queries
  - Application state
  - External APIs

Modern web applications typically combine both types of content for optimal performance and functionality.

For product managers, understanding this lifecycle helps in diagnosing performance issues, optimizing user experience, and making strategic technology decisions.

Choose the appropriate web content type for optimal performance and functionality.

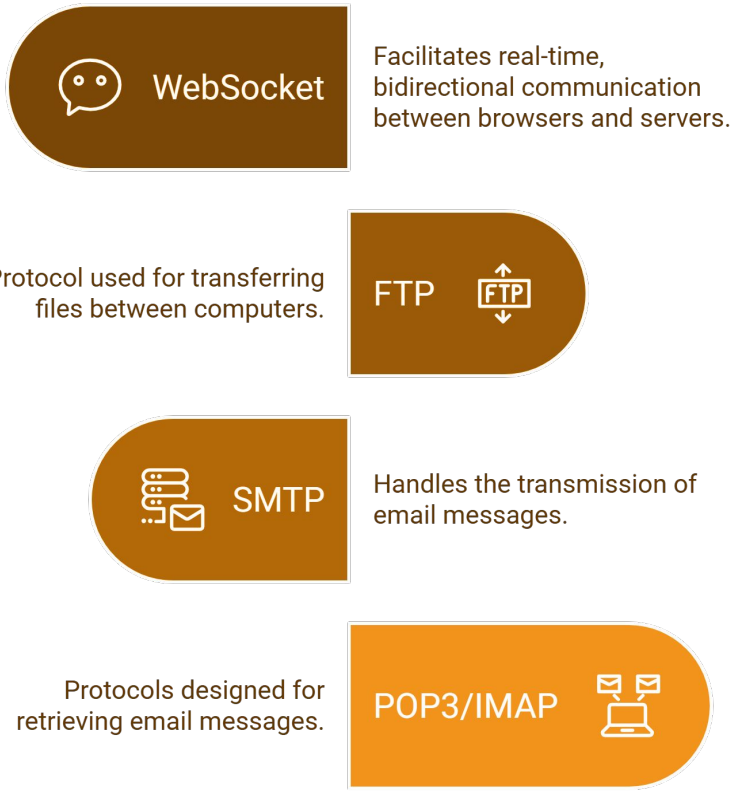


Web Protocols & Standards

Core Web Protocols

Beyond HTTP and HTTPS, several other protocols play important roles:

- ***WebSocket:*** Enables two-way real-time communication between browsers and servers
- ***FTP (File Transfer Protocol):*** Used for transferring files between computers
- ***SMTP (Simple Mail Transfer Protocol):*** Handles email transmission
- ***POP3/IMAP:*** Protocols for retrieving email



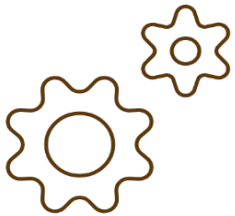
Web Standards Organizations



W3C

Develops specifications for core web technologies. Ensures interoperability and accessibility.

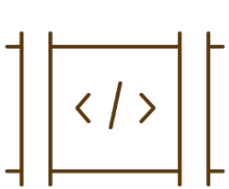
1



IETF

Creates standards for internet protocols. Ensures smooth data transmission online.

2



WHATWG

Focuses on HTML and API specifications. Drives web application evolution and standardization.

3



ECMA International

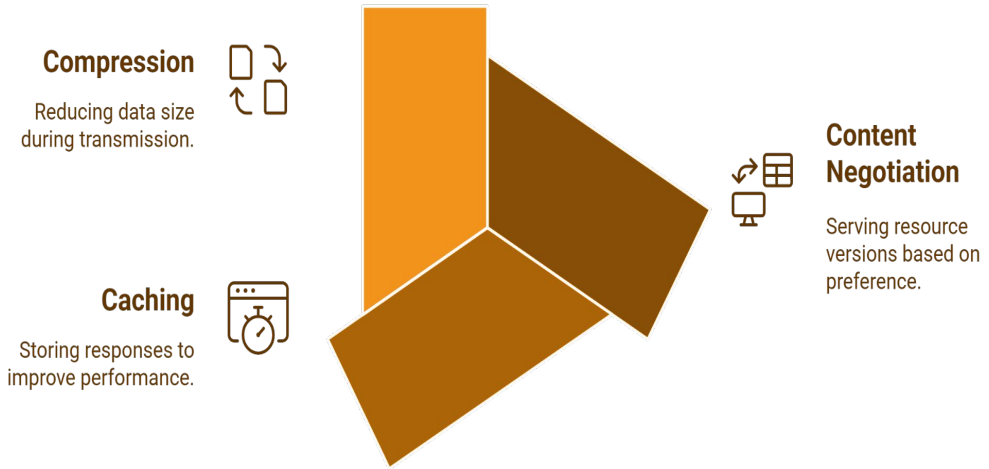
Standardizes JavaScript through ECMAScript. Ensures consistent scripting language functionality.

4

Advanced HTTP Concepts

Building on our earlier discussion of HTTP, here are some additional important aspects:

- ***Content Negotiation:*** The mechanism for serving different versions of a resource based on client preferences.
- ***Caching:*** Techniques for storing responses to reduce server load and improve performance.
- ***Compression:*** Methods for reducing data size during transmission.



REST (Representational State Transfer)

REST is an architectural style for designing networked applications, focusing on a stateless client-server relationship:

- **Resources:** Everything is a resource, identified by URLs
- **HTTP Methods:** Standard HTTP methods (GET, POST, PUT, DELETE) map to CRUD operations
- **Stateless Interactions:** Each request contains all information necessary for the server to fulfill it
- **Standardized Interfaces:** Consistent use of HTTP methods and status codes
- **Resource Representations:** Resources can have multiple representations (XML, JSON, HTML)10

RESTful APIs have become the standard for web service interfaces due to their simplicity, scalability, and alignment with web technologies.



HTTP, REST, OAuth

OAuth: Secure Delegation of Access

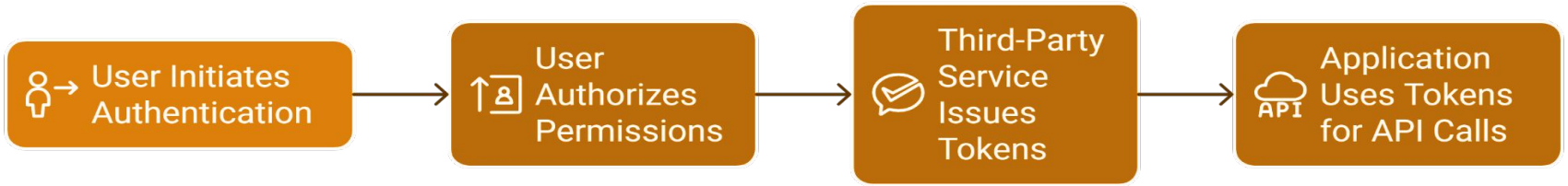
OAuth is an open standard for access delegation that enables third-party applications to access user resources without exposing credentials:

- ***Purpose:*** Allow secure delegated access to resources
- ***Token-Based:*** Uses tokens instead of sharing passwords
- ***Scoped Access:*** Grants specific permissions rather than full access
- ***Common Use Case:*** "Sign in with Google/Facebook/Apple" functionality

The OAuth flow typically involves:

1. The user initiates authentication with a third-party service
2. After authenticating, the user authorizes specific permissions
3. The third-party service issues tokens that the application can use to access resources
4. The application uses these tokens for subsequent API calls11

For product managers, understanding OAuth is crucial for implementing secure authentication systems and third-party integrations.

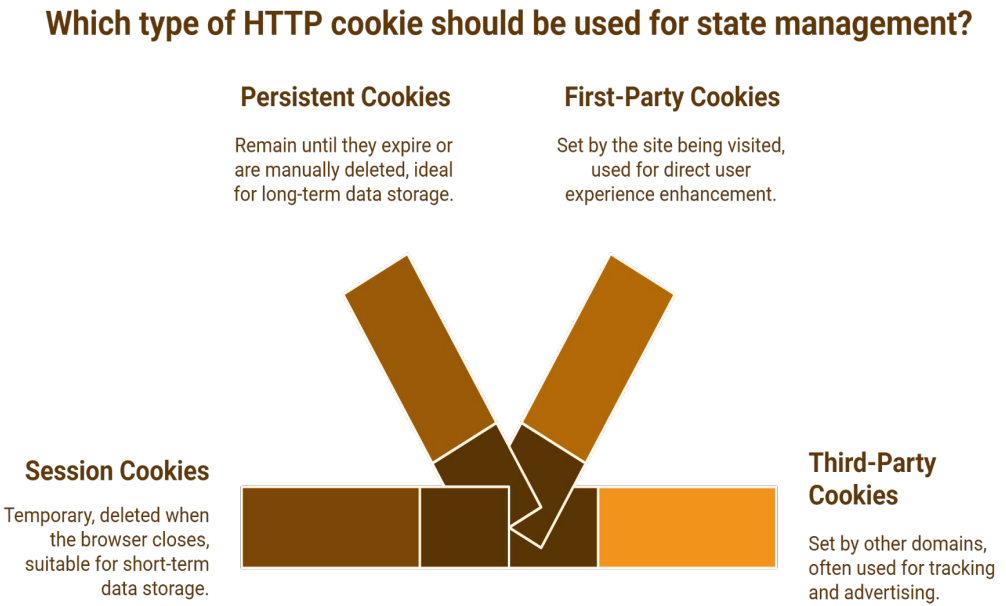


Cookies & Sessions

HTTP Cookies: Client-Side State Management

Since HTTP is stateless, websites need a mechanism to remember information about users. Cookies provide this capability:

- Definition: Small pieces of data stored in the user's browser
- Purpose: Track user information between requests and maintain state
- Structure: Key-value pairs with additional attributes like expiration time



Cookies & Sessions

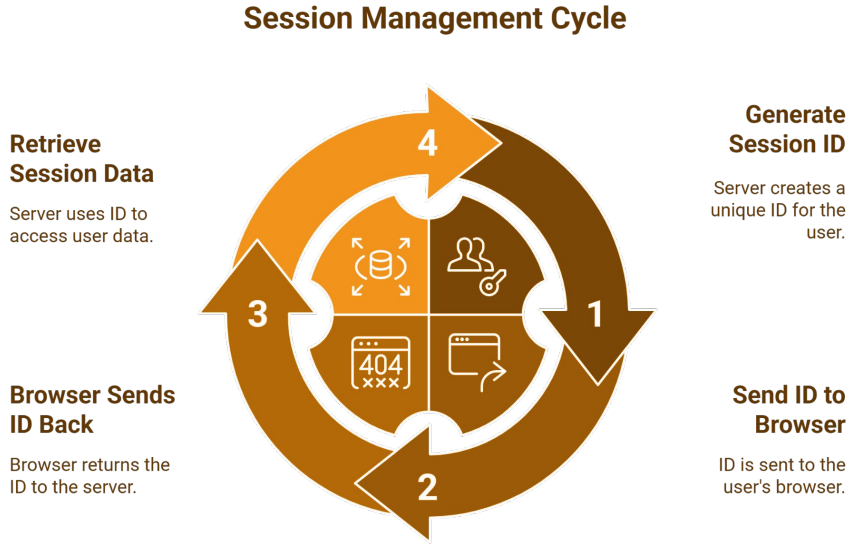
Sessions: Server-Side State Management

Sessions complement cookies by storing user-specific information on the server:

- When a user first visits a site, the server generates a unique session ID
- This ID is sent to the user's browser, typically stored in a cookie
- For subsequent requests, the browser sends the session ID back to the server
- The server uses this ID to retrieve the session data associated with that user

This approach offers several advantages:

- Sensitive information stays on the server
- Session data can be larger than cookie limitations
- Sessions can be invalidated server-side for security



Cookies & Sessions

Security and Privacy Considerations

For product managers, cookies and sessions raise important considerations:

- Privacy Regulations: Laws like GDPR and CCPA regulate cookie usage and require user consent
- Security Risks: Improperly implemented cookies can lead to vulnerabilities like session hijacking
- User Experience: Cookie consent banners and privacy controls affect user experience

Best practices include:

- Only storing necessary information
- Setting appropriate security attributes (Secure, HttpOnly, SameSite)
- Providing clear cookie policies
- Implementing proper session timeout mechanisms

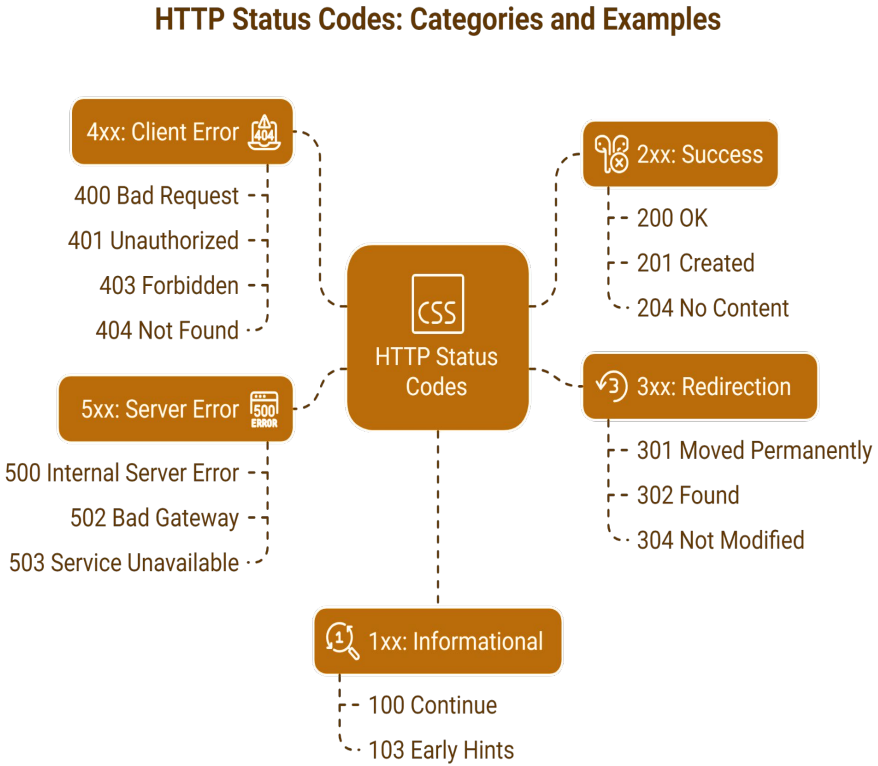


Status Codes & Methods (GET, POST, etc.)

HTTP Methods

HTTP methods define the type of operation to be performed on a resource:

- **GET:** Retrieve data without modifying it (safe and idempotent)
- **POST:** Submit data to be processed, typically creating a new resource (not idempotent)
- **PUT:** Update an existing resource by replacing it entirely (idempotent)
- **PATCH:** Apply partial modifications to a resource (may not be idempotent)
- **DELETE:** Remove a resource (idempotent)
- **OPTIONS:** Describe communication options for the target resource
- **HEAD:** Similar to GET but returns only headers, not the body





Tech for PM

***PART 2*: Core Technology Stacks and Tools for Product Managers**

HTML Fundamentals and Semantic Structure

HyperText Markup Language (HTML) forms the structural backbone of web pages, defining elements like headings, paragraphs, and images.

Every HTML document begins with `<!DOCTYPE html>`, followed by `<html>`, `<head>`, and `<body>` tags. The `<head>` contains metadata such as the page title and linked stylesheets, while the `<body>` houses visible content like `<h1>` for main headings and `<p>` for paragraphs.

Modern HTML5 introduces semantic elements like `<header>`, `<nav>`, and `<article>`, which improve accessibility and SEO by clarifying content structure.

For example, a product manager overseeing a blog platform would prioritize semantic tags to enhance search engine visibility. Key tags for PMs to understand include:

- `<form>` for user input fields
- `<img>` with `alt` attributes for accessibility compliance
- `<a>` for hyperlinks, using `href` to define destinations[20].

A basic HTML template demonstrates this hierarchy:

```
<!DOCTYPE html>
<html>
<head>
  <title>E-Commerce Site</title>
</head>
<body>
  <header>
    <h1>Welcome to Our Store</h1>
  </header>
  <main>
    <section>
      <h2>Featured Products</h2>
    </section>
  </main>
</body>
</html>
```

CSS Styling and Responsive Design

Cascading Style Sheets (CSS) separate presentation from content, enabling consistent styling across web pages. Key concepts include selectors (targeting elements by type, class, or ID), properties (e.g., `color`, `margin`), and values (e.g., `#FF5733` for red).

Media queries enable responsive design by adjusting layouts based on device characteristics:

```
@media (max-width: 768px) {
  .product-grid {
    grid-template-columns: 1fr;
  }
}
```

This rule switches product listings to a single column on mobile devices. PMs must advocate for mobile-first approaches, as 58% of global web traffic comes from smartphones.

CSS frameworks like Bootstrap accelerate development with pre-built components but may limit customization. For instance, Bootstrap's grid system uses `container` and `row` classes to create responsive layouts without manual media queries. However, overriding default styles often requires `!important` declarations, complicating maintenance.

JavaScript and Modern ES6+ Features

JavaScript (JS) adds interactivity to static pages through DOM manipulation and API calls. Core concepts include:

- Variables: let (mutable) and const (immutable)
- Arrow functions: Concise syntax for anonymous functions
- Promises: Handle asynchronous operations like data fetching.

ES6 introduced template literals for dynamic strings:

```
const userName = "Alice";

console.log(`Welcome, ${userName}!`); //

Output: Welcome, Alice!
```

This replaces cumbersome concatenation with + operators.

For PMs, understanding JS's role in Single-Page Applications (SPAs) is critical. SPAs load once and dynamically update content, reducing server requests.

However, they complicate SEO unless paired with server-side rendering solutions like Next.js

Frontend Frameworks: React vs. Angular

React employs a component-based architecture using JSX (JavaScript XML) to embed HTML-like syntax:

```
function ProductCard({ name, price }) {
  return (
    <div className="product-card">
      <h3>{name}</h3>
      <p>${price}</p>
    </div>
  );
}
```

Props (`name`, `price`) pass data between components, while hooks like `useState` manage internal state.

Angular uses TypeScript for type safety and dependency injection:

```
@Injectable({
  providedIn: 'root'
})
export class CartService {
  private items: Product[] = [];

  addToCart(product: Product) {
    this.items.push(product);
  }
}
```

PMs should evaluate:

- *React: Faster prototyping for MVPs*
- *Angular: Better for enterprise apps with strict coding standards*

Server-Side Logic and API Design

Server-side scripts handle business logic, database interactions, and authentication. In Node.js, a basic Express.js endpoint looks like:

```
app.get('/api/products', (req, res) => {
  db.query('SELECT * FROM products', (error, results) => {
    if (error) throw error;
    res.json(results);
  });
});
```

This retrieves product data from a SQL database and returns JSON.

PMs must ensure APIs follow REST principles:

- *Resource-based URLs (/products/{id})*
- *Standard HTTP methods (GET, POST)*
- *Statelessness (no client context stored between requests)*

Middleware and Security Practices

Middleware functions intercept requests for logging, authentication, or data transformation:

```
app.use((req, res, next) => {

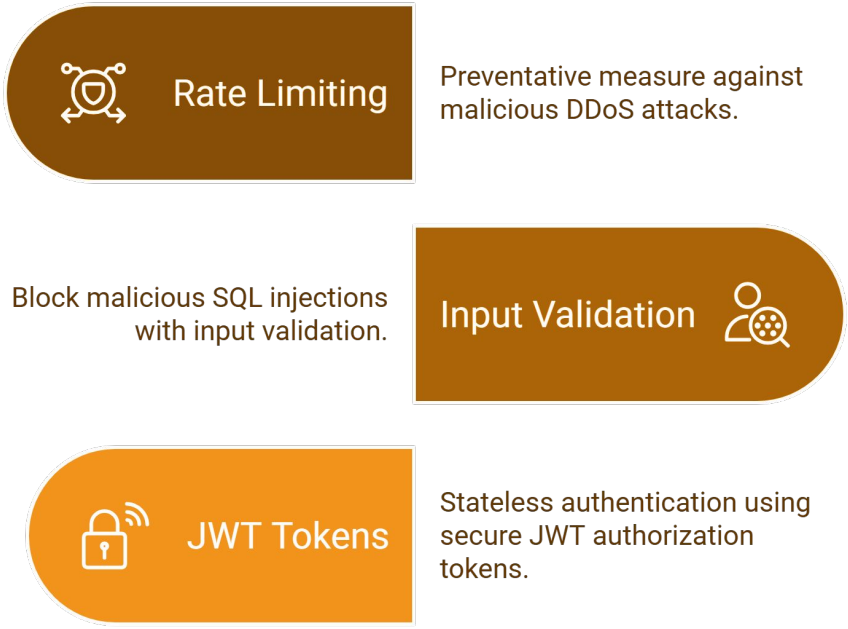
  console.log(`${req.method} request to
${req.path}`);

  next();

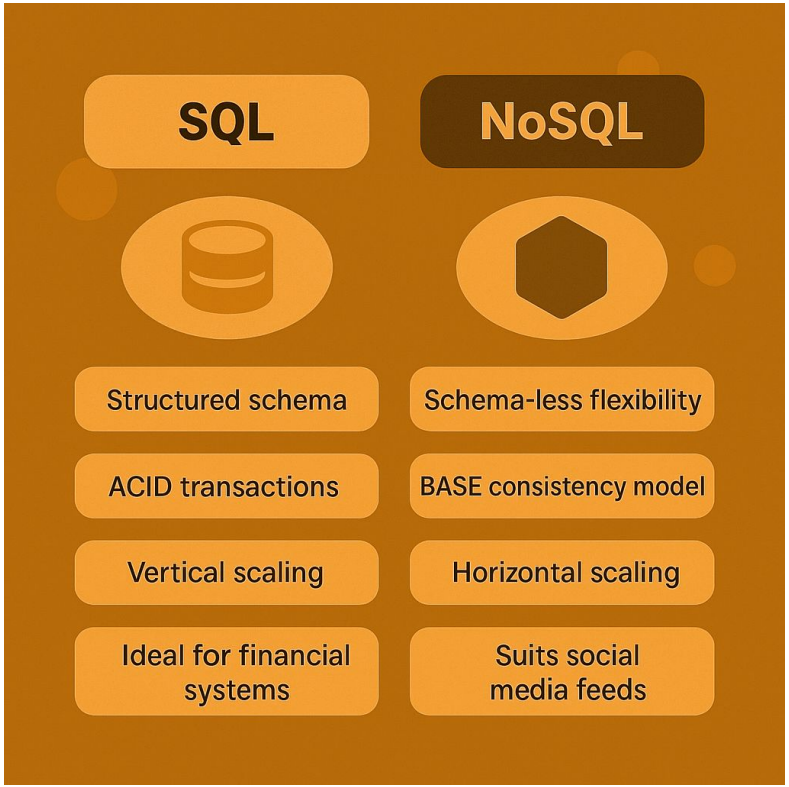
});
```

This logs all incoming requests. For security, middleware like Helmet sets HTTP headers to prevent XSS attacks, while CORS configures cross-origin resource sharing

Security Prioritization for PMs



SQL vs. NoSQL Tradeoffs



Database Schema Design Best Practices

1. **Normalization:** Eliminate redundancy by splitting data into related tables (e.g., separating `users` and `orders`).
2. **Indexing:** Accelerate queries on frequently accessed columns like `user_id`.
3. **Denormalization:** Improve read performance for analytics by duplicating data in read-optimized formats.

An e-commerce schema might include:

- `products` *(product_id, name, price)*
- `users` *(user_id, email)*
- `orders` *(order_id, user_id, product_id)*

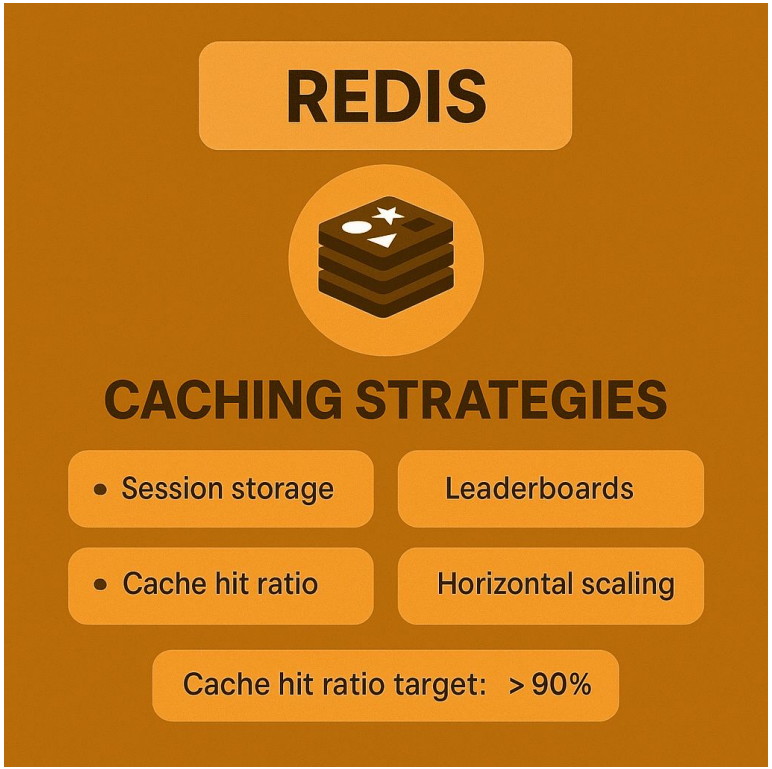
Redis Caching Strategies

Redis reduces database load by storing frequently accessed data in memory.

Common use cases:

- Session storage: Maintain user login states
- Leaderboards: Real-time score updates via sorted sets
- API rate limiting: Track request counts per IP

PMs should monitor cache hit ratios (target >90%) to balance performance gains against memory costs



Technology Stacks Overview

MERN Stack (MongoDB, Express, React, Node.js)

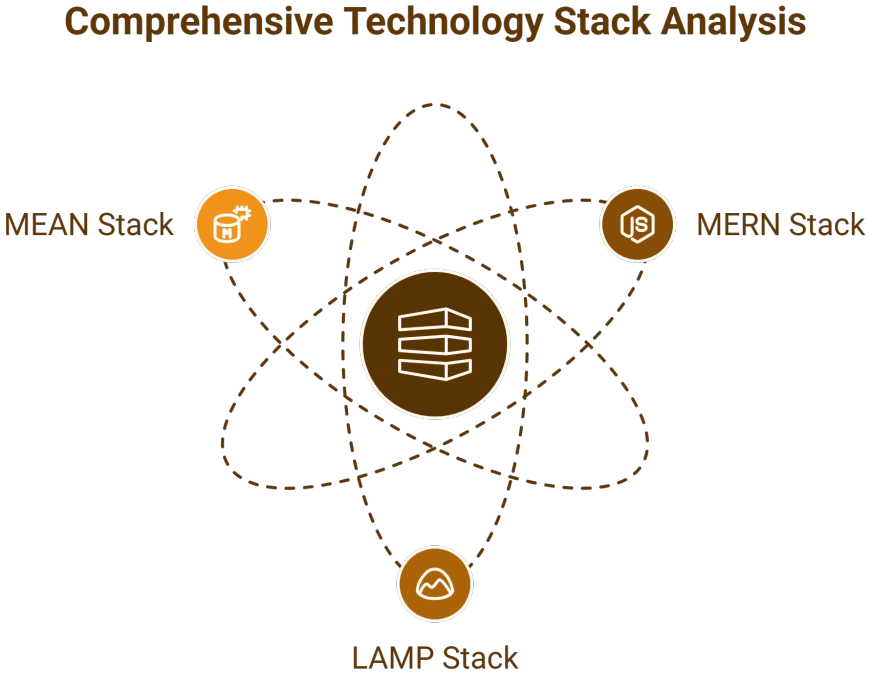
- Pros: Full JavaScript stack, ideal for SPAs
- Cons: MongoDB's eventual consistency unsuitable for financial transactions
- Use Case: Social media MVP with real-time features.

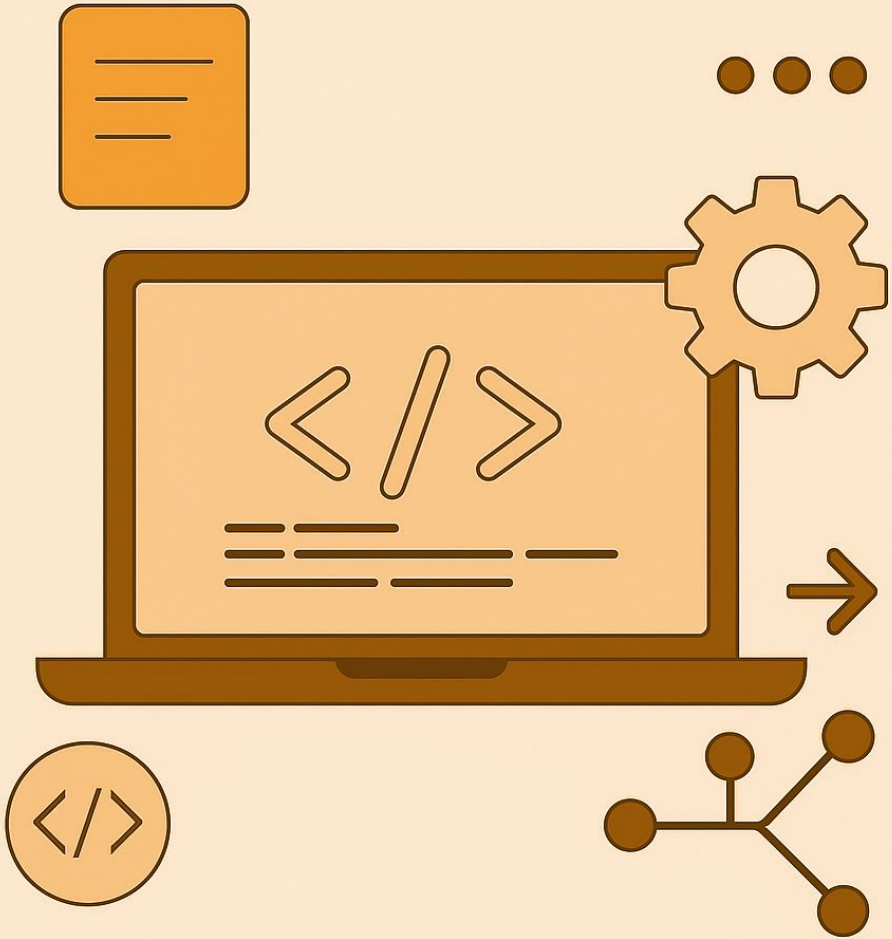
LAMP Stack (Linux, Apache, MySQL, PHP)

- Pros: Mature ecosystem, WordPress compatibility
- Cons: Monolithic architecture complicates scaling
- Use Case: Content-heavy sites like news portals.

MEAN Stack (MongoDB, Express, Angular, Node.js)

- Pros: TypeScript type safety, enterprise-ready
- Cons: Steep Angular learning curve
- Use Case: Enterprise CRM systems





Tech for PM

***PART 3*: Development & Version Control for Product Managers**

Core Principles of REST Architecture

REST (Representational State Transfer) relies on six architectural constraints to ensure scalability and interoperability:

1. ***Uniform Interface:*** Resources are uniquely identified via URIs (e.g., `/users/{id}`), and interactions use standardized HTTP methods (GET, POST). Responses include hypermedia links to guide clients to related actions, fostering discoverability.
2. ***Statelessness:*** Each request contains all necessary context (e.g., authentication tokens), eliminating server-side session storage. This simplifies scaling but requires careful header management.
3. ***Cacheability:*** Responses specify caching rules via headers like `Cache-Control`, reducing server load for repetitive requests (e.g., product catalogs).
4. ***Client-Server Decoupling:*** Frontend and backend evolve independently. For example, a mobile app (client) can update its UI without affecting the API serving data.
5. ***Layered System:*** Proxies or gateways can mediate requests without clients knowing, enabling security policies or load balancing .

HTTP Methods and Idempotency

Method	Purpose	Idempotent?	Safe?
GET	Retrieve resources (e.g., user profile)	Yes	Yes
POST	Create resources (e.g., new order)	No	No
PUT	Replace entire resource (e.g., update user)	Yes	No
PATCH	Partially update resource (e.g., change email)	No	No
DELETE	Remove resource (e.g., delete post)	Yes	No

Authentication and Authorization

OAuth 2.0

OAuth delegates access via tokens instead of sharing credentials. Common flows include:

- Authorization Code Flow: For web apps, exchanges a code for tokens via `/oauth/token`.
- Client Credentials Flow: Machine-to-machine authentication using client ID/secret4. Example scope for fitness data access: `https://www.googleapis.com/auth/fitness.activity.read`.

JSON Web Tokens (JWT)

JWTs encode user claims (e.g., `{"role": "admin"}`) in a signed token. Servers validate signatures without database lookups:

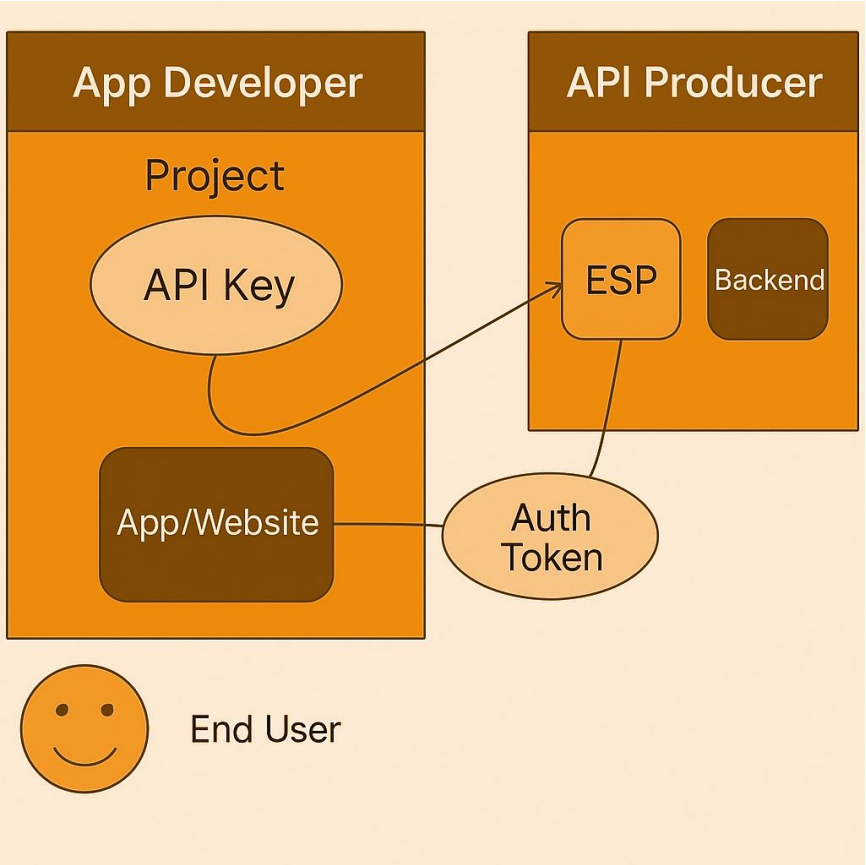
```
// Node.js JWT verification
jwt.verify(token, secretKey, (err, decoded) => {
  if (err) throw new Error("Invalid token");
  req.user = decoded;
});
```

API Keys

Simple but less secure, API keys authenticate clients via headers:

```
curl -X GET https://api.example.com/data -H
"X-API-Key: 12345"
```

Best practices: Use with HTTPS, rate limiting, and IP whitelisting



Webhooks: Event-Driven Communication

Configuring GitHub Webhooks

1. *Navigate to Repository Settings > Webhooks > Add Webhook.*
2. Set **Payload URL** to your endpoint (e.g., `https://ci-tool.com/build-trigger`).
3. Choose events (e.g., `push`, `pull_request`).
4. Add a **Secret** to validate payloads:

```
# Flask validation example
import hmac
def verify_signature(payload, signature, secret):
    digest = hmac.new(secret.encode(), payload, 'sha256').hexdigest()
    return hmac.compare_digest(f'sha256={digest}', signature)
```

Webhook Best Practices

- Idempotent Processing: Handle duplicate webhooks using unique IDs (e.g., `X-Request-ID`).
- Reconciliation Jobs: Periodically sync data to fill gaps during downtime.
- Debounce Prevention: Include unique fields like `updated_at` in payloads.

Webhooks: Event-Driven Communication

How Webhooks Work

Webhooks push real-time data to predefined URLs when events occur (e.g., payment success on Stripe). Contrast this with polling, which repeatedly checks for updates:

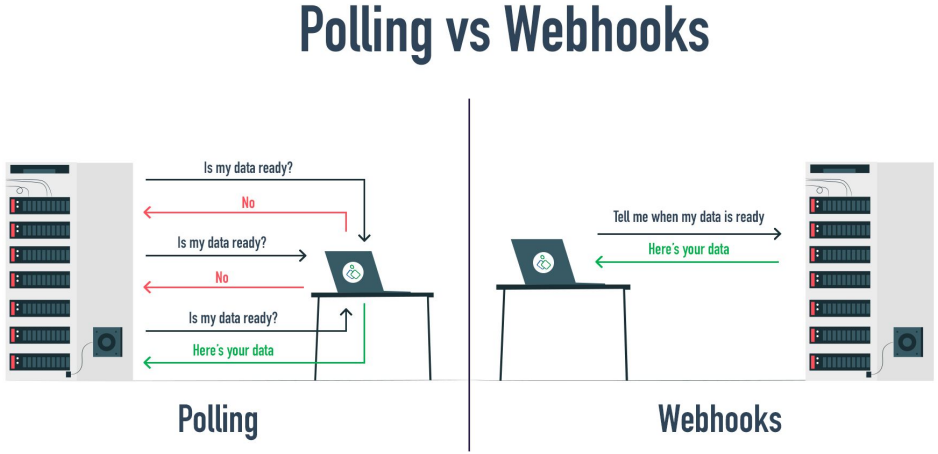
Polling:

```
Client: "Any new orders?"

Server: "No." (Repeats every 5 minutes)
```

Webhook:

```
Server: "New order #4567! Sending details to https://your-app.com/webhook."
```



API Testing with Postman

Building a Request Collection

1. Environment Variables: Store `base_url` or `api_key` for reuse.
2. Pre-request Scripts: Generate dynamic timestamps:

```
pm.environment.set("currentTime", new Date().toISOString());
```

3. Response Validation:

```
pm.test("Status code 200", () => pm.response.to.have.status(200));

pm.test("Response time < 500ms", () => pm.expect(pm.response.responseTime).to.be.below(500));
```

Automating Tests in CI/CD

Export Postman collections as JSON and run via Newman:

```
newman run orders_api.json --env-var "api_key=$API_KEY"
```

Integrate with Jenkins or GitHub Actions to block deployments on test failures.

SDKs vs. APIs: When to Use Each

Criteria	SDK	Raw API
Implementation	Pre-built functions (e.g., `stripe.charge()`)	Manual HTTP calls
Use Case	Mobile apps needing camera access	Cross-platform services
Maintenance	Handle version updates	Directly manage deprecations

Example: The Twilio SDK simplifies sending SMS:

```
from twilio.rest import Client
client = Client(account_sid, auth_token)
client.messages.create(to="+123456789",
from_="+987654321", body="Hello!");
```

Versioning Strategies

1. **URI Versioning:** `https://api.example.com/v2/users`
2. **Header Versioning:** `Accept: application/vnd.example.v2+json` .
3. **Query Parameter:** `https://api.example.com/users?version=2`.

Content Delivery Networks (CDNs)

How CDNs Accelerate Content

1. Edge Caching: Store static assets (images, CSS) in geographically distributed servers.
2. Dynamic Route Optimization: Use real-time traffic data to select the fastest path.
3. DDoS Mitigation: Absorb volumetric attacks via scrubbing centers.

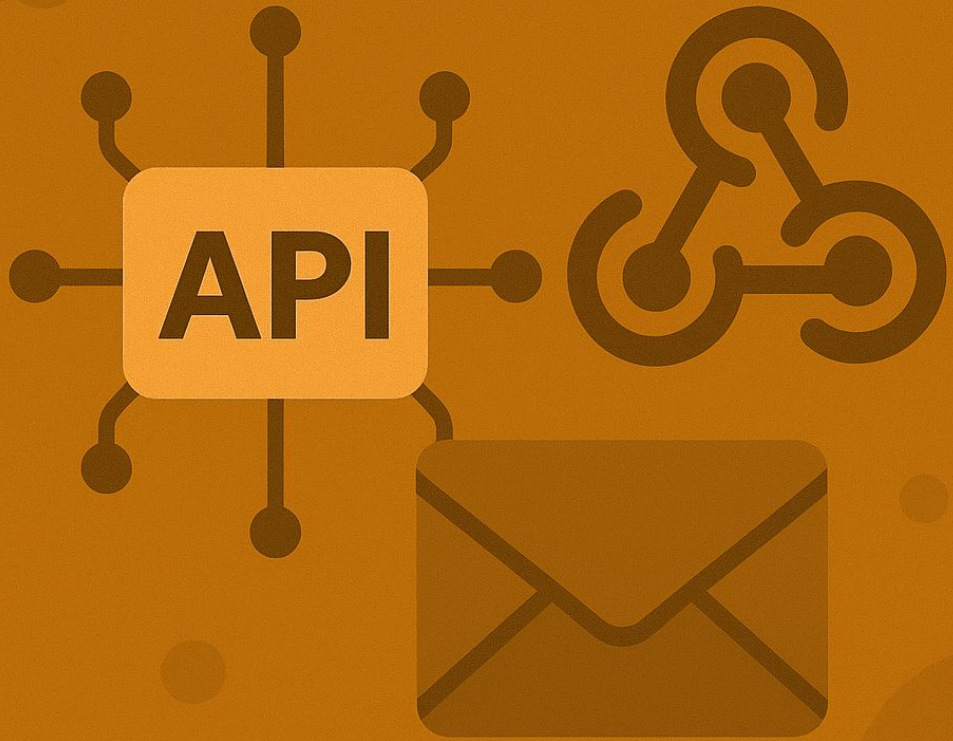
Security with CDNs

- **TLS/SSL Termination:** Offload encryption to edge nodes.
- **Bot Detection**: Block scrapers via headers like `User-Agent`.
- **Geo-blocking**: Restrict access to sanctioned regions.

Key Metrics for PMs

Metric	Target	Tool
Cache Hit Ratio	>90%	Akamai Analytics
Latency	<100ms	Pingdom
Origin Load	<70% capacity	New Relic

Example: A 1MB image served from a CDN edge in Frankfurt to a Berlin user reduces latency from 200ms (US origin) to 20ms



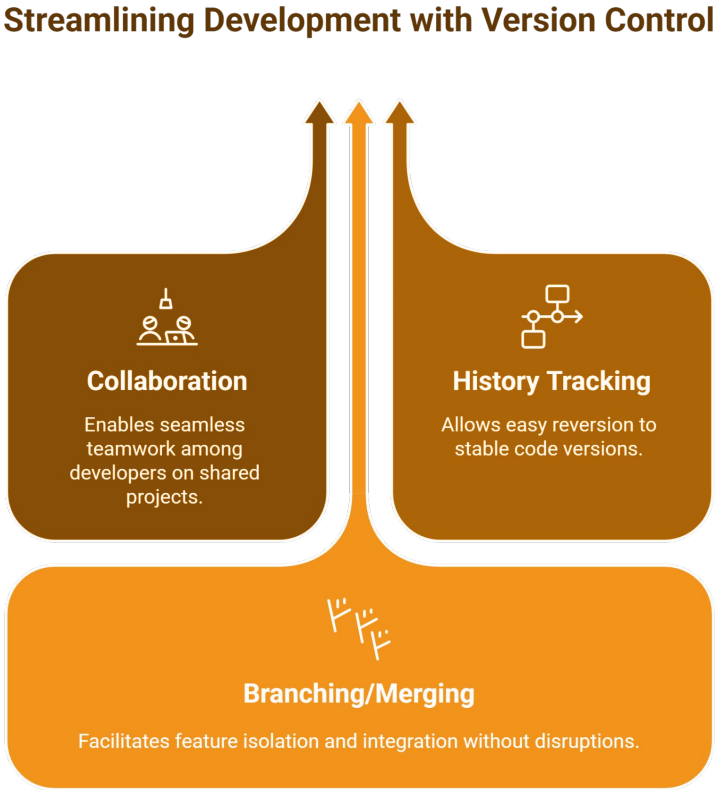
Tech for PM

***PART 4*: APIs, Webhooks, and System Communication for Product Managers**

Understanding Version Control Systems

Version control systems (VCS) track changes to code over time, enabling collaboration and error recovery.

Git, created by Linus Torvalds in 2005, is the most widely used distributed VCS. Unlike centralized systems (e.g., SVN), Git allows every developer to maintain a full repository history locally, enabling offline work and faster operations.



Core Git Concepts & Workflow

Initializing a Repository

Create a new Git repository or clone an existing one:

```
git init        # Initialize a local repo

git clone https://github.com/user/repo.git
#Clone a remote repo
```

Staging & Committing Changes

Staging: Select files to include in the next commit using git add.

```
git add index.html          # Stage a single file
git add .                   # Stage all modified files
git commit -m "Add homepage layout"
```

Remote Repositories

Link local repos to platforms like GitHub or GitLab:

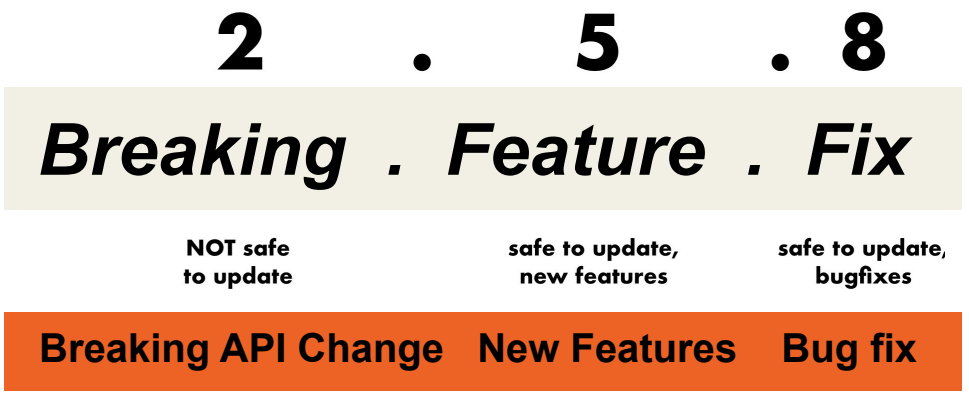
```
git remote add origin
https://github.com/user/repo.git
git push -u origin main
# Push commits to remote
```

Semantic Versioning (SemVer)

SemVer (`Major.Minor.Patch`) standardizes version numbers to communicate risk:

- **Major (X.0.0)**: Breaking changes (e.g., API redesign).
- **Minor (1.X.0)**: Backward-compatible features (e.g., new endpoints).
- **Patch (1.0.X)**: Bug fixes with no new functionality.

Example: A PM prioritizing stability might delay upgrading from v2.1.5 to v3.0.0 until compatibility layers are tested.



Branching Strategies for Scalable Development

Git Flow

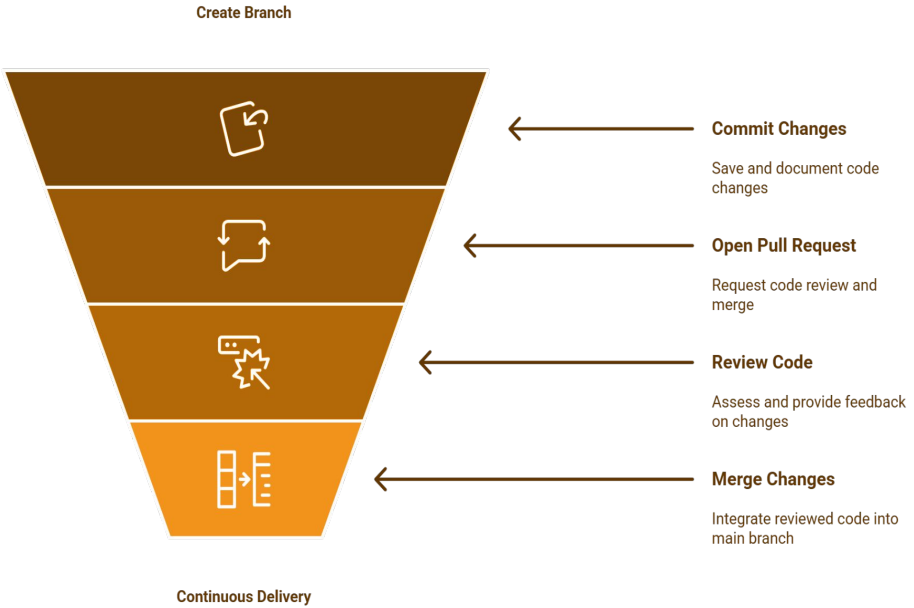
- **Main/Master**: Production-ready code.
- **Develop**: Integration branch for features.
- **Feature Branches**: Short-lived branches for individual tasks

GitHub Flow

Simplified workflow for continuous delivery:

1. Create a branch from main.
2. Commit changes and open a pull request (PR).
3. Merge after review.

PM Consideration: GitHub Flow suits teams deploying multiple times daily, while Git Flow benefits complex projects with staggered releases.



Branching Strategies for Scalable Development

Resolving Merge Conflicts

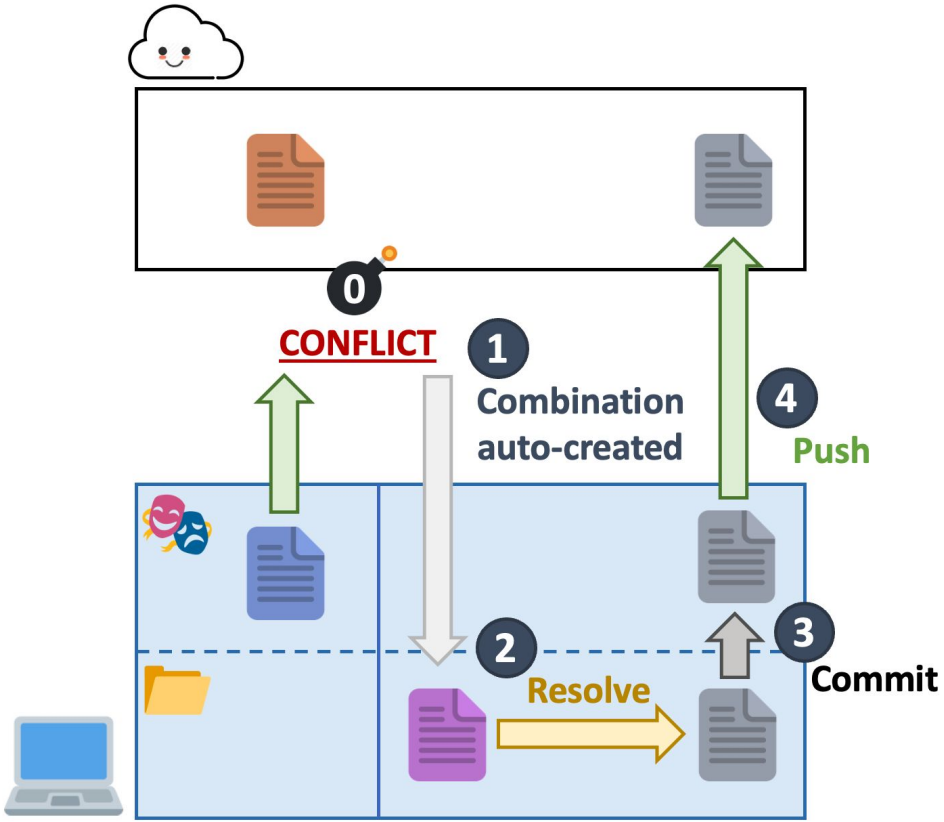
Conflicts occur when two branches modify the same code block. To resolve:

1. Identify conflicted files via `git status`.
2. Edit files to keep desired changes, removing conflict markers (`<<<<<<<`, `=======`, `>>>>>>>`).
3. Stage and commit the resolved files:

```
git add conflicted-file.js

git commit -m "Resolve merge conflict"
```

Tool Tip: Use VS Code's built-in merge editor or GitHub's web interface for visual resolution.



Feature Flags: Controlled Rollouts

Resolving Merge Conflicts

Feature flags (toggles) enable runtime control of features without redeploying code.

Lifecycle Management:

1. Development: Wrap new features in flags.
2. Testing: Enable for internal users via targeting.
3. Release: Gradual rollout (10% → 100% traffic).
4. Cleanup: Remove flags after full adoption.

Best Practices:

- Short-Lived Flags: Avoid technical debt by sunsetting flags post-release
- Server-Side Evaluation: Protect sensitive logic from client-side manipulation.

Code Example:

```
if (featureFlags.isEnabled('new_checkout')) {
  loadNewCheckout();
} else {
  loadLegacyCheckout();
}
```

Jenkins: Automating Builds & Tests

Integration: Connect Jenkins to GitHub to trigger builds on push events



Pipeline Structure

Define stages in a `Jenkinsfile`:

```
pipeline {
  agent any
  stages {
    stage('Build') {
      steps {
        sh 'npm install'
      }
    }
    stage('Test') {
      steps {
        sh 'npm test'
      }
    }
    stage('Deploy') {
      steps {
        sh 'npm deploy'
      }
    }
  }
}
```

GitHub Actions: Cloud-Native CI/CD

Define workflows in `.github/workflows/main.yml`:

```
name: CI

on: [push]

jobs:

  build:

    runs-on: ubuntu-latest

    steps:

      - uses: actions/checkout@v4

      - run: npm install

      - run: npm test
```

Advantages:

- *Native Integration: Directly tied to GitHub repositories.*
- *Matrix Builds: Test across multiple OS/node versions.*

Cion & Orchestration

Docker: Packaging Applications

Dockerfile Basics

```
FROM node:18-alpine

WORKDIR /app

COPY package*.json ./

RUN npm install

COPY . .

CMD ["npm", "start"]
```

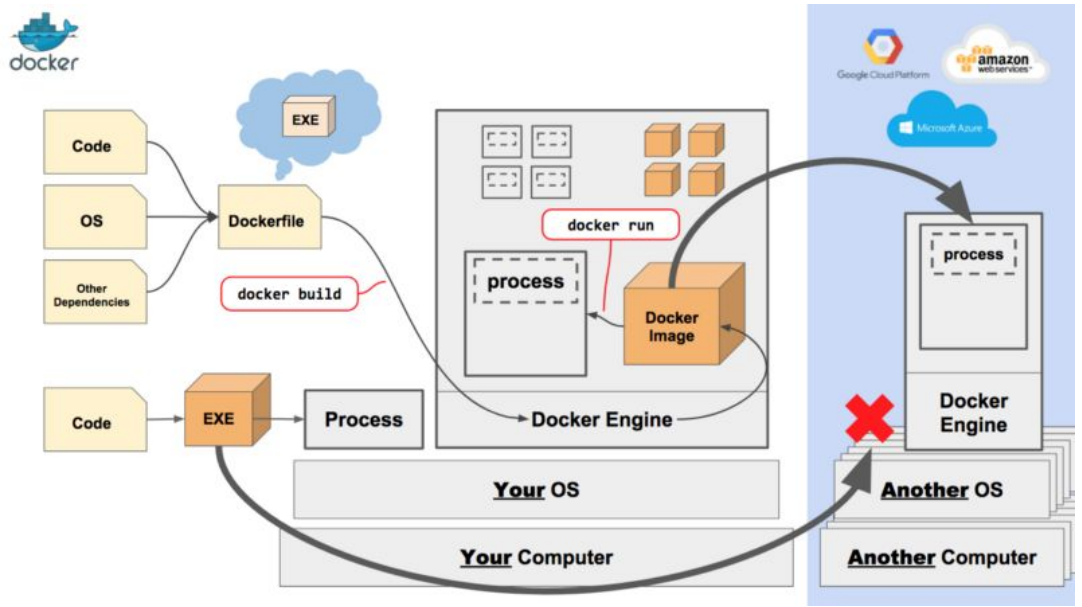
Volumes: Persist data between container restarts:

```
docker run -v /app/data my-app
```

Build and run:

```
docker build -t my-app .

docker run -p 3000:3000 my-app[10]
```



Kubernetes: Scaling Containers

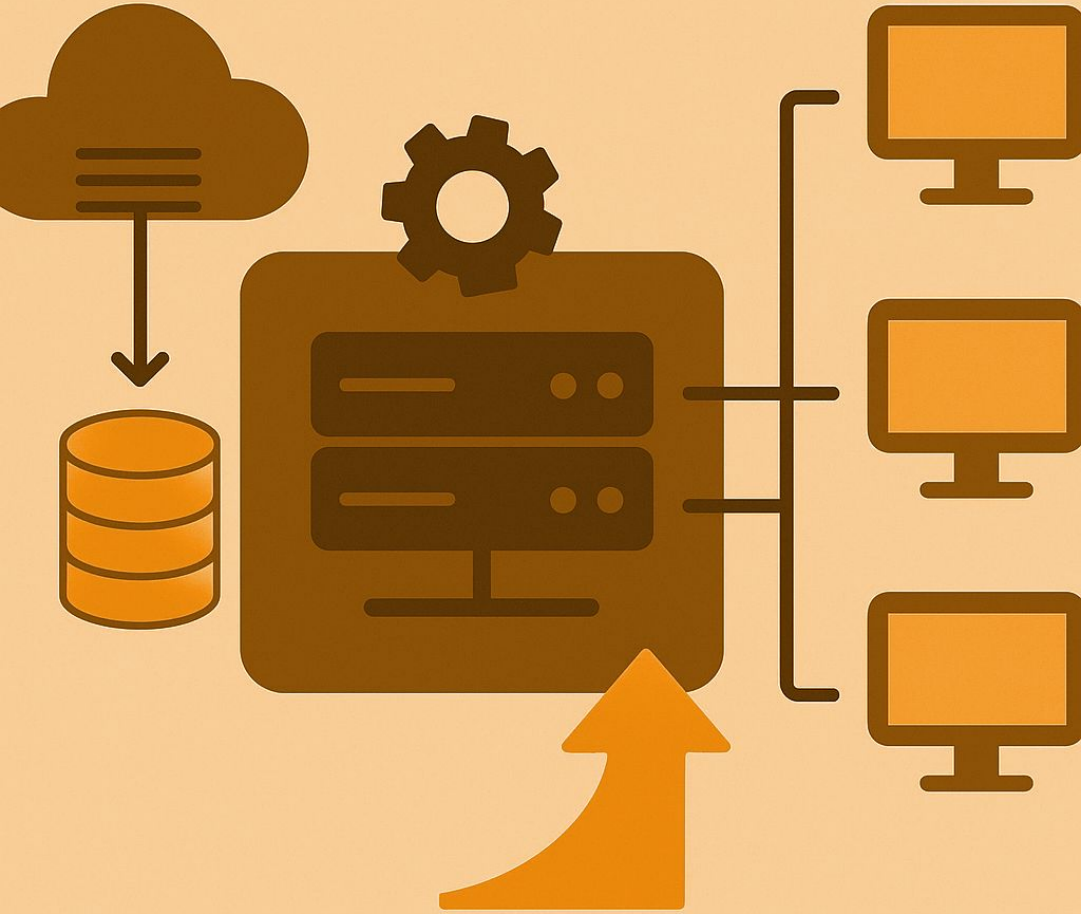
Key Components:

- Pods: Smallest deployable units (1+ containers).
- Deployments: Manage pod replicas and updates.
- Services: Expose pods via DNS or IP.

PM Use Case: Auto-scaling based on CPU usage to handle traffic spikes.

Example Deployment:

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: my-app
spec:
  replicas: 3
  selector:
    matchLabels:
      app: my-app
  template:
    metadata:
      labels:
        app: my-app
    spec:
      containers:
      - name: my-app
        image: my-app:latest
        ports:
        - containerPort: 3000
```



Tech for PM

***PART 5*: System Design & Scalability for Product Managers**

What is System Design for Product Managers?

System design for product managers differs significantly from system design for engineers. While engineers focus on detailed technical implementations, PMs need to understand how technical architecture decisions impact user experience, business objectives, and product scalability.

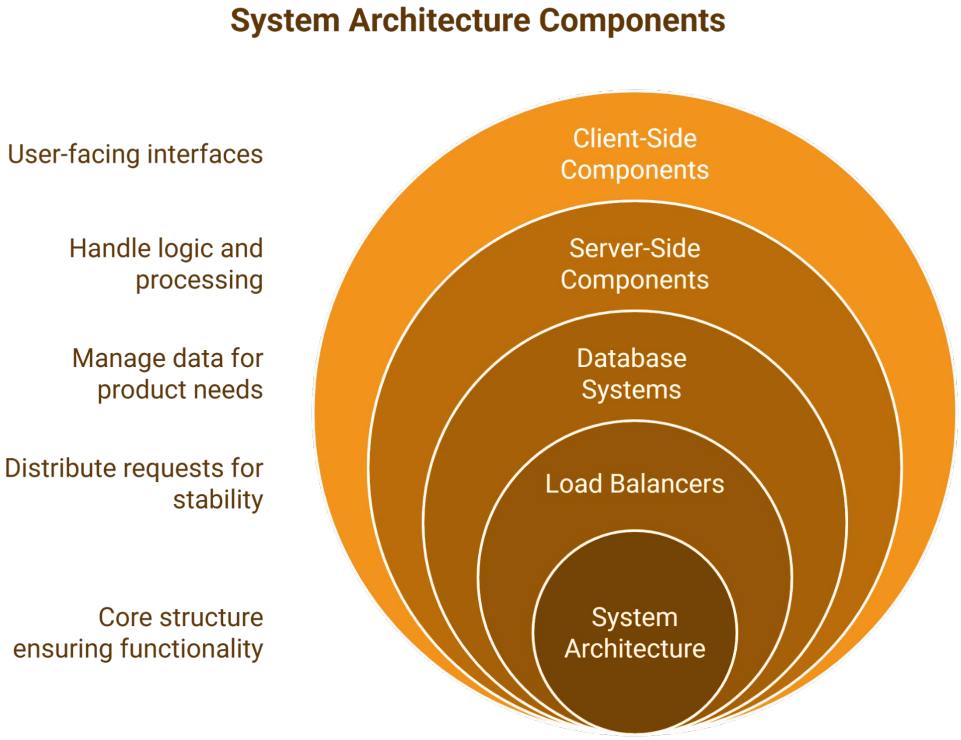
Product managers should view system design as a strategic tool for understanding how their product requirements translate into technical components and constraints.

This understanding enables better collaboration with engineering teams and more informed product decisions throughout the development lifecycle.

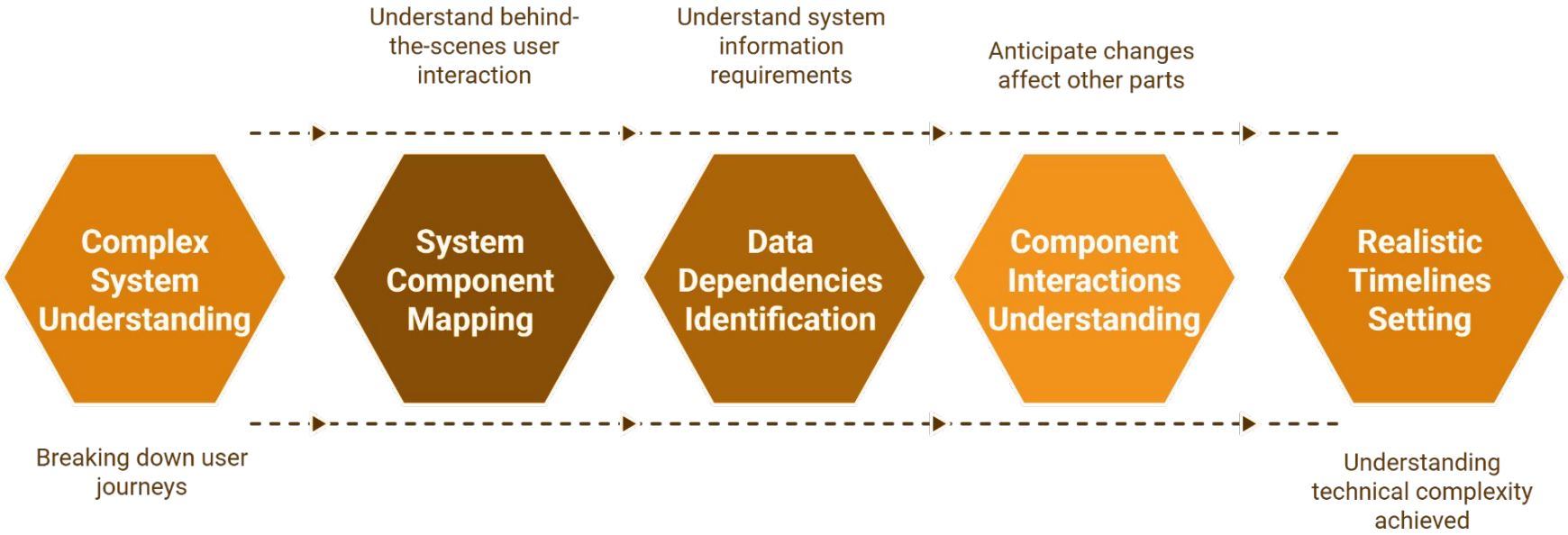
Understanding High-Level System Design

Core Components of System Architecture

Understanding the fundamental building blocks of any system helps **product managers communicate effectively with engineering teams and make informed architectural decisions.**



User Flows and Technical Requirements Translation

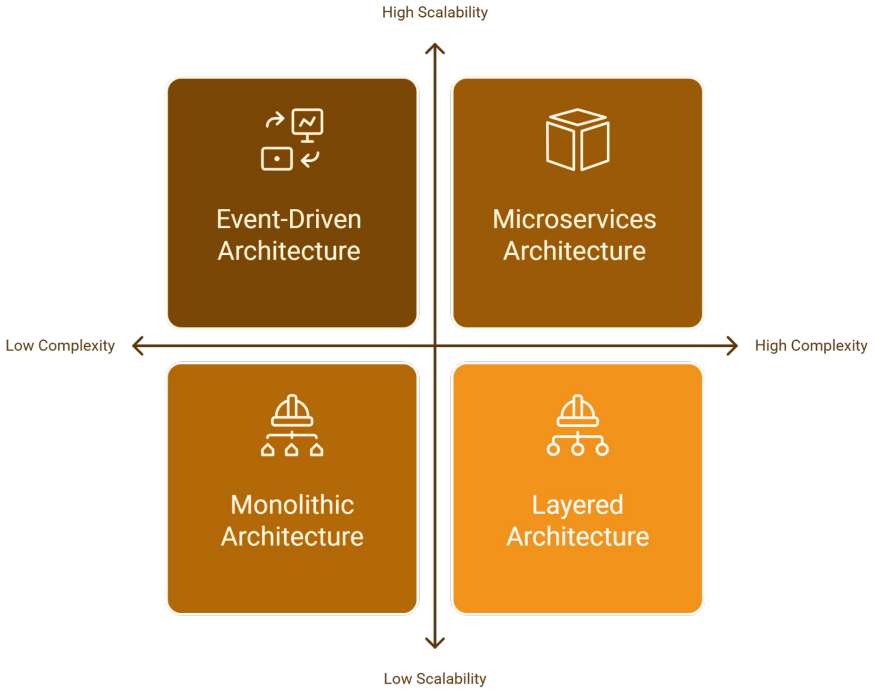


Choosing the Right Architecture Patterns

Architecture patterns provide proven solutions to common system design challenges.

For product managers, understanding these patterns helps in making informed decisions about technical approaches and understanding the trade-offs involved in different architectural choices.

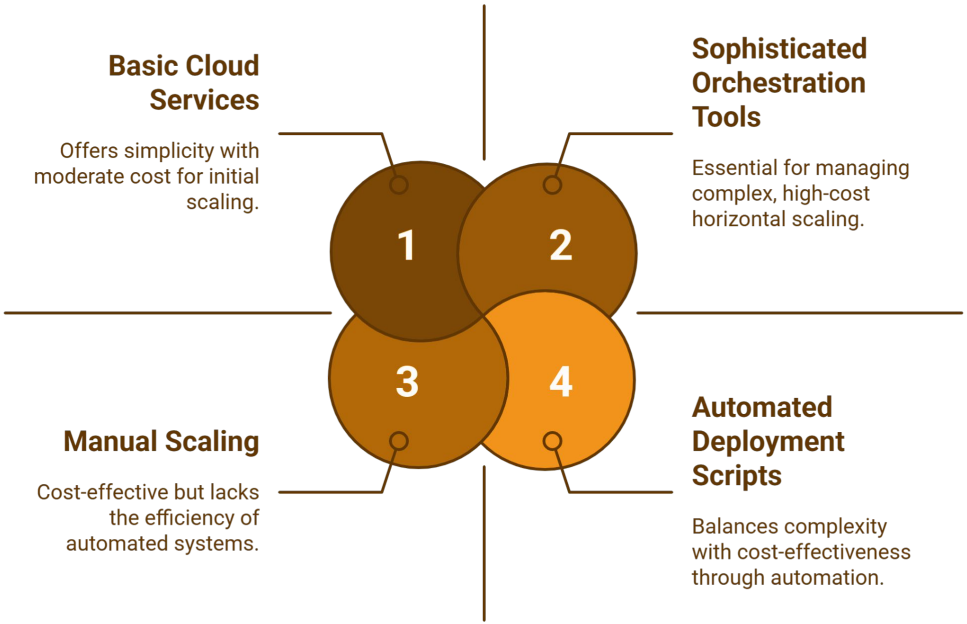
- **Monolithic Architecture:** Single unit, simple coordination but scaling individual components is difficult.
- **Microservices Architecture:** Independent services, flexible and scalable but complex coordination and monitoring.
- **Event-Driven Architecture:** Components react to events, loose coupling and real-time features.
- **Layered Architecture:** Hierarchical layers, separation of concerns and maintainability, clear boundaries for product managers.



Scaling Systems: Horizontal vs Vertical Approaches

Horizontal Scaling (Scaling Out)

Horizontal scaling involves **adding more machines or instances to handle increased demand rather than making existing machines more powerful.** This approach offers greater flexibility and growth potential but introduces architectural complexity and coordination challenges.



Scaling Systems: Horizontal vs Vertical Approaches

Understanding Scalability Fundamentals

Scalability represents your system's ability to handle increasing demands without degrading performance or user experience.

For product managers, understanding scalability concepts is crucial for anticipating growth challenges and making informed decisions about technical investments and trade-offs.

Scalability in product terms is the ability of a system to maintain performance and availability with increased usage, impacting user experience and growth opportunities.

Key performance metrics for product managers are **response time, throughput, and availability**.

Growth planning requires anticipating how **user acquisition, feature complexity, and data growth** will strain system components, necessitating a balance between growth targets and technical feasibility.

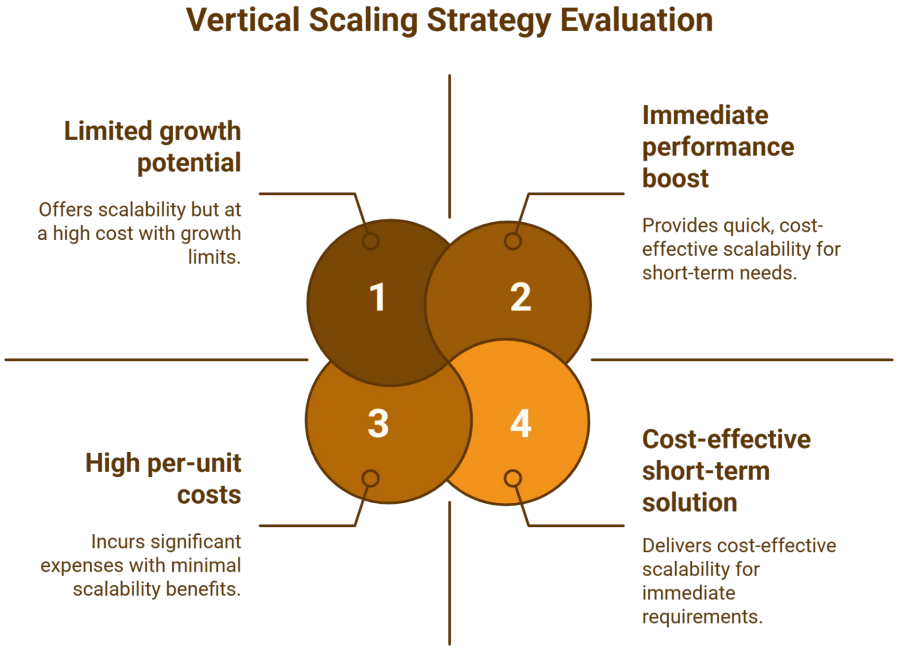
Scaling Systems: Horizontal vs Vertical Approaches

Vertical Scaling (Scaling Up)

Vertical scaling involves **increasing the capacity of existing infrastructure by adding more powerful hardware—faster processors, additional memory, or larger storage systems.** This approach represents the simpler path to handling increased demand but comes with inherent limitations and trade-offs.

Vertical scaling suits predictable, moderate growth and systems needing tight coordination, like databases. It offers immediate performance gains without architectural changes but becomes costly long-term due to premium pricing for powerful hardware and finite scaling limits.

Product managers should consider these economic and technical constraints for long-term product goals.



Scaling Systems: Horizontal vs Vertical Approaches

Auto-scaling and Dynamic Resource Management

Modern cloud platforms enable systems to **automatically adjust capacity based on current demand,** providing cost efficiency and performance optimization without manual intervention.

For product managers, understanding auto-scaling capabilities and limitations helps in setting realistic expectations and planning for growth scenarios.

Demand-Based Scaling adjusts system capacity automatically using metrics like CPU usage or request volume, optimizing costs and ensuring performance. Product Managers should understand which features trigger these scaling events.

Performance Monitoring for Product Impact necessitates metrics reflecting user experience and business performance. Auto-scaling relies on these signals, not just technical metrics, to align scaling with user impact. This understanding helps PMs define appropriate scaling triggers and thresholds with engineering teams.

Databases in Scalable Systems

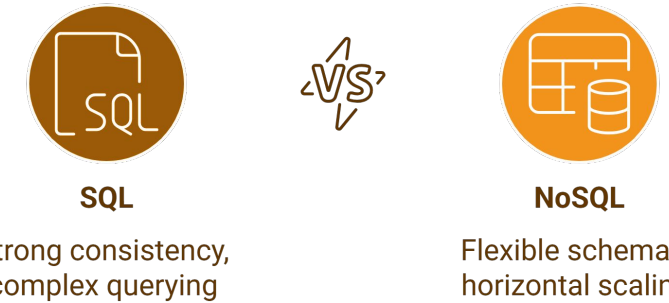
SQL vs NoSQL: Strategic Considerations for PMs

Choosing between SQL and NoSQL databases affects product features, development speed, and scaling. Product managers must understand these trade-offs for informed technical decisions and feature feasibility.

SQL databases offer strong consistency, complex querying, and mature support, ideal for financial transactions, user management, and complex reporting.

NoSQL databases provide flexible schemas, horizontal scaling, and optimized performance for content management, real-time analytics, and evolving data. They suit content-heavy products, social platforms, and recommendation systems.

Hybrid approaches use different databases for different needs, optimizing performance and speed but increasing operational complexity that PMs must consider.

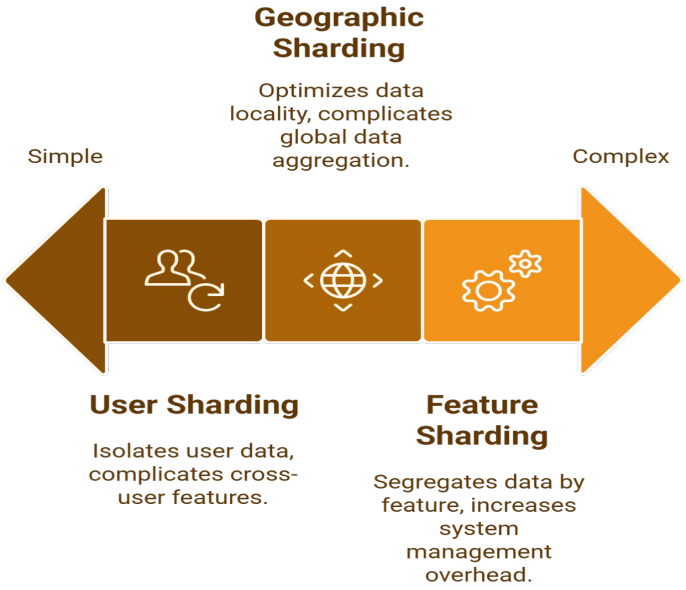


Database Sharding Strategies

Sharding distributes a database across multiple servers for horizontal scaling, improving performance for data-intensive features but complicating data management and application logic. Horizontal partitioning, dividing data by criteria like user ID, requires routing requests to specific shards and affects system complexity.

Sharding strategies like user-based, geographic, and feature-based have implications for development, user experience, and cross-functional capabilities.

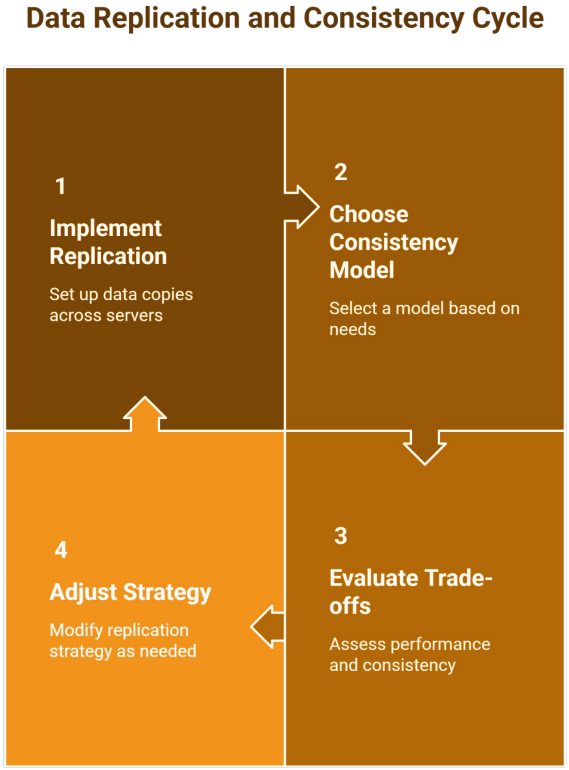
While enhancing scalability, sharding introduces trade-offs in maintaining data consistency and increases the complexity of features requiring cross-shard data.



Data Replication and Consistency Models

Data replication copies data across servers for better performance, availability, and fault tolerance. Primary-secondary replication designates a primary server for writes and secondaries for reads, improving read performance and providing backups with strong write consistency. Eventually consistent systems allow temporary inconsistencies for better performance and availability, suitable for features like social media feeds.

Understanding consistency trade-offs is crucial as strong consistency impacts performance while eventual consistency might show outdated information; PMs must balance these based on feature needs and user expectations.



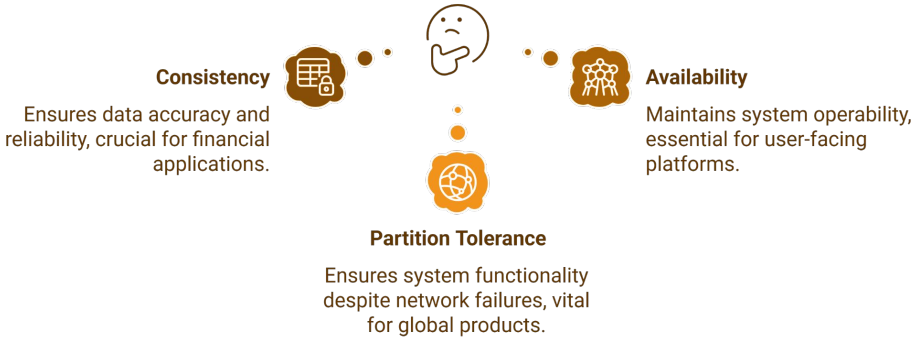
Databases in Scalable Systems

CAP Theorem and Practical Implications

The CAP Theorem outlines the trade-offs in distributed systems, stating that only two of the following guarantees can be fully met: Consistency (all nodes see the same data), Availability (system remains operational), and Partition Tolerance (system operates despite network failures).

Product managers must understand these trade-offs to make informed architectural decisions. Real-world scenarios necessitate prioritizing certain guarantees; for example, banking prioritizes consistency, while social media often prioritizes availability and partition tolerance.

Product decision frameworks help PMs evaluate these trade-offs based on business needs, aligning critical financial features with consistency and user-facing features with availability, while global products consider partition tolerance.



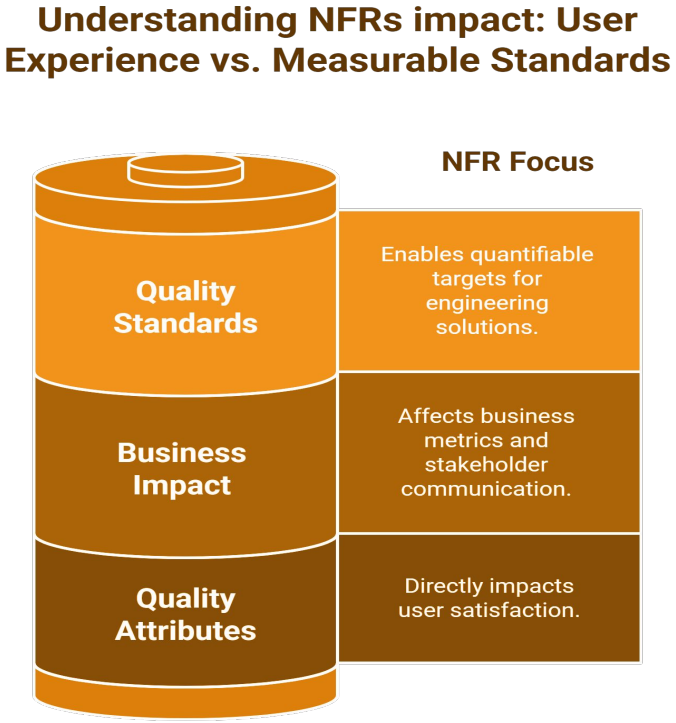
Non-Functional Requirements (NFRs)

Understanding NFRs from a Product Perspective

Non-functional requirements (NFRs) define system performance and significantly impact user experience, business performance, and competitive standing for product managers. Quality attributes like response time, availability, and reliability directly affect user satisfaction and adoption.

Understanding the business impact of technical quality, such as website loading time on conversion rates, is crucial.

Establishing measurable quality standards with specific targets, like API call completion times, is necessary for guiding engineering and evaluating product success.

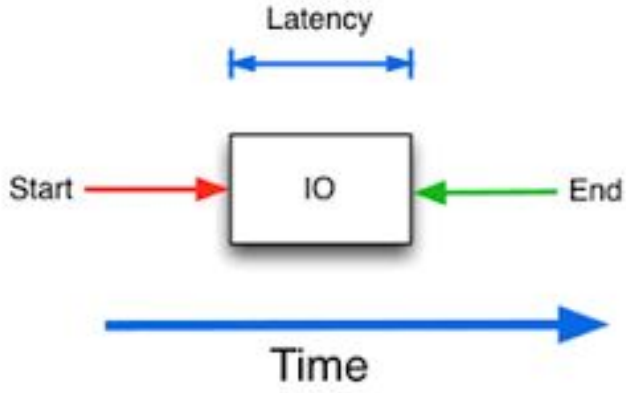


Non-Functional Requirements (NFRs)

Latency and Performance Optimization

Latency, the delay between user actions and system responses, impacts user experience. Product managers should consider user-perceived performance, influenced by factors beyond actual response time, and employ techniques like progressive loading.

Balancing feature richness with performance budgets is crucial, requiring trade-offs. Geographic and infrastructure considerations, such as using CDNs, are vital for global user experience.

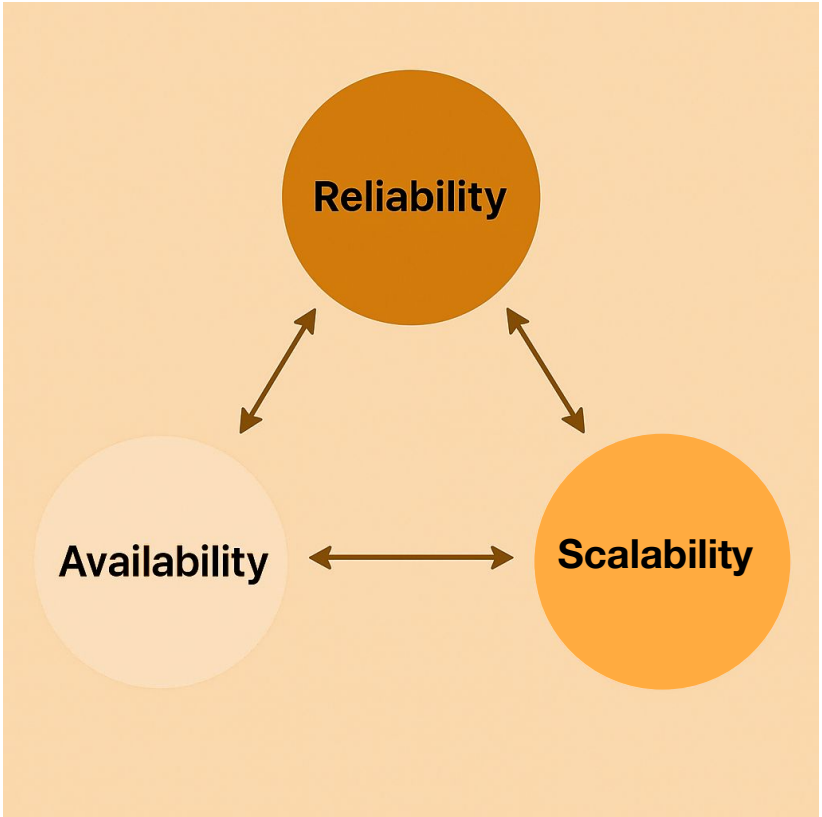


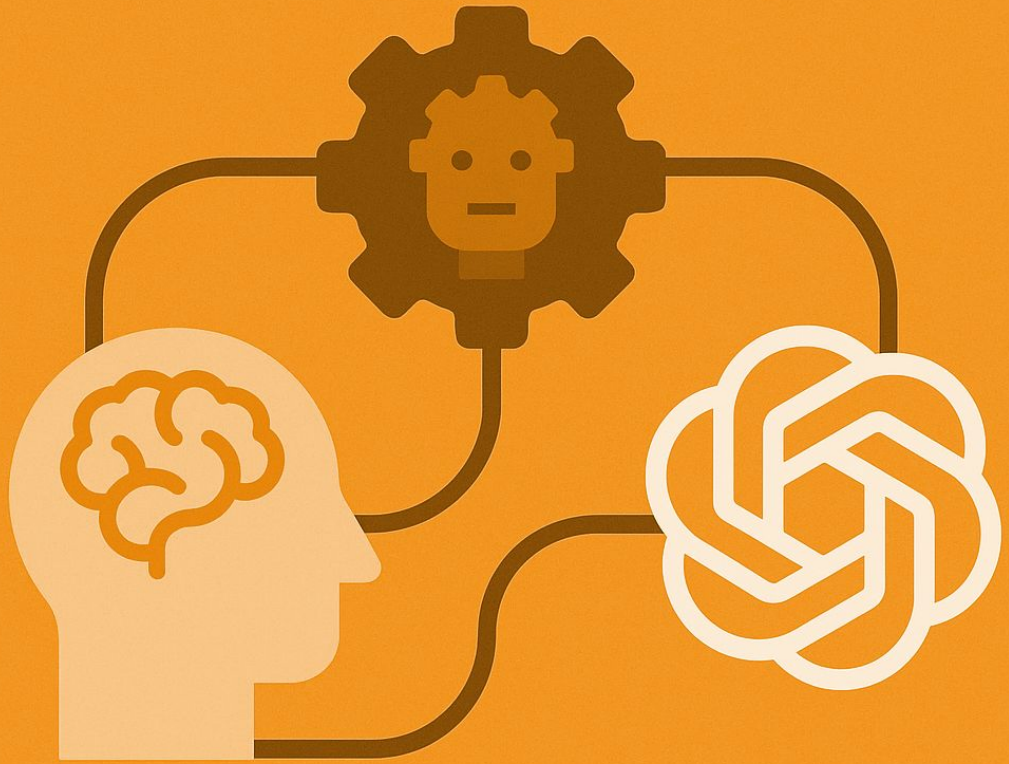
Non-Functional Requirements (NFRs)

Availability and Reliability Design

System availability, crucial for user access and functionality, necessitates redundancy and failure analysis. Uptime requirements and business impact, like revenue loss for e-commerce and engagement drops for social platforms, dictate availability targets and infrastructure investments.

Fault tolerance and graceful degradation ensure continued core functionality despite component failures, minimizing user impact. Disaster recovery planning, with RTO and RPO targets, dictates recovery speed and potential data loss, informing infrastructure decisions.





Tech for PM

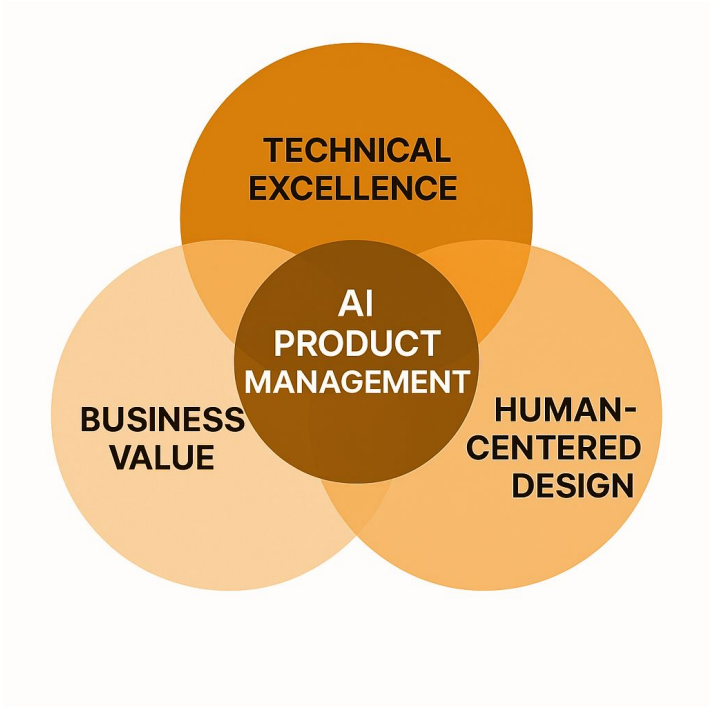
***PART 6*:Tech Concepts every PM should know**

AI, Machine Learning, and Deep Learning

Foundational Distinctions and PM Relevance

Artificial Intelligence (AI) encompasses systems designed to mimic human cognitive functions like problem-solving and decision-making.

For product managers, AI represents a toolkit for automating complex workflows, enhancing user personalization, and deriving insights from unstructured data.

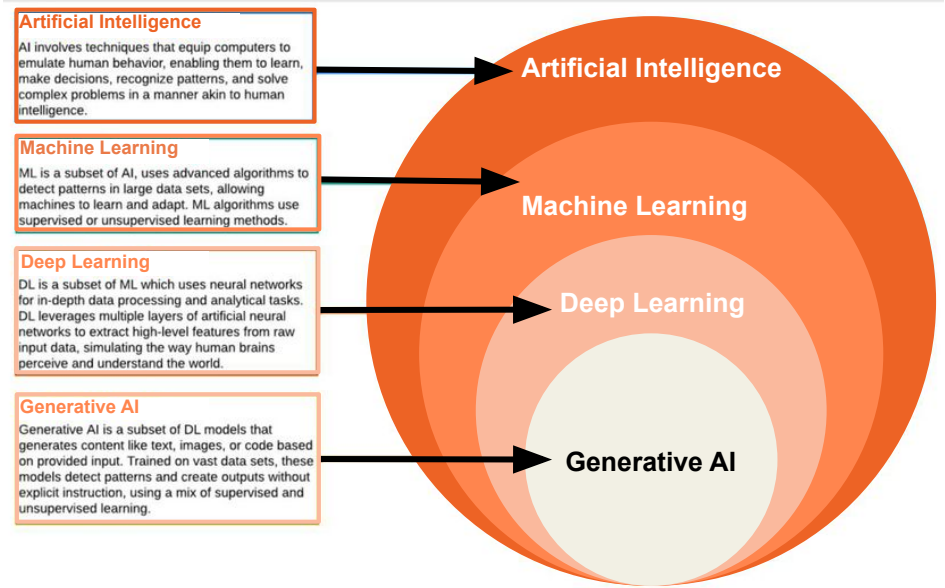


AI & ML Fundamentals

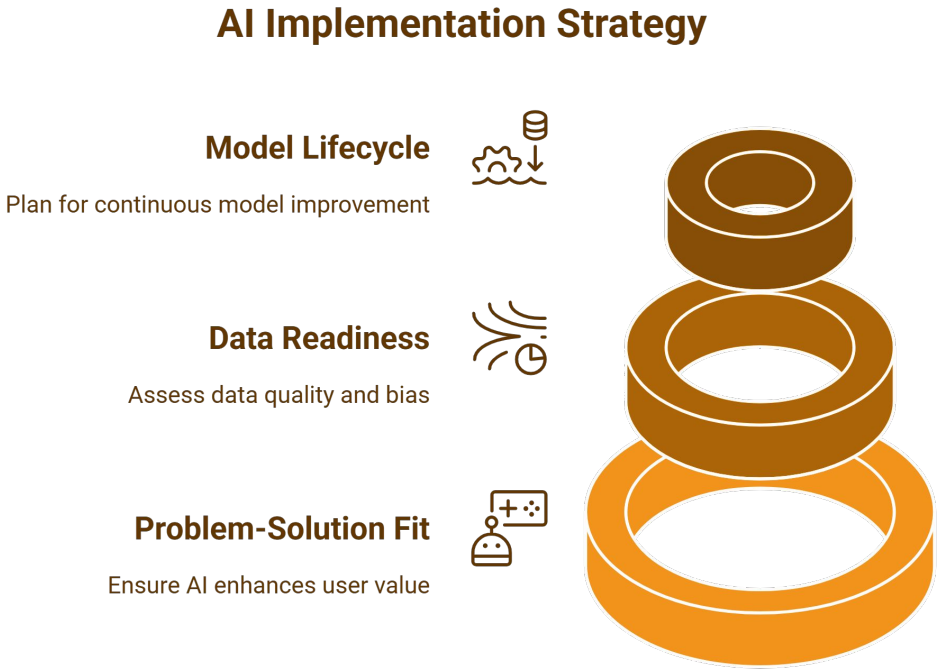
Machine Learning (ML), a subset of AI, enables systems to improve task performance through data exposure rather than explicit programming

PMs must understand three core paradigms:

- **Supervised Learning:** Uses labeled datasets for prediction tasks like customer churn analysis.
- **Unsupervised Learning:** Identifies patterns in unlabeled data for market segmentation.
- **Reinforcement Learning:** Optimizes decisions through trial-and-error feedback loops, critical for adaptive systems like recommendation engines.



Deep Learning employs multi-layered neural networks to process complex data hierarchies. Unlike traditional ML requiring manual feature engineering, **deep learning autonomously extracts features from raw data**—enabling breakthroughs in image recognition (CNNs) and sequence modeling (RNNs).



Generative AI: Beyond Content Creation

Architectural Distinctions

Generative AI differs from discriminative models by learning joint probability distributions (P(X,Y)) rather than conditional probabilities (P(Y|X)).

This enables novel content generation but introduces challenges like hallucination risks. Key architectures PMs should understand:

- **Transformers:** Self-attention mechanisms enabling context-aware text generation (e.g., GPT-4)
- **Diffusion Models:** Progressive noise removal for high-fidelity image synthesis

Ethical Considerations: Implement bias detection frameworks like **IBM's AI Fairness 360** and establish clear content moderation policies

PM Use Cases and Guardrails

Application	Value Proposition	Risk Mitigation
Content Personalization	Dynamic product descriptions tailored to user behavior	Implement output watermarking
Synthetic Data Generation	Augment training datasets for rare scenarios	Establish data provenance trails
Code Generation	Accelerate feature development cycles	Rigorous security scanning

Service Model Strategic Selection

Model	PM Considerations
IaaS (AWS EC2)	Full control but requires DevOps expertise
PaaS (Heroku)	Faster MVP development with vendor lock-in risks
Serverless (AWS Lambda)	Cost-efficient for sporadic workloads

Cost Optimization Strategies:

- Implement auto-scaling policies matched to usage patterns
- Use spot instances for non-critical batch processing
- Negotiate committed use discounts with cloud providers

Cost Optimization Strategies:

Auto-Scaling Policy Implementation

Effective auto-scaling requires aligning scaling triggers with workload patterns. For predictable traffic spikes (e.g., e-commerce holiday sales), schedule-based scaling pre-allocates resources using historical data. For variable workloads (e.g., streaming services), metric-driven scaling (CPU/memory thresholds) dynamically adjusts capacity.

Advanced Tactics:

- **Predictive Scaling:** Leverage AWS machine learning to forecast demand 24 hours ahead, reducing overprovisioning by 35%.
- **Mixed Instance Policies**: Combine On-Demand (60%), Spot (30%), and Reserved (10%) instances to balance cost-reliability tradeoffs

Spot Instance Utilization

Non-critical batch jobs (data analytics, rendering) achieve 70-90% cost savings using spot instances. Implementation requires:

- **Automated Fallback**: Tools like CAST AI shift workloads to On-Demand during spot shortages.
- **Fault-Tolerant Design**: Checkpointing systems save intermediate results hourly to mitigate interruption impacts.

Blockchain: Trust Architecture

Beyond Cryptocurrency Applications

Supply Chain Provenance

De Beers' Tracr platform tracks 100% of diamonds from mine to retail using:

- Immutable Records: SHA-256 hashing prevents tampering of 4K-resolution diamond images.
- Selective Transparency: Suppliers control data visibility via permissioned blockchain nodes.

PM Checklist:

- *Map physical-digital handoff points (e.g., RFID tagging stations)*
- *Integrate IoT sensors for real-time condition monitoring (temperature/humidity)*

Smart Contract Automation

B2B:

- **Automated Reconciliation:** Smart contracts cross-verify shipment GPS data with invoices, reducing disputes by 40%.
- **Dynamic Pricing:** Raw material quality data from IoT triggers contract price adjustments

Consensus Tradeoffs

Mechanism	Energy Use	Security Model
PoW (Bitcoin)	707 kWh/tx	51% Hash Power
PoS (Ethereum)	0.01 kWh/tx	34% Staked ETH

AR/VR Hardware Constraints Mitigation

Hardware Ecosystem Considerations

Technology	PM Use Cases	Hardware Constraints
AR (Microsoft HoloLens)	Industrial maintenance guides	Field-of-view limitations
VR (Meta Quest)	Immersive training simulations	Motion sickness mitigation
MR (Apple Vision Pro)	Hybrid retail experiences	High development costs

Content Strategy: Prioritize 3D asset optimization (polygon counts, texture mapping) and spatial UI conventions

Field-of-View Optimization (Microsoft HoloLens)

- *UI Spatial Anchoring: Fix critical HUD elements within 60° central FOV to prevent clipping.*
- *Contextual Depth Cues: Use parallax scrolling and occlusion shading to enhance depth perception in narrow 35° FOV.*

Motion Sickness Reduction (Meta Quest 3)

Comfort Modes:

- *Vignette Dimming: Reduces peripheral vision stimulation during movement.*
- *Snap Turning: 45° instantaneous rotations minimize vestibular conflict.*

Frame Rate Optimization: *Maintain 90 FPS via 3D asset LOD systems (high:20K polys → low:2K polys)*

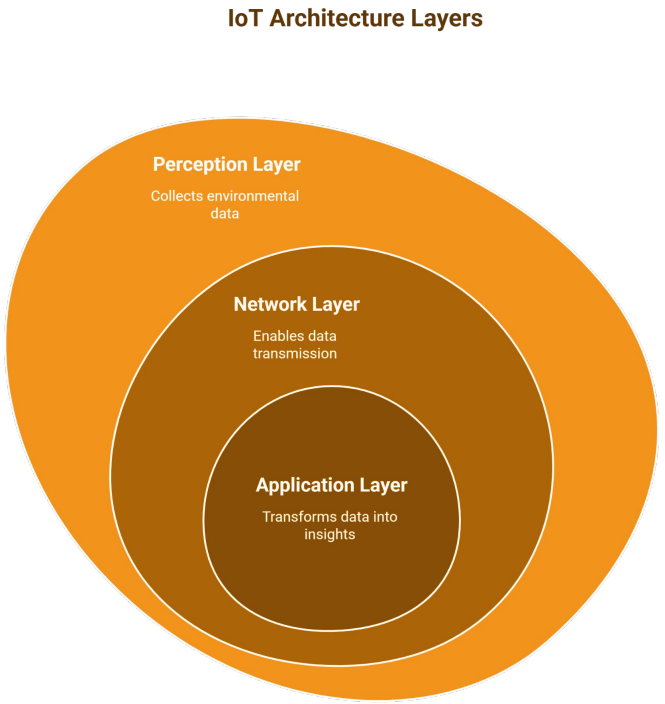
IoT and Edge Computing: Data Pipeline Revolution

IoT Architecture Layers

- Perception Layer: Sensors/actuators collecting environmental data.
- Network Layer: Connectivity protocols (LoRaWAN vs. 5G latency/power tradeoffs).
- Application Layer: Analytics platforms transforming data into insights.

Edge Computing Synergy:

- Latency-Critical Applications: Autonomous vehicle collision avoidance
- Bandwidth Optimization: Manufacturing predictive maintenance at source
- Privacy Compliance: Healthcare data processing under HIPAA



IoT and Edge Computing: Data Pipeline Revolution

IoT-Edge Synergy Framework

Perception Layer Optimization

- **Sensor Fusion:** Kalman filters combine accelerometer/GPS data, reducing false alerts by 30%.

Edge Filtering

- **Temporal Compression:** Transmit hourly averages vs. raw second-level data
- **Spatial Downsampling:** 4K→1080p video processing at edge nodes

HIPAA-Compliant Healthcare Implementation

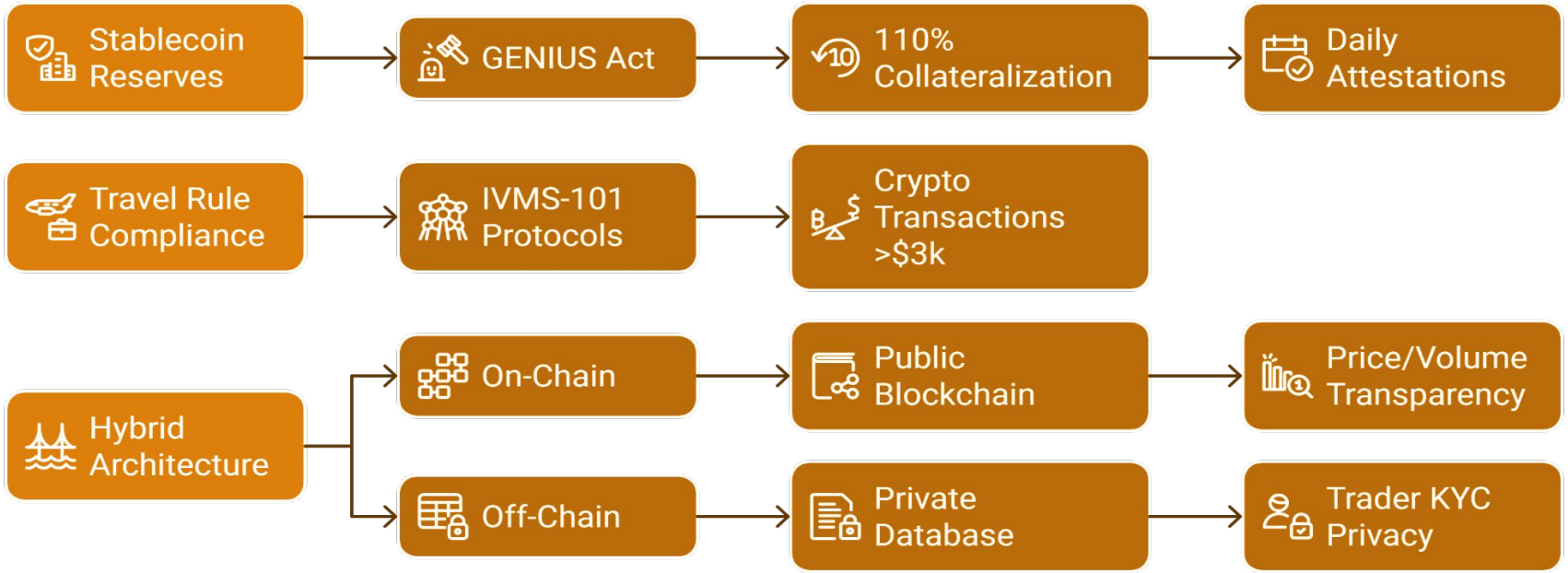
- **On-Device Encryption:** ARM TrustZone isolates patient vitals processing.
- **Federated Learning:** Train ML models across edge devices without transferring PHI.

Firmware Management

- **A/B Partitioning:** Maintain dual firmware copies for rollback during failed OTA updates.
- **Delta Updates:** 70% smaller payloads via binary diffs reduce cellular data costs

Regulatory Preparedness

Digital Asset Governance and Hybrid Architecture



FOCUS

04



PRODUCT:
Mastering Product Thinking

Product Design Frameworks

Product Design and Improvement questions are among the **most frequently asked** in Product Management interviews. These questions assess not just your creativity, but also your **user empathy**, **business thinking**, and **prioritization skills**.

What Interviewers Are Looking For:

- Can you **understand and empathize** with the user?
- Can you **solve problems** while keeping **business goals** in mind?
- Can you **prioritize** features logically?
- Are you able to **think creatively**?

These questions may be related to **digital products**, **physical products**, or a **mix of both**.

Common Mistakes to Avoid

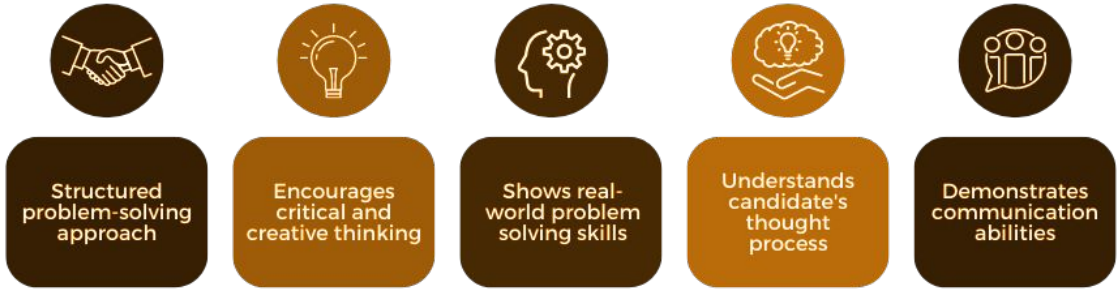
1. **Jumping to Features Too Quickly:** Don't rush to list features the moment you hear the question.
2. **Unstructured Answers:** Giving random or scattered answers makes it hard for the interviewer to follow.
3. **Making Assumptions Without Clarifying:** Don't assume things rather ask questions.

Instead:

- First understand **user problems and pain points**.
- **Pause** at key stages to gather your thoughts.
- Follow a **clear and logical framework**.

The CIRCLE Framework: Your Go-To Structure

A structured approach to product design and improvement problems.



Step	Name	What You Do
C	**Comprehend the Situation**	Ask **clarifying questions** using the **5W1H** method: What, Why, Where, When, Who, How.
I	**Identify the Customer**	List possible **user personas**. Describe their **behaviors**, **demographics**, and **needs/goals**.

The CIRCLE Framework: Your Go-To Structure

A structured approach to **product design and improvement** problems.

R	**Report the Customer's Needs**	Turn customer insights into **user stories**: As a <role>, I want <goal> so that <benefit>.
C	**Cut Through Prioritization**	Prioritize based on **Revenue**, **Customer Satisfaction**, and **Ease of Implementation**.
L	**List Solutions**	Brainstorm possible solutions. Use the **Reversal Method** to spark creativity.
E	**Evaluate Trade-offs**	Compare solutions using **criteria** and a **pros/cons** table. Be thoughtful and objective.
S	**Summarize Your Recommendations**	Wrap it up with a **summary**: your recommendation, its benefits, and why you chose it.

The HEART Framework : Measuring User Experience

The **HEART framework**, developed by Google's UX Research team, helps product managers identify and track **user-centric success metrics**. It is especially useful for assessing **user experience (UX)** during product improvement or feature launches.



Each category follows this three-part process:

1. **Goal** – The general objective.
2. **Signal** – Behavioral indicators that suggest progress toward the goal.
3. **Metric** – Measurable data used to evaluate success.

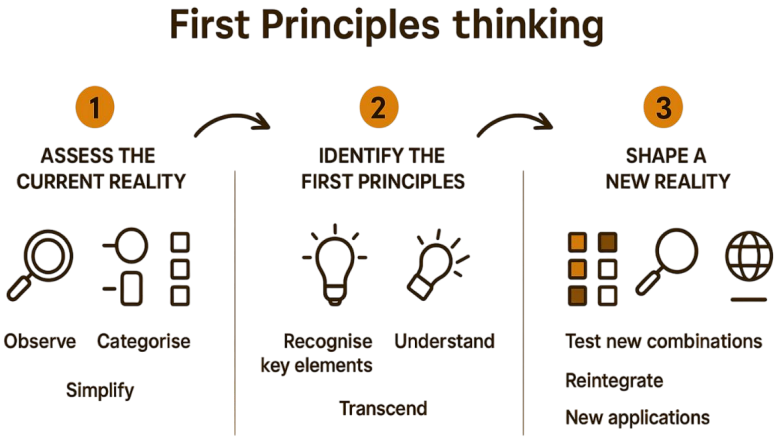
The HEART Framework : Measuring User Experience

Category	Goal	Signals	Metrics
Happiness	Make WhatsApp more delightful and user-friendly	- More message reacts - More positive reviews	Number of 5-star ratings - **NPS** (Net Promoter Score)
Engagement	Encourage users to spend more time and return frequently	- Users spend longer per session - Return more often daily	Avg. session duration - Sessions per user per day
Adoption	Attract new users to WhatsApp	- Increase in new sign-ups - More installs from referrals or marketing	Daily Active Users (DAU) - App install rate
Retention	Retain users and reduce churn to competitors	- Existing users stay longer - Uninstalls decrease	DAU, WAU, MAU trends - Churn rate
Task Success	Ensure new features are easy to use	- Feature usage increases - Error or drop-off rates fall	Feature usage rate - Error Rate % - Drop-off Rate

First Principles Thinking in Product Management

What is First Principles Thinking?

First Principles Thinking is the practice of breaking down complex problems into their most basic, fundamental truths and reasoning up from there rather than relying solely on analogy or "how things have always been done."



In Product Management, this mode of thinking allows PMs to:

- Communicate product rationale clearly.
- Enable fast, aligned decision-making across teams.
- Empower others to operate with autonomy and intent.

As companies scale, Product Managers must act as **decision architects** — shaping direction and unlocking execution without being hands-on in engineering or design.

First Principles Thinking in Product Management

This requires **clarity** in two core responsibilities:

Principle	What It Means	Analogy
A. Maximize Impact to the Mission	Develop a product strategy that delivers the greatest value to the company's core mission, using available inputs.	Think like a **navigator** — pointing to the best route.
B. Accomplish Everything Through Others	Enable the team by aligning goals, removing blockers, and prioritizing work — even though PMs don't directly build.	Think like a **coach** — enabling high performance.

First Principles Thinking in Product Management

A solid product strategy emerges from three key **inputs**:

1. What the Goal Is?

Everything must begin with the **mission**. Without a clear goal, effort is wasted.

Bad PMs quote mission statements.
Great PMs deeply **interrogate the mission**:

- What assumptions about users informed this mission?
- What constraints (ethical, design, legal) frame this mission?
- What future world is this mission building toward?

2. What People, Money, and Time Constraints Exist?

Like a rocket, your product's journey is limited by:

Constraint	What to Evaluate
People	Number, skills, experience, team structure
Money	Hiring, tools, infrastructure, marketing budget
Time	Deadlines, urgency, competitive pressure

First Principles Thinking in Product Management

A solid product strategy emerges from three key **inputs**:

3. What the Environment Is Signaling?

This includes both **customer** and **market** signals:

Think of customer signals as “GPS pings” guiding your path.
Market signals are “asteroid alerts” that require course corrections.

Signal Type	**Examples**
Customer Signals	Usage data, churn, NPS, complaints, feedback
Market Signals	Competitor launches, new tech trends, policy changes

Root Cause Analysis (RCA) for Product Issues

Root Cause Analysis is the structured process of identifying **why** a problem occurred, so corrective action can be taken.

RCA Approach — 5 Steps

Step 1: Scope the Problem

Scope Dimension	Sample Questions
Time	When did the issue start? Trend over weeks/months?
Geography	Is it global, or limited to specific regions or pin codes?
Platform	Is it across web/mobile/app versions?
Feature Specificity	Is it affecting a specific feature, category, or service?
Recent Changes	Have any UI, backend, or policy changes been made recently?

Root Cause Analysis (RCA) for Product Issues

Step 2: Analyse External Factors

Category	Examples
Competition	New rival offers, better delivery, aggressive pricing
Market	Shift in preferences (e.g., post-COVID offline surge), seasonal changes
Stakeholders	Supplier issues, delivery partner changes
Regulatory	Government restrictions, tax policy
Social/PR	Bad press, viral complaints, boycott trends on social media

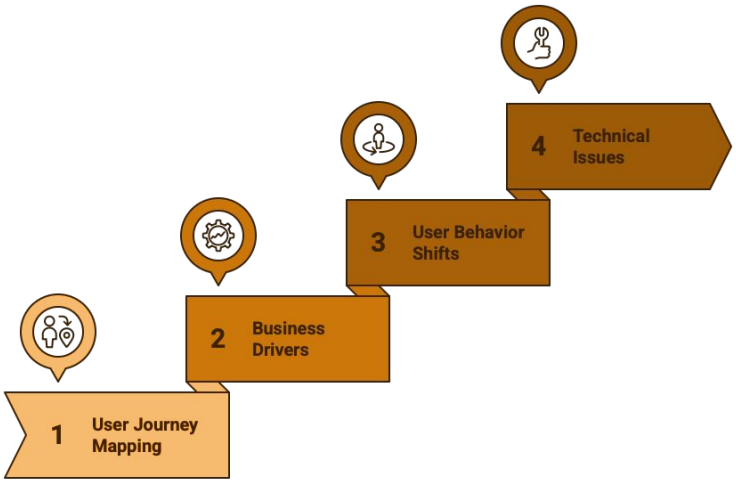
Problem Solving Exercises

Root Cause Analysis (RCA) for Product Issues

Step 3: Analyse Internal Factors

A. User Journey Mapping (Linear Path)

1. **Entry** – How are users entering? (Ads, search, emails)
2. **Discovery** – Are changes in recommendation/search algorithms impacting flow?
3. **Product Detail** – Have images/descriptions/UI been altered?
4. **Cart** – Has cart UX changed? Is price transparency reduced?
5. **Checkout** – Address input, shipping details, coupon application
6. **Payment** – Removed/buggy payment modes? Long loading times?
7. **Order Confirmation & Tracking** – Slow confirmations, no delivery updates?



Root Cause Analysis (RCA) for Product Issues

Step 3: Analyse Internal Factors

B. Business Drivers

- Increased prices or delivery fees?
- Changes in return/refund policy?
- Unfavorable shipping options?

C. User Behavior Shifts

- Shift in user demographics (new markets, devices)?
- Change in promo behavior (e.g., lower discounts)?
- Impact of recent email campaigns or retargeting?

D. Technical Issues

- Increased downtime, lag, or latency?
- Increased error rates in funnel steps?
- Bugs in checkout or payment flows?

Root Cause Analysis (RCA) for Product Issues

Step 4: Hypothesis Generation

From your scoped data, **generate specific hypotheses**. E.g.:

- “Cancellations rose in Tier-2 cities where cash-on-delivery was removed.”
- “Cancellations spiked after competitor X launched 24-hour delivery on the same product line.”
- “Drop-off rate increased 20% on Android 12 due to payment screen not loading.”

Prioritize hypotheses by:

Criterion	Examples
Magnitude	How many users are impacted?
Recency	Did it align with any recent changes/releases?
Severity	Revenue impact, user churn
Reversibility	Can the issue be quickly reverted or patched?

Root Cause Analysis (RCA) for Product Issues

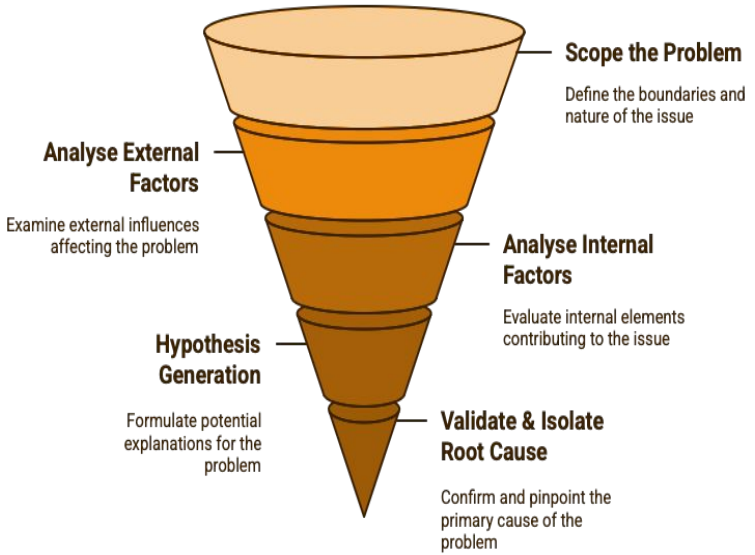
Step 5: Validate & Isolate Root Cause

Use **data deep dives, A/B tests, heatmaps, session recordings, user interviews**, and funnel drop-off analytics to test your hypotheses.

If needed, segment data further by:

- OS Version
- Device Type
- User cohort (new vs returning)
- Location (pincode/city)

Once validated, **communicate the root cause** clearly and propose the resolution path.



What is Mobile App Architecture?

Mobile app architecture is the **foundation for designing and building** mobile applications. It includes decisions around technologies, platforms, and practices that ensure:

- Smooth and fast performance
- Scalability to support growing users and features
- Strong security
- Easy maintenance and future development

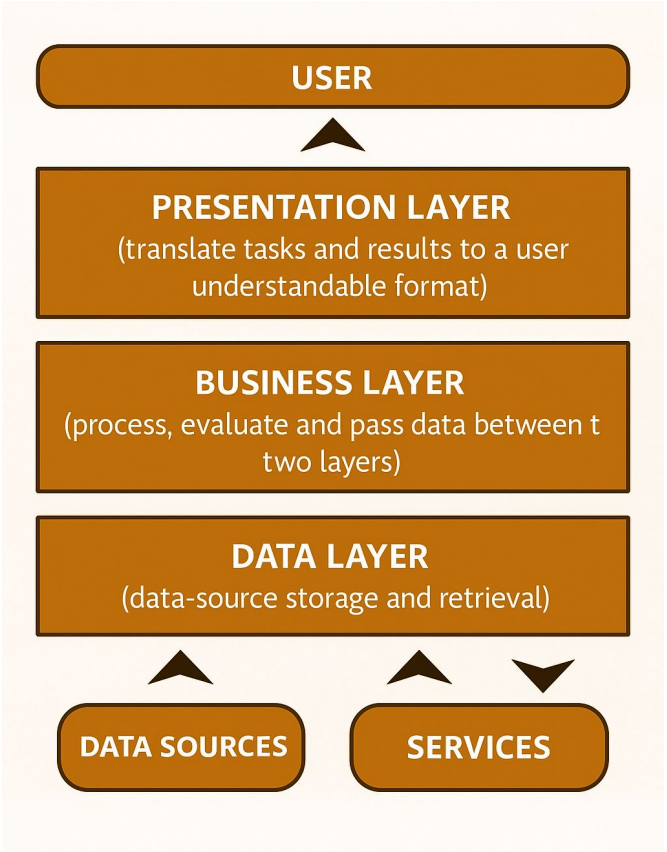
Core Principles:

- **Separation of concerns**: Divide responsibilities among UI, logic, and data handling layers.
- **Performance optimization**: Use resources (CPU, memory, battery) efficiently.
- **Scalability**: Allow the app to grow without rewriting core systems.
- **Security**: Protect user data and ensure secure communication.

Architecture Of Popular Apps

Latest Trends in Mobile App Architecture (2025)

- **Modular Architecture**: Break the app into independent modules to simplify development, testing, and reuse.
- **Cross-Platform Development**: Use tools like Flutter, React Native, or Xamarin to develop once and deploy to both Android and iOS.
- **Cloud-Native Architecture**: Leverage services like Firebase or AWS for backend, reducing infrastructure management.
- **Microservices**: Split backend into independent services that communicate via APIs.
- **Edge Computing**: Shift data processing closer to users for faster performance and lower latency.



Architecture for Native Mobile Platforms

Android (MVVM/MVP)

- **UI Layer:** Built with Activities, Fragments, or Jetpack Compose.
- **ViewModel:** Manages UI data and logic, survives configuration changes.
- **Repository:** Acts as the central source of data (from API, local DB, etc.).
- **Room:** Provides local storage with SQLite.
- **LiveData/Data Binding:** Enables reactive UI updates and clean data flow.

iOS (MVVM/VIPER)

- **UI Layer:** Built using SwiftUI (declarative) or UIKit (imperative).
- **ViewModel/Presenter:** Handles logic, UI state transformation.
- **Interactor/Data Manager:** Fetches data from network or local storage.
- **Router:** Manages navigation and screen transitions cleanly.

Dating Apps Like Bumble

Unique User Interface and Experience

Create a visually inviting and intuitive interface that makes swiping fun and memorable.

- Design vibrant UI with animations
- Use gamified visuals and playful microinteractions
- Ensure intuitive profile exploration

Marketing with a Dash of Fun

Attract users with personality-driven campaigns.

- Create quirky taglines and humorous posts
- Launch with influencer partnerships
- Use real-world dating analogies and memes

Innovative Matching Algorithms

Match users based on more than just proximity.

- Combine interests and quiz inputs
- Use ML to improve suggestions
- Integrate compatibility quizzes

Focusing on User Control

Empower users to lead the conversation.

- Implement gender-first message rules
- Add fun conversation starters
- Include profile prompts for ice-breaking

Cross-Platform Architecture

a) Flutter (Dart)

- **Widgets:** Everything in Flutter is a widget (UI elements and layout).
- **BLoC Pattern:** Separates UI from business logic using reactive streams.
- **Repository:** Interfaces with APIs, Firebase, or local storage.

b) React Native (JavaScript)

- **React Components:** Declarative UI built with reusable components.
- **Redux:** Manages application state centrally.
- **Native Bridge:** Allows communication with platform-specific modules.

Security Considerations

Mobile apps must implement strong security practices due to the sensitive data they handle.

Best Practices:

- **End-to-End Encryption:** Use TLS for secure client-server communication.
- **Authentication:** OAuth 2.0, biometric login, and multi-factor authentication.
- **Data Encryption:** Encrypt data both at rest and in transit.
- **Secure APIs:** Validate inputs, use secure tokens, and protect against attacks.

Backend and Cloud Integration

Cloud platforms handle essential backend functions such as data storage, user management, analytics, and notifications. These simplify development and improve scalability.

Firebase

- Authentication
- Firestore (NoSQL database)
- Push notifications

AWS Amplify

- GraphQL APIs
- File storage
- Real-time updates and analytics

Azure Mobile Apps

- Cloud storage and sync
- Notification services

Architecture Of Popular Apps

Dating Apps Like Bumble

Continuous Evolution of Features

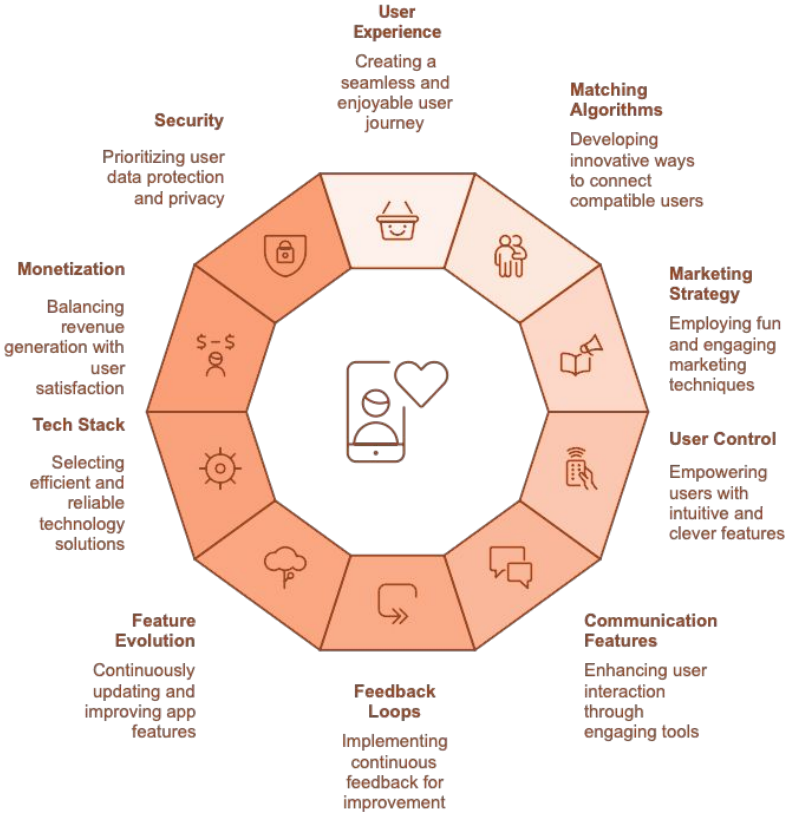
Keep the app dynamic and trend-aware.

- Regular rollouts: virtual dating, video chats
- Trend alignment (astrology, audio bios)
- A/B testing for feature validation

Efficient Tech Stack Selection

Build a scalable, secure, and responsive app.

- **Frontend:** Swift, Kotlin, Flutter
- **Backend:** Node.js, Firebase, MongoDB
- **ML Matching:** Python (TensorFlow)
- Modular architecture for flexibility



💡 Timeline: MVP → Boost → Video Date → Audio Rooms

Architecture Of Popular Apps

Dating Apps Like Bumble

Engaging Communication Features

Make messaging fun, expressive, and vibrant.

- Emojis, GIFs, voice notes, and stickers
- Enable reactions to messages/profiles
- Disappearing messages & timed replies

Feedback Loops & Participation

Let users shape the future of your app.

- In-app surveys and feedback prompts
- Gamify feedback ("Rate the vibes!")
- Prioritize roadmap with user insights

Monetization & Experience

Make money without ruining the vibe.

- Freemium model: SuperSwipe, Spotlight
- Paid profile boosts & unlimited swipes
- Keep ads minimal and non-intrusive

Security as Top Priority

Build trust through protection.

- End-to-end encryption (TLS)
- Secure OAuth 2.0 and MFA
- Content moderation & ID verification

Architecture Of Popular Apps

Dating Apps Like Bumble

User Interaction & Frontend

React Native (mobile), HTML5 + JS (web)

- PMs define UX goals and ensure design aligns with user expectations.
- Continuous iteration based on usability testing and behavioral analytics.

Video Uploading & Encoding

Provide a smooth, intuitive content upload experience.

- Upload, trim, filter, and post with minimal lag.
- PMs prioritize in-app editing tools and compression standards.

CDN Optimization

Ensure fast delivery of rich media globally.

- CDNs reduce load times by distributing content closer to users.
- PMs monitor loading speeds and CDN performance via analytics.

Backend Infrastructure

Python + Django framework

- Microservices manage feeds, likes, stories, and notifications.
- PMs collaborate on uptime targets and scaling strategies.

Architecture Of Popular Apps

Instagram Platform Architecture

Monetization Strategies

Revenue & Value Enhancement

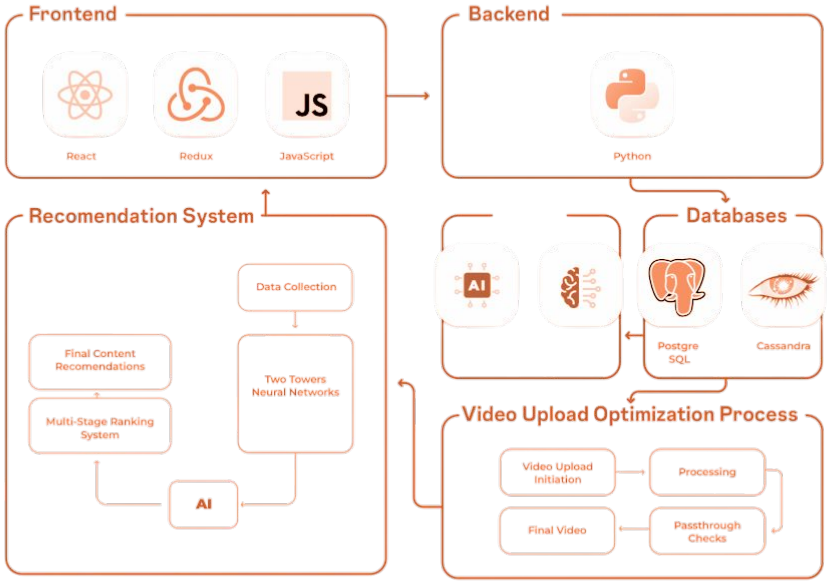
- **Sources:** Sponsored posts, in-feed ads, reels, influencer campaigns.
- **Execution:** Ad placement logic, brand safety, and targeting accuracy.
- **Optimization:** A/B testing to balance UX with ad exposure.

Database Management

Scalable Data Retrieval

- **PostgreSQL:** Relational data (posts, comments, tags).
- **Redis:** Sessions, likes, and story expiration logic.
- **Focus:** Fast querying and high-volume scalability.

Instagram Tech Stack



Architecture Of Popular Apps

Social Media Platforms like Instagram

Compliance & Privacy

Protect user data and adhere to global regulations.

- GDPR, CCPA, and platform privacy policies govern data use
- PMs align features (data export, delete account) with legal
- Collaborate with legal teams on privacy-by-design

Suggestion: Privacy toggle screen or GDPR flow

User Feedback

Create a feedback loop for continuous innovation.

- Use app store reviews, in-app surveys, and listening tools
- Prioritize features that address friction or enhance stickiness
- Run beta tests and pilot features for select cohorts

Suggestion: Feedback loop graphic (feedback dev release)

Analytics & Engagement

Measure performance and optimize product decisions.

- KPIs: DAUs/MAUs, retention, session time, story interactions
- Track funnel metrics and optimize underperforming features
- Integrate tools like Mixpanel, Amplitude, or Meta Analytics

Suggestion: Dashboard with engagement metrics and trends

YouTube Streaming Service Architecture

User Interaction & Frontend

Deliver a responsive interface for global users.

HTML5, CSS, JavaScript

- Consistent cross-device responsive design
- PMs define UI/UX via usability testing
- Dark mode & intuitive navigation

Video Uploading & Encoding

Smooth experience for content creators.

- Streamlined multiple resolution encoding
- Adaptive streaming support for various devices
- Efficiency and upload reliability focus

Backend Infrastructure

Scalability for billions of daily actions.

Python, Java, Go

- Core backend logic & API endpoints
- Modular architecture for recommendations
- Alignment with user load targets

Streaming Protocols

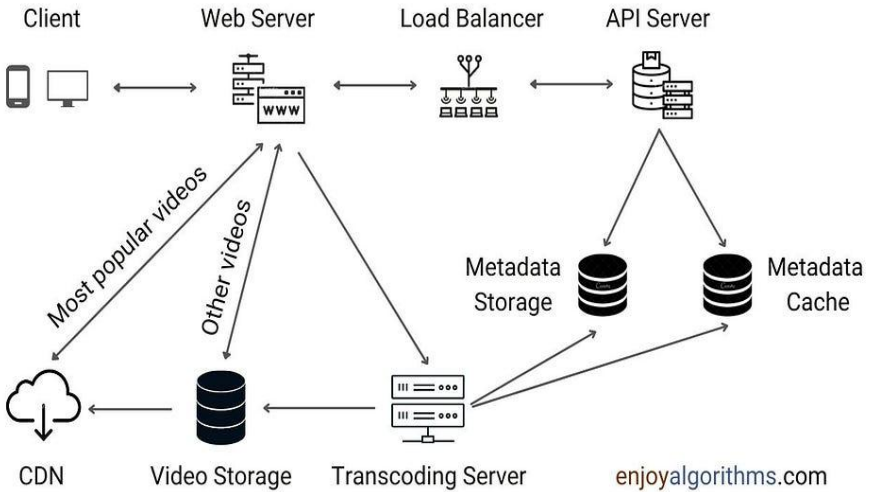
Optimize playback across networks.

HLS, MPEG-DASH

- Adaptive bitrate for optimal quality
- Data-driven regional network decisions
- Reduced buffering and resolution lag

Architecture Of Popular Apps

YouTube Streaming Service Architecture



💡 **Visual Suggestions:** World map with CDN nodes and personalization engine flows (User Input → Algorithm → Feed).

Content Delivery Network (CDN)

Reduce latency and increase video load speed through edge caching.

- Cache popular videos closer to users for improved reliability
- Monitor performance metrics and prioritize geo-optimizations
- Strategy aligns with usage trends and regional demands

Recommendation Algorithms & Data Analytics

Drive engagement through personalized content delivery.

- Uses user behavior, history, and engagement signals
- Key metrics: click-through rate, watch time, retention
- Collaboration ensures algorithm-strategy alignment

Architecture Of Popular Apps

YouTube Streaming Service Architecture

Tools & Tech

Streamlined project execution and team collaboration.

Jira: Backlog management and sprint planning

Slack: Real-time team communication

Figma: Design review and prototyping

Analytics: User behavior analysis and A/B testing

💡 **Visual Suggestion:** Toolstack overview across product lifecycle

Monetization

Maximize revenue through ads and subscriptions.

- Revenue: in-stream ads, banner ads, Premium, channel memberships
- PMs segment users to optimize formats and adoption
- Balance goals with user satisfaction

💡 **Visual Suggestion:** Flowchart of ad-supported vs. premium journeys

Compliance

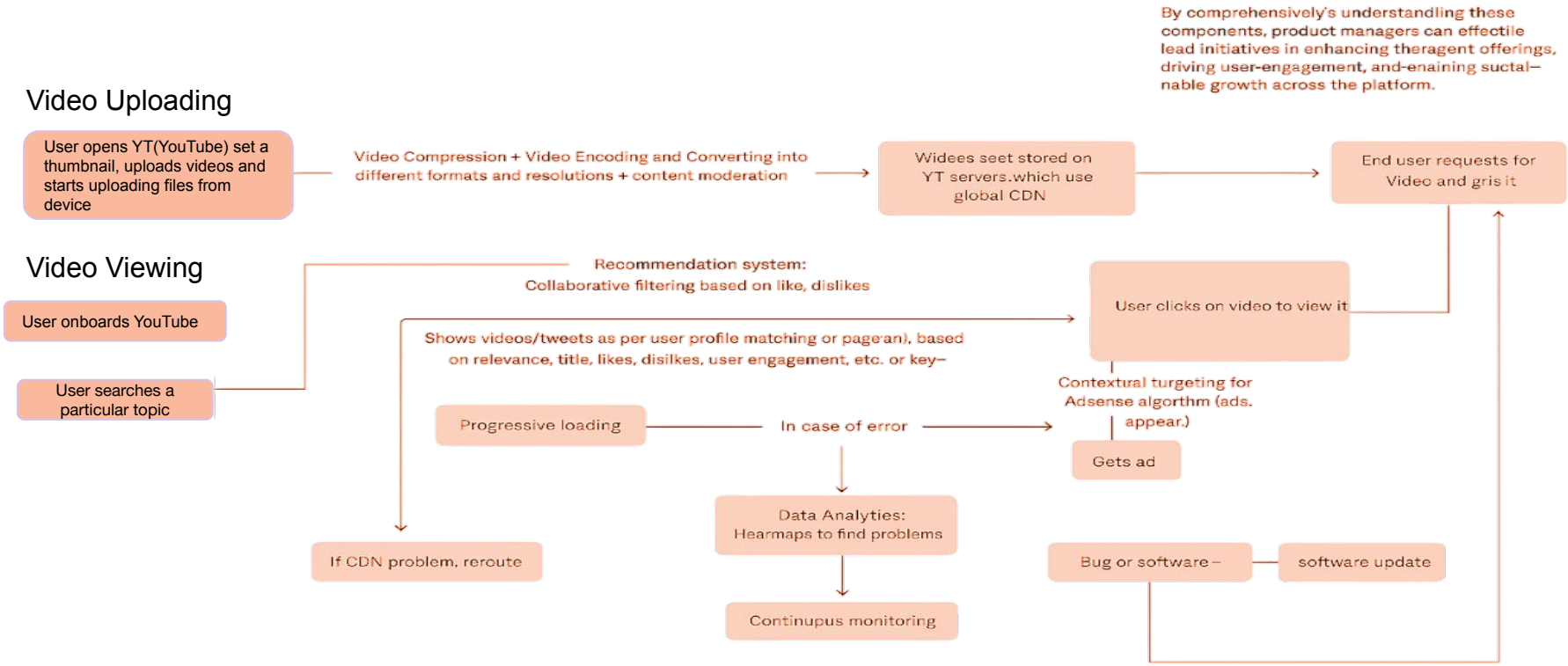
Ensure legal and ethical global operations.

- Areas: copyright laws, moderation, age restrictions
- PMs coordinate with legal and policy teams
- Focus on transparency and content flagging

💡 **Visual Suggestion:** Lifecycle diagram from upload to legal review

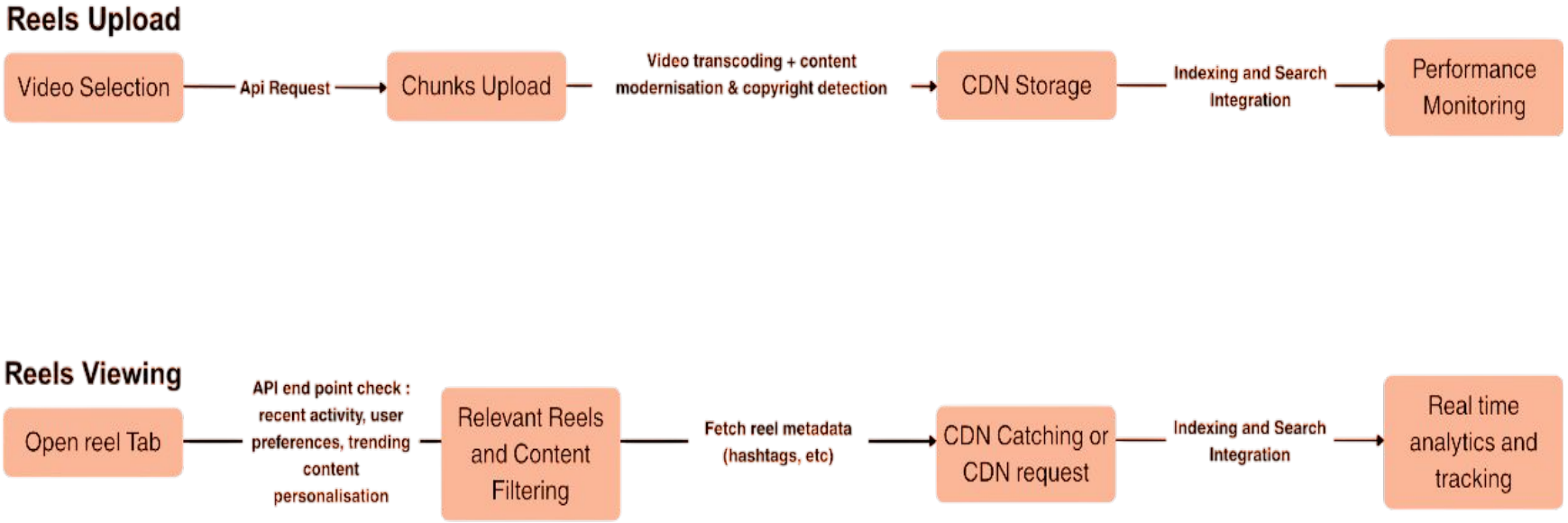
Architecture Of Popular Apps

Architecture of Reels on Instagram



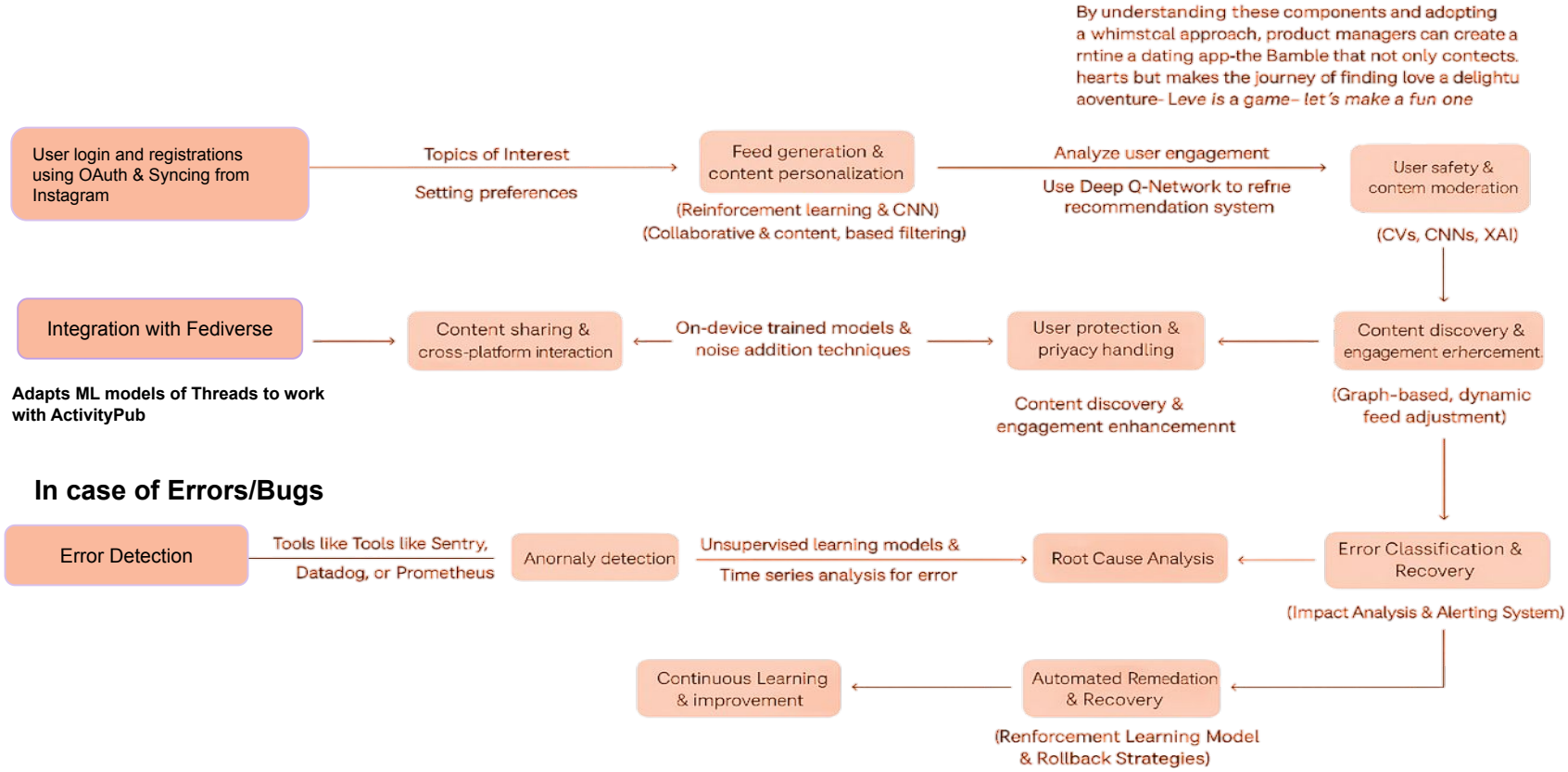
Architecture Of Popular Apps

Architecture of Reels on Instagram



Architecture Of Popular Apps

Architecture of Threads App



Product Teardown: How To

1. What Is a Product Teardown?

A product teardown is a systematic dissection of a product to analyze its features, design decisions, user experience, and overall strategy. The goal isn't just to understand what a product does, but how it does it and why it was built that way.

Unlike a casual product review, a teardown is analytical and structured. Think of it like a doctor examining a patient: you observe, diagnose, and prescribe, with explicit reasoning at every step.

You take an existing product and look at it through a PM's eyes to answer:

- Who is this product built for?
- What problem does it solve?
- What is working well, and why?
- What is broken or missing?
- What would you fix, and how would you know it worked?

Key Idea: *"You are not building something new. You are reverse-engineering the decisions behind an existing product to extract lessons you can act on.*
."

2. Why Conduct Teardowns?

If you're trying to break into PM, a published teardown beats mass-applying. Hiring managers see hundreds of similar-looking resumes and can't tell from any of them whether you actually think like a PM. A teardown fixes that by doing three things at once:

Shows critical thinking

PMs need strong analytical thinking — a teardown proves it publicly.

Demonstrates domain knowledge

It shows you understand the product deeply, not just the job description.

Creates proof of work

It's a portfolio piece anyone can read, share, and find.

The promise: *"One good teardown, published and tagged well, will open more doors than 50 job applications."*

Product Teardown: How To

3. The Teardown Process (8 Steps)

Step 1 — Set context and goals

Understand the problem statement and the teardown's purpose.

Common goals:

- Revenue
- Engagement
- Retention
- User growth

Step 2 — Define the scope

Narrow down to a specific level:

- Feature-level

Core features like search.

- Workflow-level

End-to-end journeys like checkout.

- Design-level

UI/UX, accessibility, hierarchy.

Step 3 — Identify the users

Break broad audiences into specific personas (e.g., Airbnb: hosts, guests).

Capture for each:

- Frustrations
- Motivations
- Context
- Use cases

Product Teardown: How To

Step 4 — Find the gaps and problems

This is the most important step. Find the real problems in users' lives that the product needs to solve. Build a user journey to surface where the product's offerings and user needs diverge.

Jobs To Be Done (JTBD)

Frame what users are really hiring the product to accomplish.

The 5 Whys

Keep asking "Why?" to get past symptoms to root causes.

Core Analysis Layers

Functionality

What does the product do? What are its standout features?

Usability

How easy is it to achieve goals? Where are the friction points?

Technical

What powers the product? How scalable is its architecture?

Business Alignment

How do features support business goals like revenue or acquisition?

Product Teardown: How To

Step 5 — Prioritize

Pick the problems that matter most

Align with your goals from Step 1. Prioritization is the skill being tested — listing ten problems with no ranking is a red flag.

Narrow to the top 2 or 3.

Step 6 — Solutions

Get creative. Generate multiple solutions per problem, research others, and weigh trade-offs.

Impact vs. Effort Framework:

- Impact must align with Step 1 goals.
- If effort is high, break into an experiment or MVP.

Step 7 — Success Metrics

Define how you'd know it worked using these four types:

North Star
Main goal movement.

Awareness
Discovery problem check.

Adoption
Usability problem check.

Guardrail
What must NOT get worse.

Step 8 — Summarize

Close by stating the solution, how you arrived at it, and why it's better. The ability to compress everything into one clear, concise summary is itself a signal of PM maturity. Aim for a single slide or one screen.

App Teardowns

Bumble – A Strategic Challenger in the Dating App Market

MARKET CONTEXT

Founded by a former Tinder executive, Bumble takes a familiar swiping mechanism and differentiates with a women-first messaging model.

It is the second most downloaded and grossing dating app in the U.S., closely rivaling Tinder.

Its ASO and monetization tactics make it a standout case in digital product growth.

TEARDOWN GOAL

Understand how a challenger brand drives growth and monetization against an entrenched market leader.

Growth
Sustain download momentum as a #2 player

Monetization
Convert free users to paying at scale

Differentiation
Defend the women-first brand position

App Teardowns

Bumble – A Strategic Challenger in the Dating App Market

FEATURE-LEVEL
Feature & Model
The women-first messaging mechanic and core swiping experience

BUSINESS-LEVEL
Monetization
Freemium structure, paid features, and revenue-per-download

DESIGN-LEVEL
ASO & Visuals
App Store screenshots, video strategy, and store positioning

Primary Audience

Young, slightly male-leaning

The women-first model directly shapes who uses Bumble and how. Women message first, which reframes the experience for both sides of the marketplace.

Women
Hold the initiating role, giving more control over who they engage with — the brand's core promise.

Men
Wait to be messaged, a different dynamic from competitors — a key differentiator but also a retention challenge.

App Teardowns

Bumble – A Strategic Challenger in the Dating App Market

THE EVIDENCE

462.6K	$15.1M	77.4%	3x
Monthly downloads (U.S.)	Estimated monthly revenue	Positive review rate	Global downloads since 2017

THE GAPS

No App Store video

A narrative-style video could lift downloads, but the store listing currently has none.

Shallow screenshot storytelling

Uses 7 of 10 screenshots; they explain obvious features but lack depth and richer narrative.

Domestic saturation

Strong U.S. presence but higher domestic saturation than Tinder — growth headroom is international.

App Teardowns

Bumble – A Strategic Challenger in the Dating App Market

Sorting the gaps by impact on the growth goal — the video gap is the highest-leverage, lowest-effort fix.

1	**Add an App Store video** Direct lever on conversion-to-download; competitors use it, Bumble doesn't.	High impact, low effort
2	**Richer screenshot storytelling** Three unused screenshot slots are free real estate for narrative selling.	Medium impact, low effort
3	**International expansion push** Biggest long-term upside, but a heavier, slower strategic investment.	High impact, high effort

Narrative App Store video

Given Bumble's story-driven appeal, a short narrative-style video could enhance downloads. Apple's strict video guidelines may deter teams — but testing a compliant version is advisable.

Trade-off: production cost & review risk vs. conversion upside

Fill the 3 empty screenshots

Use the remaining slots for richer storytelling — emphasize safety and privacy as a differentiator, and humanize the experience instead of listing features.

Trade-off: low cost, mainly creative effort

App Teardowns

Bumble – A Strategic Challenger in the Dating App Market

How we'd know the video & screenshot changes actually worked.

NORTH STAR

Conversion rate

Store-visit → download rate. The primary number the changes should move.

AWARENESS

Listing impressions

Are users reaching and viewing the enriched listing in the first place?

ADOPTION

Creative views

Do visitors actually engage with the new video and screenshots before deciding?

GUARDRAIL

Review rating

Richer claims must not raise expectations so high that ratings drop.

Conversion beats raw downloads

High conversion rates matter more than top-of-funnel volume — optimize the moment of decision.

Iterate with data — carefully

Bumble experiments aggressively, but overdoing it can hurt store visibility. Test, don't thrash.

Differentiation is vital

The "women message first" mechanic shapes the brand and the product. Protect what makes you distinct.

Visual storytelling sells

Screenshots do most of the heavy lifting; video is an underused asset. Sell the story, not the feature list.

Building Minimum Viable Product (MVP)

1. Brainstorm Features by Benefit

For each **benefit** in your **product value proposition**, brainstorm as many potential **feature ideas** as possible.

- Practice **divergent thinking**—generate ideas without judgment.
- Encourage creativity and “outlandish” ideas.
- Organize the generated ideas by the benefit they aim to deliver.

2. Prioritize Feature Ideas by Customer Value

After brainstorming, **review and prioritize** ideas:

- Score each feature on **expected customer value** (e.g., scale of 1–10).
- Identify the **top 3–5 features** per benefit.

3. Write User Stories

Convert your top feature ideas into **user stories** to ensure the **customer benefit** remains clear.

User Story Template:

> As a **[type of user]**,
> I want to **[do something]**,
> so that I can **[achieve benefit]**.

Ensure your stories follow the **INVEST** criteria:

- **Independent**
- **Negotiable**
- **Valuable**
- **Estimable**
- **Small**
- **Testable**

Building Minimum Viable Product (MVP)

4. Chunk Features into Smaller Pieces

Break each user story into **feature chunks** (smaller components) to reduce scope.

Example:

> “Share photo” → Share via Facebook, Share via email, Share via text

Focus on:

- Building **only the most valuable parts**.
- Delivering **value quickly**.
- Enabling faster feedback and iteration.

5. Estimate Effort with Story Points

Estimate the **development effort** for each feature chunk using **story points** (e.g., 1, 3, 5, 8).

If a feature chunk is too large:

- **Break it down** further until each chunk is **under a manageable threshold**.

Building Minimum Viable Product (MVP)

6. Prioritize Based on ROI (Return on Investment)

Now evaluate features based on **both value and effort** to determine ROI:

ROI = Value / Effort

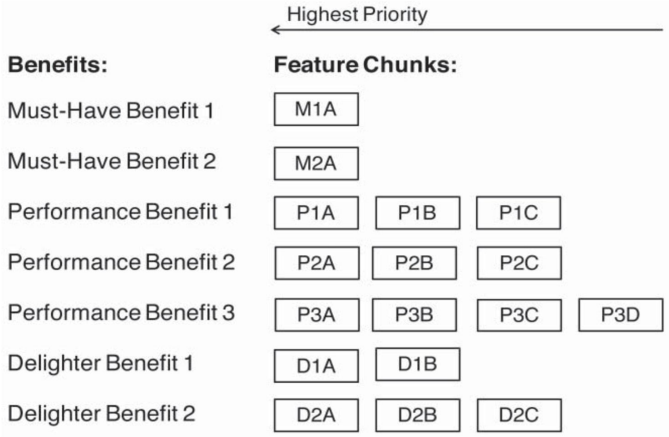
- Prefer **high-value, low-effort** features.
- If two features have equal ROI, **prioritize the smaller one**.
- **Avoid low-ROI** features that require large effort with low customer value.

If exact numbers are hard to estimate, use **High / Medium / Low** for value and effort, and place features in a 3×3 **ROI matrix**:

- Highest priority = High value, Low effort
- Lowest priority = Low value, High effort

7. Select Your MVP Candidate Features

- Use your prioritized list to select the **minimum set of feature chunks** that together:
  - Deliver your **core must-have benefit**.
  - Include a few performance benefits and delighters if they're **high-ROI**.
- Use a grid like this to organize:



Building Minimum Viable Product (MVP)

Understanding MVP Tests

- MVP tests are designed to validate hypotheses underlying your MVP.
- Different forms (e.g., landing pages, wireframes, working prototypes) depending on what you're testing.



Matrix of MVP Tests (two axes: Marketing vs Product, Qualitative vs Quantitative)

1. **Qualitative Marketing Tests**
   - Show landing pages, ads, or emails to customers.
   - Solicit feedback on messaging and value propositions (e.g., 5-second tests).
2. **Quantitative Marketing Tests**
   - **Landing page / smoke tests**: measure conversion rates.
   - Test which benefits resonate and validate demand.

Building Minimum Viable Product (MVP)

3. Qualitative Product Tests

- **Wireframes, mockups, interactive prototypes**.
- Involve direct user feedback, explore **why** users react the way they do.
- **Concierge MVP**: manually provide service to understand user needs (e.g., Airbnb's photo test).
- **Wizard of Oz MVP**: simulate functionality behind the scenes without full automation.

4. Quantitative Product Tests

- Conducted at scale on live product.
- Use **fake door / 404 tests** to measure demand for unbuilt features.
- **A/B testing and analytics tools** to understand behavior trends.

Tooling & Prototyping

- Prototyping tools: HTML/CSS/JS, Axure, Rails, or native mobile code.
- Prioritize speed and interactivity—don't overbuild before validation.

Building Minimum Viable Product (MVP)

Best Practices

- Begin with **qualitative testing** to understand customer reactions.
- Move to **quantitative testing** for broader validation.
- Avoid using MVP as a justification for low quality.
- Remember: the goal is **learning** fast with minimal waste.

	Qualitative Tests	Quantitative Tests
Marketing Tests	Marketing materials	Landing page/Smoke test Explainer video Ad campaign Marketing A/B tests Crowdfunding
Product Tests	Wireframes Mockups Interactive prototype Wizard of Oz & Concierge Live product	Fake door/404 page Product analytics & A/B tests

FOCUS

01



Building Your Profile and Acing Interviews:

Profile Building

How does one get shortlisted?

The Pre-requisites:

- Ability to pitch to and influence people
- Clear written and verbal communication
- Technical and business/sales acumen
- Eye-for-detail and quick-thinking
- Can assume ‘ownership’ of the product

Additionally, a prior PM intern, or projects showcasing your problem-solving abilities, or even managerial or decision-making experience is a plus.

The Process:



Resume → Slide decks/aptitude test* → Interviews

Rules to consider while creating resume

Rule 1: Shorter is Better

You might have to be more concise:ask yourself what about that role is most important Focus on what is important, and leave out the rest.

Rule 2: Bullets, Not Blobs

Read through your resume. Anything that’s three lines of text or more should be condensed.

Rule 3: Accomplishments, Not Responsibilities

People don’t care what you were told to do; they care what you did.

Responsibility Oriented: Lead the development of recommendation engine features for an e-commerce platform and coordinate efforts across data scientists, backend engineers, and product teams.
Accomplishment Oriented: Launched the Dynamic Deals Engine across five marketplaces, boosting conversions by 15% and adding $8M in revenue.

Building a Non - Tech CV

Where to Build Your CV

- Recommended tools: **Overleaf** or **Figma**
- Maintain consistency in **font style, weight, and spacing**
- Stick to a **one-page format** – this improves readability and demonstrates prioritization and conciseness

Crafting Strong Bullet Points

- Start with **action verbs** (e.g., spearheaded, analyzed) — refer to resources like Harvard, Glassdoor, or use ChatGPT for refining
- **Quantify** impact using metrics, percentages, or numbers
- Ensure **grammatical accuracy** and coherence
- Follow the structure:
  [Action Verb] + [What You Did] + [Result]
  (Use this format across Experience, Projects, PORs, Achievements & Extracurriculars)

Building a Non - Tech CV

Core CV Sections

1. Education

- Capture academic performance — **CPI** matters significantly
- **Minors** are optional but can strengthen relevance to your target role
- **10th/12th Grades** aren't critical unless exceptionally low

2. Experience

- Highlights your **professional exposure**
- Includes **on-campus/off-campus internships** (only pursue those with long-term value)
- Emphasize **your contributions and resulting impact**

3. Projects

- Showcase **practical learning and application**
    - **Self-Initiated**: Teardowns, case studies, intern projects, BTPs
    - **Club Projects**: Add credibility and product-building experience
    - **Competitions & Hackathons**: Double up as achievements and quality projects

Building a Non - Tech CV

4. Positions of Responsibility (PORs)

- Reflect **leadership and initiative**
  - Choose **relevant PORs** aligned with your desired roles
  - Prioritize **impact**, not just the title

5. Technical Skills

- Optional, but can **diversify your profile**
  - Especially helpful for **analyst roles**
  - Common tools: **Python, SQL, Tableau, Excel**

6. Achievements

- Demonstrates **excellence and distinction**
  - Include: Competition wins/finalists, awards, scholarships, JEE rank, branch change, etc.
  - Always **quantify** (e.g., top 1%, AIR 600, etc.)

7. Extracurriculars

- Reflects **well-roundedness beyond academics**
  - Include sports, cultural, or volunteering pursuits
  - Aim to highlight any **achievement or leadership** within these activities

Optional/Additional Sections

- **Entrepreneurial Experience**: Reflects strategic thinking and ownership
- **Research Publications**: Not essential, but good for academic signaling or compensating for low CPI
- **Key Courses Taken**: Add only if highly relevant to the role

Cover Letters

Cover letters are a mixed bag. Some companies value them highly, others don't care, and many request them without much emphasis. However, a well-crafted cover letter can be crucial especially if your background isn't an exact match for the role. It allows you to connect your experiences to the job in a way a resume might not.

Elements of a Good PM Cover Letter

- **Short (200–250 words):** Keep it concise. Long letters often go unread and suggest a lack of clarity. Brevity shows respect for the reader's time and reflects strong communication.
- **Shows Passion:** PMs should be passionate about technology and the industry. Share what excites you about the role and company.
- **Demonstrates Skills:** Align your background with key PM competencies, especially those not evident on your resume. Highlight transferable or overlooked skills here.
- **Matches Company Culture:** Tailor the tone of your letter to reflect the company's personality. If it's quirky or bold, let your writing reflect that especially if your resume is more formal.
- **Well Written:** Treat it as a writing sample. Avoid grammar errors, keep the tone crisp and direct, and balance sentence lengths for readability.

- **Consistent Storyline and Flow**
  - Storytelling is the foundation of a good slide deck.
  - Make the presentation a cohesive unit, where it ceases to be individual slides or different ideas.
  - Viewing your presentation as a story that has clear characters, conflict, and resolution will help generate a throughline in your presentation and solidify it as a cohesive unit. Consider how the information moves the story forward on each slide and figure out how one slide leads into another.
  - Try using specific tabs at the bottom or sides of the slide to depict flow.

- **Extensive User and Industry Research**
  - As a good Consultant or Product Manager, the main aim should be to present a comprehensive report on user and sector research. It can be achieved by following the given steps:
  - Identify the problem.
  - Gather the correct type of information using secondary research or surveys.
  - Analyse the information gathered and find out insights.
  - Make a fitting conclusion.

- **Design and Visual Appeal**
- Maintain consistent formatting and margins in the entire deck.
- Keep it light on text!
- Try to use pictures/charts to express your data or ideas. Graphics and charts explain the insights better than words.
- Keep the content directional.
- Use Selective Bolding to stress an important term and ensure to use clear headlines for the readers' convenience.

- **Executive Summary and Appendix**
- The executive summary slides should only be 5% - 10% of the entire presentation length. Make the executive summary focused and simple by using short paragraphs or bullets and subheadings.
- Highlight the key results, including 1 or 2 statistics that drive home the takeaway message.
- Mention the source in the footnotes and include an appendix to show all the research, calculations and basis of insights.

Crafting Impactful PM Projects

What is a PM Project?

A PM project is tangible **proof of your ability** to think and operate like a Product Manager — a practical demonstration that you've engaged in product thinking, execution, or decision-making in a real or simulated environment.

Why Are PM Projects Important?

- **Strengthen Your CV** — Especially valuable when you're early in college or have limited experience.
- **Demonstrate Product Thinking** — Shows you understand user needs, product value, and iterative improvement.
- **Differentiate Yourself** — High-impact projects make you stand out among other candidates.
- **Signal Readiness** — Communicates that you're already aligned with PM responsibilities and mindset.

Crafting Impactful PM Projects

What Makes a Good PM Project?

- **User-Centric Approach**
  Starts with a real user pain point and focuses on solving it effectively.
- **Back to Basics**
  Keeps things simple, focused, and grounded in core product principles.
- **Prioritized and Measurable**
  Every idea is prioritized and backed by metrics or clear outcomes.
- **Business-Aligned**
  The project ties into team goals, organizational vision, or market need.

What Do These Elements Signal to Interviewers?

- You **empathize with users** and prioritize their problems.
- You **understand iteration**, improvement, and tracking metrics.
- You think like a **true product manager** — balancing user value, business goals, and execution.

How to Make Good PM Pitch Decks

Cracking a PM role often starts with cracking the deck here's how you can craft compelling, clear, and impact-driven Product Management decks that get you shortlisted.

Idea Selection – Finding a Problem That Matters

Companies may give you a prompt or ask you to pick your own problem to solve.

- Look at **everyday activities** you do and find what's inefficient or frustrating.
- Think about processes that can be **digitized** (e.g., CRED for credit cards, INDmoney for investments).
- Pick **sectors you've used** or relate to. Empathy helps with ideation.

User Research – Optional, but Adds Credibility

While not always mandatory, good user research strengthens your solution.

- Ask **open-ended, unbiased** questions.
- Use the **YC framework**: "What's the hardest part about this?", "Why is it hard?".

Solution Slide – Clear, Concise, Impactful

- Focus on how your solution helps **each stakeholder**.
- Add **arrows or notes** on wireframes to guide quick reading.
- If you did surveys, show **how many users validated your idea** (e.g., 60% creators said they would pay for it).

Wireframing – Make Ideas Tangible

Wireframes bring your solution to life visually.

- Use **Figma** for high-fidelity mocks, **Balsamiq** for low-fidelity wireframes.
- Repurpose app screenshots for faster mocks.
- Use Figma plugins: **Charts**, **Iconify**, **Lorem Ipsum**, **Humaans**.

Metrics – Show You Think Like a PM

Your deck should include metrics that prove your solution's effectiveness.

- Define your **goal-first**: what does success look like?
- Use **AAAERR (Pirate metrics)** and define a **North Star Metric**.
- Use **ratios over raw numbers** (e.g., conversion rate > # of users).

Pitfalls & Bottlenecks – Don't Skip This!

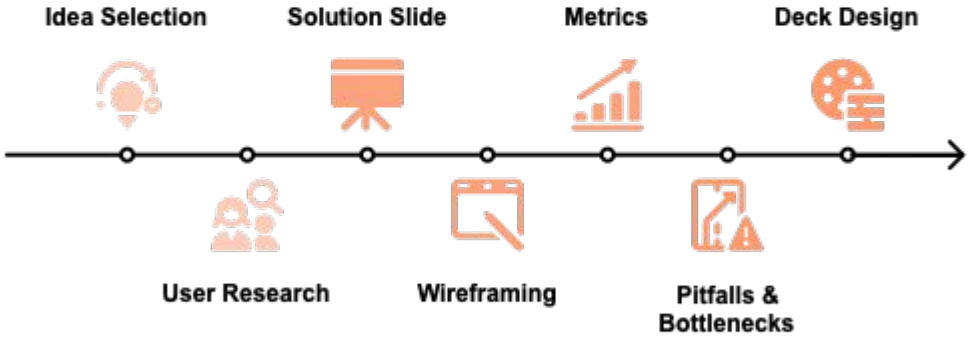
Most people forget this step, but it can show depth in your thinking.

- List **risks**, **assumptions**, and **potential blockers**.
- Add **workarounds** or next steps to navigate them.
- Even if there are no major pitfalls, include critical dependencies.

Deck Design – From Functional to Fantastic

You're usually not presenting these live, so the design must **tell the story on its own**.

- **Establish Slide Rules**: Set a base structure for solution, metrics, and wireframes so you don't start from scratch each time.
- **Use Brand Colors**: If your idea is for a known product, use their color palette
- **Tool Choice**: Use **Google Slides** for speed, **Canva** for polish. Templates are useful but may limit space flexibility.





FOCUS

02

Building Your Profile and Acing Interviews:

Interview Preparation

How to Approach a PM Interview

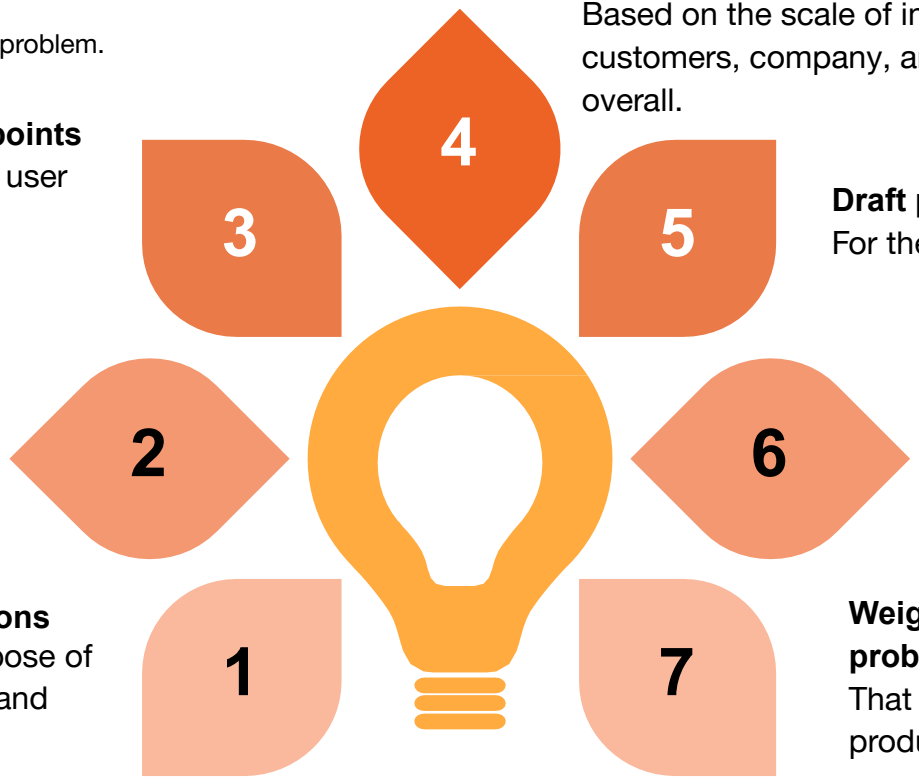
- **Understand the Problem Clearly**
  Identify key areas, ask clarifying questions, and confirm your understanding with the interviewer to ensure alignment on the context and objective.
- **Classify the Problem Type**
  Categorize the case (Product Design, Metrics, Go-to-Market, Strategy), and decide whether to apply an existing framework or break it into manageable parts.
- **Build and Use a Framework**
  Create a structured approach starting with the goal and assumptions. Use this framework to guide your analysis and decision-making.
- **Analyze and Ideate Solutions**
  Work through the problem step-by-step. Communicate your thought process clearly, propose solutions, and include at least one bold or innovative idea (e.g., tech or model change).
- **Conclude with Recommendations**
  Evaluate all ideas by feasibility, impact, and cost. Recommend the best one with an implementation plan, then summarize findings and prepare for follow-up questions.

Recommended Framework for PM Interviews

*General Approach.
May vary according to the problem.

Prioritise pain points
Based on the scale of impact on customers, company, and the industry overall.

List down the pain points
Faced by each of the user segments based on various conditions.



Draft possible solutions
For the prioritised pain points.

Create User Personas To identify potential user segments.

List KPIs/ Metrics
To analyse the impact and strategic fit of the solutions suggested.

Ask Clarifying Questions
To understand the purpose of improving the product and state assumptions.

Weigh trade-offs and probable challenges
That could arise during product launch.

How to Approach A Technical Interview Question

When tackling a technical interview question, there are a couple steps to keep in mind:

1. **Understand what is being asked**. If necessary, ask to clarify the goal and the problem statement.

2. **Work through the simple base case**. The solution and the accompanying clarifying questions will be more apparent as you work through it.

3. **Talk aloud**. It'll help the interviewer understand where you're headed. The interviewer may even intervene and provide helpful guidance.

4. **Write the technical solution, if necessary**. For most organizations, they don't expect the syntax to be perfect; pseudo code is fine.

5. **Review the code**. Evaluate the strengths and weaknesses. Revise as necessary. Few can write code without bugs in the first pass.

Common Technical PM Interview Questions

- Explain to me the architecture of a past technical product your team built
- Share a technically complex problem your team had to solve. What were some of the solutions the team came up with? What were the components that took the most time?
- What are the top 3 technology trends that will change the landscape in the next decade?
- Tell me about a time you handled a difficult technical stakeholder.
- What are the key conflicts between the development and business teams?
- Have you designed a technical solution that then became a commercial app?
- Talk about a technical blocker and what you
- What's your take on the latest engineering methodologies in our field?
- How do you integrate technical teams into the product vision effectively?

Common Technical PM Interview Questions

- Can you share an experience where a technical concept became a commercial success?
- How do you bridge the gap between market-oriented teams and technical challenges?
- Describe a successful cross-functional collaboration you were part of.
- How do you prioritize user experience in the design process?
- Describe a situation where you had to advocate for the user's needs against business pressures.
- What techniques do you use to stay ahead of industry trends?
- How do you foster a culture of continuous improvement within your team?
- How do you think Product Managers interact with engineers?
- What are the key technical challenges in developing a new product?
- How do you align engineering goals with market demands?

How to Approach A Behavioral Interview Question

- **Understand Why They're Asked**
  Behavioral questions are designed to assess both your **experience** and **communication skills**.
  Interviewers want to:
- Verify your resume matches your real contributions
- Understand the **scale and substance** of your challenges and impact
- Assess if you have the **skills, leadership qualities, and mindset** suited for the PM role

- **Structure Your Answers: Nugget + S.A.R.**
  Use a structured response for clarity:
- **Nugget First**: Start with a one-line summary (e.g., "Let me tell you about leading a product pivot.")
- **S**ituation: Give concise context
- **A**ction: Focus on what *you specifically* did
- **R**esult: Share measurable outcomes or key learnings

  Bonus Tip: Speak in bullets ("First, I did..., Then..."), it helps interviewers follow your thought process easily.

How to Approach A Behavioral Interview Question

Build a Preparation Grid

Create a table mapping major resume sections (jobs/projects/extracurriculars) to common themes:

- Leadership / Influence, Teamwork, Successes, Challenge, Mistakes / Failures
  This helps you **recall relevant stories** during the interview and ensures broad coverage.

Master 5 Key Stories

Choose five strong, flexible stories that:

- Have a substantial situation, action, and result
- Reflect who you are (creative, analytical, empathetic, resilient, etc.)
- Focus on “I” not just “we” – your contributions should stand out
- Show your ability to **understand and influence others**

Practice + Expect Follow-Ups

- Rehearse your stories aloud or with a peer—ask them to repeat back the situation, action, and result
- Prepare for follow-up questions like:
  - How did the team react?, What did you learn?, What would you do differently?, Was this your usual approach?

Common Behavioral PM Interview Questions

- Describe a decision you made that wasn't popular. How did you handle implementing it?
- Describe a time when you had to motivate employees or coworkers.
- Tell me about a time when you showed initiative.
- Tell me about a time when you had to give a presentation to people who disagreed with you.
- Tell me about a time when you had to make an unpopular decision.
- Tell me about a time when you had to sell another person or team on your idea
- Tell me about a time when you've built a team.
- What work accomplishment are you most proud of?
- Describe a scenario which required you to say no to an idea or project.
- Tell me about a time when you faced a challenge and overcame it.
- Tell me about a time when you weren't able to reach a deadline.
- Describe a major change that occurred in a job that you held. How did you adapt to this change?
- Tell me about a time when you had to deal with changing priorities. How did you handle it?

Common Behavioral PM Interview Questions

- Tell me about a time when you had to make a decision quickly or with insufficient data.
- Tell me about a time when you used a lot of data in a short period of time.
- Tell me about a time when you handled a risky situation.
- Tell me about a time when you made a mistake.
- Tell me about a time when you failed.
- Tell me about a time when you improperly analyzed a situation.
- Tell me about a time when you were disappointed with yourself.
- Tell me about a time when you were unable to juggle all your responsibilities.
- Tell me about something you're proud of accomplishing.
- Tell me about a time when you reached a goal that was important.
- Tell me a specific insight you gained from something outside of work.
- Tell me about a time when you had to work across teams to accomplish something.
- Tell me about a time when you had a disagreement at work.
- Tell me about a time when you mentored or aided a coworker.
- Tell me about a time when you had to do something you didn't want to do.
- Tell me about a time when you had to compromise.



FOCUS

03

Building Your Profile and Acing Interviews:

Mastering The Interview

General layout:

- Guesstimate
- Product/Business case study
- Strategy based problem (Pertaining to a new product/market, or growth)
- Metrics-based questions
- Product Critique
- Technical round (occasionally)
    - SQL-related question
    - Techstacks of projects on your CV

Top 5 Do's in a PM Interview

- **Ask Clarifying Questions**
  Show active listening and ensure you fully understand the problem before jumping in. Clarify assumptions and goals upfront.
- **Structure Your Approach**
  Break down problems systematically and walk the interviewer through your plan with a clear, logical structure.
- **Think Customer-First**
  Always center your answers around user needs and pain points—not personal preferences. Focus on customer outcomes.
- **Back Decisions with Data**
  Use relevant metrics or user insights to justify your decisions and show you're data-informed in your thinking.
- **Communicate Thought Process Clearly**
  Talk through your reasoning step-by-step so the interviewer can follow how you think and solve problems.

Top 5 Don’ts in a PM Interview

- **Don’t Stick Rigidly to Frameworks**
  Frameworks are helpful, but don’t force-fit them. Be flexible and adapt your approach based on the context.
- **Don’t Speak Before Thinking**
  Avoid rushing into answers. It's okay to pause, think, and organize your thoughts before responding.
- **Don’t Ignore Trade-Offs**
  Failing to prioritize or consider constraints weakens your response. Show how you balance competing needs.
- **Don’t Rely Only on Opinions**
  Avoid basing answers on personal biases. Instead, anchor your reasoning in user needs, market context, and data.
- **Don’t Treat the Interview as an Interrogation**
  It's a two-way conversation. Stay calm and engage naturally—you're also assessing if this team and role fit you.

PM Interview with the VP of the Product

Reaching the VP, Director, or Head of Product stage in a PM interview signals you're near the finish line. This individual will be your future manager, so their decision carries weight.

What the Interviewer Wants to Know

- **Evaluate Your PM Skills**
  Expect a deep dive into your product management experience, especially the finer details.
- **Assess Your Ability to Handle the Role**
  They'll gauge if you fit the specific needs of their team and environment like collaboration or leadership.
- **Determine Cultural and Personality Fit**
  They'll consider if you're coachable, pleasant to work with, and someone they'd enjoy mentoring.

5 Key Tips for Success

- **Be Ready for a Range of PM Questions**
  Prepare for questions across product estimation, design, metrics, improvement, strategy, problem solving, launches, execution, behavioral fit, and technical understanding.
  Common questions include:
  - “Tell me about yourself and your experience.”
  - “Why do you want to be a product manager?”
- **Ask Thoughtful Questions Yourself**
  Show genuine interest by asking about:
  - The structure and dynamics of the product team
  - Cross-functional collaboration
  - Strategic priorities
  - The development process
  - Who you’ll be working with closely

5 Key Tips for Success

- **Know the Interviewer and Their Background**
  Research their experience and role history. Understand their perspective and values to align your responses effectively.
- **Understand Their Key Challenges**
  Identify patterns or recurring themes from earlier interview rounds to infer what challenges the team or leadership is focused on. Use this context to tailor your answers.
- **Talk to Your Recruiter**
  Ask what the interview will focus on, and refine your prep accordingly. They can give helpful insights into what the interviewer prioritizes and expects.

Top 6 Do's in a Case Interview

- **Listen & Interact**
  Engage actively with the interviewer. Ask clarifying questions and adjust your direction based on new information. Think out loud and note key findings.
- **Structure the Problem**
  After initial discussion, lay out a clear, simple framework to guide your approach. Take a moment to think before responding.
- **Focus on High-Impact Areas**
  Prioritize issues with the greatest value potential. Make your reasoning behind selected priorities clear.
- **Think Creatively**
  Explore diverse options and consider second-order effects. Go beyond the obvious and factor in trade-offs and company context.
- **Use Business Judgment & Quantitative Thinking**
  Assess feasibility and impact of solutions. Be ready to perform quick calculations and integrate them into your analysis.
- **Summarize & Conclude**
  Revisit the original question, recap key insights, and present clear, actionable recommendations.

Top 6 Don’ts in a Case Interview

- **Don’t Jump In Blindly**
  Ensure full understanding of the problem before analyzing. Ask questions if anything is unclear.
- **Don’t Panic**
  You’re not expected to know everything. Stay calm, share your hypotheses, and collaborate with the interviewer.
- **Don’t Be Rigid**
  Avoid defending flawed answers. Be flexible and adjust based on feedback or new facts.
- **Don’t Internalize Thoughts**
  Always verbalize your logic. Treat the case as a dialogue, not a test with one correct answer.
- **Don’t Over-Rely on Frameworks**
  Use them as tools, not scripts. Avoid applying irrelevant or overly complex structures.
- **Don’t Fake or Use Leaked Cases**
  If the case is familiar, disclose it. Authenticity matters—be yourself, not who you think they want to see.

Interview Experience

Disclaimer:

The interview experiences shared here have been collected by our team to the best of our knowledge. Please note that actual experiences may vary for different candidates. This content is intended to provide a general overview and should not be considered definitive.

General Tips And Experience By Past IITG Alumni

Your CV plays a critical role—invest time in making it stand out.

Impactful Projects: Highlight projects with measurable outcomes. Inter-IIT projects, recognized hackathons (especially by reputed firms), and meaningful self-initiated work add strong value.

Internships: Having at least one internship is essential. Experience at a startup is especially appreciated, as it reflects adaptability and ownership.

Positions of Responsibility (PoRs): Companies seek leadership potential. Make sure to include relevant PoRs that showcase initiative, team management, or strategic contributions.

CPI (CGPA): For non-tech roles, academics do matter. Aim for a **CPI of 8.0+ for internship season** and **7.0+ for final placements**.

Zomato – Product Analyst Role

About The Company

1. ***Zomato, founded in 2008 by Deepinder Goyal and Pankaj Chaddah**, began as a restaurant discovery platform and has evolved into a comprehensive food delivery and dining services provider, operating in over 800 Indian cities.*
2. ***In 2025, Zomato rebranded as 'Eternal Limited'**, encompassing four major business units: Zomato (food delivery), Blinkit (quick-commerce), District (live events), and Hyperpure (kitchen supplies).*
3. ***Zomato holds a 58% market share in India's food delivery sector**, with a gross order value of ₹322.24 billion ($3.82 billion) in FY24, and anticipates a 30% annual growth rate over the next five years.*

Round 1: Online Aptitude Test
The first round was an online assessment that evaluated general aptitude and problem-solving skills.

Round 2: Offline Interview (Technical + CV Discussion)
This round was conducted in person and focused primarily resume. The interviewer asked detailed questions about:

- Internship experiences/Projects undertaken/Positions of Responsibility (PORs) and impact created
- RCA/Guesstimate type question.
- **Puzzle:** I was asked a well-known probability puzzle involving 100 coins, where 5 coins show tails and 95 show heads. The task was to divide them into two groups with equal tails — a classic from GeeksforGeeks.
- **SQL Question:** Asked to list the different types of SQL joins and explain which type would generate the maximum or minimum number of rows.(Had to write in pen paper only.)

OLA – Business Analyst Role

About The Company

1. ***In August 2024, Ola rebranded to Ola Consumer****, expanding its offerings beyond transportation to include financial services and food delivery, while focusing operations solely within India.*
2. ***Ola Electric, a subsidiary established in 2017****, manufactures electric two-wheelers and achieved a significant milestone with its $734 million IPO in 2024, marking India's largest IPO that year.*

Round 1: Online Aptitude Test
The process began with an online assessment focused on logical reasoning, quantitative aptitude, and problem-solving skills,python,data science(Questions on regression and classification).

Round 2: Offline Interview (Technical + Analytical Round)
This round was held in person and tested both technical proficiency and business acumen.

Key Components:

- **Puzzle & Data Science:** A logical puzzle similar to those found on GeeksforGeeks was asked to assess problem-solving under constraints.(Similar to GFG Puzzle on probability and statistics)
- **SQL:** Asked to write SQL queries using pen and paper, testing foundational understanding and query structuring ability.(Window function and regex type questions were asked to code on HackerRank)
- **Root Cause Analysis (RCA):** Given a scenario to identify core issues and propose actionable solutions.
- **Business Metrics:** Questions were asked around key performance indicators, especially those related to economic efficiency and product performance.

Media.Net - Senior Product Analyst

About The Company

1. ***Media.net is a leading global contextual advertising platform*** *that offers publishers cookie-free, keyword-targeted ads powered by the Yahoo-Bing network, enhancing user privacy and engagement.*
2. ***The company provides a comprehensive suite of monetization solutions****, including native, display, video, and server-side header bidding formats, serving over 125 billion ads annually across more than 1 billion URLs.*

Round 1: Online Aptitude Test

The process began with an online assessment focused on logical reasoning, quantitative aptitude, and problem-solving skills,data science(Linear Regression,classification model,probability),SQL(3 questions).

Round 2: Offline Interview (Technical + Analytical Round)

This round was held in person and tested both technical proficiency and business acumen.

Key Components:

- **Metrics Based:**Questions like state 15 metrics that you would want to have in your dashboard & prioritise them.
- **SQL:** Interview questions on interview bit.(Questions related to Primary Keys,yu had to write in chat box)
- **Root Cause Analysis (RCA):** Given a scenario to identify core issues and propose actionable solutions.
- **Guesstimate:**A general guesstimate question was asked.
- **CV:**In Depth cv overview and questioning.

HiLabs - APM + DS Roles

About The Company:

1. ***HiLabs is a healthcare AI company*** *that cleans and enriches messy provider and claims data using advanced NLP and machine learning tools.*
2. ***Its flagship product, MCheck™,*** *has corrected millions of data errors across 35+ billion records, improving healthcare operations and payments.*
3. ***Backed by $39M Series B funding,*** *HiLabs is globally active with offices in the U.S. and India, helping health plans improve data accuracy and access to care.*

Round 1: Deck Submission- This round hilabs had released the problem statement and people had to submit their decks to go on to further rounds.

Round 2:Interview -

Key Components:

- **Root Cause Analysis (RCA):** Given a scenario to identify core issues and propose actionable solutions.
- **Deck Question:** Maximum time on problem identification
- A bit of ML and DL side preferred they wanted product managers who have ML knowledge as well

Indus Insight - Associate Consultant Roles

About the Company

1. ***Indus Insights, founded in 2009 by Saurabh Sharma and Neha Sharma****, is a Gurugram-based data science consulting firm specializing in advanced analytics and machine learning solutions for the financial services sector .*
2. ***The company has delivered significant results****, such as a 25% increase in credit card applications through machine learning models and a 10% reduction in loan losses via smarter credit scoring .*
3. ***In 2022, Indus Insights secured $6 million in growth capital from Helix Investments****, aiming to expand its global presence and enhance its analytics capabilities .*

Round 1: Online Aptitude Test

The process began with an online assessment focused on logical reasoning, quantitative aptitude, and problem-solving skills,DILR.

Round 2: Offline Interview (Technical + Analytical Round)

- **Guesstimate:**Asked a guesstimate question(No. Of Lays packets sold in Delhi).
- **Root Cause Analysis (RCA):** Given a scenario to identify core issues and propose actionable solutions.
- **CV:**Questions were asked on experience and projects.

No Broker - Product Analyst

About The Company

1. ***NoBroker is a Bengaluru-based proptech platform founded in 2014*** *by Amit Kumar Agarwal, Akhil Gupta, and Saurabh Garg. It eliminates brokerage fees by directly connecting property owners with tenants or buyers through a customer-to-customer (C2C) model .*
2. ***The platform offers a range of services****, including property listings, rental agreements, home loans, packers and movers, and society management through its NoBrokerHood app. It leverages AI and machine learning to match users with suitable properties .*
3. ***NoBroker became India's first proptech unicorn in 2021*** *after raising $210 million in a Series E funding round led by General Atlantic and Tiger Global, achieving a valuation of $1.01 billion .*

Round 1: Online Aptitude Test

It had one of the most difficult OA according to seniors.(Questions similar to CAT only).Had to type answers in some of the question.

Round 2: Offline Interview (Technical + Analytical Round)

- **Root Cause Analysis (RCA):** Given a scenario to identify core issues and propose actionable solutions.
- **Guesstimate:**Asked a guesstimate question(Like No. Of Lays packets sold in Delhi).
- **CV:**Questions were asked on projects,POR,experience.
- **SQL:**Basic question asked to code

Meesho

About The Company

1. ***Meesho, founded in 2015 by IIT Delhi alumni Vidit Aatrey and Sanjeev Barnwal****, is a Bengaluru-based e-commerce platform that empowers small businesses and entrepreneurs to sell products online without any upfront investment.*
2. ***The platform leverages social media channels like WhatsApp, Facebook, and Instagram*** *to facilitate sales, focusing on unbranded, affordable products across categories such as fashion, home, and beauty, primarily targeting consumers in India's Tier-2 and Tier-3 cities.*
3. ***Meesho became India's first social commerce unicorn in April 2021****, achieving a valuation of $2.1 billion following a $300 million funding round led by SoftBank Vision Fund 2.*

Round 1: Online Aptitude Test & CV Based Shortlisting

First it had an OA with similar questions (CAT)level & then cv based shortlisting was done for further rounds.(Role preference was asked)

Round 2: Offline Interview (Technical + Analytical Round)

- **Projects:** Talked on projects and internship experiences.
- **Guesstimate:**Asked a guesstimate question(Like No. Of Lays packets sold in Delhi).
- **POR'S**:Asked about how and what have you done during this POR.
- **SQL:**Basic question asked to code
- **RCA Questions**

APPENDIX

NPS(Net Promoter Score)
Net promoter score is a widely used market research metric that typically takes the form of a single survey question asking respondents to rate the likelihood that they would recommend a company or a product.

$$\text{NPS} = \%\ \text{promoters} - \%\text{detractors}$$

Social Share of Voice (SSoV)
It is a way to measure how much people are talking about the brand on social media. It's usually calculated as a percentage of total mentions within an industry or a defined group of competitors.

$$\text{Social Share of Voice} = \frac{\text{Your brand's advertising spend}}{\text{Total advertising spend in the market}} \times 100\ \%$$

FICO Score
Financial institutions and lenders use this to determine how much credit they can offer borrowers and at what interest rate.

Knowledge Process Outsourcing
KPO refers to outsourcing knowledge-intensive activities that are data-driven and encompass the process of gathering, managing, analysing and delivering objective insights into businesses.

Personally Identifiable Information (PII)
PII refers to any user data that could be used to distinguish one person from another. Standard PII identifiers include phone numbers, email addresses, social security numbers or mailing addresses.

Return on Ad Spend (ROAS)
It is a marketing metric that measures the efficacy of a digital advertising campaign. ROAS helps online businesses evaluate which methods are working and how they can improve future advertising efforts.

$$\text{ROAS} = \frac{\text{Total Campaign Revenue}}{\text{Total Campaign Cost}}$$

Client Retention Rate
It is used to measure the number of loyal customers, i.e. the customers who continued to purchase the product and are satisfied with it. The percentage of customers and revenue lost during a period is called the **Churn Rate**.

Client Lifetime Value (CLV)
It is the profit brought by a customer to the business, and in this way, the company can individually fulfil the need of the customer.

$$\text{CLV} = \text{Lifetime Value} \times \text{Margin}$$

Time Value of Money
A sum of money today is worth more than the same sum in the future, as it can earn interest in the meantime. A delayed investment is a lost opportunity.

Cash Flows

The term cash flow refers to the net amount of cash and cash equivalents transferred in and out of a company. Cash received represents inflows, while money spent represents outflows.

Free Cash Flow

It's the residual cash flow that's left over after all of the project's requirements have been satisfied, and the implications accounted for. It's the cash flow that can be distributed to the financial claimants of the company, debt and equity.

$$FCF = (\text{Revenue} - \text{Costs} - \text{Depreciation\#}) \times (1 - \text{tax rate}) + \text{Depreciation} - \text{Capital Expenditures} - \text{Change in Net Working Capital}$$

\# - EBIT (Earnings before interest and taxes)

Free Cash Flow to Equity

It's the residual cash flow left over after all of the project's requirements have been satisfied, implications accounted for, and all debt financing has been fulfilled.

Present Value

The present value of a sum FV received in t periods discounted at a rate of interest r is given by

$$PV = \frac{FV}{(1 + r)^t}$$

The present value of a future stream of cash flows CF_t time intervals is given by

$$PV = \sum_{i=1}^{n} \frac{CF_t}{(1 + r)^t}$$

Return on Assets (ROA)

It is a measure of how much profit a business is generating from its capital. If a company's ROA is 7.5%, this means that the company earns seven and a half cents per dollar in assets.

$$\text{Return on Assets} = \frac{\text{Net Income}}{\text{Total Assets}}$$

Annuity and Perpetuity

An annuity is a finite stream of cash flows of identical magnitude and equal spacing in time.

Annuity Formula (Present Value):

For an **ordinary annuity**:

$$PV = P \times \left(1 - \frac{1}{(1+r)^n}\right) \div r$$

Where:

- P = Payment per period
- r = Interest rate per period
- n = Total number of payments

Future Value of an Annuity:

$$FV = P \times \left(\frac{(1+r)^n - 1}{r}\right)$$

Perpetuity: A **perpetuity** is a stream of **equal payments** made at **regular intervals forever** (no end date).

Examples: Endowments paying yearly scholarships

Perpetuity Formula (Present Value):

PV=P/r

Where:

- PV = Payment per period
- r = Interest rate per period

A growing perpetuity is an infinite stream of cash flows that grow at a constant rate and are evenly spaced through time.

$$PV = \frac{CF}{r - g}$$

Discount Rate/ Rate of Return

Factors like inflation and taxes influence r, i.e. the rate of return/discount rate. In the case of inflation, where the inflation rate is given by I, the inflation-adjusted resultant rate of return, say RR, can be calculated as $RR = \frac{1 + R}{1 + i} - 1$.

In the case of taxes, if the rate of taxation is t, then the resultant rate of return is RR = R (1 - t).

Net Present Value

NPV is simply the difference between the present value of cash inflows and outflows. It is a metric used to determine whether an investment is profitable. Investments with negative NPVs

are avoided. The formulae remain the same as PV's; however, we take the net cash inflow- outflow or the free cash flow in this case.

Net Present Value Rule
The NPV Rule recommends making only investments with a positive net present value.

Internal Rate of Return(IRR)
The internal rate of return is the discount rate, which would equal the NPV to zero. It helps to identify the annual growth rate. A high IRR is desirable for investment.

Net Profit Margin
It is the ratio of net profits to the revenue expressed as a percentage. It represents how much of each dollar collected in revenue translates into a profit.

$$\text{Net Profit Margin} = \frac{\text{Net Income}}{\text{Total Revenue}} \times 100$$

Break Even Point
In accounting, it refers to the production quantity where the total production revenue compensates for production costs.

Return on Investment (ROI)
ROI indicates the total growth of an investment over a period of time. The ROI is the net profit expressed as a percentage of the initial investment.

$$\text{ROI} = \frac{\text{Return} - \text{Cost} - \text{Investment}}{\text{Investment}} \times 100$$

Compounded Annual Growth Rate(CAGR)
Given the amount invested today and its expected value in the future, CAGR is the rate of interest at which the investment will have to be compounded to give the expected future value.

$$\frac{\text{Final Value}}{\text{Beginning Value}}^{(1/t)} - 1$$

Profitability Index

The profitability index is an index that attempts to identify the relationship between the costs and benefits of a proposed project through the use of a ratio calculated as:

$$\text{Profitability} = \frac{\text{PV of future cash flows}}{\text{Investment}}$$

A ratio of 1.0 is logically the lowest acceptable measure on the index, as any value lower than 1.0 would indicate that the project's PV is less than the initial investment. As values on the profitability index increase, so does the financial attractiveness of the proposed project.

Initial Investment

The initial investment is the amount required to start a business or a project. It is also called initial investment outlay or simply initial outlay. It equals capital expenditures plus working capital requirement plus after-tax proceeds from assets disposed off or available for use elsewhere.

Capital Asset Pricing Model (CAPM)

The market rate of return is the return the company could receive by investing in a well-diversified portfolio of stocks. The discount rate can be calculated as:

$$r = r_f + \beta(r_m - r_f),$$

where r is the discount rate,
r_f is the risk-free rate of return,
r_m is the market rate of return, and
β is the calculated market risk of an investment.

Bounce Rate

The bounce rate is the percentage of people who come to the landing page and leave without browsing further or clicking elsewhere on the company's website.

Key Performance Indicator (KPI)

KPIs provide targets for teams to shoot for, milestones to gauge progress, and insights that help people across the organisation make better decisions.

Business Process Reengineering (BPR)

It includes reviewing a client's business processes, eliminating unneeded or 'non-value-added' tasks, and implementing a leaner, more efficient strategy.

Recommended Books

