Japneet Sachdeva

REVOLUTIONIZING AUTOMATION & CONTEXT DRIVEN QA

SENIOR SDET

CAREER COACH & CONTENT CREATOR

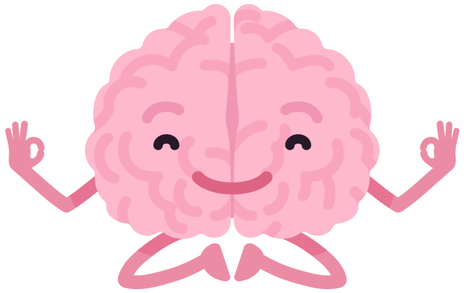
YouTube | LinkedIn | TopMate

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

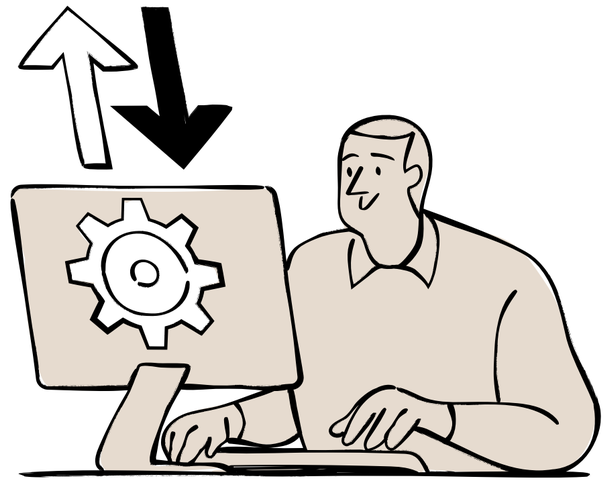
A Full Stack QA Understands Context of:

1) Business
2) Technology
3) Product
4) Customers

Why do we need it? Let's jump in a real world bug



1) Business: What are we selling?
2) Technology: How are we selling?
3) Product: What is it we are selling?
4) Customers: Whom we are selling?



What is Full Stack QA - SDET?

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

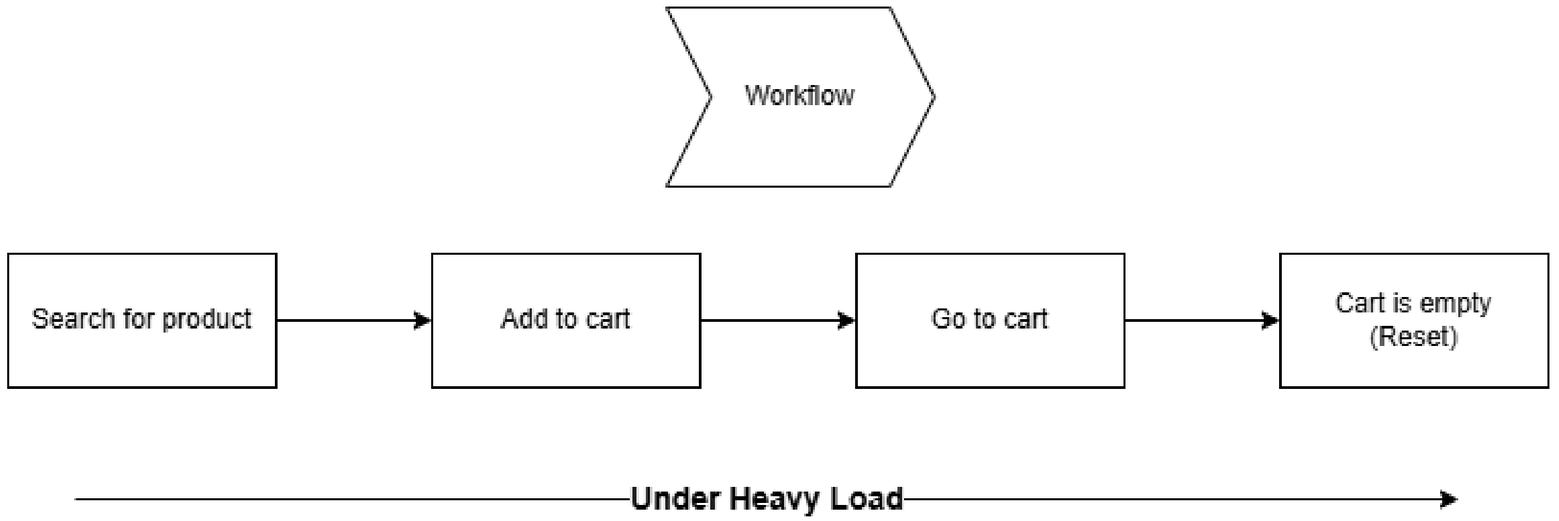
Scenario: Black Friday Sales on Amazon

Problem: At a major online retailer resembling Amazon, during peak shopping seasons, customers occasionally experienced abrupt cart resets and order failures. Users would add products, proceed to checkout, but—under heavy load—the system would suddenly clear their cart

What is Full Stack QA - SDET?

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

Scenario: Black Friday Sales on Amazon



What is Full Stack QA - SDET?

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

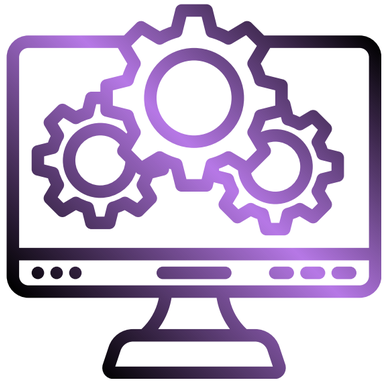
Scenario: Black Friday Sales on Amazon

What can a Full Stack QA - SDET do?

1) Validate Logs

Use **ELK Stack (Elasticsearch, Logstash, Kibana)** to aggregate logs from all microservices. Detailed logs from the session management service included timestamps for each API call, token generation, and cache access.

By correlating these logs, they noticed that when the ***response time went above 500ms***, the logs consistently showed overlapping token generation events—indicative of a race condition.



What is Full Stack QA - SDET?

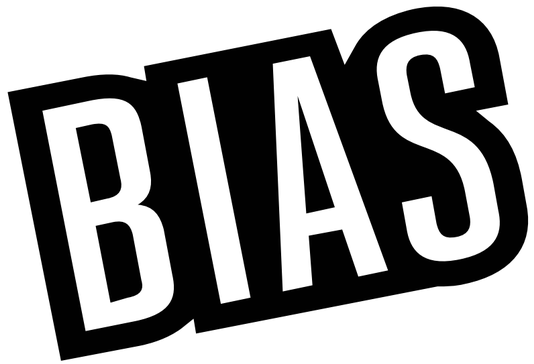
LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

Scenario: Black Friday Sales on Amazon

What can a Full Stack QA - SDET do?

2) Confirmation Bias

As we are Engineers, We do not support
Hence we deployed **New Relic** to confirm the metrics
It confirmed that the problematic API calls were indeed exceeding 500ms during load tests



Why Full Stack QA - SDET?

3) Simulation

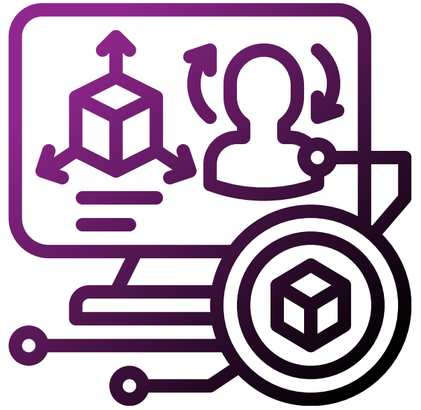
- **Realistic Load:**

By simulating thousands of concurrent users, the load tests recreated peak shopping conditions.

- **Environment Parity:**

The staging environment was configured to closely resemble production—including similar network latency, server configurations, and even simulated failures—to ensure that the tests were as realistic as possible

We can use tools like: ProxyMan, BurpSuite, Charles Proxy to simulate network failures like reduced bandwidth, network latency etc.



Why Full Stack QA - SDET?

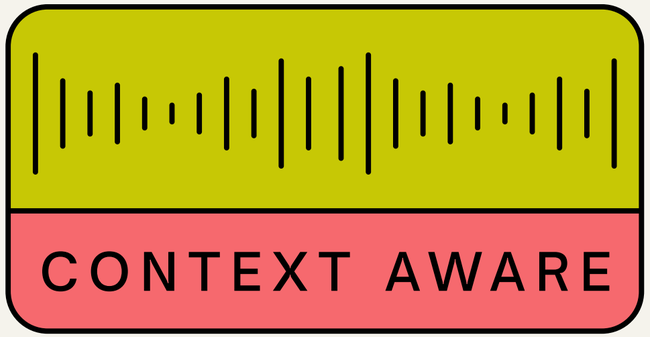
LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

4) Development Overview

- **Front-End Layer:** Built with modern frameworks (like React or Angular) to deliver a dynamic user experience.
- **API Gateway:** Routing requests to different back-end services.
- **Session Management & Cart Service:** A dedicated microservice handling user sessions and cart data.
- **Caching Layer:** Initially implemented as an in-memory cache within the session service.
- **Distributed Datastore:** Where order confirmations and transaction details were persisted.
- **Stress tests** were conducted in a staging environment—a mirror of production that was set up using containerized services (Docker, orchestrated via Kubernetes) to simulate real-world load conditions

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

Do not Test Until you are aware about the Context



Why Full Stack QA - SDET?

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

5) QA & Automation

- **End-to-End Functional Flow:** Automate the entire checkout process—from adding items to the cart, proceeding to checkout, to verifying the order confirmation. Tools like Selenium WebDriver can consistently execute these steps on every build
- **API and Integration Tests:** Use REST-Assured (or similar tools) to validate API responses from the payment gateway and session management services.
- **Load & Concurrency Tests:** Automated load tests (using JMeter, integrated with Jenkins CI/CD) simulate thousands of concurrent users to stress the system. These tests help identify performance bottlenecks and race conditions—like the 500ms delay
- **Log and Metrics Analysis:** Automated scripts can parse and correlate logs (from ELK stack or APM tools like New Relic) to detect when API responses exceed the 500ms threshold and trigger errors.

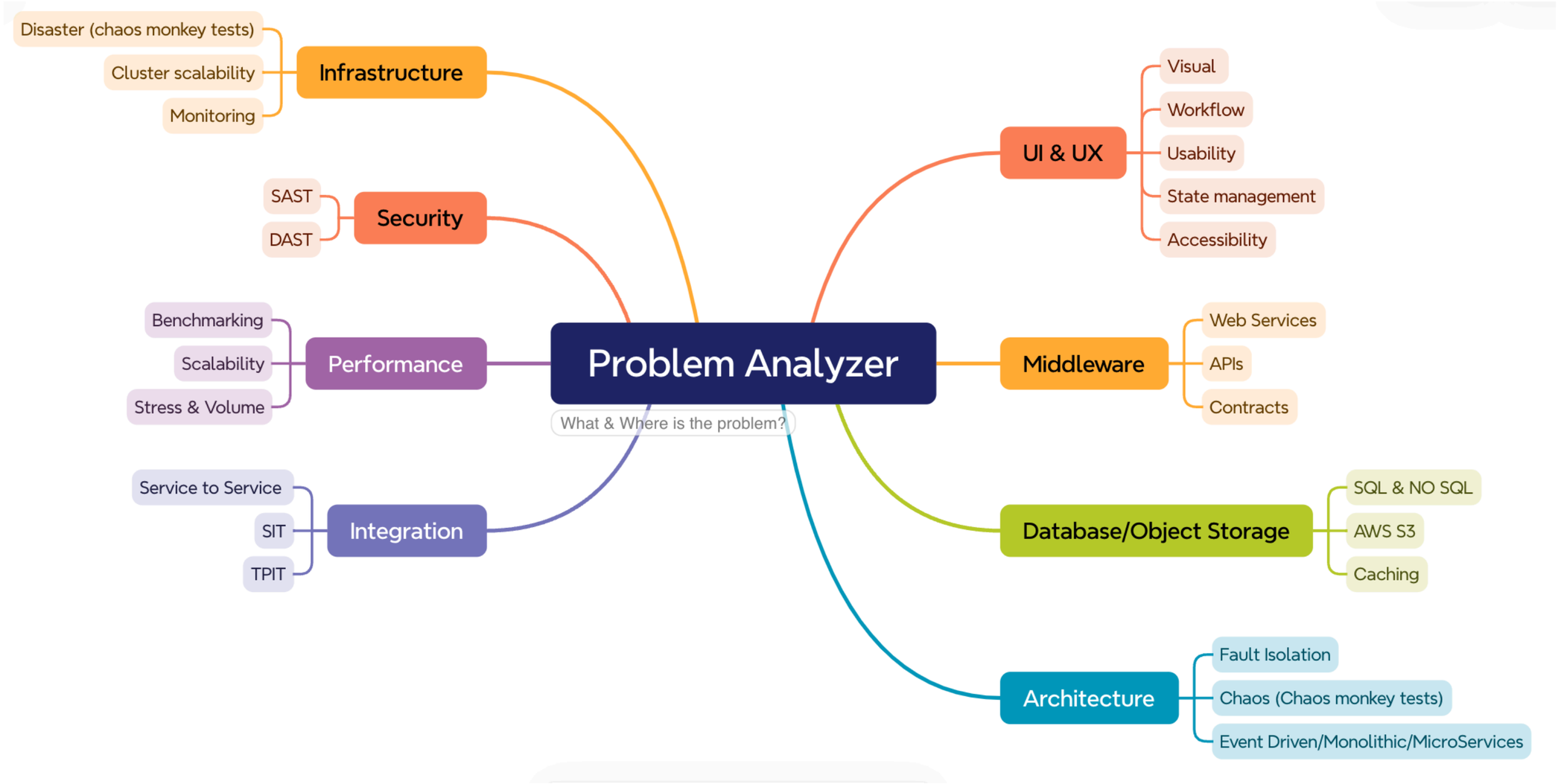
Reminder: Passing tests cannot prove software is good!

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

Why Full Stack QA - SDET?

Some common Problems a QA can find out?

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA



UI & UX

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

UI & UX

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

Prominent Automation Area's:

1) Workflow

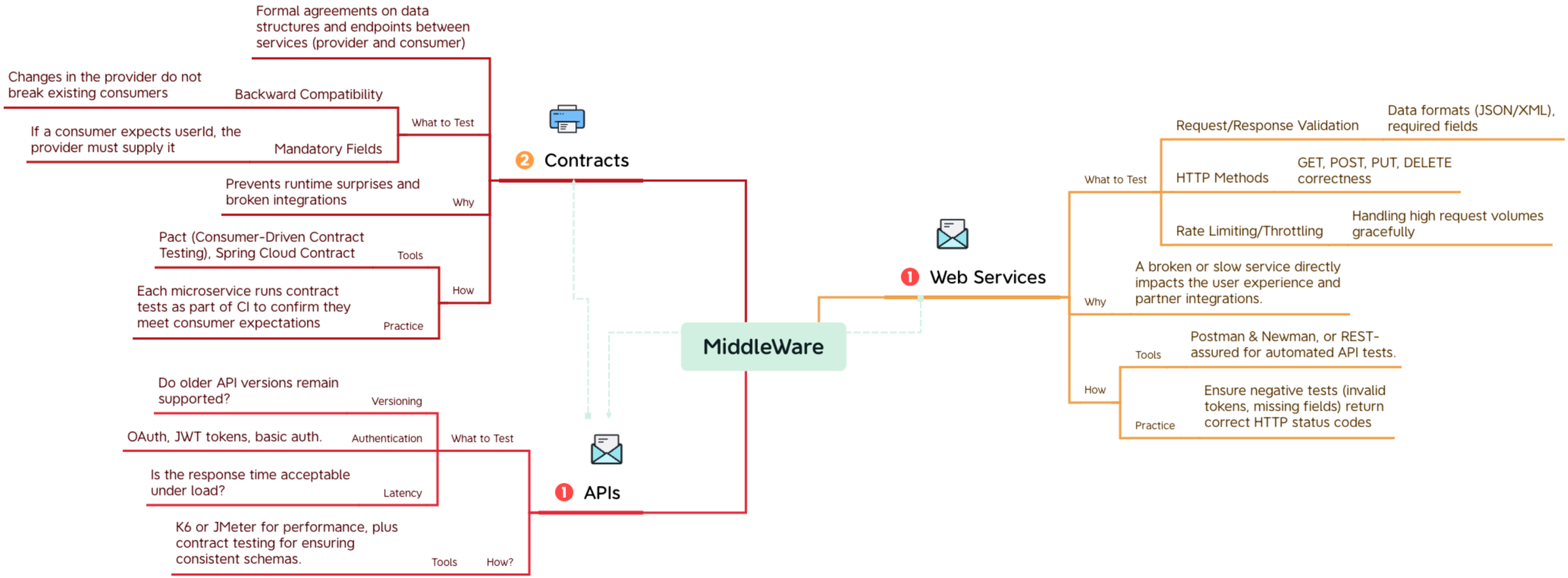
2) Usability

3) Visual

MiddleWare

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

Note: APIs vs WebServices
WebServices must need network to work, whereas APIs does not need.



MiddleWare

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

Note: APIs vs WebServices
WebServices must need network to work, whereas APIs does not need.

Prominent Automation Area's:

1) HTTP Methods
2) Authentications
3) Request & Response validations

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

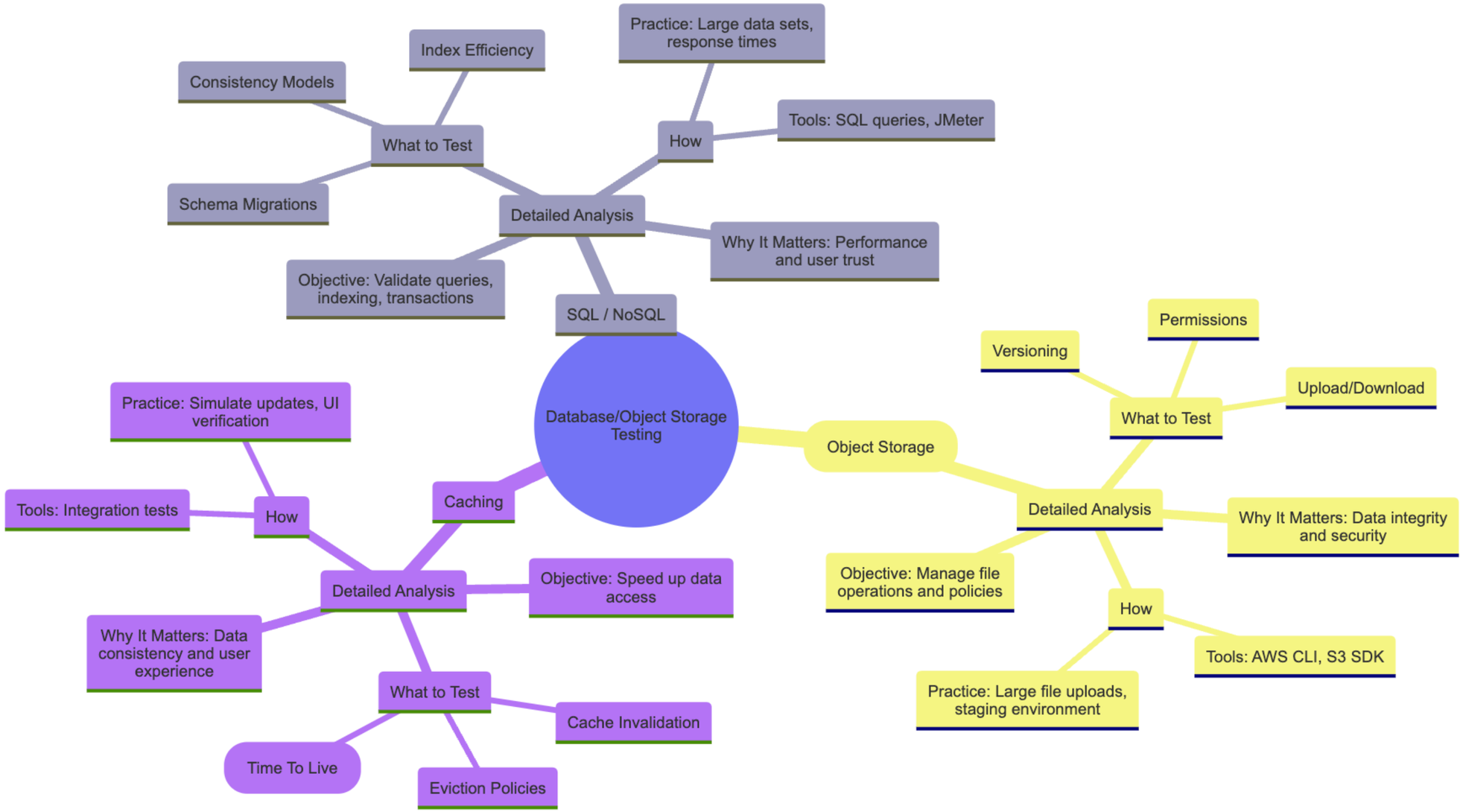
Its not about finding bugs

BUT

Its about ensuring better quality

Database & Object Storage

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA



Database & Object Storage

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

Prominent Automation Area's:

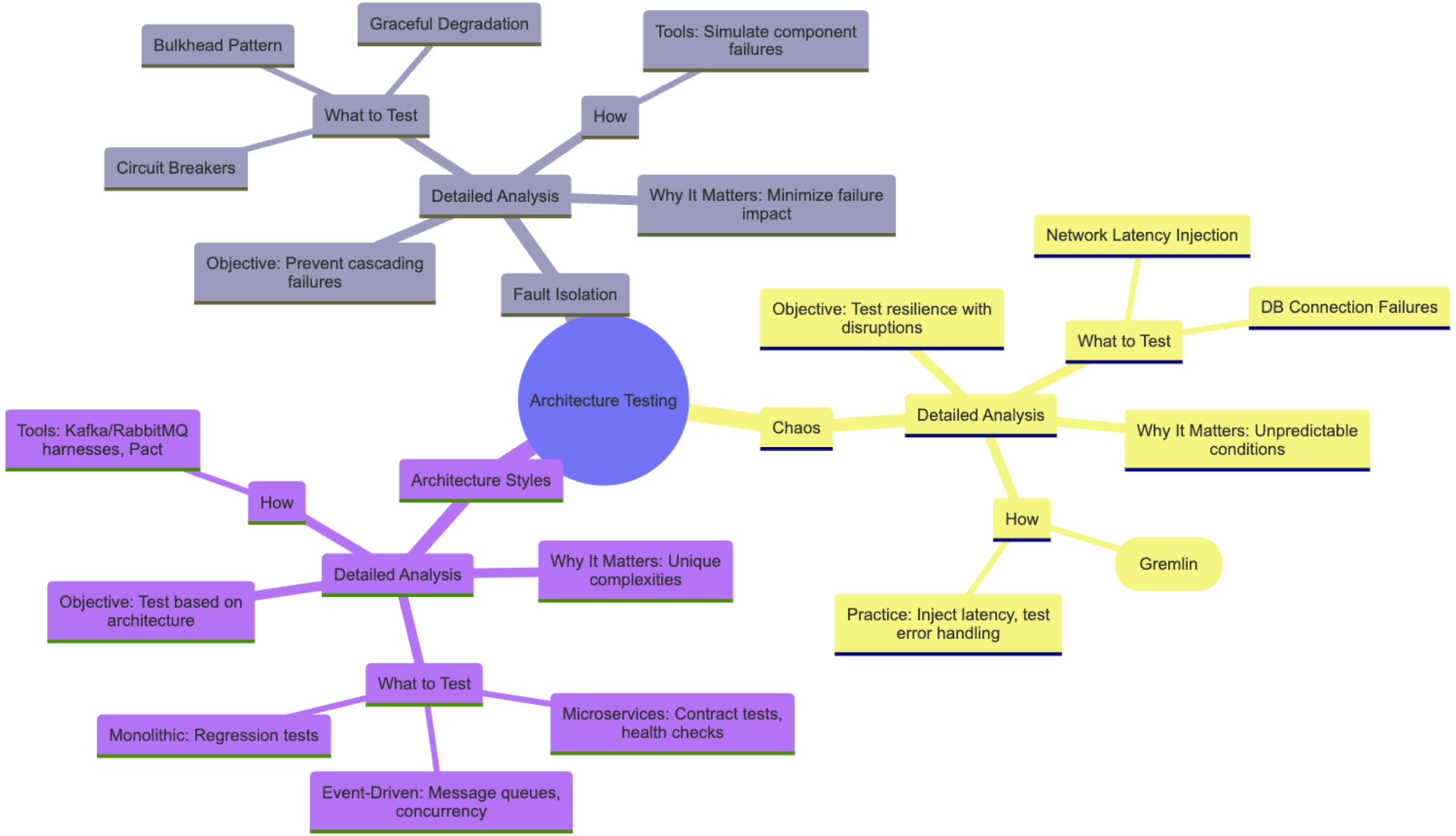
1) Cache Invalidation

2) Permissions Testing

3) SQL Index Efficiency

Architecture

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA



Architecture

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

Prominent Automation Area's:

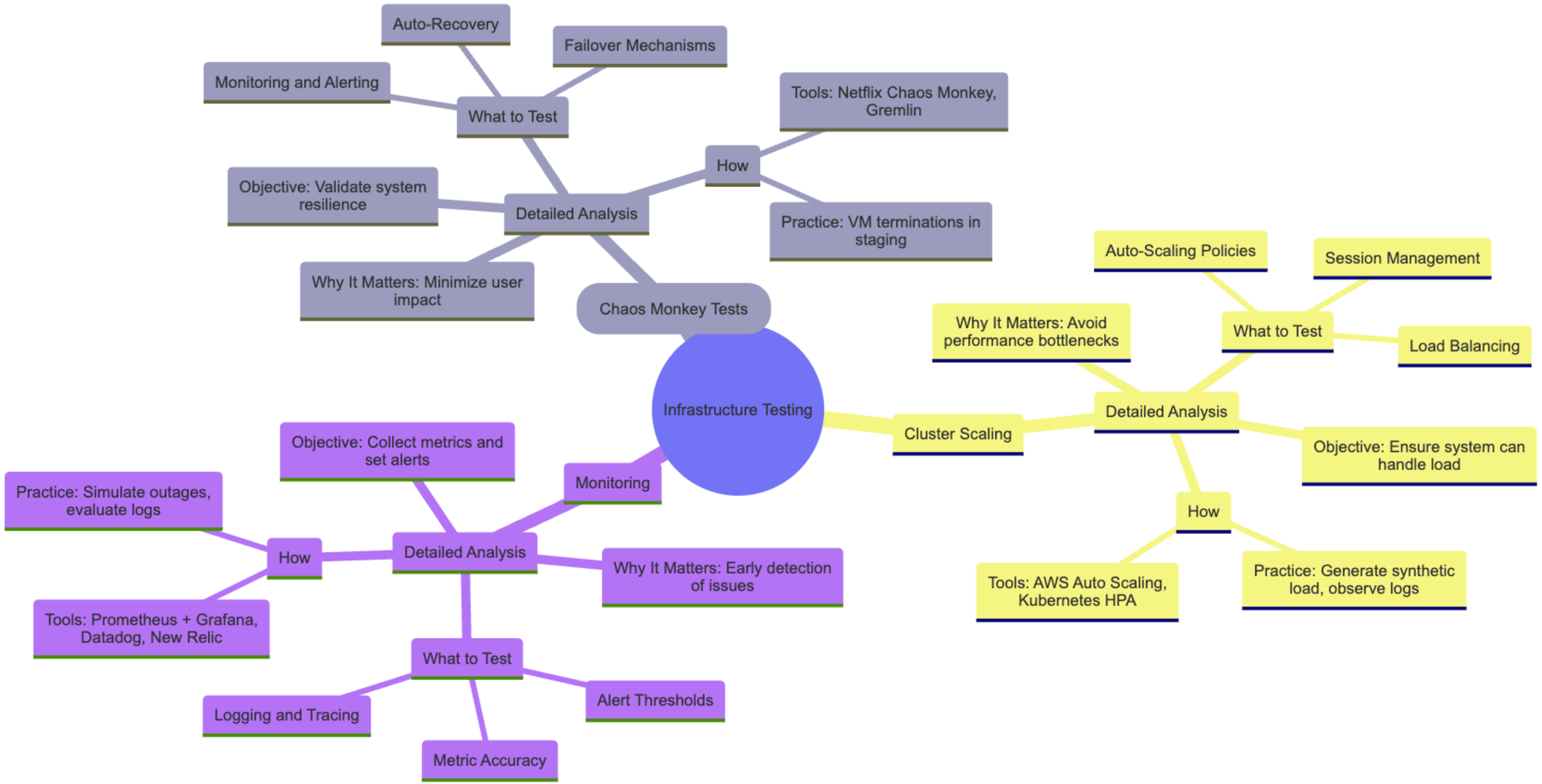
1) Graceful degradation mechanisms

2) Automated contract tests

3) Controlled Chaos

InfraStructure

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA



InfraStructure

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

Prominent Automation Area's:

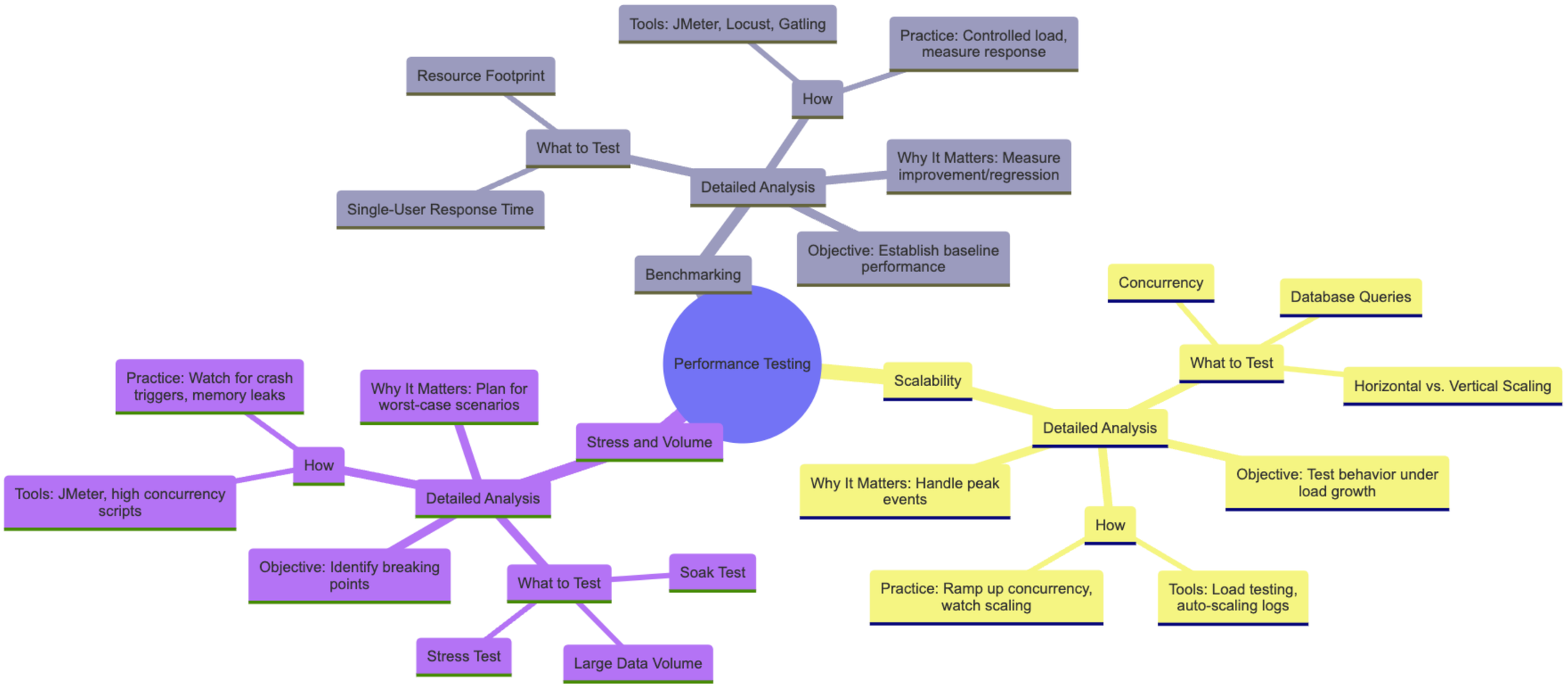
1) Auto Scaling

2) Alert delivery & Metric collection

3) Verify routing and DNS updates during failovers

Performance

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA



LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

Do not jump to learn all of this

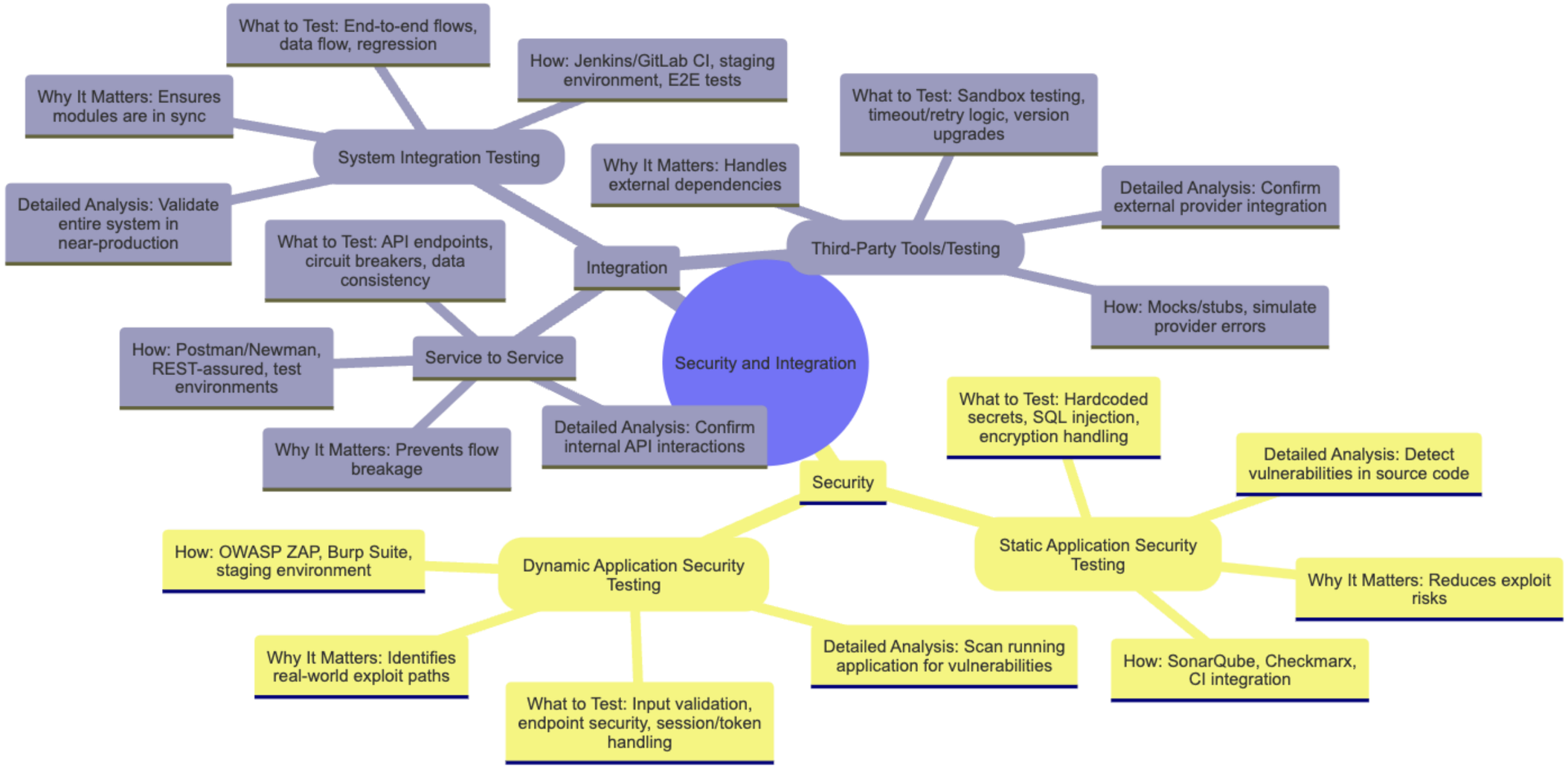
or

Get overwhelmed

Rome was not built in a day! Take your time

Security & Integration Tests

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA



LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

Do not learn a tool, learn the context behind it. Use any tool to fulfill that context!

What should be automated?

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

Prominent Automation Area's:

1) APIs form the backbone of modern applications. Failures here affect all dependent systems
   - Core business transaction endpoints (e.g., order creation, user authentication)
   - Service-to-service communication for critical paths
   - Contract validation for key consumer-provider relationships
   - Error handling and response code validation
2) End-to-end business flows directly impact revenue and customer satisfaction - Usability
   - User registration and authentication
   - Core purchase/checkout flows
   - Account management operations
   - Basic CRUD operations for main entities

What should be automated?

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

Prominent Automation Area's:

3) Security vulnerabilities can have catastrophic consequences

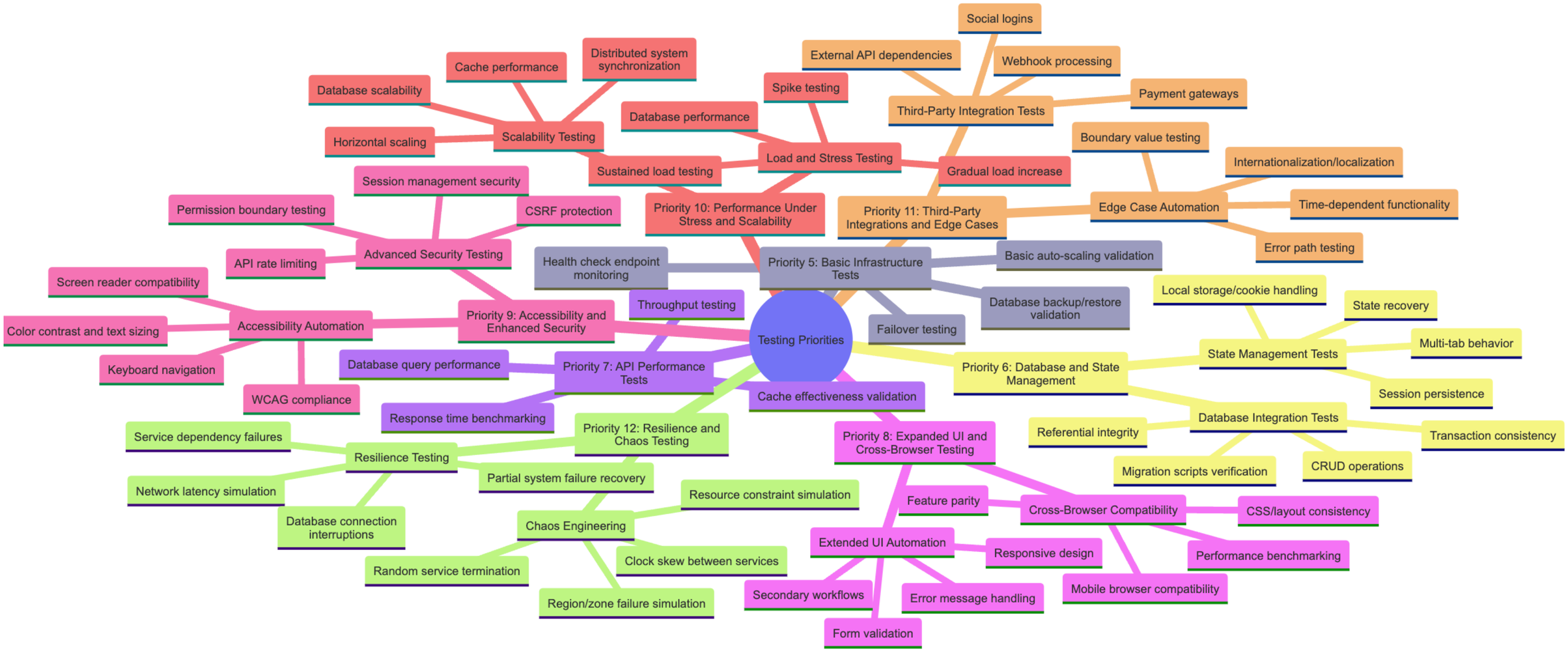
- Authentication bypass attempts
- Basic injection attacks (SQL, XSS)
- Authorization checks for protected resources
- Input validation for sensitive fields

2) UI regressions are highly visible to users

- Core page rendering across browsers
- Critical form submissions
- Navigation paths through key functionality
- Visual regression for main components

What should be automated?

LINKEDIN | YOUTUBE
JAPNEET SACHDEVA



LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

Why Critical Thinking is a Priority for QA?

Critical thinking is an analysis activity that helps to experiment and evaluate our thoughts and actions based on the information we receive or the problems we need to solve based on our understanding.
Because of this, we can decide whether this information is reliable, relevant, correct or wrong based on the situation we may encounter

Phase 1

Never make assumptions!

Phase 2

Art of asking questions

Phase 3

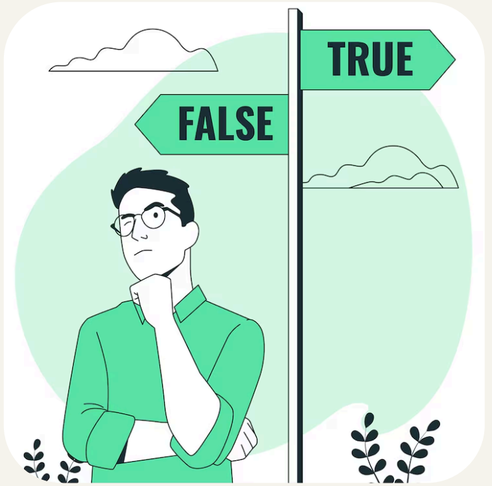
Explore and unlearn

Phase 4

Mental modelling

Phase 5

Asking & giving feedback



LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

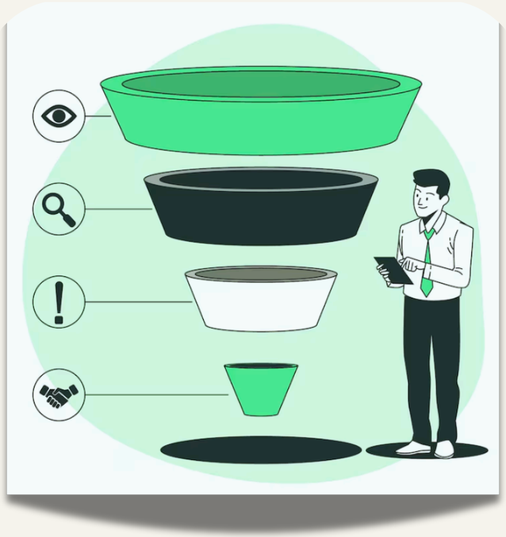
Prepare your Heuristics and Oracles

Heuristics

Heuristics are rules of thumb or guidelines testers use to explore, identify, and evaluate potential issues. They aren't strict formulas; rather, they help you think about where bugs might lurk or how to spot unexpected behavior.

Oracles

Oracles are references or mechanisms testers use to determine whether the behavior of the software is correct or not. An oracle might be a requirements document, a competing product, or even the tester's own domain knowledge



Recommended Books to Read

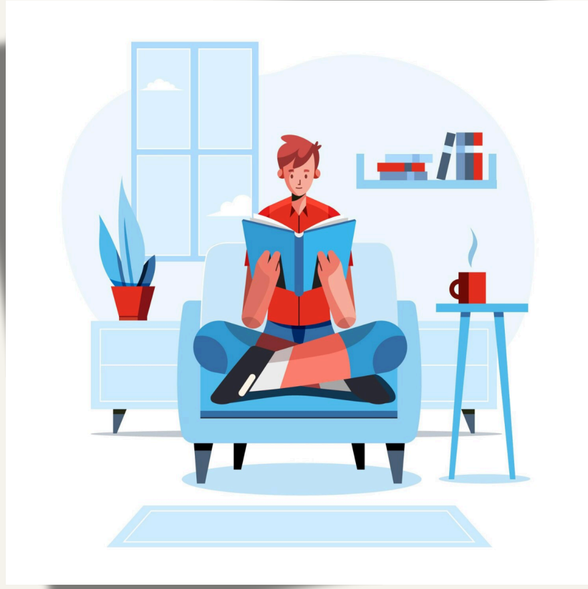
LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

Thinking fast and slow

How Google tests software

Emotional Intelligence

Atomic Habbits



LINKEDIN | YOUTUBE
JAPNEET SACHDEVA

Never stop learning
&
Questioning your thoughts/ideas

JAPNEET SACHDEVA

The End

THANK YOU FOR LISTENING

SENIOR SDET

Lavu & Adyen

CAREER COACH & CONTENT CREATOR

YouTube | LinkedIn | TopMate