📘 Chapter 2: Software Testing Methodology

- ✅ STLC
- ✅ Testing process (Scenarios, Plan, Suite, Case)
- ✅ Levels of Testing
- ✅ Functional vs. Non-functional Testing
- ✅ Testing Techniques: Black-box, White-box, Grey-box
- ✅ Test Lifecycle
- ✅ Tools: Selenium, JUnit, Postman, LoadRunner

🧠 Quick Revision – Software Testing Methodology

🎯 Testing Objective

- Ensure **software quality** by identifying and fixing defects.
- Verify that the product works **as intended** under different conditions.

📈 Principles of Software Testing

1. Testing shows presence of defects, not absence.
2. Exhaustive testing is impossible.
3. Early testing saves cost.
4. Defects cluster together.
5. Pesticide paradox – updating tests is necessary.
6. Testing is context-dependent.
7. Absence-of-errors is a fallacy.

🔁 STLC – Software Testing Life Cycle Phases

Phase	Description
1. Requirement Analysis	Understand what needs to be tested
2. Test Planning	Define scope, resources, schedule
3. Test Case Design	Write test cases & scenarios
4. Test Environment Setup	Prepare hardware/software for execution
5. Test Execution	Run test cases, log results
6. Defect Reporting	Log bugs and send to developers
7. Test Closure	Final report and lessons learned

🧪 Testing Process Components

Term	Definition
Test Scenario	High-level idea of what to test (e.g., "Verify login functionality")
Test Suite	A collection of test cases
Test Plan	Document outlining strategy, tools, schedule
Test Case	Specific steps, inputs, and expected output for validation

---

📊 Levels of Testing

Level	Purpose	Done By
Unit Testing	Test individual components	Developer
Integration Testing	Test interaction between modules	Developer/Tester
System Testing	Test the whole system	QA Team
Acceptance Testing	Verify if system meets business needs	Client/User

---

🛠️ Testing Methodologies

Type	Focus
Functional Testing	What the system does (business logic)
Non-Functional Testing	How the system performs (speed, security)

🧠 Testing Techniques

Technique	Description
Black-box	Tester has **no knowledge of internal code**; focuses on inputs/outputs.
White-box	Tester has **full code knowledge**; tests logic, loops, and conditions.
Grey-box	Tester has **partial internal knowledge**; a mix of black and white box.

🔁 Test Lifecycle Steps

- **Planning → Design → Execution → Defect Reporting → Closure**

⚙️ Automation Tools

Tool	Purpose
Selenium	Automate web applications
JUnit	Unit testing for Java
Postman	API testing
LoadRunner	Load and performance testing

📜 Summary Cheat Sheet

Topic	Example/Test Case
Test Scenario	“Verify login with correct credentials”
Test Case	Steps: 1) Enter ID, 2) Enter password, 3) Click login; Expected: Dashboard loads
Functional Test	Validate login, sign up, payments
Non-Functional Test	Measure system response time under load
Black-box Example	Tester checks UI without seeing source code
White-box Example	Developer checks loops and conditions in code
Grey-box Example	Tester uses DB queries to verify frontend behaviour

---

1. What does STLC stand for?
a) Software Test Logic Cycle
b) Software Testing Life Cycle
c) Systematic Testing Life Chart
d) Software Tool Lifecycle
✅ **Answer:** b
📝 **Explanation:** STLC stands for Software Testing Life Cycle, a sequence of activities performed during testing.

---

2. Which phase of STLC defines the scope and objective of testing?
a) Execution
b) Planning
c) Closure
d) Defect Reporting
✅ **Answer:** b

---

3. What is a Test Case?
a) A module
b) A complete system
c) A set of conditions to verify a feature
d) A requirement document
✅ **Answer:** c

---

4. Which document lists all test cases to validate the application?
a) Test Suite
b) Test Plan
c) Test Scenario
d) SRS
✅ **Answer:** a

---

5. Which type of testing checks a unit/component in isolation?
a) Integration Testing
b) System Testing
c) Acceptance Testing
d) Unit Testing
✅ **Answer:** d

---

6. Which testing ensures all components work together?
a) Unit Testing
b) Integration Testing
c) Acceptance Testing
d) Regression Testing
✅ **Answer:** b

---

7. Which level of testing validates the entire application?
a) System Testing
b) Unit Testing
c) Component Testing
d) Regression Testing
✅ **Answer:** a

---

8. Acceptance testing is primarily done by:
a) Developers
b) Testers
c) Clients/End-users
d) Designers
✅ **Answer:** c

---

9. Functional testing focuses on:
a) Performance
b) Usability
c) Security
d) Business requirements
✅ **Answer:** d

---

10. Non-functional testing includes:
a) Unit Testing
b) Load Testing
c) Regression Testing
d) Smoke Testing

✅ **Answer:** b

---

11. Which is a Black-box testing technique?
a) Statement coverage
b) Control flow testing
c) Boundary value analysis
d) Code walkthrough
✅ **Answer:** c

---

12. White-box testing focuses on:
a) User interface
b) Internal code logic
c) Functionality
d) Documentation
✅ **Answer:** b

---

13. Grey-box testing is performed by:
a) Developers only
b) Testers only
c) Testers with partial knowledge of code
d) Clients
✅ **Answer:** c

---

14. What is a Test Plan?
a) UI Layout
b) Code structure
c) Strategy and scope for testing
d) List of defects
✅ **Answer:** c

---

15. In which phase are bugs reported?
a) Design
b) Execution
c) Defect Reporting
d) Closure
✅ **Answer:** c

---

16. Selenium is primarily used for:
a) Performance Testing
b) Unit Testing
c) UI Automation Testing
d) API Testing
✅ **Answer:** c

---

17. JUnit is commonly used for:
a) Manual Testing
b) Frontend Testing
c) Unit Testing in Java
d) Load Testing
✅ **Answer:** c

---

18. Postman is best suited for:
a) UI Testing
b) API Testing

c) Database Testing
d) Load Testing
✅ **Answer:** b

---

19. LoadRunner is used for:
a) Performance and Load Testing
b) Regression Testing
c) Security Testing
d) Unit Testing
✅ **Answer:** a

---

20. The last phase of STLC is:
a) Test Planning
b) Test Execution
c) Closure
d) Design
✅ **Answer:** c

21. Which is the correct order of phases in STLC?
A. Planning → Execution → Design → Closure
B. Planning → Design → Execution → Closure
C. Design → Planning → Execution → Closure
D. Execution → Planning → Closure → Design
✅ **Answer: B**
Explanation: Correct order is: **Planning → Design → Execution → Closure**

---

22. What is the purpose of a Test Plan?
A. Execute test cases
B. Describe UI of the application
C. Define scope, approach, schedule of testing
D. Deploy the software
✅ **Answer: C**
Explanation: A test plan is a formal document that describes **scope, approach, resources, and schedule**.

---

23. What type of testing is Acceptance Testing?
A. White-box testing
B. Regression testing
C. User-level testing
D. Integration testing
✅ **Answer: C**
Explanation: Acceptance Testing is done by **end-users/clients** before go-live.

---

24. Which technique uses knowledge of code structure?
A. Black-box
B. White-box
C. Grey-box
D. Functional testing
✅ **Answer: B**
Explanation: **White-box** testing focuses on **internal logic**, requiring source code knowledge.

---

25. Which tool is best for API testing?
A. JUnit
B. Selenium
C. Postman

D. LoadRunner

✅ **Answer: C**

Explanation: **Postman** is widely used for **API testing** (GET, POST, etc.).

---

26. In which testing do we test performance under load?

A. Functional
B. Unit
C. Load Testing
D. Acceptance

✅ **Answer: C**

Explanation: **Load Testing** is a type of **Non-Functional Testing**.

---

27. What is a Test Case?

A. Input-output validation
B. Test environment setup
C. Project summary
D. Defect reporting format

✅ **Answer: A**

Explanation: A **Test Case** defines inputs, execution conditions, and expected results.

---

28. Which of the following is NOT a non-functional testing type?

A. Performance testing
B. Security testing
C. Usability testing
D. Smoke testing

✅ **Answer: D**

Explanation: **Smoke testing** is a basic functional test, not non-functional.

---

29. What is the role of LoadRunner?

A. Test user interface
B. Measure performance
C. Verify API endpoints
D. Write unit tests

✅ **Answer: B**

Explanation: **LoadRunner** is used for **performance/load testing**.

---

30. Which test level checks interaction between components?

A. Unit
B. System
C. Integration
D. Acceptance

✅ **Answer: C**

Explanation: **Integration Testing** validates interaction between modules.