1.Table of Contents

[[*TOC*]]

1.1 Definition and Key Concepts

Test automation is the practice of executing pre-written scripts and validating application behavior without human intervention. It leverages software tools to perform repetitive tasks such as UI interactions, API calls, or data setup thereby reducing manual effort and increasing consistency. Core concepts include :

- **Test Script:** A sequence of commands that exercise a system under test (SUT).
- **Test Runner:** The engine that loads, executes, and reports on test scripts.
- **Assertions:** Checks that compare actual application output against expected results.

1.2 Manual vs. Automated Testing

Aspect	Manual Testing	Automated Testing
Definition	Testing performed by human testers without using automation tools.	Testing executed using scripts or tools without manual intervention.
Speed	Slower; each test must be executed step by step manually.	Much faster; large test suites can run in minutes.
Accuracy	Prone to human error, especially for repetitive tasks.	Highly accurate and consistent across executions.
Initial Effort	Minimal setup; quick to start.	High initial effort for scripting and environment setup.
Maintenance	Low maintenance per test; but high cumulative effort over time.	Requires continuous maintenance as applications evolve.
Reusability	Limited; tests must be redone each time.	High; test scripts can be reused across releases and environments.
Test Coverage	Limited by time and human bandwidth.	Enables broad coverage including regression, data-driven, and cross-platform tests.
Cost Efficiency	Cost-effective for small projects or one-time testing.	More cost-effective for long-term and large-scale testing efforts.
Feedback Cycle	Slower; requires manual review and reporting.	Fast and integrated with CI/CD for instant feedback.

1.3 Types of Automated Tests

- **Unit Tests** validate individual functions or classes in isolation.
- **API Tests** assess endpoints, payloads, and response schemas.

- **UI Tests** automate browser or desktop interactions to simulate user workflows.
- **Performance Tests** measure application responsiveness under load.

1.4 Why Test Automation Matters for a Product-Based Company

- ## Accelerating Time-to-Market

Automated suites run in minutes what would take manual testers days, enabling faster release cycles and more frequent customer feedback loops.

- ## Improving Product Quality and Reliability

Detect defects early, reduce production incidents, and maintain a stable user experience.

- ## Enabling Continuous Integration / Continuous Delivery (CI/CD)

Automation seamlessly plugs into CI/CD pipelines (Azure DevOps), gating deployments on successful test passes and preventing regressions from reaching staging or production.

- ## Reducing Cost of Defects and Maintenance

Catching bugs in early stages is exponentially cheaper than fixing them post-release. Automated tests serve as living documentation, lowering onboarding and troubleshooting overhead.

- ## Scaling with Growing Feature Sets

As product portfolio expands, manual regression becomes unsustainable. A robust automation framework scales effortlessly, accommodating new modules with minimal incremental effort.