Introduction:

Digitization and Digital Transformation are playing the key role in Software/applications Implementation & Migration from/to On-Premise Server, Cloud (Private/Public/Hybrid/Multi-Cloud) based Infrastructures. The application quality is a key success factor for Production implementation of the application in multiple sectors/Domains not limited to BFS (Banking & Financial Services), Insurance, Logistics, Hospitality. Now a days, we are observing the lot of defect leakage w.r.t Functional & Non-Functional (Speed, Scalability, Stability, Resilience and Reliability) aspects. So, tried to highlight the key Testing Services with Its key benefits/Importance in this article. It's everyone responsibility to understand and highlight the required level of testing/services for the system/application to avoid such defect leakages in future.

Software Testing:

- It is the process of evaluating and verifying that a Software Product or application whether It's fulfils User needs/defined requirements

Key Benefits:

- Preventing Bugs
- Assure Defined User Requirements are met (Functional & Non-Functional Requirements)
- Early Identification of Architectural defects, Poor Design Decisions & Scalability Issues
- Identification of Security vulnerabilities

Following block diagram will represents the list of Testing Services w.r.t Functional & Non-Functional Testing.

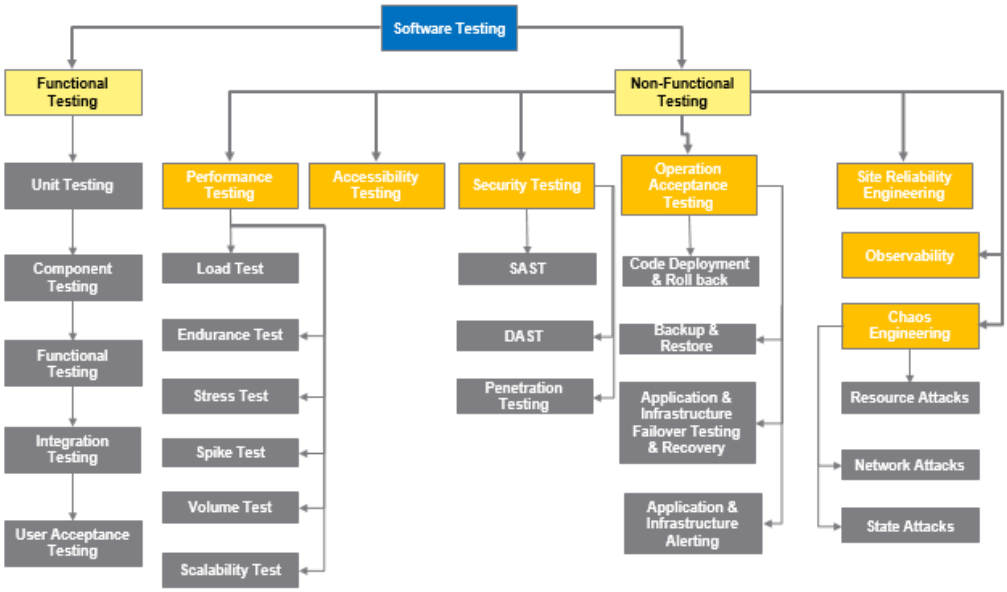

Fig 1: Overview of Testing Services

Functional Testing:

- Verifying the Software Product or application against the Functional requirements/ specifications.

Key Benefits:

- Helps to deliver the Quality Product – Defect Free Software/Application.
- All the requirements should be met.
- End user is satisfied with application

Type of Testing	Goal	Tools
Unit Testing	• A unit is the smallest testable component of an application. • Verifying that each application unit performs as expected.	• UFT - Unified Functional Testing • Selenium • Tricentis Tosca
Component Testing/ Program Testing/ Module Testing	• It is done after unit testing. Combined version of multiple Units • Each Component tested independently without integrating with other components	
Functional testing	• Verifying functions by validating business scenarios, based on functional requirements.	
Integration Testing	• Ensuring that application components and It's expected functions operate together.	

User Acceptance Testing	• Verifying whether an entire application works as expected. • Validating whether customer can use an application to complete a defined activity.	

Table 1: Functional Testing Types with Its Goal & Key Tools

Non-Functional Testing (NFT):

- Functional Testing validates the application functionality and Non-Functional Testing validates how long the application works/user is able to do the expected activity without any issues.

Performance Testing:

- Validating the Speed, Stability, Scalability, Reliability, Response Time and Resource Usage of the Software Application under a particular Concurrent User Volume/Transaction Volumes.
- It helps to avoid the application outage due to Performance Issues and avoid/minimize the impact to actual users of the application (Internal/External Users).

3S (**S**peed, **S**tability and **S**calability) are defined as follows:

- **S**peed – Determine whether the application Responds Quickly
- **S**calability – Determine the maximum user load the application can handle.
- **S**tability – Determines if the application is stable under varying loads

Key Benefits:

- Find the Performance Bottlenecks in the application. Example:
  - Application Performance Slowness while accessed by more number of users simultaneously
  - Inconsistent Application Performance while accessing using different Browsers/Operating Systems

Type of Testing	Goal	Tools
Load Test	• Evaluate the application's performance under Peak Volume of users	• Microfocus LoadRunner • Apache JMeter • NeoLoad • StresStimulus • Loadview • Grinder • Gatling • OpenSTA • Rational Performance Tester • WebLoad • LoadNinja
Endurance/ Soak Test	• Evaluates the performance of the system under load over time. • Ex: Average Load based Application usage for 8/12/24 Hours to Identify If there is any Memory Leak in the application while accessing the application for longer period of duration	
Stress Test	• To find the breaking point of the application & to determine which components fail first in the application	
Spike Test	• Evaluates the ability of the application to handle sudden Increase in the volume	
Volume Test	• Evaluate the application's ability to handle large volumes of data	
Scalability Test/ Capacity Planning Test	• Determine the application's ability to handle increasing amounts of Load and Processing	

Table 2: Functional Testing Types with Its Goal & Key Tools

Accessibility Testing:

- Ensure that the application accessible to those with disabilities, such as Vision impairment, Hearing disabilities, and other physical or Cognitive conditions.
- People with disabilities use assistive technology, which helps them in operating an application/ Software Product.
  - **Screen Reader Software** – Used to read out the text that is displayed on the screen.
  - **Speech Recognition Software** – Converts the spoken word to text, which serves as input to the computer.
  - **Screen Magnification Software**– Used to enlarge the monitor and make reading easy for vision-impaired users

Type of Disability	Description
Vision Disability	• Complete Blindness or Colour Blindness or Poor Vision • Visual problems like Visual Strobe and Flashing effect problems
Physical Disability	• Not able to use the mouse or keyboard with one hand. • Poor motor skills like hand movements and muscle slowness
Cognitive disability	• Learning Difficulties or Poor Memory or not able to understand more complex scenarios
Literacy Disability	• Reading Problems
Hearing Disability	• Auditory problems like deafness and hearing impairments • Cannot able to hear or not able to hear clearly

Table 3: Type of Disability and It's Description

Accessibility Tools	Goal/Objective
JAWS	• **J**ob **A**ccess **W**ith **S**peech • Screen Reader for Microsoft Windows • Assist users who are blind or low-vision to use a Windows computer.
DNS	• **D**ragon **N**aturally **S**peaking - Speech Recognition • Use Voice to Interact with applications/to do expected activity in the application as like Normal User
ZoomText	• Fully integrated Magnification and Reading tool used for Low-Vision Users. • Magnifier/Reader enlarges and enhances the application Page size
CCA	• Colour Contrast Checker tool helps to easily determine the contrast ratio of two colours
WAT	• Adds a toolbar to Internet Explorer to aid manual inspection of accessibility related elements on web pages
WAVE	• **W**eb **A**ccessibility **E**valuation **T**ool (WAVE) • Display the page with tags to show what elements met & didn't meet the Accessibility Standards
Dark Reader	• This eye-care extension enables night mode by creating dark themes for websites on the fly. • Dark Reader inverts Bright colours, making them high contrast and easy to read at night. • Easy to adjust the brightness, contrast, sepia filter, dark mode, font settings
Axe	• Web and Mobile Development teams find accessibility errors while coding • Helps to Reduce the time and cost of manual accessibility testing down the line
NVDA	• Free Screen Reader for Windows that is the primary screen reader for 40% users
Read & Write	• Literacy support tool that offers help for everyday tasks • It's line reader application Dyslexia

Table 4: Accessibility Testing Tools with Its Objective

Security Testing:

- To identify the threats in the system and measure its potential vulnerabilities, so the threats can be encountered and the system does not stop functioning or cannot be broken.

Key Benefits:

- To identify all possible loopholes and weaknesses of the application which might result in a Loss of information, Revenue, Repute at the hands of the Internal or External Users/Public.

Type of Testing	Goal	Tools
SAST	• Static Application Security Testing • To Identify Vulnerabilities during early development stages – In Code Level	• Fortify • AppScan • SonarQube • Veracode • Checkmarx
DAST	• To Identify Vulnerabilities in Web Applications	• AppScan • Checkmarx • OWASPZAP
Penetration Testing	• Use DAST for application penetration testing	

Table 5: Security Testing Types with Its Goal & Key Tools

Operational Acceptance Testing (OAT)/Service Readiness Testing:

- Conduct Operational Readiness (Pre-Release) of an application as part of a quality management system

Key Benefits/areas of Improvements:

- Resiliency
- Recovering Ability
- Manageability and Supportability
- Integrity

Type of Testing	Description
Code Deployment – Installation & Roll Out Testing	• Validate the defined procedures to Install/deploy the code to assure that application in working state after the Installation • Revoke the code with defined procedures and assure that application rolled back to expected version and It's working fine after the code Roll out activity.
Backup & Restore	• Validate the defined procedures to take the copy of Data/Database and restore the same when there is a business need.
Application & Infrastructure Fail Over Testing	• Validating an Application/Infrastructure capacity during a server failure to allocate sufficient resources toward recovery. • Recovery testing is the activity of testing how Soon an application is able to recover from crashes, hardware failures and other similar problems.
Application & Infrastructure Alerting	• Notify Required users when matches a predefined Condition/Threshold.

Table 6: OAT Testing Types with Its Goal & Key Tools

Chaos Engineering:

- Proactive way to Identify weakness in the application/System by conducting the defined attacks/ involves running thoughtful, planned experiments that help us to understand how systems behave in the face of failure.

Key Benefits:

- To Evaluate and Increase System Resilience and Reliability
- Early Detection of Failures with Its Resolution in Pro-Active approach
- Identify the weak points in the systems
- Understand how the systems respond to pressure as real-world events in real-time & Increases End User and Stakeholder Satisfaction
- Prepare and educate the engineering team for actual failures, Improves Application Performance Monitoring to Improves Incident Response Time
- Create a cost-efficient failover plan that saves money in the event of a failure
- Validate the resiliency of their business data in the event of a disastrous failure.

Type of Attacks	Attack Name	Goal	Tool
Resource attacks: Test against sudden changes in consumption of computing resources	CPU	Generates high load for one or more CPU cores	• Gremlin • Chaos Monkey • ChaosBlade • Chaos Mesh • Litmus • Chaos Toolkit • PowerfulSeal • ToxiProxy • Istio • KubeDoom • AWS Fault Injection Simulator
	Memory	Allocates a specific amount of RAM.	
	I/O	Puts read/write pressure on I/O devices such as hard disks.	
	Disk	Writes files to disk to fill it to a specific percentage	
Network attacks: Test against unreliable network conditions	Blackhole	Drops all matching network traffic.	
	Latency	Injects latency into all matching egress network traffic.	
	Packet Loss	Induces packet loss into all matching egress network traffic.	
	DNS	Blocks access to DNS servers.	
State attacks: Test against unexpected changes in environment such as power outages, node failures, clock drift, or application crashes	Shutdown	Performs a shutdown (and an optional reboot) on the host operating system to test how your system behaves when losing one or more cluster machines.	
	Time Travel	Changes the host's system time, which can be used to simulate adjusting to daylight saving time and other time-related events.	
	Process Kill	Kills the specified process, which can be used to simulate application or dependency crashes.	

Table 7: Chaos Engineering Attacks with Its Goal & Key Tools

Observability:

- Tooling or a technical solution that allows teams to actively debug their system.

3 Pillars	Goal	Tools
Logs	• Files that record events, warnings and errors as they occur within a software environment. • Most logs include contextual information, such as the time an event occurred and which user or endpoint was associated with it.	• NewRelic • Dynatrace • LogicMonitor • AppDynamics • AWS CloudWatch • Splunk • DataDog • ElasticObservability • Grafana
Monitoring	• Quantifiable measurements that reflect the health and performance of applications or infrastructure.	
Traces	• Data that tracks an application request as it flows through the various parts of an application. • Shows application component to process the request and pass the result to the next component. • Helps to identify which parts of the application triggers an error.	

Table 8: 3 Pillars of Observability

Site Reliability Engineering:

- Set of principles and practices that incorporates aspects of software engineering and applies them to IT infrastructure and operations.
- The main objectives are to create highly reliable and scalable software systems.

Key Benefits:

- Understand and Define the Service Level Objectives and Indicators (SLO and SLI)
- Proactive Monitoring of Application & Infrastructure
- Respond to incidents
- Collaborate to resolve issues
- Automate Cloud Deployments.

Conclusion:

Hope this article provided the high-level details about Software Testing Services Which Inclusive of Functional & Non-Functional Testing Services with Its key benefits, Types of Testing, Goal and associated key Tools. Each and Every Testing Service consist of Its own challenges, Testing Life Cycle Activities/methodologies and Tools Cost. Based on the Business needs, we can select the required services to assure the quality of the system/applications to avoid the defect leakage into Production Environment. It will help to get the better User Experience & End User Satisfaction.

About The Author:

Utilized the various tools for NFT Services Implementation and those tools are highlighted in Green Colour in this document.

Venkatesan A is holding 15+ Years of IT Experience with primary skills of Performance Testing/Engineering. Due to business needs uplifted Technical Knowledge with hands on experience for following Testing Services – *Accessibility Testing, Security Testing, Operational Acceptance Testing, Chaos Engineering, Observability and Site Reliability Engineering* in following domains – Banking and Financial Services & Regulatory, Logistics. Played following key roles to meet the organizational/Business goals - *NFT Business Partner, NFT Delivery Manager, NFT Transition Lead, NFT Quality Assurance Manager, Performance Test Manager*. He is a quick learner and self-interested to learn & adapt to meet the business goals. His technical certifications as follows **Performance Testing:** 1) Microfocus (Performance Testing Technical Certification) 2) HP AIS (LoadRunner) 3) IBM Rational Performance Tester Certified. **Accessibility Testing**: 1) Text help Level 1 Certified User – Read & Write Windows 2020 2) OAST certified: a) Section 508 Standards for Web, b) Procuring Section 508 Conformant Products, c) Trusted Tester Training for Web on Windows. **Chaos Engineering:** 1) Gremlin Certified Chaos Practitioner 2) Certified Harness Chaos Practitioner. **Observability:** Olly Certified Practical Observability Engineer. Implemented above listed NFT Services based on the Project Implementation Methodology (Waterfall/Agile/SAFe Agile) with hands on experience in On-premise/Cloud based applications (AWS, MicrosoftAzure, Pivotal Cloud Foundry, GCP) with/without Third Party System. End to End hands on NFT experience for Legacy Web Applications - Online & Mobile applications in Isolation/Integration with Third Party Applications (w.r.t Migration into - AWS/Microsoft Azure Cloud, Mainframe, AWS/Pivotal Cloud based Single Page Application, Salesforce – Classic & Lightning, Livelink & SharePoint platform) and also for following products - SAP, Siebel CRM, Microsoft Dynamics CRM, PEGA, Oracle, Citrix, Kafka based Message Processing System, Batch Jobs (ETL – Data Warehouse, Data Lake, Database) using ODI & Talend Tools & Business Intelligence Reporting Tools (IBM Cognos, TIBCO Spotfire, Tableau) based applications. Holding strong thought leadership, strategic thinking and effective communication in driving and uplifting Quality through all stages of the SDLC and within the Test practice. Extensive experience in managing a multi-vendor engagement. Well versed with Performance Testing COE Setup, Tool Selection & NFT Life Cycle Activities associated Templates Definition & Customization based on the Business needs. My Other key Whitepapers are

- Performance Testing on SharePoint, Salesforce, Livelink, AWS, MDCRM
- Performance Testing on ETL & Reporting applications.

For more details connect with author via LinkedIn.