National Software Testing Guideline

April 2026

1.0.	National Software Testing Guideline	**Title**
2.0.	The National Software Testing Guideline is a vital framework designed to enhance the quality and reliability of software developed, modified, integrated and deployed in Nigeria. Recognising the increasing reliance on technology in various sectors, this guideline establishes minimum testing standards to ensure that software meets essential quality benchmarks, complies with relevant regulations, and is fit for purpose. Through collaboration with key partners, this initiative underscores the commitment to fostering a professional and competent software testing community in Nigeria, ensuring that only thoroughly tested and compliant software is deployed for operational use.	**Explanatory Note**
2.0.	This Guideline establishes the minimum requirements for software testing in all software developed, modified, integrated with existing software, or deployed for use in Nigeria. The aim is to ensure that the software meets the necessary quality standards and complies with all relevant regulations. The regulation also seeks to ensure reliability, security, and performance of software that is developed, modified, or deployed in Nigeria, safeguarding operations and catalyzing the growth of a robust local software testing market.	**Objectives**
3.0.	This Guideline is issued under the authority provided to NITDA by the National Information Technology Development Act of 2007, which empowers NITDA to regulate the development and deployment of IT systems, including software, in Nigeria.	**Authority**
4.0.	This Guideline applies to all software developed, modified, integrated, or deployed for use in Nigeria. The guideline must be followed by any party involved in the development, modification, integration or deployment of software for use in Nigeria.	**Scope**
5.0.	This Guideline shall be read together with the National Software Development Guideline and the Software Testing Organisations Licensing Framework, which collectively	**Framework Integration Clause**

DRAFT

constitute the National Software Quality Assurance Framework for Nigeria.

6.0. The title for these guidelines is National Software Testing Guideline and shall come into effect when issued by NITDA. **Commencement**

7.0. For the purposes of this Guideline: **Definitions**

- "Software Testing" means the process of evaluating and verifying that a software application or system meets specified requirements and functions correctly.
- "NITDA" means the National Information Technology Development Agency.
- "Testing Organisation" means any entity licensed to conduct software testing in accordance with this Guideline.
- "Third-Party Testing Provider" means an independent entity licensed by NITDA to assess software systems.
- "Certification" means formal confirmation that software has passed required testing and is fit for deployment.
- "Compliance" means adherence to this Guideline and applicable laws.

8.0. **Responsibilities of Stakeholders**

NITDA

· Establish and enforce software testing standards.
· License and oversee testing organisations.
· Monitor compliance and update requirements.

Testing Organisations

· Conduct testing in accordance with this Guideline.
· Maintain independence and objectivity in testing.
· Submit reports and certification outcomes to NITDA.

Software Developers and MDAs

· Ensure software is submitted for testing prior to deployment.
· Address defects identified during testing.
· Ensure compliance with all testing requirements.

9.0. The level and depth of testing shall be determined based on the risk profile of the software system. **Risk-Based Application of Testing Guidelines**

Risk assessment shall consider:
· Data sensitivity
· Exposure to public or external systems

DRAFT

· Criticality to operations
· Potential impact of system failure
Software identified as high-risk shall undergo enhanced testing, including advanced security, performance, and reliability testing.
Notwithstanding the above, all software shall meet the minimum testing requirements defined in this Guideline.

10.0. Independent entities, licensed by NITDA, are responsible for assessing software according to the requirements in this Guideline. **Third-Party Testing Providers**

11.0. No software system shall be deployed for operational use unless it has undergone testing by a licensed testing organisation and has been certified as compliant with this Guideline. **Pre-Deployment Requirement**

12.0. All software developed, integrated or modified for use in Nigeria must undergo thorough testing by a licensed third-party testing body before deployment. Testing must comply with the provisions of this guideline. **Testing Requirements**

This guideline outlines the mandatory software quality attributes to be tested in accordance with the agreed specifications and all applicable local and international regulatory requirements. All software developed or modified for use in Nigeria must undergo comprehensive testing based on the following functional and non-functional attributes.

13.0. The following test processes must be thoroughly documented and submitted to NITDA by the testing entities upon the completion of each testing engagement. All testing entities must also engage in test process improvement activities as defined by NITDA in the software testers licensing regulation document. **Testing Process**

- **Requirement Analysis:** Clear understanding of the software's requirements, both functional and non-functional, to determine what needs to be tested, including test objectives and criteria.
- **Test Planning:** All software testing projects must begin with a comprehensive test plan outlining the scope,

DRAFT

objectives, testing tools, and schedule. The plan must align with the agreed specifications and regulatory requirements.

- **Test Design:** Work items that will be used to validate the software against the requirements are defined in this stage. Work items such as Test Cases and Scripts, Test Data, Traceability Matrix etc.
- **Test Environment Setup:** Prepare the hardware, software, network, and tools needed to execute tests. The test environment shall mirror the production environment for accurate testing.
- **Test Execution:** Testing must be executed according to established methodologies and industry best practices as defined by NITDA. Testing organisations must ensure all relevant test ware is executed to validate the software's functionality and performance.
- **Defect Management:** All software developed must go through this critical process in software testing that involves identifying, documenting, prioritising, tracking, and resolving defects (or bugs) in software. Effective defect management ensures that software is released with minimal defects, leading to improved quality, user satisfaction, and smoother deployment.
- **Test Reporting and Closure:** Detail test results must be documented in the test report. The report should include test logs, screenshots, defect analysis and management etc. Testing is closed when all testing activities are done, and the overall test coverage and results are evaluated. Test closure shall contain test closure reports, summary of testing metrics, and lessons learned.
- **Acceptance Criteria:** For software to be approved for use, it must meet predefined acceptance criteria set by NITDA. These criteria include passing functional, security, performance, usability, and compatibility tests. The software must be free of any critical or high-risk defects.
- **Documentation:** The above test processes must be thoroughly documented and submitted to NITDA by the testing body upon the completion of each testing engagement.

DRAFT

14.0. The following test must be conducted to ensure that all the defined attributes of the software perform as required and meet user expectations:

Testing Attributes

Functional Testing

The software must be evaluated to ensure that it meets the required functionality as specified in user requirements and design documents. The following aspects of functional suitability must be tested:

- **Completeness**: Testing must verify that all necessary functionalities have been implemented, and no critical features are missing.
- **Correctness:** Tests must ensure that all implemented functionalities produce accurate and expected results according to specified requirements.
- **Appropriateness**: Testing must verify that the software functions are appropriate for user needs and the intended purpose.

Non-Functional Testing

The non-functional aspects of the software must be evaluated to ensure that the system performs effectively under various conditions, is maintainable, secure, and reliable over time. The following non-functional testing must be executed:

- **Performance Testing:** Evaluation of the software's performance under expected workloads, including stress, load, and scalability tests. Tests should include:
  - **Load Testing:** This test measures how the system behaves under an expected user load. It checks whether the software can handle the anticipated number of users and transactions under normal conditions.
  - **Stress Testing:** Stress testing pushes the system beyond its normal load conditions to identify its breaking point or capacity limits.
  - **Scalability Testing:** This testing measures the software's ability to scale up (handle increasing load) or scale down (operate efficiently with reduced load) without affecting performance.

DRAFT

    - **Spike Testing:** Spike testing evaluates the system's reaction to sudden and extreme increases in user load or transaction volume.
    - **Endurance (Soak) Testing:** This testing assesses how the system handles a large amount of data in terms of input/output and database performance.
- **Security Testing:** Assessment of the software's resilience against cyber threats, data breaches, and security vulnerabilities. Key areas to test include:
    - **Confidentiality**: Ensure that sensitive information is protected from unauthorised access.
    - **Integrity**: Validate that data is protected from unauthorised modification.
    - **Non-repudiation**: Test mechanisms to ensure that actions or events cannot be denied afterward.
    - **Accountability**: Ensure that user actions can be traced and logged.
    - **Authenticity**: Verify that users and systems can be identified and authenticated.
    - **Test that the system Conforms to OWASP Top 10 Proactive Controls.**
    - **Test that the system Conforms to OWASP Application Security Verification Standard (ASVS).**
- **Usability Testing:** Usability tests must ensure that the software is easy to learn, operate, and interact with. Testing should focus on:
    - **Learnability**: Evaluate how easy it is for users to learn how to use the software.
    - **Operability**: Verify that users can efficiently operate the software without errors.
    - **User Error Protection**: Test mechanisms in place to prevent and mitigate user errors.
    - **User Interface Aesthetics**: Assess the visual appeal and layout of the software interface.
- **Compatibility Testing:** Testing must ensure that the software operates effectively within various

DRAFT

environments and can coexist or interoperate with other systems:
    - **Co-existence**: The software must be tested for its ability to work alongside other systems without causing conflicts.
    - **Interoperability**: The ability of the software to exchange and use information with other systems must be validated.
    - Conformance to the National regulations governing secure and efficient data exchange.
- **Reliability Testing**: The software must be tested to ensure that it consistently performs its intended functions without failure:
    - **Maturity**: Evaluate the software's ability to function without defects under normal operation.
    - **Availability**: Testing must assess the availability of the system to ensure it can be used when required.
    - **Fault Tolerance**: The software must be tested for its ability to continue operating in the event of failure.
    - **Recoverability**: Tests should validate the system’s ability to recover from failures, including data recovery and the restoration of operations.
- **Maintainability**
  Testing must evaluate the ease with which the software can be modified, updated, and maintained:
    - **Modularity**: Test the software for its separation into independent components, which allows for easier maintenance.
    - **Reusability**: Evaluate the potential for code and components to be reused in other systems or versions.
    - **Analysability**: Assess how easily the software can be diagnosed with defects or issues.
    - **Modifiability**: Test the software for ease of making changes and updates.

DRAFT

    - **Testability**: Evaluate the software's capacity to be easily tested through automated or manual testing tools.
- **Portability**
  Testing must assess the software's ability to operate in different environments without requiring extensive modification:
    - **Adaptability**: Verify that the software can adapt to different hardware, software platforms, or operating environments.
    - **Installability**: Evaluate the ease with which the software can be installed and configured in its target environment.
    - **Replaceability**: Test the ease of replacing the software with an alternative system.

15.0. **Additional Testing Requirements**

In addition to functional and non-functional testing, all software shall be evaluated to ensure compliance with applicable regulatory requirements, industry standards, and any specific directives issued by the regulatory authority of the sector in which the software will operate.

Software systems shall comply with additional testing requirements based on their sectoral application, operational context, risk profile, and technological characteristics. This is essential to ensure that software operates within legal and regulatory boundaries and meets the quality, security, and performance expectations of its intended environment.

- Such additional testing requirements may include, but are not limited to:
  Sector-specific compliance testing in accordance with applicable regulatory frameworks
- Advanced security testing, including vulnerability assessments and penetration testing
- Integration and interoperability validation with existing systems and national platforms
- Testing requirements for emerging technologies, including but not limited to artificial intelligence, distributed systems, and cloud-native architectures

DRAFT

Where multiple regulatory or policy frameworks apply, software systems shall ensure alignment and compliance across all applicable instruments.
NITDA may, from time to time, issue supplementary guidelines or directives to address additional testing requirements arising from sector developments, technological advancements, or national priorities.

16.0. NITDA will set and update criteria for licensing of testing entities according to industry trends and best practices in the NITDA software testers licensing regulation document. **Licensing of Testing Entities**

17.0. **Certification Process** **Software Certification**
After successful testing, certification shall be issued by a licensed testing organisation and recognised by NITDA as evidence of compliance with this Guideline.

Renewal
Software certification must be renewed, when significant changes or modifications are made to the software.

18.0. NITDA shall maintain a register of licensed testing organisations and certified software systems.
The register shall include relevant details such as certification status, validity period, and compliance standing. **Public Register of Testing Entities and Certified Software**

19.0. NITDA will take all necessary enforcement actions to ensure compliance with this Guideline. Non -compliance with this Guideline is a violation of the NITDA Act 2007 and other relevant applicable laws. **Compliance**

Compliance with this Guideline shall form part of the requirements for IT Project Clearance for all government software systems

20.0. NITDA shall monitor certified software systems to ensure continued compliance with testing and operational standards. Software may be subject to reassessment where significant changes occur or where risks are identified. **Monitoring and Continuous Compliance**

DRAFT

This Instrument was signed this _____ of April 2026

_____

Kashifu Inuwa Abdullahi *CCIE*

Director-General/CEO

National Information Technology Development Agency

DRAFT