**Software Requirements Specification (SRS) Document**

**Project Name:** Automated Software Testing System (ASTS)

## **1. Introduction**

### ****1.1 Purpose****

Automated Software Testing System (ASTS) provides an economical, efficient, and scalable automated software testing system with negligible human intervention. ASTS test cases are mechanized, bugs are trapped, and they are reported for quality improvement of the software, automating the procedure, and lowering time-to-market. ASTS is tailored for successful CI/CD of agile development teams.

### ****1.2 Scope****

ASTS is an API-based, web-based application for project managers, developers, and testers. It provides:

* Automated testing for higher efficiency and accuracy.
* Flaw detection and tracking for enhanced reliability in software.
* Secure version-controlled test case management.
* Luxurious dashboards for analytics and reporting.
* Support for several testing paradigms (unit, regression, integration, performance, security).
* AI-based generation and optimization of test cases (future direction).

### ****1.3 Definitions, Acronyms, and Abbreviations****

* **ASTS** – Automated Software Testing System
* **CI/CD** – Continuous Integration/Continuous Deployment
* **RBAC** – Role-Based Access Control
* **SRS** – Software Requirements Specification

### ****1.4 References****

* IEEE 830-1998 (Software Requirements Specification Standard)
* IEEE 829-2008 (Software Test Documentation)
* ISTQB Software Testing Standards
* ISO/IEC 25010:2011 (Software Quality Model)
* GDPR Compliance Standards

### ****1.5 Overview****

This specification gives the non-functional and functional requirements of ASTS, presenting an orderly guide for its maintenance, deployment, and development.

## **2. Functional Requirements**

### ****2.1 User Authentication and Authorization****

* Username and password-based secure login.
* Third-party integration authentication with OAuth 2.0.
* Admin, Tester, Developer, Guest-based implementation of RBAC.

### ****2.2 Test Case Management****

* Ticket generation for create, update, delete, and group of test cases.
* Support for parameterized test cases.
* Priority and application module-based categorization.

### ****2.3 Test Execution****

* Fetching executions on schedules/triggers.
* Parallel support for execution in order to better perform.
* Integration support with CI/CD pipelines so that automatic tests run on every deployment.

### ****2.4 Defect Logging & Tracking****

* User reporting, tracking, and updating of defect status (Open, In Progress, Resolved, Closed) capabilities.
* Attach screenshot/logs option for debugging capabilities.

### ****2.5 Test Reporting & Logging****

* Execution reports with pass/fail outcomes and failure analysis.
* Reports exportable in various formats (PDF, CSV, HTML).
* Trend analysis visual analytics dashboard.

### ****2.6 Integration with Development Tools****

* Version control support (Git, GitHub).
* Built-in integration with Jira, Trello for issue tracking.
* API support for external integrations.

### ****2.7 Notification System****

* In-app and email notifications for test outcomes.
* Alerts for critical failures.

### ****2.8 AI-Based Test Automation (Future Scope)****

* Test case generation based on history of execution using AI.
* Defect prediction and automated optimization of test cases.

## **3. Non-Functional Requirements**

### ****3.1 Performance Requirements****

* Unit tests should run in 5 seconds.
* Regression tests should run within 30 minutes.
* Support execution of 100+ tests in parallel.

### ****3.2 Usability Requirements****

* User-friendly and intuitive UI navigation.
* Cross-platform availability (web, mobile support).

### ****3.3 Reliability & Availability****

* Uptime of the system 99.9% to have uniform operation.
* Automated failover features for testing execution.

### ****3.4 Security Requirements****

* AES-256 encryption to store data.
* TLS 1.2 encryption to send data.
* Role-based access control (RBAC) for prevention of unauthorized access.
* Secure APIs to maintain data integrity.

### ****3.5 Maintainability & Scalability****

* Modular architecture for smooth updates and extensions.
* Horizontal scaling to handle heavy test loads.
* Low-maintenance design, which means minimal intervention by developers.

### ****3.6 Compatibility Requirements****

* Supports Windows, macOS, and Linux.
* Supports Chrome, Firefox, Edge, Safari.

### ****3.7 Compliance Requirements****

* ISTQB, IEEE 829, GDPR compliance.
* Industry-standard data policies to maintain user privacy.

## **4. Justification for Non-Functional Features**

### ****4.1 Maintainability****

* The modular design makes it simple to modify without system-wide modification.
* Independent scalability of various modules is ensured using microservices architecture.

### ****4.2 Reliability****

* Failover mechanisms in an automated fashion ensure system robustness.
* Cloud infrastructure boosts availability and disaster recovery.

### ****4.3 Security****

* Secure storage of test case data is ensured using AES-256 encryption.
* TLS 1.2 encryption keeps data transmission from being intercepted.
* Role-based access control provides correct user privileges, ensuring confidentiality and integrity of data.

### ****4.4 Reusability****

* Object-oriented design enables reusability of test components by multiple projects.
* Parameterized test cases enable variation without repeating test logic.

## **5. Appendix**

* GitHub Repository: To be updated.
* Notion Document Link: To be updated.
* Project Members & Contact Details: To be added.

## **6. Conclusion**

The Automated Software Testing System (ASTS) is conceived to transform software testing by bringing automation, defect tracking, and AI-powered augmentations together. By providing strong security, compliance, and scalability, ASTS streamlines software quality assurance processes and boosts team productivity.

This paper will be uploaded to Notion and GitHub for formal submission.

**Submission Date:** Friday, 28th February 2025, 11:00 PM.  
**Instructor Email:** [usman.younis@itu.edu.pk](mailto:usman.younis@itu.edu.pk)  
**TA Email:** [ayeshaaman.ta@itu.edu.pk](mailto:ayeshaaman.ta@itu.edu.pk)