UDC 004.415.2:004.415.53

Daria Sokol

DESIGN AND DEVELOPMENT OF A WEB-BASED SYSTEM FOR AUTOMATED SOFTWARE TESTING QUALITY EVALUATION

National Technical University "Kharkiv Polytechnic Institute"

Анотація

У роботі розглянуто процеси проектування і розробки веборієнтованої системи для автоматизованого оцінювання якості програмних тестів. Система забезпечує повний цикл обробки даних: від збору й аналізу тестових артефактів до формування інтегральної оцінки, візуалізації та експертного аналізу результатів. Передбачено інтеграцію із зовнішніми системами CI/CD, збереження історії оцінювання, а також рольову модель доступу до функціональних можливостей системи. Запропонований програмний продукт сприяє підвищенню об'єктивності оцінювання та ефективності процесів забезпечення якості програмного забезпечення.

Ключові слова: оцінювання якості тестів, тестування програмного забезпечення, веборієнтована система, FastAPI, React, PostgreSQL.

Abstract

This paper focuses on the design and development of a web-based system for automated software testing quality evaluation. The system supports the complete data processing cycle, from the collection and analysis of test artifacts to the generation of an integrated assessment, visualization, and expert analysis of the results. The developed solution provides integration with external CI/CD systems, maintains the history of evaluations, and implements a role-based access model to the system's functionality. The proposed software product contributes to improving the objectivity of evaluation and the efficiency of software quality assurance processes.

Keywords: test quality evaluation, software testing, web-based system, FastAPI, React, PostgreSQL.

Introduction

The growing complexity of software systems and the widespread adoption of automated testing have significantly increased the volume of test-related data generated during software development. Quality assurance specialists must analyse numerous metrics, execution results, and testing artifacts to assess the effectiveness and reliability of software tests. The effective processing and interpretation of large volumes of heterogeneous data have become important challenges in software quality assurance.

In practice, quality assessment is often performed using separate tools that provide only partial information about testing activities. As a result, analysts must manually collect data from multiple sources, process metrics, and prepare reports [1]. Such an approach increases labour costs, complicates decision-making, and may reduce the objectivity of evaluation results.

The development of a web-based system creates new opportunities for automating quality assessment procedures. Modern web technologies enable the integration of multiple information sources, centralized data management, analytical processing, and real-time visualization within a single software platform.

The purpose of this work is to develop a web-based system that supports automated software test quality evaluation, centralized management of quality metrics, analytical reporting, and integration with software development environments. This paper extends previous research on integrated software test quality assessment [2] by providing a detailed description of the architecture, functional structure, and implementation principles of the proposed system.

Main Part

The proposed system is designed according to a client-server architecture that ensures scalability, maintainability, and flexibility of further development. The system integrates data collection, processing, storage, visualization, and reporting functions within a unified information environment.

The functional structure of the system consists of several interconnected modules, as shown in Fig. 1. The user interface module serves as the primary means of interaction between users and the system. It provides access to system functionality through a unified web interface, allowing users to import data, review evalua-

tion results, analyse quality indicators, generate reports, and perform other tasks according to their assigned roles. The module also displays progress statuses, validation messages, and system notifications, making user interaction with the system more convenient and transparent.

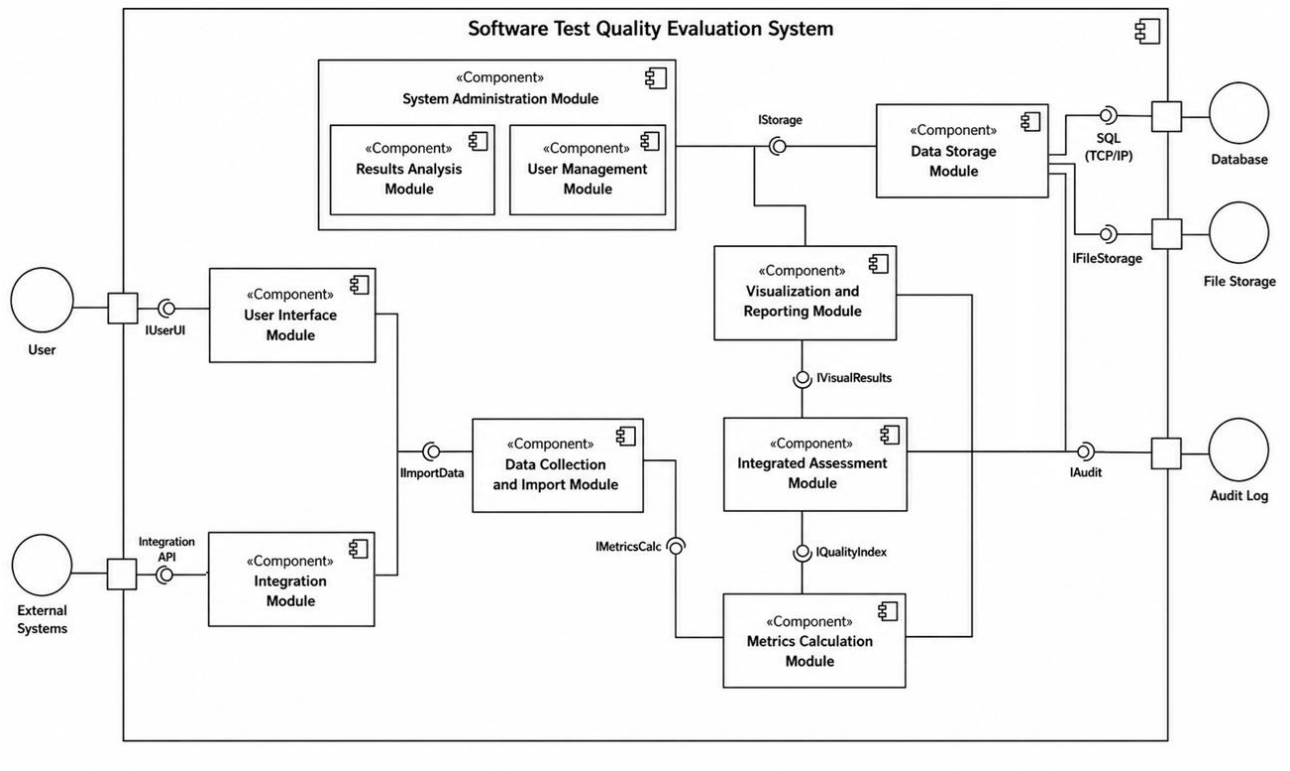

Fig. 1. UML component diagram of the web-based software test quality evaluation system

The integration module enables communication with external software development and testing tools. It supports data exchange through API interfaces and allows the system to interact with CI/CD platforms and other external services. This module provides the basis for automated import of testing results and integration of the proposed system into existing software development infrastructures. Automated data acquisition reduces the amount of manual work and improves the reliability of collected information.

The data collection and import module serves as the entry point of the system and is responsible for acquiring input data required for further analysis. The module supports both manual uploading of test artifacts and automated import from external systems via the integration module. In addition, it performs data validation, format verification, consistency checks, and transformation of imported data into a unified internal representation. Such preprocessing ensures the reliability and correctness of subsequent evaluation.

The metrics calculation module is responsible for processing received data and calculating quantitative indicators used to evaluate software test quality. The module computes a set of metrics based on test execution results, code coverage, defect statistics, automation indicators, stability characteristics, and other quality-related parameters [3, 4]. The calculated metrics form the basis for further assessment and analysis.

The integrated assessment module represents the core component of the system. Based on the calculated metrics, user-defined normalization rules, and configurable weighting coefficients, the module generates an integrated test quality index. The resulting assessment provides a generalized representation of test quality and enables objective comparison of different test suites, project versions, or testing iterations.

An important component of the system is the visualization and reporting module. It provides tools for visualizing quality indicators and assessment results. Interactive dashboards, charts, and statistical summaries enable users to monitor quality trends, identify weaknesses in testing processes, and support managerial decision-making. This module also supports the generation and export of analytical reports in formats such as PDF and XLSX that contain detailed information about quality indicators, evaluation results, and identified trends. Such reports can be used by software quality engineers, project managers, and other stakeholders involved in software development projects.

The system administration module provides facilities for configuring system settings, managing evaluation parameters, and monitoring system activity. Audit mechanisms ensure transparency and traceability of operations performed within the system. As part of the administration subsystem, the user management module is responsible for user account administration, including creating, modifying, and deleting users, as well as assigning roles. The results analysis module supports expert evaluation of assessment outcomes. It provides analysts with detailed information about calculated metrics and integrated quality indicators, allowing them to review the obtained results, assign assessment statuses such as Passed or Failed, and leave comments or recommendations for testers. This approach combines automated assessment with expert judgment and improves the reliability of decision-making.

Persistent storage of system data is provided by the data storage module. Evaluation data, user information, and configuration parameters are stored in a relational database, while uploaded test artifacts and generated reports are maintained in a dedicated file storage. The module also records significant user actions and system events in an audit log, supporting traceability and historical analysis.

The web-based implementation model provides centralized access to evaluation results and analytical information for all authorized users regardless of their location. This approach supports collaborative work of testers, analysts, and administrators within a unified information environment and simplifies the deployment of the system in organizational infrastructures.

The system is implemented using modern web technologies to ensure its efficient operation, maintainability, and extensibility. The client side is developed using React, which enables the creation of responsive and interactive user interfaces. The server side is implemented using FastAPI, providing high-performance RESTful services and efficient communication between system components. PostgreSQL is used as the database management system due to its reliability, support for complex analytical queries, and strong data consistency mechanisms.

The selected technology stack provides a clear separation between presentation, business logic, and data management layers. React enables the development of a dynamic and user-friendly interface that supports data import, visualization of quality indicators, generation of reports, and administration functions. FastAPI serves as the core of the application logic, handling data processing, metric calculation, integrated assessment procedures, user authentication, and communication between system modules. The use of RESTful APIs ensures interoperability between the client and server sides of the system and facilitates integration with third-party services. PostgreSQL provides centralized management of system data and supports efficient retrieval and processing of accumulated information. Together, these technologies provide a robust technological basis for implementing the proposed architecture.

A significant advantage of the proposed solution is its ability to integrate with DevOps infrastructures. Through standardized API interfaces, the system can automatically receive information from CI/CD pipelines and external quality assurance tools. This capability enables continuous monitoring of software test quality and supports the implementation of data-driven quality management practices [5]. The integration mechanism reduces manual data processing and allows evaluation procedures to be incorporated into existing software development workflows. As a result, software quality indicators can be monitored throughout the entire development lifecycle, enabling timely detection of quality issues and supporting informed decision-making based on objective assessment results.

The proposed architecture also provides opportunities for further development. Additional analytical modules, intelligent decision-support mechanisms, machine learning algorithms, and advanced reporting tools can be incorporated without substantial modifications to the core architecture.

Conclusions

The proposed solution combines centralized data management, automated assessment procedures, analytical visualization, reporting facilities, and integration with software development environments within a unified platform.

The system architecture supports scalability, flexibility, and efficient processing of quality-related information. The use of modern web technologies ensures high performance, convenient user interaction, and straightforward integration with software engineering tools.

The developed system contributes to improving the objectivity and efficiency of software test quality evaluation and creates a foundation for the implementation of advanced analytical and intelligent decision-support mechanisms in software quality assurance processes. Furthermore, the modular architecture of the

proposed solution enables its adaptation to different project requirements and facilitates future extension through the incorporation of new metrics, assessment methods, and analytical capabilities.

REFERENCES

1. Syerov Y. Analysis of quality management tools for software testing / Yuriy Syerov, Khrystyna Terletska, Michal Gregušml // Proceedings XIV International Academic Conference “Information, Communication, Society”, Zozuli (Lviv region), 22–24 May 2025. – Lviv, 2025. – P. 35–36.
2. Sokol D. Software Testing Quality Assessment System Based on Multi-Metric Evaluation / D. Sokol // Information Technologies: Science, Engineering, Technology, Education, Health : Abstracts XXXIV International Scientific-Practical Conference MicroCAD-2026, Kharkiv, 13–16 May 2026. – Kharkiv, 2026. – P. 1493.
3. ISO/IEC 25010:2011. Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models.
4. Athanasiou D. Test Code Quality and Its Relation to Issue Handling Performance / Dimitrios Athanasiou, Ariandi Nugroho, Joost Visser, Andy Zaidman // IEEE Transactions on Software Engineering. – 2014. – Vol. 40, no. 11. – P. 1100–1125.
5. Graham D. Foundations of Software Testing: ISTQB Certification, 4th edition / Dorothy Graham, Rex Black, Erik van Veenendaal. – 4th ed. – [S. l.] : Cengage Learning EMEA, 2019. – 288 p.

Сокол Дар’я Вікторівна – студентка групи КН-М225, навчально-науковий інститут комп’ютерних наук та інформаційних технологій, Національний технічний університет «Харківський політехнічний інститут», м. Харків, e-mail: daria.sokol@cs.khpi.edu.ua

Науковий керівник: ***Лютенко Ірина Вікторівна –*** канд. техн. наук, доцент кафедри програмної інженерії та інтелектуальних технологій управління, Національний технічний університет «Харківський політехнічний інститут», м. Харків

Sokol Daria Viktorivna – student of group KN-M225, Educational and Scientific Institute of Computer Sciences and Information Technologies, National Technical University “Kharkiv Polytechnic Institute”, Kharkiv, email: daria.sokol@cs.khpi.edu.ua

Supervisor: ***Liutenko Iryna Viktorivna*** – Cand. Sc. (Tech.), Associate Professor of the Software Engineering and Management Intelligent Technologies Department, National Technical University “Kharkiv Polytechnic Institute”, Kharkiv