

# СИСТЕМА СОВМЕСТНОЙ РАБОТЫ С РЕЗУЛЬТАТАМИ АВТОМАТИЗИРОВАННОГО ТЕСТИРОВАНИЯ

Гончарик И. А., Гуринович А. Б.

Кафедра информационных технологий автоматизированных систем, Белорусский государственный университет информатики и радиоэлектроники  
Минск, Республика Беларусь

E-mail: amsterget@gmail.com, gurinovich@bsuir.by

*В статье исследуются проблемы актуальности применения автоматизированного тестирования и вопросы, которые возникают в ходе использования данного подхода к тестированию программных продуктов. Производится описание системы, предлагающей инструменты для их решения.*

## ВВЕДЕНИЕ

Автоматизированное тестирование программного обеспечения стало неотъемлемой частью процесса разработки ПО. Это является одним из мировых трендов для компаний, занимающихся ИТ-инженерией. Использование автоматизированного тестирования позволяет оптимизировать процесс разработки программного продукта основные цели данного подхода представлены на рисунке 1.



Рис. 1 – Цели автоматизированного тестирования

Преимущества такого подхода к тестированию программных продуктов бесспорны, однако, существует ряд недостатков. Главная проблема заключается в отсутствии централизованного управления результатами автотестов. Это влечёт за собой невозможность получения информации о состоянии автоматизации в целом, например, менеджерами проекта и заинтересованными лицами.

## I. АКТУАЛЬНОСТЬ СИСТЕМЫ

Компании, использующие автоматизированное тестирование в процессе разработки ПО, вынуждены мириться с недостатками данного подхода. Проблемы связаны, в основном, со структуризацией полученных результатов автотестов, приведением результатов тестирования различных частей программного продукта к одному виду и организацией простого доступа к этим данным любому члену команды.

Автоматизация тестирования предполагает написание программного кода, тестирующего работу другого кода. В общем случае, различные части программного продукта реализуются с использованием разных технологий и языков программирования, что влечёт за собой и многообразие платформ для их тестирования. Таким образом получение актуальных данных о состоянии автоматизации и, следовательно, работоспособности функционала ПО, является задачей, требующей непосредственного контакта управляющего звена с группами тестировщиков-автоматизаторов. Это создаёт дополнительные временные затраты на обсуждение и визуализацию результатов тестирования.

В связи с этим указанные недостатки автоматизированного тестирования дают толчок к созданию инструмента агрегации и визуализации результатов автотестов, а также совместной работы с ними.

## II. ПОСТАНОВКА ЗАДАЧИ

Системе организации совместной работы с результатами автоматизированного тестирования необходима поддержка следующих возможностей:

- Универсальность. Представляет собой возможность системы работать с результатами автотестов независимо от платформы тестирования.
- Визуализация данных. Система включает в себя возможность графического отображения результатов тестирования в виде графиков и диаграмм.
- Совместный доступ. Предполагает собой возможность организации совместной работы с результатами тестирования в системе с поддержкой разграничения прав доступа к данным.
- Гибкость. Включает в себя возможность продукта по интеграции со сторонними сервисами, включая существующие системы отслеживания дефектов и сервера электронной почты.

Архитектура системы представляет собой сетевую клиент-серверную архитектуру. Совокупность взаимодействующих между собой посредством сетевых протоколов сервисов серверная часть. Сторона клиента представлена группой посредников, так называемых «агентов», являющихся связующим звеном между платформами для тестирования и серверной частью приложения. Для визуализации данных и упрощения работы с ними также реализуется сервис пользовательского интерфейса. Взаимодействие между «агентами» и серверной частью системы осуществляется через API (Application programming interface). Структурная схема системы представлена на рисунке 2.

Для реализации серверной части системы используется объектно-ориентированный язык программирования Java совместно со Spring Framework, система управления базами данных PostgreSQL. Все используемые технологии являются программным обеспечением с открытыми исходным кодом. Это даёт доступ к большому количеству инструментов и сообществу, что способствует успешному процессу разработки.

Поддержка необходимой платформы тестирования требует реализации соответствующего «агента», «понимающего» язык платформы и «умеющего» работать с API сервера.

В работе показано, что автоматизируя тестирование работоспособности на соответствие требованиям запроецированного функционала программного продукта, можно существенно уменьшить риски получения некачественного продукта и оптимизировать затраты на разработку. При переводе описания тест-кейсов в код, тестировщик-автоматизатор получает задокументированную часть функционала системы и возможность в любой момент проверить это соответствие. Для получения полного представления о готовности программного продукта, необходимо систематизировать результаты тестирования всего приложения и произвести их визуализацию. Система совместной работы с результатами автоматизированного тестирования призвана помочь в решении данных задач.

СПИСОК ЛИТЕРАТУРЫ

1. Автоматизированное тестирование программного обеспечения / Э. Дастин, Дж. Рэшка, Д. Пол // Издательство: Лори, 2003. – 580 с.
2. Experiences of Test Automation: Case Studies of Software Test Automation / D. Graham, M. Fewster // Addison-Wesley, 2012. – 607 с.
3. Problems with Test Automation and Modern QA [Electronic resource] / A. Ghahrai. – 2018. – Mode of access: <https://www.testingexcellence.com/problems-test-automation-modern-qa/>.

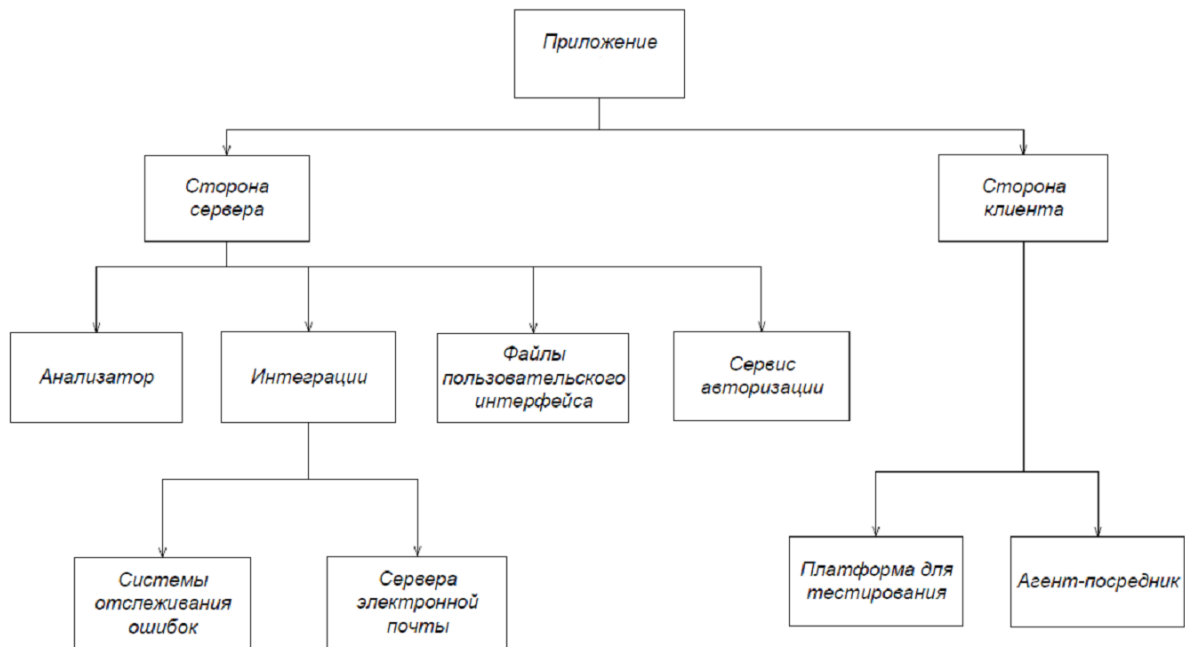


Рис. 2 – Структурная схема системы