

УДК 004.04

**СОВЕРШЕНСТВОВАНИЕ ПРОЦЕССОВ ТЕСТИРОВАНИЯ
ОБЛАЧНЫХ ИНФОРМАЦИОННЫХ СИСТЕМ: ОПЫТ ВНЕДРЕНИЯ
МЕТОДОЛОГИИ КОНТРОЛЯ КАЧЕСТВА**

Виноградская М.Ю.,

к. пед. н., доцент,

Калужский государственный университет им. К.Э. Циолковского,

Калуга, Россия

Журавлева В.В.,

магистрант,

Калужский государственный университет им. К.Э. Циолковского,

Калуга, Россия

Аннотация.

В статье рассматриваются проблемы тестирования облачных платформ на примере FKK Cloud на базе OpenStack. Анализируется недостаточность традиционных методологий для распределённых систем и предлагается адаптация модели качества ISO/IEC 25010 с учётом специфических облачных атрибутов. Реализован подход «Качество как сервис» (QaaS). С применением диаграмм Парето и Исикавы выявлены ключевые причины дефектов. Предложен комплекс мер по автоматизации тестирования, внедрение которых позволило сократить время тестирования на 40% и повысить стабильность релизов. В конце представлены выводы по проделанной работе.

Ключевые слова: облачные технологии; тестирование ПО; автоматизация тестирования; диаграмма Исикавы; диаграмма Парето; OpenStack.

***IMPROVING THE TESTING PROCESSES OF CLOUD-BASED
INFORMATION SYSTEMS: EXPERIENCE IN IMPLEMENTING QUALITY
CONTROL METHODOLOGIES***

Vinogradskaya M.Y.,

Candidate of Pedagogical Sciences, Associate Professor,

Kaluga State University named after K.E. Tsiolkovsky,

Kaluga, Russia

Zhuravleva V.V.,

Undergraduate,

Kaluga State University named after K.E. Tsiolkovsky,

Kaluga, Russia

Annotation.

This article examines the challenges of cloud platform testing using OpenStack-based FKK Cloud as an example. The inadequacy of traditional methodologies for distributed systems is analyzed and an adaptation of the ISO/IEC 25010 quality model is proposed, taking into account specific cloud attributes. A Quality as a Service (QaaS) approach is implemented. Using Pareto and Ishikawa diagrams, key defect causes are identified. A set of testing automation measures is proposed, the implementation of which reduced testing time by 40% and increased release stability. Finally, a summary of the work completed is presented.

Keywords: cloud technologies; software testing; test automation; Ishikawa diagram; Pareto diagram; OpenStack.

В эпоху передовых технологий и постепенного развития их составляющих проблема обеспечения качества становится первостепенной. Увеличение темпов разработки требует больших вложений в тестирование. Автоматизированные тесты не всегда покрывают все аспекты, представленные в требованиях, особенно если это касается облачных сервисов.

Дневник науки | www.dnevniknauki.ru | СМН ЭЛ № ФС 77-68405 ISSN 2541-8327

Традиционные методологии тестирования, разработанные для монолитных и статичных систем, оказываются недостаточно эффективными в условиях облачной парадигмы. Многообразие облачных платформ, различия в инфраструктуре, проблемы межплатформенной совместимости и необходимость многократной ревалидации сервисов на различных этапах жизненного цикла требуют переосмысления подходов к контролю качества [3].

Облачные информационные системы представляют собой сложные распределённые экосистемы, характеризующиеся рядом особенностей, существенно отличающих их от традиционного программного обеспечения. В соответствии с эталонной архитектурой NIST SP 800-145 (2011) включают пять основных характеристик (самообслуживание по требованию, широкополосный доступ к сети, пул ресурсов, быстрая эластичность, измеряемость сервиса), три модели сервиса (IaaS, PaaS, SaaS) и четыре модели развёртывания (частное, публичное, гибридное и общественное облако).

Фундаментальной основой совершенствования процессов тестирования является разработка стандартизированных моделей качества. Международный стандарт ISO/IEC 25010 [1] предоставляет базовую модель качества, включающую характеристики функциональной пригодности, надёжности, эффективности производительности, удобства использования, безопасности, совместимости, сопровождаемости и переносимости. В контексте облачных систем данная модель требует адаптации: расширение традиционной модели качества за счёт учёта специфических для облака атрибутов: эластичности, мультитенантности, оплаты по требованию и масштабируемости [6].

Перспективным направлением является концепция «Качество как сервис» (QaaS), которая интегрирует вопросы качества, включая функциональное поведение, мониторинг производительности, управление жизненным циклом и безопасность сервиса. Данный подход предполагает, что тестирование само становится облачным сервисом, доступным по требованию, что позволяет

проводить тестирование онлайн перед развёртыванием приложения, используя преимущества облака для аутсорсинга тестовых задач [2].

Наше исследование основано на практическом опыте разработки и тестирования облачной платформы FKK Cloud. FKK Cloud – облачная платформа на базе OpenStack, представляющая услуги по модели IaaS (Infrastructure as a Service). Сервис позволяет пользователям и организациям создавать и управлять виртуальными ИТ-инфраструктурами без капитальных затрат на оборудование.

Платформа OpenStack содержит ключевые компоненты: Nova для управления виртуальными машинами (ВМ); Neutron для сетевой виртуализации; Cinder для блочного хранилища; Glance для каталога образов ОС; Keystone для аутентификации и авторизации пользователей; Swift для объектного хранилища.

Эти модули обеспечивают полный цикл работы с облачными ресурсами от развёртывания ВМ до хранения данных и настройки сетевых соединений.

Первоочередное направление в тестировании — это информационная безопасность [2]. Особое внимание требуется уделить защите данных и обеспечению непрерывности бизнес-процессов. Таким образом сформирован комплексный подход к обеспечению безопасности [4], [6]: Мониторинг уязвимостей, Сетевая безопасность на основе современных решений Firewall и WAF; Антивирусная защита всех узлов инфраструктуры; Резервное копирование с долгосрочным хранением данных; Многофакторная аутентификация для критически важных сервисов; Контроль передачи чувствительных данных; Документация процедур с учётом уровней доступа; Обучение персонала и повышение компетенций.

Концепция исследования включает современные подходы к анализу дефектов с помощью диаграмм Парето и Исикавы. Применение этих инструментов позволит: Выявить основные категории дефектов; Определить приоритетные направления для улучшения; Проанализировать причины возникновения проблем; Сформировать базу для разработки корректирующих мероприятий.

Для детального анализа собрана статистика по дефектам за 1 год и рассчитана доля и кумулятивный процент (таблица 1).

Таблица 1. Категории дефектов

Причина дефекта	Количество	Доля (%)	Кумулятивный (%)
UX/UI	11	36.7%	36.7%
API Backend	4	13.3%	50.0%
Прочее	4	13.3%	63.3%
1С Интеграция	3	10.0%	73.3%
Openstack	2	6.7%	80.0%
Интеграция с ТБанк	2	6.7%	86.7%
Ошибка миграции	1	3.3%	90.0%
CI/CD	1	3.3%	93.3%
Nginx	1	3.3%	96.7%
DNS	1	3.3%	100.0%

Исходя из данных построена диаграмма Парето, которая ранжирует ошибки по степени влияния (рисунок 1).

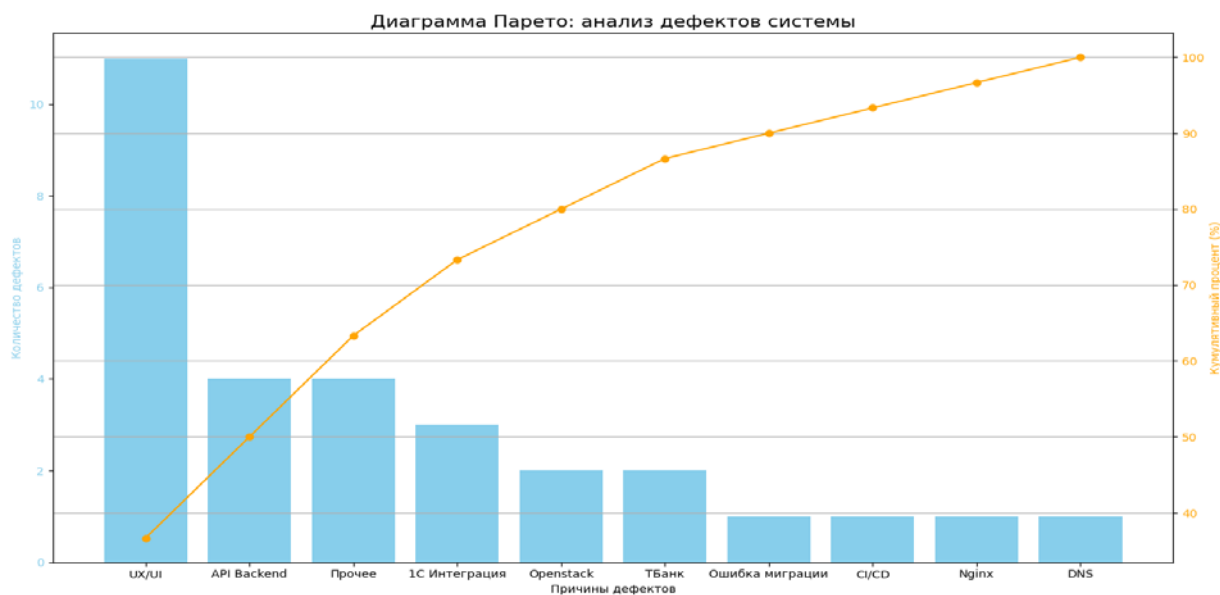


Рис. 1 - Диаграмма Парето (составлено авторами)

Анализ диаграммы выявляет ключевые системные проблемы:

- 50% всех дефектов приходится на первую категорию (UX/UI)

- Топ-3 проблемы составляют 63.3% всех дефектов

А также определяются области для улучшения:

- Проблемы с UX/UI (36.7%)
- Проблемы с API Backend (13.3%)
- Несоответствие требованиям (13.3%)

Для диаграммы Исикавы использовался метод «6М». За основу взят анализ причин возникновения дефектов (см. рисунок 2).

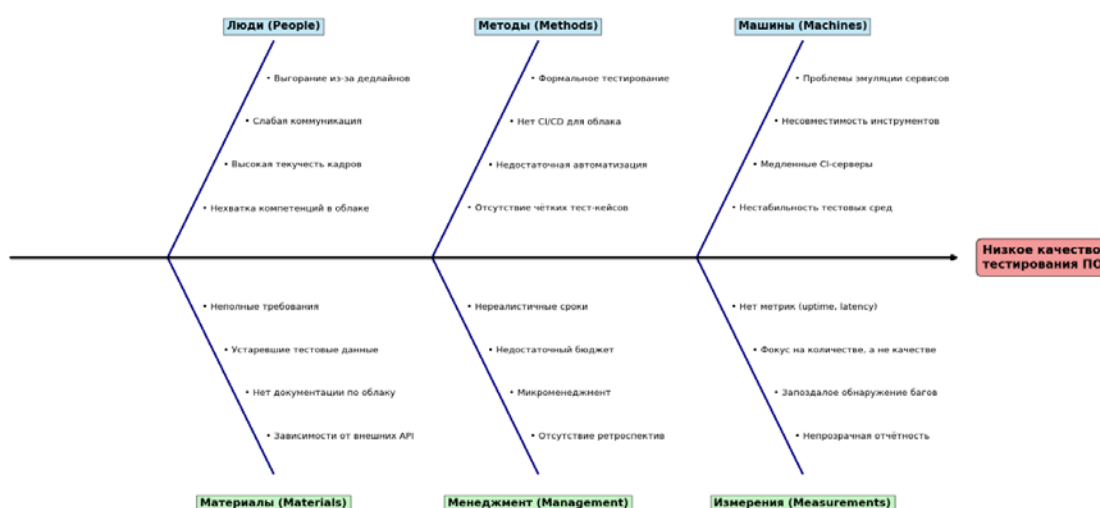


Рис. 2 - Диаграмма Исикавы (составлено авторами)

На основе диаграммы Исикавы определены ключевые проблемы:

1. Человеческий фактор:
 - о Недостаточная квалификация специалистов
 - о Слабая коммуникация между отделами
2. Методология:
 - о Отсутствие четких тест-кейсов
 - о Формальный подход к тестированию
3. Инфраструктура:

- о Нестабильность тестовых сред
- о Проблемы с CI/CD
- о Недостаточное покрытие тестами

Предложен комплекс мер по совершенствованию системы тестирования:

- Внедрение автоматизированной системы контроля качества
- Разработка методологии регрессионного тестирования
- Создание модулей API-тестирования
- Внедрение нагрузочного тестирования

Внедрение предложенных решений позволило:

- Сократить время тестирования на 40%
- Повысить качество релизов
- Улучшить показатели стабильности
- Оптимизировать процессы тестирования

Проведенные исследования показали эффективность предложенного подхода к контролю качества облачных информационных систем. Применение авторами статистических инструментов анализа дефектов — диаграммы Парето и причинно-следственной диаграммы Исикавы — позволило с высокой точностью идентифицировать «узкие места» процесса разработки. Количественный анализ выявил, что более 63% всех выявленных дефектов сконцентрированы в трех областях: пользовательский интерфейс и опыт (UX/UI) — 36,7%, ошибки бэкенд-API — 13,3%, а также прочие несоответствия требованиям — 13,3%. Это распределение указывает на системный дисбаланс между приоритетами разработки и реальными пользовательскими сценариями. Разработанные методы могут быть применены при создании и эксплуатации аналогичных систем.

Библиографический список:

1. ГОСТ Р ИСО/МЭК 25010-2015. Информационные технологии. Системная и программная инженерия. Требования и оценка качества систем и программного

обеспечения (SQuaRE). Модели качества систем и программных продуктов. — Москва: Стандартинформ, 2015. — 32 с.

2. Майерс Г. Искусство тестирования программ / Г. Майерс; пер. с англ. под ред. Позина. - М.: Финансы и статистика. - 1982. — 176 с.

3. Степанченко И.В. Методы тестирования программного обеспечения: учеб. пособие / И.В. Степанченко. - Волгоград: ВолгГТУ,- 2006. — 76 с.

4. Калбертсон Р. Быстрое тестирование / Р. Калбертсон, К. Браун, Г. Кобб. - М.: Издательский дом «Вильямс». - 2003. — 374 с.

5. Синицын С.В. Верификация программного обеспечения / С.В. Синицын, Н.Ю. Налютин.- М., - 2006. — 158 с.

6. Тамре Л. Введение в тестирование программного обеспечения / Л. Тамре ; пер. с англ. - М. : Издательский дом «Вильямс». - 2003. — 368 с.