

ПРИМЕНЕНИЕ МЕТРИКИ ХОЛСТЕДА ДЛЯ РАСЧЕТА СТОИМОСТИ АВТОМАТИЗАЦИИ ТЕСТИРОВАНИЯ ПРОГРАММНЫХ СИСТЕМ

Галимова Е. Ю., ассистент кафедры ИиУС

г. Санкт-Петербург, ФГБОУ ВО Санкт-Петербургский государственный университет промышленных технологий и дизайна

Применение автоматизации тестирования имеет ряд преимуществ. Позволяет оптимизировать временные затраты, избавиться от однообразных проверок. Автоматизированные тесты быстро запускаются на выполнение, есть возможность автоматической генерации отчетов[1]. Внедрение автоматизации является дорогостоящим процессом, поэтому рекомендуется сделать предварительный расчет стоимости.

В частности, рассмотрим первый год внедрения автоматизации. Для расчета стоимости внедрения (W_{A1}) применим формулу, использующую метрику Холстеда. Подробный вывод формулы описывается в работе[2].

$$W_{A1} = W_L + W_H \cdot (T_{A1} + T_{R1}), \text{ где}$$

W_L – стоимость лицензионного программного обеспечения для автоматизации процесса тестирования;

W_H – стоимость часа рабочего времени специалиста;

T_{A1} – количество часов, запланированных на разработку тестов;

T_{R1} – количество часов, запланированных на анализ результатов тестирования.

$$T_{R1} = \gamma \cdot h \cdot B, \text{ где}$$

h – время описания одного выявленного дефекта;

γ – коэффициент для определения уровня профессионализма специалиста;

B – метрика Холстеда, характеризующая количество возможных дефектов в программе.

На рисунках 1 и 2 представлены экспериментально полученные данные по финансовым затратам на внедрение автоматизации за первый год (W_{a1}) в зависимости от количества потенциально возможных ошибок в программном коде (B).

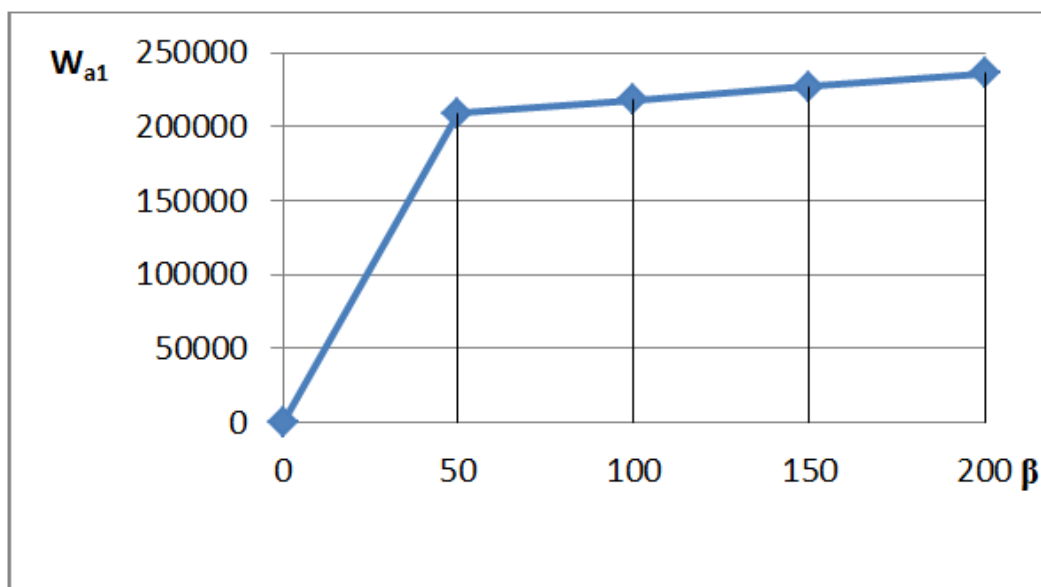


Рисунок 1 – Зависимость стоимости автоматизации от потенциально возможного количества ошибок в программном коде при платной лицензии

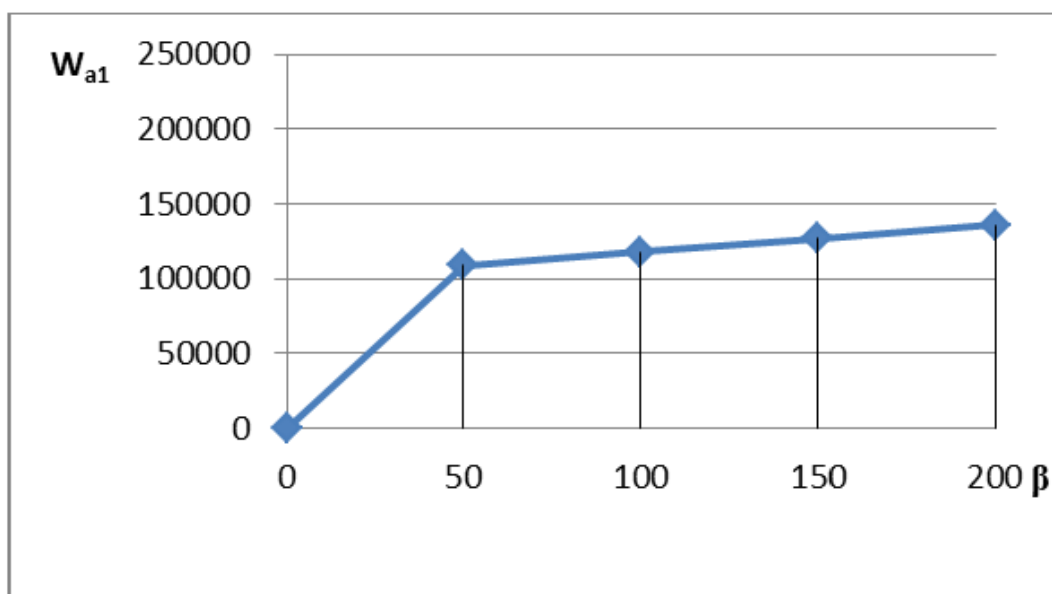


Рисунок 2 – Зависимость стоимости автоматизации от потенциально возможного количества ошибок в программном коде при бесплатной лицензии

Как наглядно видно из графиков, в первый год внедрения автоматизации значительная часть расходов приходится на покупку лицензионного программного обеспечения. Если на автоматизацию выделен небольшой бюджет, рекомендуется выбирать свободное программное обеспечение (например, Selenium, KatalonStudio, Watir, Robotframework)[3]. Selenium применяется в первую очередь для проверки веб-приложений на различных платформах с использованием большого набора браузеров [4]. Тесты можно создавать с использованием разных языков программирования, таких как Python, Ruby, C#, JavaScript. Программное обеспечение KatalonStudio создано на базе Appium и Selenium. Предназначено для тестирования API, веб и мобильных приложений. Для создания тестов преимущественно используют языки Java и Groovy. Инструмент Watir предназначен для автоматизации тестирования веб-приложений. Тесты пишут на языке Ruby. Для приемочного тестирования используют инструмент Robotframework, который позволяет создавать тесты в табличном формате на базе ключевых слов. Для работы требуется установка среды языка Python.

Современные перспективы улучшения качества программных систем связаны с применением автоматизированного тестирования в сочетании с ручным. Процесс внедрения автоматизации должен быть рациональным и поэтапным, с предварительным составлением математической модели.

Литература

1. Mark Fewster, Dorothy Graham. Software Test Automation. Effective use of test execution tools. Addison-Wesley, 1999. 574 p.
2. Галимова Е. Ю. Метод оценки затрат на применение автоматизированного тестирования программного продукта // Научно-технический вестник Поволжья, 2019. №10. С. 27 – 29.
3. Галимова Е. Ю. Анализ алгоритма принятия решения об автоматизации тестирования программного продукта с применением свободного

программного обеспечения Selenium // Теория и практика применения
свободного программного обеспечения: сборник трудов участников
Российской молодежной конференции с элементами научной
школы, 18-22 мая 2016 г. Магнитогорск: Изд-во Магнитогорск. гос.
техн. ун-та им. Г. И. Носова, 2016. С. 133-138.

4. UnmeshGundecha, CarlCocchiaro. LearnSelenium.PacktPublishing, 2019.
536 p.