

ИССЛЕДОВАНИЕ РЕСУРСОЕМКОСТИ ПРИМЕНЕНИЯ АВТОМАТИЗИРОВАННОГО ТЕСТИРОВАНИЯ ПРИ РАЗРАБОТКЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

RESEARCH OF RESOURCE INTENSITY OF AUTOMATED TESTING IN SOFTWARE DEVELOPMENT

**B. Latypov
T. Ziyazetdinov
A. Mukhetdinov**

Summary. This article describes the advantage of using automated testing to improve quality and reduce risks in software development. Particular attention is paid to the resource intensity of the use of automated testing in development. The methodology of the automated testing life cycle is considered. The characteristic features of automation of software product quality analysis processes are highlighted and described. As a research component, the authors assessed the resource intensity of testing processes and improved indicators when using tools for automated software testing and reporting. The authors of this article seek to trace the decision-making process on the implementation of automation on the project with a further choice of tools and the scope of their application. Based on the analysis of the time spent by a specialist in the field of testing, it is established that automated testing saves the company's resources and allows re-checking the software with minimal effort, which significantly reduces the time compared to manual testing and helps to reduce the number of human errors. This direction is also supplemented by the consideration of the use of various tools for automating various kinds of testing directions on the example of developing a web application.

Keywords: development, testing, software, information technology, automation.

Латыпов Булат Фиделевич

Уфимский государственный нефтяной технический
университет, Уфа
linksmtv@gmail.com

Зиязетдинов Тимур Рифович

Уфимский государственный нефтяной технический
университет, Уфа
ziyazetdinov.timur@gmail.com

Мухетдинов Амир Русланович

Уфимский государственный нефтяной технический
университет, Уфа
amir.muhetdinov@yandex.ru

Аннотация. В данной статье описывается преимущество использования автоматизированного тестирования для улучшения качества и уменьшения рисков при разработке программного обеспечения. Особое внимание уделяется ресурсоемкости применения автоматизированного тестирования при разработке. Рассматривается методология жизненного цикла автоматизированного тестирования. Выделяются и описываются характерные особенности автоматизации процессов анализа качества программного продукта. В качестве исследовательской составляющей авторами была проведена оценка ресурсоемкости процессов тестирования и улучшения показателей при использовании инструментов для автоматизированного тестирования программного обеспечения и предоставления отчетности. Авторы данной статьи стремятся проследить процесс принятия решения о внедрении автоматизации на проекте с дальнейшим выбором инструментов и областью их применения. На основании анализа затрат рабочего времени специалиста в области тестирования устанавливается, что автоматизированное тестирование экономит ресурсы компании и позволяет с минимальными усилиями повторно проверять программное обеспечение, что значительно сокращает время по сравнению с проведением ручного тестирования и способствует уменьшению количества ошибок со стороны человека. Данное направление дополняется также рассмотрением использования различных инструментов для автоматизации разного рода направлений тестирования на примере разработки веб-приложения.

Ключевые слова: разработка, тестирование, программное обеспечение, информационные технологии, автоматизация.

Введение

В современных реалиях разработки программного обеспечения люди сталкиваются с недостатком времени и ресурсов при тестировании продукта.

Для достижения быстрой и качественной проверки ПО применяют автоматизированное тестирование.

Автоматизированное тестирование — это метод тестирования программного обеспечения, который выполняется с использованием специальных программных средств, которые, в свою очередь, необходимы для выполнения набора тестовых сценариев.

В данной статье авторы описывают не только процесс внедрения автотестирования на проекте, но и на примере создания тестового сценария сравнивают ресурсоемкость ручного и автоматизированного методов проверки качества разрабатываемого программного обеспечения.

Методология автоматизированного тестирования

Жизненный цикл автоматизированного тестирования описывает этапы создания и применения данного подхода в ИТ-проекте (рис. 1).

Первым пунктом жизненного цикла идет принятие решения об использовании на проекте автоматизированного тестирования. Для этого нужно понять какие преимущества дадут изменения, поддержит ли руководство и обязательно уточнить навыки внедрения автоматизированного тестирования у действующей команды инженеров качества [1].

После анализа и принятия решения о внедрении автоматизированного тестирования, стоит определить какие сферы тестирования требуются для покрытия и какими инструментами будет достигаться.

Выбор инструмента для тестирования зависит от технологий, на которых разрабатывается продукт, для этого требуется оценить масштаб и цели проекта.

Вам нужно составить список функциональности и соответствующих требований к автоматизации.

К примеру, ваше приложение должно быть совместимо с разными браузерами, есть готовый функционал, использующий базы данных, либо у вас уже есть тестовые сценарии, разработанные вашими специалистами по тестированию, либо в приложении требуется автоматизация API.

Для начала нужно определить платформу, для автоматизации сценариев из «регрессионных», «дымовых» тест-планов, для этих целей могут подойти такие инструменты как Selenium, Appium, Robotium, для нагрузочного тестирования — «JMeter», для автоматизации API — Postman.

После того, как вы выбрали фреймворки и инструменты для автоматизированного тестирования, стоит оценить стоимость внедрения в ваш проект данного новшества, возможность использования бесплатного или проприетарных фреймворков.

Основным документом на 3 этапе жизненного цикла является PoC — “Proof of concept”, с помощью которого доказывается осуществимость внедрения на практике. В нем описывается какой будет фреймворк в проекте, его доступность, соответствие требованиям и гибкость.

Данный документ помогает увидеть риски используемых инструментов, а также предотвратить возможное недопонимание среди команды.

Следующим этапом является планирование, проектирование и разработка. На данном этапе команде тестирования требуется создать стратегию и план для автоматизации, а также определить элементы входящие и выходящие в рамках автоматизации.

Как правило, автоматизируют основной функционал разрабатываемого продукта, входящий в «регрессионный», «дымовой» и «интеграционный» тест-планы, так как они занимают огромное количество времени для проверки [2]. Так же для нагрузочного и производительного тестирования нет альтернатив в виде ручной проверки. Базовый функционал — «дымовые тесты», самый простой и относительно легко реализуемый в автоматизации тест, который может протестировать базовый функционал или доступность API.

Пятым этапом жизненного цикла является выполнение теста. На данном этапе выполняются сценарии автоматизации и предоставляются подробные отчеты об испытаниях. Инструменты, используемые для отчетности: Junit, Allure.

На этапе оценки и совершенствовании следует оценить проделанную работу, используя метрики и такие показатели как:

- ◆ процент обнаруженных дефектов;
- ◆ улучшение производительности;
- ◆ затраченное время для тестирования продукта.

Цель исследования

Провести сравнительный анализ применения двух подходов к тестированию, и выяснить какой из методов является наиболее ресурсоемким.



Рис. 1. Методология жизненного цикла автоматизированного тестирования

Материалы и методы исследования

Авторами был составлен тест-кейс (табл. 1), затрагивающий область авторизации и заказа продукта в web-приложении [3].

На основе данного тест-кейса был составлен автоматизированный тест с использованием таких инструментов как: Selenium, NUnit, Specflow, Gherkin. Specflow и Gherkin позволяют описывать сложнейшие методы на доступном для бизнеса и других членов команды языке, в нашем случае на русском. Также для написания методов был использован язык программирования C# и IDE Rider (рис. 2).

Составленный отчет по прохождению автоматизированного теста представлен на рисунке 3.

Результаты исследования и их обсуждение

После сравнения методов ручного и автоматизированного тестирования на примере оформления заказа в онлайн-магазине мы получили результаты, указанные в таблице 2.

Каждый день тестировщик сталкивается с релизом нового функционала, который нужно проверить и убедиться, что он не повлиял на основные функции системы. Поэтому при проверке новой версии программы, тестировщиком должно проводиться дымовое тестирование системы, которое включает в себя около 15 тест-кейсов, на прохождение которых уходит в среднем по 2 минуты. Дымовые тесты являются важной частью регрессионных тест-кейсов, так что задача команды инженеров качества заключается в быстрой проверке работоспособности и стабильности программного обеспечения в целом.

Автоматизированные тесты позволяют тестировщику существенно уменьшить временные затраты на прохождение проверки. Исходя из вышенаписанного, тестировщику требуется тратить порядка 30 минут на проверку стенда в день, когда автоматизированные тесты не требуют временных затрат.

Так же стоит упомянуть и о регрессионном тестировании продукта, которое как правило, в зрелых командах, проходит два раза в месяц. Регрессионный тест-план содержит большое количество тест-кейсов, в больших проектах он может достигать значения порядка 1000, но примем, что в нашем проекте тестиров-

Таблица 1. Тест-кейс

№	Шаги	Ожидаемый результат
1	Войти на сайт	Отображается главная страница сайта
2	Ввести в поле поиска «Детектив»	В поле поиска отображается введенный поисковый запрос
3	Нажать кнопку «Поиск»	1. Отображается вкладка поиска 2. На вкладке поиска отображаются карточки книг соответствующих поисковому запросу
4	Нажать на первую карточку книги	Отображается страница выбранной книги с описанием и ценой
5	Нажать кнопку «Купить»	На месте кнопки «Купить» отображается кнопка «Оформить заказ»
6	Нажать кнопку «Оформить заказ»	1. Отображается вкладка корзины 2. На вкладке корзины отображается количество товаров и общая стоимость заказа в корзине
7	Нажать кнопку «Оформить заказ»	Отображается модальное окно авторизации
8	В поле Логин ввести «Linksmtv@gmail.com»	В поле Логин отображается введенный логин
9	В поле Пароль ввести «123444321b»	В поле Пароль отображается введенный пароль в зашифрованном виде
10	Нажать кнопку «Войти»	1. Модальное окно авторизации закрывается 2. В правом углу на месте кнопки «Войти» отображается имя пользователя 3. Открывается вкладка Оформления заказа
11	Выбрать способ доставки «Магазины сети»	Под пунктом «Магазины сети» отображается выпадающая вкладка с картой магазинов города
12	Выбрать магазин	Отображается выпадающий список адресов магазинов
13	Выбрать магазин и нажать «Заберу отсюда»	1. Выпадающая вкладка Способа доставки закрывается 2. Раскрывается выпадающая вкладка Оплаты
14	Выбрать «Оплата при получении заказа»	1. Чек-бокс «Оплата при получении заказа» становится активным 2. Активируется кнопка «Указать контактные данные»
15	Нажать кнопку «Указать контактные данные»	1. Вкладка Оплаты закрывается 2. Раскрывается выпадающая вкладка «Контактные данные»
16	Ввести контактные данные	1. Контактные данные отображаются в соответствующих полях 2. Активируется кнопка «Подтвердить данные»
17	Нажать кнопку «Подтвердить данные»	1. Выпадающая вкладка «Контактные данные» закрывается 2. Раскрывается выпадающая вкладка «Подтверждение» 3. Во вкладке Подтверждение отображается информация по заказу: название и количество книг, цена и итоговая стоимость заказа
18	Нажать «Завершить оформление заказа»	1. Вкладка оформления заказа закрывается 2. Отображается окно принятия заказа 3. В окне принятия заказа отображается номер заказа и ожидаемая дата доставки

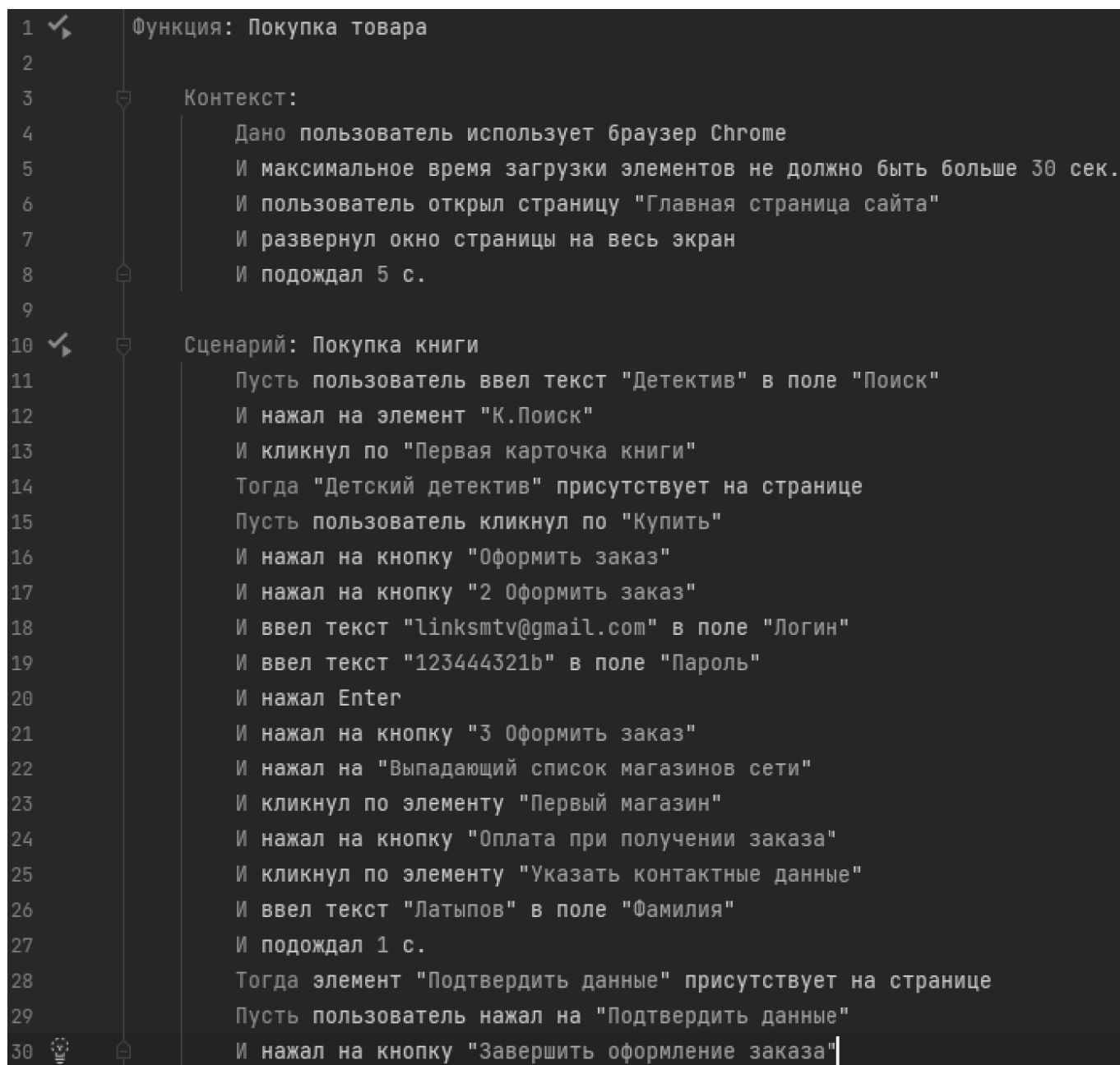


Рис. 2. Автоматизированный тест

щик делает 100 проверок, из которых возможно и целесообразно автоматизировать порядка 80 тестов.

Исходя из вышенаписанного тестировщик должен проводить дымовое и регрессионное тестирование ПО, временные затраты на которые за период одного месяца приведены в таблице 3.

Исходя из выполненных расчетов, мы видим, что в нашем случае, автоматизируя процесс тестирования, удастся достичь оптимизации трудозатрат инженера

качества для дымового тестирования уже за месяц, а регрессионного спустя 8 месяцев.

Из других преимуществ внедрения автотестирования можно отметить, что по данным портала по поиску работы SuperJob зарплата тестировщика во многих компаниях достигает и даже превышает 100 тысяч рублей [4]. При этом средний доход в России в 2022 году по данным Lenta.ru составляет 56,5 тысяч [5]. Исходя из этого, профессию тестировщика можно смело отнести к высококвалифицированной и как следствие

```

Дано пользователь использует браузер Chrome
-> done: DriverStub.UseChrome() (2,6s)
И максимальное время загрузки элементов не должно быть больше 30 сек.
-> done: WebBrowsersActions.SetTimeout(30) (0,0s)
И пользователь открыл страницу "Главная страница сайта"
Open page:
-> done: PageManipulationActions.OpenPage("Главная страница ...") (6,7s)
И развернул окно страницы на весь экран
-> done: PageManipulationActions.MaximizePageWindow() (0,2s)
И подошел 5 с.
-> done: WebBrowsersActions.WaitForFewSeconds(5) (5,0s)
Пусть пользователь ввел текст "Детектив" в поле "Поиск"
-> done: WebBrowsersActions.EnterTextInput("Детектив", "Поиск") (4,1s)
И нажал на элемент "К.Поиск"
Кликнул по элементу //i[@class='chg-icon chg-icon-search']
-> done: WebBrowsersActions.ClickByElement("К.Поиск") (4,1s)
И кликнул по "Первая карточка книги"
Кликнул по элементу //div[contains(text(), 'Хулиганский детектив: лесные истории')]
-> done: WebBrowsersActions.ClickByElement("Первая карточка к...") (4,2s)
Тогда "Детский детектив" присутствует на странице
Пусть пользователь кликнул по "Купить"
Кликнул по элементу //button[@class='button product product-action js__product_card_button js__button-detail']
-> done: WebBrowsersActions.ClickByElement("Купить") (0,1s)
И нажал на кнопку "Оформить заказ"
Кликнул по элементу //span[@class='js__card_button_text'] [contains(text(), 'Купить')]
-> done: WebBrowsersActions.ClickByElement("Купить") (0,1s)
И нажал на кнопку "Оформить заказ"
Кликнул по элементу //button[@class='button product product-action js__product_card_button js__button-detail']
-> done: WebBrowsersActions.ClickByElement("Оформить заказ") (4,2s)
И нажал на кнопку "2 Оформить заказ"
Кликнул по элементу //button[@class='btn basket__btn_buy go-to-order js__go-to-order']
И ввел текст "linksmnt@gmail.com" в поле "логин"
-> done: WebBrowsersActions.EnterTextInput("linksmnt@gmail.com", "логин") (0,4s)
И ввел текст "12344321b" в поле "пароль"
-> done: WebBrowsersActions.EnterTextInput("12344321b", "пароль") (0,1s)
И нажал Enter

```

```

-> done: WebBrowsersActions.PressEnter() (0,0s)
И нажал на кнопку "3 Оформить заказ"
Кликнул по элементу //button[contains(text(), 'Оформить заказ')]
-> done: WebBrowsersActions.ClickByElement("3 Оформить заказ") (4,1s)
И нажал на "Выпадающий список магазинов сети"
Кликнул по элементу //div[@class='js__shop_delivery_type'] //div[@class='map-view__container-content'] //div[1] //div[1] //img[1]
-> done: WebBrowsersActions.ClickByElement("Выпадающий список...") (4,9s)
И кликнул по элементу "Первый магазин"
Кликнул по элементу //body/div[contains(@class, 'page-map')] //main[@class='main'] /div[@id='order-page'] /form[@id='order-form'] /div[@class='order-form js__step-content_delivery'] /div[@class='js__delivery_types_mapper'] /div[@class='js__shop_delivery_type'] /fieldset[@class='fieldset fieldset_map_hidden'] /div[@class='map'] /div[@id='map-index-8'] /div[@class='map-view__container-content'] /div[@class='delivery__mapper js__delivery_mapper'] /div[@class='shop js__shop js__delivery_type'] /div[@class='shop-list js__shop-list clearfix shop-list_active'] /div[@class='shop-list_viewport js__shop-list_viewport'] /div[1] /div[1] /div[1] /div[1]
-> done: WebBrowsersActions.ClickByElement("Первый магазин") (0,2s)
И нажал на кнопку "оплата при получении заказа"
Кликнул по элементу //span[contains(text(), 'Оплата при получении заказа')]
-> done: WebBrowsersActions.ClickByElement("Оплата при получе...") (4,4s)
И кликнул по элементу "Указать контактные данные"
Кликнул по элементу //div[@class='btn_large js__complete_step']
-> done: WebBrowsersActions.ClickByElement("Указать контактные...") (0,1s)
И ввел текст "Латыпов" в поле "фамилия"
-> done: WebBrowsersActions.EnterTextInput("Латыпов", "Фамилия") (0,3s)
И подождал 1 с.
-> done: WebBrowsersActions.WaitForFewSeconds(1) (1,0s)
Тогда элемент "Подтвердить данные" присутствует на странице
-> done: WebBrowsersActions.ElementExistsInPage("Подтвердить данные") (0,0s)
Пусть пользователь нажал на "Подтвердить данные"
Кликнул по элементу //div[contains(text(), 'Подтвердить данные')]
-> done: WebBrowsersActions.ClickByElement("Подтвердить данные") (0,1s)
И нажал на кнопку "завершить оформление заказа"
Кликнул по элементу //div[@class='btn_large js__step-send_proof js__complete_step']
-> done: WebBrowsersActions.ClickByElement("Завершить оформле...") (8,4s)
-> done: WebBrowsersActions.WaitForFewSeconds(10) (10,0s)
Закреть браузер

```

Рис. 3. Составленный отчет Allure прохождения автотеста

Таблица 2. Сравнение результатов затраченного времени на ручное и автоматизированное тестирование

	Ручное тестирование	Автотестирование
Временные затраты на написание одного тест-кейса	30 минут	30 минут
Временные затраты на составление одного автотеста	–	30 минут
Средние временные затраты на прохождение одного тест-кейса	2 минуты	

Таблица 3. Сравнение трудозатрат тестировщика при ручном и автоматизированном тестировании на дистанции одного месяца

	Ручное тестирование	Автотестирование
Временные затраты на написание 15 тест-кейсов (дымовое тестирование)	450 минут	450 минут
Временные затраты на составление 15 автотестов (дымовое тестирование)	–	450 минут
Временные затраты на прохождение тест-кейсов в рамках дымового тестирования (примем 22 рабочих дня за месяц)	$2 \text{ минуты} * 15 * 22 = 660 \text{ минут}$	–
Временные затраты на написание 80 тест-кейсов (регрессионное тестирование)	2400 минут	2400 минут
Временные затраты на составление 80 автотестов (регрессионное тестирование)	–	2400 минут
Временные затраты на прохождение тест-кейсов в рамках регрессионного тестирования (два раза в месяц)	$2 \text{ минуты} * 80 * 2 = 320 \text{ минут}$	–

высокооплачиваемой, так что тратить ресурсы такого специалиста на регулярное прохождение тестов в рамках проведения регресса является нецелесообразным для руководителя ИТ-команды. Именно поэтому в высокотехнологичных современных компаниях, следующим принципам бережливого производства, и внедряется методика применения автотестирования, а освободившиеся людские ресурсы занимаются более высокотехнологичными операциями, а не рутинной, которая поддается автоматизации [6].

Заключение

Применение метода автоматизированного тестирования позволяет ускорить процесс проверки качества разработанного функционала, путем использования вычислительных ресурсов специализированных программ, благодаря чему специалист по тестированию получает возможность переключиться на более приоритетные задачи проекта, например, написание новых тест-кейсов, корректировка тест-плана, проверка нового функционала.

ЛИТЕРАТУРА

1. Автоматизированное тестирование программного обеспечения. Внедрение, управление и эксплуатация / Элфрид Дастин, Джефф Рэшка, Джон Пол — Лори — Москва, 2003. — 592 с.
2. Daniel Galin, "Software Testing," in Software Quality: Concepts and Practice, IEEE, 2018, pp.255–317, doi: 10.1002/9781119134527.ch14.
3. Тестирование программного обеспечения. Базовый курс / С.С. Куликов. — 3-е изд. — Минск: Четыре четверти, 2015–2020. — 312 с.
4. Вакансия: Тестировщик [Электронный ресурс]. URL: <https://russia.superjob.ru/vakansii/testirovschik.html> (дата обращения 19.06.2022)
5. Названа средняя зарплата в России. [Электронный ресурс]. URL: <https://lenta.ru/news/2022/04/26/sred/> (дата обращения 19.06.2022)
6. Agile-тестирование. Обучающий курс для всей команды / Джанет Грегори, Лайза Криппин — Манн, Иванов и Фербер — Москва, 2019. — 530 с.

© Латыпов Булат Фиделевич (linksmtv@gmail.com),

Зиязетдинов Тимур Рифович (ziyazetdinov.timur@gmail.com), Мухетдинов Амир Русланович (amir.muhetdinov@yandex.ru).

Журнал «Современная наука: актуальные проблемы теории и практики»