

Министерство образования Республики Беларусь
Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»

УДК 004.415.538

На правах рукописи

МИЩЕНКО
Антон Вадимович

**АЛГОРИТМЫ АВТОМАТИЗАЦИИ СИСТЕМНОГО ТЕСТИРОВАНИЯ
ИНФОРМАЦИОННЫХ РЕСУРСОВ**

АВТОРЕФЕРАТ

диссертации на соискание степени
магистра техники и технологий

по специальности 1-39 81 01 – Компьютерные технологии
проектирования электронных систем

Минск 2019

Работа выполнена на кафедре проектирования информационно-компьютерных систем учреждения образования «Белорусский государственный университет информатики и радиоэлектроники»

Научный руководитель: **ПИСКУН Геннадий Адамович**,
кандидат технических наук, доцент, доцент кафедры проектирования информационно-компьютерных систем учреждения образования «Белорусский государственный университет информатики и радиоэлектроники»

Рецензент: **РАЗУМЕЙЧИК Вита Станиславовна**,
кандидат технических наук, доцент кафедры электронных вычислительных машин и систем учреждения образования «Брестский государственный технический университет»

Защита диссертации состоится «26» июня 2019 года в 13³⁰ на заседании Государственной экзаменационной комиссии по защите магистерских диссертаций в учреждении образования «Белорусский государственный университет информатики и радиоэлектроники» по адресу: 220013, Минск, ул. П.Бровки, 6, корп. 1, ауд. 408, тел. 293-20-80, e-mail: kafpiks@bsuir.by

С диссертацией можно ознакомиться в библиотеке учреждения образования «Белорусский государственный университет информатики и радиоэлектроники».

ВВЕДЕНИЕ

Все информационные ресурсы и программные средства, выпускаемые в настоящее время так или иначе нуждаются в тестировании качества разработки и использования, для исключения вероятности возникновения критических для пользователя ошибок.

Как правило, тестирование программных продуктов происходит параллельно с их разработкой, при таком подходе цена ошибки разработчика намного меньше, так как намного больше вероятность нахождения её до выпуска готового продукта уже на этапе разработки или составления спецификации.

Традиционные методы разработки тестов вручную уже не могут обеспечить качественное тестирование современных программных систем. Очень часто время на тестирование растет пропорционально количеству добавляемой функциональности, при этом крайне важным становится провести валидацию исправления дефектов, функциональное тестирование новых функций и повторное (регрессионное) тестирование старых в короткий срок, до выхода конечного продукта. В таком случае время на ручное тестирование, значительно возрастает, также увеличивается со временем и количество тестовых сценариев. В таком случае целесообразно рассмотреть возможность автоматизации системного тестирования информационных ресурсов. Появляется все большее число методик автоматизации и инструментальных средств, направленных на повышение качества и сокращение ресурсозатрат на тестирование. Современные веб-приложения характеризуются быстрыми циклами разработки, соответственно появляется необходимость иметь возможность быстро и в срок протестировать выпускаемое приложение. Данную проблему помогает решить автоматизация тестирования, но при этом инженеры по тестированию, как правило, уделяют мало внимания качеству своих автоматизированных тестовых наборов, они быстро становятся трудно поддерживаемыми, в ходе того, как тестируемое приложение развивается.

Наиболее простые подходы к автоматизации системного тестирования информационных ресурсов используют идеи и инструментарий модульного тестирования. Однако в таком случае построение тестов связано с разработкой дополнительной библиотеки, не имеющей непосредственного отношения к тестируемому ресурсу. Кроме того, качество такого тестового набора и программной библиотеки для тестирования напрямую зависит от квалификации инженера и может содержать недостаточно перспективных сценариев.

В последнее время определенный прогресс в автоматизации тестирования достигнут благодаря появлению сторонних готовых библиотек для тести-

рования, на основе которых можно строить решение для конкретного информационного ресурса. Такие библиотеки позволяют унифицировать описание структуры и функциональных требований к системам, но при этом не дают полной гарантии, что они подойдут для абсолютно всех информационных ресурсов и все еще требуют от инженера по тестированию квалифицированных знаний в области автоматизации и языков программирования.

Однако существующие библиотеки для системного тестирования информационных ресурсов не позволяют существенно снизить трудоемкость разработки тестов. Саму модель системы, критерии тестового покрытия, наборы функциональных требований, средства, связывающие модель с реальной системой, приходится описывать вручную на языках программирования. Сложность такого описания сравнима с ручной записью тестов, качество тестирования при этом все еще существенно зависит от квалификации инженера по тестированию.

В существующих технологиях не учитывается тот факт, что повысить степень автоматизации при построении модели интерфейса программирования и генерации тестов можно за счет анализа уже существующей спецификации интерфейса информационных ресурсов.

Подходы к автоматизации тестирования различаются по своей сложности, а при внедрении автоматизации пропорционально возрастает цена разработки продукта, так как требует дополнительных работ, не связанных непосредственно с тестированием, а заключающихся в анализе тестируемого приложения, выборе подхода и способа тестирования и разработке дополнительной библиотеки программных методов (фреймворка), позволяющих взаимодействовать с тестовыми ресурсами. То есть, во многих случаях, когда автоматизация тестирования может сократить время на тестирование, но при этом существенно увеличить бюджет проекта, её внедрение не актуально. Проблема, создания такого программного интерфейса, который позволил бы быстро генерировать набор программных функций для автоматизации тестируемого ресурса, без затрат дополнительных ресурсов времени на разработку специфического фреймворка является одной из основных в области автоматизации тестирования.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы исследования

Автоматизация тестирования позволяет существенно сократить время на поиск и валидацию дефектов информационного ресурса, при этом требует больших затрат на разработку решения по автоматизации. Современные подходы и фреймворки, применяемые при автоматизации тестирования, все чаще

являются универсальными для того, чтобы их можно было применять для любого ресурса.

В связи с вышесказанным, актуальной задачей является поиск общего для всех информационных ресурсов решения, которое бы позволило автоматически генерировать программные методы для взаимодействия с тестируемым ресурсом с целью экономии времени, затрачиваемого на написание таких методов вручную.

Степень разработанности проблемы

В современных исследованиях, представленных в научно-технической литературе, приведены результаты, подтверждающие возможность использования автоматической генерации программных методов для взаимодействия с тестируемым ресурсом. Д. Жарий, С. Кузмин, *A. Stocco, M. Leotta, F. Ricco* в своих работах исследовали алгоритмы для обработки и автоматической генерации объектных классов страниц при тестировании информационных веб-ресурсов.

Одним из недостатков современных разработок, представленных в технической литературе является неполное рассмотрение возможности выбора стратегий поиска уникальных локаторов элементов на страницах тестируемого информационного ресурса, за счет чего чаще всего тесты создаваемые с помощью таких программ быстро устаревают и требуют дополнительных затрат на свою поддержку.

Предложенная разработка направлена на устранение данного недостатка, с помощью добавления в программный код стратегий обхода элементов, позволяющих автоматически находить более универсальные локаторы и таким образом сокращать время на поддержку автоматизированных тестовых сценариев.

Цель и задачи исследования

Целью диссертации является разработка программы для автоматической генерации классов объектных моделей страниц тестируемого ресурса и классов для взаимодействия со сгенерированными методами, в виде тестовых шагов сценария, на основе алгоритма оптимизации поиска уникальных локаторов элементов.

Поставленная цель работы определяет **следующие основные задачи:**

1. Провести обзор методов тестирования и алгоритмов автоматизации, а также программных продуктов для автоматической генерации классов тестируемого приложения, с обозначением их недостатков и преимуществ.

2. Разработать алгоритм оптимизации поиска локаторов элементов на основе стратегии обхода элементов в дереве объектной модели документа тестируемого ресурса и получения их атрибутов.

3. На основе разработанного алгоритма поиска локаторов элементов разработать программу для автоматической генерации классов тестируемого ресурса и взаимодействия с ним с возможностью выбора используемых локаторов.

Область исследования

Содержание диссертации соответствует образовательному стандарту высшего образования второй ступени (магистратуры) ОСВО 1-39 81 01-2012 специальности 1-39 81 01 «Компьютерные технологии проектирования электронных систем».

Теоретическая и методологическая основа исследования

В основу диссертации легли работы российских и зарубежных ученых в области автоматизированного тестирования и автоматической генерации программного кода для тестирования информационных ресурсов.

Информационная база исследования сформирована на основе литературы, открытой информации, сведений из электронных ресурсов, а также материалов научных конференций и семинаров.

Научная новизна

Научная новизна работы заключается в разработке алгоритма, позволяющего генерировать локаторы для взаимодействия с элементами тестируемого ресурса, с использованием их уникальных атрибутов, без привязки к их положению в дереве объектной модели исходного документа.

Теоретическая значимость работы заключается в детальном анализе существующих алгоритмов автоматизации системного тестирования информационных ресурсов с учетом их особенностей.

Практическая значимость диссертации состоит в разработанной программе для автоматической генерации классов тестируемого ресурса, которая позволяет на основе разработанного алгоритма оптимизировать процесс автоматизации информационных ресурсов.

Основные положения, выносимые на защиту

1. Обзор методов тестирования, алгоритмов автоматизации и программных продуктов для автоматической генерации классов тестируемого приложе-

ния, основанных на автоматическом распознавании локаторов элементов тестируемого ресурса, позволяющий выявить их основные недостатки и преимущества.

2. Алгоритм оптимизации поиска элементов, основанный на использовании стратегии нахождения элементов в дереве объектной модели документа тестируемого ресурса и получении их атрибутов, позволяющий автоматически распознавать объекты взаимодействия в исходном коде страницы.

3. Программный продукт для автоматической генерации классов тестируемого ресурса и взаимодействия с ним с возможностью выбора используемых локаторов, основанный на использовании разработанного алгоритма оптимизации поиска элементов, позволяющий сократить время на автоматизацию тестирования информационных ресурсов до 50%.

Апробация и внедрение результатов исследования

Результаты исследований по теме диссертации докладывались и обсуждались на II Международной научно-практической конференции «Молодёжный форум: прикладная математика. Математическое моделирование систем и механизмов» (г. Воронеж, Российская Федерация, 2019 г.), IV Международной открытой конференции «Современные проблемы анализа динамических систем. Приложения в технике и технологиях» (г. Воронеж, Российская Федерация, 2019 г.), 54-ой научной конференции аспирантов, магистрантов и студентов БГУИР (г. Минск, Республика Беларусь, 2018 г.), 55-ой научной конференции аспирантов, магистрантов и студентов БГУИР (г. Минск, Республика Беларусь, 2019 г.).

Публикации

Изложенные в диссертационной работе основные положения и выводы опубликованы в 8 печатных работах. В их числе 2 статьи в сборниках материалов научных конференций, 5 тезисов докладов на научных конференциях и 1 публикация в научном журнале.

Общий объем публикаций по теме диссертации составляет 19 страниц.

Структура и объем работы

Диссертация состоит из введения, общей характеристики работы, трех глав с краткими выводами по каждой главе, заключения, библиографического списка и приложений.

В первой главе проведен обзор методов тестирования, алгоритмов автоматизации и программных продуктов для автоматической генерации классов тестируемого приложения, основанных на автоматическом распознавании локаторов элементов тестируемого ресурса.

Установлено, что применяемые при автоматизированном тестировании методы и алгоритмы обладают рядом недостатков, в частности, время поиска локаторов и разработка программного кода для взаимодействия с элементами значительно превышает время на непосредственно тестирование и нуждается в постоянной поддержке, что также увеличивает время на проверку информационного ресурса.

Показано, что использование современных программных комплексов для автоматизации тестирования не позволяет автоматически распознавать локаторы элементов в исходном коде страницы.

Во второй главе представлен алгоритм оптимизации поиска элементов, основанный на использовании стратегии нахождения элементов в дереве объектной модели документа тестируемого ресурса и получении их атрибутов, позволяющий автоматически распознавать объекты взаимодействия в исходном коде страницы.

В третьей главе представлены программный продукт для автоматической генерации классов тестируемого ресурса и взаимодействия с ним с возможностью выбора используемых локаторов.

Также приведены результаты практического применения разработанного программного продукта, а также результаты эксперимента по сравнению разработанного алгоритма с аналогами и его результаты, показана эффективность разработанного алгоритма и его преимущества.

В приложении представлены публикации автора, акт внедрения, справка на антиплагиат и графический материал, иллюстрирующий основные результаты диссертационной работы.

Общий объем диссертации составляет 94 страницы. Из них 54 страницы основного текста, 15 иллюстраций на 3 страницах, 4 таблицы на 1 странице, 3 листинга кода на 1 странице, библиографический список из 72 наименований на 6 страницах, список собственных публикаций соискателя из 8 наименований на 1 странице, 4 приложения на 28 страницах.

ОСНОВНОЕ СОДЕРЖАНИЕ РАБОТЫ

Во **введении** рассмотрены современное состояние проблемы автоматизации тестирования, подходы для её решения и причины, по которым возникла необходимость автоматизировать тестирование информационных ресурсов, сформулированы причины, по которым возникает необходимость в сокращении времени, затрачиваемого на создание программных методов взаимодействия с тестируемым ресурсом и представлена автоматическая генерация кода как способ решения проблемы.

В общей характеристике работы показана актуальность проводимых исследований, степень разработанности проблемы, сформулированы цель и задачи диссертации, обозначена область исследований, научная (теоретическая и практическая) значимость исследований, а также апробация работы.

В первой главе приведен обзор методов и средств тестирования и подходов, которые могут быть применены для автоматизации информационных ресурсов. Рассмотрена пирамида тестирования и то место, которое занимает в ней тестирование информационных ресурсов. Исследованы алгоритмы тестирования методами «белого», «черного» ящика, структурное тестирование и основные алгоритмы тестирования потоков данных, доменов, метод разделения по категориям. Также рассмотрена проблема неизменяемого набора тестов, составленных на основе спецификации тестируемого ресурса и способ её решения при помощи модели.

Были приведены алгоритмы генерации тестов на основе теории графов в применении к модели тестируемого ресурса, а также инструментарий для автоматизации выполнения тестирования. При этом было отмечено, что в стандартных инструментариях для автоматизации выполнения тестирования, никак не автоматизируется процедура их выполнения. Приведено описание использования объектной модели страницы для тестирования информационных ресурсов и основные преимущества её применения. Сформулированы основные задачи автоматизации тестирования и способы их достижения, а также рассмотрены особенности тестирования пользовательского интерфейса и проблемы возникающего при автоматизации этого процесса, а также поставлена задача автоматизации тестирования пользовательского интерфейса информационных ресурсов.

Во второй главе приведена разработка и описание алгоритма для автоматической генерации локаторов элементов, определены необходимые требования к локаторам, которые должны автоматически формироваться при автоматической генерации объектных классов страниц. Рассмотрены средства автоматизации пользовательского интерфейса, которые предоставляют методы взаимодействия с элементами, но не предоставляют методов для автоматической генерации локаторов для поиска элемента на странице. При таком подходе в случае, если положение элемента на странице изменилось, тестовые сценарии не смогут взаимодействовать с элементами и будут помечены как неуспешные, хотя изменение положения элемента на странице, при сохранении его функциональности, как правило, не является ошибкой приложения.

Разработанный алгоритм позволяет на основе уникальных атрибутов элементов составить несколько локаторов. Например, при наличии *XPath* элемента в виде `//div[1]/ul[2]/a[3]`, алгоритм разбивает его на 3 части:

1. `div[1]`.

2. *ul*[2].

3. *a*[3].

Для каждого из элементов получают его уникальные атрибуты, предположим, что у элемента 1 присутствует атрибут *class='div_class'*, у элемента 2 атрибут *class='ul_class'*, у элемента 3 атрибут *href='https://www.integral.by'*. Как результат работы алгоритма, будет получено два локатора:

1. *//div[@class='div_class']//a[@href='https://www.integral.by']*.

2. *//ul[@class='ul_class']//a[@href='https://www.integral.by']*.

В случае если у элемента отсутствуют атрибуты, он будет проигнорирован, например, для предыдущего примера, если у элемента *div* отсутствует какой-либо уникальный атрибут, будет составлен только второй локатор.

Определены требования, которые должны реализовываться в программном комплексе, использующем представленный алгоритм.

В третьей главе представлена структурная схема программы, использующей разработанный алгоритм. На рисунке 1 видно, что взаимодействие между частями программы осуществляется через интерфейс, с помощью *JavaScript* функций, внедряемых в исходный код страницы отслеживаются действия пользователя с элементами, для выбранных элементов составляются локаторы в виде уникальных *XPath* выражений, после завершения всех действия генерируется код для взаимодействия с элементами на основе стандартных методов *Java* и *.vm* шаблонов.



Рисунок 1 – Структурная схема программы

На рисунке 2 представлен пользовательский интерфейс разработанной на основе структурной схемы и разработанного алгоритма программы. После

старта браузера необходимо начать поиск элементом путем нажатия кнопки *Start Search*. Это активирует возможность при взаимодействии с элементами при нажатии на правую кнопку мыши автоматически сформировать локатор для элемента и добавить его в объектную модель страницы. При завершении всех действия со страницей, нажимается кнопка *Add New Page* и *Generate Page Object Class For Page*.

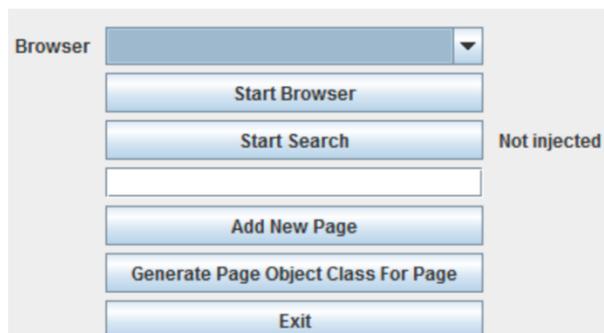


Рисунок 2 – Фрагмент пользовательского интерфейса программного продукта для автоматизированного тестирования информационных ресурсов

Это активирует окно выбора локатора элемента из всех сгенерированных вариантов. Окно выбора возможного локатора представлено на рисунке 3.



Рисунок 3 – Фрагмент диалогового окна «выбора локатора»

Также представлен эксперимент по подтверждению адекватности разработанной методики посредством автоматизации разработанного тестового сценария для информационного ресурса *www.integral.by*, приведены его результаты, в которых сравнивается затрачиваемое время и генерируемые локаторы при использовании различных алгоритмов.

ЗАКЛЮЧЕНИЕ

Основные научные результаты диссертации

1. Представлен обзор методов тестирования, алгоритмов автоматизации и программных продуктов для автоматической генерации классов тестируе-

мого приложения. Он основан на анализе автоматического распознавании локаторов элементов тестируемого ресурса и позволяет выявить их основные недостатки и преимущества. Проведенный анализ позволил сформулировать требования к алгоритму для оптимизации поиска элементов в дереве исходного документа информационных ресурсов.

2. Разработан алгоритм оптимизации поиска элементов, основанный на использовании стратегии нахождения элементов в дереве объектной модели документа тестируемого ресурса и получении их атрибутов. Разработанный алгоритм позволяет автоматически распознавать объекты взаимодействия в исходном коде страницы и формировать для них локаторы поиска, устраняя необходимость их поиска вручную. При помощи алгоритма формируется несколько локаторов на выбор пользователя, каждый из которых использует уникальные атрибуты элемента.

3. Разработан программный продукт для автоматической генерации классов тестируемого ресурса. Программный продукт позволяет взаимодействовать с тестируемым ресурсом с генерацией нескольких локаторов на выбор пользователя и основан на использовании разработанного алгоритма оптимизации поиска элементов. Использование программного продукта позволило сократить время на автоматизацию тестирования информационных ресурсов до 50%.

Рекомендации по практическому использованию результатов

Полученные результаты внедрены в учебный процесс на кафедре проектирования информационно-компьютерных систем учреждения образования «Белорусский государственный университет информатики и радиоэлектроники» в лекционный курс «Проектирование электронных модулей, устройств и систем».

СПИСОК ПУБЛИКАЦИЙ СОИСКАТЕЛЯ

Статьи в рецензируемых журналах

1. Мищенко, А.В. Тестирование промышленных программных систем с использованием пирамиды тестирования / А.В. Мищенко, А.В. Мельников // Актуальные направления научных исследований XXI века: теория и практика – 2019. – №1. – С.257 – 261.

2. Мищенко, А.В. Применение технологий контейнеризации для автоматизации тестирования API / А.В. Мищенко, А.В. Мельников // Актуальные направления научных исследований XXI века: теория и практика – 2019. – №6. – С.240 – 243.

3. Мищенко, А.В. Актуальные вопросы применения автоматизации тестирования информационных ресурсов / А.В. Мищенко // Международный

научный журнал «Вестник Науки и Творчества». – 2019. – № 6. – Принято в печать.

Тезисы конференций

4. Мищенко, А.В. Автоматизация тестирования мобильных приложений с использованием технологии компьютерного зрения / А.В. Мищенко, А.В. Мельников, А.П. Житко // Компьютерное проектирование и технология производства электронных систем: сборник тезисов 54 научной конференции аспирантов, магистрантов и студентов, Минск, 23–27 апреля 2018 г. / Белорусский государственный университет информатики и радиоэлектроники – Минск, 2018. – С. 77 – 78.

5. Мищенко, А.В., Мельников А.В. Фреймворк автоматизации тестирования пользовательского интерфейса / А.В. Мищенко, А.В. Мельников // Компьютерное проектирование и технология производства электронных систем: сборник тезисов 54 научной конференции аспирантов, магистрантов и студентов, Минск, 23–27 апреля 2018 г. / Белорусский государственный университет информатики и радиоэлектроники – Минск, 2018. – С. 79 – 80.

6. Житко А.П. Выбор языков программирования для реализации программного средства обучения ИТ-специалистов / А.П. Житко, Н.А. Шафар, А.В. Мищенко // Компьютерное проектирование и технология производства электронных систем: сборник тезисов 54 научной конференции аспирантов, магистрантов и студентов, Минск, 23–27 апреля 2018 г. / Белорусский государственный университет информатики и радиоэлектроники; отв. ред. Раднёнок А. Л. – Минск, 2018. – С. 67 – 68.

7. Мищенко, А.В., Мельников А.В. Анализ подходов к автоматизации тестирования / А.В. Мищенко, А.В. Мельников // Компьютерное проектирование и технология производства электронных систем: сборник тезисов 55 научной конференции аспирантов, магистрантов и студентов, Минск, 22–26 апреля 2019 г. / Белорусский государственный университет информатики и радиоэлектроники – Минск, 2019.

8. Мищенко, А.В., Мельников А.В. Подходы к автоматизации мобильных приложений / А.В. Мищенко, А.В. Мельников // Компьютерное проектирование и технология производства электронных систем: сборник тезисов 55 научной конференции аспирантов, магистрантов и студентов, Минск, 22–26 апреля 2019 г. / Белорусский государственный университет информатики и радиоэлектроники – Минск, 2019.

РЭЗІЮМЭ

Мішчанка Антон Вадзімавіч

Алгарытмы аўтаматызацыі сістэмнага тэставання інфармацыйных рэсурсаў

Ключавыя словы: аўтаматызаванае тэставанне, інфармацыйна-ныя рэсурсы.

Мэта працы: павышэнне ўстойлівасці аўтаматызаваных тэставых сцэнарыяў да зменаў тэставага рэсурсу за кошт аўтаматычнай генерацыі праграмных метадаў для ўзаемадзеяння з элементамі рэсурсу.

Атрыманыя вынікі і іх навізна: выкананы аналіз сутнасці-ваюць падыходаў да аўтаматызацыі тэставання інфармацыйных рэсурсаў. Выяўлена, што ў цяперашні час, як у айчынных, так і ў замежных крыніцах адсутнічае прыдатны для аўтаматычнай генерацыі пра-грамнага кода для ўзаемадзеяння з тэстоўваным рэсурсам, праграмны інтэрфейс; былі ўлічаны недахопы існуючых аналагаў распрацоўваецца-мага алгарытму; у выніку распрацоўкі і ўкаранення алгарытму ў праграммны інтэрфейс, эксперыментальна пацверджана палепшэнне якасці тэставых сцэнарыяў, што дазваляе аптымізаваць час на распрацоўку і тэставанне інфармацыйных рэсурсаў.

Ступень выкарыстання: вынікі ўкаранены ў навучальны працэс на кафедры праектавання інфармацыйна-камп'ютэрных сістэм ўстанова адукацыі «Беларускі дзяржаўны ўніверсітэт інфарматыкі і радыёэлектронікі» у лекцыйны курс «Праектаванне электронных модуляў, прылад і сістэм».

Вобласць ужывання: распрацоўка праграмнага забеспячэння, цесці-раванне праграмнага забеспячэння, аўтаматызацыя тэставання.

РЕЗЮМЕ

Мищенко Антон Вадимович

Алгоритмы автоматизации системного тестирования информационных ресурсов

Ключевые слова: автоматизированное тестирование, информационные ресурсы.

Цель работы: повышение устойчивости автоматизированных тестовых сценариев к изменениям тестируемого ресурса за счет автоматической генерации программных методов для взаимодействия с элементами ресурса.

Полученные результаты и их новизна: выполнен анализ существующих подходов к автоматизации тестирования информационных ресурсов. Выявлено, что в настоящее время, как в отечественных, так и в зарубежных источниках отсутствует подходящий для автоматической генерации программного кода для взаимодействия с тестируемым ресурсом, программный интерфейс; были учтены недостатки существующих аналогов разрабатываемого алгоритма; в результате разработки и внедрения алгоритма в программный интерфейс, экспериментально подтверждено улучшение качества тестовых сценариев, что позволяет оптимизировать время на разработку и тестирование информационных ресурсов.

Степень использования: результаты внедрены в учебный процесс на кафедре проектирования информационно-компьютерных систем учреждения образования «Белорусский государственный университет информатики и радиоэлектроники» в лекционный курс «Проектирование электронных модулей, устройств и систем».

Область применения: разработка программного обеспечения, тестирование программного обеспечения, автоматизация тестирования

SUMMARY

Mishchenko Anton Vadimavich

Algorithms for automation of system testing information resources

Keywords: automated testing, information resources.

The object of study: To increase the maintainability of automated test scenarios to changes in the tested resource due to the automatic generation of program methods for interacting with the resource elements.

The results and novelty: an analysis of existing approaches for test automation of information resources was carried out. It is revealed that at present, both in domestic and in foreign sources there is no software interface suitable for automatic generation of a program code for interaction with a tested resource; the drawbacks of the existing analogues of the developed algorithm were taken into account; as a result of the development and implementation of the algorithm in the program interface, experimentally confirmed improvement in the quality of automated test scenarios, which allows to optimize the time for developing and testing of the information resources.

Degree of use: the results are implemented in the educational process at the Department of Information and Computer-Aided Systems Design of the Education Institution “Belarusian State University of Informatics and Radioelectronics” in the lecture course «Design of electronic modules, devices and systems».

Sphere of application: software development, software testing, test automation.