

УДК 004.054

## ИССЛЕДОВАНИЕ И РЕАЛИЗАЦИЯ UI-ТЕСТИРОВАНИЯ ПЛАТФОРМЕННО-ЗАВИСИМЫХ ПРИЛОЖЕНИЙ НА ОСНОВЕ КЛИЕНТ-СЕРВЕРНОЙ АРХИТЕКТУРЫ

Деменев Т.Г. (ИТМО), Харитонов А.Е. (ИТМО)  
Научный руководитель – Харитонов А.Е. (ИТМО)

**Введение.** В наше время множество программ имеют пользовательский интерфейс (UI) для удобства взаимодействия с пользователями. Эти интерфейсы могут быть простыми, например, текстовыми, или сложными, с множеством элементов, имеющими различные состояния и демонстрирующими разнообразное поведение. Тестирование пользовательского интерфейса является непростой задачей, ввиду непредсказуемости действий пользователя. Сложность такого вида тестирования напрямую зависит от нагруженности и нестандартности UI. Необходимо проверить работу системы на различных платформах, чтобы гарантировать её качественную функциональность на различных устройствах с разными операционными системами.

**Основная часть.** На основании проведенных исследований можно выделить два подхода к тестированию пользовательского интерфейса: использование встроенных программных средств операционных систем и анализ работы приложения с помощью нейронных сетей, распознающих изображение.

Первый подход заключается в использовании готовых программных средств для тестирования интерфейса приложений, предоставляемых операционными системами [1]. Использование встроенных средств обеспечивает поддержку всех особенностей работы с оконными приложениями в данной ОС. Недостатком является специфичность этих инструментов для конкретной ОС и использование информации об устройстве интерфейса рабочего окна, как следствие набор возможностей и порядок использования будут сильно различаться. Элементы пользовательского интерфейса могут быть построены с использованием нестандартных технологий для конкретной ОС, что также затрудняет тестирование их с помощью встроенных средств.

Другой подход для тестирования пользовательского интерфейса заключается в использовании нейронных сетей, анализирующих происходящее на экране [2]. Преимущество данного подхода заключается в его универсальности — нейронные сети, подобно человеку, учитывают внешний вид элемента интерфейса, а не то, как он представлен в коде программы. Это же и является недостатком данного метода — подходящие для UI-тестирования нейронные сети требуют серьезных вычислительных мощностей, что не позволяет использовать их повсеместно.

Как альтернатива этим методам, был сформирован подход тестирования пользовательского интерфейса на основе архитектуры «клиент-сервер». Предложенный метод не накладывает никаких ограничений на реализацию клиента и сервера. Было разработано демонстрационное программное обеспечение для тестирования нестандартного приложения. Особенность тестируемого приложения заключалась в использовании собственных реализаций таких компонентов интерфейса как кнопки, флажки, выпадающие списки, что делало невозможным использование встроенных инструментов операционной системы для их тестирования. Суть метода в соединении тестируемого приложения с сервером и последующей отправке содержимого интерфейса в виде разметки. Это позволяет использовать локаторы для поиска элементов. Использование клиент-серверного подхода позволяет проводить параллельное тестирование интерфейса приложения на нескольких устройствах с разными аппаратными-программными компонентами. За счет свободы в реализации серверного приложения возможно добавить поддержку тестовых сценариев, записанных в формате JSON. Это может быть удобно при наличии тестовых сценариев, сформированных с помощью Selenium IDE, для веб-приложения, которое планируется перевести в формат

исполняемого значения.

**Выводы.** В рамках проделанной работы были рассмотрены существующие подходы к UI-тестированию Desktop-приложений, предложен новый метод для проведения тестирования платформенно-зависимых приложений и создана программа-сервер для проведения демонстрационного тестирования.

**Список использованных источников:**

1. Accessibility tools - Inspect [Электронный ресурс]. — URL: <https://learn.microsoft.com/en-us/windows/win32/winauto/inspect-objects> (дата обращения: 10.02.2024)

2. Learn GUI Testing Using Deep Learning [Электронный ресурс]. — URL: <https://www.nextgenerationautomation.com/post/learn-gui-testing-using-deep-learning> (дата обращения: 11.02.2024)