

Использование типовых решений при разработке фреймворка для тестирования

Мишота А.А., Ковалева И.Л.

Белорусский национальный технический университет

Современное состояние процесса разработки программного обеспечения (ПО) характеризуется значительным сокращением времени отдельных циклов разработки. В этих условиях автоматизация тестирования становится все более значимой задачей. Применение типовых решений при разработке фреймворков для тестирования позволяет в дальнейшем сократить время тестирования ПО.

Разработанный фреймворк для тестирования использует типовые элементы, которые видны пользователю: кнопки, формы, таблицы, дропдауны т. д. Типизация элементов выполняется на основании общепринятого представления, существующего в данный момент для конкретного элемента.

Например, для типового элемента «кнопка» – это представление о том, что такое кнопка и что она должна уметь, вне зависимости от конкретного интерфейса.

Главное – это та функциональность кнопки, по которой ее нельзя спутать с другим элементом, например, с таблицей или дропдауном.

Использование разработанного фреймворка исключает необходимость описания при тестировании нового ПО таких низкоуровневых действий, как поиск в таблице, проверка набора полей или выбор в меню. Все эти действия уже учтены во фреймворке. Элементы делают только то, что соответствует их функциональности, т. е. то, что от них ожидают.

Например, на кнопку можно нажать, но текст на ней написать нельзя. Для тестирования нового ПО дополнительно нужно описывать только те действия, которые связаны с конкретным бизнесом, заказчиком. Например, рассмотрим действие «купить продукт».

Типовые элементы не знают, что такое «купить продукт», и это нужно описывать для конкретных объектов.

Работа с простыми типовыми элементами (кнопками, текстом) позволяет сделать тесты более наглядными. А работа со сложными элементами (дропдаунами, таблицами, чек-листами) – дает возможность сократить количество кода, который нужно написать тестировщику.

Необходимая информация об элементах (локаторах) расположена в одном месте, и такой код проще поддерживать. Во фреймворке также реализовано логирование, что еще больше упрощает поддержку тестов.