

- 2) упрощение работы сотрудников, которые непосредственно ответственны за создание продукта и сотрудников, которые контролируют и управляют проектом;
- 3) организация системы контроля версий для файлов, которые участвуют в процессах;
- 4) организация взаимодействия и общения сотрудников.

Программное средство автоматизации рабочих процессов студии дизайна предусматривает выполнение основных функций:

- 1) создание и управление различными рабочими процессами;
- 2) наличие готового списка самых распространенных процессов;
- 3) наличие возможности добавления и изменения версий файлов, которые участвуют в процессах;
- 4) распределение ролей в рамках одного проекта (менеджеры и исполнители);
- 5) возможность контроля за состоянием проекта со стороны менеджеров;
- 6) возможность добавить комментарии и пометки к файлам, участвующим в процессе.

Разработанное программное средство позволяет студиям дизайна эффективно и качественно создавать различные информационные продукты. Благодаря автоматизации процессов сокращается время и соответственно цена конечного продукта, что выгодно выделяет данную компанию на фоне конкурентов.

Список использованных источников:

1. Рихтер Дж. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.5 на языке C#/ Рихтер Дж., Пер. с англ. – М.: Издательский дом «Вильямс», 2012. – 896 с.
2. Carstensen, P.H.; Schmidt, K. Computer supported cooperative work: new challenges to systems design/ Grudin, J. – Tokyo, 1999. – 19 с.

АВТОМАТИЗАЦИЯ ТЕСТИРОВАНИЯ ВЕБ-ПРИЛОЖЕНИЙ

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Хильчук А.С.

Куликов С.С. – канд. техн. наук, доцент

В настоящее время все больше появляется приложений, взаимодействующих с пользователем с помощью веб-интерфейса, за отображение которого отвечает веб-браузер. Как правило, страницы для браузера генерируются программным продуктом во время его работы. За продолжительное время использования в такие приложения добавляются новые функциональные возможности, изменяются существующие или даже наоборот – удаляются функции, которые уже не нужны. Такие изменения могут приводить к появлению нарушений в отображении страниц, а также к отказам самого ПО.

Тестирование ПО – это процесс исследования, испытания программного продукта, по результатам которого можно выявить ситуации, в которых поведение программы является неправильным, нежелательным или не соответствующим спецификации. [1]

Тестирование, направленное на то, чтобы удостовериться в отсутствии ошибок в ранее работоспособной функциональности, которые могли возникнуть в результате изменений в приложении или среде его функционирования, называется регрессионным. [2] Этот вид тестирования практически всегда присутствует в плане по обеспечению качества ПО, т.к. по статистике исправление одного дефекта с высокой вероятностью влечёт появление нового. Причинами появления одних и тех же ошибок могут быть сами разработчики, слабая техника управления системами контроля версий ПО, неактуальность исправления дефекта в последующих сборках и т.д.

Регрессионное тестирование может занимать до 90% от общего объёма времени, затрачиваемого на проверку очередной сборки ПО. Для снижения данных затрат внедряется автоматизация проведения тестирования. Автоматизированное тестирование ПО – это процесс верификации программного обеспечения, при котором основные функции и шаги теста, такие как запуск, инициализация, выполнение, анализ и выдача результата, выполняются автоматически, что помогает сократить время тестирования и упростить его процесс. [3]

Также с помощью автоматизированного тестирования можно осуществить нагрузочное тестирование, которое является актуальным для образовательных ресурсов. Данный вид тестирования направлен на исследование способности приложения сохранять заданные показатели качества при нагрузке в допустимых пределах и некотором превышении этих пределов. [2] Автоматизация его выполнения позволяет имитировать нагрузку на приложение посредством выполнения разного рода действий большого количества виртуальных пользователей.

Одним из наиболее популярных фреймворков для создания автоматизированных тестов веб-приложений является Selenium, использующий веб-драйвер для взаимодействия с браузером, который, в свою очередь, взаимодействует с самим веб-приложением. Данные веб-драйверы разрабатываются для браузеров Firefox, Internet Explorer и Safari, Opera, Microsoft Edge, а также для мобильных браузеров Android и iOS. Аналогично и с фреймворком – он доступен для языков Java, .Net (C#), Python, Ruby, JavaScript.

Частым архитектурным решением при проектировании автоматизированных тестов является паттерн PageObject. Основными его преимуществами является инкапсулирование работы с отдельными элементами веб-страницы, что позволяет уменьшить количество строк кода, а также его поддержку. Также к плюсам можно

отнести тот факт, что все действия с по работе с веб-страницей объединены в одном месте. При использовании данного паттерна, все описания элементов, а также методы взаимодействия с этими элементами описываются в PageObject классе тестируемой страницы. Методы взаимодействия, в свою очередь, вызываются из классов-тестов, описывающие порядок выполнения действий по взаимодействию с веб-страницей.

Плюсы автоматизации тестирования очевидны: быстрое выполнение, исключение «человеческого фактора», возможность высвобождения времени тестировщика, а также автоматическая генерация отчётов. Но существует и ряд минусов, основным из которых можно назвать трудоёмкость – несмотря на то, что автоматические тесты позволяют устранить ручное выполнение части однотипных операций и непосредственное выполнение тестирования, много затрат может приходиться на поддержку в актуальном состоянии самих тестов после изменения функциональности приложения.

Второй недостаток автоматизации – однотипность: все автоматизированные тесты выполняются строго по заложенному сценарию. При прохождении теста вручную сотрудник команды тестирования может обратить внимание на другие детали в приложении, изменить логику выполнения теста и обнаружить дефект, который был бы проигнорирован автоматическим тестом.

Следует помнить, что процесс автоматизации тестирования – это дополнение к тестированию, повышающее его эффективность за счёт уменьшения затрат. При постановке задачи внедрения автоматизированного тестирования в процесс создания образовательных ресурсов стоит учитывать возможность появления дополнительных затрат времени и иных ресурсов. Поэтому одной из важных задач подразделения тестирования является выбор степени автоматизации процесса тестирования программного продукта, в том числе и образовательного ресурса.

Таким образом, для веб-приложений, будь это электронные библиотеки, интернет-магазины или системы автоматизации производства, очень важна корректность работы, способность выдерживать большие нагрузки, соответствие заданным требованиям безопасности и т.д. Автоматизированное тестирование может значительно ускорить проведение тестов, которые позволяют удостовериться в том, что функциональность предыдущих версий всё корректно работает в текущей сборке приложения, а само приложение корректно работает на различных конфигурациях и окружениях. Так же автоматические тесты могут проводить проверки работы атомарных участков кода и взаимодействий между модулями приложения. Например, для систем управления предприятием, где несанкционированный доступ в систему недопустим, авто-тесты могут проводить различные проверки безопасности: прав доступа, открытых портов, уязвимостей в текущих версиях ПО и т.д. А автоматический тест, направленный на то, чтобы удостовериться что производительность приложения не падает при больших объемах данных, например, в базе данных, будет актуален для интернет-магазинов.

Список использованных источников:

1. Гленфорд Майерс, Том Баджетт, Кори Сандлер. Искусство тестирования программ, 3-е издание (TheArtofSoftwareTesting, 3rdEdition.) — М.: «Диалектика», 2012. — 272 с.
2. Святослав Куликов. Тестирование программного обеспечения. Базовый курс. — EPAM Systems, 2015-2016, 288 с.
3. Про Тестинг [Электронный ресурс] // protesting.ru : Тестирование Программного Обеспечения. URL: <http://www.protesting.ru/automation> (дата обращения: 15.01.2017).

АУТЕНТИФИКАЦИЯ ПО РАДУЖНОЙ ОБОЛОЧКЕ ГЛАЗА

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Чечет А.С.

Прохорчик Р.В. - м.т.н., ассистент

Существует несколько способов контроля предоставления доступа к информации. Одним из них является аутентификация. Аутентификация – это процесс, в ходе которого на основании пароля, ключа или какой-либо иной информации, пользователь подтверждает, что является именно тем, за кого себя выдает.

Для более подробного рассмотрения была выбрана аутентификация по биометрическим признакам, а именно по радужной оболочке. Это достаточно перспективное направление, поскольку для ее реализации необходима лишь обычная камера. Несколько лет назад фотоаппараты обладали камерами в 1-3Мп а сегодня мы можем позволить себе телефон с 20Мп. С течением времени качество камер будет усиливаться, что может позволить получать качественные изображения глаза даже на мобильном телефоне. Уже существуют телефоны с возможностью аутентификации по отпечатку пальца, например Apple iPhone 6 128Gb, Honor 7, Samsung Galaxy S7.