

АВТОМАТИЗИРОВАННОЕ ТЕСТИРОВАНИЕ ПРОГРАММНОГО СРЕДСТВА РАДИОИНФОРМАЦИОННЫХ СИСТЕМ

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

Саидмурадов С.Ш.

Ползунов В.В. - к.т.н., доцент,

При проектировании и разработке программного обеспечения, ключевую роль играет тестирование данного разрабатываемого средства. В данной работе рассмотрены основные проблемы автоматизированного тестирования и подходы к их решению. Были проанализированы характерные особенности и произведен сравнительный анализ фреймворков для автоматизированного тестирования, в частности для языка Ruby. Выявлено и обосновано использование автоматизированных тестов и поддержка их на протяжении всего цикла разработки программного средства. В результате проведенного анализа разрабатываются тестовые случаи и тестовые скрипты, которые позволяют проверить программное средство на наличие критических дефектов.

Тестирование программного обеспечения является важным процессом для оценки качества программного продукта. Тестирование представляет собой процесс анализа программного средства на обнаружение различий между фактическим и ожидаемым результатом. Тестирование программного обеспечения является деятельностью, которая должна производиться в течение всего жизненного цикла разработки программного обеспечения.

Основными целями тестирования можно назвать следующие:

- продемонстрировать разработчикам и заказчикам, что программа соответствует требованиям;
- выявить ситуации, в которых поведение программы является неправильным, нежелательным или не соответствующим спецификации.

Тестирование призвано выявлять дефекты, то есть обнаруживать слабые места в программном продукте для их дальнейшего удаления или исправления, однако нахождение подобных неисправностей может занять много времени и не поддается прогнозированию. Поэтому для упрощения процесса тестирования и сокращения временных затрат на этапе контроля качества в процессе разработки программного обеспечения часто применяют так называемое автоматизированное тестирование. Упрощение процесса тестирования с сокращением временных затрат достигается путем использования программных средств для выполнения тестов.

Как и у любого этапа разработки программного средства, у этапа тестирования есть свои характерные особенности, в частности среди проблем автоматизированного тестирования можно выделить поддержку тестов в актуальном состоянии. При изменении функционала или добавлении функционала потребуется обновление тестов для актуальных входных данных. Так же проблемы возникают уже при выборе тест кейсов, которые необходимо автоматизировать. Не все тест кейсы поддаются автоматизации, и бывают ситуации, когда некоторые из них лучше оставить в ручном тестировании. Очень редко удается перейти полностью от ручного тестирования к автоматизированному. Не стоит забывать, что ручное и автоматизированное тестирование – это не взаимоисключающие, а взаимодополняющие методы. Например, невозможно автоматизировать тестирование таких вещей как:

- Установка операционной системы
- Проверка напечатанного принтером документа
- Проверка содержимого картинки, видео

Цель: Написать автоматизированные тесты с использованием специализируемых фреймворков языка Ruby для обнаружения критических дефектов и сокращения временных затрат.

Для достижения цели нужно выполнить ряд задач:

1. Провести разбор и анализ имеющихся аналогов.
2. Разработать алгоритм тестирования.
3. Написать автоматизированные тесты.
4. Оценка результатов

В настоящее время широко используются фреймворки для автоматизированного тестирования.

PHPUnit - это ориентированный на программистов фреймворк тестирования для PHP. Он построен на архитектуре XUnit для фреймворков модульного тестирования.

JUnit – это Java фреймворк для тестирования, т. е. тестирования отдельных участков кода, например, методов или классов.

TestNG – это фреймворк с открытым кодом, где NG в TestNG означает Next Generation (Новое поколение). TestNG похож на JUnit, но намного функциональней него. Он разработан лучше, чем JUnit, особенно, если тестировать вложенные классы.

Разработка автоматизированных тестов выполняется с использованием объектно-ориентированного языка программирования Ruby, с использованием библиотек и фреймворков которые основываются на технологиях TDD и BDD. Используя библиотеку Сарубага и фреймворк RSpec, разрабатываются тесты, которые ориентированы на поведении конечного пользователя, это позволит смоделировать ситуации, при которых необходимо протестировать систему с точки зрения конечного пользователя, при этом выполнить все возможные тесты автоматизировано.

В ходе исследования будут получены теоретические и практические знания в областях экстремального программирования (Extreme Programming, XP), автоматизированного тестирования и обеспечения качества. Разработанные тесты позволят выявить дефекты в программном обеспечении, которые могут несущественно влиять на систему так и критические дефекты, связанные с безопасностью данных и потерей функциональности приложения. Данная разработка позволит проводить тестирование информационных систем более тщательно, с сокращением времени на тестирование системы в целом, что приведет к уменьшению затрат на обеспечение качества конечного продукта с высоким показателем стабильности системы.

Список использованных источников:

1. 12 лучших фреймворков автоматизированного тестирования PHP [Электронный ресурс]. – Режим доступа: http://www.internet-technologies.ru/articles/article_2668.html
2. Тестирование программы, JUnit [Электронный ресурс]. – Режим доступа: <http://java-online.ru/blog-junit.xhtml>
3. TestNG Framework [Электронный ресурс]. – Режим доступа: <http://autoqa.org/testng-framework/testng-framework.html>
4. Проблемы автоматизации тестирования и подходы к их решению [Электронный ресурс]. – Режим доступа: <http://cyberleninka.ru/article/n/problemy-avtomatizatsii-testirovaniya-i-podhody-k-ih-resheniyu>