

Выводы

Усовершенствованная математическая модель расчета экономической эффективности, основанная на индексе вероятности и коэффициентах дисконтирования позволяет рассчитать эффект для проектов ИАС, что позволяет ИТ-менеджеру принять правильное решение о внедрении ИАС на предприятии.

Предложенный сценарный подход к оценке различных вариантов расчета экономической эффективности проектов ИАС, позволяет выбрать наилучший вариант.

ЛИТЕРАТУРА:

1. Скрипкин К.Г. Экономическая эффективность информационных систем. – М.: АйТи, 2002. – 236 с.
2. Титоренко Г.А., Макарова Г.Л., Дайитбегов Д.М. и др. Информационные технологии в маркетинге. – М.: ЮНИТИ-ДАНА, 2000. – 335 с.
3. Компьютерные информационные системы управленческой деятельности: Учебн. пособие / Под ред. проф. Титоренко Г.А.; ВЗФЭИ. – М.: Экономическое образование, 1999. – 356 с.
4. Левыкин В.М., Костенко О.П. Оценка экономической эффективности информационно-аналитической системы // Праці Луганського відділення Міжнародної Академії інформатизації. – 2004. – № 1-2 (4-5). – С. 162-168.
5. Гайдаенко Т.А. Маркетинговое управление. Полный курс МВА. Принципы управленческих решений и российская практика. – 2-е изд., перераб. и доп. – М.: Эксмо, 2006. – 496 с.

© Шостак В.Ф., 2007.

© Костенко А.П., 2007.

© Лабзов С.Ю., 2007.

Шостак Владимир Федорович, д.т.н., проф.

Костенко Александр Петрович, доцент кафедры «Информационные технологии проектирования» Кременчугского университета экономики, информационных технологий и управления.

Лабзов Сергей Юрьевич – студент 4 курса специальности «Информационные технологии проектирования».

Кременчугский университет экономики, информационных технологий и управления.

УДК 621.315.592

МОДЕЛЬНООРИЕНТИРОВАННЫЙ ПОДХОД К ФУНКЦИОНАЛЬНОМУ ТЕСТИРОВАНИЮ WEB-БАЗИРОВАННЫХ ИНФОРМАЦИОННЫХ СИСТЕМ

В.М. Левыкин, М.О. Тихенко.

Харьковский национальный университет радиоэлектроники.

Предложен подход к функциональному тестированию Web-базированных информационных систем, в рамках которого осуществляется моделирование предъявляемых к системе требований и представление их в формализованном виде.

Ключевые слова: тестовые сценарии, качество ПО, функциональное тестирование, Web-базированные ИС, язык UML.

Введение

С появлением глобальной сети Интернет доступ к различным информационным ресурсам стал более простым и удобным. Более того, сама сеть стала объектом ведения бизнеса. Электронные магазины, рекламные порталы, поисковые системы, порталы по продаже недвижимости, электронные периодические издания – это те информационные системы (ИС), которые сегодня функционируют в Интернет. Такие системы получили название Web-базированных, или Web-базированных ИС. Преимуществом таких систем является то, что их пользователи получают унифицированный интерфейс, доступный на любой клиентской платформе. Для доступа к такой системе достаточно наличия подключения к сети Интернет и обычного браузера, встроенного в любую современную операционную систему.

Помимо Интернет-порталов, ориентированных на множество пользователей Интернет, другой важной нишей Web-базируемых ИС являются корпоративные ИС. Постепенно они вытесняют традиционные клиент-серверные системы с оконным интерфейсом благодаря своей легкой масштабируемости и простоте развертывания. Корпоративные ИС отличаются от Интернет-порталов наличием ограниченного набора пользователей и, как правило, более сложной бизнес-логикой.

При проектировании современных Web-базируемых ИС возникает ряд сложностей, связанных со спецификой их функционирования. Поэтому очень важным при разработке таких систем является применение моделей, методов и средств, позволяющих, с одной стороны, ускорить процесс разработки, а, с другой, повысить качество и прозрачность разработки.

Постановка задачи исследования

Проекты Web-базируемых информационных систем (ИС), как и других систем, являются крупномасштабными и имеют высокую степень сложности. Это обуславливает необходимость систематического тестирования систем, которое бы обеспечило соответствие получаемой в ходе разработки системы предъявляемым к ней функциональным требованиям. Традиционные подходы к тестированию не обеспечивают требуемое качество разрабатываемых информационных систем, а также приводят к большим трудовым и временным затратам. В этих условиях актуальным является создание такого подхода к тестированию, который бы обеспечивал качественное тестирование ИС и высокую степень автоматизации процесса разработки тестовых сценариев.

Особенности функционального тестирования Web-базируемых ИС

Функциональное тестирование Web-базируемых ИС имеет ряд особенностей, которые обусловлены спецификой данных систем.

Под Web-базируемой ИС будем понимать такую автоматизированную ИС, интерфейс которой реализован в виде динамически генерируемых HTML-страниц, доступ к которому осуществляется удаленно при помощи Web-браузера по протоколу передачи гипертекста HTTP. Основной информацией о действиях пользователя являются HTTP запросы и ответы, которые пользователь отправляет/получает на/от сервера Web-приложений, который обслуживает данную ИС. При этом взаимодействие клиента с сервером, обслуживающим систему, осуществляется через локальную или глобальную (Internet) сети. Современные Web-базируемые ИС ориентированы на интенсивную обработку данных: они характеризуются наличием больших объемов обрабатываемых данных и сложной бизнес-логикой, реализуемой на стороне сервера [1].

С точки зрения пользователя, Web-базируемая ИС состоит из совокупности динамических Web-страниц, с которыми он работает. Наполнение этих страниц может быть различным в зависимости от режима работы системы, прав пользователя и текущего наполнения БД системы.

Функциональные требования к системе формулируются заказчиком (пользователем), поэтому функциональное тестирование Web-систем рассматривается как метод контроля качества клиентского программного обеспечения (ПО). Для проверки соответствия полученной системы функциональным требованиям применяются методы тестирования «черного ящика» [2], в виду того, что вся бизнес-логика приложения реализована на сервере и пользователь получает результаты обработки своих запросов в виде Web-страниц. Функциональное тестирование Web-базируемых ИС заключается в проверке соответствия требованиям пользователей поведения системы при выполнении запросов к ней. Проведение данного тестирования традиционным способом приводит к большим затратам времени. На сегодняшний день существует ряд подходов к функциональному тестированию Web-систем, позволяющих автоматизировать данный процесс и сократить временные затраты.

Подходы к функциональному тестированию Web-базируемых ИС

Существующие подходы к функциональному тестированию Web-базируемых ИС, такие как Capture & Playback, Data Driven, KeywordDriven и другие, характеризуются наличием автоматизации процесса тестирования и отсутствием автоматизации процесса получения тестовых сценариев. В рамках данных подходов сценарии разрабатываются вручную, что приводит

к большим трудовым и временным затратам как во время разработки, так и во время адаптации тестовых сценариев в случае изменения требований к системе [3].

Самым распространенным является подход, называемый Capture & Playback (другие названия – Record & Playback, Capture & Replay). Суть этого подхода заключается в том, что сценарии тестирования создаются на основе работы пользователя с тестируемым приложением. Реализация такого подхода осуществляется инструментальным средством, которое обеспечивает перехват и запись действий пользователя. Результат каждого действия также запоминается и служит эталоном для последующих проверок. При этом в большинстве таких средств, реализующих этот подход, воздействия (например, нажатие кнопки мыши) связываются не с координатами текущего положения мыши, а с объектами HTML-интерфейса (кнопки, поля ввода и т.д.), на которые происходит воздействие, и их атрибутами. При тестировании инструментальное средство автоматически воспроизводит ранее записанные действия и сравнивает их результаты с эталонными, при этом точность сравнения может настраиваться. Можно также добавлять дополнительные проверки – задавать условия на свойства объектов (цвет, расположение, размер и т.д.) или на функциональность приложения (содержимое сообщения и т.д.). Кроме элементов интерфейса, такие инструментальные средства могут оперировать HTTP-запросами, последовательность которых также может записываться при работе пользователя, а затем модифицироваться и воспроизводиться.

Основное достоинство этого подхода – простота освоения. Вместе с тем, у подхода имеется ряд существенных недостатков. Например, для разработки тестов не предоставляется никакой автоматизации; фактически, инструментальное средство записывает процесс ручного тестирования. Если в процессе записи теста обнаружена ошибка, то в большинстве случаев создать тест для последующего использования невозможно, пока ошибка не будет исправлена (инструментальное средство должно запомнить правильный результат для проверки). При изменении тестируемого приложения набор тестов трудно поддерживать в актуальном состоянии, так как тесты для изменившихся частей приложения приходится записывать заново.

Описанный выше подход лучше всего использовать для создания прототипа теста, который впоследствии может служить основой для ручной доработки. Одна из возможных доработок – параметризация теста для проверки тестируемого приложения на различных данных (Data Driven [4]). Основным ограничением является то, что перебираемые данные не должны изменять поведение тестируемого приложения, поскольку проверки, записанные в тестовом сценарии, не подразумевают какой-либо анализ входных данных, т.е. для каждого варианта поведения системы нужно создавать свой сценарий тестирования со своим набором данных. Некоторые инструментальные средства, реализующие Capture & Playback, предоставляют возможность по перебору данных. Описанный подход основывается на построении тестов с использованием тестируемого приложения.

В подходе KeywordDriven предпринимается попытка сделать процесс создания тестов независимым от реализации. Суть подхода заключается в том, что действия, выполняемые в ходе тестирования, описываются в виде последовательности ключевых слов из специального словаря («нажать», «ввести», «проверить» и т.д.). Специальный компонент тестовой системы переводит эти слова в воздействия на элементы интерфейса тестируемого приложения. Таким образом, никакого программирования для создания тестов не требуется. Однако при изменении интерфейса, необходимо изменять компонент, который отвечает за перевод слов из «словаря» в последовательность воздействий на приложение. Комплект тестов может разрабатываться пользователями, не владеющими навыками программирования. В качестве примера инструментального средства, поддерживающего такой подход к разработке тестов, можно привести Certify от WorkSoft, в котором поддерживается библиотека функций для работы с каждым компонентом интерфейса (окна, гиперссылки, поля ввода и т.д.) и предоставляется язык воздействий на эти элементы (InputText, VerifyValue, VerifyProperty и т.д.).

Основные преимущества этого подхода заключаются в том, что он позволяет создавать тесты, не дожидаясь окончания разработки приложения, руководствуясь требованиями и дизайном интерфейса. Созданные тесты можно использовать как для автоматического выполнения, так и для ручного тестирования.

Основной недостаток этого подхода – отсутствие автоматизации процесса разработки тестов. В частности, все тестовые последовательности разрабатываются вручную, что приводит к проблемам, как на стадии разработки, так и на стадии сопровождения тестового набора. Эти проблемы особенно остро проявляются при тестировании Web-приложений со сложным интерфейсом.

Модельноориентированный подход к функциональному тестированию Web-базированных ИС

Учитывая сложность современных систем, актуальной проблемой является автоматизация подготовки тестовых сценариев на основании предъявляемых к системе требований. Сложность этого процесса состоит в том, что требования к ИС представляются в неформализованном виде: документов (каталоги требований и т.д.); знаний экспертов о предметной области (знаний аналитиков и проектировщиков). Поэтому для решения проблемы, необходимо описать требования к ИС в формализованном виде, что даст возможность сгенерировать тестовые сценарии, выполняющие проверку соответствия системы предъявляемым к ней требованиям.

На рис. 1 приведена схема процесса тестирования с использованием такого подхода.

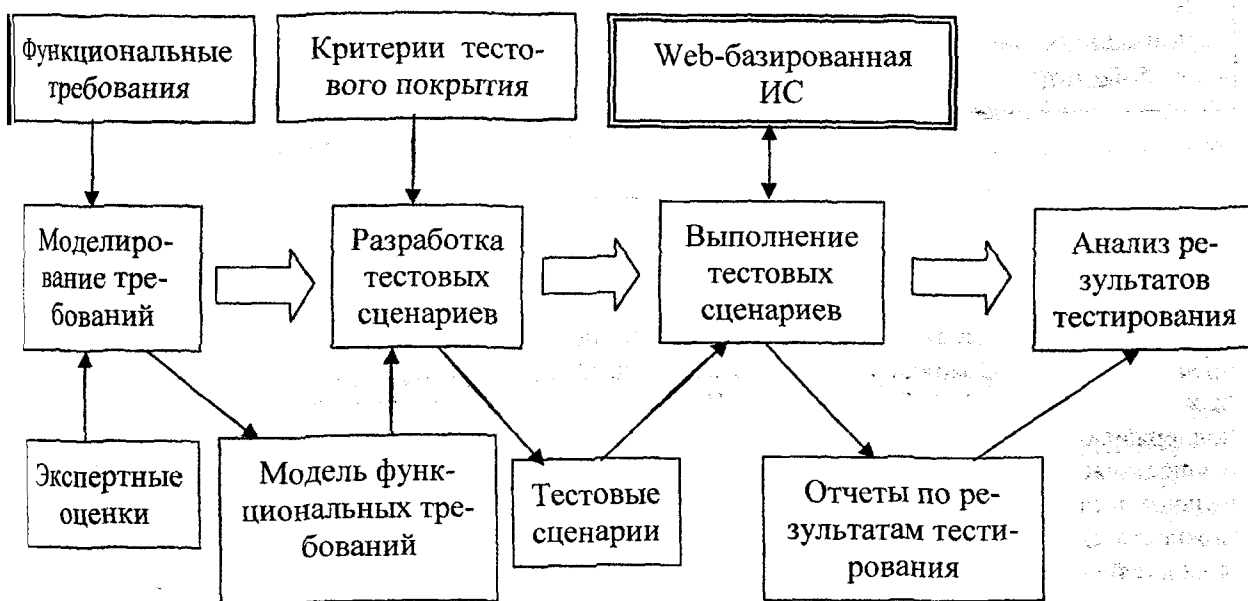


Рис. 1. Схема процесса тестирования

Процесс тестирования можно представить в виде следующих этапов:

- анализ функциональных требований, предъявляемых к системе;
- моделирование функциональных требований с использованием экспертных оценок;
- формирование критериев полноты тестирования;
- автоматизированная разработка тестовых сценариев;
- выполнение тестовых сценариев: осуществление воздействий на систему;
- формирование отчетов по результатам тестирования;
- анализ результатов тестирования.

В рамках данного процесса особую сложность представляют задачи формализованного представления требований к системе, а также механизмы автоматизированной разработки тестовых сценариев на основании модели требований к системе с учетом критериев тестового покрытия.

На основании проведенных исследований был сделан вывод о том, что построение модели формирования функциональных требований целесообразно осуществлять с использованием языка UML. Это обусловлено тем, что данный язык поддерживается многими инструментальными средствами, а также признан мировым сообществом де-факто при проектировании ИС

различных классов. UML напрямую не предназначен для моделирования Web-базированных ИС, однако для этих целей могут использоваться его расширения.

Так как моделирование будет осуществляться с использованием расширений языка UML, необходимо разработать инструментальное средство, которое бы обеспечивало механизмы получения тестовых сценариев на основании спроектированной модели функциональных требований с учетом критериев тестового покрытия. Архитектура такого инструментального средства представлена на рис. 2.

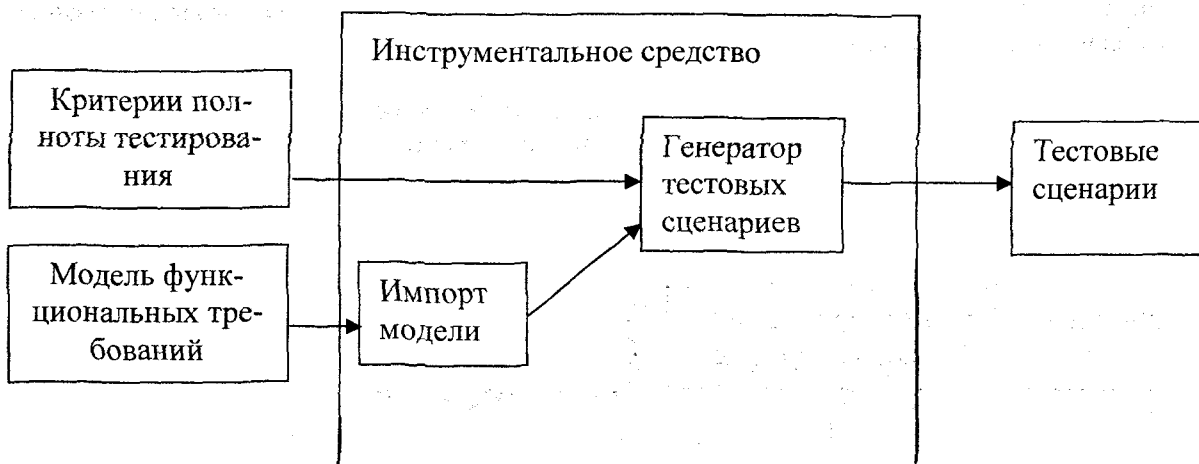


Рис. 2. Архитектура инструментального средства для предлагаемого подхода

Выводы

Таким образом, в предлагаемом модельноориентированном подходе на ранних стадиях проектирования ИС может использоваться язык UML для построения модели функциональных требований к Web-базированной ИС. С использованием инструментального средства, поддерживающего переход от модели функциональных требований к тестовым сценариям, осуществляется автоматизированная генерация сценариев, позволяющих выполнить функциональное тестирование разрабатываемой системы. Анализ результатов тестирования дает возможность сделать вывод о соответствии разработанной Web-базированной ИС предъявляемым к ней функциональным требованиям.

ЛІТЕРАТУРА:

1. Стотлемайер Диан. Тестирование Web-приложений. Пер. с англ. – М.: Издательский дом «Кудиц-Образ», 2003. – 240с.
2. Борис Бейзер. Тестирование черного ящика. – СПб.: Издательский дом «Питер», 2004. – 317с.
3. Элфрид Дастин, Джефф Рэшка, Джон Пол. Автоматизированное тестирование программного обеспечения. – М.: Издательство «Лори», 2003. – 567с.
4. Keith Zambelich. Totally Data-Driven Automated Testing.
URL: http://www.sqa-test.com/w_paper1.html

© Левыкин В.М., 2007.

© Тихенко М.О., 2007.

Левыкин Виктор Макарович, д.т.н., проф., зав. кафедрой Информационных управляющих систем Харьковского национального университета радиоэлектроники, сл. тел. 70-21-451, iyc@kture.kharkov.ua

Тихенко Максим Олегович, магистрант группы ИУСТм-06-1 Харьковского национального университета радиоэлектроники.

ваний на порівнянні й виборі найкращого варіанту.

Табл. 1. Лл. 3. Бібліогр.: 5 найм.

УДК 621.315.592:669.791

Модельноорієнтований підхід до функціонального тестування web-базованих інформаційних систем / В.М. Левикін, М.О. Тихенко // Нові технології. – 2007. - № 1-2 (15-16). – С. 82.

В роботі розглянуті проблеми й особливості функціонального тестування сучасних Web-базованих інформаційних систем та існуючі підходи до тестування таких систем. З урахуванням недоліків існуючих підходів запропоновано підхід, що має в основі моделювання вимог до Web-базованих інформаційних систем. Цей підхід складається з етапів: моделювання вимог до системи, автоматизована генерація тестових сценаріїв за допомогою інструментального засобу, виконання тестових сценаріїв, генерація звітів по результатам тестування та їх аналіз.

Лл. 2. Бібліогр.: 4 найм.

УДК 681.3.06; 004.3

Система тестування взаємодії WEB та SQL-серверів / Р.І. Макачук, В.Я. Ляшкевич, О.Я. Олар, С.В. Мельничук // Нові технології. – 2007. - № 1-2 (15-16). – С. 87.

Розглянуті фактори, які впливають на середній час обробки запиту та потоку запитів. Проведений аналіз існуючих способів розташування серверів в мережі. Запропонована система тестування процесу взаємодії Web та SQL-серверів. За допомогою системи проведено тестування серверів Firebird та MySQL. Результати тестування наведені на рисунках у вигляді графіків та екранних зображень.

Лл. 11. Бібліогр.: 6 найм.

УДК 621.8

Виявлення спама на основі алгоритмів штучного імунітету / О.І. Михальов, Ю.О. Каліберда // Нові технології. – 2007. - № 1-2 (15-16). – С. 92.

У статті запропонований новий підхід щодо виявлення й фільтрації спама на основі одного з алгоритмів штучного імунітету. Відмінною рисою запропонованого алгоритму є спосіб побудови детекторів, що складаються з сигнатур, що однозначно визначає семантику листа.

Лл. 2. Бібліогр.: 4 найм.

УДК 004.056.53:347.211

Інформаційна безпека як фактор захисту інтелектуальної власності в кіберпросторі / В.М. Левикін, Т.В. Гавриш // Нові технології. – 2007. - № 1-2 (15-16). – С. 95.

Розглянуто проблему охорони інтелектуальної власності в кіберпросторі в контексті інформаційної безпеки електронного бізнесу й електронної комерції. Показано невідповідність існуючих законодавчих норм в галузі захисту програмних продуктів і баз даних як об'єктів інтелектуальної власності сучасним технологіям розробки програмних засобів і рівня творчого начала у формуванні баз даних. Визначено тенденції в диференціації підходів до правового захисту комерційних і некомерційних програмних продуктів, творчих і нетворчих баз даних, обґрунтована доцільність захисту інтелектуальних

UDC 621.315.592:669.791

Model-based approach to web-based information systems functional testing / V.M. Levykin, M.A. Tykhenko // New Technologies. – 2007. - № 1-2 (15-16). – P. 82.

Problems, particular qualities of Web-based information systems functional testing and existing approaches to such systems testing are considered in this work. Taking into account disadvantages of existing approaches, an approach based on a model of Web-based information systems functional requirements is suggested. This approach consists of such stages: modeling system requirements, automated test case generation with a help of special tool, test case execution, report generation and analysis.

Fig. 2. Ref.: 4 items.

UDC 681.3.06; 004.3

System of testing of the WEB and SQL-servers interaction / R.I. Makarchuk, V.Ja. Lyashkevich, O.Ja. Olar, S.V. Melnichuk // New Technologies. – 2007. - № 1-2 (15-16). – P. 87.

Factors which affect mean time of treatment of a query and stream of queries are considered. The analysis of the existing methods of servers location in the network is performed. The system to test the process of the Web and SQL-servers interaction is offered. By the system proposed Firebird and MySQL servers have been tested. The results of testing are shown in figures in the form of graphs and screen shots.

Fig. 11. Ref.: 6 items.

UDC 621.8

spam detecting on the basis algorithms of artificial immunity / O.I. Mikhalev, Yu.O. Kaliberda // New Technologies. – 2007. - № 1-2 (15-16). – P. 92.

In article the new approach for spam detect and filtration on the basis algorithms of artificial immunity. Distinctive feature of the offered algorithm is the way construction of the detectors, consisting of the signatures, unequivocally defining semantics of the letters.

Fig. 2. Ref.: 4 items.

UDC 004.056.53:347.211

Information security as factor of protection of intellectual property in cyberspace / V.M. Levikin, T.V. Gavrish // New Technologies. – 2007. - № 1-2 (15-16). – P. 95.

The problem of protection of intellectual property in cyberspace in the context of information security of electronic business and electronic commerce has been researched. There has been shown lagging behind of the present legislative basis in the sphere of protection of program products and database as objects of intellectual property from modern technologies of working out of program means and the level of creative work in forming database. The tendencies in differentiation of approaches and legislative protection of commercial and non-commercial program products, creative and non-creative database have been worked out. The necessity to protect