

УДК 004.415.53

В.В. Гаркин, Р.Н. Чен

Харьковский национальный экономический университет, Харьков

## РАЗРАБОТКА И ИССЛЕДОВАНИЕ МЕТОДИКИ НАГРУЗОЧНОЙ И ФУНКЦИОНАЛЬНОЙ ОЦЕНКИ КАЧЕСТВА ИНФОРМАЦИОННЫХ СИСТЕМ

*Качество информационных систем в настоящее время является первостепенным условием продвижения программного продукта на информационном рынке. Одним из эффективных методов определения критериев качества, таких как: результативность и эффективность, являются методы нагрузочных и функциональных испытаний. Рассмотрены вопросы разработки и исследования методики нагрузочной и функциональной оценки веб-сервиса факультета экономической информатики ХНЭУ.*

**Ключевые слова:** информационная система, программный продукт, методика нагрузочной оценки, функциональная оценка, качество ИС.

### Введение

Для информационных систем, которым предстоит работа с большими объемами данных, наиболее эффективным методом определения критериев качества, таких как результативность и эффективность<sup>1</sup>, является метод нагрузочных испытаний, потому что чаще всего система показывает хорошую производительность на данных небольшого объема, но с ростом загрузки возникают разного рода проблемы, которые могут быть связаны с неоптимальным построением запросов, технологическими конфликтами различных элементов системы, ограничениями системных ресурсов и т.д.

Если подобные проблемы обнаруживаются уже в ходе эксплуатации, в условиях реальной пиковой нагрузки, результатом становится резкое снижение быстродействия или даже остановка системы. На поиск причин сбоя и их устранение может уйти несколько дней, что для онлайн-приложений обычно приводит к потерям, несопоставимым со стоимостью затрат на проведение испытаний [4]. Кроме того, помимо оценок производительности и отказоустойчивости работы системы под нагрузкой, целями проведения нагрузочных испытаний может быть также сравнение эффективности работы различных систем или анализ быстродействия работы на разном оборудовании или в различном окружении.

### Постановка задачи и цель работы

Современная индустрия создания программного обеспечения информационных систем основана на моделях систем эффективного менеджмента [8], таких как:

<sup>1</sup> Результативность (Effectiveness) – степень реализации запланированной деятельности и достижения запланированных результатов (ISO 9001:2009)

Эффективность (Efficiency) – связь между достигнутым результатом и использованными ресурсами (ISO 9001:2009) [5]

The TickIT Guide – «Использование ISO 9001:2000 для построения систем менеджмента качества программных продуктов, сертификации и непрерывного улучшения» [2].

ISO/IEC 12207 – «Информационные технологии – Процессы жизненного цикла программ» [1].

CMM-SW (Capability Maturity Model for Software) – разработана специалистами института SEI (Software Engineering Institute) университета Карнеги Мелон при поддержке министерства обороны США [2].

Для оценки качества ИС [9] в соответствии со стандартом ISO 9001:2009 была выбрана информационная система факультета экономической информатики Харьковского национального экономического университета, которая представляет веб-сайт, размещенный по адресу: <http://www.hneu.edu.ua>.

Данная информационная система характеризуется обработкой больших объемов данных, к которым одновременно в режиме онлайн может обращаться достаточно большое число пользователей и в данном случае становится важной оценка нагрузочного режима работы сайта.

Анализ исследований и публикаций в области оценки веб-сервисов показал, что рассмотрение таких аспектов тестирования, как: нагрузочное и функциональное значительный практический интерес [6].

**Целью данной работы** является разработка и исследование нагрузочного и функционального оценок для проверки качества веб-сайта.

### Разработка и исследование методик нагрузочной и функциональной оценок

В любом проекте по нагрузочному тестированию выделяются следующие самостоятельные задачи [8]:

Анализ, проектирование и реализация модели нагрузки.

Подготовка тестовых баз данных.  
Разработка типовых сценариев.  
Реализация модели нагрузки.  
Проведение нагрузочных испытаний.  
Анализ результатов испытаний.

На этапе подготовки и проведения исследований, особенно на базах большого объема, не обойтись без специализированных средств автоматизированного тестирования и мониторинга, которые позволяют моделировать загрузку и конфигурацию системы, сценарии работы пользователей, запускать сценарии в отложенном режиме (в ночное время) и т.д. Поэтому в данной работе использованы стандартные процедуры автоматизированного тестирования [3], для которых автором написаны специальные нагрузочные скрипты для проведения тестирования выбранной информационной системы (<http://www.ei.ksue.edu.ua>).

Целью и результатом решения задачи «Анализ, проектирование и реализация модели нагрузки» является моделирование типовых процессов, выполняемых пользователями. Эта задача является наиболее трудоемкой и занимает до 50% затрат всего проекта. В результате проведения нагрузочного тестирования определяются слабые места на серверном оборудовании, или программном обеспечении, кроме того, появляется возможность оптимизации самого интернет ресурса, в данном случае сайта факультета экономической информатики.

Для построения модели нагрузки необходимо описать типовые сценарии работы пользователей (типовые наборы операций) и выделить группы пользователей, работающих с системой. Каждый сценарий работы пользователей включает один или несколько автоматизированных скриптов, которые в ходе выполнения эмулируют работу пользователя. Для каждого типового сценария работы пользователей выделяются наборы данных, с которыми выполняется этот сценарий и измеряемые параметры производительности тестируемого приложения. Дополнительно идентифицируются измеряемые параметры производительности оборудования.

Реализация модели нагрузки заключается в разработке тестовых скриптов, реализующих отдельные бизнес-операции. При создании нагрузочного скрипта исключались http-запросы на загрузку файлов с расширениями:

- \*.jpg
- \*.gif
- \*.png
- \*.ico
- \*.jpeg
- \*.css
- \*.js,

так как загрузка этих файлов очень сильно загружает сетевой канал и не позволяет достоверно

проверить максимально возможную нагрузку на сервер. Разработанные скрипты должны эмулировать работу без привлечения уровня пользовательского интерфейса приложения – это необходимо для возможности параллельной эмуляции работы большого количества пользователей на одном компьютере.

В результате, модель нагрузки представляет собой расписание выполнения (Test Suite, Schedule), содержащее группы пользователей, распределение групп пользователей по компьютерам в момент выполнения и сценарии работы пользователей. Для записи и последующего воспроизведения на этапе выполнения автоматизированных скриптов используются специализированные инструментальные средства.

Для определения графика зависимости производительности системы от пользовательской нагрузки и от нагрузки по данным (так называемых профилей производительности), необходимо создать набор тестовых баз данных, которые имеют разный объем (мощность), но совпадают по структуре [7]. Целесообразно формировать такие наборы тестовых данных, чтобы число записей в них отличалось в разы, например, 2000, 4000, 8000 и т.д. Это позволяет оценить изменение параметров производительности тестируемого программного обеспечения и определить законы изменения измеряемых параметров в зависимости от нагрузки.

Логарифмическая шкала позволяет оценить скорость роста времен отклика и обнаружить узкие места при росте пользовательской нагрузки.

График пользовательской нагрузки на сайт для проведения нагрузочного теста показан на рис. 1.

Все виртуальные пользователи при прохождении нагрузочного тестирования выполняли 3 сценария в заданном порядке:

Прохождение авторизации и аутентификации на сайте

Навигация по первым 5 страницам сайта

Поиск по ключевому слову

Весь процесс нагрузочного тестирования заключался в отправке http – запросов и получение http – ответов.

Суммарная длительность нагрузочного тестирования составила 1 час, а общее количество виртуальных пользователей – 500, первые 10 минут количество пользователей линейно увеличивалось до максимума, затем 40 минут проходила полная нагрузка на сайт, и последние 10 минут количество пользователей линейно уменьшалось (рис. 1).

Проведение испытаний заключается в выполнении тестовых пакетов с определенным количеством пользователей. По результатам данного этапа необходимо получить значения измеряемых параметров при заданных конфигурациях и заданной

нагрузке. Чаще всего измеряемыми параметрами являются время отклика системы и процент загрузки системных ресурсов (чаще всего оперативная память и процессор).

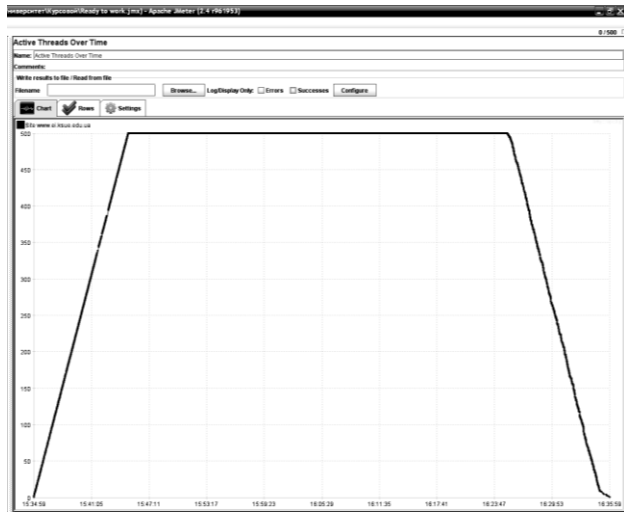


Рис. 1. График реального количества пользователей для проведения нагрузочного теста

Данные испытаний выводятся в таблицу значений измеряемых параметров:

Общее количество запросов (#Samples).

Среднее время выполнения (Average).

Время выполнения в середине теста (Median).

Время выполнения запроса на 90% выполнения теста (90% Line).

Минимальное значение времени выполнения запроса (Min).

Максимальное значение времени выполнения запроса (Max).

Процент ошибок (Error %).

Суммарное время выполнения всех запросов со средним временем (Throughput).

Скорость передачи данных (KB/sec).

В ходе каждого цикла исполнения модели определяются средние значения и отклонения от средних значений для измеряемых параметров.

Таким образом, по итогам серии испытаний, получают графики зависимости измеряемых параметров при изменении нагрузки (числа пользователей) и неизменной конфигурации системы. На графиках показываются зависимости времени отклика сервера и загрузки системы на определенные запросы (пользовательской нагрузки).

Основными результатами проведения нагрузочных испытаний является расчет показателей производительности такие, как время отклика на запрос и загруженность системных ресурсов сервера в зависимости от загрузки (числа подключенных пользователей). Эти результаты обычно представляются в виде сравнительных отчетов по времени

выполнения различных функций программного обеспечения при одинаковой нагрузке, при использовании различных программно-аппаратных конфигураций, при работе с различными объемами данных и т.д.

Наиболее распространенные виды отчетов о результатах нагрузочных испытаний [3]:

Response vs Time (resources) – отклики по времени, загрузка ресурсов;

Performance – производительность, время отклика;

Compare Performance – сравнение производительностей.

В отчет о тестировании обычно также включается перечень “узких мест” программной системы с указанием причин.

В результате нагрузочного тестирования сайта ХНЭУ по адресу <http://www.hneu.edu.ua> выявлены следующие типы ошибок, которые возвращает сервер, когда не успевает справиться с нагрузкой:

код ответа 200, который подтверждает удачное прохождение запроса и получение ответа;

код ответа 503 сообщает об ошибке на сервере.

Указанным способом определяются так называемые первичные результаты, т.е. результаты, определяемые непосредственно из испытаний. На их основе могут быть построены прогнозные модели, позволяющие оценить производительность тестируемого ПО на других конфигурациях и при другой нагрузке. Полученные таким образом результаты, относятся к классу вторичных, или аналитических результатов испытаний.

Если результаты тестирования подтвердили необходимость увеличения мощности системных ресурсов, то фактически встает вопрос о выборе программно-аппаратной платформы, на которой ПО будет поддерживать параллельную работу заданного количества пользователей. В этом случае, возможно, потребуется проведение серии испытаний системы на различных платформах.

Функциональное тестирование занимает ключевую позицию в процессе обеспечения контроля качества разрабатываемых информационных систем [3]. Одним из самых популярных методов оценки качества является автоматизированное функциональное тестирование.

Качественная всесторонняя проверка функциональности систем включает в себя проверку качества информационной системы на основе трех ключевых принципов:

соответствие конечной реализации предъявляемым к системе функциональным требованиям;

защита системы от некорректного использования;

удобство использования и соответствие мировым стандартам разработки.

Для разработки и исследования методики функциональной оценки качества информационных систем был разработан программный продукт, (фрагмент кода программы, класс Main показан ниже) который реализует функционал автоматизированного функционального тестирования приложения.

```
//USEUNIT Excel
//USEUNIT CommonFunc
//=MAIN FUNCTION =//
function Main()
{
    try
    {
        UserForms.MainForm.ShowModal();
        a = UserForms.MainForm.ModalResult;
        if(a != 3)
        {
            if(UserForms.MainForm.logintest.Checked ==
true)
            {
                mainlogintest();
            }
        }
        if(UserForms.MainForm.MainNavigateByPages.Checked == true)
        {
            fNavigateByPages();
        }
        if(UserForms.MainForm.LeaveAndDeleteComment.Ch
```

```

ecked == true)
        {
            fLeaveAndDeleteComment();
        }
    }
    if(UserForms.MainForm.SearchFunctionality.Checked == true)
    {
        fSearchFunctionality();
    }
}
else
{
    Log.Error("Run tests is stop", exception.description);
}
}
catch(exception)
{
}
}
WriteDataToExcel();
}
}
```

В результате работы теста генерируется отчет в виде Excel-файла, в котором находятся результаты работы теста по проверки функциональности ИС, т.е. веб-сайта факультета экономической информатики (рис. 2, фрагмент).

По результатам работы программы можно сделать вывод, что все необходимые шаги теста были выполнены успешно и без ошибок.

| #  | A                              | B      | C  | D | E | F |
|----|--------------------------------|--------|--|---|---|---|
| 1  | #                              | Result | Discription                                    |   |   |   |
| 2  | #####                          |        |  |   |   |   |
| 3  | FUNCTION LOGIN TESTING         |        |  |   |   |   |
| 4  | 1                              | Pass   | Link to main page is exists                    |   |   |   |
| 5  | #####                          |        |  |   |   |   |
| 6  | NAVIGATION BY PAGES            |        |  |   |   |   |
| 7  | 1                              | Pass   | Link to main page is exists                    |   |   |   |
| 8  | 0                              | Pass   | Navigate to 0 page is successful               |   |   |   |
| 9  | 1                              | Pass   | Navigate to 1 page is successful               |   |   |   |
| 10 | 2                              | Pass   | Navigate to 2 page is successful               |   |   |   |
| 11 | 3                              | Pass   | Navigate to 3 page is successful               |   |   |   |
| 12 | 4                              | Pass   | Navigate to 4 page is successful               |   |   |   |
| 13 | #####                          |        |  |   |   |   |
| 14 | LEAVE AND DELETE THE COMMENT   |        |  |   |   |   |
| 15 | 1                              | Pass   | Link to main page is exists                    |   |   |   |
| 16 | #####                          |        |  |   |   |   |
| 17 | NAVIGATION TO FIRST POST       |        |  |   |   |   |
| 18 | 1                              | Pass   | Link to main page is exists                    |   |   |   |
| 19 | 1                              | Pass   | The link [Add comment] is exist and clicked    |   |   |   |
| 20 | #####                          |        |  |   |   |   |
| 21 | POST THE COMMENT AND DELETE IT |        |  |   |   |   |
| 22 | 0                              | Pass   | The text area for comment is exist and clicked |   |   |   |
| 23 | 1                              | Pass   | The text area for comment is not exist         |   |   |   |
| 24 | 2                              | Pass   | The button for post comment is exist           |   |   |   |
| 25 | 3                              | Pass   | Link for delete comment is present             |   |   |   |
| 26 | 4                              | Pass   | Button for complete the deleting is present    |   |   |   |
| 27 | #####                          |        |  |   |   |   |
| 28 | SEARCH FUNCTIONALITY TESTING   |        |  |   |   |   |
| 29 | 1                              | Pass   | Link to main page is exists                    |   |   |   |
| 30 | 1                              | Pass   | Textbox for search work is exists              |   |   |   |
| 31 | 2                              | Pass   | Text has entered to textbox                    |   |   |   |
| 32 | 3                              | Pass   | Link for search is present                     |   |   |   |
| 33 | 4                              | Pass   | Link for the 0 page is present                 |   |   |   |
| 34 | 4                              | Pass   | Link for the 1 page is present                 |   |   |   |
| 35 | 4                              | Pass   | Link for the 2 page is present                 |   |   |   |

Рис. 2. Фрагмент отчета о результатах работы программы

## Выводы

Нагрузочное тестирование выполняет достаточно широкий круг задач, но в данной работе была поставлена задача убедиться в том, что информационная система по адресу: <http://www.hneu.edu.ua> способна успешно выдерживать нагрузки, которые она будет испытывать в процессе производственной эксплуатации и иметь необходимый запас прочности.

В результате проведенного нагрузочного тестирования можно сделать следующие выводы:

Производительность системы при нормальных рабочих условиях и плановой нагрузке составляет не более 100 пользователей в режиме он-лайн, при этом изменение интенсивности работы, увеличение количества данных в базе, изменение аппаратной платформы также ухудшат производительность информационной системы.

В критических условиях при увеличении пользователей свыше 100 пользователей система даст сбой и возвращает сообщение с кодом ошибки 503 – сбой на сервере.

Таким образом, разработчикам ИС по адресу: <http://www.hneu.edu.ua> можно быть уверенными в том, что система будет успешно функционировать при запуске в производство и по истечении прогнозируемых сроков эксплуатации, так как вероятность одновременного доступа к серверу в режиме онлайн свыше 100 человек очень мала.

Результаты функционального тестирования позволяют убедиться в том, что информационная система факультета экономической информатики соответствует функциональным требованиям, предъявляемым к ней конечными пользователями, защищена от несанкционированного использования и некорректных действий пользователей, имеет удобный пользовательский интерфейс, поддерживает заявленные аппаратные и программные платформы.

## Список литературы

1. Система управління якістю. Вимоги. ДСТУ ISO 9001:2009 Національний стандарт України [Чинний від 01.09.2009] Держспоживстандарт України [Електронний ресурс]. – Режим доступу к ресурсу: [http://www.gerele.dp.ua/index/info\\_dstu\\_iso\\_9001-2009.html](http://www.gerele.dp.ua/index/info_dstu_iso_9001-2009.html).
2. Системи управління якістю. Основні положення та словник термінів ДСТУ ISO 9000:2007 Національний стандарт України [Чинний від 01.01.2008] Держспоживстандарт України [Електронний ресурс]. – Режим доступу к ресурсу: <http://www.ukrndnc.org.ua/>.
3. Автоматизированное тестирование Веб-приложений [Электронный ресурс]. – Режим доступа к ресурсу: <http://www.developers.org.ua/archives/max/2006/06/05/selenium-ide-avtomatizirovannoe-testirovanie-veb-prilozheniy-za-15-minut/> – сайт Selenium
4. Аудит информационных систем и технологий [Электронный ресурс]. – Режим доступа к ресурсу: <http://www.isaca.ru/audit>.
5. Булат А. Метрики по обеспечению качества [Электронный ресурс] / А. Булат. – Режим доступа к ресурсу: <http://www.protesting.ru/qa/metrics.html>.
6. Вдовин И.А. COBIT 4.1 / И.А. Вдовин. – М.: Аудит и контроль информационных систем. 2008. – 16 с.
7. Дорохов О.В. Огляд критеріїв якості розробки програмного забезпечення / О.В. Дорохов, В.В. Гаркін, М. Драшкович // Системи обробки інформації: зб. наук. пр.; Проблеми і перспективи розвитку ІТ-індустрії, Харків, 18-19 листопада 2010 р. – Х., 2010. – Вип. 7 (88). – С. 181-182.
8. Модели менеджмента при разработке программных продуктов. ISO портал [Электронный ресурс]. – Режим доступа к ресурсу: <http://staratel.com/iso/InfTech/DesignPO/index.html>.
9. Оценка качества информационных систем (ИС) [Электронный ресурс]. – Режим доступа к ресурсу: <http://www.intuit.ru/department/hardware/sapr/16/>.

Поступила в редколлегию 22.04.2011

**Рецензент:** д-р экон. наук, проф. А.И. Пушкарь, Харьковский национальный экономический университет, Харьков

## РОЗРОБКА ТА ДОСЛІДЖЕННЯ МЕТОДИКИ НАВАНТАЖУВАЛЬНОЇ ТА ФУНКЦІОНАЛЬНОЇ ОЦІНКИ ЯКОСТІ ІНФОРМАЦІЙНИХ СИСТЕМ

В.В. Гаркін, Р.М. Чен

*Якість інформаційних систем в даний час є першорядною умовою просування програмного продукту на інформаційному ринку. Одним з ефективних методів визначення критеріїв якості, таких як: результативність і ефективність, є методи навантажувальних і функціональних випробувань. Розглянуто питання розробки і дослідження методики навантажувальної та функціональної оцінки веб-сервісу факультету економічної інформатики ХНЕУ.*

**Ключові слова:** інформаційна система, програмний продукт, методика навантажувальної оцінки, функціональна оцінка, якість ІС.

## DEVELOPMENT AND RESEARCH METHODS HOOK AND FUNCTIONAL EVALUATION OF THE QUALITY OF INFORMATION SYSTEMS

V.V. Garkin, R.N. Chen

*The quality of information systems at this time is a primary condition to promote the product in the information market. One effective way of determining quality criteria, such as: the effectiveness and efficiency, are the methods of load and functional testing. The issues of development and research methodology are loading and functional evaluation of a web service of Economic Informatics KNUU.*

**Keywords:** information system, software product, the method load assessment, functional assessment, IS's quality.