

УДК 681.324:623.618(06)

В.Г. Малюга, В.В. Калачова, В.Ф. Третяк, О.А. Трублін

Харківський університет Повітряних Сил імені Івана Кожедуба, Харків

ДОСЛІДЖЕННЯ ШЛЯХІВ АВТОМАТИЗАЦІЇ ТЕСТУВАННЯ ПРИ РОЗРОБЦІ КОМПЛЕКСУ ПРОГРАМ СПЕЦІАЛІЗОВАНОЇ АВТОМАТИЗОВАНОЇ СИСТЕМИ

Актуальною задачею, що стоїть як перед Харківському університеті Повітряних Сил (ХУПС), так і перед іншими вищими військовими навчальними закладами ЗС України, є підготовка висококваліфікованих військових фахівців. Вирішити її можна лише при якісному плануванні навчального процесу. Розклад занять є складовою частиною планування, який пов'язує в єдину систему різноманітні ланки й елементи навчального процесу і регламентує навчальну роботу її учасників. Автоматизація процесу конструювання розкладу дозволяє уникнути багатьох помилок, суб'єктивізму та значно скоротити час відведений на цю процедуру. В поточний період в ХУПС ведеться розробка комплексу програм автоматизованої системи конструювання розкладу занять. Одним із заключних та найважливіших етапів розробки комплексу має стати тестовий контроль якості програмного продукту. Успішність проведення тестувальних робіт багато в чому залежить від застосування автоматизованих засобів контролю, здатних значно скоротити час його проведення, вилучити фактор суб'єктивізму з процесу перевірки, оптимізувати програмний код та, взагалі, отримувати відповідь на ключове запитання: чи відповідає розроблений програмний продукт цільовому рівню вимог замовника чи ні.

Ключові слова: розклад навчальних занять, якість програмного забезпечення, рівні автоматизації тестування, тестовий набір, автоматизований тест-скрипт, бекенд процеси, end-to-end сценаріях.

Вступ

Постановка проблеми. Чітка організація навчального процесу у вищому військовому навчальному закладі є гарантом успіху підготовки високоякісних військових фахівців та може бути досягнута лише при якісному його плануванні. Одним з найбільш відповідальних, трудомістких і складних завдань планування навчального процесу є складання семестрових розкладів навчальних занять (екзаменів). Трудомісткість завдання складання розкладу обумовлена участю багатьох представників різних ланок управління, планування і забезпечення навчального процесу при підготовці, обробці і використанні великої кількості нормативної навчальної методичної документації та іншої інформації. Тому великі часові витрати, помилки та суб'єктивізм – є тим не повним переліком недоліків, що вимушує звертатися до наукоємних інформаційних технологій, які взмозі звести до мінімуму існуючі негаразди шляхом автоматизації процесу конструювання розкладу занять та створення відповідного комплексу програм.

Заключним етапом розробки комплексу програм автоматизованої системи конструювання розкладу є тестування працездатності створеного програмного забезпечення. Метою тестового контролю є виявлення якомога більшої кількості помилок і недоліків у створеному програмному продукті, ще до введення його в активну експлуатацію, з ціллю подальшого їх вилучення та досягнення максимальної якості та оптимальності програмного коду.

Спроба підвищити швидкість проведення тестувальних робіт, збільшити вірогідності виявлення дефектів та знизити собівартість тестування напря-

му зв'язана з автоматизацією цього процесу. Сьогодні на ринку програмного забезпечення присутня велика кількість програмних продуктів, реалізуючих процедуру автоматизованого тестування. Завдяки використанню в них різних видів, технік, методик, алгоритмів та стратегій тестування результати контролю якості теж відрізняються – програмний код має різний рівень оптимальності. У зв'язку з цим, питання аналізу особливостей проведення автоматизованого тестування, обрання відповідного підходу для його виконання та взагалі ефект від впровадження автоматизаційної складової, відкривають шлях до обґрунтованого вибору одного з програмних засобів автоматизації тестувальних робіт для здійснення перевірки працездатності комплексу програм автоматизованої системи конструювання розкладу занять.

Аналіз останніх досліджень і публікацій. Відомості щодо питання впровадження новітніх інформаційних технологій у навчальний процес в вищих військових навчальних закладах ЗС України докладно розкрито в [1-4]. Основні рівні автоматизації тестування та його застосування представлені [5-7]. Основні принципи здійснення автоматизованого тестового контролю якості програмних продуктів наведено в [8]. Недоліки та переваги програмних продуктів для автоматизованого тестування подано у [9, 10].

Формулювання мети статті. Дослідження основних принципів та особливостей застосування автоматизації в процесах перевірки якості програмної розробки, аналіз переваг і недоліків, що з цим пов'язані, ставе на меті надати тестувальникам та розробникам комплексу програм автоматизованої системи конструювання розкладу занять, що зараз

розробляється в Харківському університеті Повітряних Сил, можливість здійснення обґрунтованого вибору одного з представлених на ринку програмних засобів продукту для виконання процедури тестування розробки, що дасть можливість врахувати всі вимоги замовника та буде гарантом досягнення максимальної якості і оптимальності програмного коду.

Виклад основного матеріалу досліджень

Автоматизація є одним з традиційних шляхів зниження вартості та підвищення продуктивності для будь яких процесів. В зв'язку з тим, що більшість дефектів у програмному забезпеченні виявляється на стадії тестування, визначальною для економії коштів стає процедура її автоматизації.

Автоматизація тестувальних робіт дозволяє зменшити собівартість розробки і підвищити вірогідність виявлення дефектів.

Відправною точкою в автоматизації тестування програмних продуктів є усвідомлення і формалізація тих функціональних вимог, які пред'являє замовник до сервісу або програмного продукту. Під кожен зі сформульованих вимог створюється тест, перевіряючий правильність реакції системи. До всіх тестів прив'язуються процедури тестування, які можуть бути як ручні, так і автоматичні. Наприклад, при ручному тестуванні тестувальник отримує повідомлення про те, що йому необхідно відкрити певне вікно, виконати якусь послідовність дій, внаслідок чого він побачить очікуваний результат. В кінці процедури йому задається питання - побачив він цей результат чи ні. Відповідь фіксується. Навпаки, при автоматичному виконанні тесту всі кроки виконуються комп'ютером, що емулює дії користувача.

Ефектом від впровадження засобів автоматизації тестування є залежність витрат, які несе організація від необхідного рівня якості. Сумарна вартість якості складається із витрат на тестування і на виправлення помилок в системі із заданою якістю. В результаті вартість якості має певний мінімум, який відповідає оптимуму витрат. Якщо використовуються засоби автоматизації, то витрати на тестування зменшуються і, відповідно, знижуються витрати, що забезпечують досягнення оптимального показника якості. В той же час само значення оптимальної якості зміщується у бік збільшення.

Застосування автоматизації потрібно у ряді випадків майже обов'язково:

1) у важкодоступних місцях системи (бекенд процеси, запис в БД);

2) для функціональності, яка часто використовується та ризику від помилок в неї достатньо високі: автоматизувавши перевірку критичної функціональності, можна гарантувати швидке знаходження помилок, а значить і швидке їх усунення;

3) у рутинних операціях, таких як перебори даних (форми з великою кількістю полів для введення:

автоматизувати заповнення полів різними даними і їх перевірку після збереження);

4) у валідаційних повідомленнях: автоматизувати заповнення полів некоректними даними і перевірку на появу тієї або іншої валідації;

5) у довгих end-to-end сценаріях.

6) при перевірці даних, що вимагають точних математичних розрахунків;

7) при перевірці правильності пошуку даних.

І це ще не повний перелік випадків, коли застосування автоматизації має велике значення.

Автоматизоване тестування програмного забезпечення (ПЗ) заключається в верифікації програмного забезпечення, при якому основні функції і кроки тесту, такі як запуск, ініціалізація, виконання, аналіз і видача результату, виконуються автоматично за допомогою інструментів для автоматизованого тестування (спеціального програмного забезпечення) технічним фахівцем з автоматизованого тестування (Software Automation Testers), який забезпечує створення, відладку і підтримку працездатного стану тест скриптів (наборів інструкцій, для автоматичної перевірки певної частини програмного забезпечення), тестових наборів (комбінації тест скриптів, для перевірки певної частини програмного забезпечення, об'єднаною загальною функціональністю або цілями, що переслідуються запуском даного набору) і інструментів для автоматизованого тестування.

Для ефективнішого використання автоматизації тестування має сенс розробляти окремі тест-кейси, які будуть перевіряти:

- базові операції створення/читання/зміни/видалення сутностей (так звані операції CRUD - Create / Read / Update / Delete);

- типові сценарії використання додатку або окремі дії;

- інтерфейси роботи з файлами та інші моменти, незручні для тестування вручну.

Підхід до автоматизації тестування залежить від процесу розробки додатку. Якщо узяти за основу RUP (Rational Unified Process), то процедуру можна розбити на фази:

- початкова фаза (inception phase): вибір інструменту автоматизації, залежно від якого вирішується чи використовуватимуться вже готові напрацювання (фреймворки) або ж все буде написано "з нуля";

- фаза детальної розробки (elaboration phase): написання тестів на основну архітектуру (надалі ці тести використовуватимуться для прийому білда - Build Verification Tests);

- фаза побудови (construction phase): детальніша автоматизація - критична функціональність, перевірка регресій, end-to-end сценарії;

- перехідна фаза (transition phase): підготовка тестів до передачі замовникові (якщо це потрібно).

Умовно тестування додатку можна розбити на три рівні:

- рівень модульного тестування (Unit Tests Layer): під автоматизованими тестами на цьому рівні розуміються компонентні або модульні тести написані розробниками (тестувальниками). Наявність подібних тестів на ранніх стадіях проекту, а також постійне їх поповнення новими тестами, перевіряючими «баг фікси», убереже проект від багатьох серйозних проблем;

- рівень функціонального тестування (Functional Tests Layer non-ui): як правило не всю бізнес логіку додатку можна протестувати через GUI шар. Це може бути особливістю реалізації, яка ховає бізнес логіку від користувачів. Саме з цієї причини за домовленістю з розробниками, для команди тестування може бути реалізований доступ безпосередньо до функціонального шару, що дає можливість тестувати безпосередньо бізнес логіку додатку, минувши призначений для користувача інтерфейс;

- рівень тестування через призначений для користувача інтерфейс (GUI Tests Layer): на даному рівні є можливість тестувати не тільки інтерфейс користувача, але також і функціональність, виконуючи операції, що визивають бізнес логіку додатку. Є думка, що такі тести, що проходять наскрізь, мають більший ефект ніж просто тестування функціонального шару, оскільки ми тестуємо функціональність, емулюючи дії кінцевого користувача, через графічний інтерфейс.

Для забезпечення кращої якості програмного додатку, рекомендується автоматизувати всі три рівні його тестування. А для того щоб автоматизація тестування мала позитивний результат, а саме скоротила час його проведення, має сенс притримуватися наступних рекомендацій:

1) написанням тестів повинні займатися фахівці з автоматизованого тестування. Після написання, тести бажано передати команді ручного тестування, яка буде здійснювати їх щоденний запуск і аналіз результатів. Тим самим автоматизовані тести теж пройдуть тестування, яке збільшить їх надійність і життєздатність;

2) написані і відладжені тести також можуть передаватися команді розробки, для відладки нових версій;

3) команді розробки рекомендується здійснювати щоденну збірку, з прогнозом всіх написаних тестів на всіх рівнях автоматизації тестування. І лише після того, як нова версія починає задовольняти критеріям якості, здійснювати установку нової версії на QA платформу.

Застосування автоматизації є важливим при здійсненні: 1) тестування продуктивності (навантаження, стресостійкість, тестування на стабільність); 2) регресійного тестування, яке означає перевірку ПЗ на коректність функціональності, випущеної і протестованої в попередній версії., виконується з регулярною частотою, що задається залежно від умов: у когось з кожним новим білдом, а у когось - з кожною версією для замовника; 3) конфігураційного тесту-

вання – виконання одних і тих же тестів в різних умовах, тобто коли один або декілька компонентів архітектури системи потрібно перевірити в різному оточенні, зазвичай заявленому в початкових вимогах; наприклад, підтримка СУБД від різних виробників, робота в різних клієнтських браузерах, використання в декількох ОС і тому подібне, тобто це є аналог регресійного тестування, але в рамках однієї версії системи; 4) при функціональному тестуванні мова йде про перевірку нового функціонала. Іноді буває, що без автоматизації ніяк не обійтися, навіть якщо потрібно виконати тестування тільки один раз. Зазвичай, згодом ці тести використовуються для регресу.

Основними перевагами, які надає тестувальникові автоматизація є: 1) швидке виконання: автоматизованому скрипту не потрібно звірятися з інструкціями і документацією; 2) менші витрати на підтримку: коли скрипти вже написані, на їх підтримку і аналіз результатів потрібно, як правило, менший час, чим на проведення того ж об'єму тестування вручну; 3) не існує впливу «людського чинника»: люди не застраховані від помилок, а тест-скрипт, що виконується не пропустить тест по необережності і нічого не наплютає в результатах; 4) звіти про результати тестування: вони автоматично розсилаються і зберігаються; 5) виконання без втручання: під час виконання тестів інженер-тестувальник може займатися іншими корисними справами, або тести можуть виконуватися в неробочий час;

Разом з тим треба враховувати і ряд недоліків, пов'язаних із застосуванням автоматизації тестування:

1) повторюваність: всі написані тести завжди виконуватимуться одночасно, що одночасно є і недоліком і перевагою, оскільки тестувальник, виконуючи тест вручну, може звернути увагу на деякі деталі і знайти дефект, що виник, а скрипт цього, на жаль, зробити не може,

2) витрати на підтримку: чим частіше змінюється додаток, тим вони вищі;

3) великі витрати на розробку: розробка автоматизованих тестів це складний процес, тому що фактично йде розробка додатку, який тестує інший додаток;

4) вартість інструменту для автоматизації: у випадку, коли використовується ліцензійне програмне забезпечення, його вартість може бути досить висока, вільно поширювані інструменти, як правило, відрізняються скромнішим функціоналом і меншою зручністю роботи.

5) пропуск дрібних помилок: автоматичний скрипт може пропускати дрібні помилки, на перевірку яких він не запрограмований.

Висновки

Таким чином, враховуючи всі особливості проведення процедури автоматизованого тестування та зробивши оцінку переваг та недоліків, які з цим безпосередньо пов'язані, можна зробити висновок, що при тестуванні комплексу програм автоматизованої сис-

теми конструювання розкладу занять, що зараз розробляється в Харківському університеті Повітряних Сил, має сенс використовувати автоматизоване тестування розробки підчас верифікації функціональних вимог; контролю якості інтерфейсу, призначеного для користувача; перевірки роботи окремих модулів; комплексному тестуванні; аналізі складності програмних модулів; тестуванні покриття програмного коду; тестуванні швидкості завантаження системи; тестуванні граничних умов; тестуванні витоку пам'яті.

Список літератури

1. Біла книга 2012. ЗС України. – К. Видання МОУ, 2013. – 78 с.
2. Наказ Міністра освіти України від 02.06.1993 р. №161 "Про затвердження положення про організацію навчального процесу у вищих навчальних закладах"
3. "Інструкція про організацію освітньої діяльності у вищих військових навчальних закладах Збройних Сил України та військових навчальних підрозділах вищих навчальних закладів України", затверджена наказом Міністра освіти і науки України та Міністра оборони України від 13.04.2005 р. № 221/217
4. "Інструкція з планування та обліку діяльності науково-педагогічних (педагогічних) працівників вищих військових навчальних закладів (військових навчальних підрозділів вищих навчальних закладів, навчальних центрів Збройних Сил України" введена наказом Міністра оборони України та Міністра освіти і науки України від

08.05.2002 р. №155/291 зі змінами згідно сумісного наказу МОУ та МОНУ від 21.02.2006 №99/116

5. Гленфорд Майерс *Искусство тестирования программ, 3-е издание = The Art of Software Testing, 3rd Edition / Гленфорд Майерс, Том Баджетт, Кори Сандлер — М.: Диалектика, 2012. — 272 с. — ISBN 978-5-8459-1796-6.*
6. Лайза К. *Гибкое тестирование: практическое руководство для тестировщиков ПО и гибких команд = Agile Testing: A Practical Guide for Testers and Agile Teams / Лайза Кристин, Джанет Грегори. — М.: Вильямс, 2010. — 464 с. — (Addison-Wesley Signature Series).*
7. Канер Кем. *Тестирование программного обеспечения. Фундаментальные концепции менеджмента бизнес-приложений / Канер Кем, Фолк Джек, Нгуен Енг Кек. — К.: ДиаСофт, 2001. — 544 с. — ISBN 9667393879.*
8. Калбертсон Роберт, Браун Крис, Кобб Гэри *Быстрое тестирование. — М.: «Вильямс», 2002. — 374 с. — ISBN 5-8459-0336-X.*
9. Синуцын С.В. *Верификация программного обеспечения / С.В. Синуцын, Н.Ю. Налютин. — М.: БИНОМ, 2008. — 368 с. — ISBN 978-5-94774-825-3.*
10. Бейзер Б. *Тестирование чёрного ящика. Технологии функционального тестирования программного обеспечения и систем / Б. Бейзер. — СПб.: Питер, 2004. — 320 с.*

Надійшла до редколегії 16.09.2015

Рецензент: д-р фіз.-мат. наук проф. С.В. Смеляков, Харківський університет Повітряних Сил ім. І. Кожедуба, Харків.

ИССЛЕДОВАНИЕ ПУТЕЙ АВТОМАТИЗАЦИИ ТЕСТИРОВАНИЯ ПРИ РАЗРАБОТКЕ КОМПЛЕКСА ПРОГРАММ СПЕЦИАЛИЗИРОВАННОЙ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ

В.Г. Малуго, В.В. Калачева, В.Ф. Третьяк, А.А. Трублин

Актуальной задачей, которая стоит как перед Харьковском университете Воздушных Сил (ХУВС), так и перед другими высшими военными учебными заведениями ВС Украины, является подготовка высококвалифицированных военных специалистов. Одним из возможных путей ее решения является качественное планирование учебного процесса. Расписание занятий является составной частью планирования, которое связывает в единую систему многообразие звеньев и элементов учебного процесса и регламентирует учебную работу ее участников. Автоматизация процесса конструирования расписания позволяет избежать множества ошибок, субъективизма и значительно сократить время отведенное на эту процедуру. В текущий период в ХУВС ведется разработка комплекса программ автоматизированной системы конструирования расписания занятий. Одним из заключительных и важнейших этапов разработки комплекса должен стать тестовый контроль качества созданного программного продукта. Успешность проведения работ по тестированию во многом зависит от применения автоматизированных средств контроля, способных значительно сократить время его проведения, отойти от субъективизма во время проверки, оптимизировать программный код и, вообще, получить ответ на ключевой вопрос: отвечает ли разработанный программный продукт целевому уровню предъявляемых требований заказчика или нет.

Ключевые слова: расписание учебных занятий, качество программного обеспечения, уровни автоматизации тестирования, тестовый набор, автоматизированный тест-скрипт, бекенд процессы, end-to-end сценарии.

RESEARCH OF WAYS OF AUTOMATION OF TESTING AT DEVELOPMENT OF PROGRAMS COMPLEX OF THE SPECIAL AUTOMATED SYSTEM

V.G. Malyuga, V.V. Kalachova, V.F. Tretyak, O.A. Trublin

An actual task, which stands both before the Kharkov university of Aircrafts (KHUA) and before other higher soldiery educational establishments of SS of Ukraine, is preparation of highly skilled soldiery specialists. One of possible ways of its decision is the high-quality planning of educational process. A curriculum of employments is component part of planning which binds in the single system the variety of links and elements of educational process and regulates educational work of its participants. Automation of process of constructing of time-table allows to avoid the great number of errors and considerably to shorten time taken on this procedure. In a current period development of complex of the programs of the automated system of constructing of curriculum of employments is conducted in KHUVS. Test control of quality of the created software product must one of final and major design of complex times become. Success of lead through of works on testing in a great deal depends on application of the automated controls, capable considerably to shorten time of his lead through, step back from subjectivism during test, optimize a programmatic code and, in general, get an answer for a key question: whether the developed software product answers to the having a special purpose level of the produced requirements of customer or not.

Keywords: curriculum of lessons, quality of software, levels of automation of testing, test set, automated test-script, backend processes, end-to-end scenarios.