

УДК 004.4,378.146

ОСНОВНІ ВИМОГИ ДО КОМП'ЮТЕРНИХ СИСТЕМ ТЕСТУВАННЯ ЗНАНЬ (КСТЗ)

Фетісов В.С.

У статті розглядається класифікація комп'ютерних систем тестування знань, а також критерії, знання яких дозволить користувачеві краще зорієнтуватися під час вибору потрібної системи.

Ключові слова: тест, класифікація комп'ютерних систем тестування знань, вимоги до комп'ютерних систем тестування знань.

В статье рассматривается классификация компьютерных систем тестирования знаний, а также критерии, которыми следует руководствоваться при выборе такой системы.

Ключевые слова: тест, классификация компьютерных систем тестирования знаний, требования к компьютерным системам тестирования знаний.

The article considers the classification of computer-based knowledge testing systems and the criteria for selection of the appropriate system.

Key words: test, classification of computer-based knowledge testing systems, requirements for computer-based knowledge testing systems.

На ринку програмного забезпечення пропонується велика кількість найрізноманітніших КСТЗ, що розрізняються за різноманітними критеріями. І це дуже добре, оскільки тим, хто збирається працювати з ними, є з чого вибирати. Проте при цьому виникає проблема вибору: яка ж система, власне кажучи, нам потрібна і чим саме слід керуватися у її виборі? Для цього, на думку автора, слід керуватися певними критеріями, які і пропонуються в статті.

Проте перш ніж перейти безпосередньо до розгляду цих критеріїв, на мій погляд, доцільно розглянути класифікації КСТЗ.

Класифікація КСТЗ

КСТЗ становлять специфічну групу програмного забезпечення. Їх можна класифікувати за різними класифікаціями.

За ступенем універсальності: універсальні спеціалізовані. Універсальні призначені для створення тестових завдань у будь-яких галузях, спеціалізовані – орієнтовані на певну галузь, наприклад, математику, хімію.

За рівнем автономності:

1. У складі LMS (Learning Management System), тобто систем управління навчанням (*Moodle, SharePoint Learning Kit*).

2. Окремий модуль.

3. Комплексний набір модулів.

За складом модулів:

1. Наявність окремих модулів для складання тестів і проведення тестування.

2. Наявність спеціального модуля тільки для складання тестів.

За характером проведення тестування.

1. За допомогою інструментальних оболонок, призначених саме для проведення тестування.

2. За допомогою браузера. Таке тестування ще називають *WWW-тестуванням* або *Інтернет-тестуванням*.

Нарешті, як і усе інше програмне забезпечення, КСТЗ поділяються на комерційне (пропріетарні) і безкоштовне (freeware). У складі безкоштовного програмного забезпечення у свою чергу можна виділити таке, що має відкритий код, що дозволяє бажаним удосконалювати систему і з закритим кодом, коли за програмний код відповідає тільки розробник системи.

Надалі розглядаються найбільш типові, на думку автора, представники програмного забезпечення, серед яких були виділені представники трьох груп КСТЗ:

1) такі, що входять до складу систем управління навчанням;

2) комплексна система, що використовує спеціалізовані оболонки для складання тестових завдань, проведення тестування й адміністрування;

3) система, що має спеціальний модуль тільки для складання тестів, а тестування здійснюється в Інтернеті за допомогою браузера.

Оптимальною структурою КСТЗ є модульна побудова, у складі якої є три компоненти:

1. Модуль створення, підготовки і редагування тестових завдань.

2. Модуль проведення тестових завдань (інколи його називають *Player* тестів).

3. Модуль адміністрування.

Загальні вимоги до КСТЗ

КСТЗ повинна виконувати свою найважливішу функцію – *забезпечення ефективного тестування знань*. Але ефективне тестування знань складається з багатьох складових. Одні з них стосуються КСТЗ у цілому, інші – її конкретних модулів. При цьому, звичайно, до КСТЗ у конкретних умовах будуть висуватися конкретні вимоги. Але серед них можна виокремити кілька основних вимог, яким практично завжди має відповідати КСТЗ:

1. *Універсальність*. КСТЗ має використовуватися як для внутрішнього, так і для зовнішнього тестування. Можливість використання КСТЗ для внутрішнього і зовнішнього тестування досягається гнучкою побудовою системи, що передбачає розподіл її користувачів за ролями, підтримкою різних систем і шкал оцінювання тощо. Наприклад, для користувача з роллю “свій студент” слід вести систему оцінок, яка відповідає навчальному плану навчального закладу, а для зовнішнього користувача можна застосувати іншу, наприклад, накопичувальну систему оцінювання успішності тестування.

2. *Підтримка безпечного й універсального механізму керування користувачами з розподілом прав доступу*. Це питання тісно пов'язане з функцією універсальності. КСТЗ повинна дозволяти мати користувачів із різними правами, що забезпечують доступ до різних функцій і складових системи. Наприклад, можна виділити такий тип користувачів: “свій” студент, студент, який не є студентом навчального закладу, де використовується система, викладач, розробник тестів, адміністратор.

3. *Захищеність інформаційної бази тестів і результатів тестування*, доступ до яких повинні мати тільки користувачі з відповідними правами.

4. *Шифрування* інформаційного наповнення тестів, принаймні це стосується відповідей. Останні три вимоги складають те, що можна назвати *інформаційною безпекою* тестів.

5. *Підтримка основних форм тестових завдань* (завдання із простим і складним множинним вибором, відкритої форми, на встановлення відповідності та на відновлення правильної послідовності).

6. *Підтримка двох типів проведення тестування*. За своїм функціональним призначенням тести можуть бути двох типів: для проведення соціологічних або психологічних тестів і для проведення тестів знань (контрольних тестів). За першим варіантом особа, яка тестується, відповідає на всі завдання тесту. За результатами тесту за заздалегідь визначеними правилами підраховується сума балів, який відповідає певний висновок. За другим варіантом особа відповідає на *заздалегідь визначену кількість випадково відібраних контрольних завдань у тестовій формі* для визначених у разі настроювання розділів, тем.

7. *Підтримка різних режимів тестування*:

- *Екзаменаційний режим*. У цьому режимі за невірні відповіді віднімаються бали, а окремі контрольні завдання дозволяється пропускати, за що бали не додаються і не віднімаються.

- *Режим самотестування або тренажера*. Під час тестування виводяться повідомлення про помилки або рецензія на відповідь.

- *Режим навчання*. Може містити вступ і пояснення до завдання, різноманітні навчальні

матеріали у текстовому або мультимедійному вигляді.

8. Можливість імпорту тестів у текстовий формат і формат HTML, а також експорту тестів з електронних текстових версій паперових тестів або з інших КСТЗ.

9. *Наявність української (російської) локалізації*.

10. *Наявність докладної документації українською (російською) мовою*.

11. *Технічна підтримка КСТЗ*.

Крім цих основних вимог, до конкретних КСТЗ можуть бути висунуті також такі.

1. *Вартість самої системи*, якщо вона є пропрієтарною. Одночасно слід урахувати, що потужні КСТЗ можуть використовувати додаткові програмні засоби, наприклад, платні СУБД (Microsoft SQL Server), тому – за їх наявності – слід урахувати вартість придбання відповідного програмного забезпечення.

2. Якщо КСТЗ планується використовувати на національному або міждержавному рівнях, то вона має відповідати державним і міжнародним стандартам, наприклад, для другого варіанту відповідати стандарту IMS QTI (Information Model Specification Question & Test Interoperability). У цьому випадку на “повістку дня” постає також питання підтримки КСТЗ різних операційних систем (багато-платформеність), оскільки зовсім не є фактом, що всі користувачі будуть використовувати ту саму операційну систему. Більше того, на Україні, як і всьому світі, спостерігається стійка тенденція до розширення використання UNIX-подібних операційних систем, тобто вільного програмного забезпечення.

3. КСТЗ, які мають за мету отримати широке розповсюдження у навчальному процесі, мають підтримувати стандарт збереження освітнього контенту SCORM. SCORM (Sharable Object Reference Model) – це набір стандартів і специфікацій, що був розроблений для дистанційного навчання. Він зокрема містить вимоги до організації навчального матеріалу. Згідно концепції SCORM навчальний матеріал має подаватися відносно невеликими блоками, які можуть бути використані у різних навчальних курсах і використовуватися у системах дистанційного навчання незалежно від того, у межах якої системи вони були створені. Отже, SCORM дозволяє забезпечити сумісність компонентів курсів і можливість їх багаторазового використання.

4. Не є обов'язковою вимога відсутності *прив'язування до конкретної предметної галузі знань*. Але інколи як раз потрібна саме спеціалізована оболонка для складання тестових завдань за допомогою якої можна, наприклад, здійснювати молекулярне моделювання у хімії, будувати електричні схеми у фізиці та ін.

5. Можливість створення за допомогою КСТЗ *паралельних завдань*. Для цього застосовуються так звані *фасети*. Фасет – це форма запису кількох варіантів того самого завдання.

6. Окремим питанням є здатність КСТЗ відповідати вимогам *адаптивного тестування*. Такого роду тестування враховує індивідуальні здібності особи, яка тестується, шляхом аналізу відповідей на тестові завдання. Якщо відповідь правильна, то вважається,

що рівень підготовки особи, яка тестується, вищий за складність тестового завдання і вона здатна вирішувати завдання не тільки цієї складності, а ще й вище, в іншому випадку – ні. Адаптивне тестування враховує індивідуальні особливості і відповідно підвищує або зменшує рівень складності подальших тестових завдань. У результаті особа, яка проходить тестування, самостійно встановлює для себе певний рівень складності завдань. Звичайно, що й оцінювання завдання більш високої складності відповідно має оцінюватися більш високими балами.

У свою чергу конкретні вимоги можна застосувати і до кожного модуля КСТЗ.

Вимоги до модуля створення, підготовки і редагування тестових завдань

1. Створення необмеженої кількості тестів, тем, завдань і варіантів відповідей на них.

2. Максимально просте і зручне формування та модифікація наповнення тестів, у тому числі керування тестовими завданнями: зміна їх порядку, вставка, переміщення, вилучення і копіювання, а також пошук і відбір (*фільтрування*) їх за розділом, темою, завданням.

3. Перевірка цілісності створюваного тесту, тобто наявність засобів контролю відповідності тексту завдань і варіантів відповідей, коректність посилань на ресурси і т.ін.

4. Здійснення орфографічного контролю.

5. Наявність базових функцій з форматування тексту завдань і варіантів відповідей, у тому числі вибір накреслення, розмірів, шрифтів і кольору; використання таких видозмін як верхні і нижні індекси, закреслення і застосування їх на рівні слів, речень, окремих завдань, відповідей; створення списків.

6. Визначення ваги тестового завдання.

7. Формування індивідуальної інструкції з виконання для кожного тестового завдання.

8. Можливість додавання мультимедійних об'єктів, у першу чергу графічні зображення, а також відео та аудіоресурсів. Зрозуміло, наприклад, що графічні зображення є необхідною складовою тестів у хімії, біології, фізики тощо. Крім цього, у деяких випадках можлива потреба додавання OLE-об'єктів (діаграми, креслення, таблиці, формули).

9. Можливість формування додаткових форм тестових завдань.

10. Якщо планується використовувати КСТЗ у режимі навчання, то доцільно мати у складі КСТЗ можливість додавання текстових файлів, зміст яких є навчальною складовою.

11. Друк паперового варіанту тесту.

Зручною можливістю модуля є наявність його *portable* версії – класу програм, що не вимагають попередньої інсталяції і можуть бути завантажені з будь-якого носія, в тому числі змінного. Це дозволяє розробнику тестів використовувати будь-який комп'ютер для формування тесту, а особі, яка проводить тестування – для проведення тестування без необхідності попередньої інсталяції програми.

Доцільно надати розробнику тесту мати можливість формувати:

1) загальний заголовок, стислу анотацію, стислу інформацію щодо автора (розробника тесту);

2) загальну інструкцію з виконання;

3) нотатки (будь-які зауваження до тесту розробника, що бачить тільки він).

Настроювання модуля має містити таке:

1. Можливість розташування завдань і варіантів відповідей на них як випадковим чином, так і строго послідовно.

2. Можливість застосування різних *шкал оцінювання*. Оптимальним варіантом є можливість застосування будь-якої шкали, але для конкретного навчального закладу може бути достатнім наявність тих шкал, які застосовують у ньому застосовують, наприклад, 5-бальну систему оцінювання.

3. Обмеження кількості завдань за темами.

Складання тестових завдань за допомогою КСТЗ
Кожна форма тестового завдання має певні особливості, які повинні враховувати КСТЗ і відповідно мати засоби їх реалізації під час складання тестових завдань.

Тестові завдання відкритої форми

Під час формування тестових завдань відкритої форми КСТЗ повинна мати такі можливості:

1) введення відповіді у вигляді тексту;

2) чутливість (або її відсутність) до відповіді, що вводитьься;

3) введення відповіді у вигляді числа;

4) завдання допусків для відповіді у вигляді числа;

5) формування кількох варіантів правильних відповідей;

6) неоднозначність відповіді на завдання відкритої форми вимагає необхідності інструкції з виконання.

Слід зауважити, що для багатьох тестових завдань інструкція з виконання буде одноковою, тому дуже зручно, якщо КСТЗ надає розробнику тестів можливість збереження типових інструкцій з виконання, наприклад, у відповідному довіднику і підключення їх за необхідності до завдань.

Тестові завдання, що автоматично генеруються

Завжди існує певний відсоток осіб, які завдяки гарній пам'яті або шляхом звичайного зубріння запам'ятовує якщо не всі, то переважну кількість завдань і правильних відповідей на них. Найкращим варіантом подолання цієї проблеми є істотне збільшення кількості завдань. Розглянемо такий приклад. Розмірність звичайної таблиці множення дозволяє створити 81 завдання (розмірність таблиці множення "9*9") на зразок "Результат множення числа "2" на "2" дорівнює...". Але вже для таблиці розмірністю "100*100" кількість відповідей становить 10000. І якщо у першому випадку це можливо без особливих зусиль увести до тесту 81 завдання, то для другого – зробити це буде дуже важко, не кажучи вже про стрімке непродуктивне зростання кількості завдань і, відповідно, обсягу тесту.

Розв'язати таку проблему дає змогу використання у тестах тестових завдань, змінна частина яких формується самостійно деяким випадковим чином. Відносно наведеного вище прикладу це означає, що замість того, щоб формувати завдання для всіх можливих комбінацій множення, формується загальна текстова частина завдання "Результат множення числа "n" на "m" дорівнює...", до якого автоматично випадковим чином генеруються і

підставляються значення “*n*” і “*m*”. Зрозуміло, що для цього потрібна невеличка програма, що виконує генерації чисел, а потім – дію їх множення. Фантазія програміста може підказати найрізноманітніші способи генерації чисел “*n*” і “*m*”. Їх можна сформувати випадковим чином, наприклад, за допомогою генератора випадкових чисел, залежно від значень поточної дати і часу і т.ін.

Сучасні засоби програмування надають можливість створення такого типу додатків. Наприклад, усе більш широке застосування знаходять інтерактивні програми в інтернеті. Інша справа, щоб інструментальна оболонка для створення таких тестів мала засоби для підключення сторонніх модулів.

Звичайно, це стосується відкритої форми тестового завдання, проте і саме ця форма вважається більш адекватною для оцінки знань.

Існує небагато КСТЗ, які дозволяють створити таке тестове завдання. Зокрема, це – Moodle. Вона як раз і містить особливий вид тестового завдання “Обчислювальне”. Це – математичний обчислювальний вираз, що містить поля для підстановки, які будуть випадковим чином автоматично заповнюватися з певної множини значень під час проходження тесту.

Тестові завдання із множинним вибором

1. Можливість формування найпростішого різновиду завдань із простим множинним вибором, що часто виокремлюють в окремий тип тестового завдання – альтернативне завдання. При цьому бажано мати функції формування списку відповідей і вибору з нього, оскільки відповіді на такого роду завдання, як правило, є стандартними, на зразок “Так” – “Ні”, “Правильно” – “Не правильно”.

2. Контроль помилкової відсутності правильної відповіді.

3. Відсутність обмежень на розміри тексту завдання і відповідей для запобігання ситуації, коли розробнику тесту не вистачає відведеної кількості позицій, хоч, зрозуміло, що довжина завдання і відповідей не може бути дуже великою.

Тестові завдання на встановлення відповідності

Можливість уведення різної кількості елементів у стовпчиках із завданнями і відповідями з метою уникнення пасивного додавання останнього завдання до останнього завдання.

Тестові завдання на відновлення правильної послідовності

Можливість упорядкування послідовностей як за зростанням, так і за спаданням.

Вимоги до модуля проведення тестових завдань

1. Головною вимогою до модуля тестування є *ергономічність*. Інтерфейс оболонки для проходження тестування має бути максимально зручним. Особа, яка проходить тестування, має витратити мінімум часу на вивчення інтерфейсу і пошук потрібних для відповіді елементів (наприклад, полів введення). Саме вікно модуля повинне містити тільки потрібні для тестування елементи. Оптимальним варіантом є можливість налаштування інтерфейсу “під себе” шляхом вилучення службових елементів вікна (панелі інструментів, допоміжної текстової інформації, годинника, що відображає час проходження тестування).

2. Захист тесту на рівні завдання пароля для обмеження доступу до загальних налаштувань модуля.

3. Вибір організації тестування: у локальному режимі – на кожному комп’ютері окремо або в інтернеті.

4. За умови організації тестування у локальному режимі зручно мати portable-версію модуля, що дозволяє провести тестування на будь-яких комп’ютерах, не витрачаючи часу на інсталяцію КСТЗ.

5. Експорт результатів тестування у вигляді текстового файлу для подальшого аналізу, імпорт в електронні таблиці тощо.

6. Якщо за закінченням тестування передбачається публікація його результатів в інтернеті, то модуль має містити можливість експорту результатів тестування у форматі HTML.

Найважливіше значення мають питання, пов’язані із аналізом результатів тестування:

1. Збирання, збереження і подання у зручній формі різноманітної статистичної інформації результатів тестування. Найкращим варіантом є наявність засобів (так званих “конструкторів форм”), за допомогою яких можна самостійно будувати потрібні звіти результатів тестування.

2. Інтелектуальний аналіз тестів з метою оцінки якості, придатності окремих завдань у тестовій формі, виявлення некоректно сформульованих або неправильних завдань.

Модуль повинен також надавати користувачеві здійснювати низку налаштувань, серед яких найбільш важливими можна назвати такі:

1. Вибір режиму тестування.

2. Завдання кількості тестових завдань.

3. Встановлення обмеження за часом.

4. Можливість вибору варіанту формування завдань у тестовій формі та (або) відповідей: у випадковому порядку або послідовно відповідно до порядку їх розташування у тесті.

5. Можливість завдання способу відбору завдань з тестової вибірки, що формується з кількох тем. Наприклад, повинна бути можливість відбору до вибірки завдань у тестовій формі пропорційно їх кількості за темами.

6. Можливість урахування рівня складності під час формування тестової вибірки.

7. Великий відсоток осіб із погіршеним зором потребує наявності оперативної зміни масштабу відображення інформації.

Можна навести також кілька другорядних налаштувань. Наприклад, дехто з осіб, які проводять тестування, мають бажання чути звуковий сигнал за закінченням тестування, деякі особи, які проходять тестування, бажують бачити час тестування. А під час аналізу результатів тестування доцільно надати можливість користувачеві самостійно визначити кольорову інтерпретацію правильних, частково правильних і помилкових відповідей.

Вимоги до модуля адміністрування

1. Можливість контролювати процес тестування у реальному режимі часу.

2. Захист результатів тестування.

3. Можливість призупинення тестування, запам’ятовування його результату, і продовжити через певний час із використанням збережених даних.

4. За умовою організації тестування в інтернеті слід визначитися, чи є необхідність у виділеному сервері.

5. Простота надбудови, масштабування системи. набувають питання, пов'язані з інформаційною
6. Якщо передбачається розташувати тести на безпекою тестів, зокрема можливість шифрування
серверах в інтернеті, то подальшого поглиблення тестів безпечними алгоритмами шифрування.

Література

1. Офіційний сайт Башлакова А. С. [Електронний ресурс]. – Режим доступу: mytest.klyaksa.net. – Назва з екрану.
2. Сайт Moodle community [Електронний ресурс]. – Режим доступу: <http://moodle.org/>. – Назва з екрану.
3. Сайт розробників Hot Potatoes [Електронний ресурс]. – Режим доступу: <http://hotpot.uvic.ca/>. – Назва з екрану.
4. Фетисов В. С. Критерии создания систем для проведения тестирования / В. С. Фетисов // Научный часопис імені М.П.Драгоманова. Серія № 5 “Педагогічні науки: реалії та перспективи” : збірник наукових праць / за ред. В. П. Сергієнка. – К. : Вид-во НПУ імені М.П.Драгоманова, 2010. Вип. 20. – 2010. – С. 152–155.