

ЗАСТОСУВАННЯ ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ ГНУЧКОГО ТЕСТУВАННЯ РІВНЯ ЗНАНЬ У СЕРЕДОВИЩІ MOODLE

В статті досліджено проблему підвищення ефективності тестування рівня отриманих знань із використанням комп'ютерних засобів. Запропоновано інформаційну технологію гнучкого тестування рівня знань, що формує репрезентативні набори тестових завдань та адаптивно обирає тестові завдання в процесі тестування. Представлено тестові програмні продукти, що реалізують запропоновану інформаційну технологію гнучкого тестування й дозволяють підтвердити її високу наукову та практичну цінність.

Ключові слова: тестування, тестові завдання, навчальні матеріали, Moodle, аналіз, ключові терміни.

O. BARMAC, O. MAZURETS, A. MATVIICHUK

Khmelnytskyi National University

APPLYING OF THE INFORMATION TECHNOLOGY OF THE FLEXIBLE TESTING OF KNOWLEDGE LEVEL IN MOODLE ENVIRONMENT

The problem of increasing efficiency of testing of the level of obtained knowledge with the help of computer technologies was researched in the article. Information technology of dynamic testing of knowledge level, which forms the representative set of tests and adaptively chooses test tasks was suggested. Test software products, which are implementing suggested information technology of flexible testing and prove it high scientific and practical value were reviewed.

Keywords: testing, test exercises, educational materials, Moodle, analysis, key terms.

Постановка проблеми в загальному вигляді

Вивчаючи певний предмет чи предметну область, людина отримує нові знання та навички, які є упевненим розумінням предмету, вміння самостійно поводитися з ним, розбиратися в ньому, а також використовувати для досягнення поставленої мети. Здобувши певні знання, виникає необхідність в визначенні їхнього рівня та компетентності людини в певній сфері [1]. Існує ряд методів для визначення рівня знань: попередній контроль, біжучий контроль, усна перевірка знань, письмова перевірка, практична перевірка, тестова перевірка. Останнім часом все більшої популярності набуває тестова перевірка знань, яка дає змогу об'єктивно оцінити рівень обізнаності особи в конкретній галузі [2]. Тестова перевірка включає в себе набір тестових завдань різної складності, що робить результат тестування більш об'єктивним. За допомогою ретельного конструювання тесту можна забезпечити відповідний рівень дискримінативності, а це саме те, в чому тести виграють у порівнянні з іншими формами визначення рівня знань. Проблемаю визнається низька гнучкість сучасних автоматизованих систем оцінювання рівня знань при формуванні тестових вибірок.

Аналіз останніх досліджень

З розвитком комп'ютерних технологій і їх впровадженням в навчальний процес з'явився великий вибір систем для тестування рівня знань. Умовно всі системи можна поділити на онлайн-системи та системи-додатки, що встановлюються на комп'ютер користувача. Перевагою онлайн-систем є можливість працювати з ними з різних машин і централізовано проводити оновлення програмних алгоритмів. Однак для роботи необхідний постійний, стабільний доступ в Інтернет. Системи тестування, що є окремими додатками, дозволяють працювати без доступу в Інтернет, і це дозволяє забезпечити стабільнішу роботу програми й ставити менше вимог для її роботи. Проте такі системи складніші в обслуговуванні, так як оновлення програмного продукту повинне бути здійснене для кожного комп'ютера на який така програма встановлена. Прикладами реалізованих систем для тестування рівня знань є система комп'ютерного тестування знань студентів OpenTEST [3], розроблена Харківським національним університетом радіоелектроніки, яка передбачає нелінійне проходження тесту з можливістю недискретного переходу між запитаннями, та система тестування Test-W2 [4], яка надає можливість проходження тестування з діагностикою й забезпечує різну послідовність запитань на сусідніх комп'ютерах. Втім, більшість існуючих систем тестування генерує тести, підбираючи й комбінуючи тестові завдання випадковим чином, що знижує гнучкість і інтерактивність процесу тестування.

Питанню підвищення адаптивності систем контролю знань присвячена велика кількість публікацій вітчизняних та закордонних авторів. Зокрема, автори [5] пропонують ряд моделей, за допомогою яких здійснюється зміна складності запитань у режимі реального часу, розглянуто технологію адаптивного визначення рівня підготовленості особи, що тестується. Розробці й удосконаленню методів, алгоритмів та технологій контролю знань присвячені роботи багатьох авторів, що досліджують проблеми автоматизованих систем навчання. Наприклад, автор [6] пропонує використання зважування понять моделі онтології для підвищення ефективності інтелектуального аналізу тесту, автори [7] розглядають моделі та методи адаптивного контролю знань, в роботі [8] розглядається модель побудови інтелектуальної системи автоматизованого контролю знань студентів вузів. Велика кількість публікацій та практична потреба в підвищенні гнучкості моделей та алгоритмів контролю рівня отриманих знань підтверджує актуальність

даної проблеми.

Прикладом онлайн-системи, що використовується для навчання та оцінювання, є Moodle – безкоштовна, відкрита система управління навчанням [9]. Вона реалізовує взаємодію між викладачами та учнями через мережу Інтернет. Система Moodle потребує постійного і стабільного підключення до Інтернету, цей фактор не дає змоги використовувати дану систему в місцях, де є проблеми з підключенням до Інтернету. Вона працює як серверна система і для забезпечення можливості одночасної роботи великої кількості користувачів потрібно мати потужний сервер, на якому вона буде запущена. Попри це, в класі онлайн-систем Moodle є достатньо ефективним та функціональним зразком програмного забезпечення (рис. 1). Недоліком її тестової складової є негнучкий метод випадкової генерації тестів, що при підборі запитання не враховує його складності та результати відповіді на попередні запитання. Така генерація тесту призводить до того, що різні учні отримують різний за сукупною складністю набір запитань, що ставить їх в нерівні умови і зменшує точність процесу тестування.

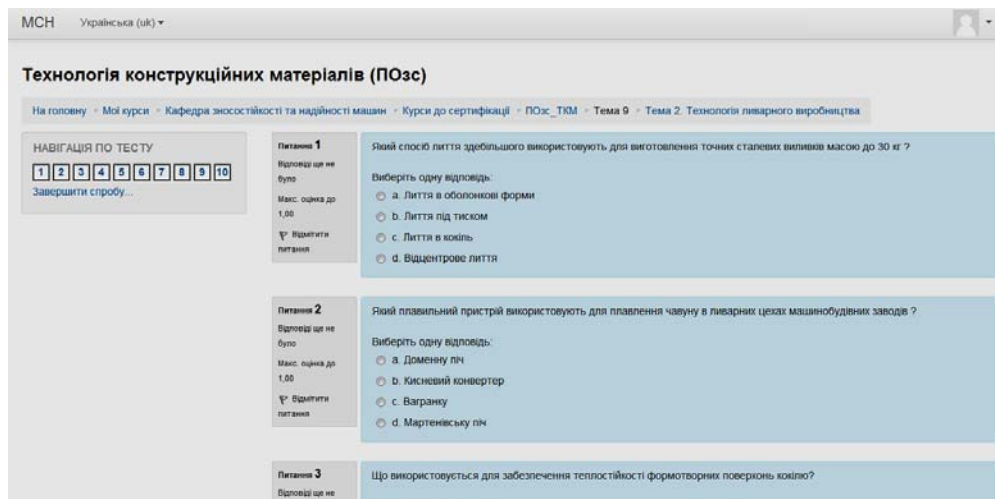


Рис. 1. Приклад проходження тесту у середовищі Moodle

Оскільки система Moodle є прийнятним і широкоживаним зразком онлайн-системи, доречним є удосконалення її можливостей тестування в напрямку гнучкості й адаптивності до ходу тестування, а не розробка альтернативної системи тестування з перспективним алгоритмом. Для семантичного зв'язування тестових завдань із матеріалом, засвоєння якого перевіряється, авторами у попередніх роботах запропоновано метод їх прив'язки до онтології навчального матеріалу [10] через визначення й аналіз ключових термінів, що використовуються в контенті навчальних та тестових матеріалів [11].

Постановка задачі

Метою даної роботи є розробка інформаційної технології гнучкого тестування рівня отриманих знань та дослідження можливостей її застосування у середовищі Moodle.

Викладення основних матеріалів дослідження

Середовище Moodle надає інструменти для розробки курсів, можливість створення тестів різних типів, швидке та зручне їхнє редагування, автоматичне форсування тестів, перегляд результатів тестування та обрахунок статистичних даних по проведеному тестуванню. При генерації тесту можна вказувати необхідну кількість завдань, час проходження тесту, кількість спроб які може здійснити учень, час початку та кінця проведення тестування. Середовище Moodle надає можливість для створення окремих програмних блоків – плагінів, які інтегруються в систему та додають нові можливості. Розробка такого плагіну дозволяє надати гнучкості алгоритму вибору запитань для тестування, проте для онтоприв'язки тестових завдань до контенту навчального матеріалів необхідний окремий додаток. Окрім гнучкості самого процесу тестування, важливим аспектом коректної перевірки знань є репрезентативність самих завдань у тесті, база яких має бути структурно відповідною основним поняттям, що висвітлені у навчальних матеріалах.

У зв'язку з вище наведеним, було розроблено структуру інформаційної системи (рис. 2), призначеної для реалізації гнучкого методу тестування.

Автоматизована система для гнучкого тестування складається з двох основних складових, що відрізняються за типом роботи та метою застосування (рис. 3) – середовища розробки тестів «Semantic Test Serializer» й середовища гнучкого тестування, що містить плагін «Flexible Testing System». Проміжною ланкою тут виступає пакет тестових завдань у вигляді файлу *.xml. Цей файл формується за стандартами Moodle так, щоб він міг бути завантажений в це середовище стандартними засобами Moodle. Після завантаження ці запитання зберігаються в базі даних Moodle і можуть бути використані будь-якою системою тестування в рамках цього середовища.

Середовище розробки тестів складається з автоматизованої системи розробки тестів та бази даних навчальних курсів. Автоматизована система розробки тестів призначена для допомоги розробникові тестів та контролю за тим, щоб тестові завдання були збалансовані за типом запитання, рівнем складності та

рівномірно покривали предметну область, знання з якої належить перевірити.

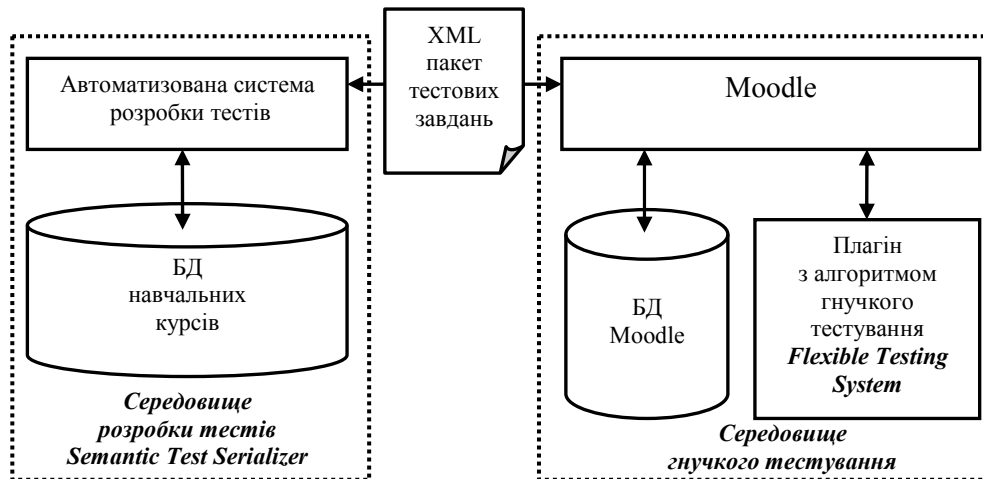


Рис. 2. Структура автоматизованої системи для гнучкого тестування

Семантична рівномірність покриття навчального матеріалу тестовими завданнями контролюється шляхом їх співставлення елементам структури навчального матеріалу, які формують онтологію навчального матеріалу. При цьому співставлення проводиться не з власне елементами онтології, а з ключовими термінами (поняття, визначення) в рамках структури навчального матеріалу, формуючи разом з нею повну онтологію [10] навчального матеріалу (рис. 4).

З семантичної точки зору, ключові терміни є найнижчим рівнем повної онтології навчального матеріалу. Їх особливістю є наявність властивості рівня прив'язки; адже поряд із термінами, що використовуються суто в рамках певних підтем, інші терміни можуть бути актуальними в масштабах тем, модулів, і навіть усього навчального матеріалу. Структура навчального матеріалу визначає місцезнаходження визначених термінів, але не включає їх безпосередньо. Для цього система розробки тестів за допомогою алгоритму аналізу текстів на базі дисперсної оцінки [11] визначає ключові терміни в заданому навчальному матеріалі та пропонує розробнику створювати запитання відповідно до цих термінів. Запропонована система є автоматизованим редактором тестових завдань, що дозволяє зберігати їх у Moodle-сумісному *.xml файлі, додаючи до обов'язкових властивостей тестових завдань дані про пов'язані терміни та оцінку складності. Інформація про пов'язані з тестовим завданням терміни відповідно до їх оцінки визначає глибину засвоєння відповідного матеріалу.



Рис. 3. Функціональна схема автоматизованої системи для гнучкого тестування

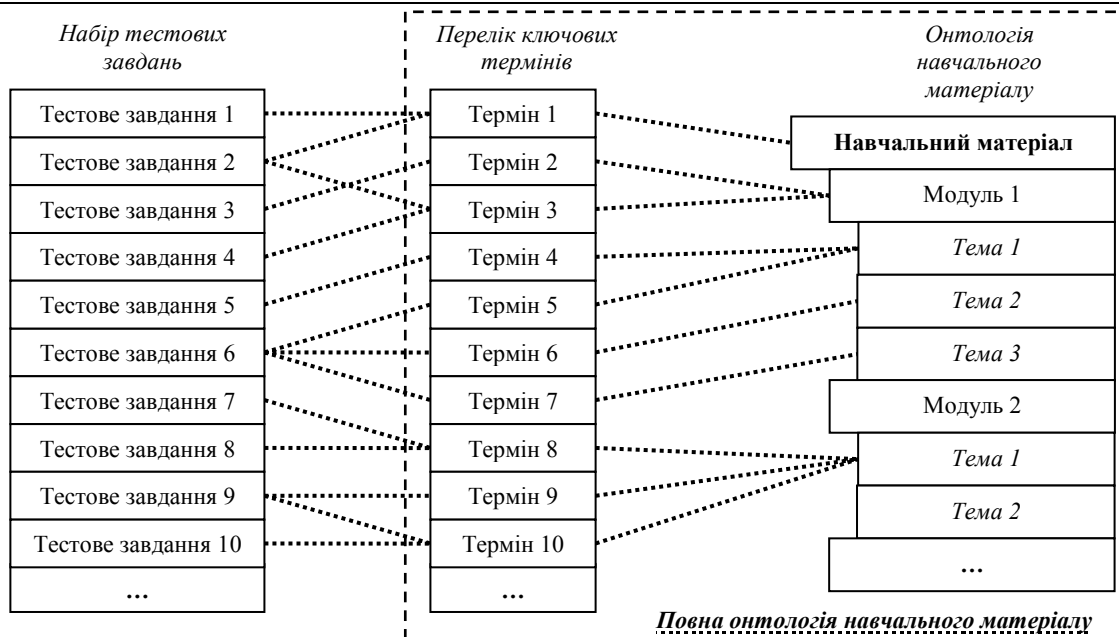


Рис. 4. Фрагмент повної онтології опису навчальної дисципліни

База даних навчальних курсів призначена для зберігання навчальних матеріалів, переліків термінів до них та тестових завдань з усіма властивостями, розробка яких проводиться. Запропоновано представлену на рисунку 5 даталогічну модель бази даних навчальних курсів.

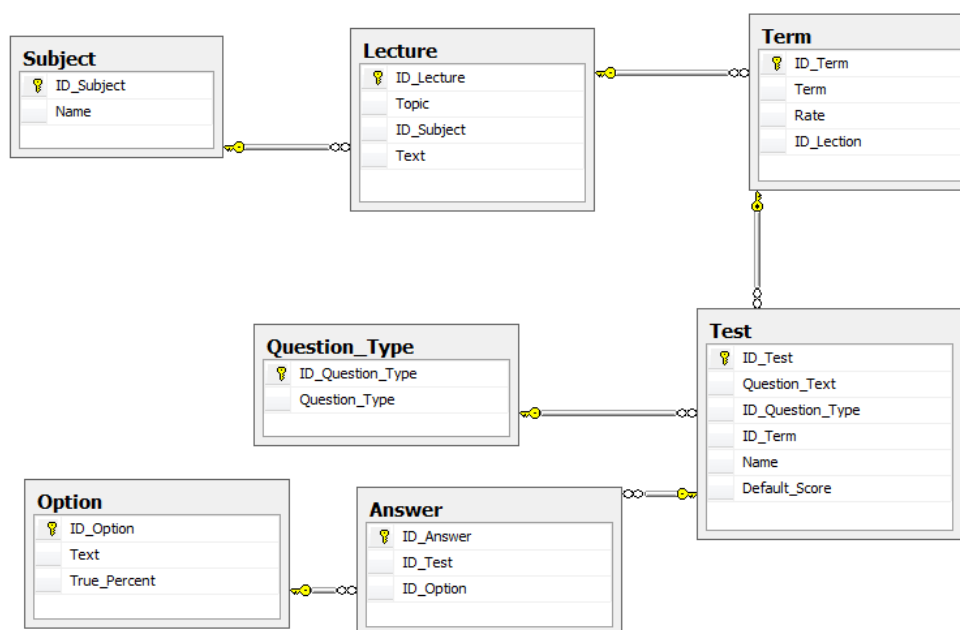


Рис. 5. Структура бази даних середовища розробки тестів

Наведена структура бази даних середовища розробки тестів призначена не тільки для збереження вхідних даних, а й даних проміжних етапів роботи. Так, таблиця «Subject» призначена для об'єднання групи лекцій за предметом, до складу якого вони входять, в одну структуру. Інформація про тему лекції та її контент зберігаються у таблиці «Lecture». Кожна з лекцій містить терміни, які студент повинен засвоїти, й розуміння яких відповідно слід перевірити тестом. Для збереження переліку цих термінів, їх відповідності певній лекції та рівня складності (що відповідає важливості терміна в рамках лекції) використовується таблиця «Term». В процесі роботи з цим модулем користувач створюватиме тестові завдання, параметри та умови використання яких зберігаються у таблиці «Test». Вона також містить інформацію про тип запитання, відповіді, термін що перевіряється, назву завдання, нумерацію відповідей. Текст відповіді зберігається у таблиці «Option» із зазначенням відсотку правильності кожного варіанта. Поеднувати декілька варіантів відповідей з тестовим завданням призначена таблиця «Answer».

Середовище гнучкого тестування включає в себе безпосередньо платформу Moodle й відповідну базу даних, а також пагін, що реалізує алгоритм гнучкого тестування.

Для визначення рівня здобутих знань в системі Moodle є вбудований модуль, що дозволяє

формувані тестові запитання та проводити тестування. Власний алгоритм Moodle, що проводить тестування, не є оптимальним, адже він проводить вибірку запитань випадковим чином. В результаті такого тестування не можна отримати повної картини про рівень та глибину знань учня, що проходив дане тестування. Алгоритм вибору запитань випадковим чином є досить примітивним і не забезпечує об'єктивності перевірки. Використовуючи його, не можливо гарантувати, що у тестовій вибірці будуть обрані запитання різної складності та з різних лекційних блоків.

Система Moodle є системою з відкритим програмним кодом, що дозволяє дописувати модулі для розширення її функціоналу, наприклад, окремих програмних блоків – плагінів, які інтегруються в систему та додають нові можливості [12]. Існує понад 40 різновидів відомих плагінів, що додають Moodle різний функціонал. Так, існують плагіни, що формують різного типу звіти, створюють нові політики безпеки, формують статистику, створюють тести, тощо. Таким чином, задача модифікації базового алгоритму проведення тестування Moodle може бути вирішена шляхом розробки відповідного плагіну.

Для вирішення поставленої задачі, було розроблено алгоритм (рис. 6), ключовими особливостями якого є наступні:

- охоплення максимальної кількості лекційних блоків;
- низхідний аналіз, що починається з перевірки розуміння найбільш семантично вагомих термінів, й поступово переходить до менш важливих термінів;
- мінімізація затраченого на процес тестування часу при збереженні об'єктивного оцінювання рівня знань.

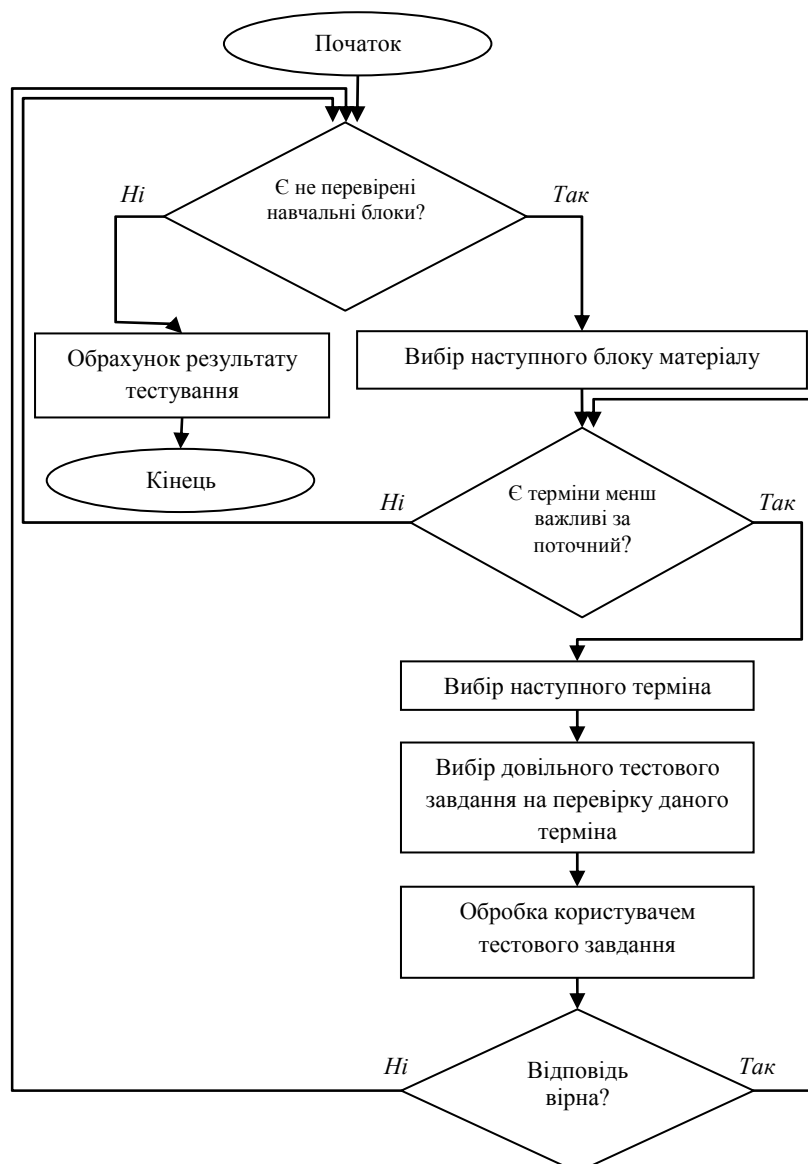


Рис. 6. Алгоритм гнучкого вибору тестових запитань

Наведений алгоритм гнучкого вибору тестових запитань дозволяє проводити тестування по максимальній кількості лекційних блоків, розпочинаючи тестування від найбільш загального терміну. Складність запитань збільшується у разі правильної відповіді. Якщо ж користувач не спроможний дати вірну відповідь навіть на найлегше запитання, то немає сенсу продовжувати опитування блоку та перевіряти

знання на складніших запитаннях. Тому повна перевірка знань поточного лекційного блоку закінчується достроково у разі невірних відповідей або ж після правильних відповідей на усі запитання, включаючи найскладніші. У разі завершення тестування по одному блокові, відбувається перехід до тестування наступного, якщо такий є. Тест визнається пройденим у випадку, коли оброблені усі лекційні блоки. Оцінка за тест визначається як оцінка за кожен лекційний блок, помножена на ваговий коефіцієнт цього блоку, що визначається розробником навчального курсу.

Реалізація функціональності модуля динамічного тестування потребує збереження тестових завдань, налаштувань тесту, та результатів тестування у базі даних. Система Moodle має свою базу, яка дозволяє додавати до неї таблиці, необхідні для роботи нового модуля. Структура таблиць бази даних, що необхідні для роботи модуля динамічного тестування зображенні на рисунку 7.

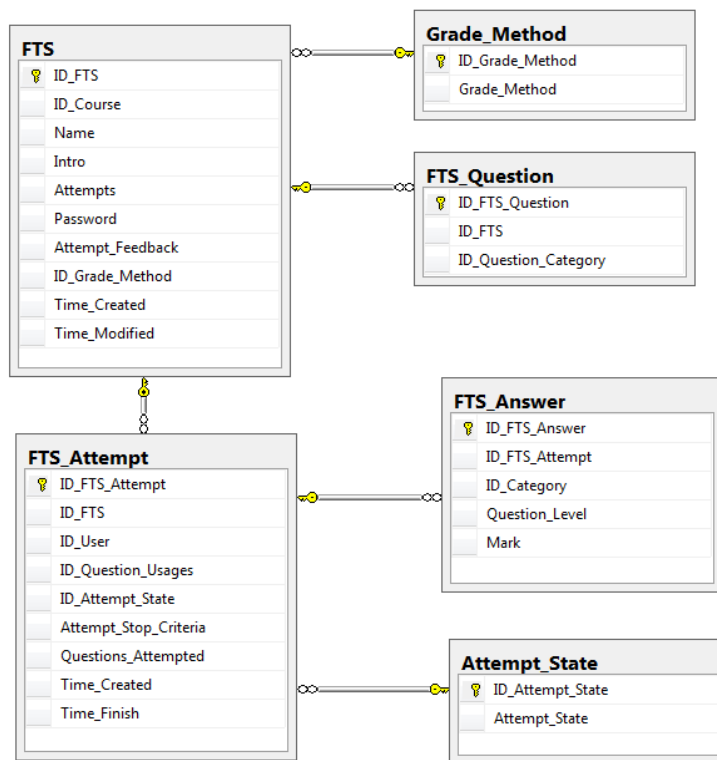


Рис. 7. Структура додаткових таблиць БД середовища Moodle

Розроблений фрагмент структури бази даних має тісну інтеграцію з системними таблицями Moodle. Таблиця «FTS» є однією з ключових таблиць модуля й зберігає інформацію про створенні тестові модулі плагіну у системі Moodle, та містить посилання на таблицю «Course» системи Moodle, що зберігає створені у системі курси. Таблиця «Grade_Method» містить можливі варіанти визначення оцінки тесту. «FTS_Question» є розвідною таблицею, що визначає зв'язок між створеними тестовими модулями плагіну та категоріями запитань, по яким передбачено проведення тестування (системна таблиця «Category»). Призначенням «FTS_Attempt» є збереження результатів кожної спроби. Дана таблиця містить зв'язки із модулем, в якому відбувається дана спроба, системними таблицями користувачів та запитань. Також в ній зберігається інформація про початок та завершення спроби, причину припинення та кількість тестових запитань, що було задано. Результати відповідей на кожне запитання зберігаються в таблиці «FTS_Answer». Вона зберігає посилання на спробу та категорію запитань, з якої було обране завдання, а також рівень складності завдання та результат відповіді, даної користувачем.

Отже, у запропонованій архітектурі системи для гнучкого тестування середовище гнучкого тестування засобами Moodle реалізує адаптивний механізм тестування й рівномірне покриття тестом структури навчального матеріалу, а середовище розробки тестів призначене для формування наборів тестових завдань, що містять необхідний для гнучкого тестування набір параметрів.

Програмна реалізація

З метою перевірки ефективності запропонованої технології, її складові були реалізовані у вигляді тестових програмних продуктів.

Автоматизована система розробки тестів «Semantic Test Serializer», реалізована з використанням мови програмування C# на платформі .NET Framework, дозволяє автоматизувати створення тестових запитань та контролювати рівномірність покриття тестовим набором навчального матеріалу. Також система допомагає розробникові тестів сформувати набір, тестові завдання якого збалансовані за типом запитання та рівнем складності. Для зберігання даних навчальних матеріалів, переліків термінів до них та тестових завдань і їх властивостей використовується база даних у СКБД Microsoft SQL Server розглянутої вище

структури.

Завантажені навчальні матеріали зберігаються в базі даних у вигляді байтового масиву, що описує формат файлу *.rtf. Такий вигляд збереження інформації, на відміну від звичайного текстового, дозволяє зберігати цифрові документи у форматі, включаючи заголовки, шрифти, картинки, таблиці та ін. (рис. 8). В базі даних зберігаються як вхідні, так і вихідні, так і проміжні етапи роботи, що покликане зробити процес розробки тесту більш комфортним. З'єднання програми з базою відбувається одноразово і використовується протягом роботи усієї програми.

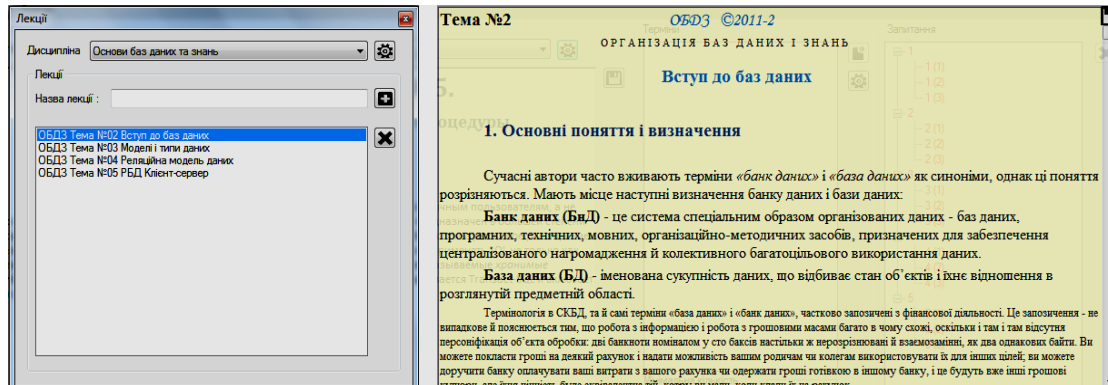


Рис. 8. Інтерфейс занесення навчальних матеріалів у систему розробки тестів

Збережена таким чином інформація, читається з бази даних зворотнім чином: вона перетворюється в файл з розширенням *.rtf, після чого у текстовому вигляді присвоюється елементам, що підтримують відформатований текст.

Кожен структурний елемент навчального матеріалу (модуль, тема, лекція тощо) має свій впорядкований перелік термінів, які в ньому розглядаються. Для цього було використано алгоритм для знаходження ключових термінів навчального матеріалу (рис. 9) за допомогою алгоритму аналізу текстів на базі дисперсної оцінки [11]. Більш важливі терміни мають більше значення дисперсної оцінки. Оскільки кожне запитання, що створюється, автоматично співставляється мінімум одному із знайдених термінів та зберігає даний зв'язок, вдається не лише повністю покрити навчальний матеріал запитаннями, а й визначити рівень складності кожного запитання відповідно до значення дисперсної оцінки терміна, що ним перевіряється.

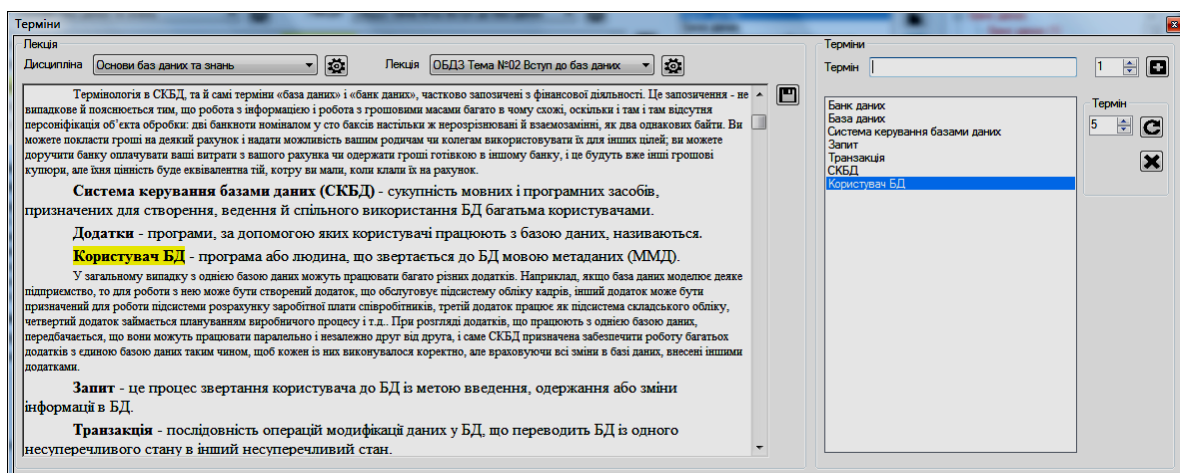


Рис. 9. Інтерфейс роботи з термінами системи розробки тестів

При розробці тестових завдань система автоматично визначає мінімальну кількість запитань, яка необхідна для покриття навчального матеріалу і проведення об'єктивного тестування, їх співвідношення з всіма елементами повної онтології навчального матеріалу (рис. 10).

Система Moodle дозволяє імпортувати та експортувати тестові завдання у файл, що було використано для забезпечення завантаження розроблених тестових завдань. За результатами формуванні тестової вибірки генерується файл формату *.xml зі структурою, що підтримується Moodle. У подальшому цей файл може бути завантажений у систему Moodle за опцією «Імпорт» (рис. 11). Завантажені завдання можуть використовуватися системою для організації тестових перевірок знань. При збереженні запитання маркуються програмою «Semantic Test Serializer» таким чином, щоб зберегти взаємозв'язок між запитаннями та складністю терміна, який воно перевіряє. В подальшому розроблений плагін Moodle використовує це маркування для визначення складності запитання та вибору наступних запитань.

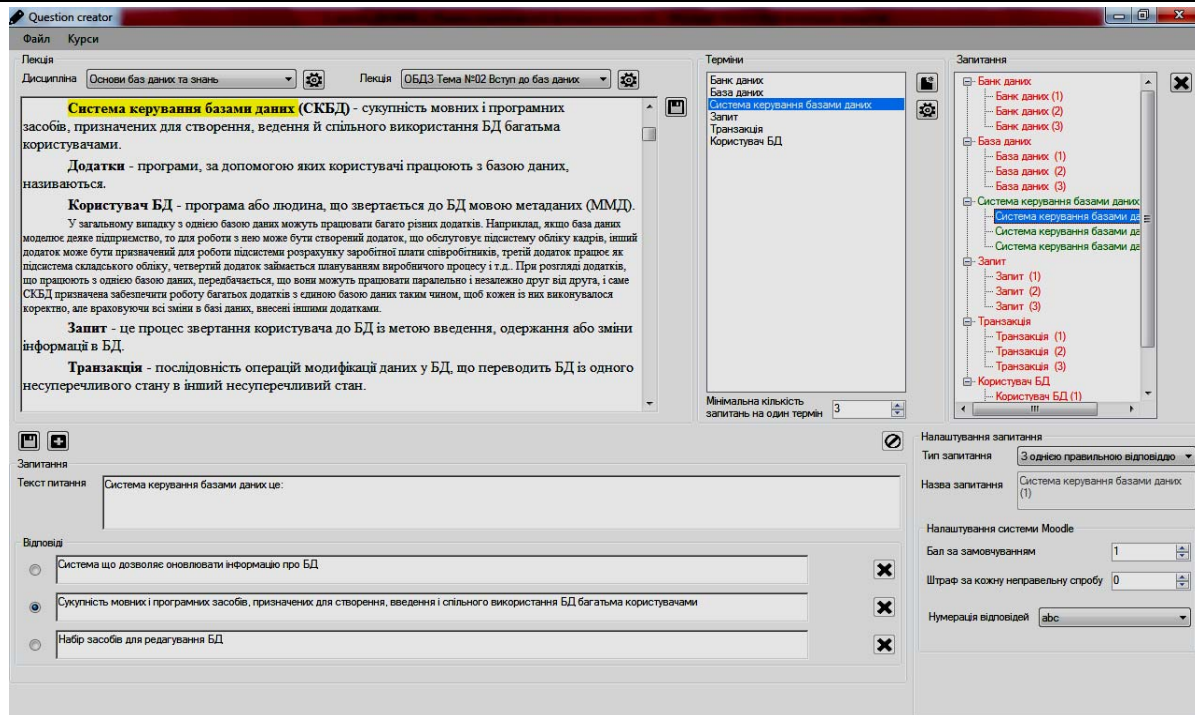


Рис. 10. Інтерфейс розробки тестів та навігації в онтології

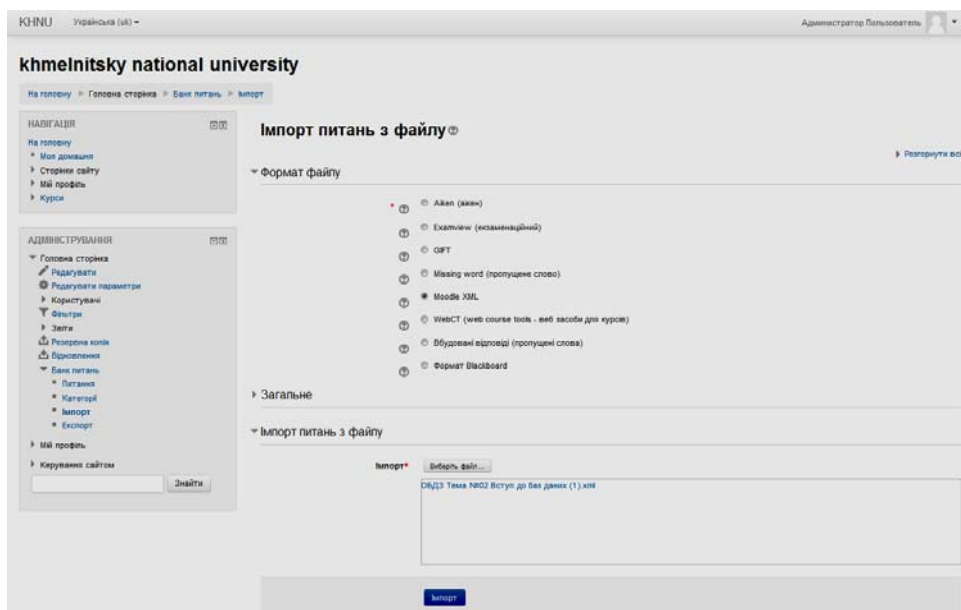


Рис. 11. Імпорт тестових завдань у середовище Moodle

Усі тестові завдання у системі Moodle мають спільні властивості. До них відносяться: назва запитання, його текст та перелік варіантів відповідей. Moodle підтримує велику кількість різних типів тестових завдань, окрім цього нові види запитань можуть бути розроблені та додані у систему в якості плагінів. Кожне запитання повинно містити перелік відповідей. В залежності від виду завдання, набір властивостей його відповідей може бути різний. Однак, всі запитання повинні мати обов'язковий перелік полів: текст відповіді, формат та відсоток який характеризує наскільки даний варіант відповіді є вірним. Усі види тестових завдань мають ряд обов'язкових властивостей які можуть бути розширені даним типом запитань. До обов'язкових властивостей відносять: бал завдання, штраф за кожну наступну спробу та перелік тегів.

Наведений вище алгоритм гнучкого тестування, який базується на аналізі попередніх відповідей користувача, було інтегровано в середовище Moodle шляхом розробки відповідного *плагіна для гнучкого тестування* «Flexible Testing System» (рис. 12). У Moodle передбачена можливість створення плагінів, що модифікують процес тестування та аналіз відповідей, реалізуючи нові ідеї та алгоритми. Так як вся система написана за допомогою технології PHP, то і написання плагінів до неї теж здійснюється з використання цієї технології. Для створення різних типів плагінів потрібно обов'язково наслідувати і перевизначати існуючі класи та методи Moodle, що дає можливість системі взаємодіяти та інтегруватися з значною кількістю нових плагінів.

Встановити модуль з ZIP пакету

Тип модуля* Модуль (mod)

✓ Каталог, D:\xaking\moodle\server\moodle\mod, призначений для модулів вказаного типу доступний для запису.

ZIP пакет* Вибірть файл...

Flexible Testing System.zip

Підтвердження* ☒ Я розумію, що перед встановленням додаткових модулів потрібно створити повну резервну копію цього сайту. Я розумію та погоджуюся з тим, що додаткові модулі (в тому числі ті, які отримано з офіційних джерел, а особливо ті, які отримано з неофіційних джерел) можуть містити діри в безпеці, що може призвести до того, що сайт стане недоступним, а також до знищення даних.

[Показати додаткове...](#)

Встановити модуль із ZIP пакету

Рис. 12. Встановлення розробленого плагіну в Moodle

Для встановлення плагіна в Moodle потрібно заархівувати усі файли з програмним кодом в *.zip архів, та за допомогою панелі адміністрування завантажити цей архів у систему. Moodle розархівовує файли в папку, що відповідає типу даного модуля, і в подальшому працює з ними, причому встановлений плагін буде відображатися у списку всіх встановлених модулів системи (рис. 13).

Огляд модулів

Встановлено: 355 | Відключено: 67 | Сторонні: 2

[Показати тільки сторонні](#)

[Перевірити наявність оновлень](#)

Остання перевірка оновлень 22 Березень 2017, 00:15

Назва модуля	Джерело	Версія	Випуск	Доступність	Дії	Нотатки
Модулі діяльності						
Flexible Testing System mod_adaptivequiz	Розширення	2017020805	2.5-1.1 (Build 2017020800)	Включено	Видалити	
Завдання mod_assign	Стандарт	2014111001		Включено	Налаштування	Потрібно для: tool_assignmentmenu
Книга mod_book	Стандарт	2014111000		Включено	Налаштування Видалити	

Рис. 13. Огляд переліку встановлених плагінів у середовищі Moodle

Інтегрування в навчальний ресурс середовища Moodle тесту з алгоритмом гнучкого тестування відбувається вибором даного плагіну зі списку доступних елементів курсу (рис. 14).

Програмні модулі плагіну впливають на ряд властивостей середовища Moodle, що відображено на рисунку 15. Модуль спроби кожного разу коли користувач розпочинає тестування запускає відповідний процес, робить запис в базу даних про час початку, користувача тощо. Модуль завершення тесту модифікує останню сторінку при проходженні тесту, на якій висвітлюється перелік тестів і є кнопка «Відправити все та завершити». Модуль завершення впливає на сторінку, яка записує результати тесту в БД і редиректує користувача на головну сторінку курсу; в даному випадку гнучкого тестування, користувач не може повертатися до попередніх тестів і запис про результат кожного тесту відбувається одразу після натиску на кнопку «Наступне запитання», тому ця сторінка просто записує дату завершення тесту і повертає користувача на головну сторінку. Модуль логіки тестування використовується системою кожного разу, щоб визначити рівень складності наступного запитання, переконатися чи користувач не вичерпав ліміт своїх завдань (не досягнув кінця тесту). Модуль-контейнер інформації про тестову спробу використовується для показу результатів тестування, дозволяє будувати графіки на основі даних, містить інформацію про кількість заданих запитань, середню складність, список правильних і неправильних відповідей, викладач може переглянути як студент відповідав на запитання. Модуль для видалення спроби користувача дозволяє викладачу зайти в систему і видалити певні результати спроб користувача. Модуль витягування завдань з БД забезпечує на основі даних з модулю логіки витягування запитання відповідної складності та відповідної категорії чи типу з БД Moodle. Навіть якщо БД містить багато завдань однієї складності, то цей клас обирає завдання випадковим чином, але водночас слідкує, щоб воно не повторювалося. Модуль оцінок формує

сторінку, що виводить список студентів, їхні спроби та оцінки за кожну спробу. Модуль налаштувань дозволяє розробнику задати наступні налаштування для курсу: ім'я, кількість доступних спроб, пароль для того щоб розпочати спробу, спосіб визначення кінцевої оцінки (найкращий результат/середній), обирає групу запитання що вже завантаженні в Moodle і з цієї групи обирає запитання під час тесту. Модуль перегляду результатів тестування забезпечує завантаження завдань всього тесту й відповіді, які обрав користувач, і правильні відповіді.

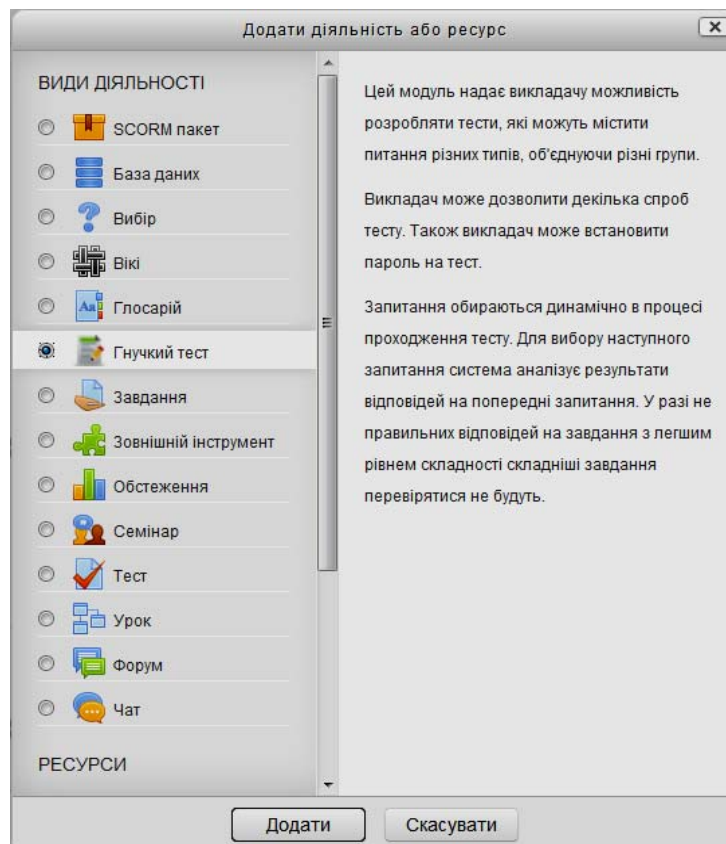


Рис. 14. Додавання в навчальний ресурс елементу гнучкого тестування в середовищі Moodle

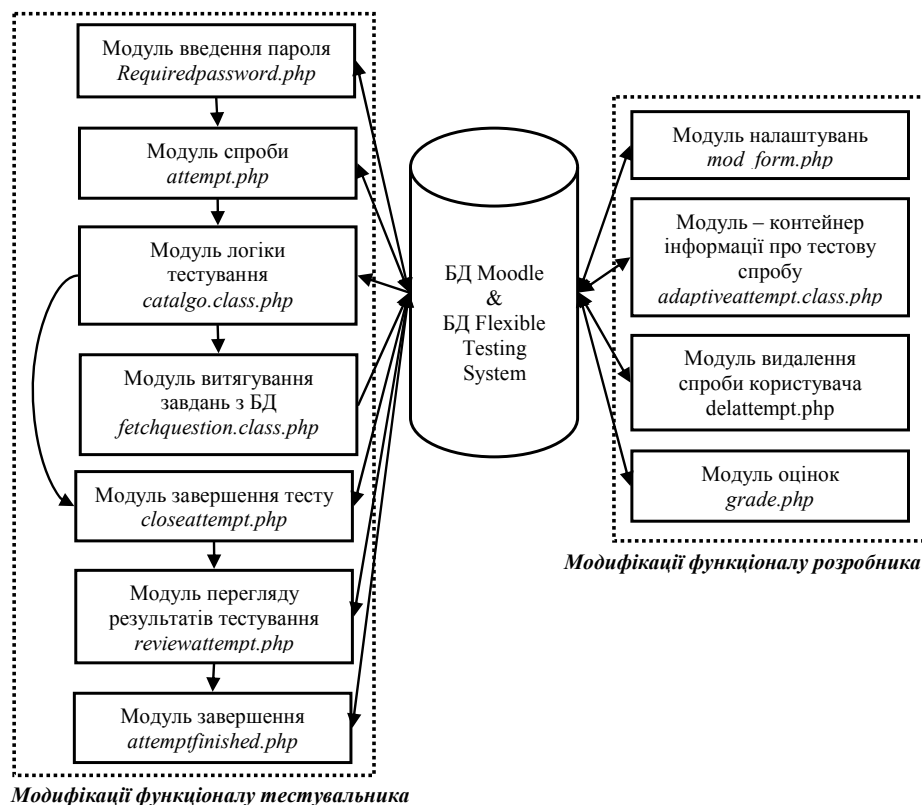


Рис. 15. Схема взаємодії компонентів плагіну для гнучкого тестування

Усі тестові завдання в середовищі Moodle мають спільні властивості. До них відносяться: назва запитання, його текст та перелік варіантів відповідей. Moodle підтримує велику кількість різних типів тестових завдань, окрім цього нові види запитань можуть бути розроблені та додані у систему в якості плагінів. Кожне запитання повинно містити перелік відповідей. В залежності від виду завдання, набір властивостей його відповідей може бути різний. Однак, всі запитання повинні мати обов'язковий перелік полів: текст відповіді, формат та відсоток який характеризує наскільки даний варіант відповіді є вірним. Усі види тестових завдань мають ряд обов'язкових властивостей, які можуть бути розширені даним типом запитань. До обов'язкових властивостей відносять: бал завдання, штраф за кожну наступну спробу та перелік тегів.

Розроблений плагін використовує маркування тестових завдань для визначення складності запитання та вибору наступного запитання необхідної складності. Плагін задає запитання послідовно (рис. 16) для того, щоб аналізувати відповідь на попереднє запитання та в залежності від його результату обирати наступне. Відтак, користувач не знає, на яку кількість запитань йому ще доведеться дати відповідь.

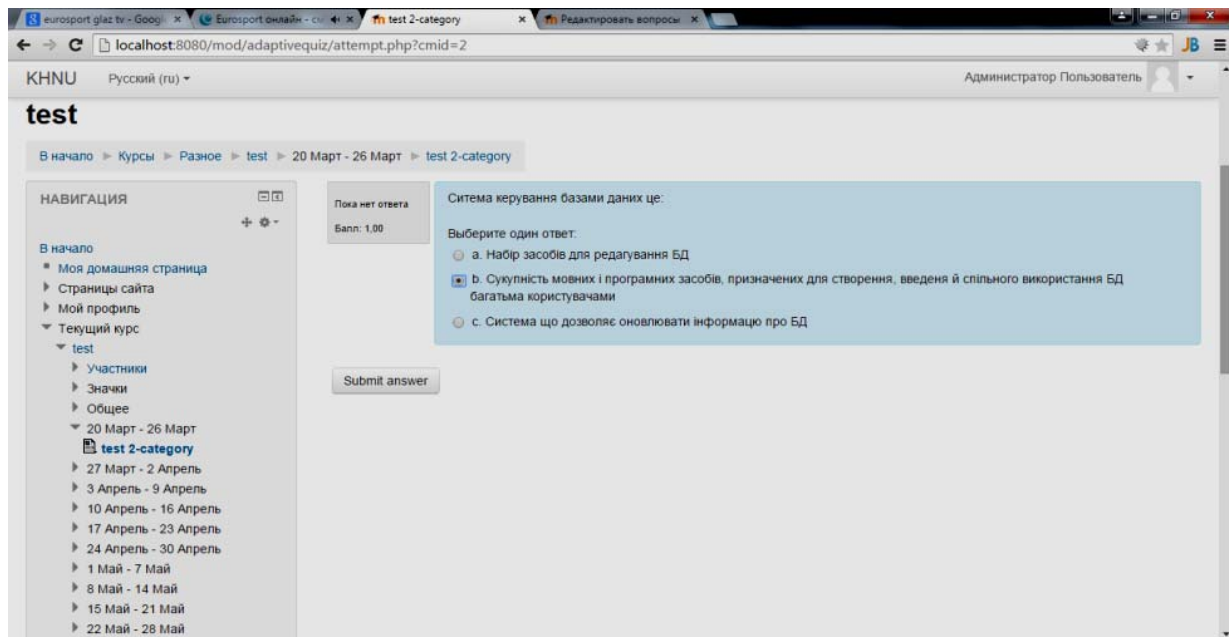


Рис. 16. Приклад гнучкого тестування з використанням розробленого плагіна у середовищі Moodle

Для проходження тесту потрібно обрати створений елемент у відповідному курсі. Створений модуль організовує перевірку знань за алгоритмом гнучкого тестування, й система задаватиме запитання, опираючись на результати відповідей. Алгоритм гнучкого тестування дозволяє більш повноцінно охопити весь навчальний матеріал, завчасно припинити тестування у разі якщо користувач не дав правильної відповіді на найлегші запитання, а також максимально точно визначити рівень знань людини що проходить тестування. Результат тесту відображається відразу після його завершення. Також він доступний для перегляду викладачу та адміністратору системи.

Дискусія

В результаті роботи було розроблено два програмних продукти, які дозволяють створювати тестові завдання та реалізують алгоритм гнучкого тестування. Розроблена система є більш гнучкою за існуючі аналоги, оскільки обирає запитання не випадковим чином, а аналізуючи їхню складність та відповіді користувача на попередні запитання [13].

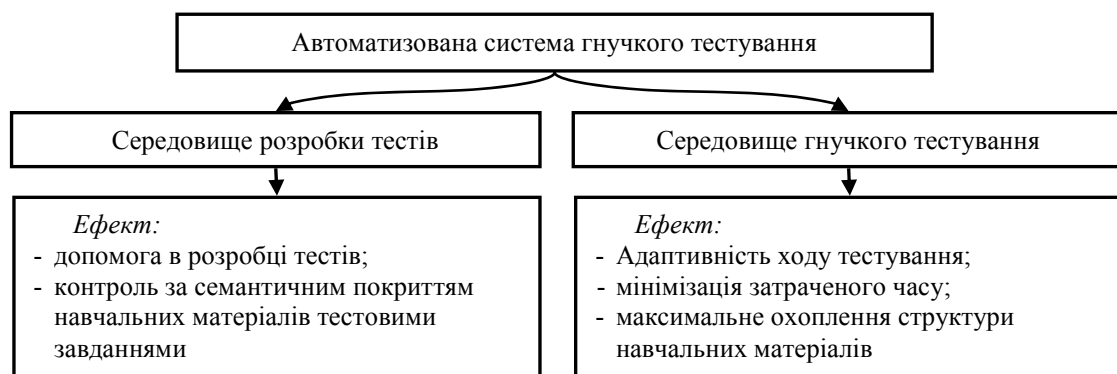


Рис. 17. Ефект від застосування інформаційної технології гнучкого тестування рівня знань

Автоматизована система гнучкого тестування містить дві основні підсистеми, кожна з яких дає свій корисний ефект (рис. 17).

На об'єктивність результатів суттєво впливає правильність сформованих тестових запитань. Адже для створення коректного тестового набору, тестові завдання формуються так, щоб вони рівномірно покривали навчальний матеріал.

На коректність сформованого тестового набору впливає як суб'єктивний фактор розробника й якості навчального матеріалу (який природно виступає сировиною для формування контенту тестових завдань), так і ефективність автоматизованої побудови повної онтології предметної області [14].

При коректно сформованому наборі тестових запитань, запропонований підхід дозволяє більш точно визначити глибину знань та виявити прогалини в розумінні вивченого матеріалу.

Висновки

В статті було досліджена проблема підвищення ефективності тестування рівня отриманих знань із використанням комп'ютерних засобів. Розглянуто інформаційну технологію гнучкого тестування рівня знань, що формує репрезентативні набори тестових завдань та адаптивно обирає тестові завдання в процесі тестування. Представлено тестові програмні продукти, що в комплексі з середовищем Moodle реалізують запропоновану інформаційну технологію гнучкого тестування й дозволяють підтвердити її високу практичну та наукову цінність.

Подальші дослідження спрямовані на вдосконалення методів та технологій, що забезпечують автоматизоване визначення впорядкованих переліків ключових термінів та побудову повної онтології предметної області.

Література

1. Нові інформаційні технології в освіті [Електронний ресурс]. – Режим доступу : <http://it-tehnolog.com/statti/novi-informatsiyni-tehnologiyi-navchannya/>.
2. Аванесов В. С. Композиция тестовых заданий / В. С. Аванесов. – М. : Центр тестирования, 2002.
3. Комп'ютерна програма тестування знань OpenTEST 2 [Електронний ресурс]. – Режим доступу : <http://opentest.com.ua/kompyuternaya-programma-testirovaniya-znaniy-opentest-2/>
4. Контрольно-діагностична система Test-W2 [Електронний ресурс]. – Режим доступу : http://teach-inf.at.ua/load/programi/testi/test_w2_kontrolno_diyagnostichna_sistema/16-1-0-7
5. Снитюк В. Е. Интеллектуальное управление оценением знаний / В. Е. Снитюк, К. Н. Юрченко. – Черкассы, 2013. – 262 с.
6. Даревич Р. Р. Підвищення ефективності інтелектуального аналізу тесту шляхом зважування понять моделі онтології / Р. Р. Даревич // Штучний інтелект. – 2005. – № 3. – С. 571–577.
7. Зайцева Л. В. Моделі та методи адаптивного контролю знань / Л. В. Зайцева, Н. О. Прокоф'єва // Educational Technology & Society. – 2004. – № 7 (4). – С. 265–277.
8. Катеринчук И. С. Интеллектуальная система автоматизированного контроля знаний студентов высших учебных заведений / И. С. Катеринчук, Р. В. Рачок, В. В. Кравчук, В. М. Кулик // Информационные технологии в образовании : сборник научных трудов. – Херсон : Изд-во ХГУ, 2009. – Выпуск 4. – С. 139–147.
9. Moodle – вільна енциклопедія [Електронний ресурс]. – Режим доступу : <http://uk.wikipedia.org/wiki/Moodle>
10. Поліщук А. О. Інформаційна технологія автоматизації аналізу відповідності тестових завдань лекційним матеріалам навчальних дисциплін / А. О. Поліщук, О. В. Мазурець // Збірник наукових праць за матеріалами дев'ятої міжнародної науково-технічної конференції «Актуальні проблеми комп'ютерних технологій 2015». – Хмельницький, 2015. – С. 207–218.
11. Бармак О. В. Інформаційна технологія автоматизованого визначення термінів у навчальних матеріалах / О. В. Бармак, О. В. Мазурець // Вимірювальна та обчислювальна техніка в технологічних процесах. – Хмельницький, 2015. – № 2. – С. 94–102.
12. Особливості розробки плагінів до Moodle [Електронний ресурс]. – Режим доступу : https://docs.moodle.org/dev/Main_Page
13. Матвійчук А. О. Інформаційна технологія гнучкого тестування рівня знань / А. О. Матвійчук, О. В. Мазурець // Збірник наукових праць за матеріалами дев'ятої міжнародної науково-технічної конференції «Актуальні проблеми комп'ютерних технологій 2015». – Хмельницький, 2015. – С. 158–169.
14. Крак Ю. В. Практичне дослідження ефективності інформаційної технології автоматизованого визначення семантичних термінів в контенті навчальних матеріалів / Ю. В. Крак, О. В. Бармак, О. В. Мазурець // Прикладне програмне забезпечення : збірник наукових праць за матеріалами десятої міжнародної науково-практичної конференції по програмуванню «УкрПРОГ'2016». – К., 2016. – С. 237–245.

Рецензія/Peer review : 27.3.2017 р.

Надрукована/Printed : 19.4.2017 р.

Рецензент: д.т.н., проф. Сорокати Р.В.