

**Олег Дубровін**, кандидат технічних наук, доцент

*Дрогобицького державного педагогічного університету  
імені Івана Франка*

**Максим Дубровін**, студент

*Національного університету "Львівська політехніка"*

## **РОЗРОБКА СТРУКТУРИ УНІВЕРСАЛЬНОЇ ТЕСТУЮЧОЇ СИСТЕМИ**

*У статті розглянуто задачу проектування універсальної комп'ютерної системи тестування з використанням сучасної клієнт-серверної бази даних.*

**Актуальність теми.** У сучасній школі – як у середній, так і вищій, з кожним десятиліттям суттєво зростає обсяг інформації з яким учневі необхідно ознайомитися і засвоїти. Це вимагає від системи освіти нових ефективних форм та методів навчання, які базуються на широкому використанні сучасних технічних засобів. До таких засобів належить сучасний комп'ютер. Комп'ютеризація сфери освіти – невід'ємний і дуже важливий етап побудови інформаційного суспільства, створення його інформаційної інфраструктури. Діапазон використання комп'ютерів у школі досить широкий – від тестування учнів, обліку їх успішності, проведення лабораторних експериментів до ігор. У середній та вищій школі комп'ютер виступає як:

1. Об'єкт вивчення – учні вивчають його, здобувають основні практичні вміння та навички роботи з комп'ютером – оволодівають комп'ютерною грамотністю.

2. Засіб навчання – комп'ютер є потужним засобом підвищення ефективності навчання, що досягається використанням спеціально розроблених комп'ютерних програм, які керують навчальною діяльністю учня. З широким поширенням протягом останнього десятиліття глобальної мережі Internet комп'ютерні навчальні системи набрали нової якості в системі дистанційного навчання. Окрім навчально-контролюючих функцій, комп'ютер може виступати в ролі електронного довідника та спеціального тренажера.

Окремим видом навчально-контролюючих систем є автоматизовані системи тестування. Останні роки вони набувають широкого поширення завдяки впровадженню модульно-рейтингової системи навчання. На кожній кафедрі вищого навчального закладу розробляються десятки і навіть сотні різних тестових завдань для оцінки рівня підготовки студентів. При цьому, ряд тестових завдань певною мірою дублюються. Як

приклад, розглянемо тести, які використовуються в конкретних модулях. При здачі заліка та екзамену частина цих тестів знову включається до опитування. І, нарешті, при підсумковому (бакалаврському) екзамені, який проводиться з профільних предметів спеціальності, тести з окремих модулів (назвемо їх базовими) входять до складу результуючого тесту.

В описаному вище випадку найбільш доцільним, на наш погляд, є тестування з використанням комп'ютерної техніки. Весь необхідний базовий набір тестів при цьому слід зберігати в клієнт-серверній базі даних, а всі похідні тести формувати за допомогою спеціально розробленого інтерфейсу.

**Мета статті.** Розглянути основні етапи проектування сучасної клієнт-серверної бази даних, яка дозволяє ефективно маніпулювати базовими наборами тестів, формуючи з них нові завдання для контролю знань студента.

**Виклад основного матеріалу.** Проектування баз даних складається з трьох етапів [1], [2], [4]:

1. Концептуальне проектування – створення концептуального представлення бази даних, яке включає визначення типів найважливіших сутностей та існуючих між ними зв'язків.

2. Логічне проектування – перетворення концептуального подання у логічну структуру бази даних, включаючи проектування відношень.

3. Фізичне проектування – прийняття рішень про те, як логічна модель буде фізично реалізована (за допомогою таблиць) у базі даних, яка створюється за допомогою вибраної СУБД.

**Концептуальна модель бази даних.** У системі тестування усі інформаційні блоки даних повинні зберігатися у базі даних. Для проектування цієї бази даних спочатку необхідно побудувати концептуальну модель. В результаті попереднього дослідження системи предметної області тестування було виявлено такі концептуальні твердження:

## РОЗРОБКА СТРУКТУРИ УНІВЕРСАЛЬНОЇ ТЕСТУЮЧОЇ СИСТЕМИ

1. На кафедрі проводиться тестування студентів.

2. Тестування проводиться з предметів, які викладаються викладачами кафедри для різних спеціальностей (наприклад, “макроекономіка для спеціальності банківський рекет”).

3. Існує декілька форм контролю за знаннями студентів:

- погочний модульний контроль;
- залік;
- екзамен;
- підсумковий екзамен з кваліфікації “бакалавр”.

4. Тести поділяються на дві категорії:

- базові тести, які формуються викладачами для конкретних модулів;  
 - комбіновані тести, які формуються на основі ряду базових тестів (наприклад, для таких форм контролю, як залік та екзамен).

5. Тести містять таку інформацію:

- запитання студентів;
- перший варіант відповіді;
- другий варіант відповіді;

- прізвище студента;
- ім'я студента;
- назву групи;
- дату тестування;
- число правильних відповідей;
- число неправильних відповідей;
- число неподаних відповідей;
- суму набраних балів;
- логін студента.

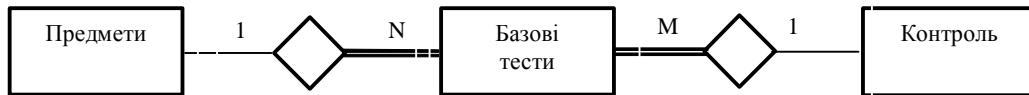
На основі аналізу предметної області бази даних одержуємо такі сутності:

**Предмети** (Код\_Предмета, Назва\_Предмету) – містить навчальні предмети.

**Контроль** (Код\_Контролю, Форма\_Контролю) – містить форми контролю.

**Базові тести** (Код\_Тесту, Тест, Кількість\_Питань) – містить базові тести.

Кожен базовий тест складається для конкретного предмету та конкретної форми контролю, тобто між сутностями “Предмети”, “Базові тести” і “Контроль” існують взаємні зв'язки. Розглянемо ці зв'язки – **Рисунок 1:**



- третій варіант відповіді;
- четвертий варіант відповіді;
- перший номер правильного варіанту відповіді;
- другий номер правильної відповіді;
- третій номер правильної відповіді.

6. Усі студенти, які братимуть участь у тестуванні реєструються.

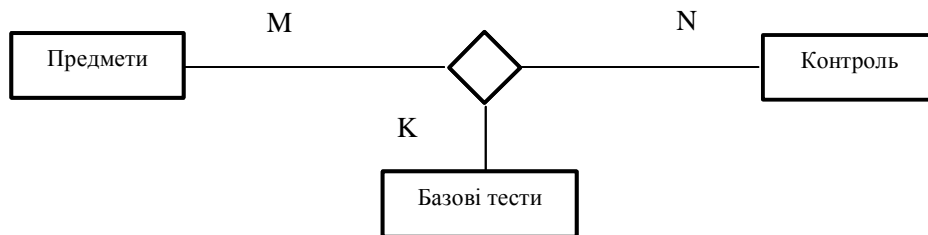
7. Реєстрація містить такі дані про студента:

- логін студента (наприклад, номер студентського білета), за яким він реєструється;
- пароль студента, за яким він проводить тестування;

Кожен базовий тест пов'язаний з одним предметом, предмет може мати декілька базових тестів (для різних форм контролю), тому зв'язок між сутностями “Предмет” та “Базовий тест” – (1 : N). Аналогічно, зв'язок між сутностями “Контроль” та “Базовий тест” – (1 : M).

Між сутностями “Предмети”, “Базові тести” та “Контроль” існує зв'язок, який обумовлює з яких базових тестів формується комбінований тест для заданого предмету та форми контролю. Тип цього зв'язку “багато-до-багатьох” –

**Рисунок 2:**



- прізвище студента;
- ім'я студента;
- назва групи студента.

8. У процесі тестування студенти відповідають на запитання. Дані про результати зберігаються. Вони містять такі відомості:

- предмет тестування;
- форма тестування;

Для проведення тесту і фіксації результату використовуються такі сутності:

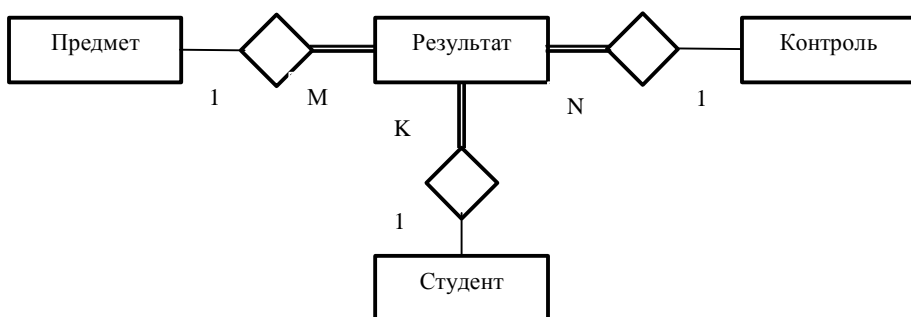
**Студент** (Логін, Прізвище, Ім'я, Група, Пароль)

**Результат** (Номер, Число\_Правильних\_Відп, Число\_Неправильних\_Відп, Число\_Неподаних\_Відп, Сума\_балів)

Сутність “Результат” має зв'язки із

## РОЗРОБКА СТРУКТУРИ УНІВЕРСАЛЬНОЇ ТЕСТУЮЧОЇ СИСТЕМИ

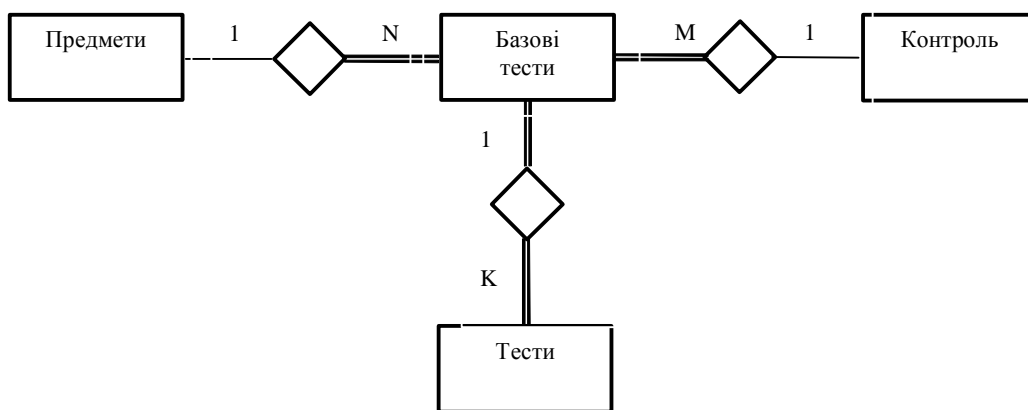
сутностями “Предмет”, “Контроль” та “Студент” “один-до-багатьох” реалізуються шляхом міграції – **Рисунок 3:**



Сутність “Базові тести” містить множинний атрибут “Тести”, який складається зі списка запитань, відповідей та ін. Перетворимо цей атрибут шляхом виявлення нової сутності “Тести”,

Кожен результат тестування пов’язаний з одним предметом, предмет може входити до багатьох результатів (для різних студентів), тому зв’язок між сутностями “Предмет” та

яка має зв’язок з сутністю “Базові тести” виду (1 : K), оскільки одному коду тесту відповідає багато тестових запитань – **Рисунок 4.**

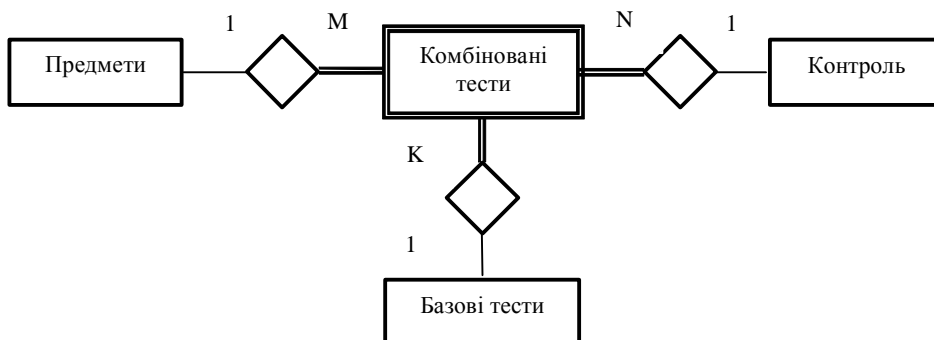


“Результат” – (1 : M). Аналогічно, зв’язок між сутностями “Контроль” – “Базовий тест” – (1 : N) та зв’язок між сутностями “Студент” – “Результат” – (1 : K).

Сутність при цьому “Тест” має вид: **Тест** (Номер тесту, Запитання, Перша відповідь, Друга відповідь, Третя відповідь, Четверта відповідь, Правильна відповідь\_1, Правильна відповідь\_2, Правильна відповідь\_3)

**Логічна модель бази даних.** За логічну модель бази даних вибираємо реляційну модель як найбільш поширену в даний час [2]. В реляційній моделі всі сутності подаються у

Для того, щоб вилучити зв’язок “багато-до-багатьох”, показаний на рисунку 2, вводиться нова слабка сутність “Комбіновані тести” – **Рисунок 5.**



вигляді реляційних таблиць. На етапі логічного проектування зв’язки “багато-до-багатьох” подаються у виді окремої сутності, багатозначні атрибути виділяються в окрему сутність, а зв’язки

Таким чином, остаточна інфологічна модель бази даних “Тест” подається сукупністю рисунків 2, 4, 5. Усі зв’язки “один\_до-багатьох”, які

## РОЗРОБКА СТРУКТУРИ УНІВЕРСАЛЬНОЇ ТЕСТУЮЧОЇ СИСТЕМИ

зображуються на цих рисунках реалізуємо шляхом міграції первинних ключів батьківських сутностей до дочірніх сутностей, де вони стають зовнішніми ключами.

Після виконання цієї операції одержимо такий набір реляційних таблиць з остаточно визначеним набором атрибутів, що відповідають логічній моделі бази даних:

**Предмет** (Код\_Предмету, Назва\_Предмету)  
**Primary Key** (Код\_Предмету)

**Форма\_Контролю** (Код\_Контролю, Форма\_Контролю)  
**Primary Key** (Код\_Контролю)

**Базові тести** (Код\_Тесту, Код\_Предмету, Код\_Контролю, Кількість\_Питань)

**Primary Key** (Код\_Тесту)

**Foreign Key** Код\_Предмету **References** Предмет(Код\_Предмету)

**Foreign Key** Код\_Контролю **References** Предмет(Код\_Контролю)

**Комбіновані Тести** (Код\_Тесту, Код\_Предмету, Код\_Контролю, Кількість)

**Primary Key** (Код\_Тесту, Код\_Предмету, Код\_Контролю)

**Foreign Key** Код\_Тесту **References** Базові\_Тести(Код\_Тесту)

**Foreign Key** Код\_Предмету **References** Предмет(Код\_Предмету)

**Foreign Key** Код\_Контролю **References** Предмет(Код\_Контролю)

**Тест** (Номер\_Тесту, Код\_Тесту, Запитання, 1\_Відповідь, 2\_Відповідь, 3\_Відповідь, 4\_Відповідь, Правильна\_Відп\_1, Правильна\_Відп\_2, Правильна\_Відп\_3)

**Primary Key** (Номер\_Тесту)

**Foreign Key** Код\_Тесту **References** Базові\_Тести(Код\_Тесту)

**Студент** (Логін, Прізвище, Ім'я, Група, Пароль)

**Primary Key** (Логін)

**Результати** (Номер, Логін, Код\_Предмету, Код\_Контролю, Дата, Число\_Правильних\_Відп, Число\_Неправильних\_Відп, Число\_Неподаних\_Відп, Сума балів)

**Primary Key** (Номер)

**Foreign Key** Логін **References** Студент(Логін)

**Foreign Key** Код\_Предмету **References** Предмет(Код\_Предмету)

**Foreign Key** Код\_Контролю **References** Предмет(Код\_Предмету)

Тут **Primary Key** є первинним ключем таблиці, а **Foreign Key** – зовнішнім ключем.

**Фізична модель бази даних.** Базу даних тест будемо реалізувати на клієнт-серверній СУБД “Fire Bird 1.5” [3], яка досить добре описана, без ліцензійно поширюється через Internet і відносно проста в адмініструванні. Вибір клієнт-серверної СУБД зумовлений необхідністю проводити одночасне тестування багатьох студентів з використанням локальної мережі.

Для спроектованої нами клієнт-серверної бази даних необхідно забезпечити доступ:

**1. Адміністратору бази даних**, який створює базу, всі її компоненти, а також слідкує за її цілісністю.

**2. Викладачу**, який заповнює даними усі таблиці, окрім таблиць Resultat та Student, а також формує завдання до чергового тесту.

**3. Студенту**, який реєструється в системі, помістивши свої дані до таблиці Student. Після завершення тестування, його результати повинні автоматично заноситися до таблиці Resultat.

Для кожної із цих задач необхідний відповідний інтерфейс. Як інтерфейс адміністратора СУБД Fire Bird, можуть бути використані стандартні інструментальні програми, такі, як InterBase Console, IBaseExpert та інші.

Інтерфейси викладача та студента повинні розроблятися окремо. За програмний засіб для такої розробки ми вибрали систему візуального проектування Delphi [5]. Такий вибір зумовлюється широким поширенням у нашій країні цього інструментального засобу, а також, що дуже важливо, наявністю саме в цьому програмному середовищі розвинутих засобів для роботи з СУБД InterBase/Fire Bird. Нами було розроблено та відлагоджено два Delphi-проекти, які ми назвали “Монітор викладача” та “Тестовий монітор”, що є засобами інтерфейсу викладача та студента відповідно.

**Висновки.** Тестування студентів у вищих навчальних закладах доцільно проводити у комп’ютерному класі. На рівні кафедр слід розробити набір базових тестових завдань, на основі яких можна формувати комбіновані тести. Набір базових тестів необхідно зберігати у спеціально спроектованій базі даних. Доступ до цієї бази даних повинні мати адміністратори, викладачі та студенти. Для організації такого доступу слід розробити ряд тестових моніторів, які забезпечать дружній інтерфейс у процесі

## ФОРМУВАННЯ ЗНАТЬ ВИПУСКНИКІВ ВИЩИХ НАВЧАЛЬНИХ ЗАКЛАДІВ

підготовки тестових завдань та проведення тестування.

1. Гарсиа-Молина Г., Ульман Д., Уидом Д. Системы баз данных. Полный курс. Вильямс. М., С.Пб, К. 2004.

2. К. Дейт. Введение в системы баз данных. Шестое издание. Вильямс. М., С.Пб, К. 2000.

3. Ковязин А, Востриков С. Мup InterBase. М., 2002.

4. Конноли Т., Бегг К. Базы данных. Проектирование, реализация и сопровождение. Второе издание. Вильямс. М., С.Пб, К. 2001.

5. Тейксейра С., Пачеко К. Delphi 5. Руководство разработчика. Том 2. Вильямс. М., С.Пб, К. 2001.

Володимир Леськів, кандидат технічних наук,  
доцент кафедри основ виробництва  
Дрогобицького державного педагогічного університету  
імені Івана Франка

Дмитро Леськів, магістр

## ФОРМУВАННЯ ЗНАТЬ ВИПУСКНИКІВ ВИЩИХ НАВЧАЛЬНИХ ЗАКЛАДІВ

У статті розглянуто питання впливу основних факторів підготовки випускників вищих навчальних закладів технічного профілю на якість їх знань і запропоновано заходи для її поліпшення.

**Постановка проблеми.** Розвиток науки і техніки, впровадження у виробництво сучасних технологій, особливо нанотехнологій, вимагає досконалої підготовки спеціалістів, зокрема технічних спеціальностей. На жаль, теперішні випускники вищих навчальних закладів не завжди відповідають вимогам часу. Як стверджується [2], відтік здібної молоді від навчання інженерним спеціальностям може стати причиною страшних техногенних ситуацій. Підтвердженням цьому є трагічні події, в Україні і Росії за останні роки.

**Мета статті** – проаналізувати фактори, що впливають на рівень підготовки випускників вищих навчальних закладів (ВНЗ) і вказати можливі шляхи його поліпшення.

**Виклад основного матеріалу.** Суму знань, які може одержати випускник ВНЗ, можна розглядати, як добуток трьох співмножників  $Z_v = (B+M+B)(Z+P)П$ . Перший множник  $(B+M+B)$  складається з трьох величин, що залежать від навчального закладу. Це ці три точки опори, на яких стоїть ВНЗ, розглянемо їх:

**B** – робота викладацького складу включає у себе якість лекцій, практичних і лабораторних занять. Якість роботи викладачів залежить від їх кваліфікації, умов праці і ставлення до роботи, які своєю чергою залежать від навантаження і зарплати. Якщо в 50-ті роки (десять років після такої страшної війни) місячна зарплата доцента к.т.н. з десятирічним стажем роботи при річному

навантаженні 700 – 800 годин в рік становила 3200 крб. (зарплата висококваліфікованого робітника була 800 – 900 крб), то тепер зарплата доцента майже в 2 рази менша, при значно вищій зарплаті в промисловості. У 50 – 80-ті роки доцент міг розраховувати на позачергове одержання житла, тепер це питання практично не розв'язується. Тому в даний час в аспірантуру і на викладацьку роботу у ВНЗ ідуть не завжди найкращі випускники, а ті, які мають у даному населеному пункті житло і їх влаштовує рівень зарплати. Кращі випускники часто ідуть за кордон, або шукають роботу у приватних підприємств. У результаті в США вже кожен четвертий програміст і кожен п'ятий математик – українець, Німеччина офіційно запрошує до себе на роботу десятки тисяч фахівців в галузі електроніки з України [6]. І якщо так буде продовжуватись, то ніякі фінансові вливання не зможуть швидко підняти професіоналізм працівників [5].

Для постійного підвищення свого рівня викладач ВНЗ повинен паралельно займатися науковою роботою. Наукова робота в області прикладних наук вимагає відповідної експериментальної бази, а вона у навчальних закладах часто застаріла. На придбання нових приладів і установок для проведення наукової роботи грошей хронічно не вистарчає.

Певною мірою кадрове питання розв'язується за рахунок випускників, які працювали за