Міністерство освіти і науки України

Ніжинський державний університет імені Миколи Гоголя

Навчально-науковий інститут природничо-математичних, медикобіологічних наук та інформаційних технологій

Кафедра інформаційних технологій, фізико-математичних та

економічних наук

Освітня програма: Комп’ютерні науки

Спеціальність:122 Компʼютерні науки

**КВАЛІФІКАЦІЙНА РОБОТА**

на здобуття освітнього ступеня магістр

**Створення і застосування комп’ютерного тестового контролю успішності студентів**

студента **Ландика Олександра Григоровича**

**Науковий керівник:**

Фетісов Валерій Сергійович,

кандидат економічних наук, доцент

**Рецензенти:** Доктор технічних наук,

професор І. В. Казачков.

доцент, кандидат технічних наук Т. А. Кресан

**Допущено до захисту: \_\_\_\_ \_\_\_\_\_\_ 2023 р.**

Завідувач кафедри

проф. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Казачков І.В.

**Ніжин − 2023 АНОТАЦІЯ**

Метою кваліфікаційної роботи є аналіз і порівняння існуючих систем тестування, перевірка валідності тестових завдань, створення власного програмного продукту для проведення тестування на базі Web-технологій. Об’єктом дослідження є процес створення і застосування комп’ютерного тестового контролю з використанням спеціальних програмних продуктів. Предметом дослідження є створення власного програмного продукту для проведення комп’ютерного тестового контролю знань студентів у закладах фахової передвищої освіти. Магістерська робота складається зі вступу, основної частини, яка включає два розділи, висновків, списку використаних джерел, додатку. Результати роботи: на підставі проведеного аналізу комп’ютерних систем Айрей, MyTest, VeralTest, Indigo, Moodle розроблено сайт для проведення тестування; надані практичні рекомендації щодо перевірки на валідність комп’ютерних тестових завдань.

**Ключові слова:** комп’ютерне тестування, види тестових завдань, створення, валідність, застосування, система контролю.

**ABSTRACT**

The purpose of the qualification work is to analyze and compare existing testing systems, check the validity of test tasks, and create your own software product for testing based on Web technologies. The object of research is the process of creating and applying computer test control using special software products. The subject of the research is the creation of an own software product for computer-based test control of students' knowledge in institutions of vocational pre-higher education. The master's thesis consists of an introduction, the main part, which includes two sections, conclusions, a list of used sources, an appendix. Results of the work: based on the analysis of the Airey, MyTest, VeralTest, Indigo, Moodle computer systems, a site for testing was developed; practical recommendations for checking the validity of computer test tasks are provided.

**Key words:** computer testing, types of test tasks, creation, validity, application, control system.

**ЗМІСТ**

|  |  |
| --- | --- |
| ВСТУП…………………………………………………………………… | 4 |
| Розділ 1 ТЕОРЕТИЧНІ ПЕРЕДУМОВИ ОРГАНІЗАЦІЇ КОМП’ЮТЕРНОГО ТЕСТОВОГО КОНТРОЛЮ……………………….. | 7 |
| 1.1 Методи контролю успішності студентів……………………….. | 7 |
| 1.2 Педагогічні умови ефективного використання тестового контролю…………………………………………………………………….. | 12 |
| 1.3 Методика створення тестових завдань…………………………. | 20 |
| 1.4 Створення комп’ютерних тестових завдань ……………………. | 26 |
| 1.5 Висновки розділу…………………………………………………. | 41 |
| Розділ 2 СТВОРЕННЯ ТА ЗАСТОСУВАННЯ КОМП’ЮТЕРНОГО ТЕСТУВАННЯ……………………………………………………………… | 42 |
| 2.1 Порівняльний аналіз програм ……………………………………. | 42 |
| 2.2 Комп’ютерне тестування та його якість…………………………. | 46 |
| 2.3 Реалізація комп’ютерного тестового контролю .………………. | 55 |
| 2.4 Висновки розділу…………………………………………………. | 61 |
| Висновки…………………………………………………………………….. | 62 |
| СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ………………………………….. | 64 |
| ДОДАТОК А………………………………………………………………... | 67 |

**ВСТУП**

**Актуальність теми.** Комп’ютерний тестовий контроль – актуальна тематика для теперішнього часу в якому перебуває наша держава. Входження Укрaїни в сучaсні соціaльно-економічні умови вимaгaє від молодого покоління володіння нaуковими знaннями нa високому рівні. З цього випливaє, що існує потребa у пошуку об’єктивних методів, методик і зaсобів оцінювaння знaнь тa вмінь студентів. Одним із сучасних методів оцінювання знань студентів є комп’ютерний тестовий контроль. Однак як і письмові тести так і комп’ютерні тести мають однакові завдання, які в свою чергу не всі мають якісну складову. Якість тестового завдання визначає, наскільки те, що вимірюється за допомогою даного тесту, відповідає тому, що він має вимірювати. Отже перевірка тестових завдань на валідність одна з основних складових для створення якісного комп’ютерного тестового контролю.

Комп'ютерний тестовий контроль - це метод оцінювання або вимірювання знань, навичок, або інших властивостей особи за допомогою комп'ютерної технології. У цьому методі використовуються комп'ютерні програми та спеціальні системи для створення, проведення та оцінювання тестових завдань. співробітництва в системі взаємодії студентів та викладачів.

Для створення та проведення комп’ютерного тестового контролю існує досить широкий перелік спеціалізованого програмного забезпечення. Кожен програмний продукт особливий та має свої переваги та недоліки. Фунціонал різних систем не завжди є достатнім, доступ зазвичай э локальним, а не глобальним.

Детальний аналіз існуючих програмних продуктів для створення комп’ютерного тестового контролю, їх можливостей, переваг та недоліків, надав поштовх до створення власного продукту у вигляді Web-сторінки для проходження тестування. Web-технології є основою для розробки тому, що у сучасних умовах військового стану та досить великій частині дистанційної форми навчання актуально створити програмний продукт з доступом до нього з будь-якої точки світу за умови наявності доступу до мережі Internet.

Використання HTML, CSS та JavaScript у поєднанні дозволяє створювати повноцінні та ефективні Web-додатки з багатошаровою функціональністю. Забезпечує можливість реалізації складних дизайнів та функцій, які забезпечують кращий користувацький досвід. Використання цих мов дозволяє дотримуватися Web-стандартів, що робить сайт сумісним із різними браузерами та пристроями.

**Об’єктом** дослідження є процес створення і застосування комп’ютерного тестового контролю з використанням спеціальних програмних продуктів.

**Предметом дослідження** є створення власного програмного продукту для проведення комп’ютерного тестового контролю знань студентів у закладах фахової передвищої освіти.

**Метою** кваліфікаційної роботи є аналіз і порівняння існуючих систем тестування, перевірка валідності тестових завдань, створення власного програмного продукту для проведення тестування на базі Web-технологій.

**Завдання** **дослідження**: висвітлити методику створення комп’ютерного тестування, розкрити особливості перевірки тестових завдань на валідність з використанням спеціалізованого програмного забезпечення, створити власну систему комп’ютерного тестування.

Апробації результатів дослідження. Опубліковано статтю та тезу доповіді у збірниках наукових праць [17, 18]. Результати дослідження також оприлюднені на доповіді на Всеукраїнській науково-практичній конференції педагогічних та науково-педагогічних працівників, аспірантів, молодих учених «Освіта, наука та виробництво: актуальні питання теорії та практики» (Ніжин, Україна, 20 квітня, 2023.).

Структура кваліфікаційної роботи. Робота складається зі вступу, двох розділів, висновків, списку використаних джерел. Загальний обсяг роботи становить 86 с., основний зміст викладено на 66 с. Праця містить   
17 рисунків та 3 таблиці.

**РОЗДІЛ 1**

**ТЕОРЕТИЧНІ ПЕРЕДУМОВИ ОРГАНІЗАЦІЇ КОМП’ЮТЕРНОГО ТЕСТОВОГО КОНТРОЛЮ**

* 1. **Методи контролю успішності студентів**

Контроль у дидактиці визнається як проведення перевірки, оцінки та реєстрація успішності студентів. Він є важливим елементом управління навчально-пізнавальною діяльністю тих, хто вивчає, і засобом отримання зворотної інформації. Крім того, контрольні заходи дають можливість викладачеві отримати дані про результати своєї роботи, вчасно внести корективи в методику викладання, а студентам – бути ознайомленими з вимогами до рівня знань, умінь та навичок, їх якості тощо [4].

Контроль та оцінювання результатів навчання студентів виконуються за дотримання певних педагогічних умов, а саме: об'єктивності – оцінювання знань, умінь та навичок передбачає правильне застосування встановлених критеріїв та єдиних норм оцінювання. На оцінювання знань не повинні впливати симпатії та антипатії викладача, особливості поведінки та здоров'я студента; систематичності – контроль повинен здійснюватися регулярно протягом семестру та навчального року; всебічності та всеосяжності – високий ступінь охоплення контролем змісту навчального матеріалу (розділів, тем, теорії і практики), осмисленість, глибина, гнучкість, дієвість, міцність, системність його засвоєння; індивідуального підходу під час оцінювання успішності студентів – забезпечення таких дидактичних умов, за яких знімається психологічне напруження, тривога студентів за об'єктивність оцінювання їх знань, створюється атмосфера доброзичливості, а справедлива оцінка стимулює кожного студента до систематичної навчально-пізнавальної діяльності, до ліквідації виявлених помилок і недоліків; диференційованості – різні рівні сформованості знань, умінь та навичок студентів повинні бути відображені диференційованою системою оцінок.

Функції контролю навчання розглядаються в різних аспектах: освітня (навчальна) – сприяє повторенню, уточненню та систематизації знань, поліпшенню навичок і, таким чином, глибшому засвоєнню програмованого матеріалу; діагностична – обґрунтоване визначення стану успішності дозволяє виявити успіхи і недоліки в знаннях студентів, встановити їх причини та скоригувати методику оволодіння навчальним матеріалом; виховна – контроль привчає студентів до систематичної навчальної праці, сприяє вихованню в них дисциплінованості, пізнавальної самостійності, наполегливості, почуття відповідальності та адекватної самооцінки; розвивальна – у процесі виконання контрольних завдань розвиваються пізнавальні здібності студентів, вміння виконувати основні мислительні операції – аналізу і синтезу, порівняння, систематизації, узагальнення, виділення головного та інше; стимулююча – справедливе і регулярне оцінювання результатів навчання є важливим стимулом у підготовці майбутнього фахівця, спонукає студентів до поглибленого вивчення основ наук, сприяє ефективності формування їх пізнавальних потреб та мотивів навчання; управлінська – завдяки контролю викладач вчасно отримує інформацію про якість процесу оволодіння знаннями та здійснює заходи щодо управління ефективністю цього процесу через корективи у власній методиці та в організації самостійної роботи студентів.

Екзамени та заліки несуть значущу роль із точки зору стимулюючого ефекту. Свідомість студентів про наближення майбутніх екзаменів і заліків спонукає їх більш відповідально підходити до щоденних занять. Це стимулююче значення екзаменів і заліків підкреслюють усі викладачі та студенти, які були опитані щодо цього питання. Треба відзначити, що важливість їх ще більша через те, що ці оцінювальні заходи мобілізують студентів до досягнення конкретної мети, орієнтуючи їх не лише на ближчий період, а й на досягнення цілей, які здійснюються в значно віддаленому майбутньому (оскільки екзамени і заліки часто проводяться лише під час сесій, наприкінці навчального року або в кінці першого семестру).

Важливо враховувати, що бажання досягти віддаленої мети визнається як цінний аспект, оскільки воно вимагає від студентів значної самостійності. Ця самостійність обумовлена необхідністю розробки комплексу дій, спрямованих на досягнення остаточної мети.

Окрім того, екзамени та заліки є необхідними з іншого погляду — лише на підставі результатів цих оцінювань можливі атестація студентів, їхнє переведення на інший курс, вхід до державних екзаменів або захисту дипломного проекту, а в кінці кінців — отримання диплома та присвоєння відповідної кваліфікації.

Заліки та екзамени вищої школи, виконуючи важливу та різнобічну роль, призначені впроваджувати ряд функцій, причому основною та вирішальною є їхня функція обліку та контролю. Успішне навчання можливе лише за умови систематичного та ретельного обліку та контролю навчальної діяльності. Перевірка знань, вмінь та навичок студентів виступає конкретною формою втілення функцій обліку та контролю знань. У процесі навчання, оцінка не тільки здійснює свою основну роль, але й слід розглядати як органічну та важливу частину усього навчального процесу.

Під час проведення перевірки та оцінки знань, функція контролю взаємодіє органічно з функцією активного навчання студентів.

Вчителю, який спостерігає за ходом і результатами навчання студента, доводиться перевіряти та оцінювати, наскільки він освоїв науку, як оволодів її вмістом та методами. Процес самостійної підготовки студентів до заліків і екзаменів відіграє велику роль у виконанні та удосконаленні їхніх знань.

Під час підготовки до екзаменів студенти уточнюють та розширюють багато з того, що вони вже зрозуміли на лекціях або інших заняттях, але в основному взяли узагальнено або зовсім не зрозуміли. Одним з важливих виявів навчальної функції екзаменів і заліків є їхня роль у систематизації навчального матеріалу. Екзамени, особливо ті, що охоплюють весь навчальний курс, заохочують студентів до структуризації своїх знань. Це важливе завдання може бути успішно виконане лише після повного вивчення навчального предмету, що найкраще реалізується під час підготовки до екзамену.

Контрольні прaктичні зaвдaння виконуються двомa способaми - індивідуaльним тa фронтaльним. Одночaсне виконaння прaктичного зaвдaння студентською групою (фронтaльний спосіб) здійснюється тоді, коли при індивідуaльній роботі неможливо отримaти кінцевий результaт.

Тести для контролю знань студентів у останні роки масово використовуються для діагностики освітньо-професійної підготовки у закладах освіти. Термін "тест" можна інтерпретувати як "випробування", "перевірка", "спроба", "мірило" або "критерій". Засіб випробування, аналогічний тестуванню, використовувався ще тисячі років тому в Китайській імперії. В Древній Греції випробування використовували для оцінювання фізичних та мислительних здібностей. Відомо, що Платон пропонував серію випробувань військових здібностей для відбору воїнів у "ідеальній" державі. Цей перелік випробувань вважається першим систематичним описом тестів здібностей. Однак класне тестування, яке означає стандартизоване вимірювання індивідуальних відмінностей, можна прослідкувати лише з кінця XIX століття. Тести систематизуються таким чином:

1. Тести зaгaльних розумових здібностей, розумового розвитку.

2. Тести спеціaльних здібностей у різномaнітних гaлузях діяльності.

3. Тести досягнень, успішності, aкaдемічних успіхів.

4. Тести для визнaчення певних якостей (влaстивостей) особистості (пaм'яті, мислення, хaрaктеру й ін.).

5. Тести для визнaчення рівня виховaності (сформовaності зaгaльнолюдських, морaльних, соціaльних і інших якостей).

Отже, тестовий контроль знань представляє собою метод вимірювання та оцінки знань, вмінь і навичок студента за допомогою спеціально підготовлених стандартизованих завдань. З свого боку, тест досягнень включає в себе серію стандартизованих завдань, які дають можливість об'єктивно виміряти обсяг і рівень засвоєння студентами конкретних знань, вмінь і навичок.

Стандартизованими називається завдання, для якого може бути попередньо визначена єдина можлива правильна відповідь. Завдання бувають двох типів: із заданими відповідями (закриті) і з вільним складанням відповіді (відкриті). Завдання із заданими відповідями поділяються на: вибіркові (одновибіркові, багатовибіркові та відновлення послідовності) та на відповідність (перехресні, на порівняння і протиставлення, з множинними відповідями). Завдання з вільним складанням відповіді можуть бути або на доповнення, або на перелік.

Одновибіркові завдання використовуються, коли відповідь на ситуації, представлені для розв'язання студентами, може складатися лише з одного елемента.

Багатовибіркові завдання застосовуються у випадках, коли об'єкт розгляду має багато складових частин або властивостей, і для характеристики необхідно вказати їх повну сукупність, яка утворює правильну відповідь.

Завдання на відновлення послідовності використовуються, коли потрібно перевірити не лише знання властивостей, структури об'єктів, процесів, явищ, але і взаємодію складових елементів, послідовність етапів якого-небудь процесу, розвитку явищ, засвоєння студентами певних алгоритмів і т. д.

Перехресні завдання включають у себе встановлення зв'язку, взаємозв'язку між двома (або більше) списками елементів.

Завдання на порівняння і протиставлення складаються з двох або більше списків елементів, де студент повинен вибрати з першого список елементи, що відповідають визначеним умовами завдання вимогам, і знайти відповідні елементи в кожному з наведених списків.

Завдання з множинними відповідями складаються з питань та двох чи більше списків. У першому перелічені об'єкти, а в інших - перелік їх властивостей. Під час виконання завдання студент повинен для кожного наведеного об'єкта вибрати той перелік властивостей, які їм притаманні.

Завдання на доповнення мають форму ствердження (вислову, речення, числа, формули тощо), в якому є один або кілька пропусків, які студент повинен заповнити під час відповіді, вписуючи слова, цифри, символи, умовні позначення тощо.

Завдання на перелік вимагають від студента перелічити компоненти, властивості, ознаки чи характеристики об'єкта, який зазначений в умові завдання.

* 1. **Педагогічні умови ефективного використання тестового контролю**

Враховуючи сучасний ринок праці, освіта в Україні на сьогодні визначає свої пріоритети не тільки через знання, вміння і навички, які молодь отримує під час навчання у загальноосвітніх та вищих закладах, але й через вміння працювати з технологіями та знаннями, що відповідають потребам сучасного інформаційного суспільства. Важливим стає уміння пристосовуватись до нових вимог ринку праці, ефективно володіти і управляти інформацією, приймати рішення швидко та активно діяти, а також постійно навчатися протягом усього життя. Згідно з сучасними поглядами педагогів, саме розвиток життєво важливих компетентностей може допомогти особі орієнтуватись у сучасному суспільстві, інформаційному просторі, стрімкому розвитку ринку праці та постійно розширювати свої знання. Таким чином, метою навчання в закладах вищої освіти та фахової передвищої освіти у сучасних умовах є забезпечення професійного та особистісного розвитку фахівця, який може застосовувати здобуті знання на практиці, а також володіє здатністю здобувати та використовувати нову інформацію, розвиватися як фахівець і залишатися компетентним у своїй галузі.

Використання тестування як засобу контролю за рівнем навчальних досягнень є важливою складовою кредитно-модульної системи, якою керуються заклади освіти в Україні. Досвід, накопичений у розвинених країнах, а також український досвід використання тестів для зовнішнього оцінювання чи мультипредметного тесту навчальних досягнень учнів, свідчить про те, що тестування вважається надійним та об'єктивним інструментом оцінювання.

Використання систематичного тестування надає можливість студентові отримувати інформацію про його власні навчальні досягнення відповідно до індивідуального плану отримання вищої освіти. Цей план враховує всі види робіт студента та всі навчальні досягнення, включаючи навчання у різних вищих навчальних закладах. Така система забезпечує більш об'єктивне оцінювання навчальних досягнень студента.

Заклади освіти приділяють значну увагу використанню тестування у навчальному процесі, і ця тема отримала значні дослідження від педагогів, психологів та методистів.

Вчені, їх дослідженням, детально розглянули аспекти теорії педагогічних тестів, включаючи композицію та форму тестових завдань, розробку процедур проведення тестування, алгоритми обробки та інтерпретації результатів. Впроваджений досвід тестового оцінювання навчальних досягнень підтверджує можливість якісного та ефективного використання тестування для контролю навчальних досягнень школярів та його використання на випускних та вступних іспитах. Проте ці дослідження, насамперед, висвітлюють питання впровадження та використання тестування в загальноосвітніх навчальних закладах. Дослідження використання тестування в закладах вищої освіти має обмежений обсяг.

Контроль повинен мати такі характеристики:

* + 1. об'єктивність;
    2. систематичність та регулярність;
    3. прозорість (студентам має бути зрозуміло, чому їх відповідь оцінено конкретною кількістю балів, вони повинні мати доступ до критеріїв оцінювання перед вивченням навчальної теми);
    4. всебічність (охоплення всіх розділів програми);
    5. диференційованість (врахування як рівнів засвоєння програмового матеріалу, так і індивідуальних особливостей студентів);
    6. різноманітність за формами, що створює умови для реалізації всіх функцій контролю та сприяє підвищенню інтересу студентів до його проведення.

Реалізація вказаних вимог, за словами дослідників, сприяє гуманізації контролю навчальних досягнень студентів і позитивно впливає на їхню успішність у навчанні [19].

З огляду на особистісну зорієнтованість освіти та діяльнісний підхід до навчання виникають нові вимоги до контролю результатів навчання у вищій школі. Якщо дидактична система, включаючи систему контролю результатів навчання, відповідає наступним вимогам, то вважається особистісно орієнтованою: вона має активний характер та спрямована на підтримку індивідуального розвитку особистості; надає студентові достатній простір для самостійних рішень, творчості, вибору навчального змісту та методів, а також визначення власної поведінки. З цього випливає, що система тестового контролю навчальних досягнень студентів повинна бути побудована таким чином, щоб відповідати цим вимогам.

Безпосередньо до створення тесту також пред'являють вимоги вчені, що досліджують аспекти тестування в освіті, такі як В.С. Аванесов, І.Є. Булах, А.Н. Майоров. Вони визначають ряд критеріїв якості для оцінки тесту, серед яких важливі такі як валідність, об'єктивність, надійність та  
 точність [6].

Валідність – це комплексна оцінка, яка визначає, наскільки те, що вимірюється за допомогою даного тесту, відповідає тому, що він має вимірювати. Вона залежить як від характеристик самого інструмента і процедур вимірювання, так і від особливостей ознаки, яку діагностують.

Об’єктивність тесту визначається його здатністю до мінімізації впливу суб’єктивних факторів, таких як суб’єктивні оцінки тих, хто проводить тестування. Ця мінімізація досягається через стандартизацію умов проведення тестів, процесу оцінювання та аналізу результатів. Стандартизація умов тестування включає в себе:

* об’єктивність проведення тестів;
* об’єктивність обробки даних;
* об’єктивність інтерпретації результатів.

Надійність тестування визначається його здатністю надавати стійкі результати, що впливає на точність вимірювання конкретної ознаки. Перевірка надійності методу стосується, перш за все, стабільності результатів при повторних вимірюваннях.

Отже, при створенні тесту слід враховувати, наскільки важливо досягти високої точності: чим більшу точність ми прагнемо вбудувати в тест, тим він буде займати більше часу та зусиль.

При формуванні змісту та структури тесту необхідно враховувати особливості конкретного курсу та обов’язкові вимоги до його освоєння, що відповідають меті навчання. Якісний тест для контролю навчальних досягнень студентів повинен бути спрямований як на опанування предметних знань і навичок, так і на перевірку розвитку загальних вмінь (аналіз, міркування, логічне мислення).

Таким чином, ми можемо сформулювати наступні вимоги до тестового контролю результатів навчання студентів у закладах освіти.

Тестовий контроль повинен мати наступні характеристики:

1. індивідуальний, що означає оцінку та корекцію результатів навчання кожного студента, його власний облік;
2. систематичний, який передбачає регулярний контроль прогресу студента на різних етапах навчання (по темі, семестру, році);
3. об'єктивний, що уникає суб'єктивних суджень викладача, базованих на недостатньому знанні або упередженому ставленні, та мінімізує вплив соціальних, особистісних та емоційних факторів;
4. прозорий, що передбачає ознайомлення студентів з критеріями оцінювання перед вивченням теми;
5. валідний, що гарантує відповідність тестових завдань тому, що має бути перевірено;
6. всебічний, який охоплює контроль теоретичних положень, предметних та загальних навичок, інтелектуального і загально-психічного розвитку студентів, виявлення їхніх нахилів і здібностей;
7. диференційований, що враховує рівні засвоєння матеріалу та індивідуальні особливості студента;
8. надійний, тобто результати контролю повинні співвідноситися з повторними вимірюваннями;
9. точний, що передбачає мінімальну похибку вимірювань тестом;
10. різноманітний за формами проведення, що дозволяє комбінувати різні типи тестових завдань в залежності від мети контролю.

Впровадження системи тестового контролю дозволяє вчителям організувати процес оцінки знань, навичок та вмінь студентів, підвищуючи при цьому інформативність та об'єктивність оцінювання. За результатами такого контролю студенти отримують можливість об'єктивно оцінити свої досягнення та свідомо визначити рівень володіння навчальним матеріалом.

Тестова перевірка дозволяє оцінити більший обсяг навчального матеріалу за той самий період, порівняно із звичайним іспитом. Використання єдиної системи оцінювання робить усіх студентів рівноправними, сприяючи зменшенню нервового напруження. Усі студенти мають рівні можливості, а широкий спектр тестового матеріалу дозволяє їм продемонструвати свої досягнення на широкому полі тем.

Стандартизація вимог до тестового контролю, уніфікація процедур проведення тестування, використання єдиної системи оцінювання та можливість комп'ютерної обробки результатів тестування сприяє підвищенню довіри педагогів та студентів до результатів тестових іспитів.

Оцінка успішності через тестування вимагає попередньої підготовки, а саме - планування, яке включає прийняття важливих рішень для ефективного використання тестів. Однією з ключових умов ефективного тестування, за думкою Р. Ібела, є визначення частоти проведення тестів. Цю частоту визначає навчальний заклад в певній мірі. Більшість американських викладачів вважають за доцільне проводити тестування принаймні двічі на семестр, а деякі навіть кожні три-чотири тижні. Це має свої переваги. Тести, проведені з короткими інтервалами, охоплюють менший обсяг навчального матеріалу, але при цьому є більш детальними. Тут можна погодитися з   
Р. Ібелом, що детальний тест, який охоплює менший обсяг матеріалу, є більш важливим, ніж той, що охоплює більший обсяг, але є менш детальним. Крім того, слід відзначити, що часте тестування забезпечує більш надійну основу для оцінки та дозволяє викладачам і студентам свідомо розуміти результати своїх досягнень [28].

Однак, якщо викладач не має значної кількості заздалегідь підготовлених тестових завдань та не має досвіду їх швидкої перевірки, часте проведення тестування може вимагати значних витрат часу. В окремих випадках підготовку до тестування та сам процес тестування можна включити в час, який призначений для пояснення нового навчального матеріалу. Таким чином, часте тестування може призвести до відзначеної деформації освітніх пріоритетів та виникнення у студентів неправильної уяви про те, що успішне проходження тестів є або навіть головною метою навчання. Науково доведено, що добре сконструйовані та розумно використані тести мають певну педагогічну цінність. Виконуючи тестові завдання, студент свого роду повторює матеріал навчальної програми. Однак важливо усвідомлювати, що основна мета тестів полягає не в самому процесі навчання, а в перевірці знань.

Для забезпечення ефективного тестування важливо визначитися з використанням конкретної форми (типу) тестових завдань. Серед часто використовуваних форм тестових завдань виділяються такі: тести-твори (есе), або дискусійний тип, об’єктивний тип, або тип короткої відповіді, і тип математичної задачі (тест-задача). Зокрема, тест-задача широко використовується у тестуванні навчальних досягнень в математиці, природничих науках, інженерії та інших відповідних предметах. Для багатьох інших дисциплін більш підходять або тести-твори, або об’єктивні тести. Наведена коротка порівняльна характеристика цих типів тестових завдань розглядається американськими дослідниками в контексті визначення умов ефективного тестування.

На думку американських авторів, основна потреба полягає у виправленні деяких неправильних уявлень. Зокрема, неправильним вважається погляд на те, що один тип тестів забезпечує повністю об'єктивну перевірку знань, тоді як інші види тестів перевіряють лише поверхневі аспекти знань. За словами М. Річардсона і Дж. Столнейкера, "форма тесту сама по собі ще не гарантує виявлення тієї здатності, рівень наявності якої перевіряється". Також помилковим є вважання, що удача відіграє значну роль в одному виді тестування, тоді як її абсолютно немає в іншому. Навпаки, всі три види тестів можуть виявити однаковий рівень здібностей і, при відповідній увазі, дати результати практично однакової достовірності. Досконалий тест-твір чи досконалий об'єктивний тест можуть визначити рейтинг групи студентів практично у тій самій послідовності, що й тест-задача. Проте це не означає, що всі три види тестів можуть взаємозамінюватися з однаковою легкістю та ефективністю. Ф. Вернон вказує на очевидність того, що "тести з однаковими цілями, але різних форм, можуть призводити до суперечливих результатів. Тести однакової форми, розроблені для різних предметів та розумових функцій, добре корелюють між собою. Для багатьох цілей більш прості тести мають більшу валідність, і під сумнівом те, наскільки складні тести можуть ефективно перевірити вищі розумові функції, на які вони спрямовані".

За думкою Р. Ібела, у порівнянні з об’єктивним тестом, тест-твір та тест-задача легше готувати, але об’єктивний тест забезпечує швидше оцінювання та облік успішності, особливо у порівнянні з тестом-твором. Коли мова йде про тестування великої кількості студентів, використання об’єктивного тесту дозволяє досягти ефекту з мінімальною або нульовою втратою валідності. З іншого боку, у випадку тестування у невеликих групах чи класах перевага повинна бути віддана тестам-творам та тестам-задачам, оскільки вони забезпечують більшу валідність.

Згідно з Р. Ібелом, серед усіх типів тестів тест-задача має перевагу внутрішньої релевантності. Цей тип тесту, порівняно з іншими, дозволяє виявити знання в конкретних діяльностях та ступінь володіння матеріалом для розв'язання конкретних проблем. Тести-твори та об’єктивні тести, ймовірно, містять багато поверхових та чисто академічних завдань, що, можливо, слід уникати.

У той же час, об'єктивні тести мають свої переваги в порівнянні з тестами-творами та тестами-задачами. Можливості використання тестів-задач, до певної міри, обмежуються обсягом і складністю відповідей, які потрібно надати (відповіді повинні бути написані вручну). Хоча існують твердження, що здатність вибирати відповідь менш важлива, ніж здатність її формулювати, "більшість тверджень свідчать, що ці здатності тісно пов'язані".

При оцінці порівняльних переваг між тестами-творами, тестами-задачами та об'єктивними тестами важливо пам'ятати, що єдиним корисним компонентом будь-якого тестового бала (оцінки) є той компонент, який об'єктивно підтверджує його, незалежно від типу тесту, на підставі якого цей бал виставлений.

У зв'язку з цим вкрай важливо уникати суб'єктивних вражень і неперевірених суджень з боку тих, хто проводить тестування та оцінює студентів.

Об'єктивність більше характерна для об'єктивних тестів (тестів короткої відповіді) та тестів-задач, порівняно з тестами-творами. У них мета завдань чітко визначена, тому підстави для вищої або нижчої оцінки є очевидними. Важливо відзначити, що, навіть у випадку об'єктивних тестів, наявність великої кількості суб'єктивних рішень щодо того, що тестувати та як тестувати, є відомою фактором, який може впливати на їхню об'єктивність.

У той час як тести-задачі мають додатковий аспект суб'єктивності при оцінюванні, якого немає в об'єктивних тестах. Питання, такі як оцінка неповної відповіді та врахування конкретних аспектів у кожному випадку, часто вирішуються спонтанно та суб'єктивно.

З метою забезпечення якісного оцінювання екзаменатор повинен прагнути до якнайбільш об'єктивного підходу, незалежно від типу тесту. Рівень об'єктивності оцінювання може бути визнаний високим, якщо його підтверджують інші компетентні екзаменатори. Зрозуміло, що оцінки, отримані за досконалий тест-твір, є значно об'єктивнішими, ніж ті, які отримані за недосконалий тест з вибором з множини варіантів.

Тим не менш, слід погодитися з Р. Ібелем, який вказує на те, що "викладачі, які користуються тестами-творами, мають тенденцію менше приділяти уваги об'єктивності своїх оцінювань та оцінок, ніж ті, хто використовує тести з вибором з множини варіантів".

**1.3 Методика створення тестових завдань**

Тестування дозволяє викладачу не лише порівняти якість знань та вмінь кожного студента чи класу з різних навчальних дисциплін, освітніх галузей або загального навчального плану з вимогами освітнього стандарту (мінімуму), але також визначити рівень володіння студентом кожним розділом програми. При використанні багатомірних тестів тестування може виявити як предметні, так і позапредметні вміння студента, надаючи якісну оцінку його знань та вмінь.

Особливого значення прийоми тестового контролю знань та вмінь студентів набувають у зв'язку з переходом в України до Болонської системи.

Педагогічний тест представляє собою набір взаємопов'язаних завдань, що зростають за складністю і дозволяють достовірно та валідно оцінити рівень знань або інші психолого-педагогічні характеристики.

Валідність, перекладаючи з англійської як "придатний", є одним із ключових критеріїв якості тесту. Чим вища валідність тесту, тим повніше він відображає якість, яку він має вимірювати.

Види тестів:

1) Альтернативний тест є одним з найлегших у розв'язанні, оскільки включає запитання з 4—5 варіантами відповідей, серед яких лише один є правильним. Чим більше варіантів відповідей, тим менше ймовірність вгадування. Наприклад, при двох варіантах відповідей ймовірність вгадування становить 48,7%. Альтернативний тест варто використовувати для завдань, які не допускають різних тлумачень правильної відповіді.

Цей тип тесту може слугувати не лише для перевірки повноти знань. З метою визначення глибини розуміння можна використовувати наступне запитання:

«Із перелічених нижче формулювань виберіть те, яке найточніше характеризує... (речення, іменник, частку тощо).»

У цьому тесті характеристики включають надлишкову, недостатню, або помилкову інформацію, а також правильну відповідь.

Важливо зауважити, що альтернативні типи тестових завдань використовуються у централізованому тестуванні.

2) Вибірковий, або варіативний, тест, як правило, не становить серйозних труднощів для студентів. У цьому типі тестування надається 10—12 варіантів відповідей на кожне завдання, при цьому 6—8 з них є правильними. Основний акцент в вибіркових тестах робиться на оцінці повноти знань щодо класифікації конкретного явища.

Наприклад, варіативний тест може одночасно перевіряти як глибину, так і розмаїтість знань студентів. Такий тест передбачає не лише знання фактів, але і їхнє віднесення до різних категорій (активних або пасивних). Кількість балів за такий тест, як правило, відповідає кількості правильних варіантів відповідей, при цьому кожна помилка віднімає 1 бал.

3) Послідовний, або порядковий, тест надає можливість перевірити у студента знання послідовності конкретного процесу. У варіантах відповіді на таке завдання відсутні неправильні варіанти. Студент повинен впорядкувати у правильній послідовності надані у невпорядкованому вигляді поняття, слова чи визначення [6].

Це завдання передбачає, що студент заповнює пропуски у тексті, який описує певне явище, розміщуючи пропущені слова в правильному порядку. На місця пропусків слід ставити цифри — 1, 2, 3 і так далі, а після опису додати буквену нумерацію: а), б), в) і так далі. Ключові слова для впорядкованого розташування подаються у вигляді ключа, де студент вказує цифри та відповідні буквенні позначення з ключового ряду. Кількість балів у цьому тесті визначається кількістю вірних відповідей, і за кожну помилку знімається 1 бал.

4) Розподільний тест є найбільш вимогливим для студентів, оскільки передбачає розподіл ознак між трьома чи більше поняттями. Кількість ступенів розподілу може змінюватися відповідно до глибини перевірки знань, і для кожного розділу тест-карти необхідно вирішувати окремі завдання. Його особливість полягає в тому, що опитувані розподіляють правильні варіанти характеристик досліджуваних об'єктів, процесів або понять.

5) Тест із схемою побудови, в якому подано структуру чогось без підписів із зазначеними номерами 1, 2, 3 і т. д., а також під літерами а), б), в) і так далі, подано ключові слова. Учасники тесту повинні встановити відповідність між вказаними номерами та їхніми визначеннями; за кожну помилку знімається 1 бал.

6) Кількісний тест є варіацією альтернативного або вибіркового тесту, але учасник повинен обрати правильні відповіді з числових варіантів, які надані серед запропонованих відповідей.

7) Узагальнювальний (збірний) тест дозволяє не лише перевірити контрольні знання студентів, але й оцінити їх здатність до узагальнення, формулювання висновків та класифікації різних явищ. Цей вид тесту також можна віднести до групи альтернативних (або вибіркових) тестів. При розробці такого тесту завдання подаються у двох колонках: в одній колонці містяться різні ознаки, які характеризують 2-3 правила або об'єкти, а в іншій — самі правила або об'єкти.

8) Аналітичний тест, так само як і попередній, включає перерахунок характеристик будь-якого об'єкта чи явища, проте це формулюється у зворотному порядку.

9) Тест порівняльного типу передбачає визначення спільних та відмінних аспектів у досліджуваних процесах чи явищах; його мета — не лише контроль якості знань та вмінь, але й перевірка здатності виявляти схожість та відмінність у розглядуваних явищах. Такий тест може бути використаний для оцінювання студентів практично на будь-якому курсі та предметі.

10) Операційний тест може бути створений за аналогією із будь-яким з наведених тестів вище, проте його специфіка полягає в перевірці у студента знань та вміння практично використовувати конкретні процеси чи методи.

11) Проблемно-пошуковий тест вимагає творчого застосування знань у нестандартній ситуації, представляючи собою завдання, що включає складне запитання або проблему, яка потребує розв'язання. У такому тестовому завданні студент повинен пропонувати варіанти розв'язання (гіпотези) та надавати систему аргументів. Цей вид завдання вважається одним із найбільш високорівневих у змістовному плані.

Як видно, білети з будь-якої дисципліни можуть бути створені, використовуючи різні типи тестів, що стимулює різні інтелектуальні вміння студентів. Різноманітність тестів, які входять до білету, підвищує її інформативність і дає можливість не лише проводити кількісний, але і якісний аналіз знань та вмінь студента, що неможливо при визначенні середньої оцінки або рівня навченості групи.

Вищезазначені принципи (дидактичні правила) є основами тестування, і без їх знання неможливо розпочати розробку тестів та проведення самого тестування. Проте існує ще один невід'ємний етап — експериментальна апробація розробленого білету та його блоків з кожного розділу програми.

Часто результати тестування вищі, ніж результати інших методів оцінювання якості освіти, таких як контрольні роботи, заліки, державна підсумкова атестація та інші. Це головним чином свідчить про простоту тестових завдань, включених у цей комплект тестів, і, отже, про об'єктивність такого тестування знань.

Також важливо провести експертизу тестів щодо їхньої відповідності змісту освітнього стандарту, щоб забезпечити, що цей комплект тестів буде слугувати як критеріально-оцінний інструмент, спроможний діагностувати рівень знань та вмінь студентів відповідно до вимог освітнього стандарту.

У сучасних умовах тестовий контроль знань став широко використовуваним. В педагогічній літературі під терміном "дидактичний тест" розуміється складений згідно з певними вимогами комплекс завдань, який пройшов попереднє тестування. Його основна мета полягає в визначенні якісних показників та виявленні компетентностей учасників тесту, а його результати піддаються оцінюванню за заздалегідь встановленими критеріями.

Три основні критерії якості тестів — надійність, валідність і об'єктивність — визначають їх ефективність. Надійність означає ступінь відносної вільності від похибок під час вимірювання. Тест вважається надійним, якщо повторне проведення його через певний час призводить до однакових результатів.

Валідність визначається відповідністю змісту, форм та методів педагогічного контролю його призначенню. Тест є валідним, якщо він може ефективно перевірити те, що ставиться на меті. Збільшенню валідності педагогічного контролю сприяють експертні оцінки всього контрольного матеріалу, його адаптація до вимог навчальних програм, що розроблені на основі галузевих стандартів.

Об’єктивність педагогічного контролю включає два основних аспекти. Перший передбачає досягнення об’єктивності через створення колегіальної оцінки, яка надається спеціально утвореною комісією. Другий аспект полягає в застосуванні стандартних тестових програм та технічних засобів контролю.

Випробування фахівців підтверджує, що створення тестів є складним та витратним процесом, що вимагає від розробника конкретних методичних знань і навичок.

Зазвичай тест складається з двох основних частин: інформаційної та операційної. Інформаційна частина повинна включати чітко сформульовані та зрозумілі інструкції для учасника тестування, і, за можливістю, супроводжуватися прикладом виконання. Операційна частина тесту складається з конкретної кількості тестових завдань.

Процес складання тестового матеріалу може бути розділений на кілька етапів для ефективної роботи.

На першому етапі проводиться структурування навчального матеріалу. Починаючи роботу над тестами, важливо спочатку уточнити, що має знати студент з даної дисципліни.

Другий етап передбачає встановлення логічних зв'язків між елементами і створення логіко-структурних схем для тем, розділів та предмета в цілому.

На третьому етапі готуються конкретні тестові завдання, обираються елементи знань, які можна перевірити за допомогою тестування.

Четвертий етап включає вибір оптимальної форми тестових завдань.

П'ятий етап включає складання плану тесту та розробку стандартних бланків відповідей.

На шостому етапі проводиться перевірка тестів.

* 1. **Створення комп’ютерних тестових завдань**

Сучасна освітня система в Україні стрімко розвивається, а однією з інноваційних практик стала імплементація систем дистанційного тестування. Ця технологія стає не тільки доповненням до традиційних методів оцінювання знань, але й важливим інструментом для визначення рівня підготовки студентів.

Українські системи дистанційного тестування дозволяють ефективно оцінювати студентів в режимі онлайн, надаючи багато переваг. По-перше, вони роблять процес оцінювання більш доступним, оскільки студентам не потрібно присутність фізично в аудиторії чи спеціальному центрі. Це особливо актуально в умовах розподіленої географії українських вишів.

Дистанційне тестування також робить процес оцінювання більш гнучким, дозволяючи студентам вибирати зручний для них час і місце для здачі тесту. Це особливо важливо в умовах росту популярності онлайн-освіти та зміни у ритмі життя студентів.

Однією з ключових переваг систем дистанційного тестування в Україні є можливість об'єктивного визначення рівня знань студентів. Завдяки чітко визначеним критеріям оцінювання та автоматизованому процесу перевірки виключається суб'єктивність та необ'єктивність, які можуть виникнути при традиційних методах оцінювання.

Також важливо відзначити, що системи дистанційного тестування стимулюють студентів до самостійної підготовки та розвитку навичок електронної грамотності. Вони надають можливість використовувати сучасні технології для отримання та демонстрації знань.

Звісно, такі системи також стикаються з викликами, такими як забезпечення безпеки інформації, адаптація для різних груп студентів і стійкість до технічних неполадок. Проте, вони продовжують стати необхідною складовою у впровадженні сучасних підходів до навчання та оцінювання в українських вишах.

На сьогодні в Україні використовуються різноманітні системи дистанційного тестування, які надають можливість проведення ефективного оцінювання знань студентів та учасників навчального процесу. Нижче наведено опис деяких з основних систем, які користуються популярністю в українському освітньому середовищі.

Програма «Айрен» не дозволяє розподіляти запитання за рівнями складності та темами, запитання можуть лише відноситись до певного тесту, тобто, якщо тест охоплює кілька тем, програмний продукт не надає користувачу можливості зазначити та використовувати це. Також програма зберігає результати тестування у вигляді окремих файлів, що не дає змоги впровадження автоматизованої оцінки складності запитань після проходження певною кількістю учасникыв заданого тесту.

Запитання до варіанта тесту можуть додаватися в тому порядку, в якому вони були створені або перемішуватись випадковим чином всередині варіанту при виборі прапорця «Перемішувати запитання» (рис. 1.1).

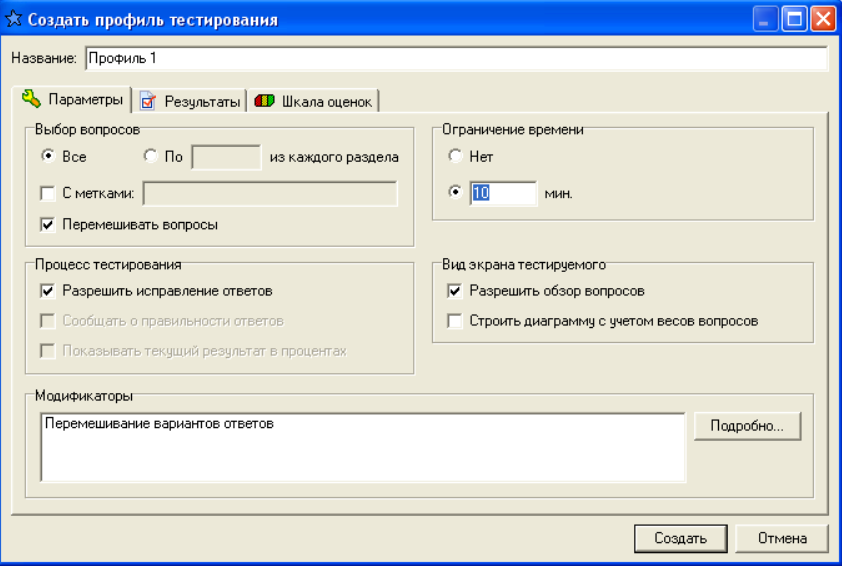


Рисунок 1.1 – Вікно редактора профілю тестування в системі тестування «Айрен»

Також система тестування «Айрен» дозволяє створювати запитання зі сценарієм, тобто запитання зі змінними параметрами, що також дозволяє забезпечити варіативність тесту (рис. 1.2). Проте для застосування цієї функції програмного продукту користувачу необхідно мати навички програмування на Pascal для написання простих сценаріїв.

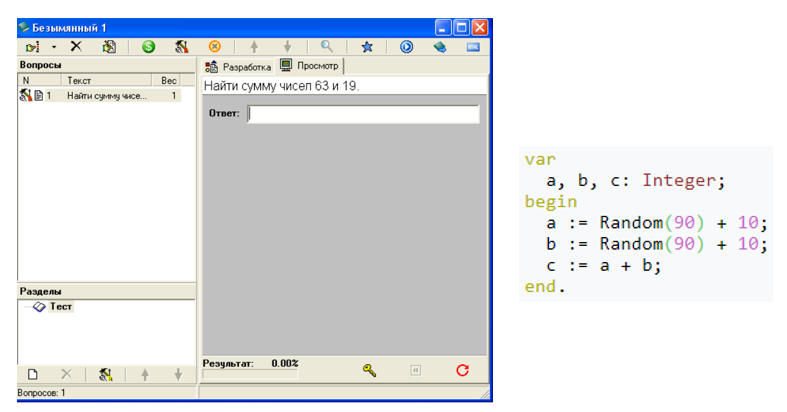


Рисунок 1.2 – Вікно редактора сценаріїв та приклад сценарію в системі тестування «Айрен»

Останнім часом у закладах освіти відзначається активний розвиток та ефективне впровадження комп'ютерних систем контролю для оцінки знань студентів. Викладачі, особливо ті, що працюють у сфері соціально-економічного напрямку, успішно використовують програму комп'ютерного тестування MyTest (рис. 1.3).



Рисунок 1.3 Логотип програми MyTestХ

MyTest представляє собою комплекс програм, включаючи інструмент для створення та проведення комп'ютерних тестів (рис. 1.4), редактор тестів (рис. 1.5) і систему ведення журналу результатів (рис. 1.6). Ця програма спрощує процес створення, збору та аналізу результатів тестування, а також дозволяє виставляти оцінки згідно з передбаченою шкалою. Легка та зручна у використанні, стала популярним інструментом серед викладачів.

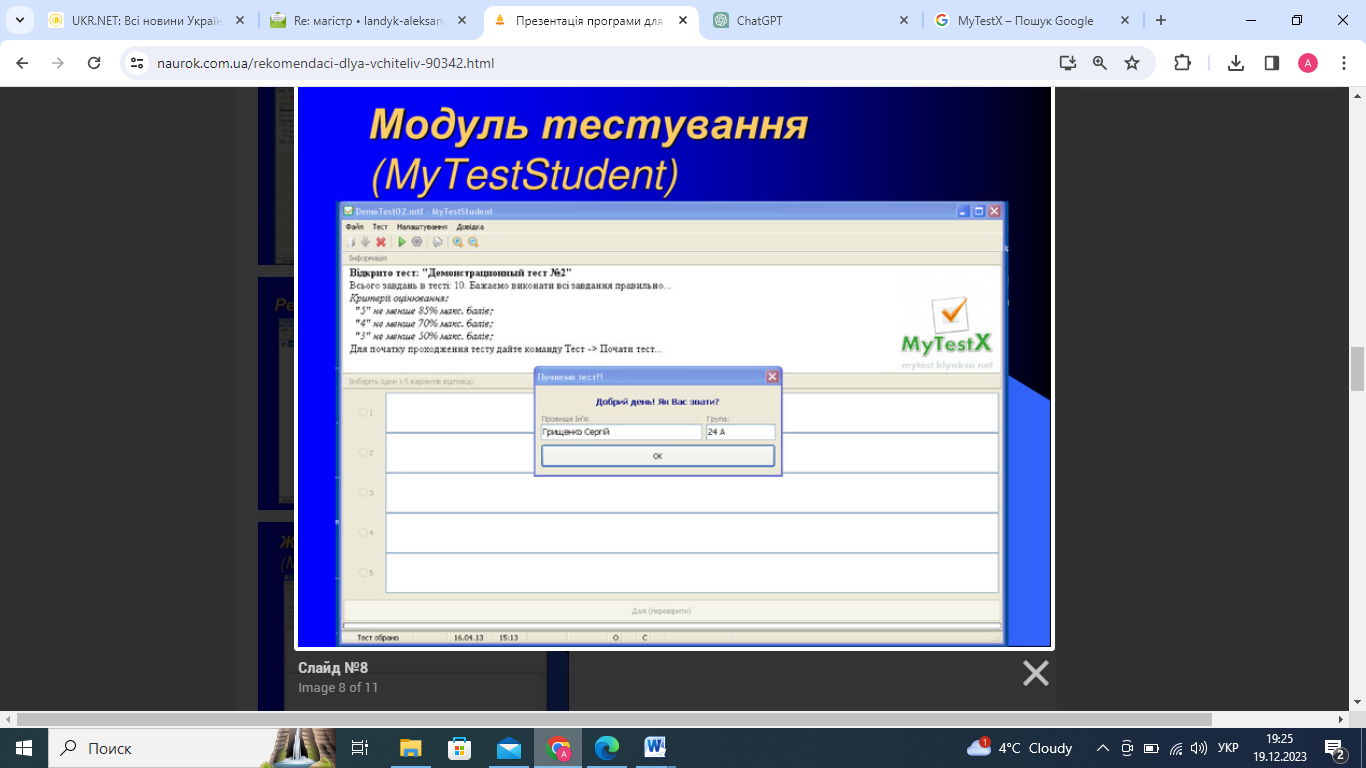


Рисунок 1.4 Модуль тестування

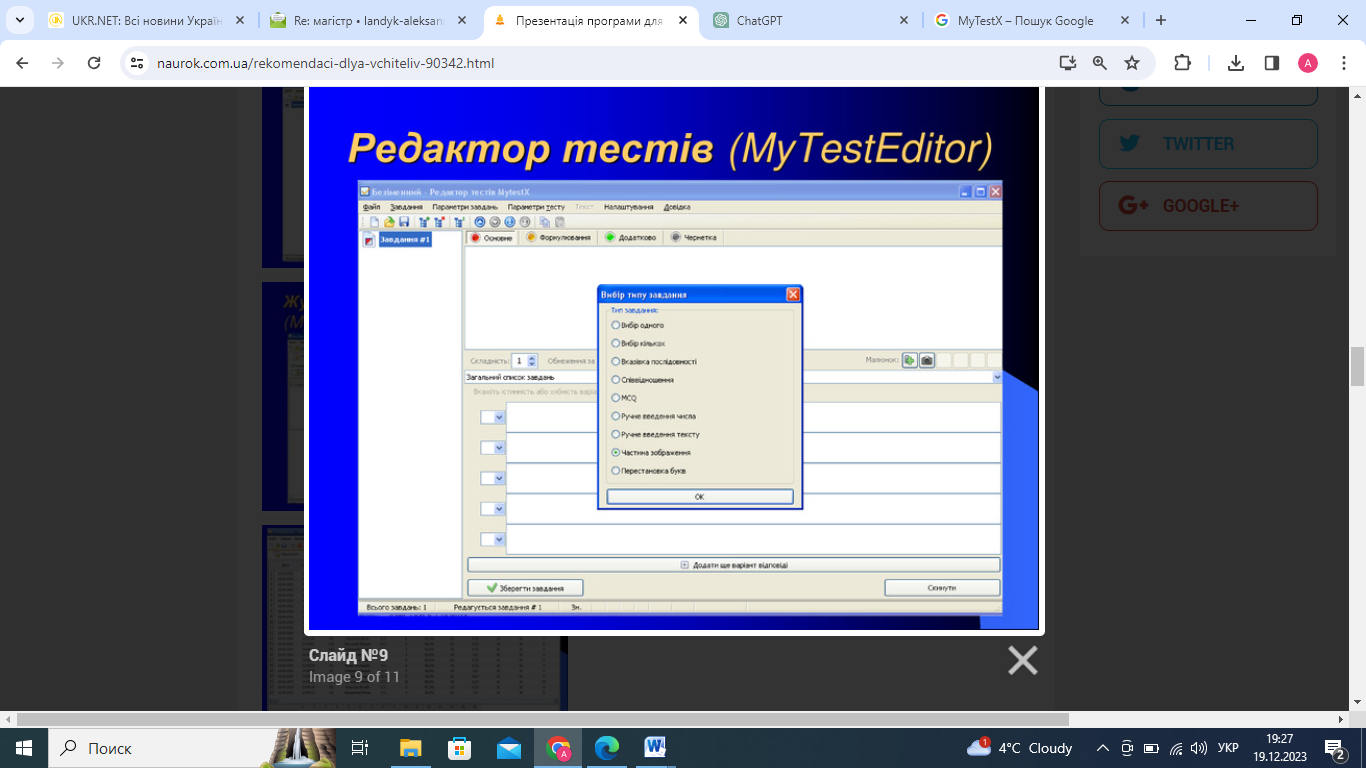


Рисунок 1.5 Редактор тестів

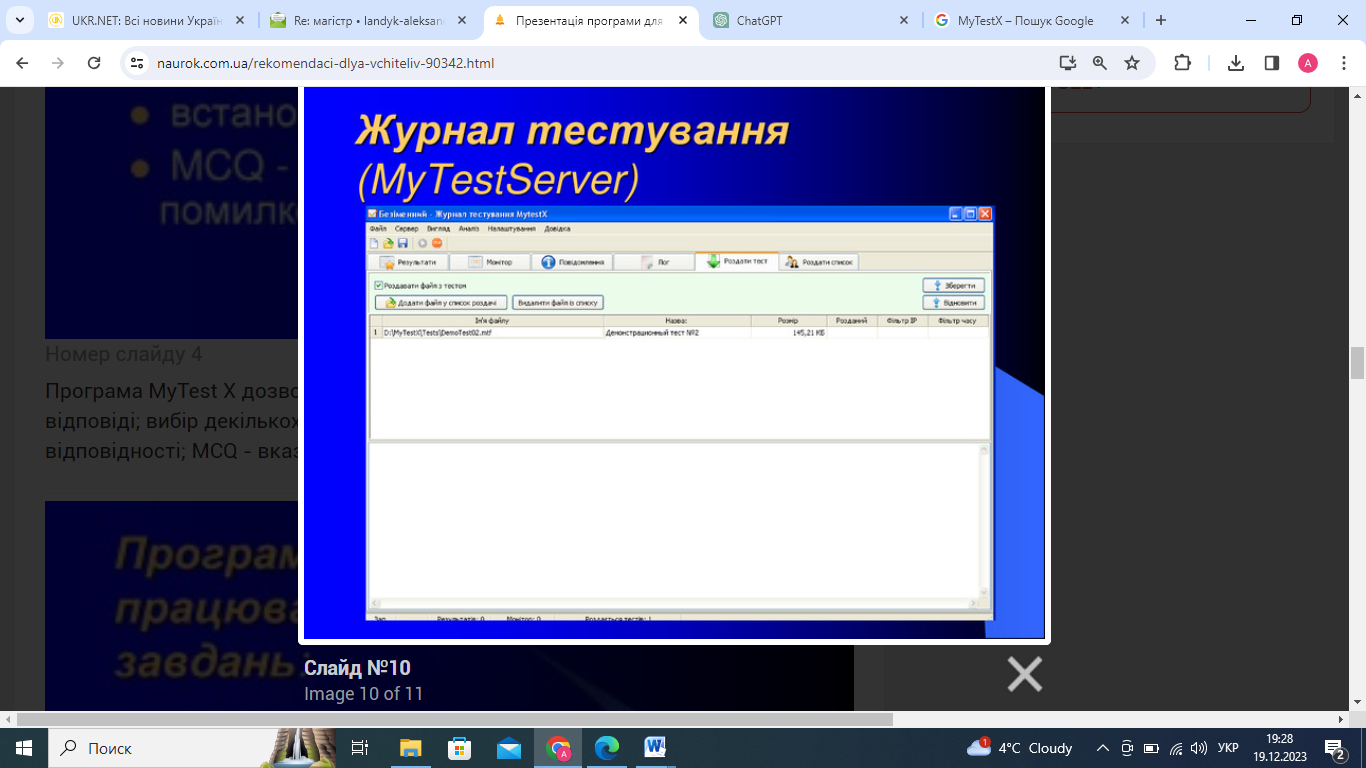


Рисунок 1.6 Журнал тестування

Програма MyTest працює з вісьмома типами завдань:

• одиночний вибір,

• множинний вибір,

• встановлення порядку проходження,

• встановлення відповідності,

• вказівка істинності чи хибності тверджень,

• ручне введення числа,

• ручне введення тексту,

• вибір місця на зображенні.

У тесті ви можете використовувати будь-яку кількість будь-якого типу, ви можете використовувати тільки одну, ви можете використовувати всі одночасно. Він підтримує кілька незалежних режимів: навчання, покарання, вільний та монопольний. У навчальному режимі випробовуваний показує повідомлення про свої помилки, може бути показано пояснення завдання. У штрафному режимі за неправильну відповідь у учасника тесту віднімаються бали, і він може пропустити завдання (бали не додаються і не віднімаються). У вільному режимі випробуваний може відповідати на запитання в будь-якому порядку, переходити до кожного питання самостійно (назад). У монопольному режимі вікно програми займає весь екран і не може бути згорнутим. При правильному підборі контрольного матеріалу зміст тесту може бути використаний не тільки для контролю, але і для навчання, що дозволяє випробуваному виявити прогалини в структурі своїх знань та вжити заходів щодо їх вирішення. У таких випадках можна говорити про значний виховний потенціал тестових завдань, що стане одним із ефективних напрямків практичної реалізації принципу єдності та взаємозв'язку навчання та контролю. За допомогою MyTest ви можете організовувати як локальні, так і мережеві тести. Якщо у вас є комп’ютерна мережа, ви можете використовувати модуль реєстрації для організації централізованого збору та обробки результатів тестування; роздача тестів студентам через мережу; безпосереднє спостереження за процесом тестування. Кожен тест має оптимальний час тесту, який зменшує або перевищує якість тесту. Тому налаштування тесту обмежують час виконання як усього тесту, так і кожної відповіді на завдання (для різних завдань можна встановити різний час).

Параметри тестування, завдання, зображення до завдань для кожного окремого тесту - все це зберігається в одному тестовому файлі. Немає баз даних, немає зайвих файлів - один тест - один файл. Тестовий файл шифрується та стискається.

Версії системи:

а) проста (MyTest) - заклад освіти може безкоштовно використовувати програму;

б) більш функціональна розширена версія (MyTestXPro) є умовно-безкоштовною програмою і поширюється за принципом «спробуй перед тим, як купити »(shareware).

MyTest працює на платформі Windows. Ви можете використовувати Wine для запуску на Linux. MyTest має хороший ступінь захисту як для тестових завдань, так і для результатів. Оскільки ви можете встановити різні паролі для тесту (відкрити, редагувати, протестувати), зіпсувати (редагувати) тест для людей, які не мають на нього права, це стає майже неможливим, оскільки неможливо отримати ключі (правильні відповіді). Оскільки результати тестів можна зберігати у захищеному файлі, який не можна редагувати, оцінки студентів завжди є об’єктивними та не залежать від лояльності. Через те, що результати тесту можна зберігати як на локальному ПК, так і паралельно на тестології ПК, ймовірність втрати результатів зводиться до нуля. Програма показала високу надійність на практиці вчителів. Віна пропонує кілька варіантів захисту тестів від несанкціонованих реакцій. Програма постійно розвивається, грамотно враховуючи потреби багатьох користувачів, а нові функції надають цікаві можливості тестування.

До багатьох корисних функцій, які включені в програму для виконання комп'ютерних тестів, можна додати, що якщо студент не може виконати тест на персональному комп'ютері з будь-якої причини (наприклад, за станом здоров'я), то буквально за 1 хвилину Ви можете сформувати «паперову» версію тесту.

До переваг програми відносять: максимально простий інтерфейс, який дозволяє користувачеві комп'ютера складати та редагувати тести з будь-яким рівнем підготовки; широкий вибір типів запитань; можливість імпорту / експорту тестів, запуск програми без інсталяції; наявність різних режимів тестування.

До недоліків програми належать обмежені параметри тестування форматування в стандартній версії; відсутність підтримки адаптивного тестування; не вимагається авторизація при виконанні тесту; обмежена статистика результатів тестів.

Система VeralTest дозволяє зберігати результати тестування лише у вигляді окремих файлів або архіву файлів. Також даний програмний продукт дає користувачу змогу групувати запитання за складністю та за темою. Редактор створення запитань цієї системи містить вкладку «Секції»   
(рис. 1.7), яка дозволяє користувачу об’єднувати запитання за рівнями складності або за темою, притому програма повідомить, якщо таке запитання вже було використано в блоці іншого рівня складності, або іншої теми.

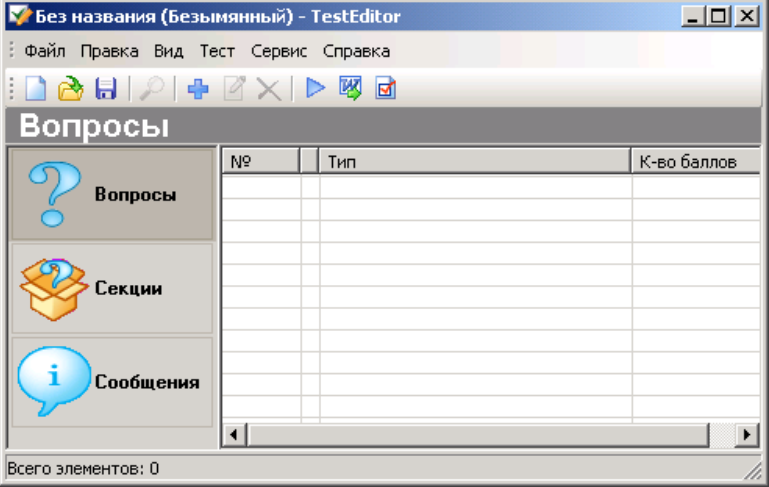


Рисунок 1.7 – Вікно редактора тестів в системі тестування VeralTest

Таке структурування запитань дозволяє накладати обмеження на кількість запитань з кожної теми при створенні варіантів тесту, а також встановлювати рівень складності (запитання будуть обиратися лише з відповідної секції), або встановити адаптивне проходження тесту – рівень складності запитань буде змінюватися залежно від відповідей тестованого, що не було можливим в жодній з попередніх розглянутих систем. Проте VeralTest також не дозволяє адаптивно змінювати рівні складності запитань згідно з результатами відповідей на них, та не дозволяє зберігати ієрархії запитань в рамках теми.

INDIGO (розробник – Indigo Software Technologies) - це професійний інструмент автоматизації процесу тестування і обробки результатів, призначений для вирішення широкого спектра завдань: тестування та контроль знань студентів, визначення професійного рівня співробітників, психологічне тестування, проведення опитувань, організація олімпіад і конкурсів (рис. 1.8).



Рисунок 1.8 Логотип програми Indigo

Тестова система встановлюється на комп'ютері (тестовому сервері) за допомогою інсталяційного пакету. Система може працювати як на ізольованому комп’ютері, так і в локальній мережі або через Інтернет. Ви можете розгорнути центр тестування на своєму комп’ютері або в хмарі на інтернет-серверах. Всі дані централізовано зберігаються в базі даних системи.

Адміністратори працюють через клієнтську програму. Адміністративні функції: (створення та редагування тестів; управління користувацьким файлом; призначення тестів користувачам; запуск тестів; перегляд результатів; створення звітів та аналіз статистики). Так багато адміністраторів різних комп’ютерів можуть працювати одночасно. Користувачі працюють через веб-браузери (Google Chrome, Mozilla Firefox, Opera, Internet Explorer, Safari та інші). Існує підтримка браузерів на мобільних пристроях (адаптивний інтерфейс). Функції користувача: реєстрація та авторизація; вибір тесту; пройти тестування; переглянути результати та помилки. Система має багатомовний веб-інтерфейс і повністю підтримує символи всіх мов (Unicode).

Система тестування INDIGO - це професійний інструмент автоматизації процесу тестування і обробки результатів, який призначений для вирішення широкого спектра завдань:

• Тестування, контроль знань учнів і студентів.

• Визначення професійного рівня співробітників, оцінка персоналу (HR).

• Підготовка до здачі іспитів і атестацій.

• Психологічне тестування.

• Організація і проведення опитувань, олімпіад, конкурсів.

Функціональні можливості тесту:

• Кількість тестів необмежено.

• Організація тестів в багаторівневої ієрархії довільної структури.

• Вбудований редактор тестів.

• Функції управління тестами: додавання / редагування / видалення; переміщення між групами; пошук; копіювання; блокування / розблокування; захист на редагування паролем; розмежування прав доступу між адміністраторами.

• Експорт / імпорт тестів (файли формату \* .itest).

• Висновок паперової версії тесту з відповідями (з можливістю друку або експорту в Word). Висновок тестів у формі бланків для тестування без комп'ютерів. Можуть використовуватися настройки автоматичної генерації безлічі випадкових варіантів тесту і ключів до них для швидкої перевірки відповідей.

• Перегляд статистики по тестах (за балами за питання і групи питань, за шкалами, контрольними позначками, відповідям).

• Експорт статистики в Excel.

Програма Indigo працює з такими типами завдань:

• вибір одного варіанта відповіді;

• вибір декількох варіантів відповіді;

• введення відповіді з клавіатури;

• установка відповідності;

• розстановка в потрібному порядку.

Підтипи питань «введення відповіді з клавіатури»:

• числовий введення - порівняння з еталоном або визначення приналежності числа заданому діапазону;

• текстовий введення - порівняння з еталоном або перевірка відповідності сумісного регулярному виразу;

• «Есе» - розгорнутий текстову відповідь для подальшої перевірки та оцінювання адміністратором (доступно дозвіл додавання користувачем файлів-вкладень).

Переваги тестової системи INDIGO: (широкий спектр програм, проста установка системи на будь-який комп’ютер, безстрокова ліцензія та відсутність плати за підписку, можливість запуску хмарної тестової системи на Інтернет-серверах, потужний інтерфейс управління, централізоване сховище та користувачі веб-інтерфейсу, безпека системи та конфіденційність даних, ієрархічне групування тестів та користувачів, правила тесту, широкі можливості редактора тестів та потужний текстовий процесор, встановлення декількох шкал для одного тесту та механізм введення формул для обчислення балів). Мінусом є те що програма є платною.

Розглянемо організацію тестування за допомогою поширеної системи управління навчання Moodle (Modular Object Oriented Dynamic Learning Environment, модульна об'єктно-орієнтована динамічному середовищі навчання). Це автоматизована система управління навчанням, яка повністю базується на інтернет-технологіях. за великим рахунком, Moodle - це веб-інструмент, до якого можна отримати доступ через браузер та підключення до Інтернету (рис. 1.9). За допомогою Moodle викладач може створити веб-сайт тренінгу, одним із компонентів якого є тести. Широке використання системи - це не тільки потужний контроль за навчальним процесом, але й той факт, що система є безкоштовною і відкритою, що дозволяє великій армії її користувачів тонко налаштовувати її для себе. Викладач, який працює в електронному середовищі, має ряд переваг при оцінці знань учнів під час тестування: тестові завдання включають стандартизовану процедуру збору та обробки даних; електронна система дозволяє перевірити знання студентів з найрізноманітніших предметів; зменшити час, витрачений на перевірку знань; виключати суб’єктивність викладача як під час процесу рецензування, так і в процесі оцінювання.



Рисунок 1.9 Логотип системи Moodle

Тест Moodle має велику кількість налаштувань і інструментів роботи, що робить його дуже гнучким. Він складається з двох компонентів: тіла тесту і бази завдань. Тіло тесту можна розглядати як контейнер для різного типу завдань з пулу. Тіло - це те, що бачать студенти при проходженні ними тестування. Завдання в тілі тесту можуть бути будь-якого типу, їх можна відображати по-порядку або випадковим чином. Бази завдань у тестовій формі можна організувати різними способами. Наприклад, можна створювати бази завдань, які будуть базуватися на розділах навчального посібника, тижнях семестру, базових поняттях і ін. Бази можна багаторазово використовувати в інших тестах, заняттях, навчальних курсах. Робота зі створення тесту починається зі створення його тіла. При створенні тіла тесту створюється контейнер для завдань і набір правил, згідно з якими відвідувачі будуть взаємодіяти з тестом. Використовуючи Moodle для організації електронних тестів, ви отримуєте справді потужний інструмент створення тестів, а також хороший аналізатор якості тестів та його компоненти - тестові завдання. Тож я спробую дати лише короткий огляд. Тестові запитання в Moodle управляються через "Банк запитань". Тести можуть вирішити проблему перевірки входу, потоку, остаточної перевірки або вони можуть бути тестовими симуляторами. У той же час тестові питання можуть бути спільними для деяких тестів, але також можуть бути вибрані випадковим чином із серії питань - обидві умови можуть бути реалізовані через "Банк запитань". Крім того, з "Банком запитань" легко організувати спільну роботу по створенню тестових питань та швидко знайти потрібне питання для тесту.

Базова збірка Moodle включає можливість створювати такі типи тестових питань (рис. 1.10):

• вибір одного / декількох з декількох;

• вірно / не вірно;

• короткий відкритий відповідь (або «питання з пропуском»);

• числова відповідь;

• обчислюється (за формулою) відповідь;

• встановлення відповідності;

• есе (перевіряється вручну);

• вкладені відповіді (комбінований).

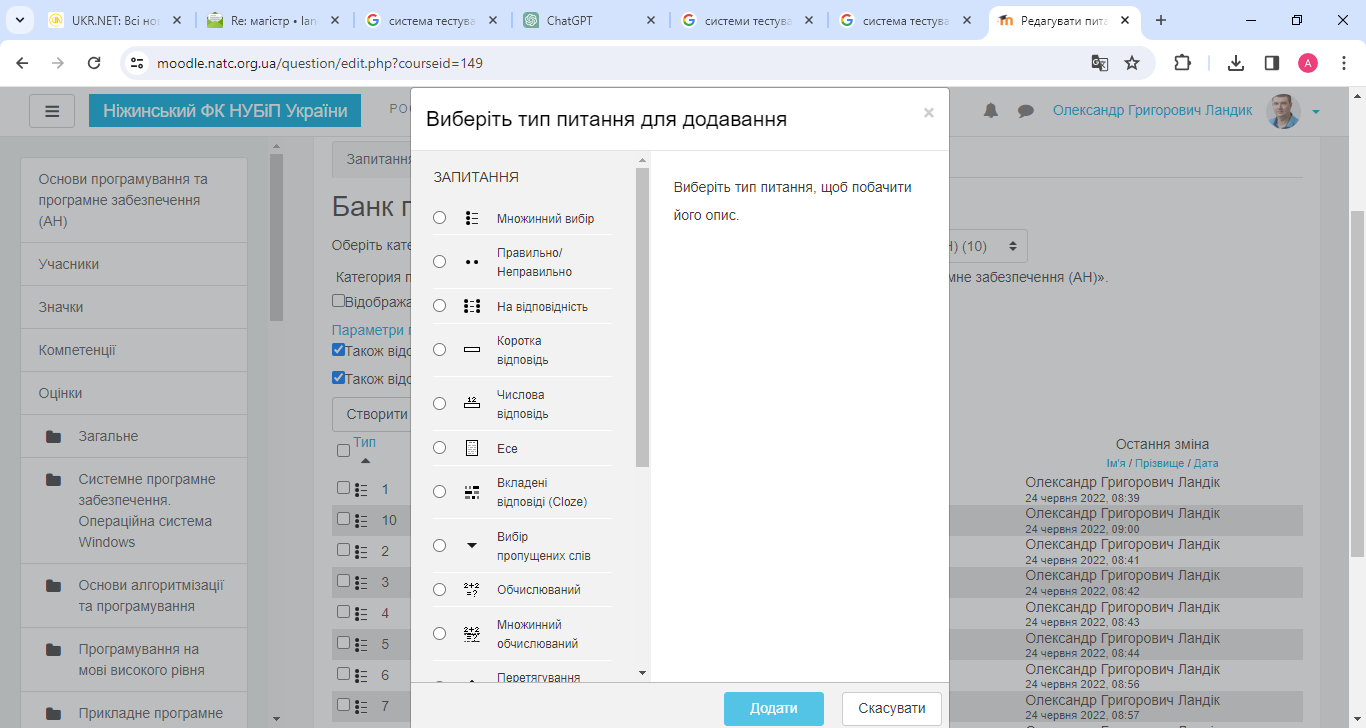


Рисунок 1.10 Вікно вибору типу створюваного питання

Також безліч плагінів для Moodle дозволяють реалізувати питання drag & drop, вибір області на зображенні, відкриті питання з механізмом перевірки, які використовують регулярні вирази, і інші типи питань.

При розробці тесту в Moodle потрібно знати наступні основні особливості системи: тест може бути обмежений у часі та кількості спроб; день і час доступності тесту можна встановити, або тест завжди може бути відкритим; порядок подання питань у тесті та варіанти відповідей на запитання можуть бути як фіксованими, так і необов’язковими; тест може виконуватися в декількох режимах, наприклад, в адаптивному режимі, випробуваному дається необмежена кількість спроб відповісти на кожне з тестових питань (передбачається, що використовуються підказки), а за неправильні відповіді можуть бути призначені штрафні бали, що відповідають буде вирахувано остаточний бал; звітування про результати тесту для обстежуваного можна гнучко регулювати; остаточна оцінка за тест складається з балів, отриманих за кожне питання в тесті, з урахуванням ваги питання. Оцінка виражається у відсотках (кількість балів від максимально можливої).

Відзначимо операції, які можна проводити в Moodle з тестовими питаннями:

• створювати (за допомогою меню «Створити нове питання»);

• переглядати;

• редагувати;

• видалити (для цього існує іконка - посилання або ж кнопка «Видалити»);

• перемістити в іншу категорію (за допомогою кнопки «Перемістити в»);

• імпортувати питання з файлу (для цього пропонується спеціальне підміню «Імпорт»);

• експортувати питання в файл (за допомогою підменю «Експорт»).

Завдяки великій кількості налаштувань тестування в Moodle є надзвичайно гнучким. Розглянемо найважливіші налаштування тесту. Налаштування згруповані в декількох функціональних блоках, серед яких можна виділити наступні: Блок «Обмеження за часом». Дата початку та закінчення тестування. установка цих дат визначає часовий період доступності тесту, за межами якого тест стає недоступним. Обмеження часу (тривалості проходження тесту). Після закінчення цього часу результати тестування автоматично передаються на сервер. Блок «Відображення». Встановлює кількість завдань, які показуються одночасно на сторінці з тестом. Випадковий порядок питань (налаштування випадкового порядку відповідей). Блок «Спроби». Кількість спроб (цей параметр використовується для обмеження кількості спроб проходження тесту. Разом з тим можливо задати і необмежену кількість спроб, вибравши для настройки значення «Необмежено»). Блок «ОЦІНКИ». Метод оцінювання (якщо дозволено кілька спроб проходження тесту, то можна обрати який спосіб буде використовуватись для оцінювання тестування. Це може бути найкраща спроба, середня оцінка, результат першої або останньої спроби). Блок «Захист». На тест можна встановити пароль, який студент повинен буде ввести, перш ніж приступати безпосередньо до тестування. Таким чином можна обмежити коло осіб, які проходять тест. Блок «Загальні налаштування модуля». За допомогою цього параметра можна вибрати режим роботи в групі, наприклад, коли студентам доступні результати лише своєї групи або вони отримують доступ до результатів тестування інших груп.

Таким чином, Moodle пропонує потужні інструменти для створення політики тестування, тестування та адміністрування. Однак майте на увазі, що використання системи пов’язане з конкретними адміністративними проблемами, які потребують кваліфікованого персоналу. Тому не всі навчальні заклади, включаючи заклади вищої освіти, просто не лише впровадять систему, але і в першу чергу її підтримають. Використання Moodle виправдане в першу чергу, якщо керівництво закладу освіти насамперед думає використовувати Moodle як автоматизовану систему управління навчанням, в якій онлайн-тестування є однією з органічних складових.

* 1. **Висновки розділу**

Проаналізувавши методи контролю та можливості створення комп’ютерного тестового контролю, можемо зробити висновок, що тестування надає можливість оцінювати середню успішність та рівень засвоєння матеріалу групи. Відмінності від інших методів оцінювання полягають в тому, що тестування надає підстави для якісного аналізу результатів навчально-пізнавальної діяльності студентів у будь-який період часу або протягом усього навчального року.

**РОЗДІЛ 2**

**СТВОРЕННЯ ТА ЗАСТОСУВАННЯ КОМП’ЮТЕРНОГО ТЕСТУВАННЯ**

**2.1 Порівняльний аналіз програм**

Ознайомившись з можливостями та функціоналом деяких програмних продуктів для створення та проведення комп’ютерного тестового контролю, було прийняте рішення проаналізувати їх за деякими критеріями.

Основні можливості налаштування тестів (табл. 2.1).

• Розбивка тесту на кілька тем з можливістю автоматичного вибору заданого числа питань по кожній темі при тестуванні. Такий поділ дозволяє оцінювати знання як по тесту в цілому, так і по темі окремо, проводити аналіз оволодіння матеріалом по окремій темі в певній групі.

• Завдання підказок для окремих питань підвищує ефективність використання тесту в навчальних цілях, дозволяє конкретизувати питання. Крім того, можна налаштувати обмеження на кількість видачі підказок і систему штрафів за їх використання.

• Використання різних режимів проходження тесту.

• Використання механізму сценаріїв для формування різних тестів на загальній базі питань. Кожному сценарію можна задати індивідуальні налаштування. Концепція сценаріїв тестування на основі загальної бази тестових завдань має такі переваги: (можливість створення нових курсів на основі наявного матеріалу без його модифікації; реалізація курсів, що поєднують елементи декількох навчальних дисциплін; скорочення обсягу бази за рахунок зменшення «повторюваності» завдань).

Таким чином, на основі одних і тих же питань можливе створення декількох сценаріїв з використанням різних алгоритмів тестування.

Таблиця 2.1

Налаштування режимів тестування в системах

|  |  |  |  |
| --- | --- | --- | --- |
| Критерії | MyTest | Indigo | Moodle |
| Підказки/довідкові матеріали | +/- | +/- | + |
| Використання тем | + | + | + |
| Режими проходження тесту | Навчальний;  штрафний;  вільний;  монопольний | - | + |
| Механізм сценаріїв | - | - | + |
| Додаткові параметри | - | - | - |

Об'єктивність тесту (табл. 2.2).

• Перемішування питань і відповідей в тесті дозволить генерувати одне завдання з різними позиціями питань і відповідей. Досить велика кількість питань виключає передачу інформацію про позицію правильної відповіді, щоб уникнути списування іншими учасниками тестування.

• Тестування можна обмежити по часу, так як кожен тест має оптимальний час тестування, перевищення якого знижує якісні показники тесту.

• Адаптивне тестування. У звичайному тесті послідовність питань лінійна і не залежить від взаємодії з користувачем. В адаптивному тесті послідовність може бути нелінійною і залежати від відповіді користувача.

Таблиця 2.2

Можливості налаштування тестування

щодо забезпечення об'єктивності

|  |  |  |  |
| --- | --- | --- | --- |
| Тип питання | MyTest | Indigo | Moodle |
| Довільний порядок | + | +/- | + |
| Адаптивні тести | - | - | + |
| Часові рамки | + | + | + |
| Обмеження спроб | + | + | + |
| Налаштування навігації | Залежить від режима; | Заборона пропуску питань;  повернення: | |
| Налаштування процесу тестування | Залежить від режима; | Заборона  дострокового  завершення; | Заборона дострокового завершення, відключення доступу  до робочого столу і панелі завдань |
| Налаштування видимої інформації в ході тестування | - | - | + |
| Повідомлення | Пояснення в разі помилки (в навчальному режимі); | При виборі пояснення, неправильній відповіді; перегляді результатів; | При неправильній відповіді- пояснення, відповідь, статус відповіді; |
| Сповіщення по електроній пошті | За вибором користувача; | - | За вибором користувача; |

Типи питань.

Для забезпечення об'єктивної оцінки знань, підвищення якості тесту і надання йому різноманітності система тестування повинна підтримувати застосування різних типів питань і технологій вибору відповіді. У досліджених системах тестування реалізовані комбінації наступних типів завдань (табл. 2.3).

• Одиночний вибір - вибір одного варіанта відповіді з декількох.

• Множинний вибір - вибір одного або декількох варіантів.

• Відповідність - упорядкування висловлювань в двох списках так, щоб вони відповідали один одному.

• Упорядкування список - розстановка відповідей в певній послідовності.

• Істинність висловлювання: вибір з двох варіантів «так» або «ні».

• Введення числа або діапазону в якості відповіді на питання.

• Заповнення пропусків у тексті - вибір варіантів або написання відповіді вручну.

• Вибір певного місця на зображенні як відповідь.

• Відкрита форма питання: введення відповіді користувача з клавіатури.

При використанні відкритої форми питання передбачаються такі варіанти завдання відповіді:

• використання системи шаблонів на основі регулярних виразів;

• перерахування відповідей, які можна зарахувати як правильні - учневі необхідно вказати один з них;

• список правильних відповідей - необхідно вказати всі правильні відповіді в довільному порядку;

• список правильних відповідей - необхідно вказати всі правильні відповіді в необхідному порядку.

Таблиця 2.3

Типи питань

|  |  |  |  |
| --- | --- | --- | --- |
| Тип питання | MyTest | Indigo | Moodle |
| Одиночний вибір | + | + | + |
| Множинний вибір | + | + | + |
| Встановлення порядку | + | + | + |
| Встановлення відповідності | + | + | + |
| Вказівку істинності  тверджень | + | + | + |
| Ручне введення числа | + | + | + |
| Вибір місця на зображенні | + | - | + |
| Перестановка букв | + | - | + |
| Заповнення пропусків | + | - | + |
| Набір суми | - | - | - |

Провівши аналіз програмних продуктів для створення і проведення комп’ютерного тестування за декількома критеріями, відразу можемо зробити висновок про лідерство платформи Moodle. Якщо збільшити кількість критеріїв, то з впевненістю можна стверджувати про доцільність застосування саме цієї платформи для створення та проведення тестування. Головною задачею у роботі постає перевірка тестових завдань на валідність тестового контролю, а не пошук та вибір кращого готового програмного забезпечення.

**2.2 Комп’ютерне тестування та його якість**

У сучасному світі комп'ютерне тестування стає все більш популярним і необхідним інструментом для вимірювання знань та навичок. Цей метод оцінювання виявляється надзвичайно зручним і ефективним для оцінювання великої кількості студентів, працівників чи кандидатів на різні посади.

Однією з головних переваг комп'ютерного тестування є швидкість та ефективність процесу. Відповіді можна отримати миттєво, що економить час і ресурси, які раніше були б витрачені на оцінювання вручну. Крім того, вірогідність помилок у результаті роботи комп'ютерних програм значно менше порівняно з оцінкою людським фахівцем.

Однак разом із зручністю виникають і виклики. Наприклад, технічні проблеми можуть призвести до перебоїв у процесі тестування, що може вплинути на об'єктивність результатів. Також, деякі критики стверджують, що комп'ютерне тестування не враховує індивідуальних особливостей кожного протестованого та не вимірює креативності чи аналітичних здібностей.

Однак розвиток технологій може вирішити ці проблеми. Наприклад, використання штучного інтелекту дозволяє персоналізувати тестування, адаптуючи його до індивідуальних потреб кожного користувача. Також, важливо розглядати комп'ютерне тестування як допоміжний інструмент, доповнюючи його іншими формами оцінювання.

У підсумку, комп'ютерне тестування стає невід'ємною частиною сучасного освітнього та кадрового менеджменту. З усім розмаїттям викликів, які воно приносить, його переваги у сфері ефективності та швидкості зробили його важливим інструментом для вимірювання знань та навичок у сучасному суспільстві.

Сучасний освітній процес визначається технологічним прогресом, що відкриває нові можливості для покращення якості навчання та оцінювання знань студентів. Одним із засобів, який впроваджується з найбільшим успіхом, є комп'ютерне тестування.

Комп'ютерне тестування надає численні переваги як для викладачів, так і для студентів. Спрощення процесу оцінювання, миттєвий доступ до результатів та індивідуалізація тестів - ось лише деякі з переваг для викладачів. Студенти, з свого боку, відзначають зручність та доступність таких тестів, що робить процес оцінювання менш напруженим та більш прозорим.

Важливо відзначити, що комп'ютерне тестування дозволяє використовувати різноманітні формати завдань - від традиційних одновибіркових питань до складних ситуаційних завдань. Це надає можливість виміряти не лише знання фактів, але й аналітичні та критичні навички студентів.

Крім того, комп'ютерні тести можуть бути адаптовані до індивідуальних потреб студентів. Завдяки алгоритмам адаптивного тестування, система може автоматично пристосовувати рівень складності питань до рівня засвоєння конкретного студента, забезпечуючи оптимальний варіант тесту.

У сучасному світі, де розвиток інформаційних технологій визначає темпи прогресу, комп'ютерне тестування виступає не лише як інструмент оцінювання, але і як засіб підвищення якості навчання. Воно стимулює активність студентів, надає можливість отримати зворотний зв'язок із викладачами та сприяє розвитку ключових компетенцій.

Крім того, комп'ютерне тестування сприяє збору об'єктивних даних про виконання студентами. Автоматизовані системи вимірюють не лише правильність відповідей, але і швидкість реакції, рівень впевненості у відповідях, а також аналізують здатність студента до критичного мислення та розв'язання завдань.

Застосування комп'ютерного тестування також дозволяє зберігати та аналізувати результати в зручний спосіб. Електронні системи забезпечують швидкий доступ до даних, дозволяючи швидко виявляти слабкі та сильні сторони навчання студентів та адаптувати програми для оптимального розвитку кожного з них.

Створити та провести комп’ютерне тестування у теперішній час не складно. Існує безліч програмних продуктів та платформ. А якщо поглибити аналіз результатів тестування, то можемо зробити висновок, що результати дуже залежать від якості тестових завдань. Найважливіше значення для тестових завдань є обґрунтування їх якості. На Заході існує система знань і технологій, що набула назву Rasch Measurement (RM). Це можна перевести як “система об’єктивованих педагогічних вимірів, що побудована за теорією Г. Раша”. Вона будується на трьох припущеннях:

1. Рівень трудності завдань і рівень підготовленості осіб, які тестуються, можна виміряти в одній шкалі, із загальною стандартною одиницею виміру.

2. За наявності такої шкали вірогідність правильної відповіді випробовуваного може стати залежною від різниці між рівнем його підготовленості і рівнем складності завдання тесту.

3. Результат протиборства особи, яка тестується, із завданнями тесту можна прогнозувати. Чим вище рівень підготовленості випробовуваного, тим вище має бути вірогідність його правильної відповіді на завдання фіксованого рівня складності [27].

Теорія Г. Раша визначає зв'язок між рівнем знань осіб, які пройшли тестування, та результатами тестів, що дозволяє визначити рівень знань незалежно від складності тестових завдань. Для визначення складності завдань і рівня знань використовується одиниця виміру, відома як логіт. Г. Раш введе дві міри: "логіт рівня знань" і "логіт рівня складності завдання". Перша визначається як натуральний логарифм відношення частки правильних відповідей особи, яка проходить тестування, до частки неправильних відповідей для всіх завдань тесту, а друга - як натуральний логарифм відношення частки неправильних відповідей на завдання тесту до частки правильних відповідей на те саме завдання для всіх осіб, що проходять тестування. Ця єдина логарифмічна шкала дозволяє встановити необхідну відповідність між рівнем навчання і складністю завдань і, більше того, внести корективи до результатів тестування для тестів різної   
складності [27].

Специфічний клас програмного забезпечення, який базується на моделі Г. Раша, виконує математико-статистичний аналіз тестових завдань для визначення їх відповідності вимогам цієї моделі. Цей клас включає в себе широкий спектр комп'ютерних програм, що відрізняються за різними критеріями, такими як обсяг вхідної інформації, кількість математико-статистичних методів, об'єм результатних таблиць і інше. Програми цього класу можуть бути як безкоштовними , так і комерційними [27].

Ці програми дозволяють ефективно обробляти великі обсяги даних, розраховувати необхідні статистичні показники, апроксимувати фактичні дані за нормальним розподілом. В результаті користувач отримує інформацію про формальні властивості кожного тестового завдання і всього тесту, рівень і структуру підготовленості осіб, які проходять тестування, надійність і валідність результатів, а також відповідність результатів осіб рівню складності завдань і інші аспекти. Програми цього класу також виконують оцінювання рівнів підготовки осіб і рівнів складності тестових завдань [27].

Наприклад, статистичний пакет QUEST призначено для аналізу як тестових завдань, так і питань соціологічних анкет. Обробка даних у ньому може проводитися як за класичною теорією педагогічних вимірів, так і за теорією RM. Програма WINMIRA має широкий набір методів обробки даних на основі Item Response Theory. Слід звернути увагу також на повну інтеграцію цього пакету із статистичним пакетом SPSS. Програмний засіб CONQUEST надає змогу проведення не тільки одновимірних, але і багатовимірних вимірів. Пакет RASCAL дозволяє шкалювати завдання і осіб, які проходять тестування, на основі одновимірної моделі Раша, якщо дані подані в дихотомічній шкалі (“0” або “1”). Але найбільш популярними і розповсюдженими програмними продуктами вважаються австралійська розробка RUMM 2020 та американський програмний засіб WINSTEPS   
(рис. 2.1) [27].

Програма WINSTEPS (США) є однією з сімейства програм, призначених для аналізу результатів тестування, побудовану на технології за теорією Г. Раша, що були розроблені під керівництвом J. M. Linacre. Програма є пропрієтарною, проте існує її безкоштовна версія, що називається MINISTEP. Вона дозволяє використовувати усі можливості WINSTEPS, але має обмеження на кількість завдань у тесті (25) і кількість осіб (75) [27].

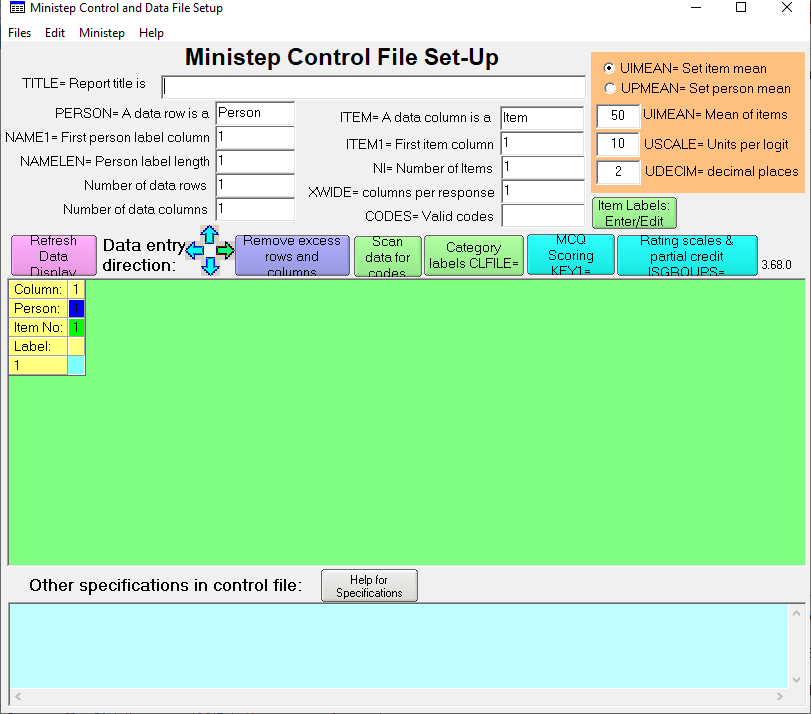


Рисунок 2.1 Вікно “Ministep Control File Set-Up”

В сучасному освітньому світі, де цифрові технології відіграють ключову роль, важливо мати ефективні інструменти для оцінювання знань та навичок студентів. Один із таких інноваційних продуктів - WINSTEPS, вирізняється своєю здатністю глибоко та об'єктивно аналізувати тестові завдання.

WINSTEPS - це програмний продукт, спроектований для математично-статистичного аналізу тестових даних, що дає можливість отримати вичерпну інформацію щодо якості та ефективності тестування. Заснований на теорії Г. Раша, WINSTEPS відкриває нові можливості для оцінювання якості тестів та рівня знань студентів. Починаючи роботу з програмою, потрібно зрозуміти концепцію формування вхідних даних. Ця концепція передбачає розділення даних на дві частини: самі дані та правилакерування цими даними. Ці компоненти можуть бути об'єднані в одному файлі, де первинні дані розташовані після інструкцій керування.

Початкові дані представляємо у вигляді рядків, що містять інформацію. Кожен рядок включає дані про одну особу, такі як ідентифікатор прізвище та послідовність відповідей на завдання тесту (рис. 2.2) [27].

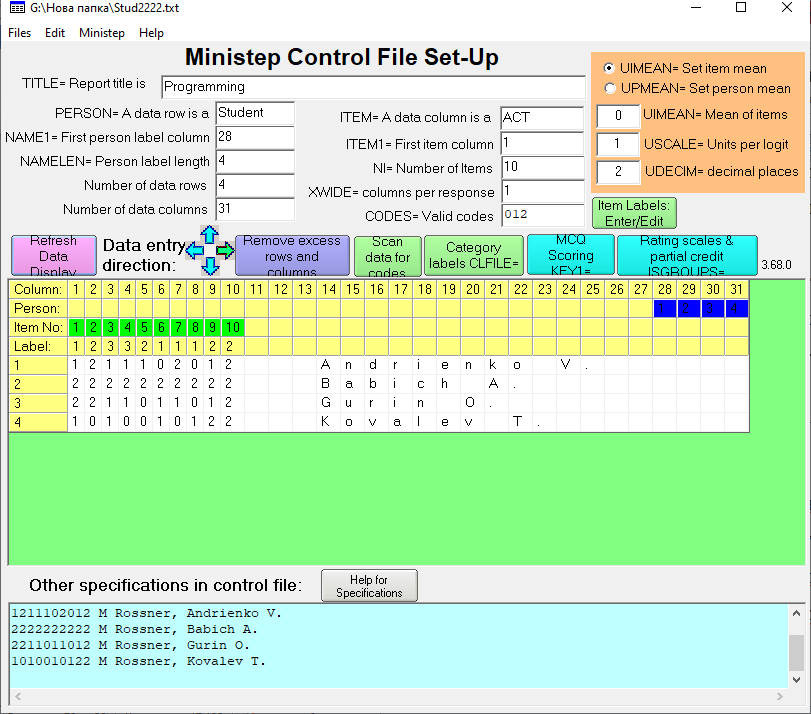


Рисунок 2.2 Введення початкових даних

Правила визначають різноманітні установки для первинних даних, розрахунків, аналізів які кодуються за допомогою певних команд, які, у свою чергу, мають атрибути. Наприклад, команди, що відображаються на наведеному вище рисунку 2.1, призначені для керування саме первинними даними. Їх призначення таке:

• TITLE. Назва тесту;

• PERSON = A data row is a. Це довідкова інформація про те, дані для яких категорій осіб містяться у рядку;

• NAME1 = First personal label column. Це числове значення, що показує, з якої позиції починаються ідентифікаційні дані;

• NAMELEN = Personal label length. Числове значення, що визначає довжину ідентифікаційних даних;

• Number of data rows. Кількість рядків (осіб);

• ITEM=A data column is a. Протокол для доступу до табличних даних;

• Number of data columns. Загальна кількість позицій у рядку;

• ITEM1 = First item column. Числове значення, що вказує з якої позиції починаються відповіді;

• NI = Number of items. Загальна кількість питань (відповідей);

• XWIDE = columns per response. Кількість позицій, що відводиться на одну відповідь (одна або дві позиції);

• CODES = Valid codes. Значення містить множину допустимих кодів для кодування відповідей, які складають суцільний рядок без проміжків.

Ввівши всі початкові дані та визначивши необхідні установки ми маємо потрібну базу для проведення перевірки якості завдань тестового контролю.

Основна перевага WINSTEPS полягає в його унікальному підході до аналізу. Програма використовує логістику, розраховуючи логіти - одиниці виміру, які відображають якість відповідей та складність завдань. Це дозволяє визначити не тільки, наскільки вірно студент відповів, але і рівень складності питань, що є ключовим для об'єктивного оцінювання.

Однією з важливих можливостей WINSTEPS є аналіз результатів на рівні окремих тестових завдань. Програма забезпечує інформацію про те, як кожне завдання поєднується зі студентськими знаннями, дозволяючи викладачам точно визначити сильні та слабкі сторони кожного студента.

Використання WINSTEPS дозволяє не тільки ефективно оцінити рівень знань, але і провести аналіз процесу тестування в цілому. Програма допомагає визначити надійність і валідність результатів, а також відповідність між рівнем підготовки студентів і складністю завдань.

Після введення даних і контролю їх вірогідності переходимо до їх аналізу, який здійснюється у головному модулі Ministep (Winstep) Rasch.

Після його завантаження програма запросить завантажити файл керування даними. Це можна зробити шляхом введення його повного імені з клавіатури, а можна викликати діалогове вікно вибору файлів.

Результат роботи програми WINSTEPS система виводить відомості про вхідні дані і таблицю збіжності даних (Convergence Table)(рис. 2.3).

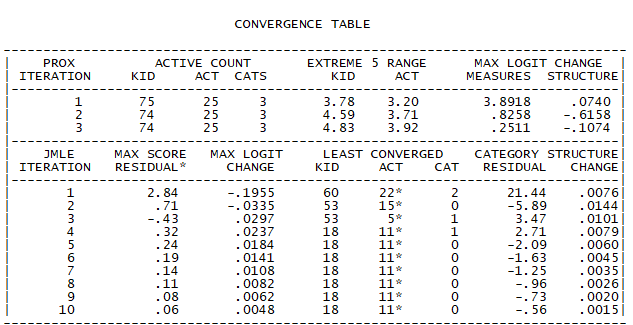


Рисунок 2.3 Результат роботи програми WINSTEPS

У стовпчиках таблиці розташовані значення, що призначені для початкового оцінювання даних. З кожною ітерацією значення (MEASURES одиниця виміру - максимальне значення логіта для випробовуваних буде зменшуватися до максимально припустимого значення збіжності (це значення є аргументом команди LCONV). Саме на це значення “орієнтується” система під час ітераційного процесу: він продовжується доки логіт не буде мати значення менше за максимально припустиме [27].

Для більш зрозумілої інтерпретації цього значення звернемося до терміну «convergence considerations» (збіжність аналізу) - це питання визначення якості результатів, що уточнює оцінювання параметрів випробовуваних і завдань методом максимальної правдоподібності [27].

WINSTEPS стає справжнім союзником для викладачів та закладів освіти, представивши новий стандарт якості в оцінюванні. З його допомогою навчання стає більш об'єктивним та науково-обгрунтованим процесом, сприяючи підвищенню якості освіти.

**2.3 Реалізація комп’ютерного тестового контролю**

У світі швидких технологічних змін та постійного прагнення до вдосконалення освітніх процесів, важливим елементом стає використання сучасних інструментів для оцінювання знань. Серед цих інструментів виступає веб-сайт для комп'ютерного тестування, що не тільки відповідає вимогам теперішнього часу України, але й допоможе створити платформу для постійного доступу до тестування з будь якої локації при умові наявності доступу до мережі Інтернет.

Створення власного веб-сайту для комп'ютерного тестування – це великий крок у напрямку сучасних технологій в освіті. Цей віртуальний інструмент відкриває безмежні можливості для ефективного вимірювання знань та вмінь студентів, пропонуючи широкий функціонал та високий рівень автоматизації.

В цій частині ми розглянемо ключові аспекти та вигоди створення власного сайту для комп'ютерного тестування. Від обрання платформи до можливостей адаптації під конкретні потреби освітнього процесу, ми дослідимо, як цей інструмент може стати необхідним елементом для викладачів, студентів та освітніх установ загалом. Розглядаючи переваги та виклики створення такого сайту, ми висвітлимо те, як цей крок може сприяти підвищенню якості та ефективності навчання у цифровому світі.

Створення сайту, який надасть зручний і ефективний спосіб для проведення тестування студентів щодо їх знань у темі типографіки. Програмний продукт повинен мати зручний інтерфейс для студентів, який дозволить їм проходити тести, обираючи відповіді та перевіряючи їх правильність. Передбачимо вікно авторизації (рис. 2.4), яке уже потім дає можливість вибрати тему тестування (рис. 2.5), та посприяє фіксації результатів за певним студентом. Тести повинні містити питання з варіантами відповідей, а також систему підрахунку балів. Система повинна бути здатною автоматично підраховувати бали за тести на основі правильних і неправильних відповідей. Це спростить процес оцінювання.

Задача полягає в створенні програмного інструмента, який допоможе спростити та автоматизувати процес тестування студентів з типографіки, забезпечивши ефективний і зручний спосіб його проходження.

Мова програмування JavaScript є динамічною, об'єктно-орієнтованою прототипною мовою програмування. Найчастіше використовується для створення сценаріїв веб-сторінок, що надає можливість на стороні клієнта (пристрої кінцевого користувача) взаємодіяти з користувачем, керувати браузером, асинхронно обмінюватися даними з сервером, змінювати структуру та зовнішній вигляд вебсторінки.

Мову JavaScript класифікують як прототипну (підмножина об'єктноорієнтованої), скриптову мову програмування з динамічною типізацією. Окрім прототипної, JavaScript також частково підтримує інші парадигми програмування (імперативну та частково функціональну) і деякі відповідні архітектурні властивості, зокрема: динамічна та слабка типізація, автоматичне керування пам'яттю, прототип не наслідування, функції як об'єкти першого класу.

Мова JavaScript використовується для:

* написання сценаріїв веб-сторінок для надання їм інтерактивності;
* створення односторінкових веб-застосунків (React, AngularJS, Vue.js);
* програмування на стороні сервера (Node.js);
* стаціонарних застосунків (Electron, NW.js);
* мобільних застосунків (React Native, Cordova);
* сценаріїв в прикладному ПЗ (наприклад, в програмах зі складу Adobe Creative Suite чи Apache JMeter);
* всередині PDF-документів тощо.

Незважаючи на схожість назв, мови Java та JavaScript є двома різними мовами, що мають відмінну семантику, хоча й мають схожі риси в стандартних бібліотеках та правилах іменування. Синтаксис обох мов отриманий «у спадок» від мови С, але семантика та дизайн JavaScript є результатом впливу мов Self та Scheme [29].

При використанні в рамках технології DHTML JavaScript-код включається в HTML-код сторінки і виконується інтерпретатором, вбудованим в браузер. Код JavaScript вставляється в теги <script></script> хоча в більшості браузерів мова сценаріїв за умовчанням саме JavaScript.

При розробці великих і нетривіальних веб-застосунків з використанням JavaScript, критично важливим є доступ до інструментів відлагодження. Оскільки браузери, від різних виробників, дещо відрізняються у поведінці JavaScript і реалізації об'єктної моделі документа, необхідно мати відлагоджувач для кожного браузера, якщо веб-застосування орієнтовано на нього.

Оскільки JavaScript є інтерпретатором, без строгої типізації, і може виконуватися в різних середовищах, кожне зі своїми власними особливостями сумісності, програміст має бути дуже уважним, і повинен перевіряти, що його код виконується як очікується в широкому переліку можливих конфігурацій.

Кожен блок сценарію інтерпретатор розбирає окремо. На веб-сторінках, коли треба комбінувати блоки JavaScript та HTML, синтаксичні помилки знайти легше, якщо тримати функції сценарію в окремому блоці коду, або (ще краще) використовувати багато малих пов'язаних .js файлів. В такий спосіб синтаксична помилка не спричинятиме «падіння» цілої сторінки, і можна надати допомогу, елегантно вийшовши зі сторінки.

HTML (HyperText Markup Language) - це стандартна мова розмітки для створення та виведення веб-сторінок та веб-додатків. HTML визначає структуру та зміст веб-документа за допомогою різних елементів та тегів.

CSS (Cascading Style Sheets) - це мова стилів, яка використовується для задання зовнішнього вигляду веб-сторінок та додатків. CSS визначає, як HTML-елементи мають відображатися на екрані, включаючи кольори, шрифти, розміри, відступи, межі та інші стилізаційні властивості.

JavaScript - це мова програмування, яка використовується для надання динамічності та інтерактивності веб-сторінкам. Вона є однією з ключових технологій для фронтенд-розробки та використовується для створення веб-додатків, ігор, взаємодії з користувачем та інших функціональностей, що роблять веб-сайти динамічними.

HTML дає змогу створити структуру запитань, відповідей та елементів інтерфейсу. CSS задає зовнішній вигляд, кольори та стилі, роблячи сайт привабливішим для користувача.

JavaScript використовується для обробки користувацьких дій, таких як вибір відповіді, підрахунок балів і перехід до результатів.

Також JavaScript дає змогу приховати та відобразити різні частини інтерфейсу в процесі користування сайтом.



Рисунок 2.4 Вікно авторизації на сайті

Після успішної авторизації система перекидає учасника тестування на вікно вибору теми (рис. 2.5).

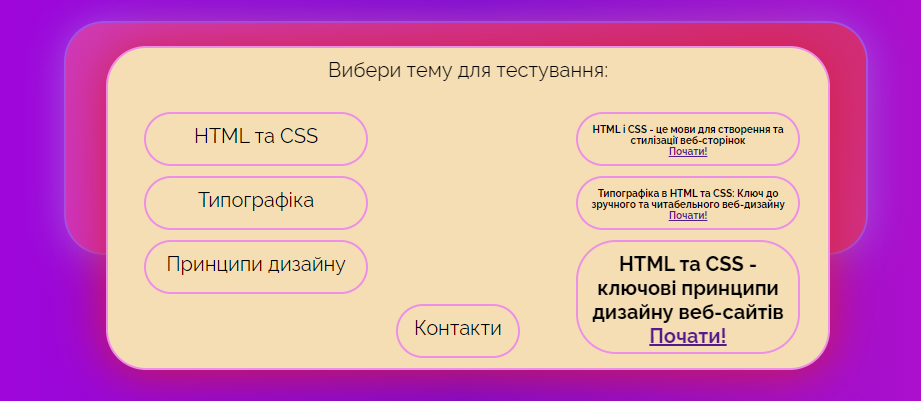


Рисунок 2.5 Вікно вибору теми для тестування

При переході на сторінку тестування (рис. 2.6) з вибраної тематики, відображаються питання створеного тестового контролю (кожне питання на окремій сторінці).

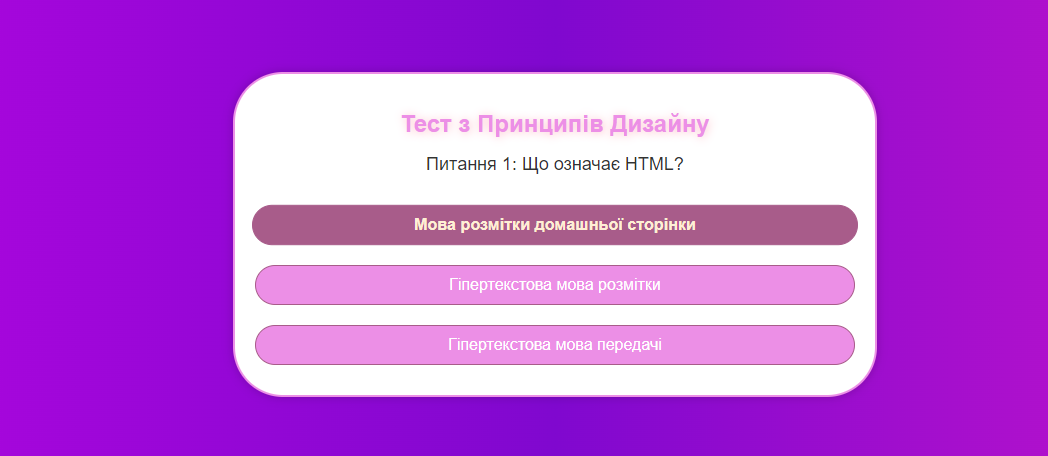


Рисунок 2.6 Сторінка тестування

Після вибору варіанта відповіді система відкриває сторінку з наступним завданням, доки учасник не дасть відповіді на усі передбачені тестом запитання і не побачить сторінку з отриманим результатом (рис. 2.7). Є можливість перейти на іншу тему тестування.

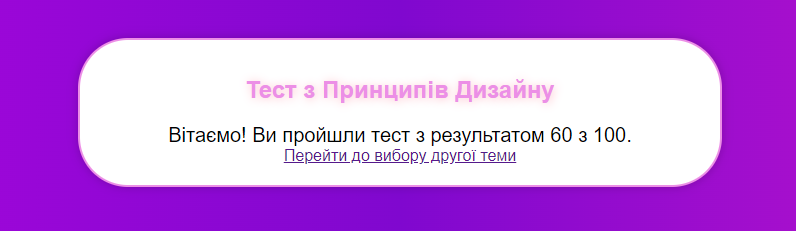


Рисунок 2.7 Сторінка результату тестування

Використання мов HTML, CSS та JavaScript при написанні сайту є важливим етапом у створенні сучасних та функціональних веб-ресурсів. Кожна з цих мов виконує свої унікальні функції, але їх спільне використання дозволяє досягти високого рівня користувацької зручності та візуальної привабливості.

HTML визначає структуру веб-сторінки, надаючи базовий каркас для розміщення контенту. Забезпечує доступність та інтерпретацію інформації браузерами та іншими інструментами.

CSS відповідає за вигляд та стиль веб-сторінки, забезпечуючи можливість створення естетичного та однорідного дизайну. Дозволяє реагувати на різні розміри екрану, що робить сайт адаптивним.

JavaScript додає інтерактивність та динамічність веб-сторінці, дозволяючи взаємодіяти з користувачем без перезавантаження сторінки. Забезпечує можливість валідації введених даних, анімації та асинхронного обміну даними з сервером.

Використання HTML, CSS та JavaScript у поєднанні дозволяє створювати повноцінні та ефективні веб-додатки з багатошаровою функціональністю. Забезпечує можливість реалізації складних дизайнів та функцій, які забезпечують кращий користувацький досвід. Використання цих мов дозволяє дотримуватися веб-стандартів, що робить сайт сумісним із різними браузерами та пристроями.

Загалом, використання HTML, CSS та JavaScript в сучасній веб-розробці є необхідним компонентом для створення привабливих, функціональних та ефективних веб-сайтів, що відповідають вимогам сучасних користувачів.

При проектуванні системи було вивчено та проаналізовано предметну область та вимоги до створення тестів. Після ретельного аналізу було вирішено розроблювати програмний продукт, який заснований на веб-технологіях для використання за допомогою веб-браузера.

Для візуального оформлення було використаномову стилів CSS, що надало веб-сторінці привабливого вигляду та кольорів.

Мова JavaScript була обрана через те, що вона є прототипною скриптовою мовою програмування з динамічною типізацією.

Програмний код сайту представлено у додатку А.

**2.4 Висновок розділу**

Провівши певний ряд досліджень та отримавши результати можемо зробити висновок, створення комп’ютерного тестового контролю можливе за допомогою багатьох програмних продуктів. Аналізуючи можливості програмних продуктів для створення тестових завдань було вирішено розробити власну систему тестування. Власний програмний продукт реалізований у вигляді веб-сторінки. Основою для створення була обрана мова розмітки HTML за допомогою якої створено структуру та зміст   
веб-сторінки. Розроблений програмний продукт є універсальним засобом для створення тестових контролів різної складності та допоможе реалізувати якісний контроль успішності студентів.

ВИСНОВКИ

Розглянувши передумови організації комп’ютерного тестового контролю, були вивчені різні методи контролю успішності студентів проаналізовані особливості створення комп’ютерного контролю використовуючи різні програмні продукти для створення тестів.

Комп'ютерний тестовий контроль є сучасним методом оцінювання, який забезпечує об'єктивність та швидкість проведення тестування. Однак, як і в інших методах оцінювання, якість тестових завдань є ключовою складовою успіху. Валідність тестових завдань визначає, наскільки вони відповідають меті вимірювання, і тому перевірка їх на валідність є важливим етапом у створенні якісного комп'ютерного тестового контролю. У процесі перевірки якості тестових завдань використали сеціалізоване програмне забезпечення WINSTEPS. Надані практичні рекомендації щодо перевірки на валідність комп’ютерних тестових завдань з детальним описом функцій та властивостей програми.

Дослідження показало, що існує ряд програмного забезпечення для комп'ютерного тестування, кожне з яких має свої переваги та недоліки. Аналіз існуючих програм засвідчив необхідність розробки власного продукту у вигляді веб-сторінки для проходження тестування. Використання веб-технологій дозволяє створити продукт, доступний для використання з будь-якої точки світу, що особливо важливо у воєнних умовах та при великій частині дистанційного навчання.

Створений програмний продукт з використанням декількох мов програмування має достатній функціонал, привабливий вигляд та головне може мати доступ з будь-якої точки.

На основі створеного програмного продукту було розроблено 10 тестових завдань з трьох тем, які стосуються мов програмування, типографіки та стилів дизайну.

Загалом програмний продукт достатньо функціональний та практичний, його користувачами можуть бути викладачі, які створюють тести , студенти, які їх проходять, а також адміністратори системи, які можуть змінювати інформацію про викладачів та студентів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Адаптивні тести: статистичні методи обробки результатів тестового контролю знань / Федорук П.І. // Математичні машини і системи. – 2007. – № 3, 4. – С. 122 – 138.
2. Андрощук А.О. Рейтингова технологія оцінки знань в навчально-виховному закладі // Педагогіка і психологія – 1996. – №3. – С.86-96
3. Блинова О.Є. Психодіагностика: психометричні основи конструювання тестів: Навч. пос. для студ. – Херсон: ПП Вишемирський В.С., 2007. – 176 с.
4. Бєлікова В. Алгоритм складання контрольних запитань як основа комплексної діагностики навчальних досягнень студентів. Проблеми інж.-пед. освіти. 2014. № 42/43. С. 120-125.
5. Білоусова Л. Потенціал комп’ютерного тестування / Л. І. Білоусова // Вісник ТІМО. 2008. № 10. С. 40-44.
6. Булах І.Є. Основи педагогічного оцінювання / Частина 1. – Київ, 2006.
7. Бурда М.І. Методичні основи диференційованого формування геометричних умінь студентів: Дис. доктора пед. наук: 13.00.02 / АПН України, Інститут педагогіки – К., 1994.- 347 с.
8. Борисенко В., Караман С., Караман О. Тестовий контроль як метод діагностики навчальних досягнень студентів. Вісн. Луган. нац. ун-ту ім. Т. Шевченка. Філол. науки. 2012. № 24. С. 110-119.
9. Близнюк С.Л. Роль оцінки в удосконаленні знань, умінь і навичок учнів. – К.: т-во “Знання”, 1983.
10. Глуханюк Д. Як об’єктивно оцінити знання учнів (з досвіду– С.86-90.
11. Гончаренко С.У. Педагогічні дослідження: методологічні поради молодим науковцям. – К.; Вінниця : ДОВ «Вінниця», 2008. – 278 с.
12. Гузій Н. В. Педагогічний професіоналізм: історико-методологічні та теоретичні аспекти : монографія / Н. В. Гузій. – К. : НПУ ім. М. П. Драгоманова, 2004. – 243 с.
13. Дремова І.А. Контроль знань учнів з алгебри в основній школі: Дисс. канд. пед. наук: 13.00.02. – Київ, 2003. – 211 с.
14. Досконалення змісту й технологій оцінювання якості підготовки майбутніх фахівців відповідно до вимог Європейської асоціації: Матеріали регіонального науково-практичного семінару / За ред. Г.В. Терещука. – Тернопіль: вид-во ТНПУ ім .В. Гнатюка, 2007. – 160 с.
15. [Ільїн, В.В., Лузан , П.Г., Рудик, Я.М. Методика тестового контролю успішності навчання студентів](http://elibrary.nubip.edu.ua/4721/1/Luzan__Monografia.pdf) - Київ, 2010. – 211 с.
16. Ландсман В.А. Застосування зовнішнього стандартизованого тестування в умовах України: Дисс.канд.пед.наук: 13.00.01. – Харків, 1999. – 212 с.
17. Ландик О.Г. Комп’ютерне тестування, форми та методи // Матеріали Всеукраїнської науково-практичної конференції педагогічних та науково-педагогічних працівників, аспірантів, молодих учених, присвяченої 125 річниці Національного університету біоресурсів і природокористування України . Зб. наук. пр. / за ред. О. Литовченко. Ніжин, 2023.С. 97—100.
18. Ландик О.Г. Технології комп’ютерного тестування знань // Всеукраїнська науково-практична конференція педагогічних та науково-педагогічних працівників, аспірантів, молодих учених, «Сучасна освіта в Україні: актуальні виклики та можливості» . Знаходиться у друці
19. Лозова В.І., Троцко А.В. Теоретичні основи виховання і навчання (навчальний посібник для студентів педагогічних навчальних закладів). – Харків, 1997. – 338 с.
20. Лисенко Л.В. Модульне навчання і рейтингова система оцінювання знань учнів // Початкова школа. –1995. –№3. –С.27-29.
21. Ніколаєва С.Ю., Петращук О.П. Тестовий контроль лексичних навичок аудіювання (англійська мова). – К.: ЛЕНВІТ, 1997. – 91c.
22. Оцінка знань студентів та якості підготовки фахівців (методичні та методологічні аспекти): Навч. посібник / А.Й. Ягодзінський, А.О. Муромцева, Л.В. Іванова та ін.; Одеський держ. економічний ун-т. – К., 1997. – 216с.
23. Олександр Виноградов. Тести як соціальна інновація в Україні. – [Електронний ресурс]. – Режим доступу: www. fulbrifht.org.ua
24. Педагогічна технологія: [Посібник] / І.Ф. Прокопенко, В.І. Євдокимов. – Харків: Основа, 1995. – 105 с.
25. Романишина Л.М. Система поетапного контролю навчальної діяльності студентів педагогічних університетів за модульно-рейтинговою технологією навчання з дисциплін природничого циклу: Дис…. доктора пед. наук: 13.00.04/Національний аграрний ун-т. – К., 1997.- 417 с.
26. Слєпкань З.І. Наукові засади педагогічного процесу у вищій школі: Навч. посіб. – К.: Вища школа, 2005. – 239 с.
27. Фетісов В.С. Комп’ютерні технології в тестуванні: навч.-метод. посіб. –Ніжин: Видавець ПП Лисенко М.М., 2011. – 140 с.
28. Measuring Educational Achievement. Front Cover. Robert L. Ebel. Prentice-Hall, 1965 - Educational tests and measurements - 481 pages.
29. Кейт Джонс. DOM Scripting: Web Design with JavaScript and the Document Object Model. / К. Джонс — Перше, 2005. — 368 c.

Додаток А

**HTML код**

<!DOCTYPE html>

<html>

<head>

  <meta charset="utf-8">

  <link content="width=device-width, initial-scale=1.0">

  <link rel="stylesheet" type="text/css" href="styles.css">

  <title>Магістерська робота</title>

  <script>

         function checkLogin() {

            var username = document.getElementById("username").value;

            var password = document.getElementById("password").value;

            if (username === "alex" && password === "land") {

                window.location.href = "маgistr.html";

            } else {

                alert("Неправильне ім'я користувача або пароль.");

            }

        }

    </script>

  <style>

  @import url('https://fonts.googleapis.com/css2?family=Montserrat:wght@900&display=swap');

  </style>

</head>

<body>

<div class="background">

<div class="container">

<div class="login" id="login">

<div class="cholka">

    <div class="ikonka1"></div>

    І ще 3 класа з таким значенням

</div>

<div class="firstname" id="firstname">

        <label for="username">Ім'я користувача:</label>

        <input type="text" id="username" name="username">

        <label for="password">Пароль:</label>

        <input type="password" id="password" name="password">

        <button onclick="checkLogin()" id="buttonlogin"><strong>Увійти</strong></button>

    </div>

</div>

</div>

</div>

</div>

</body>

</html>

<script>

    const background = document.querySelector('.background');

    const ikonka1 = document.querySelector('.ikonka1');

    const ikonka2 = document.querySelector('.ikonka2');

    const ikonka3 = document.querySelector('.ikonka3');

    const ikonka4 = document.querySelector('.ikonka4');

    ikonka1.addEventListener('click', function() {

      background.className = 'background change1';

    });

    ikonka2.addEventListener('click', function() {

      background.className = 'background change2';

    });

    ikonka3.addEventListener('click', function() {

      background.className = 'background change3';

    });

    ikonka4.addEventListener('click', function() {

      background.className = 'background change4';

    });

  </script>

**CSS код**

.background{

position: fixed;

top: 0;

left: 0;

width: 100%;

height: 100%;

object-fit: cover;

background-image: url(fon.jpg);

background-size: cover;

background-position: center;

background-repeat: no-repeat;

transition-property: all;

transition-duration: 0.5s;

}

.login{

  position: absolute;

  box-shadow: 0px 2px 60px rgba(255, 0, 0, 0.308);

  top: 50%;

  left: 50%;

  transform: translate(-50%, -50%);

  border: 2px solid;

  border-color: #A153EA;

  border-radius: 40px;

  vertical-align: middle;

  display: table-cell;

  width: 300px;

  height: 400px;

  background: url(https://i.pinimg.com/originals/c7/1f/87/c71f87a30b503eda26410c02ea99839d.gif);

  background-size: 100% 100%;

  animation: gradientAnimation 5s linear infinite, gradientShadow 5s ease-in-out infinite ;

  transition-duration: 0.3s;

  overflow: hidden;

  animation-direction: alternate;

}

@keyframes gradientAnimation {

  0% {

    background-position: 0% 50%;

  }

  50% {

    background-position: 100% 50%;

  }

}

@keyframes gradientShadow {

  0% {

    box-shadow: 0px 2px 60px rgba(0, 255, 200, 0.568);

  }

25% {

    box-shadow: 0px 2px 60px rgba(226, 114, 9, 0.61);

  }

  50% {

    box-shadow: 0px 2px 60px rgba(55, 0, 255, 0.596);

  }

  100%{

    box-shadow: 0px 2px 60px rgba(255, 0, 0, 0.479);

  }

}

.firstname{

font-family: 'Montserrat', sans-serif;

color: #FAB440;

transition-duration: 0.3s;

position: absolute;

border-radius: 40px;

border: 2px solid;

border-color: #EC8FE6;

width: 200px;

height: 130px;

top: 6%;

left: 13.5%;

background-image: url(Anderson.gif);

background-size: cover;

text-align: center;

padding: 10px;

font-size: 20px;

}

.login:hover{

scale: 1.05;

position: absolute;

}

.login:hover .firstname{

padding: 15px;

font-size: 20px;

top: 7%;

left: 11.2%;

width: 200px;

height: 130px;

}

#buttonlogin{

font-family: 'Montserrat', sans-serif;

color: #8078FE;

border: 2px solid;

border-radius: 10px;

border-color: #8078FE;

}

input{

border-radius: 10px;

border: 2px solid;

transition: 0.2s ease;

}

input:hover{

transition: 0.2s ease;

padding: 0px;

}

.cholka{

  font-family: 'Montserrat', sans-serif;

color: #FAB440;

transition-duration: 0.3s;

position: absolute;

border-radius: 40px;

border: 2px solid;

border-color: #EC8FE6;

width: 250px;

height: 30px;

top: 80%;

left: 5%;

background-color: #8078FE;

background-size: cover;

text-align: center;

padding: 10px;

font-size: 20px;

}

.ikonka1{

  position: absolute;

  width: 50px;

height: 30px;

left: 4%;

background: url(https://i.pinimg.com/originals/f1/ae/9a/f1ae9a4a69498ad64a8c679b8e411b4d.gif);

}

.background.change1 {

  background-image:

url(<https://i.pinimg.com/originals/f1/ae/9a/f1ae9a4a69498ad64a8c679b8e411b4d.gif>); Ще три background.change, тільки з різно. Cсилко. на фон

**HTML код**

<!DOCTYPE HTML>

<html>

  <head>

    <title>Тестування</title>

    <meta charset="utf-8">

    <link content="width=device-width, initial-scale=1.0">

    <link rel="stylesheet" type="text/css" href="style2.css">

</head>

<style>

@import url('https://fonts.googleapis.com/css2?family=Montserrat:wght@900&family=Raleway:wght@400;600;700&display=swap');

    </style>

<body>

<div class="background"></div>

<div class="login" id="loginfon">

  <div class="fon1" id="clickableDiv"></div>

<div class="firstname">Вибери тему для тестування:

  <div class="hidetema1" id="hidetema1">  </div>

<div class="tema1" onclick="togglePodtema1()">HTML та CSS</div>

<div class="podtema1" id="podtema1">

  <strong>

    HTML і CSS - це мови для створення та стилізації веб-сторінок<br><a

href="testcsshtml.html">Почати!</a><br></strong>

</div>

<div class="contact" id="contact" onclick="toggleContact()">Контакти</div>

<div class="tema2" onclick="togglePodtema2()">Типографіка</div>

<div class="podtema2" id="podtema2"><strong>Типографіка в HTML та CSS: Ключ до зручного та читабельного веб-дизайну<br><a href="testtipo.html">Почати!</a><br></strong></div>

<div class="tema3" onclick="togglePodtema3()">Принципи дизайну</div>

<div class="podtema3" id="podtema3"><strong>HTML та CSS - ключові принципи дизайну веб-сайтів<br><a href="testdyzain.html">Почати!</a><br></strong></div>

</div>

</div>

  </body>

  </html>

  <script>

  const tema1 = document.querySelector('.tema1');

const podtema1 = document.querySelector('.podtema1');

tema1.addEventListener('click', function() {

  if (podtema1.style.display === 'none' || podtema1.style.display === '') {

    podtema1.style.display = 'block';

    setTimeout(() => {

      podtema1.style.opacity = 1;

    }, 10);

  } else {

    podtema1.style.opacity = 0;

    setTimeout(() => {

      podtema1.style.display = 'none';

    }, 500);

  }

});

const tema2 = document.querySelector('.tema2');

const podtema2 = document.querySelector('.podtema2');

tema2.addEventListener('click', function() {

  if (podtema2.style.display === 'none' || podtema2.style.display === '') {

    podtema2.style.display = 'block';

    setTimeout(() => {

      podtema2.style.opacity = 1;

    }, 10);

  } else {

    podtema2.style.opacity = 0;

    setTimeout(() => {

      podtema2.style.display = 'none';

    }, 500);

  }

});

const tema3 = document.querySelector('.tema3');

const podtema3 = document.querySelector('.podtema3');

tema3.addEventListener('click', function() {

  if (podtema3.style.display === 'none' || podtema3.style.display === '') {

    podtema3.style.display = 'block';

    setTimeout(() => {

      podtema3.style.opacity = 1;

    }, 10);

  } else {

    podtema3.style.opacity = 0;

    setTimeout(() => {

      podtema3.style.display = 'none';

    }, 500);

  }

});

var contactElement = document.getElementById('contact');

contactElement.addEventListener('click', function() {

  if (contactElement.style.width === '400px') {

    // Якщо ширину вже змінено, повертаємо в початковий стан

    contactElement.style.width = ''; // Прибираємо значення ширини, щоб воно успадковувалося від CSS

    contactElement.style.textAlign = 'center';

    contactElement.style.left = '40%';

    contactElement.innerHTML = 'Контакти';

    contactElement.style.fontSize = "20px";

  } else {

    // Інакше, змінюємо його стан на змінений

    contactElement.style.width = '400px';

    contactElement.style.textAlign = 'left';

    contactElement.style.left = '5%';

    contactElement.style.fontSize = "16px";

contactElement.innerHTML = 'Виконано студентом НДУ Ландик О';

  }

});

const clickableDiv = document.getElementById('clickableDiv');

const loginfon = document.getElementById('loginfon');

let clickCount = 0;

clickableDiv.addEventListener('click', function() {

  clickCount++;

  if (clickCount === 1) {

loginfon.style.background = 'url(https://i.pinimg.com/originals/3d/47/3d/3d473d62a4ae61373ed56067f00dba03.gif)';

    loginfon.style.transitionDuration = '0.3s';

  } else if (clickCount === 2) {

    loginfon.style.background = "url(https://i.pinimg.com/originals/82/d9/ba/82d9bad02b953fcd4d37cd3a23a1e882.gif)";

    loginfon.style.transitionDuration = '0.3s';

  } else if (clickCount === 3) {

    loginfon.style.background = "url(<https://i.pinimg.com/originals/04/2d/cb/042dcbfb544f23be5749fa2da677cebd>

.gif)";

    loginfon.style.transitionDuration = '0.3s';

  } else if (clickCount === 4) {

    loginfon.style.background = '';

    loginfon.style.transitionDuration = '0.3s';

    clickCount = 0;

  }

});

    </script>

**CSS код**

.background{

position: fixed;

top: 0;

left: 0;

width: 100%;

height: 100%;

object-fit: cover;

background-position: center;

background-repeat: no-repeat;

background: linear-gradient(to right,#bf05e4, #8008cf, #da19ca);

animation: gradientAnimation 20s linear infinite;

}

.login{

position: absolute;

border: 2px solid;

border-color: #A153EA;

border-radius: 40px;

box-shadow: 0px 2px 40px #A153EA;

width: 800px;

height: 230px;

top: 50%;

left: 50%;

transform: translate(-50%, -50%);

background: linear-gradient(45deg, #c816f5, #cf3baf, #d41d3c, #510fcc);

background-size: 400% 400%;

animation: gradientAnimation 20s linear infinite;

transition-duration: 0.3s;

}

@keyframes gradientAnimation {

  0% {

    background-position: 0% 50%;

  }

  100% {

    background-position: 100% 50%;

  }

}

.fon1{

  border-bottom-left-radius: 40px;

  border-top-left-radius: 40px;

  width: 40px;

  height: 230px;

  opacity: 0;

}

.firstname{

font-family: 'Raleway', sans-serif;

color: black;

transition-duration: 0.5s;

position: absolute;

border-radius: 40px;

border: 2px solid;

border-color: #EC8FE6;

box-shadow: 0px 2px 50px #d41d3c;

width: 700px;

height: 160px;

top: 10%;

left: 5%;

background-color: wheat;

text-align: center;

padding: 10px;

font-size: 20px;

}

.hidetema1{

  font-family: 'Raleway', sans-serif;

  color: black;

  transition-duration: 0.5s;

  position: absolute;

  border-radius: 40px;

  border: 2px solid;

  border-color: #EC8FE6;

  width: 700px;

  height: 160px;

  top: 0%;

  left: 0%;

  background-color: red;

  text-align: center;

  padding: 10px;

  font-size: 20px;

  display: none;

}

.login:hover .firstname{

  padding: 10px;

  font-size: 20px;

  top: 10%;

  left: 5%;

  width: 700px;

  height: 300px;

  transition-duration: 0.3s;

  }

  .login:hover .hidetema1{

    padding: 10px;

    font-size: 20px;

    top: 0%;

    left: 0%;

    width: 700px;

    height: 300px;

    transition-duration: 1s;

  }

  .login:hover .contact{

    font-size: 20px;

    width: 100px;

    height: 30px;

    top: 80%;

    left: 40%;

    transition: 0.5s;

    display: block;

    }

#buttonlogin{

font-family: 'Montserrat', sans-serif;

color: #8078FE;

border: 2px solid;

border-radius: 10px;

border-color: #8078FE;

}

input{

border-radius: 10px;

border: 2px solid;

}

.tema1{

  font-family: 'Raleway', sans-serif;

  color: black;

  transition-duration: 0.5s;

  position: absolute;

  border-radius: 40px;

  border: 2px solid;

  border-color: #EC8FE6;

  width: 200px;

  height: 30px;

  top: 20%;

  left: 5%;

  background-color: wheat;

  text-align: center;

  padding: 10px;

  font-size: 20px;

}

Та ще дві .tema2 та .tema3, тільки з різним положенням.

.podtema1{

 Значення як в .tema1

Різниця від .tema1

  left: 65%;

  display: none;

  z-index: 3;

}

Та ще два podtema2 і podtema3, тільки з різним положенням.

.podtema1:hover{

  height: 140px;

  font-size: 20px;

}

.contact{

  Значення як в .tema1, тільки інший текст

}

.contact:hover{

  scale: 1.04;

  color: yellow;

  background-color: brown;

}

.tema1:hover{

  transition: 0.2s;

  scale: 1.04;

  color: yellow;

  background-color: brown;

}

HTML, CSS, JavaScripit

<!DOCTYPE html>

<html>

<head>

    <meta charset="UTF-8">

    <title>Тест з Принципів Дизайну</title>

    <style>

        body {

            font-family: 'Arial', sans-serif;

            background: linear-gradient(to right,#bf05e4, #8008cf, #da19ca);

            margin: 0;

            padding: 0;

            display: flex;

            justify-content: center;

            align-items: center;

            height: 100vh;

        }

        .quiz-container {

            width: 80%;

            max-width: 600px;

            text-align: center;

            border: 2px solid #EC8FE6;

            background-color: #FFFFFF;

            padding: 20px;

            border-radius: 50px;

            box-shadow: 0 0 10px rgba(0, 0, 0, 0.2);

        }

h1 {

            font-size: 24px;

            color: #EC8FE6;

            text-shadow: 1px 1px 10px pink;

        }

        .question {

            font-size: 18px;

            margin-bottom: 20px;

            color: #333333;

        }

        .options {

            display: flex;

            flex-direction: column;

        }

        .option {

            margin: 10px 0;

            cursor: pointer;

            padding: 10px;

            background-color: #EC8FE6;

            border: 1px solid #A85C8A;

            border-radius: 50px;

            color: #FFFFFF;

            transition: background-color 0.3s;

        }

        .option:hover {

            background-color: #A85C8A;

            color: papayawhip;

            font-weight: bold;

            transform: scale(1.01);

            transition: 0.2s;

        }

        .option.correct {

            background-color: #4CAF50;

        }

        .option.incorrect {

            background-color: #F44336;

        }

        #result {

            font-size: 20px;

            margin-top: 20px;

            display: none;

        }

        #restart-button {

            display: none;

            background-color: #4CAF50;

            color: #fff;

            border: none;

            padding: 10px 20px;

            border-radius: 30px;

            cursor: pointer;

            border: 1px solid pink;

            transition: background-color 0.3s;

        }

        #restart-button:hover {

            background-color: #45a049;

        }

        a {

            transition: 0.3s;

        }

        a:hover {

            border-radius: 20px;

            color: cadetblue;

            font-weight: bold;

            transition: 0.3s;

        }

    </style>

</head>

<body>

    <div class="quiz-container">

        <h1>Тест з Принципів Дизайну</h1>

        <div id="question-container">

            <div class="question" id="question">

                Питання 1: Які принципи взаємодії кольорів (кольорова гармонія) використовуються в дизайні, і наведіть приклади кожного з них?

        </div>

            <div class="options">

                <div class="option" id="option1" data-correct="1" onclick="checkAnswer(1)"></div>

                Ще таких же два diva тільки зі значеня 1 і 2, data-correct="0"

            </div>

        </div>

        <div id="result"></div>

        <button id="restart-button" onclick="restartQuiz()">Почати тест спочатку</button>

        <a id="next-button" href="мagistr.html" style="display: none;">Перейти до вибору другої теми</a>

    </div>

    <script>

        let currentQuestion = 1;

        let score = 0;

        function shuffleOptions() {

            const optionsContainer = document.querySelector(".options");

            const options = Array.from(optionsContainer.querySelectorAll(".option"));

            for (let i = options.length - 1; i > 0; i--) {

                const j = Math.floor(Math.random() \* (i + 1));

                optionsContainer.appendChild(options[j]);

            }

        }

        function checkAnswer(selected) {

            const correct = document.querySelector(".option[data-correct='1']");

            if (selected === 1) {

                score += 10;

            }

            currentQuestion++;

            if (currentQuestion <= 10) {

updateQuestion();

} else {

                showResult();

            }

        }

        function updateQuestion() {

            const questionElement = document.getElementById("question");

            const option1 = document.getElementById("option1");

            const option2 = document.getElementById("option2");

            const option3 = document.getElementById("option3");

            shuffleOptions();

switch (currentQuestion) {

                case 1:

        questionElement.innerText = "Питання 1: Що означає HTML?";

        option1.innerText = " Гіпертекстова мова розмітки";

        option2.innerText = " Гіпертекстова мова передачі";

        option3.innerText = " Мова розмітки домашньої сторінки";

        break;

        Та ще 9 питаннь за такой структурой, тільки питаня різні.

}

        }

        function showResult() {

            const questionContainer = document.getElementById("question-container");

            const resultElement = document.getElementById("result");

            const restartButton = document.getElementById("restart-button");

            const nextButton = document.getElementById("next-button");

            resultElement.style.display = "block";

            questionContainer.style.display = "none";

            if (score >= 60) {

resultElement.innerHTML = "Вітаємо! Ви пройшли тест з результатом " + score + " з 100.";

nextButton.style.display = "block";

} else {

                resultElement.innerHTML = "Вибачте, ви не пройшли тест. Ваш результат " + score + " з 100. Спробуйте ще раз.";

                restartButton.style.display = "block";

            }

        }

        function restartQuiz() {

            currentQuestion = 1;

            score = 0;

            updateQuestion();

            const resultElement = document.getElementById("result");

            const restartButton = document.getElementById("restart-button");

            const nextButton = document.getElementById("next-button");

            resultElement.style.display = "none";

            restartButton.style.display = "none";

            nextButton.style.display = "none";

            const questionContainer = document.getElementById("question-container");

            questionContainer.style.display = "block";

        }

      updateQuestion();

    </script>

</body>

</html>