

Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту
(повна назва)Кафедра Інформатики
(повна назва)Рівень вищої освіти другий (магістерський)Спеціальність 122 Комп'ютерні науки
(код і повна назва)Тип програми освітньо-професійнаОсвітня програма Інформатика
(повна назва освітньої програми)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

« ____ » _____ 2024 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУстудентові Мягкому Богдану Денисовичу
(прізвище, ім'я, по батькові)1. Тема роботи Дослідження та розробка інтерактивних тестів з інформатики для мобільних пристроїв

затверджена наказом по університету від 3 листопада 2023 року № 1280Ст

2. Термін подання студентом роботи до екзаменаційної комісії 01 січня 2024 р.3. Вихідні дані до роботи науково-методична література, тестовий набір даних для проведення дослідження, теоретичні відомості про інтерактивні тести для мобільних пристроїв, перелік використаних програмних засобів: мова програмування JavaScript, середовище розробки IDE VSCode.

4. Перелік питань, що потрібно опрацювати в роботі

1. Сучасний стан розвитку інтерактивних тестів для мобільних пристроїв.
2. Актуальність дослідження сфери розробки інтерактивних тестів для мобільних пристроїв та їх роль в сучасному навчанні
3. Визначення вимог та етапів конструювання тестів.
4. Аналіз та опрацювання реалізултьатів тестування, методи їх представлення.
5. Програмна реалізація моделі дослідження та створення логіки її роботи.
6. Тестування розробленого застосунку та аналіз результатів проведеного дослідження.

5. Перелік графічного матеріалу із зазначенням креслень, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) актуальність сфери розробки інтерактивних тестів для мобільних пристроїв та їх роль в сучасному навчанні, ілюстрація роботи інтерактивних тестів, аналіз досліджених форм тестів, тестові дані, візуалізація роботи розробленого застосунку.

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	03.11.2023	
2	Аналіз завдання, підбір літератури	03.11.23-6.11.23	
3	Аналіз літератури з досліджуваної проблеми	7.11.23-12.11.23	
4	Аналіз технічних засобів	13.11.23-14.11.23	
5	Розробка методу	14.12.23-16.12.23	
6	Програмна реалізація	14.12.23-19.12.23	
7	Оформлення пояснювальної записки	10.12.23-19.12.23	
8	Перевірка на плагіат	01.12.2023	
9	Рецензування	02.12.2023	
10	Підготовка презентації та доповіді	05.12.2023	
11	Занесення роботи в електронний архів	09.01.2024	
12	Попередній захист кваліфікаційної роботи	09.01.2024	

Дата видачі завдання 3 листопада 2023 р.

Студент _____
(підпис)

Керівник роботи _____
(підпис)

к.т.н. Вечірська І.Д
(посада, прізвище, ініціали)

РЕФЕРАТ/ABSTRACT

Пояснювальна записка до кваліфікаційної роботи: 106 с., 2 табл., 40 рис., 2 дод., 40 джерел.

ІНТЕРАКТИВНІ ТЕСТИ, РОЗРОБКА ДОДАТКІВ, МОБІЛЬНІ ПРИСТРОЇ, ІНФОРМАТИКА, ТЕСТУВАННЯ ДОДАТКІВ, ПЛАТФОРМИ МОБІЛЬНИХ ПРИСТРОЇВ, ПРОГРАМУВАННЯ, ОЦІНКА ТЕСТІВ, ТЕСТУВАННЯ ПРОДУКТИВНОСТІ, ІНТЕРФЕЙС КОРИСТУВАЧА.

Об'єктом дослідження є інтерактивні тести в контексті навчання в сфері інформатики.

Метою цих досліджень є вдосконалення сучасних методів розробки інтерактивних тестів з інформатики для мобільних пристроїв, покращення користувацького досвіду при взаємодії з застосунками для створення та проходження інтерактивних тестів.

В рамках даної теми досліджується інтерактивна сфера навчання та оцінки знань з інформатики на мобільних пристроях. Обґрунтовується вибір технологій та інструментів для розробки інтерактивних тестів, що дозволяють забезпечити зручний та ефективний процес навчання. Результатом цих досліджень є розробка нових підходів до створення інтерактивних тестів для мобільних платформ, а також їхня подальша імплементація та тестування.

INTERACTIVE TESTS, ADD-ON DEVELOPMENT, MOBILE DEVICES, INFORMATICS, ADD-ON TESTING, MOBILE DEVICE PLATFORMS, PROGRAMMING, TEST EVALUATION, PERFORMANCE TESTING, USER INTERFACE.

The object of research is interactive tests in the context of computer science education.

The purpose of this research is to improve current methods of developing interactive computer science tests for mobile devices, to improve the user experience when interacting with applications for creating and passing interactive tests.

Within this topic, the interactive sphere of learning and knowledge assessment in informatics on mobile devices is explored. The choice of technologies and tools for developing interactive tests that provide a convenient and effective learning process is justified. The result of this research is the development of new approaches to creating interactive tests for mobile platforms, as well as their subsequent implementation and testing.

ЗМІСТ

Перелік умовних позначень, символів, одиниць, скорочень і термінів.....	7
Вступ.....	8
1 Сучасний стан розвитку інтерактивних тестів для мобільних пристроїв.....	10
1.1 Актуальність дослідження сфери розробки інтерактивних тестів для мобільних пристроїв та їх роль в сучасному навчанні.....	10
1.2 Роль мобільних пристроїв в сучасному навчанні.....	12
1.3 Сучасний стан сфери розробки інтерактивних тестів.....	16
1.4 Види інтерактивних тестів та їх класифікація.....	18
1.5 Постановка задачі дослідження.....	22
2 Розробка інфраструктури та обрання методології розробки інтерактивних тестів для мобільних пристроїв.....	23
2.1 Визначення вимог та етапів конструювання тестів.....	23
2.2 Формування та аналіз правил конструювання та форм тестових завдань.....	28
2.3 Аналіз та опрацювання результатів тестування, методи їх представлення.....	40
2.4 Розробка теоретичної моделі дослідження та опис логіки її роботи.....	49
3 Реалізація, тестування та оцінка інтерактивних тестів для мобільних пристроїв з інформатики з власним набором даних.....	52
3.1 Вибір технологій та інструментів розробки інтерактивних тестів для мобільних пристроїв.....	52
3.2 Налаштування програмного середовища для дослідження та розробки інтерактивних тестів	61

3.3 Створення набору тестових даних для проведення дослідження.....	70
3.4 Ілюстрація роботи інтерактивних тестів для мобільних пристроїв зі створеним набором тестових даних.....	76
3.5 Аналіз результатів тестування та оцінка ефективності, зіставлення їх з поставленими вимогами.....	95
Висновки.....	99
Перелік джерел посилання.....	100
Додаток А. Набір тестових даних, що використовувався при створенні тестів.....	104
Додаток Б. Приклади результату проходження тестування групою студентів.....	105

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ПЗ – програмне забезпечення

CV – Computer Vision

JS – JavaScript

API – Application Programming Interface

JWT – Json Web Token

SDK – Software Development Kit

URI – Uniform Resource Identifiers

CRUD – Create, Read, Update, Delete

HTML – HyperText Markup Language

CSS – Cascading Style Sheets

DOM – Document Object Model

HTTP – HyperText Transfer Protocol

NPM – Node Package Manager

ВСТУП

Сучасний етап розвитку освітніх технологій вимагає постійного удосконалення методів та засобів навчання, зокрема в сфері створення інтерактивних тестів для мобільних пристроїв. Розробка та впровадження таких застосунків стає ключовим аспектом підвищення якості навчання та сприяє активній участі студентів у навчальному процесі.

Метою даної роботи є вдосконалення сучасних методів розробки інтерактивних тестів з інформатики для мобільних пристроїв, покращення користувацького досвіду при взаємодії з застосунками для створення та проходження інтерактивних тестів. Важливим та актуальним є необхідність проведення аналізу існуючих форм та видів інтерактивних тестів, визначити вимоги до їх конструювання, обрати технології розробки та створити функціональний застосунок для їх проведення.

У сучасному освітньому середовищі інтерактивні тести стали необхідною складовою процесу навчання та оцінювання. Ця форма тестування дозволяє ефективно взаємодіяти з навчальним матеріалом та забезпечує більше можливостей для активного засвоєння знань. Особливо важливим стає використання інтерактивних тестів у сфері інформатики для мобільних пристроїв, оскільки ця галузь надає нам можливість поєднувати інтерактивність з переносністю, що дозволяє студентам навчатися де завгодно і коли завгодно.

З урахуванням широкого поширення мобільних пристроїв серед студентської аудиторії, можливість використання інтерактивних тестів на цих пристроях стає важливим фактором для забезпечення доступу до навчального контенту в будь-який час і в будь-якому місці.

Актуальність дослідження визначається необхідністю створення ефективних, інноваційних інструментів для навчання, які враховують сучасні тенденції в освіті та відповідають вимогам цифрової ери.

Сучасний світ переживає процес цифрової трансформації, що призводить до змін у всіх сферах життя. У навчанні ця тенденція виражається у використанні сучасних технологій, включаючи мобільні пристрої, для покращення процесу навчання та доступу до знань.

Інтерактивні тести не обмежуються тільки освітою. Вони знаходять застосування в наборі сфер, включаючи підбір персоналу, медичну діагностику, ігрову індустрію та багато інших. Ця універсальність робить їх потужним інструментом для розв'язання різних завдань.

Усі ці фактори свідчать про те, що дослідження та розробка інтерактивних тестів для навчання є актуальною та перспективною областю, яка сприяє покращенню освіти та навчання у сучасному світі.

1 СУЧАСНИЙ СТАН РОЗВИТКУ ІНТЕРАКТИВНИХ ТЕСТІВ ДЛЯ МОБІЛЬНИХ ПРИСТРОЇВ

1.1 Актуальність дослідження сфери розробки інтерактивних тестів для мобільних пристроїв та їх роль в сучасному навчанні.

Говорячи про сучасний стан розвитку інтерактивних тестів для мобільних пристроїв свідчить необхідно звернути увагу на стрімкий розквіт цієї технології та її значущість у сфері освіти та навчання. Мобільні телефони та планшети стали невід'ємною частиною життя багатьох людей, і це робить їх ідеальними пристроями для навчання. Це сприяє зростанню популярності і використання мобільних пристроїв у навчальних цілях.

Онлайн-освіта та освітні платформи динамічно розвиваються, і багато освітніх установ та компаній активно розробляють мобільні додатки та веб-сервіси з інтерактивними тестами для студентів та учнів. Це створює нові можливості для навчання, особливо в умовах, коли доступ до комп'ютера обмежений або відсутній.

Сучасні інтерактивні тести можуть адаптуватися до індивідуальних потреб кожного користувача. Вони можуть враховувати рівень знань, інтереси та швидкість навчання, що дозволяє навчальним платформам надавати більш ефективне та персоналізоване навчання. Це збільшує ефективність навчання та може покращити рівень розуміння матеріалу.

Інтерактивні тести використовують передові технології, такі як штучний інтелект, аналіз даних, віртуальна реальність та інші, щоб зробити навчання цікавішим та ефективнішим. Вони можуть створювати враження учням, яке наближається до реального досвіду та робить навчання більш захопливим [1].

Загалом, сучасний стан розвитку інтерактивних тестів для мобільних пристроїв свідчить про їхню важливість у покращенні якості освіти та

навчання, а також різноманітні можливості застосування в різних сферах життя.

Дослідження сфери розробки інтерактивних тестів для навчання є дуже актуальним і важливим завданням. Ця актуальність обумовлена кількома факторами.

Спершу, триваючий розвиток онлайн-освіти в останні роки став нормою в багатьох освітніх установах та компаніях. Цей тренд створює попит на якісні інтерактивні тести, які можуть бути використані в онлайн-навчанні.

Другою важливою причиною є зростання інтересу до персоналізованого навчання. Інтерактивні тести можуть бути адаптивними і враховувати індивідуальні потреби та рівень знань кожного студента. Це сприяє покращенню якості навчання та підвищенню ефективності освітнього процесу.

Також важливо враховувати можливість використання новітніх технологій у розробці інтерактивних тестів. Це дозволяє використовувати передові інструменти, такі як штучний інтелект, аналіз даних, віртуальна реальність та інші, для створення найбільш ефективних та захоплюючих тестів.

Запит на розробку та використання інтерактивних тестів з метою навчання зростає кожного дня.

Також, говорячи про дослідження актуальності даної теми необхідно звернути увагу на низку факторів і тенденцій, які впливають на сферу освіти та навчання, а саме таких як наприклад цифрова трансформація. Також інтерактивні тести можуть зробити процес навчання більш цікавим та поглибленим. Вони дозволяють створювати завдання з використанням графіки, анімації та відео, що сприяє кращому розумінню матеріалу та активному вивченню.

Також такий тип тестів дозволяє вимірювати ефективність навчання в реальному часі. Використання аналітики та звітності допомагає вчителям та викладачам вчасно виявляти та коригувати слабкі місця студентів.

Важливим аспектом є і те, що освіта у галузі інформатики та комп'ютерних наук стає все більш важливою у сучасному світі. Інтерактивні тести допомагають покращити якість навчання в цій галузі та готувати студентів до викликів цифрової епохи.

За допомогою передових технологій, таких як штучний інтелект та аналіз даних, інтерактивні тести можуть надавати користувачам негайний звіт про їхні результати, що дозволяє вчителям та учням відслідковувати прогрес та виявляти слабкі місця [2].

Усі ці аспекти підкреслюють актуальність дослідження та розробки інтерактивних тестів для мобільних пристроїв, особливо у сфері інформатики, у сучасному навчанні. Це дозволяє покращити якість навчання, підвищити мотивацію студентів та учнів та адаптувати навчальний процес до потреб сучасного суспільства.

1.2 Роль мобільних пристроїв в сучасному навчанні

Використання мобільних пристроїв в навчальних процесах та проведення тестувань є надзвичайно важливим аспектом сучасного освітнього середовища. Мобільні пристрої, такі як смартфони і планшети, забезпечують широкі можливості для поліпшення навчання та здійснення оцінки студентів.

У першу чергу, вони забезпечують доступність та зручність навчання. Мобільні пристрої завжди під рукою у більшості учнів та студентів, що дозволяє навчатися в будь-який час та в будь-якому місці. Це особливо важливо в сучасному ритмі життя, коли навчання може відбуватися як у класі, так і поза ним.

Крім того, вони сприяють активній взаємодії студентів з навчальним матеріалом, адже мати електронну копію матеріалів завжди легше, а ніж реальні копії. Також якщо порівнювати сучасні реалії та популяризацію онлайн освіти, а також порівняти вартість сучасних мобільних пристроїв та персональних комп'ютерів, то однозначно, в даному аспекті виграють мобільні пристрої.

Навчання через мобільні пристрої та інтерактивні тести також сприяють розвитку навичок самодисципліни та саморегуляції, оскільки вони вимагають від студентів планування часу та самостійності в навчальному процесі [3].

Загалом, мобільні пристрої відкривають нові горизонти для сучасного навчання, роблять його більш доступним, захопливим та ефективним, а також допомагають студентам розвивати необхідні навички для успішного навчання та розвитку.

Вони стали доступними практично для кожного та можуть бути використані для навчання та тестування в будь-якому місці та в будь-який час. Мобільні пристрої володіють інтерактивними можливостями завдяки сенсорним екранам та дотиковому ввіду, що дозволяє створювати навчальні програми та тести більш привабливими та залучаючими для студентів.

Також вони підтримують адаптивність, що означає, що завдання можуть адаптуватися до рівня складності в залежності від успішності студентів, забезпечуючи індивідуальний підхід до навчання. Мобільні пристрої також сприяють розширенню можливостей навчання завдяки підтримці мультимедіа, що дозволяє використовувати відео, аудіо та інші ресурси для кращого роз'яснення матеріалу.

Завдяки Інтернету та мобільному з'єднанню, студенти можуть швидко отримувати доступ до онлайн-ресурсів, посібників та навчальних матеріалів. Також мобільні пристрої спрощують створення, оцінку та аналіз тестів, що допомагає вчителям проводити ефективне тестування та відстежувати успішність студентів.

Застосування мобільних пристроїв у навчанні робить освіту більш доступною, цікавою та ефективною, сприяючи покращенню якості освіти та залученню студентів до навчального процесу.

Онлайн-освіта та освітні платформи швидко розвиваються, а вчителі та лектори в пошуках нових методів навчання. Інтерактивні тести стали необхідним інструментом для ефективного навчання. Вони можуть містити різноманітні завдання, від простих вибору відповіді до складних завдань, що сприяють розвитку критичного мислення та аналізу. Мобільні пристрої також забезпечують можливість віддаленого навчання та викладання, що стало особливо важливим у сучасних умовах. Вони дозволяють брати участь у віртуальних уроках, виконувати завдання та спілкуватися з вчителями та спільнотами студентів через мобільні додатки та платформи.

Інтерактивні тести також стимулюють мотивацію до навчання завдяки гейміфікації. Вони можуть пропонувати бали або винагороди за успішність у тестах, що робить навчання цікавішим та заохочує до досягнень.

Загалом, інтерактивні тести для мобільних пристроїв є потужним інструментом для покращення навчання та розвитку навичок учнів та студентів у сучасному освітньому середовищі. Вони допомагають зробити навчання більш доступним, захопливим та ефективним [4].

Мобільні пристрої мають суттєве значення в сучасному навчанні інформатики та комп'ютерних наук. Вони дозволяють забезпечити доступність та зручність навчання завдяки тому, що завжди легко доступні для більшості учнів та студентів, що дозволяє навчатися у будь-який час та в будь-якому місці.

Окрім того, мобільні пристрої є дуже практичними, оскільки вони компактні та легкі. Це дозволяє вивчати інформатику навіть під час подорожі або вільного часу.

На ринку існує велика кількість навчальних додатків, спрямованих на вивчення інформатики та комп'ютерних наук, що робить навчання більш захоплюючим та доступним.

Працюючи з мобільними пристроями, студенти можуть виконувати практичні завдання та проекти, пов'язані із створенням мобільних додатків або вебсайтів, що допомагає застосовувати свої знання на практиці.

Мобільні пристрої сприяють ефективному використанню вільного часу для навчання та надають можливість зв'язку з реальним світом через створення проектів та додатків, які мають практичне застосування.

Використання мобільних пристроїв в навчальних процесах та проведенні тестів є надзвичайно важливим аспектом сучасного освітнього середовища.

Мобільні пристрої, такі як смартфони і планшети, завжди доступні та мобільні, що дозволяє студентам навчатися та проходити тести в будь-якому місці та в будь-який час. Ця доступність робить навчання більш гнучким та зручним.

Інтерактивність мобільних пристроїв, яка забезпечується сенсорними екранами і дотиковим вводом, створює можливості для інтерактивного навчання та проходження тестів. Студенти можуть взаємодіяти з навчальними матеріалами та завданнями, що підвищує рівень залученості та зацікавленості в навчанні.

Адаптивність програм і тестів дозволяє створювати індивідуальні підходи до навчання, змінюючи рівень складності завдань в залежності від успішності студентів. Це сприяє більш ефективному засвоєнню матеріалу.

Мобільні пристрої підтримують відтворення мультимедійних ресурсів, таких як відео, аудіо та зображення, що дозволяє використовувати різноманітні навчальні матеріали для кращого розуміння та запам'ятовування інформації.

Застосування мобільних пристроїв в навчанні допомагає поліпшити якість освіти, зробити її більш доступною та цікавою для студентів та підвищити ефективність навчання в сучасному освітньому середовищі.

Загалом, мобільні пристрої відіграють важливу роль у навчанні інформатики, спрямовуючи їхню доступність, практичність та можливості навчання та застосування знань у цій важливій галузі.

1.3 Сучасний стан сфери розробки інтерактивних тестів

На сьогоднішній день сфера розробки інтерактивних тестів переживає активний розвиток і стала надзвичайно важливою в контексті навчання та оцінки знань. Сучасний стан цієї галузі вражає своєю динамікою та постійними змінами.

Популярність онлайн-навчання продовжує зростати, а це призводить до розширення використання інтерактивних тестів як інструмента для вимірювання та оцінки навичок. Вони стають необхідною складовою онлайн-освіти та надають можливість вчителям взаємодіяти зі студентами, використовуючи інтерактивні завдання.

Ще однією важливою характеристикою є можливість реалізації адаптивних тестів, які враховують індивідуальні потреби студентів та змінюють рівень складності запитань залежно від їх відповідей. Це сприяє персоналізації навчання та покращенню ефективності освітнього процесу.

Технології, такі як штучний інтелект та аналіз даних, активно використовуються для створення більш складних інтерактивних завдань та аналізу результатів. Гейміфікація також використовується для зроблення навчання більш захоплюючим і мотивуючим, завдяки елементам гри [3].

Зростання кількості доступних платформ і інструментів для створення інтерактивних тестів робить їх більш доступними та дозволяє вчителям і розробникам створювати тести, які відповідають їхнім конкретним потребам.

Загалом, сучасний стан розробки інтерактивних тестів характеризується широким застосуванням, інноваціями та активним розвитком, що допомагає покращувати навчальний процес та оцінку знань.

У сфері розробки інтерактивних тестів також можна відзначити наступні аспекти, що характеризують сучасний стан:

Зростання мобільності: розробники інтерактивних тестів активно адаптують їх для мобільних пристроїв, таких як смартфони та планшети. Це враховує змінність робочого середовища і дозволяє студентам проходити тести на будь-якому пристрої з доступом до Інтернету.

Використання мультимедіа: сучасні інтерактивні тести можуть включати різноманітні мультимедійні елементи, такі як відео, зображення та аудіо. Це робить тести більш захоплюючими та допомагає краще передавати матеріал.

Застосування штучного інтелекту: штучний інтелект використовується для автоматизації процесу створення та оцінки інтерактивних тестів. Алгоритми можуть аналізувати відповіді студентів та надавати індивідуальні рекомендації щодо подальшого навчання.

Забезпечення безпеки інформації: у зв'язку зі зростанням онлайн-тестування важливо забезпечувати безпеку інформації і запобігати плагіату. Сучасні розробники використовують різноманітні методи аутентифікації та антивідкрадачі для забезпечення чесності тестування.

Інтеграція з LMS і SIS: інтерактивні тести все частіше інтегруються з системами управління навчанням (LMS) і інформаційними системами студентів (SIS). Це спрощує управління тестами та дозволяє автоматично зберігати результати.

Дистанційне тестування: зростання популярності дистанційного навчання та роботи з віддаленими студентами призвело до збільшення попиту на інтерактивні тести, які можна проходити віддалено.

Загалом, сучасний стан сфери розробки інтерактивних тестів характеризується широким спектром інновацій та вдосконалень, спрямованих на поліпшення навчального процесу та оцінки знань [5].

Результати дослідження показують, що сфера інтерактивних тестів розвивається динамічно та відіграє важливу роль у сучасній освіті.

Інтерактивні тести стають все більш популярними завдяки зростанню онлайн-навчання та можливості використовувати їх для персоналізації навчання. Технології, такі як штучний інтелект, гейміфікація та аналіз даних, роблять інтерактивні тести більш ефективними та захоплюючими. Інтерактивні тести знаходять застосування не лише в освіті, але і в інших сферах. Загалом, ця галузь розвивається швидко та продуктивно, допомагаючи поліпшити навчання та оцінку знань.

1.4 Види інтерактивних тестів та їх класифікація

Слово «тест» походить від англійського слова «test», яке в свою чергу має корені в латинському слові «testum», що означає «гончарська чашка» або «посудина». Початково, слово «test» в англійській мові вказувало на дію або процес випробування, перевірки чогось на придатність або якість.

Значення слова «тест» включає:

- випробування. В широкому контексті, «тест» це інструмент або процедура для перевірки або вимірювання якогось параметра або якості;
- завдання. Тест також може вказувати на набір завдань або питань, які призначені для перевірки знань, навичок або здібностей людини;
- метод дослідження. У психології та науках про освіту «тест» використовується для опису методу дослідження, де людина або об'єкт піддається спеціальним завданням або ситуаціям для вимірювання певних характеристик [6].

Тестування – це процес вимірювання, оцінки або перевірки чогось за допомогою тестів. В освіті та психології тестування використовується для визначення знань, навичок, характеристик особистості та інших параметрів. Тестування може включати в себе проведення завдань, питань або ситуацій, аналіз результатів та зроблення висновків на основі отриманих даних.

Слова «тест» і «тестування» мають широке застосування в різних галузях, і вони вказують на процеси перевірки та вимірювання, які важливі для освіти, науки та інших сфер життя.

Перед початком розробки необхідно детально розглянути види інтерактивних тестів і їх класифікації, а також розкрити мети використання інтерактивних тестів з інформатики для мобільних пристроїв. Необхідність розгляду різних типів тестів, їх складність, цілі використання, типи відповідей і способи використання викликана тим, що треба зрозуміти різноманітність цього інструменту та його практичну значущість у сучасному освітньому процесі.

Розглядаючи інтерактивні тести як сутність, можна найчастіше виділяють такі види. Перший вид це текстові тести – це найпоширеніший тип інтерактивних тестів. Вони складаються з питань та варіантів відповідей. Користувачі обирають відповіді, які вони вважають вірними, і отримують результати після завершення тесту.

Множинний вибір – у цьому типі тесту користувачам пропонується вибрати одну або кілька правильних відповідей із списку варіантів [7].

Порядкові тести – вимагають встановлення правильної послідовності елементів, наприклад, розташування подій у правильному порядку або нумерація елементів відповідно до їх важливості.

Тести з відкритим відгуком – у таких тестах користувачі можуть вільно ввести свої відповіді, і їхні відповіді оцінюються вручну. Цей тип тесту зазвичай використовується для питань, на які неможливо надати конкретну відповідь.

Розглядаючи ж класифікацію тестів, можна виділити наступні критерії розподілення. Зазвичай за складністю виділяють прості тести, які містять базові запитання та відповіді, які можуть бути відразу ж очевидні для користувача. Тести середньої складності, які вимагають більше розглядання та аналізу перед вибором відповідей. Складні тести містять складні запитання або вимагають великого обсягу знань та розуміння для відповіді.

Іншим видом поділом є поділ за цілями класифікація іде поділяє тести на оцінювальні, які використовуються для оцінки знань індивіда.

Результати можуть бути використані для виділення слабких та сильних сторін студента. Також виділяють навчальні тести, які мають на меті навчати новий матеріал або поглиблювати розуміння існуючого.

Після відповіді на питання користувачам можуть надаватися пояснення.

За типом відповідей частіше всього поділ іде на одноелементні тести, у яких користувачі мають вибрати лише одну правильну відповідь та багатоелементні тести, де дозволяють користувачам вибрати більше однієї правильної відповіді.

За способом використання виокремлюють самостійні тести, в яких користувачі можуть проходити їх в будь-який час без додаткового нагляду та керовані тести, які зазвичай використовуються під наглядом вчителя або навчального спеціаліста [8].

У результаті дослідження видів інтерактивних тестів та їх класифікації можна зрозуміти, що цей інструмент став важливою частиною навчання і оцінювання в сучасній освітній системі.

Інтерактивні тести дозволяють ефективно перевіряти та засвоювати знання, а також стимулюють самонавчання та підвищують доступність навчання завдяки мобільним пристроям.

Зокрема, важливо визначити, як інтерактивні тести можуть допомогти в навчанні інформатики, яка є ключовою галуззю у сучасному світі. Розробка інноваційних методів навчання з використанням мобільних пристроїв та інтерактивних тестів може значно покращити підготовку фахівців у галузі інформатики та комп'ютерних наук.

Це дослідження видів та форм тестів також ставить перед собою завдання дослідити, як використання мобільних пристроїв та інтерактивних тестів може сприяти зростанню мотивації студентів та учнів до навчання. Можливість вивчати матеріал в інтерактивній та цікавій формі на власних

смартфонах та планшетах може збільшити зацікавленість студентів у предметі та допомогти покращити їхні навчальні результати [9].

Зокрема, важливо враховувати можливість використання різноманітних педагогічних підходів та інтерактивних методів, таких як гейміфікація та квестові завдання, щоб зробити навчання більш захоплюючим та зрозумілим для студентів.

Також необхідно буде дослідити такі аспекти як, вплив інтерактивних тестів на підвищення мотивації та залученості студентів. Дослідження може охопити аналіз того, наскільки інтерактивні тести сприяють підвищенню інтересу та мотивації студентів до навчання. Виміряти, як це впливає на їхню активність та успішність.

І на останок проаналізувати засоби оцінювання та вимірювання результатів навчання, тобто дослідити та обрати кращу систему оцінювання та вимірювання результатів навчання через інтерактивні тести, а також визначити їхню надійність та валідність

Ці аспекти дослідження можуть бути корисними для розширення обсягу та значущості вашої наукової роботи щодо розробки інтерактивних тестів для мобільних пристроїв у сучасному навчанні.

Для досягнення цієї мети, розроблений застосунок повинен допомагати визначати і проаналізувати рівень знань користувачів, а також забезпечувати можливість адаптації тестів для різних цільових аудиторій та платформ [10].

У підсумку, дослідження має на меті не лише розробити інтерактивні тести для мобільних пристроїв, але й зрозуміти, як ці тести можуть впливати на навчання та мотивацію студентів, а також виявити шляхи подолання можливих труднощів у впровадженні цих інновацій у навчальний процес.

Використання інтерактивних тестів з інформатики для мобільних пристроїв має свої вагомі переваги та відкриває нові можливості для студентів та вчителів.

1.5 Постановка задачі дослідження

Таким чином, створення інтерактивних навчальних тестів з інформатики для мобільних пристроїв є актуальним завданням для покращення навчально процесу та розвитку системи мобільних застосунків.

Об'єктом дослідження є інтерактивні тести в контексті навчання в сфері інформатики.

Метою цих досліджень є вдосконалення сучасних методів до розробки інтерактивних тестів з інформатики для мобільних пристроїв, покращення користувацького досвіду при взаємодії з застосунками для створення та проходження інтерактивних тестів.

Для досягнення мети необхідно вирішити такі завдання:

- провести аналіз існуючих видів та форм інтерактивних навчальних тестів [11].
- визначити вимоги та етапи конструювання тестів;
- обрати технології та інструменти для розробки інтерактивних тестів для мобільних пристроїв;
- розробити власний застосунок, який би міг створювати та надавати можливість проходити тести;
- створити ефективний датасет, за допомогою якого можна буде протестувати застосунок та його ефективність.

2 РОЗРОБКА ІНФРАСТРУКТУРИ ТА ОБРАННЯ МЕТОДОЛОГІЇ РОЗРОБКИ ІНТЕРАКТИВНИХ ТЕСТІВ ДЛЯ МОБІЛЬНИХ ПРИСТРОЇВ

2.1 Визначення вимог та етапів конструювання тестів

Визначення вимог та етапів інтерактивних тестів є невід'ємною частиною процесу створення ефективних інструментів оцінювання знань та навичок. У сучасному освітньому середовищі, де інтерактивні тести стають все більш поширеними, необхідно мати чіткі критерії для їх розробки. В цьому контексті, необхідно розглянути ключові аспекти визначення вимог до тестів, принципи їх конструювання та основні принципи проектування структури інтерактивних тестів.

Перед розробкою тестів важливо чітко визначити цілі та об'єктиви оцінювання, а також цільову аудиторію. Це визначить характер тестових завдань, їх складність та формат. Правильно сформульовані вимоги допомагають забезпечити, що тести будуть відповідати поставленим завданням та надавати цінну інформацію про знання та навички випробуваних. У процесі конструювання тестів, важливо дотримуватися принципів об'єктивності, надійності та зручності для користувача. Надійність тестів гарантує, що вони будуть давати стабільні результати при повторних вимірюваннях, тоді як зручність користувача сприяє ефективній взаємодії з тестами та забезпечує задоволення від проходження тестування.

Важливо підібрати оптимальний баланс між часом та кількістю завдань на тесті, оскільки недостатній час може збільшити ймовірність випадкового вибору відповідей, а надмірно короткий час при великій мотивації випробуваного може призвести до порушень дисципліни та надмірного використання підказок [12].

У сучасних умовах освіти відбувається перехід від традиційного розуміння процесу тестування до більш комплексного уявлення про створення науково-обґрунтованих педагогічних тестів. Як виявилось, не обмежившись лише знанням змісту предмета, недостатньо для розроблення якісних тестів. Це вимагає впровадження методики, яка охоплює комплекс методів аналізу як змісту, так і форми тестових завдань. Через це важливо дослідити теорію тестових методів, або, методологію. Саме поняття «методологія» в контексті створення тестів включає в себе не лише розробку тестових методів, але і їх застосування для різних цілей, інтерпретацію результатів тестування, розробку принципів організації роботи тестових центрів та інші питання, що вимагають системного підходу.

Процес створення тесту починається з чіткого визначенням його мети. Ця мета дуже важлива, оскільки вона визначає всю подальшу розробку тесту та його використання. Залежно від мети і призначення тесту, етапи розробки також можуть різнитися. Отже, важливо враховувати природу і мету кожного конкретного тесту. Також виділяється важлива різниця між двома основними категоріями тестових завдань: запитаннями з багатьма варіантами відповідей і завданнями, в яких екзаменовані мають самі надати відповіді. Ці дві категорії тестових завдань вимагають від респондентів різних когнітивних навичок, і це є ключовим аспектом при створенні тестів [13].

Важливо дотримуватися принципів складання тестових завдань, обирати правильні формати та технології конструювання, а також уникати технічних недоліків, оскільки ці чинники мають вирішальне значення для якості тесту. До того ж, тест, створений для перевірки знань учнів після вивченої теми, і тест для підсумкової перевірки повинні відрізнятися.

При складанні тестових завдань важливо враховувати індивідуальні особливості учнів, забезпечуючи різноманітність та відповідність рівню їхніх знань. Завдання повинні бути чіткими та лаконічними, уникайте подвійних запитань, які можуть призвести до непорозуміння.

В галузі розробки інтерактивних навчальних тестів існує різноманітні підходи та класифікації етапів, запропоновані різними науковцями. Проте, важливо враховувати, що після уважного аналізу та уніфікації цих класифікацій можна виділити загальні етапи процесу конструювання тестів. Розробка інтерактивних навчальних тестів зазвичай включає кілька основних кроків, які необхідно розглянути перед початком розробки тестових даних, програмного забезпечення та методології аналізу відповідей в результаті пройденого тестування (табл. 2.1).

Таблиця 2.1 – Етапи конструювання тестів

Назва етапу	Сутність етапу
1	2
1. Визначення мети тестування	Встановлення основної мети тестування. Потрібно визначити, які конкретні знання, навички або якості має оцінити тест. Для реалізації етапу необхідно визначити цілі, для яких створюється тест та зробити перелік основних параметрів, які потрібно вимірювати.
2. Визначення ресурсних можливостей розробників	Оцінка обсягу ресурсів (часу, коштів, кадрів) доступних для розробки тесту. Аналіз можливостей команди або організації, виділення часу і коштів на створення тесту.
3. Добір змісту навчального матеріалу	Визначення конкретних тем і питань, які буде оцінювати тест. Для реалізації необхідно проаналізувати навчальні програми та плани у відповідній галузі, існуючі матеріали та дослідження.
4. Конструювання технологічної матриці та її експертиза	Включає в себе створення загального плану тесту та перевірку цього плану на відповідність визначеним на початкових етапах метам. Для реалізації цього етапу розробки тесту, слід створити матрицю, яка включатиме в себе огляд всіх тем і питань.

Продовження таблиці 2.1.

1	2
5. Вибір форматів та створення тестових завдань	<p>Під «форматами та створенням тестових завдань» розуміється обрання структури або типів питань, які ви включите в тест, і створення конкретних питань або завдань для кожної теми тесту. Дія цього етапу полягає в тому, щоб визначити, якими саме тестовими завданнями буде перевірятися знання, і створити ці завдання.</p>
6. Апробація тесту	<p>Сутність цього етапу включає в себе:</p> <p>Проведення тестового тестування: Тобто проведення тесту на невеликій вибірці осіб. Це може бути невелика група осіб.</p> <p>Аналіз результатів: Оцінка результатів тестування цією малою групою, включаючи аналіз того, як учасники впоралися з завданнями, чи були завдання зрозумілими, тощо.</p> <p>Отримання відгуків: Збір відгуків від учасників тестування щодо їхнього досвіду та сприйняття тесту. Відгуки можуть стосуватися важкості завдань, чіткості формулювань, часу, необхідного для виконання, загальної враження від тесту і так далі.</p>
7. Визначення та розрахунок показників якості тестування та оцінювання результатів	<p>Реалізація цього етапу полягає в аналізі даних, зібраних під час апробації тесту. Ви маєте виміряти різні показники якості тестування і дізнатися, наскільки ефективним є тест для оцінки знань або навичок цільової аудиторії. Ці показники можуть допомогти вам внести корекції в тест та покращити його якість перед широким використанням.</p>
8. Побудова кінцевого варіанту тесту	<p>В цьому етапі передбачається підбір і організація завдань таким чином, щоб кінцевий тест був збалансованим і давав змогу ефективно оцінити навички чи знання цільової аудиторії.</p>
9. Стандартизація тесту	<p>Наостанок необхідно встановити стандарти і критерії для оцінювання результатів тестування, щоб забезпечити об'єктивність та консистентність усього процесу.</p>

Аналізуючи усі ці етапи можна зробити висновок, що основними з них є чотири, а саме конструювання тестових завдань, апробація тестових завдань, створення тесту з базового набору тестових завдань, та на останок проведення та оцінювання результатів тестування, а усі інші процеси можна віднести до того чи іншого основного етапу (рис. 2.1).

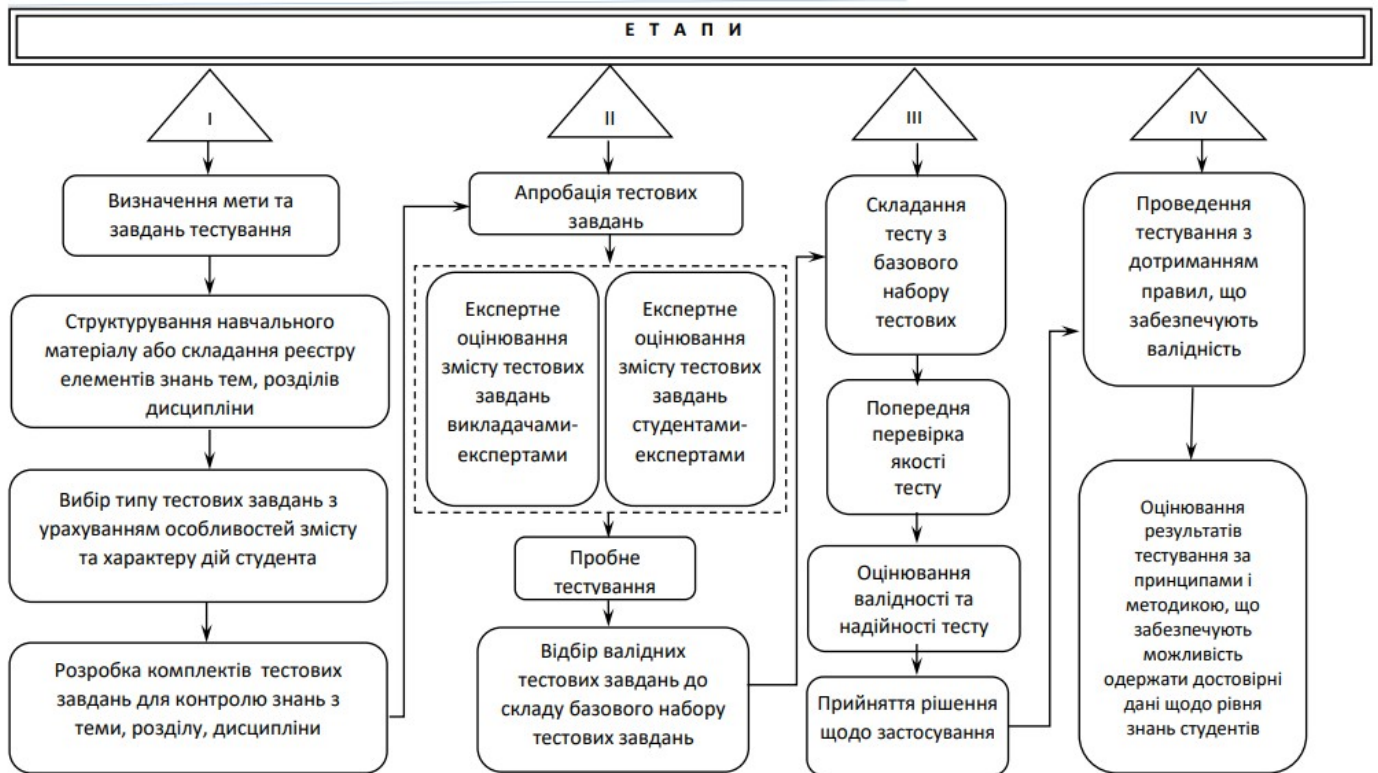


Рисунок 2.1 – Технологія створення та оцінювання тестування та навчальних досягнень

Повний цикл створення тесту завершується після проведення та оцінки тестування, коли з'являються дані та результати, які можуть бути використані для прийняття важливих рішень у навчальному або оцінювальному процесі [14]. Важливість кожного з цих етапів важко переоцінити, оскільки від їх якості залежить об'єктивність та надійність отриманих результатів. Дотримуючись цих ключових етапів, розробники

можуть створити ефективні тести, які відповідають своїм цілям і завданням, і надають надійну інформацію для подальшого аналізу та використання [15].

Загалом, дослідження основних етапів створення, адаптації та оцінювання інтерактивних тестів, а також виділення основних з них, є важливим завданням у контексті сучасної освіти та розвитку мобільних технологій. Розробка та дослідження інтерактивних тестів дозволяють оптимізувати процес тестування, зробити його більш ефективним і зручним для користувачів.

2.2 Формування та аналіз правил конструювання та форм тестових завдань

Створення якісних тестових завдань вимагає дотримання ряду важливих правил. Перш за все, тестові завдання повинні бути чіткими та однозначними. Це означає, що екзаменована особа повинна зрозуміти, що саме вимагається від неї та яка відповідь вважається правильною. Якщо тестове завдання виявляється двозначним, це може призвести до недорозуміння та некоректних результатів.

Крім того, при створенні тестових завдань слід включити дистрактори – це неправильні варіанти відповідей. Дистрактори повинні бути логічно обгрунтованими, але невірними. Це допомагає вимірювати розуміння матеріалу, а не просто вгадувати відповідь.

Збалансованість та адекватність питань також є важливими аспектами. Тест повинен покривати всі важливі теми та навички, та бути адекватним рівню тестованих осіб. Однак, важливо також враховувати мету тестування та підлаштовувати завдання під цільові знання чи навички [16].

Також жоден дистрактор не повинен бути частково правильною відповіддю, що перетворюється за певних додаткових умов на правильну

відповідь, а відповідь на одне завдання не повинна бути ключем до правильних відповідей на інші завдання тесту.

Вибір правильного формату для тестових завдань (багатовибіркові, короткі відповіді, есе) також має велике значення, а також правильне формулювання питань та перевірка граматики та структури речень. Налаштування кожного завдання відповідно до його мети та контексту допомагає створити ефективний тестовий інструмент для оцінки знань і навичок протестованих осіб.

Під час конструювання тестових завдань також важливо враховувати гнучкість тесту, дозволяючи використовувати різні типи питань та оцінювати різні аспекти знань і навичок протестованих осіб.

Окрім того, треба пильно слідкувати за ймовірністю випадкового вибору відповідей, особливо в завданнях з альтернативними варіантами відповідей, де випробуваний вибирає правильну відповідь із запропонованих варіантів. Ця ймовірність залежить від кількості альтернатив та виділеного часу на відповідь. Мотивація випробуваного також може впливати на спроби вгадати відповіді [17].

Додатково, важливо дотримуватися інструкцій щодо вибору правильних відповідей та розрахунку балів, а також мати стандартизований процес експертизи тестових завдань та вимірювання їхньої якості.

Крім того, постійний аналіз і оновлення тесту можуть забезпечити актуальність та ефективність тестового інструменту відповідно до змін в освітньому процесі та вимогах. Ще однією важливою аспектною є уникнення зближення між правильними відповідями та неправильними варіантами (дистракторами), щоб не ускладнювати процес оцінки та забезпечити точність результатів тестування.

Існує декілька думок щодо класифікації тестових завдань. Наприклад, І. Є. Булах класифікує їх за видом тестового завдання. Отож, тести бувають з відкритими та закритими тестовими завданнями (рис 2.2).



Рисунок 2.2 – Класифікація тестів за видом тестового завдання

Тести з відкритими тестовими завданнями – це такі тести, відповіді до яких не надаються ані особі, яка тестується, ані особі, яка їх перевіряє, тобто це тестові завдання з вільною формою відповіді. Цей тип тестів вимагає глибшого розуміння матеріалу та здатності аргументовано висловлювати свої думки (рис 2.3).



Рисунок 2.3 – Класифікація тестів з відкритими тестовими завданнями

Переваги:

- сприяють глибшому розумінню матеріалу;
- оцінюють здатність висловлювати власні думки та аргументувати їх;
- можуть бути використані для тестування креативності;

Недоліки:

- часозатратні у процесі оцінювання;
- суб'єктивність при оцінці відповідей;
- вимагають більш ретельної підготовки питань та критеріїв оцінювання [18].

Завдання закритої форми – тестові завдання, що пропонують тестованому перелік можливих варіантів відповідей, з яких вони мають вибрати одну або кілька правильних (рис 2.4).



Рисунок 2.4 – Класифікація тестів з закритими тестовими завданнями

Переваги:

- швидше оцінюються;
- можуть бути використані для великої кількості питань;
- можливість автоматичної обробки результатів;

Недоліки:

- обмежений виявленням реального розуміння та творчого мислення;
- можуть бути схильні до відомого випадку «угадайки» відповідь.

Тести з відкритими та закритими тестовими завданнями – це два основних типи тестів, які використовуються для оцінки знань та навичок учнів, студентів та інших осіб. Вони відрізняються за структурою та тим, яким чином відповіді оцінюються.

У практиці, часто використовуються обидва типи тестових завдань, особливо для комплексної оцінки знань та навичок учнів чи студентів. Користуючись різними видами тестів, можна отримати більш об'єктивну та різносторонню оцінку навчальних досягнень [19].

Говорячи про форми тестових завдань, вважається, що оптимальним варіантом для створення тестових завдань є формулювання їх у вигляді тверджень. Це впливає з ряду причин, адже твердження в тестових завданнях часто мають більш зрозумілий мовний стиль, оскільки вони не включають зайвих слів і не потребують складних інтерпретацій. Запитання, навпаки, може вимагати більше слів та знаків для виразу потрібного змісту і інтонації. Але важливо пам'ятати, що добре сформульоване запитання може бути так само ефективним, як і твердження у формі тестових завдань. Проте, якщо спробувати використовувати форму запитання для створення завдань з двома запереченнями, це може призвести до абсолютної незрозумілості для випробуваного.

Рекомендації голландського інституту СІТО з проблеми створення тестових завдань підкреслюють важливість використання прямих запитань, які формулюються так, щоб утворювати повні речення зі знаком запитання в кінці. На практиці виявляється, що понад 90% завдань, використовуваних під час атестації учнів, мають форму прямих запитань. Обираючи форму створення тестового завдання – запитання чи твердження, важливо враховувати два ключові аспекти: зрозумілість для випробуваних та вимоги конкретного завдання [20].

Тестові завдання, де потрібно обрати одну правильну відповідь, називаються одновибірковими. Цей формат існує вже з 1950-х років і залишається дуже популярним. Він дозволяє оцінювати рівень знань, розуміння та вміння застосовувати знання з великою об'єктивністю на різних рівнях освіти.

Переваги цього формату включають швидке, легке, об'єктивне і надійне оцінювання. Завдання одновибіркового формату мають чітку

структуру і легко зрозумілі. Вони дають можливість вимірювати знання від простих до складних, і неправильні відповіді надають інформацію про рівень освоєння матеріалу. Також цей формат мінімізує вплив випадкових відповідей порівняно з іншими видами тестів. Все це робить цей формат ефективним для оцінювання знань.

Однак у ньому є деякі недоліки, такі як великий час, потрібний для створення запитань, і важкість знаходження правдоподібних неправильних відповідей. Він також обмежений у вимірюванні навичок у вирішенні проблем та можливості студентів виразити себе (рис 2.5).

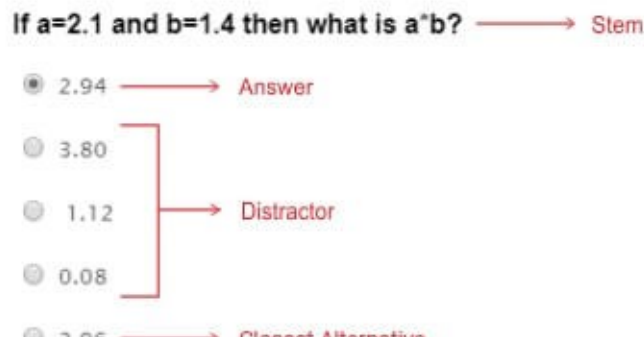


Рисунок 2.5 – Приклад одновибіркового тесту

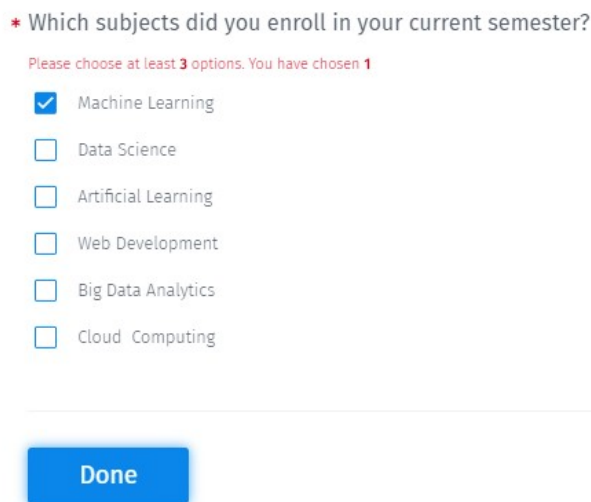
Більш складні тестові завдання це багатовибіркові. Вони дозволяють оцінити розуміння навчального матеріалу, а не лише факт засвоєння знань. У цих завданнях кількість правильних варіантів відповіді повинна бути такою ж, як і кількість неправильних або варіантів, які можуть вводити у помилку.

В освітній практиці існують дві модифікації таких завдань: ті, де заздалегідь вказано, скільки правильних відповідей потрібно вибрати, і ті, де ця кількість не вказана [21].

У масовому педагогічному тестуванні також застосовуються обидва види завдань: з заздалегідь вказаною кількістю правильних відповідей і без такої вказівки. Зазвичай кількість варіантів відповіді становить від 5 до 12. У

таких тестових завданнях екзаменованій повинен обрати всі правильні відповіді із запропонованого списку варіантів.

У цьому типі тестів кожен варіант відповіді повинен бути абсолютно правильним або абсолютно неправильним, щоб уникнути неоднозначності. Такий формат тестового завдання так само, як і попередній, допомагає діагностувати знання, розуміння та здатність до їх застосування (рис 2.6).



* Which subjects did you enroll in your current semester?

Please choose at least 3 options. You have chosen 1

- Machine Learning
- Data Science
- Artificial Learning
- Web Development
- Big Data Analytics
- Cloud Computing

Done

Рисунок 2.6 – Приклад багатовибіркового тесту

Завдання на встановлення правильної послідовності широко використовуються у самоосвіті та при навчанні студентів на дистанційних курсах. Вони дозволяють перевірити розуміння та здатність встановлювати послідовність дій, операцій, алгоритмів, етапів та стадій процесів тощо. Цей формат тестового завдання, який вимагає від учасника визначити правильний порядок подій, дій, чи об'єктів. У такому завданні, учаснику надається список елементів або подій, і він повинен визначити логічну послідовність, в якій ці елементи чи події повинні відбуватися. Завдання цього типу можуть бути використані для оцінки різних навичок і здібностей, таких як логічне мислення, розв'язання проблем, розуміння послідовностей подій, а також здатність встановлювати порядок важливих етапів або дій в різних контекстах. При складанні завдань на встановлення правильної послідовності

важливо дотримуватися чіткості та послідовності в інструкціях, а також варіювати рівень складності завдань від простих до більш складних. Такі завдання часто використовуються в освіті, психометричних тестах, а також в інших областях, де важливо оцінювати здатність особи до логічного мислення та розв'язання послідовних завдань (рис 2.7).

Order these cities by population size - use the mouse to drag the cities to left hand side, putting the city with the largest population at the top.

1	Genoa
2	Palermo
3	Rome
4	Turin
5	Naples
6	Milan

Рисунок 2.7 – Приклад тесту на встановлення правильної послідовності

Тестові завдання на встановлення відповідності – це формат завдань, де учасник повинен зіставити елементи з одного списку з відповідними елементами з іншого списку [22].

Зазвичай такі завдання складаються з двох колонок. У першій колонці містяться початкові елементи, а в другій – відповідні елементи чи описи. Учасник повинен визначити правильну відповідність між елементами з першої та другої колонки шляхом їх зіставлення.

Цей формат завдань може використовуватися для оцінки різних навичок, таких як розуміння концепцій, лексичних співвідношень, або встановлення відповідностей у різних областях знань.

Завдання на встановлення відповідності широко використовуються в освіті, наукових дослідженнях та психометричних оцінках.

Вони можуть бути корисними для перевірки рівня розуміння конкретної теми чи для оцінки логічного мислення і аналітичних здібностей учасників.

Цей тип завдань відноситься до категорії логічних пар і відомий як формат розширеного вибору.

Завдання такого типу складається з інструкції-завдання та двох колонок. Перша колонка містить перелік початкових умов (слів, фраз, речень, дат, формул, термінів і т. д.), які тестований повинен зіставити з відповідями у другій колонці (праворуч), яку називають списком відповідей.

Тестований повинен утворити правильні логічні пари, порівнюючи матеріал з обох колонок. Інструкція розміщується перед завданням. Існують дві модифікації цих завдань: одна дозволяє обрати лише одну відповідь, інша – декілька.

Завдання можуть бути у формі вибору відповідей або в формі введення відповідей (де у лівій колонці потрібно власноруч ввести відповідь для створення логічної пари). Завдання на встановлення відповідності широко використовуються в освіті, наукових дослідженнях та психометричних оцінках.

Завдання на встановлення відповідності широко використовуються в освіті, наукових дослідженнях та психометричних оцінках. Вони дозволяють систематизувати та оцінювати логічне мислення та здатність встановлювати логічні пари.

Завдання такого типу є важливим інструментом для перевірки рівня розуміння конкретної теми та оцінки логічного мислення і аналітичних здібностей учасників.

Вони можуть бути корисними для перевірки рівня розуміння конкретної теми чи для оцінки логічного мислення і аналітичних здібностей учасників (рис 2.8).

**Directions: Match the class of mail with its primary characteristic.
Write your answer in the blank beside the number.**

Characteristic	Class
_____ 1. Newspapers, magazines, and other periodicals	A. Airmail
_____ 2. All sealed written material	B. First class
_____ 3. Circulars, books, catalogs	C. Second class
_____ 4. All mailable matter weighing 16 ounces or more that is not included in first or second class.	D. Third class
	E. Fourth class (Parcel post)

Рисунок 2.8 – Приклад тесту на встановлення правильної відповідності

Завдання з заповненням пропусків або тест з вставками – це формат тестового завдання, де учасник повинен заповнити пропуски в реченні чи тексті, вставивши відсутні слова чи фрази. Зазвичай ці пропуски позначаються підкресленням або порожніми просторами, і учасник повинен визначити правильні слова або вирази, які краще пасують до контексту [23].

Такі завдання можуть бути використані для оцінки різних мовних навичок, таких як лексика, граматики і розуміння тексту. Вони дозволяють перевірити, наскільки добре учасник розуміє текст і як він може виразити свої думки або відповіді у вигляді словосполучень.

Такий формат досить популярний в навчальних матеріалах, тестах і психометричних оцінках, оскільки він дозволяє оцінювати різні аспекти мовних навичок та розуміння матеріалу.

До переваг відносять простоту розробки, так як реалізація тестів з заповненням пропусків може бути відносно простою з точки зору програмування. Це передбачає створення текстового поля для введення відповіді та механізму для перевірки правильності введеної відповіді.

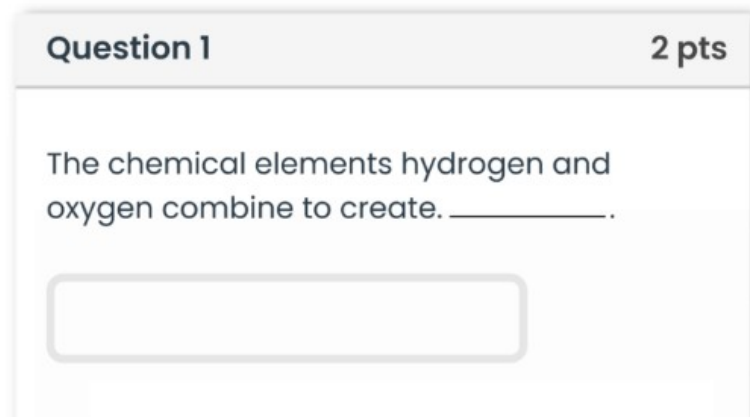
Іншою перевагою виділяють широку застосовуваність. Цей формат може бути використаний для різних предметів та завдань, що робить його універсальним у різних областях.

З недоліків частіш за все вказують варіабельність оцінювання. Оцінювання тестів з заповненням пропусків може бути складним, оскільки

існує можливість декількох правильних відповідей. Розробники повинні враховувати цю варіабельність при розробці системи оцінювання.

Іншим недоліком вказують можливість угадування. Учасники можуть вгадувати правильні відповіді, не завжди розуміючи контекст. Це може змінити характер оцінки та важкість завдань [18].

Загалом, тести з заповненням пропусків можуть бути корисними в ІТ-розробці для створення тестових платформ, але потребують уважного управління варіабельністю відповідей та оцінюванням, щоб бути ефективними та об'єктивними (рис 2.9).



Question 1 2 pts

The chemical elements hydrogen and oxygen combine to create. _____.

Рисунок 2.9 – Приклад тесту з вставками

Тест на правильність – це формат тестового завдання, де учаснику пропонуються різні ствердження, і він повинен визначити, чи вони правдиві (true) чи неправдиві (false). Цей формат також відомий як «true-or-false test» на англійській мові.

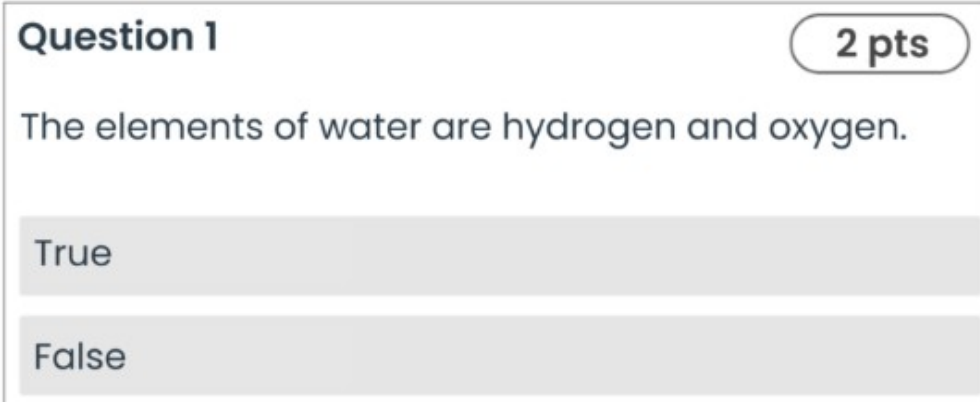
Тести на правильність є досить простими в усіх аспектах – як для учасників, так і для розробників. Учасникам потрібно лише обрати «правда» (true) або «неправда» (false) для кожного ствердження. Розробники можуть легко створити і оцінити такі тести [24].

Цей формат має широкий спектр застосувань і використовується в освіті, психометричних вимірюваннях, тестуванні знань і навичок, а також онлайн-тестах на засвоєння матеріалу.

Оцінка результатів тестів на правильність відбувається дуже швидко і об'єктивно, оскільки кожне ствердження має одну правильну або неправильну відповідь.

Тести на правильність можуть бути створені з різним рівнем складності, починаючи від дуже простих і очевидних стверджень до більш складних і обміркованих.

Однак важливо правильно формулювати ствердження, щоб уникнути двозначних або обманливих варіантів відповідей. Також варто уникати занадто багато однотипних стверджень, щоб не збільшувати вірогідність вгадування (рис 2.10).



Question 1 2 pts

The elements of water are hydrogen and oxygen.

True

False

Рисунок 2.10 – Приклад тесту на правильність

Всі ці правила конструювання тестових завдань мають на меті створити об'єктивний, надійний та адекватний інструмент для вимірювання знань і навичок тестованих осіб, що відображає їхній реальний рівень підготовки і співставляється з метою тестування.

2.3 Аналіз та опрацювання реалізує результатів тестування, методи їх представлення

Аналіз та обробка результатів тестування є кроком, що перетворює накопичену інформацію у корисні знання та висновки, які можуть вплинути на подальші дії та рішення. У цьому контексті, ми розглянемо, як забезпечити якість тестування та його надійність та валідність. Надійність визначається ступенем стійкості результатів тестування при повторних вимірюваннях, тоді як валідність вказує на те, наскільки тест вимірює те, що дійсно має бути виміряно [25].

Важливим аспектом є визначення та розуміння концепції валідності в контексті педагогічних тестів. Термін «валідність» походить від англійського слова «valid», що означає обґрунтований, дійсний, придатний або той, що має силу. Американський психолог Анна Анастасі дала наступне визначення валідності педагогічного тесту: «Валідність тесту – це поняття, яке визначає, що вимірює тест і наскільки якісно це здійснюється». Отже, валідність вказує на відповідність результатів тестування його меті та вимогам оцінювання.

Невалідний тест – це тест, в якому отримані результати не відображають того, що мало б бути виміряно метою тестування. Це може бути результатом декількох факторів, таких як невідповідність між метою тесту та інструментами вимірювання, невідповідність навчальної програми та вмінь, які тест має виміряти, або якість самих тестових завдань.

Щоб забезпечити валідність педагогічного тесту, важливо враховувати комплекс характеристик тесту, процедур тестування та методик оцінювання. Деякі дослідники розглядають валідність з різних точок зору, визначаючи валідність методу та валідність тесту, яка оцінюється за допомогою аналізу якості тестових завдань, процедур тестування та процедур оцінювання. Валідність є комплексною характеристикою, яка враховує як параметри засобу вимірювання, так і властивості досліджуваної ознаки (рис. 2.11).

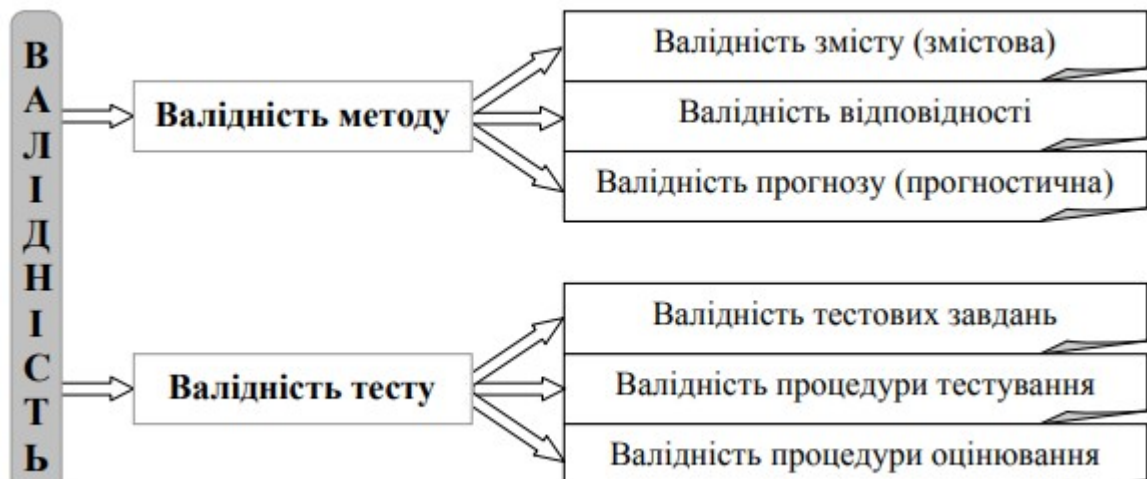


Рисунок 2.11 – Класифікація валідності

І. Є. Булах та М. Р. Мруга зазначають, що валідність визначає, наскільки точно тест вимірює те, що він має виміряти, та наскільки це відповідає його призначенню [26].

Важним етапом у створенні тестувальних завдань є оцінка їх складності. Давайте спершу розглянемо метод визначення потенціалу складності завдань. Сучасна тестологія використовує показник, який називається «потенціал труднощі». Цей показник розраховується як відношення кількості неправильних відповідей на завдання до кількості правильних відповідей, використовуючи наступну формулу.

$$P = q_i / p_i \quad (2.1)$$

де P – потенціал труднощі;

p_i – частка правильних відповідей випробуваного;

q_i – частка неправильних відповідей випробуваного;

Складність завдання тесту визначається на основі статистичного аналізу, який вказує, скільки випробуваних виконали або не виконали певне завдання. Наприклад, якщо тільки 20% випробуваних виконали завдання, то

воно вважається складним для цієї вибірки, а якщо 80% виконали його, то воно вважається легким [27].

Під надійністю тестового завдання розуміють ступінь, з якою можна визначити певну ознаку з точністю. Це вказує, наскільки можна довіряти результатам тестування і наскільки вони стабільні (рис 2.12).



Рисунок 2.12 – Критерії надійності

Оцінка надійності тестового завдання іноді базується на тому, як однаково учасники тестування розподіляються за показниками успішності у різних випробуваннях. Якщо після повторних тестувань учасники виявляються на тих самих місцях за показниками успішності, то такий тест можна вважати надійним.

Надійність тесту може залежати від кількості тестових завдань. Зазвичай, для досягнення достатньої надійності підсумкового контролю, тест має містити не менше 40 завдань [28].

Термін «надійність» може мати два значення, як вказує на це П. Клайн. Внутрішня надійність вказує на узгодженість тестових завдань всередині самого тесту. Надійність вимірює ступінь стабільності результатів тестування при повторному тестуванні через певний час.

Надійність тесту зростає, коли результати учасника під час повторного тестування з використанням того ж самого тесту демонструють високу згоду

або схожість. Цю згоду можна виміряти за допомогою Коефіцієнта надійності Кьюдера-Річардсона.

$$\bar{R}_n = \frac{N}{N-1} \left(1 - \sum_{i=1}^n p_j q_j \right) \quad (2.2)$$

$$q_j > S_x^2$$

де p_j — доля правильних відповідей на j -е завдання;

q_j — доля неправильних відповідей;

S_x^2 — дисперсія по розподіленню балів;

n — число завдань тестування;

Крім надійності, важливо розглядати ефективність тесту, яка дозволяє порівнювати тести за їх якістю та ефективністю. Ефективний тест вимірює знання учасників з меншою кількістю завдань, якісніше і швидше. Також важливо враховувати складність тестових завдань, оскільки вона впливає на їхню ефективність.

Один з останніх кроків у створенні тестів – це аналіз результатів тестування з використанням математичних і статистичних методів. При цьому досліджуються такі показники, як рівень складності завдань, дисперсія, показники різноманітності тестових балів і рівень складності завдань (середньоквадратичне відхилення, коефіцієнт варіації), а також показники спроможності завдань розрізняти між собою відповіді (коефіцієнт дискримінативності, точково-бісеріальний коефіцієнт кореляції, бісеріальний коефіцієнт кореляції).

Коефіцієнт кореляції – це показник, який визначає ступінь взаємозв'язку між двома залежними ознаками (наборами тестових балів або іншими видами оцінювання). Він може мати значення від +1 до -1. Позитивне значення коефіцієнта кореляції свідчить про прямий зв'язок між

ознаками, в той час як негативне значення вказує на зворотний зв'язок. Чим ближче коефіцієнт кореляції до одиниці, тим сильніший цей зв'язок.

Дисперсія – це показник різноманітності кількісних ознак і відображає середній квадрат відхилень окремих значень ознаки (бала кожного тестованого) від її середнього значення (середнього арифметичного бала всіх тестованих) [25].

Середнє значення в цілому генеральної сукупності називається загальною середньою. Ця величина відображає загальні риси всієї сукупності. Середнє значення для окремої вибірки називається вибірковою середньою, і воно відображає загальні риси даної групи.

Існують різні методи обчислення середніх величин, але найбільш поширеним є використання середньої арифметичної характеристики.

$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n} = x_{1 + \frac{x_2 + \dots + x_n}{n}} \quad (2.3)$$

Множина даних це $X = (x_1, x_2, \dots, x_n)$, тоді вибіркоче середнє зазвичай позначається горизонтальною рисою над змінною, як середнє із n значень.

Якщо всі вагові коефіцієнти однакові, то можна використовувати середню гармонічну у такому вигляді:

$$\bar{x}_{\text{гарм}} = \frac{n}{\sum_{i=1}^n \frac{1}{x_i}} \quad (2.4)$$

де n – число індивідуальних значень ознак;

$\sum \frac{1}{X}$ – сума обернених значень ознак;

Для розрахунку середньоквадратичної вагової величини використовується така формула:

$$x_{\text{кв}} = \sqrt{\frac{\sum_{i=1}^n x_i^2 w_i}{\sum_{i=1}^n w_i}} \quad (2.5)$$

Якщо ваговий коефіцієнт $w_i = 1$, для всіх $i = 1, 2, \dots, n$, отримуємо просто середнє квадратичне. Ці величини характеризують «концентрацію» даних вибірки навколо середнього [29].

Для завдань, де використовується лише бінарна оцінка (1 або 0), дисперсія визначається відповідно до такої формули:

$$S_j^2 = p_j * q_j \quad (2.6)$$

В даній формулі p_j , q_j – частки правильних і неправильних відповідей у кожному завданні. Слід пам'ятати, що чим більша дисперсія балів, тим краща диференціувальна здатність завдання.

Для обчислення коефіцієнта кореляції використовують коефіцієнт моментів Пірсона:

$$r_{xy} = \frac{\sum_{i=1}^m (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^m ((x_i - \bar{x})^2) \sum_{i=1}^m (y_i - \bar{y})^2}} = \frac{\text{cov}(x, y)}{\sqrt{S_x^2 S_y^2}} \quad (2.7)$$

де r – коефіцієнт кореляції;

x – результат по кожному завданню;

y – бал (результат) по всьому тесту;

n – кількість попарних добутків;

Σ – знак суми.

Для визначення позиції особи у вибірці стандартизації, початковий результат (кількість правильно виконаних завдань) перетворюється в певну відносну міру. Основна мета цього перетворення полягає в покращенні якості вимірювань і перехід від порядкового шкалювання до інтервальної шкали, яка дозволяє ранжувати осіб, враховуючи їхні різні рівні підготовки, шляхом введення сталої міри для вимірювань [30].

Проте, варто відзначити, що не завжди таке перетворення приводить до інтервальної або хоча б квазіінтервальної шкали. В деяких випадках цей перехід вважається спірним і може бути відхилений під час аргументованої критики на основі експертної оцінки якості шкали і самого методу вимірювань.

Також, сучасні методи тестування включають в себе використання інформаційних технологій та інтерактивних засобів. Це може значно розширити можливості оцінки та зробити процес тестування більш захопливим та зрозумілим для учнів. Застосування комп'ютеризованих тестів дозволяє отримати швидкі та об'єктивні результати, а також забезпечує можливість миттєвого аналізу та зворотного зв'язку для подальшого вдосконалення навчального процесу.

Важливо зауважити, що спроби покращити вимірювання через шкалювання спрямовані на розширення можливостей інтерпретації та порівняння тестових балів, але не завжди призводять до підвищення точності вимірювань. У загальному контексті можна стверджувати, що похідні показники слугують двом цілям: уточненню положення особи в нормативній вибірці та порівнянню результатів осіб, отриманих за допомогою різних тестів [31].

Представити результати тестування краще всього буде у вигляді матриці результатів. Припустимо, що маємо результати тестування групи,

яка складається з n осіб, які відповідали на m різних питань у рамках конкретного тесту. Зазвичай ці дані подаються у вигляді матриці A , де значення представлені в формі «особа – питання» та мають розмірність n на m .

$$A(n \times m) = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{pmatrix} = \{a_{ij} \mid \begin{matrix} j = \overline{1, m} \\ i = \overline{1, n} \end{matrix}\} \quad (2.8)$$

Елемент a_{ij} матриці A – результат виконання j -го завдання для i -го тестованого. А сама матриця буде виглядати наступним чином, та буде представлена у вигляді таблиці (табл. 2.2).

Таблиця 2.2 – Приклад матриці тестових результатів

№	User Name	Test Numbers										Result	
		1	2	3	4	5	6	7	8	9	10		
1	User_1	1	1	1	1	1	1	1	1	1	1	1	09
2	User_2	1	1	1	1	1	1	1	1	1	0	0	08
3	User_3	1	1	1	1	1	1	1	1	0	0	0	07
4	User_4	1	1	1	1	1	1	0	0	0	0	0	06
5	User_5	1	1	1	1	1	0	0	0	0	0	0	05
6	User_6	1	0	0	0	0	0	1	1	1	1	1	05
7	User_7	1	1	1	1	0	0	0	0	0	0	0	04
8	User_8	1	1	1	0	0	0	0	0	0	0	0	03
9	User_9	1	1	0	0	0	0	0	0	0	0	0	02
10	User_10	1	0	0	0	0	0	0	0	0	0	0	01
% test passability		0	1	8	7	6	5	5	4	3	2	0	–

У нижньому рядку таблиці повинна бути вказана сума балів, набраних кожною особою за кожним із завдань. Обернена величина цієї суми використовується як міра складності тестових завдань і один із критеріїв

включення завдань до загального пулу завдань з даної навчальної дисципліни. Для кінцевого відбору завдань (їх включення до остаточного списку завдань, що використовуються для тестування) слід провести статистичний аналіз результатів пробного тестування.

В останньому стовпчику можна побачити результат кожного конкретного юзера, який проходив тестування. У самій же таблиці можна побачити результат окремої особи, що проходила тестування відносно кожного окремого тесту. Під час реалізації такого функціоналу краще звернути увагу на зручну навігацію або виведення самого тестового завдання, для того щоб особа, що проходила тестування могла зробити роботу над помилками [32].

2.4 Розробка теоретичної моделі дослідження та опис логіки її роботи

За результатами проведеного аналізу, після врахування усіх аспектів та етапів проведення тестування можна розробити актуальну модель, яка буде враховувати усі досліджені вимоги та найкращім чином поєднувати у собі усі теоретичні вимоги та переводити їх у практичний простір. Така модель найкращім чином буде відтворюватись у вигляді застосунку, який можна буди використовувати на будь якому десктопному чи мобільному пристрої. В собі він буде містити інтерактивні тести з інформатики, модель для їх додавання та видалення, також фінальну інформацію про проведене тестування, яку можна буде застосовувати як особі, що проходила тестування – з метою отримання оцінки знань, так і особою, що створювала тести – з метою оцінки якості тестів та їх подальшого дослідження та покращення.

При розробці інтерфейсу інтерактивних тестів для мобільних пристроїв, важливо враховувати кілька ключових аспектів з точки зору користувальницького інтерфейсу (UI) та користувацького досвіду (UX). Такий інтерфейс повинен бути не лише зручним, але й ефективним для

навігації та використання користувачами. Одним із перших кроків у розробці є створення чіткої та інформативної назви тесту. Заголовок повинен вказувати на суть тесту та повідомляти користувачам, що вони будуть тестувати. Він є важливим елементом для спрямування уваги та визначення мети тесту [31].

Слід звернути особливу увагу на формулювання питань тесту. Кожне питання має бути чітким і лаконічним, і, можливо, виокремленим за допомогою великого шрифту чи жирного тексту, щоб воно стало помітним для користувачів. Питання тесту повинні бути безпосередньо пов'язані з темою, з якої особа проходить тестування.

Крім того, важливо, щоб варіанти відповідей були легко розрізнятися один від одного. Для цього можна використовувати радіокнопки, чекбокси чи чіткі кнопки для вибору відповідей. Маркувальні пункти, такі як номери чи крапки, допоможуть краще структурувати варіанти відповідей.

Забезпечення легкої навігації між питаннями є важливим завданням. Додавання кнопки «Далі» чи «Вперед» для переходу до наступних запитань сприяє зручності користувачів. Також може бути корисною можливість переходу до попереднього питання за допомогою кнопки «Попереднє», якщо це необхідно. Надати можливість користувачам відкривати список усіх запитань, з можливістю легкого переходу між ними, також поліпшить користувацький досвід [33].

Після вибору усіх відповідей користувачам необхідно надати чіткий зворотній зв'язок щодо правильності їхнього вибору та якості правильних відповідей. Для досягнення цього можна використовувати кольорову індикацію, наприклад, зелений колір для позначення правильних відповідей та червоний для помилкових, що підсилює результати тесту та робить їх більш видимими для користувачів.

По завершенні тесту, необхідно надати користувачам можливість пройти тест знову, можливо з іншими чи схожими питаннями з метою закріплення теми навчання.

Розробка усіх досліджених раніше видів тестів є критичним аспектом в процесі створення інтерфейсу інтерактивних тестів для мобільних пристроїв. Важливо враховувати, що різні види тестів можуть вимагати різних підходів до інтерфейсу та функціональності.

Наприклад, при розробці тестів з одним правильним варіантом відповіді, важливо забезпечити зручність вибору та перевірки правильності відповіді. У такому випадку радіокнопки або перевірка одного варіанту можуть бути оптимальними рішеннями [34].

Для тестів з вибором декількох правильних варіантів відповіді важливо створити інтерфейс, який дозволить користувачам відзначити всі необхідні варіанти та легко їх перевірити. Чекбокси або можливість вибору декількох варіантів допоможуть досягти цієї мети.

З іншого боку, тести з відкритими питаннями можуть вимагати включення полів для введення тексту та забезпечення можливості зручної подачі відповідей. Такий рід тестів вимагає ретельної обробки введеного тексту та можливості перевірки правильності відповідей.

Отже, при розробці інтерфейсу для тестів на мобільних пристроях, враховуючи різні види тестів, важливо дотримуватися індивідуального підходу до кожного типу тесту та забезпечити оптимальні умови для користувачів у відповідності до конкретних завдань та вимог тестування. На останок, важливо переконатися, що інтерфейс адаптується до різних пристроїв, таких як комп'ютери, планшети та смартфони. Забезпечення доступності для різних категорій користувачів, включаючи людей з обмеженими можливостями, також має високий пріоритет у розробці інтерфейсу інтерактивних тестів [35].

У підсумку, процес розробки інфраструктури для інтерактивних тестів, які призначені для використання на мобільних пристроях, вимагає ретельного та завзятого дослідження різних типів тестів, які можуть бути використані. Кожен тип тесту вимагає унікального підходу до інтерфейсу та функціональності, щоб забезпечити зручність та ефективність користування.

Однак, це лише аспект розробки. Для успішної інфраструктури також важливо враховувати специфічні завдання та вимоги самого тестування. Отже, розробка інфраструктури інтерактивних тестів для мобільних пристроїв – це складний процес, що вимагає глибокого розуміння специфіки тестів та вимог користувачів.

3 РЕАЛІЗАЦІЯ, ТЕСТУВАННЯ ТА ОЦІНКА ІНТЕРАКТИВНИХ ТЕСТІВ ДЛЯ МОБІЛЬНИХ ПРИСТРОЇВ З ІНФОРМАТИКИ З ВЛАСНИМ НАБОРОМ ДАНИХ

3.1 Вибір технологій та інструментів розробки інтерактивних тестів для мобільних пристроїв

Перед початком розробки додатку для інтерактивних тестів для мобільних пристроїв необхідно обрати відповідні технології та інструменти, завдяки яким буде здійснена розробка. Це є ключовим завданням, оскільки це визначає технічні можливості, продуктивність та зручність використання створених додатків.

Також необхідно ретельно проаналізувати різні технологічні рішення та інструменти, які можуть бути використані під час розробки інтерактивних тестів для мобільних пристроїв у галузі інформатики. Вибір технологій пов'язаний з рішеннями щодо програмування, архітектури додатків, використання платформ, мов програмування та інших факторів, які впливають на результати розробки. Завданням є знайти оптимальне рішення для створення інтерактивних тестових додатків, які відповідають вимогам та очікуванням нашої аудиторії.

Лише після цього можна буде перейти до практичної реалізації інтерактивних тестів та подальшого їхнього тестування та оцінки, використовуючи обрані технології та інструменти [36].

Враховуючи той факт, що в поставлених вимогах є необхідність того, щоб створені інтерактивні тести можна було запустити на будь-якому мобільному чи планшетному пристрої або комп'ютері, першою технологією яку треба буде розглянути буде HTML.

HTML є основною мовою розмітки для створення структури веб-сторінок. Його основне призначення - визначати різні елементи контенту на

веб-сторінці. За допомогою HTML можна визначити заголовки, параграфи, списки, таблиці, зображення, посилання та інші елементи.

HTML, або HyperText Markup Language, функціонує як основна мова розмітки для визначення структури веб-документів. Кожен HTML-тег визначає різні елементи контенту та їхню взаємодію. Наприклад, теги, такі як `

`, `

`, `

`, встановлюють структуру сторінки, а теги `` та `` відповідають за посилання та вставку зображень.

HTML також пропонує семантику, що полегшує розуміння структури сторінки для браузерів та розробників. Використання таких тегів, як ``, ``, та ``, сприяє створенню чіткої ієрархії та робить код більш зрозумілим. Він не надає дизайну або стилів, але створює базову структуру, яка потрібна для подальшого оформлення за допомогою CSS.

CSS використовується для стилізації та внутрішнього оформлення вебсторінок, надаючи їм зовнішній вигляд. За допомогою CSS можна задавати кольори, шрифти, розташування елементів, відступи, рамки та інші стилізаційні властивості. CSS дозволяє створювати адаптивний дизайн, визначаючи, як виглядатиме вміст на різних пристроях та розмірах екрану. CSS відповідає за стилізацію веб-документів. Визначаючи правила для вигляду та розміщення елементів, CSS дозволяє створювати естетично приємний та адаптивний дизайн. Використання селекторів, властивостей та значень, CSS надає точний контроль над зовнішнім виглядом сторінки.

Разом HTML і CSS взаємодіють для створення ефективних та зручних веб-сторінок. HTML визначає структуру та семантику сторінки, вказуючи, як розміщати різні елементи контенту. CSS відповідає за зовнішній вигляд цих елементів, створюючи гармонійний та естетичний дизайн.

HTML і CSS разом створюють зручний та легко читабельний інтерфейс для користувачів мобільних пристроїв. Вони формують основу для розробки інтерактивних тестів, забезпечуючи структуру, семантику та естетичний вигляд, необхідні для зручного користувацького досвіду.

Використання HTML і CSS для реалізації інтерактивних тестів на мобільних пристроях дозволить створити користувачам зручний та легко читабельний інтерфейс. Оптимізація для різних розмірів екрану та підтримка адаптивного дизайну забезпечать приємний досвід користувача на різних пристроях [37].

JavaScript і TypeScript є двома ключовими мовами програмування, що використовуються в розробці веб-додатків, включаючи інтерактивні тести для мобільних пристроїв.

JavaScript – це високорівнева, інтерпретована мова програмування, яка використовується для надання динаміки та інтерактивності веб-сторінкам. Вона виконується безпосередньо в браузері та надає можливість маніпулювати DOM для зміни вмісту та структури сторінки в реальному часі. JavaScript дозволяє обробляти події, взаємодіяти з користувачем та виконувати асинхронні запити, що робить його основною мовою для клієнтської розробки веб-додатків.

JavaScript визнається за ключову мову програмування у сфері розробки веб-фронтенду, що зумовлено кількома факторами. В першу чергу, широкий розповсюдження JavaScript обумовлено підтримкою майже всіма сучасними браузерами, роблячи її універсальним інструментом для веб-розробки.

Одним із ключових аспектів JavaScript є його динамічність та здатність надавати інтерактивність веб-сторінкам. Мова дозволяє розробникам здійснювати маніпуляції з DOM змінювати вміст сторінки в реальному часі, ініціювати асинхронні запити на сервер та взаємодіяти з подіями користувача.

JavaScript є багатофункціональною мовою, підтримуючи асинхронне програмування, обробку винятків, функціональні та об'єктно-орієнтовані парадигми. Це надає розробникам можливість писати ефективний та зручний код для веб-додатків.

Крім того, JavaScript користується великою популярністю завдяки розширеній екосистемі бібліотек та фреймворків, які спрощують та

прискорюють процес розробки. Використання таких інструментів, як React, Angular чи Vue.js, дозволяє розробникам швидше створювати веб-додатки та взаємодіяти з DOM більш ефективно.

Узагальнюючи, JavaScript володіє рядом переваг, що роблять його ідеальним інструментом для розробки динамічних та інтерактивних веб-додатків. Його широкі можливості, доступність та багата екосистема роблять його важливим елементом сучасного веб-розробника.

TypeScript є надмножиною JavaScript, яка додає статичну типізацію та інші функції до мови. Використання TypeScript дозволяє розробникам визначати типи даних для змінних, аргументів та значень повернення, що допомагає виявляти та виправляти помилки під час розробки. TypeScript також підтримує об'єктно-орієнтоване програмування, забезпечуючи зручний інструментарій для створення складних веб-додатків.

В першу чергу, TypeScript надає статичну типізацію, що робить розробку більш передбачуваною та полегшує виявлення помилок ще на етапі компіляції коду. Це особливо важливо для великих проєктів, де чітка структура даних може вирішити багато проблем [38].

До того ж, TypeScript підтримує інші концепції, такі як об'єктно-орієнтоване та функціональне програмування, що дозволяє розробникам використовувати різні парадигми для реалізації складної логіки.

Ще однією перевагою TypeScript є його сумісність з JavaScript, що дозволяє поступово переходити від однієї мови до іншої та використовувати існуючий JavaScript-код без необхідності повного переписування.

Узагальнюючи, TypeScript забезпечує розробникам більшу стабільність та контроль над кодом у порівнянні з JavaScript. Використання статичної типізації та інших функцій TypeScript дозволяє підвищити якість коду та полегшити його обслуговування, особливо в великих та розподілених проєктах.

Використання JavaScript і TypeScript у контексті інтерактивних тестів дозволяє створювати складну логіку взаємодії та перевірки відповідей

користувачів. JavaScript взаємодіє з елементами DOM, забезпечуючи динаміку та зручний користувацький досвід, тоді як TypeScript додає елементи статичної типізації, полегшуючи розробку та підтримку коду. JavaScript і TypeScript є невід'ємною частиною розробки веб-додатків, забезпечуючи не тільки інтерактивність, але й зручну та ефективну розробку складних функціональностей, які важливі для інтерактивних тестів на мобільних пристроях.

Також більш сучасним та кращим рішенням буде використовувати якийсь фреймворк чи бібліотеку JavaScript для розробки інтерфейсів. React – це бібліотека для розробки інтерфейсів, яка визначається своєю декларативністю та компонентною архітектурою. Обрання React для розробки вебфронтенду обумовлено кількома ключовими аспектами.

В першу чергу, React дозволяє створювати користувацькі інтерфейси за допомогою компонентів, які є незалежними та багаторазовими блоками. Це спрощує розробку та управління кодом, особливо в разі великих проєктів.

Декларативний підхід React дозволяє описувати, як виглядає інтерфейс у різних станах, і бібліотека відповідає за ефективне оновлення DOM для відображення цих змін. Це призводить до оптимізованої продуктивності та високої швидкодії.

Ще однією перевагою React є його відкритість до інших технологій та зручна інтеграція з різними інструментами та бібліотеками. React може використовуватися разом із TypeScript, щоб забезпечити статичну типізацію та більшу безпеку.

Багата екосистема та активна спільнота, що оточує React, забезпечують наявність безлічі готових рішень, компонентів та інструментів. Це спрощує розробку та забезпечує високий рівень підтримки.

React вирізняється своєю простотою використання, ефективністю та гнучкістю. Обираючи React, розробники отримують не тільки потужний інструмент для створення користувацьких інтерфейсів, але і різноманітні можливості для вдосконалення та оптимізації процесу розробки.

Для створення стилізованих компонентів гарним рішенням буде використовувати Material-UI – це популярна бібліотека компонентів інтерфейсу користувача для React, яка використовує дизайн-мову Material Design від Google. Вибір Material-UI для створення користувацького інтерфейсу веб-додатка на React обумовлюється декількома ключовими факторами.

По-перше, Material-UI надає великий набір готових до використання компонентів, які відповідають стандартам Material Design. Це спрощує розробку та стилізацію інтерфейсу, роблячи його сучасним та зручним для користувачів.

Material-UI відзначається гнучкістю та можливістю налаштування стилів компонентів відповідно до потреб проєкту. Відкритий код та активна спільнота роблять його легким у використанні та адаптації під конкретні вимоги. Ця бібліотека також пропонує компоненти, які легко інтегруються з іншими бібліотеками та фреймворками, забезпечуючи єдність дизайну у всьому додатку.

Material-UI підтримує реактивний підхід та реагує на зміни в стані додатка, що дозволяє створювати інтерактивні та динамічні інтерфейси без зайвого зусилля.

Після написання клієнтської частини застосунку треба буде перейти до вибору технологій та інструментів для серверної частини та способу їх з'їдання. Тут може допомогти REST API, який є стандартом архітектури для розробки мережевих застосунків та служб. Вибір REST API для взаємодії з сервером у фронтенді визначається кількома ключовими характеристиками.

В першу чергу, REST API сприяє створенню ефективного та розширюваного інтерфейсу для взаємодії між клієнтом та сервером. Він використовує HTTP протокол для передачі даних та взаємодії за стандартними CRUD операціями, що робить його простим та зрозумілим для розробників.

REST API спирається на концепцію ресурсів, які ідентифікуються унікальними URI. Це робить API гнучким та дозволяє легко масштабувати систему, додаючи нові ресурси чи змінюючи існуючі.

Ще однією перевагою REST API є його незалежність від платформи та мови програмування. Будучи основаним на стандартних HTTP методах, він може використовуватися з будь-якими мовами програмування та на будь-яких платформах.

REST API також підтримує принципи стандарту «stateless», що означає відсутність зберігання стану на сервері між запитами від клієнта. Це спрощує управління станом та робить API більш надійним та легким для розуміння. Він є потужним засобом для взаємодії між клієнтом та сервером в веб-фронтенді. Використання стандартів HTTP, гнучкість у визначенні ресурсів та платформенна незалежність роблять його ефективним та універсальним засобом для розробки веб-застосунків.

Для розробки серверної частини застосунку кращим вибором буде використовувати Node.js. Це середовище виконання JavaScript, яке базується на двигуні V8 від Google та призначене для розробки серверних застосунків. Обираючи Node.js для створення серверної частини веб-додатків, розробники керуються кількома ключовими факторами.

По-перше, Node.js відзначається високою швидкістю завдяки використанню двигуна V8. Він розроблений для асинхронного вводу-виводу, що робить його ідеальним вибором для високонавантажених серверів та додатків, які вимагають обробки багатьох одночасних запитів.

Node.js базується на подійному циклі, що дозволяє реалізовувати асинхронну обробку подій. Це дозволяє створювати ефективні та швидкодіючі сервери, а також забезпечує можливість побудови масштабованих застосунків.

Node.js використовує мову JavaScript, що дозволяє розробникам використовувати одну мову програмування для як клієнтської, так і серверної частини. Це спрощує розробку та управління кодовою базою.

Node.js має широкий вибір модулів та пакетів у публічних репозиторіях, таких як npm, що полегшує використання сторонніх бібліотек та інструментів. Node.js став популярним серед розробників завдяки своїй ефективності, швидкодії та асинхронності. Обираючи Node.js, розробники отримують інструмент для створення високопродуктивних серверів та серверних застосунків, який відмінно взаємодіє з клієнтською частиною на основі JavaScript.

Як і в випадку з TypeScript та React, найкращим вибором з Node.js буде використовувати фреймворк. Це не тільки покращить кодову базу, але і зробить застосунок більш надійним та сучасним.

Express – це мінімалістичний та гнучкий веб-фреймворк для Node.js, який дозволяє швидко розробляти серверні додатки та RESTful API. Вибір Express для створення серверної частини веб-додатків зумовлений декількома ключовими перевагами.

В першу чергу, Express надає простий та інтуїтивно зрозумілий інтерфейс для обробки HTTP-запитів та відповідей. Він спрощує рутинні завдання, такі як маршрутизація, обробка параметрів запиту, та робить розробку серверної логіки більш простою.

Express дозволяє швидко створювати middleware – функції, які обробляють запити до сервера. Це дозволяє реалізувати різноманітні функціональності, такі як автентифікація, логування та обробка помилок.

Важливою перевагою Express є його гнучкість та можливість розширення. Він не нав'язує жодних обмежень, що дозволяє розробникам використовувати інші бібліотеки та модулі в залежності від потреб проєкту.

Express підтримує широкий спектр перехоплювачів подій, які спрощують роботу з HTTP-запитами та відповідями. Це включає в себе можливість обробки GET та POST запитів, роботу з параметрами шляху, а також обробку статичних файлів.

Загалом, Express є ідеальним вибором для розробників, які шукають простий та ефективний інструмент для розробки серверної частини веб-

додатків на платформі Node.js. Його легкість використання, гнучкість та активна спільнота роблять його популярним вибором в середовищі Node.js.

Але під час роботи з серверною частиною у нас у будь якому випадку виникне необхідність десь зберігати дані, отримані для клієнта. І тут найкращим вибором буде використовувати одну із сучасним систем управління базою даних.

PostgreSQL – це об’єктно-реляційна система управління базами даних (СУБД), яка вирізняється своєю надійністю, розширюваністю та високою рівнем функціональності. Обираючи PostgreSQL для зберігання та управління даними, розробники враховують кілька ключових переваг.

В першу чергу, PostgreSQL відзначається великою надійністю та стабільністю. Він має розширені механізми виявлення та виправлення помилок, а також підтримує транзакції, що робить його ідеальним для використання в критичних застосунках.

PostgreSQL підтримує об’єктно-реляційну модель даних, що дозволяє створювати складні та зв’язані структури даних. Це робить PostgreSQL потужним інструментом для зберігання структурованих даних у великих та складних проєктах.

Ще однією вагомою перевагою PostgreSQL є його розширюваність та можливість розподіленого зберігання даних. Він підтримує реплікацію та розподілені операції, що дозволяє розширювати обсяги даних та оброблювати велику кількість одночасних підключень.

PostgreSQL відкритий для розширень та розширень сторонніх розробників. Він підтримує велику кількість доповнень, які розширюють його функціонал та надають нові можливості, такі як географічна інформаційна система (GIS) чи повнотекстовий пошук.

Загалом, PostgreSQL є потужним інструментом для зберігання та управління базами даних. Відкритість, розширюваність та надійність роблять його привабливим вибором для проєктів різних розмірів та складності.

Після ретельного аналізу технологій та інструментів для розробки інтерактивних тестів для мобільних пристроїв були визначені оптимальні варіанти. HTML обрано в якості основної мови розмітки, щоб надати чітку структуру контенту. CSS використовується для стилізації та внутрішнього оформлення, забезпечуючи зручний та привабливий дизайн. Також в процесі вибору враховано важливість JavaScript та TypeScript для динамічної та інтерактивної логіки додатка. React використовується як потужний інструмент для створення компонентного інтерфейсу, забезпечуючи гнучку архітектуру та зручне управління станом. Для забезпечення ефективної взаємодії між клієнтом та сервером обрано REST API. Node.js та Express використовуються для реалізації серверної частини, забезпечуючи швидкість та простоту розробки. Для зберігання та управління даними використовується PostgreSQL, що гарантує стабільність та розширюваність системи управління базами даних. Усі ці компоненти об'єднуються в єдиний стек технологій, щоб забезпечити ефективний та сучасний розвиток інтерактивних тестів для мобільних пристроїв.

3.2 Налаштування програмного середовища для дослідження та розробки інтерактивних тестів

Для ефективного проведення дослідження та розробки інтерактивних тестів необхідно встановити та налаштувати оптимальне програмне середовище. Це середовище базується на використанні ряду ключових інструментів та технологій, які були обрані відповідно до вимог та завдань проекту.

При розробці використовується інтегроване середовище розробки Visual Studio Code. Це потужний та легкий редактор коду, який надає розширені можливості роботи з кодом та інтеграції з іншими інструментами. VS Code підтримує різноманітні розширення, що дозволяє розробникам

налаштовувати середовище під свої потреби та використовувати різні мови програмування.

Для ефективного керування версіями коду використовується система контролю версій Git. Git дозволяє ефективно відстежувати та управляти змінами в кодовій базі проекту, а також надає можливість спільно працювати над проектом кільком членам команди. Використання Git дозволяє зберігати резервні копії коду та контролювати його історію.

Додатково, встановлення і використання npm (Node Package Manager) є необхідним для управління залежностями та пакетами проекту. npm є пакетним менеджером для JavaScript, який дозволяє легко встановлювати та оновлювати бібліотеки та інші інструменти, необхідні для розробки. Використання npm забезпечує легкість розгортання проекту та управління його залежностями. Для початку роботи з розробкою інтерактивних тестів з використанням React, TypeScript та Material-UI, необхідно провести ініціалізацію середовища, що включає в себе встановлення та налаштування необхідних інструментів. Далі, для ініціалізації проекту з використанням React, TypeScript та Material-UI, використовується команда `npm create-react-app` з параметром --template typescript`. Це автоматично створює необхідну структуру проекту та налаштовує TypeScript.`

Для початку розробки серверної частини проекту на Node.js та Express, першим кроком є встановлення Node.js. Це можна здійснити завантаживши виконуючий файл (executable) з офіційного веб-сайту Node.js та виконавши процедуру інсталяції. Встановлення Node.js надасть можливість використовувати JavaScript для серверної розробки [39].

Після встановлення Node.js, наступним етапом є ініціалізація проекту з використанням Express.js. Цей процес включає створення нового проекту та встановлення необхідних залежностей за допомогою пакетного менеджера npm. При цьому визначається структура проекту та конфігураційні файли, необхідні для роботи з Express.

Після цього можна почати визначати маршрути (routes) для обробки HTTP-запитів, налаштовувати параметри сервера та впроваджувати необхідні функціональності, що дозволяє створювати веб-додатки з використанням Express.js.

Для ініціалізації бази даних у розробці інтерактивних тестів на Node.js, використовуючи систему керування базами даних PostgreSQL та пакет pg-promise, необхідно виконати кілька ключових кроків.

По перше, почати треба з ініціалізації PostgreSQL. Цей процес включає завантаження виконуючого файлу PostgreSQL з офіційного веб-сайту та подальшу інсталяцію відповідно до інструкцій. Після цього необхідно налаштувати пакет pg-promise для взаємодії з PostgreSQL. Це виконується за допомогою npm, пакетного менеджера для JavaScript. Під час інсталяції pg-promise необхідно вказати необхідні параметри для з'єднання з базою даних.

З цими етапами, база даних буде ініціалізована, і можна перейти до визначення моделей, виконання запитів та налагодження взаємодії серверної частини з базою даних, використовуючи pg-promise.

Під час розробки інтерактивних тестів для мобільних пристроїв, створення та управління користувачами та їх тестовими результатами вимагає дбайливого проектування бази даних. Основні сутності в системі для розробки інтерактивних тестів для мобільних пристроїв включають користувачів та результати тестувань. Давайте детальніше розглянемо обидві ці сутності.

За допомогою таблиць «users» та «testing_results», система забезпечить ефективне зберігання та взаємодію з інформацією про користувачів та їхні тестові результати, що є критичним для надання якісних інтерактивних тестів на мобільних пристроях.

Користувачі в системі є ключовою сутністю. Кожен користувач має унікальний ідентифікатор «id», що використовується для однозначної ідентифікації. Інформація про користувача включає його особисті дані, такі

як ім'я «name», прізвище «surname», електронна пошта «email», а також захешований пароль «password».

Для управління правами користувачів використовується колонка «role», де значення 1 відповідає ролі «користувач», а значення 2 - «адміністратор» (рис. 3.1).

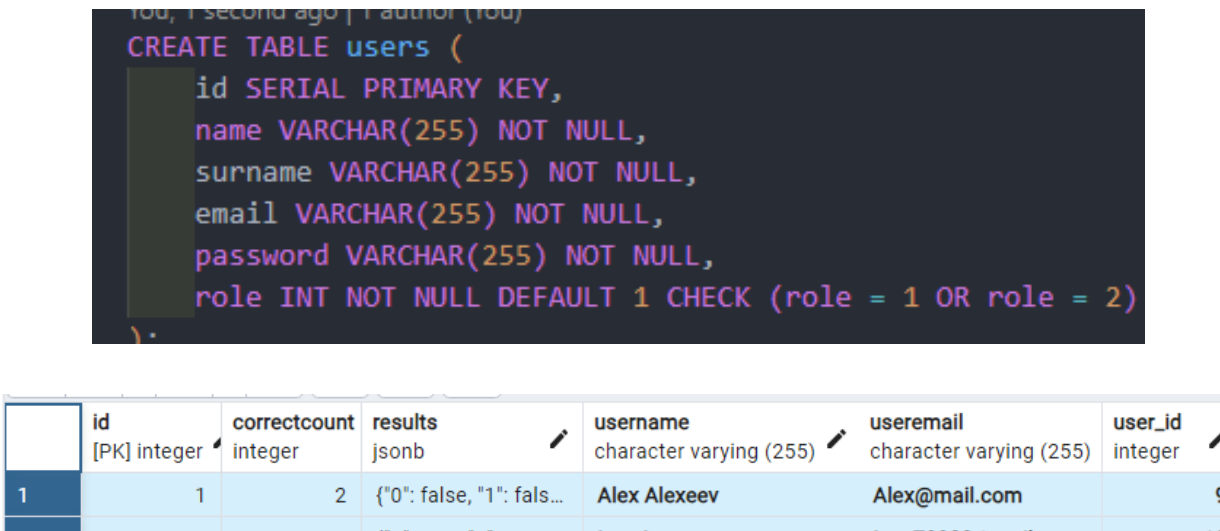
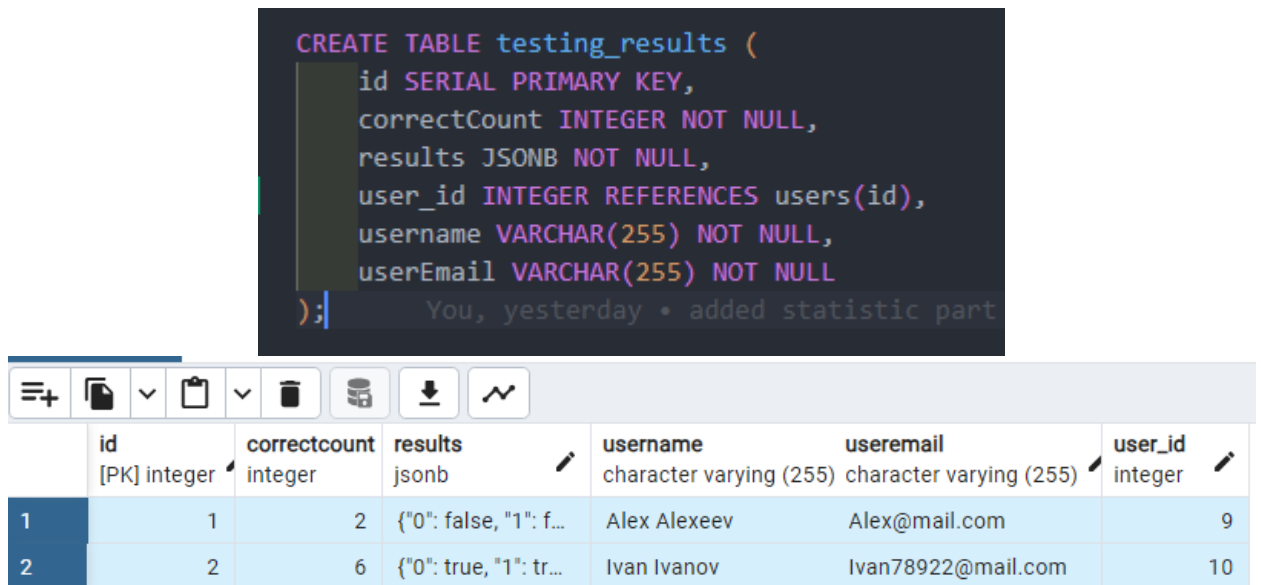


Рисунок 3.1 – Схема створення таблиці users та її вигляд

Результати тестування зберігають інформацію про тести, які користувачі проходять. Кожен результат має унікальний ідентифікатор «id». «correctCount» представляє кількість правильних відповідей у тесті, а «results» зберігає структуру результатів у форматі JSONB. Щоб зв'язати результати тестування з конкретним користувачем, використовується зовнішній ключ «user_id», який посилається на ідентифікатор користувача в таблиці «users».

Крім того, інформація про користувача, така як ім'я «username» та електронна пошта «user_email», також зберігається (рис. 3.2).



```

CREATE TABLE testing_results (
  id SERIAL PRIMARY KEY,
  correctCount INTEGER NOT NULL,
  results JSONB NOT NULL,
  user_id INTEGER REFERENCES users(id),
  username VARCHAR(255) NOT NULL,
  userEmail VARCHAR(255) NOT NULL
);

```

	id [PK] integer	correctcount integer	results jsonb	username character varying (255)	useremail character varying (255)	user_id integer
1	1	2	{"0": false, "1": f...	Alex Alexeev	Alex@mail.com	9
2	2	6	{"0": true, "1": tr...	Ivan Ivanov	Ivan78922@mail.com	10

Рисунок 3.2 – Схема створення таблиці testing_results та її вигляд

Ці основні сутності надають основу для зберігання інформації про користувачів та їх тестові результати в системі. Данні таблиці створені з урахуванням потреб розробки інтерактивних тестів для мобільних пристроїв, надаючи надійне та ефективне зберігання інформації про користувачів та їх тестові результати.

Після опису основних таблиць користувачів треба розглянути основні таблиці тестів, які також є важливою частиною розробки інтерактивних тестів для мобільних пристроїв. Вони надають візуальне представлення структури тестів та їх логіки, допомагаючи розробникам краще розуміти та вдосконалювати тестові сценарії. Перш ніж розглядати конкретну блок-схему, треба ввести загальний огляд цього етапу розробки. Блок-схеми тестів допомагають відобразити послідовність дій, що приймаються системою тестування, а також умови та рішення, що призводять до відповідних відповідей користувачів. Це важливий етап проектування, що допомагає чітко розуміти та відобразити логіку тестових сценаріїв перед їх реалізацією у вигляді таблиць.

Таблиця «single_answer_tests» призначена для зберігання інформації про тести з одним правильним варіантом відповіді в нашій системі

інтерактивних тестів з інформатики для мобільних пристроїв. Кожен тест отримує унікальний ідентифікатор «id», який автоматично збільшується з кожним новим записом у таблиці. Стовпець «question» визначений як «ТЕХТ» та слугує для зберігання текстового опису питання тесту. «correct_answer» – це числовий індекс, який вказує на правильний варіант відповіді серед варіантів, що зберігаються в полі «options». «options» використовується для зберігання масиву текстових варіантів відповідей для кожного тесту.

Ця структура таблиці дозволяє ефективно зберігати та отримувати інформацію про тести з одним правильним варіантом відповіді, включаючи питання, варіанти відповідей та правильний варіант.

```
CREATE TABLE single_answer_tests
  id SERIAL PRIMARY KEY,
  question TEXT NOT NULL,
  correct_answer INT NOT NULL,
  options TEXT[] NOT NULL
```

id	question	correct_answ	options
[PK] integer	text	integer	text[]

Рисунок 3.3 – Схема створення таблиці single_answer_tests та її вигляд

Таблиця «multiple_choice_tests» призначена для зберігання інформації про тести з кількома варіантами правильних відповідей в системі інтерактивних тестів з інформатики для мобільних пристроїв. Кожен тест отримує унікальний ідентифікатор «id», який автоматично збільшується з кожним новим записом у таблиці. Стовпець «question» визначений як «ТЕХТ» та використовується для зберігання текстового опису питання тесту.

Стовпець «correct_answers» – це масив значень типу «BOOLEAN», що визначає правильні відповіді для кожного тесту. Кілька значень «TRUE» в цьому масиві можливі, щоб враховувати ситуації, коли є декілька правильних варіантів. Стовпець «options» використовується для зберігання масиву текстових варіантів відповідей для кожного тесту. Ця структура таблиці дозволяє ефективно зберігати та отримувати інформацію про тести з кількома варіантами правильних відповідей, забезпечуючи гнучкість визначення правильних відповідей.

```
CREATE TABLE multiple_choice_tests (
  id SERIAL PRIMARY KEY,
  question TEXT NOT NULL,
  correct_answers BOOLEAN[] NOT NULL,
  options TEXT[] NOT NULL
```

id	question	correct_answers	options
[PK] integer	text	boolean[]	text[]

Рисунок 3.4 – Схема створення таблиці multiple_answer_tests та її вигляд

Таблиця «true_false_tests» визначена для зберігання інформації про тести з варіантами правильно/неправильно в нашій системі інтерактивних тестів з інформатики для мобільних пристроїв. Кожен тест отримує унікальний ідентифікатор «id», який автоматично збільшується з кожним новим записом у таблиці. Стовпець «question» визначений як «TEXT» та використовується для зберігання текстового опису питання тесту.

Стовпець «correct_answer» це значення типу «BOOLEAN», яке вказує на правильність відповіді для кожного тесту з точки зору правда/неправда.

Ця структура таблиці дозволяє ефективно зберігати та отримувати інформацію про тести з обмеженим набором відповідей, де користувач може вибрати лише між «Правда» та «Неправда».

```
CREATE TABLE true_false_tests (
  id SERIAL PRIMARY KEY,
  question TEXT NOT NULL,
  correct_answer BOOLEAN NOT NULL
);
```

	id [PK] integer	question text	correct_answer boolean

Рисунок

3.5 – Схема створення таблиці true_false_tests та її вигляд

Таблиця «fill_in_the_blanks_tests» служить для зберігання інформації про тести, в яких користувачі повинні заповнити пропуск у реченні. Кожен тест отримує унікальний ідентифікатор «id», який автоматично збільшується з кожним новим записом у таблиці.

Стовпець «question» визначений як «ТЕХТ» та використовується для зберігання текстового опису питання тесту.

Стовпець «missing_word» – це текстове поле, в якому зберігається слово чи фраза, яку користувач повинен вставити у пропуск.

```
CREATE TABLE fill_in_the_blanks_tests (
  id SERIAL PRIMARY KEY,
  question TEXT NOT NULL,
  missing_word TEXT NOT NULL
);
```

	id [PK] integer	question text	missing_word text
1	1	спосіб вивчення глибинних...	тест

Рисунок 3.6 – Схема створення таблиці fill_in_the_blanks_tests та її вигляд

Таблиця «sequencing_tests» служить для зберігання інформації про тести, в яких користувачам потрібно встановити послідовність відповідей у правильному порядку. Кожен тест має унікальний ідентифікатор «id», який автоматично збільшується з кожним новим записом у таблиці. Стовпці «question_order» та «question_text» дозволяють ефективно зберігати та отримувати інформацію про тести з визначеним порядком. «question_order», масив цілих чисел, який визначає порядок питань у тесті. «question_text», масив текстових описів питань. Користувачі повинні встановити відповіді у порядку, вказаному у «question_order».

```
CREATE TABLE sequencing_tests (
  id SERIAL PRIMARY KEY,
  question_order INT[] NOT NULL,
  question_text TEXT[] NOT NULL
```

id	question_order	question_text
[PK] integer	integer[]	text[]

Рисунок 3.7 – Схема створення таблиці sequencing_tests та її вигляд

Таблиця «matching_tests» призначена для зберігання інформації про тести, де користувачам потрібно встановити відповідність між питаннями та відповідями. Кожен тест має унікальний ідентифікатор «id», який автоматично збільшується з кожним новим записом у таблиці.

Стовпці «questions» та «answers» представляють собою масиви текстових описів питань та відповідей відповідно. Стовпець «matching» використовується для зберігання інформації про відповідність між питаннями та відповідями у форматі JSONB. Ця структура таблиці дозволяє ефективно зберігати та отримувати інформацію про тести, де користувачам потрібно правильно встановити відповідності.

```
CREATE TABLE matching_tests
  id SERIAL PRIMARY KEY,
  questions TEXT[] NOT NULL,
  answers TEXT[] NOT NULL,
  matching JSONB NOT NULL
```

id	questions	answers	matching
[PK] integer	text[]	text[]	jsonb

Рисунок 3.8 – Схема створення таблиці `matching_tests` та її вигляд

В результаті роботи було визначено та детально описано структуру таблиць, які використовуватимуться для зберігання інформації про різновиди тестів в системі інтерактивних тестів для мобільних пристроїв. Кожна таблиця спроектована для конкретного типу тесту, забезпечуючи ефективне зберігання та отримання інформації, необхідної для проведення тестування. Використання реляційної бази даних дозволяє нам структурувати дані таким чином, щоб легко взаємодіяти з ними та отримувати необхідну інформацію для подальших етапів розробки системи.

3.3 Створення набору тестових даних для проведення дослідження

Перед початком розробки інтерактивних тестів, важливо створити набір тестових даних, який відображатиме різноманітні сценарії та варіанти використання системи. Створення адекватного та репрезентативного набору тестових даних є ключовим етапом перед переходом до етапу тестування розроблених інтерактивних тестів. Цей набір даних повинен охоплювати різноманітні сценарії використання, різні типи тестів та варіації відповідей. Відправною точкою є створення тестових записів, які відображають реальні

умови використання системи, щоб забезпечити достовірність та реалістичність проведеного дослідження.

Докладний аналіз функціоналу і особливостей кожного типу тесту дозволить створити варіативний та вичерпний набір даних. З урахуванням різних категорій користувачів, рівнів важкості питань та різноманітних сценаріїв відповідей, можна забезпечити належне охоплення всієї функціональності системи.

Усі створені тести до усіх таблиць буде скомпоновано та додано до таблиці у вигляді набору тестових даних, що використовувався при створенні тестів (додаток А).

Створення тестових даних для таблиці «single_answer_tests» передбачає додавання записів, що відображають окремі тести з одним правильним варіантом відповіді. Важливо визначити поля, які мають бути заповнені для кожного тесту.

Для кожного тесту необхідно обрати тему або категорію, в даному випадку це будуть оператори, умовні конструкції, цикли.

Для початку треба сформулювати питання тесту, наприклад, «Яким буде результат виразу $3 + 4 * 2$?» або «Як правильно визначити умову в мові програмування JavaScript?»

Далі для кожного тесту потрібно визначити варіанти відповідей. У цьому випадку вони є числовими індексами або іншими ідентифікаторами, які позначають правильний варіант. Наприклад, для тесту з операціями можна визначити варіанти 1, 2, 3, 4, де 2 є правильною відповіддю.

Далі кожен створений тест треба додати до таблиці «single_answer_tests». Кожен запис повинен містити питання, варіанти відповідей і правильний варіант.

Цей процес можна автоматизувати, наприклад, використовуючи скрипти програмування або створений додатковий інтерфейс для додавання тестів у систему.

Лістинг 3.1 Приклад SQL-запиту для додавання тесту у таблицю «single_answer_tests»:

```
INSERT INTO single_answer_tests (question, correct_answer, options)
VALUES ('Яким буде результатом виразу 3 + 4 * 2?', 2, '{"10", "17", "14", "11"}');
```

У цьому запиті «correct_answer» вказує на правильний варіант, а «options» є масивом варіантів відповідей.

Створення тестових даних для таблиці «multiple_choice_tests» включає додавання записів, які представляють тести з кількома правильними варіантами відповідей. Цей процес можна розглядати як послідовність кроків.

Кожний тест вибирає тему або категорію, після чого завданням є створення питання тесту, наприклад, «Як правильно оголосити змінну в мові програмування Python?».

Далі для кожного тесту визначаються варіанти відповідей, які можуть бути числовими індексами або іншими ідентифікаторами, позначаючи правильні відповіді. Наприклад, для тесту про оголошення змінної в Python можна вказати варіанти 1, 2, 3, 4, де правильними є 2 та 3.

Кожен створений тест потім додається до таблиці «multiple_choice_tests». Кожен запис включає питання, варіанти відповідей та правильні варіанти.

Лістинг 3.2 Приклад SQL-запиту для додавання тесту у таблицю «smultiple_choice_tests»:

```
INSERT INTO multiple_choice_tests (question, correct_answers, options)
VALUES ('Як правильно оголосити змінну в мові програмування Python?', '{true, false, true, false}', '{"Variable x = 5;", "var x = 5;", "x = 5;", "int x;"}');
```

Створення тестових даних для таблиці «true_false_tests» передбачає додавання записів, які представляють тести з вибором між двома можливими відповідями: правильно чи неправильно. Цей процес можна описати в такому контексті:

Для кожного тесту обирається тема або категорія, наприклад, «HTML» або «цикли». Наступним етапом є формулювання питання тесту, наприклад, «HTML - це скорочення від HyperText Markup Language?».

Таке питання буде відповідати поставленій задачі та буде відповідати загальній темі тестування.

Потім для кожного тесту визначається правильна відповідь, яка може бути true або false. Наприклад, для тесту про HTML правильною відповіддю буде true.

Лістинг 3.3 Приклад SQL-запиту для додавання тесту у таблицю «true_false_tests»:

SQL-запит для додавання тесту у таблицю «true_false_tests» може виглядати наступним чином:

```
INSERT INTO true_false_tests (question, correct_answer)
VALUES ('HTML – це скорочення від HyperText Markup Language?',
true);
```

Створення тестових даних для таблиці «fill_in_the_blanks_tests» передбачає додавання записів, кожен з яких містить у собі конкретне питання з пропущеним словом та правильною відповіддю для цього пропущеного слова.

Як приклад, можна сформулювати питання типу «це галузь штучного інтелекту, яка дозволяє комп'ютерам набувати здатності самостійно навчатись.», де пропущеним словом буде «Машинне навчання»

Лістинг 3.4 Приклад SQL-запиту для додавання тесту у таблицю «fill_in_the_blanks_tests»:

```
INSERT INTO fill_in_the_blanks_tests (question, missing_word)
VALUES ('Це галузь штучного інтелекту, яка дозволяє комп'ютерам
набувати здатності самостійно навчатись.', 'Машинне навчання');
```

Такий підхід застосовується до кожного запису у таблиці, де для кожного питання визначено пропущене слово.

Створення тестових даних для таблиці «sequencing_tests» передбачає додавання записів, кожен з яких містить у собі конкретне тестове завдання з визначеною послідовністю питань та текстовим описом кожного питання. Як приклад, можна сформулювати тестове завдання на тему «Машинне навчання» із трьома питаннями. Розташуйте етапи розв'язання задачі в машинному навчанні в правильному порядку: збір та підготовка даних, вибір моделі, тренування моделі.

Лістинг 3.5 Приклад SQL-запиту для додавання тесту у таблицю «sequencing_tests»:

```
INSERT INTO sequencing_tests (question_order, question_text)
VALUES ('{1, 2, 3}', '{"Розташуйте етапи розв'язання задачі в
машинному навчанні в правильному порядку: збір та підготовка даних, вибір
моделі, тренування моделі."}');
```

Такий підхід застосовується до кожного запису у таблиці, де для кожного тесту визначена послідовність питань та текстовий опис кожного питання.

Створення тестових даних для таблиці «matching_tests» передбачає додавання записів, кожен з яких містить у собі конкретне тестове завдання з визначеними питаннями, можливими відповідями та відповідністю між ними.

Для кожного тесту визначається послідовність питань, можливі відповіді та відповідність між ними. Тестове завдання має питання «Виберіть мову програмування: Python, Java, C++»

та варіанти відповідей «Виберіть галузь розробки: Web development, Mobile app development, Data science». Користувач повинен відповісти на кожне питання, вибравши відповідний варіант з кожної групи. Правильні відповіді на цей тест Python – Data science, Java – Web development, C++ – Mobile app development.

У тестових даних визначено правильні відповіді через JSON-об'єкт, де ключі вказують на порядковий номер питання, а значення представляють правильні варіанти відповідей. Наприклад, відповідь «Python» відповідає галузі «Data science» (ключ «1»).

Такий формат дозволяє системі перевіряти правильність вибраних користувачем відповідей під час тестування.

Лістинг 3.6 Приклад SQL-запиту для додавання тесту у таблицю «matching_tests»:

```
INSERT INTO matching_tests (questions, answers, matching)
VALUES ('{"Python", "Java", "C++"}', '{"Web development", "Mobile app
development", "Data science"}', '{"1": "Data science", "2": "Web development",
"3": "Mobile app development"}');
```

Такий підхід застосовується до кожного запису у таблиці, де для кожного тесту визначена послідовність питань, можливі відповіді та відповідність між ними.

Використовуючи наведені приклади, можна ефективно створювати та управляти тестовими даними в базі даних для подальших досліджень, розробляти інтерактивні тести з інформатики для мобільних пристроїв та тестувати створений застосунок. Використання цього підходу розглядає кожен запис у таблиці як окремий тест, де для кожного тесту визначається

послідовність питань, можливі відповіді та їх відповідність. Цей метод дозволяє ефективно створювати та управляти тестовими даними в базі даних для проведення подальших досліджень.

3.4 Ілюстрація роботи інтерактивних тестів для мобільних пристроїв зі створеним набором тестових даних

Інтерактивні тести стали важливим інструментом для оцінювання знань та навичок студентів у сучасній освіті. Зокрема, розробка та впровадження тестів для мобільних пристроїв визначає новий рівень доступності та зручності для користувачів. Набір тестових даних, представлений у попередніх розділах, становить основу для розробки та впровадження інтерактивних тестів. Ці дані охоплюють різні типи тестів, включаючи тести із одним правильним варіантом відповіді, тести на встановлення відповідності, тести на встановлення послідовності та інші.

Розглянемо конкретні приклади використання цих тестових даних для створення інтерактивних тестів для мобільних пристроїв. Зокрема, детально розглянемо реалізацію та функціональні можливості розроблених тестів, а також оцінимо їхню ефективність та користь для вивчення інформатики та пов'язаних з нею тем. Такий підхід дозволяє наочно проілюструвати переваги та можливості використання інтерактивних тестів у сучасному освітньому процесі та надає практичний взірець їхнього впровадження на мобільних пристроях.

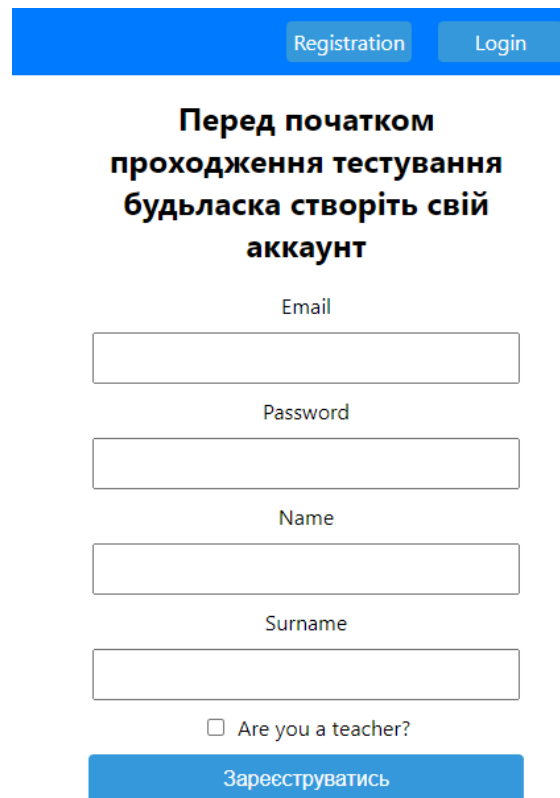
Робота з застосунком починається з реєстрації користувача. Перед тим як користувач розпочне проходження тестів, необхідно створити акаунт. Для цього передбачено інтерактивну форму реєстрації, яка дозволяє введення основних особистих даних та обрання ролі – студента чи вчителя.

Для простоти тестування застосунку логіка валідації ролі здійснюється через чекбокс «Are you a teacher?», проте для забезпечення більш високого

рівня безпеки та автентифікації, дану логіку треба перенести на бекенд. Такий підхід передбачає валідацію за допомогою електронної пошти, де вчителі мають особливі емейли, що надають їм відповідну роль.

Під час реєстрації, користувач повинен ввести свою електронну пошту, пароль, ім'я та прізвище. У разі обрання ролі вчителя, ця інформація доповнюється відповідним чекбоксом. Після натискання на кнопку «Зареєструватись», дані валідуються відповідно до заданої схеми, і якщо вони коректні, користувач створюється в системі.

Цей підхід до реєстрації дозволяє забезпечити безпеку та зручність для користувачів, зокрема тих, хто користується мобільними пристроями.



The image shows a registration form interface. At the top, there is a blue header bar with two buttons: "Registration" and "Login". Below the header, the text reads: "Перед початком проходження тестування будьласка створіть свій аккаунт". The form consists of several input fields: "Email", "Password", "Name", and "Surname". Below these fields is a checkbox labeled "Are you a teacher?". At the bottom of the form is a blue button labeled "Зареєструватись".

Рисунок 3.9 – Сторінка реєстрації користувача

У разі наявності особистого акаунта користувач може увійти в свій акаунт. Можливість використовувати вхідну форму надає зручний та безпечний спосіб входу.

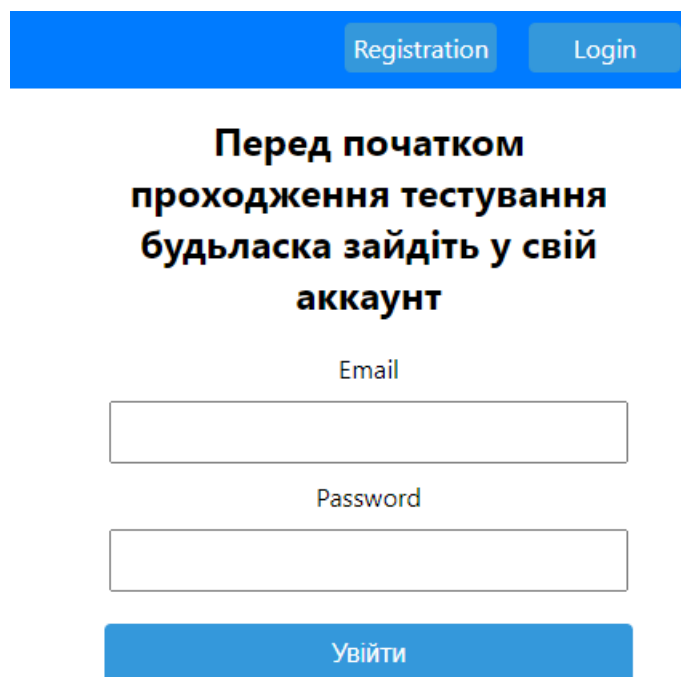
Вхідна форма містить поля для введення електронної пошти та пароля. Для полегшення входу використовується бібліотека Formik для створення форм та Yup для визначення схеми валідації даних.

Користувач вводить свою електронну пошту та пароль у відповідні поля, і після натискання на кнопку «Увійти», введені дані піддаються валідації. У разі успішного входу, користувач авторизується в системі.

При невірному вводі або інших помилках система надає відповідне повідомлення.

Система також надає можливість переходу на сторінку реєстрації для нових користувачів або тих, хто ще не створив акаунт.

Використовуючи посилання «Створіть акаунт», користувач може перейти до форми реєстрації та розпочати процес створення свого облікового запису.



The image shows a login form interface. At the top, there is a blue horizontal bar containing two buttons: 'Registration' and 'Login'. Below this bar, the text reads: 'Перед початком проходження тестування будьласка зайдіть у свій акаунт'. Underneath, there are two input fields: the first is labeled 'Email' and the second is labeled 'Password'. At the bottom of the form, there is a large blue button labeled 'Увійти'.

Рисунок 3.10 – Сторінка логізації користувача

Додаток для тестування призначений для різних категорій користувачів, зокрема для вчителів і учнів, надаючи змогу проведення тестів та подальшого аналізу результатів.

При вході в систему відбувається ідентифікація користувача, і відповідно до його ролі в системі відображається відповідний функціонал.

Для учнів передбачено можливість пройти тести, вибираючи відповіді та перемикаючись між питаннями.

Після завершення тесту учень може переглянути свої результати та правильні відповіді, а отримані дані будуть збережені в системі для подальшого аналізу вчителем.

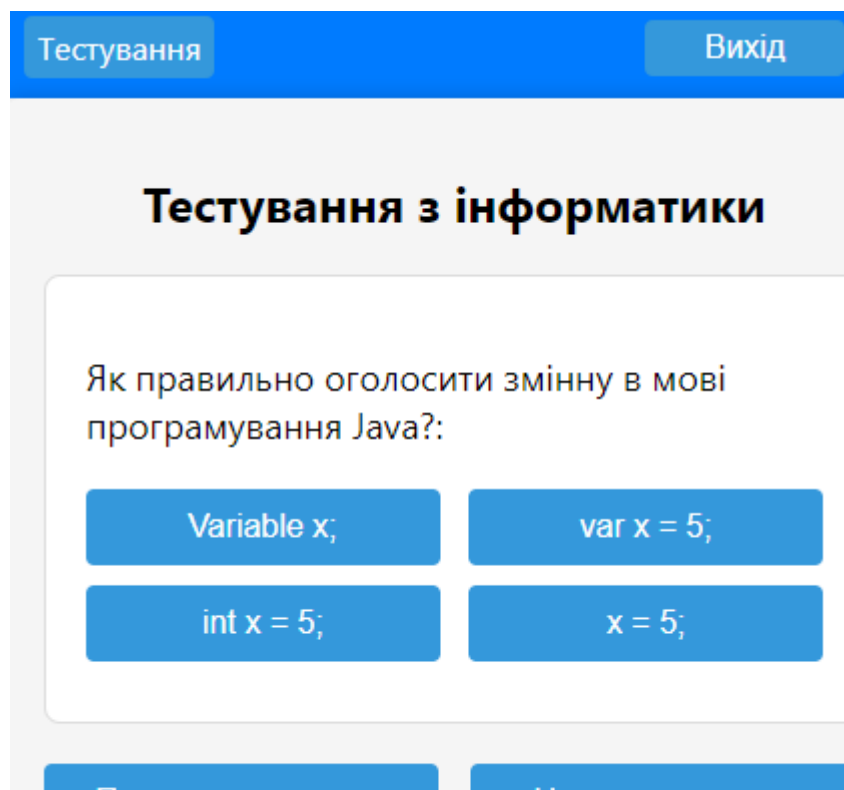


Рисунок 3.11 – Головна сторінка застосунку для учня

Вчителям, у свою чергу, доступні розширені можливості, такі як перегляд статистики тестувань учнів, аналіз результатів, оновлення або видалення тестів.

Для зручності вчителі можуть перейти до сторінки статистики для більш детального розгляду отриманих даних.

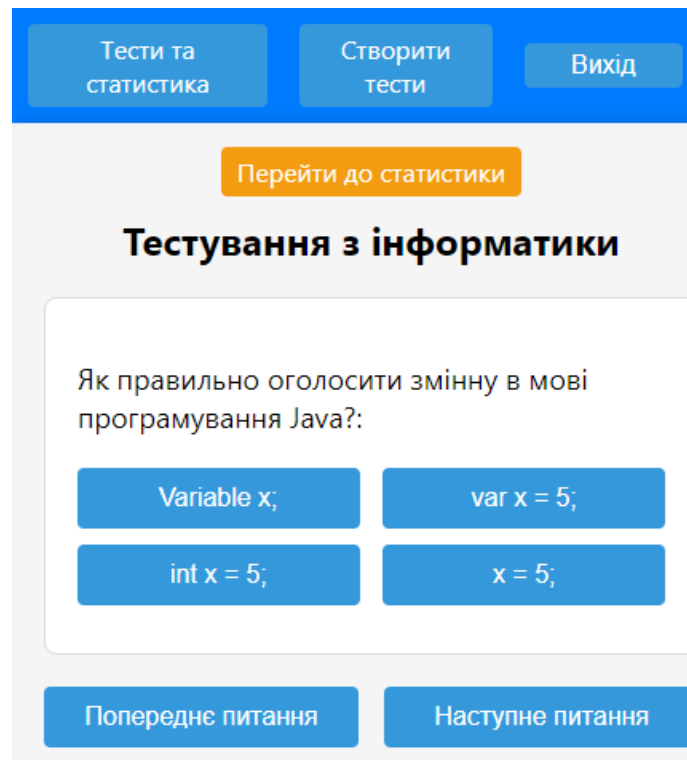


Рисунок 3.12 – Головна сторінка застосунку для вчителя

Важливо відзначити, що ідентифікація ролі «вчитель» здійснюється на основі інформації про електронну пошту, яка визначається під час реєстрації в системі.

При натисканні на кнопку «Створити тести» відкривається сторінка з інтерфейсом для вчителів, де вони можуть створити новий тест. Забезпечуючи зручний вибір типу тесту, сторінка дозволяє вчителям швидко і легко визначитися з необхідною структурою питань.

При завантаженні сторінки вчителю пропонується обрати тип тесту з різних категорій. Кожен варіант представлений кнопкою з чітким описом типу тесту. При натисканні на будь-яку з кнопок, вчитель переходить на відповідну сторінку для створення тесту обраного типу.

Важливим аспектом є використання бібліотеки MUI, яка надає стильні та адаптивні елементи інтерфейсу, такі як кнопки та текст. Це покращує якість інтерфейсу та забезпечує його сучасний вигляд. Це все спрощує процес створення тестів для вчителів, дозволяючи їм ефективно вибирати та налаштовувати необхідний тип тестування.

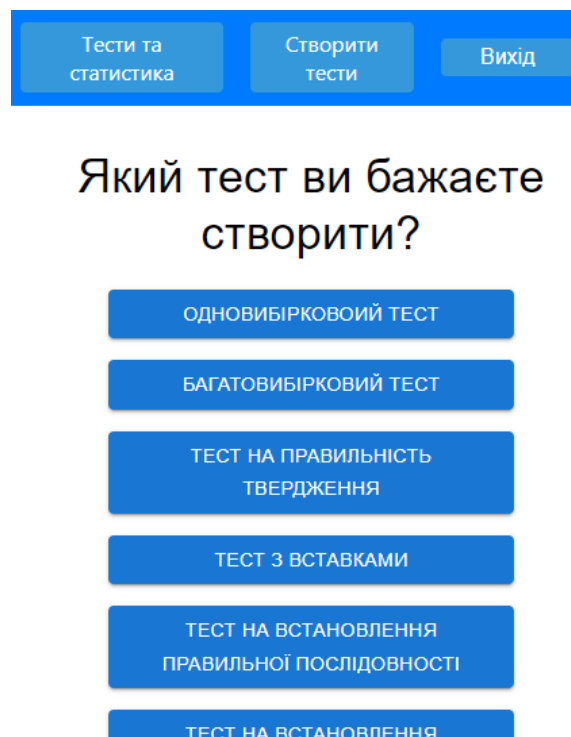
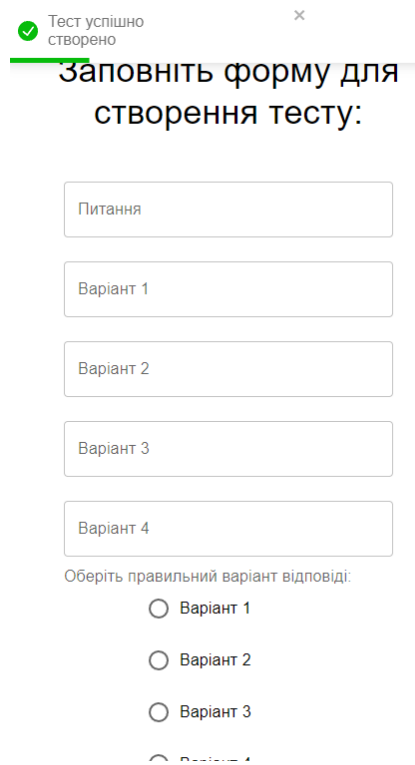


Рисунок 3.13 – Головна сторінка вибору типу тесту

При виборі створення одновибіркового тесту користувач переходить на відповідну сторінку. Ця сторінка представляє собою інтерфейс для створення тестів із одним варіантом відповіді. У процесі створення тесту користувач може вказати питання, визначити чотири варіанти відповідей та вказати

правильний варіант. Вона базується на бібліотеках MUI та Formik для спрощення створення інтерактивних форм.

На цій сторінці використовуються текстові поля для введення питання та варіантів відповідей. Щоб обрати правильний варіант, користувач використовує RadioGroup. Після введення всіх необхідних даних користувач може відправити їх на сервер за допомогою кнопки «Submit». Для зручності використання введено ToastContainer, який відповідає за відображення коротких повідомлень про стан взаємодії з сервером.



Тест успішно створено

Заповніть форму для створення тесту:

Питання

Варіант 1

Варіант 2

Варіант 3

Варіант 4

Оберіть правильний варіант відповіді:

Варіант 1

Варіант 2

Варіант 3

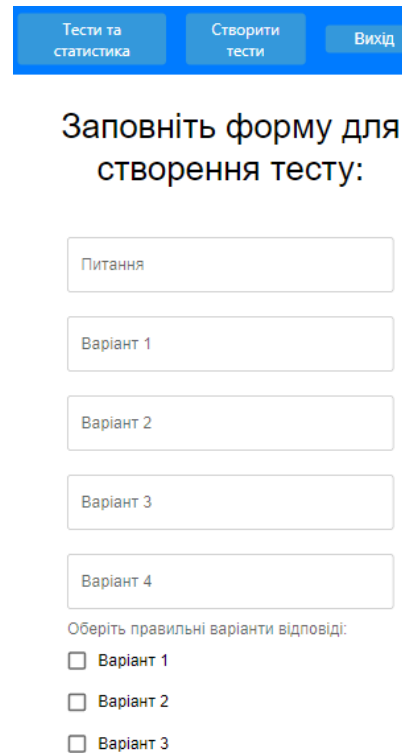
Варіант 4

Рисунок 3.14 – Сторінка створення одновибіркового тесту

При переході на сторінку з інтерфейсом для створення тестів із кількома варіантами відповідей. Використовуються бібліотеки MUI та Formik для створення інтерактивних форм. Користувач може ввести питання та чотири варіанти відповідей, обираючи правильні варіанти за допомогою

чекбоксів. При завершенні введення даних, користувач може відправити їх на сервер за допомогою кнопки «Submit».

У випадку успішного створення тесту виводиться повідомлення про успіх у верхньому правому куті екрану. У випадку помилки також виводиться повідомлення про помилку, щоб користувач був інформований про будь-які проблеми під час створення тесту.



The screenshot shows a web interface for creating a test. At the top, there are three blue buttons: "Тести та статистика", "Створити тести", and "Вихід". Below the buttons, the text "Заповніть форму для створення тесту:" is centered. The form consists of several input fields: "Питання", "Варіант 1", "Варіант 2", "Варіант 3", and "Варіант 4". Below these fields, there is a label "Оберіть правильні варіанти відповіді:" followed by three checkboxes labeled "Варіант 1", "Варіант 2", and "Варіант 3".

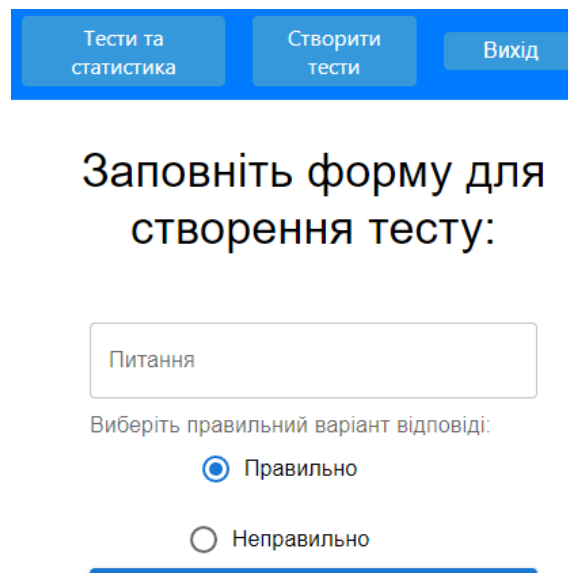
Рисунок 3.15 – Сторінка створення багатовибіркового тесту

При переході на сторінку створення тесту на правильність твердження користувачам пропонується створити тести з питаннями типу «Правда/Неправда». Вона використовує інтерфейс, заснований на формах, щоб зручно вводити необхідну інформацію. Робочий процес користувача виглядає наступним чином. Спершу користувач звертається до сторінки тестування з питаннями типу «Правда/Неправда». На цій сторінці йому доступні поля для введення тексту питання, вибору правильної відповіді

(через радіогрупу з опціями «Правильно» і «Неправильно»), а також чотирьох варіантів відповідей.

Після введення всіх необхідних даних користувач має можливість відправити створений тест на сервер, натиснувши кнопку «Submit». Після цього відображається повідомлення про результат операції. У випадку успішного створення тесту користувач отримує повідомлення про успіх, інакше відображається повідомлення про помилку.

Такий інтерфейс надає зручний і простий спосіб для користувачів створювати тести і взаємодіяти з додатком.



Тести та статистика Створити тести Вихід

Заповніть форму для створення тесту:

Питання

Виберіть правильний варіант відповіді:

Правильно

Неправильно

Рисунок 3.16 – Сторінка створення багатовибіркового тесту на правильність

На сторінці тесту з вставками користувач взаємодіє з інтерфейсом для створення тесту, де присутнє поле для введення пропущеного слова. Крім того, користувач може вказати текст питання, використовуючи відповідне текстове поле.

Після введення необхідної інформації, користувач може відправити створений тест на сервер, натиснувши кнопку «Submit». Після цього

відображається повідомлення про результат операції. У випадку успішного створення тесту користувач отримує повідомлення про успіх, інакше відображається повідомлення про помилку.

Цей інтерфейс забезпечує зручний спосіб для користувачів створювати тести з вставками та ефективно взаємодіяти з функціоналом додатку.

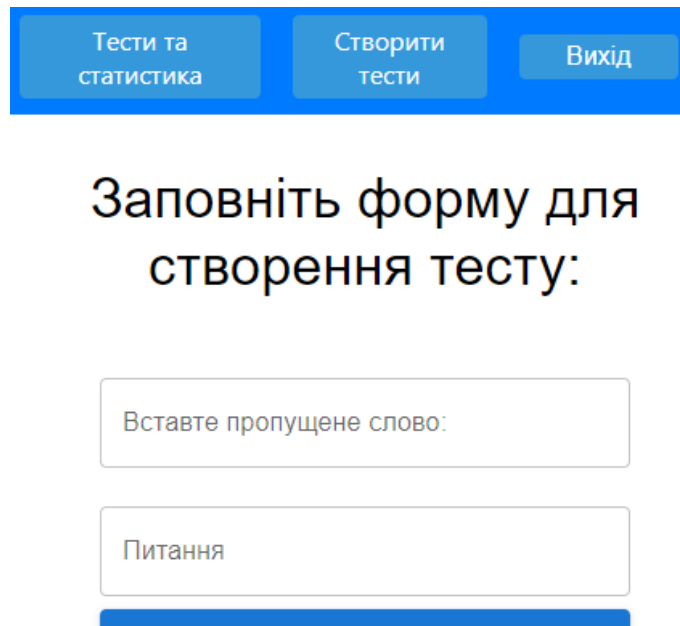


Рисунок 3.17 – Сторінка створення тесту з вставками

На сторінці тестування з послідовністю користувач має можливість створити тест, взаємодіючи з інтерактивним інтерфейсом. Спочатку користувач заповнює тексти для чотирьох питань відповідно до їхнього порядкового номера. Кожне поле питання обов'язково для заповнення.

Далі, користувач вказує порядок цих питань, обираючи значення від 1 до 4 для кожного питання окремо. Це важливо, оскільки порядок впливає на структуру тесту.

Щоб забезпечити коректність введення, застосована валідація, яка перевіряє, чи всі поля заповнені, чи порядкові значення унікальні та чи вони відповідають діапазону від 1 до 4.

Після натискання кнопки «Submit» введені дані відправляються на сервер для подальшої обробки. Після завершення операції користувач отримує відповідне повідомлення: про успіх при створенні тесту або про помилку в разі невідповідності вимогам або технічних проблем.

Заповніть форму для створення тесту:

	Порядок
Порядок питання 1	1
	Порядок
Порядок питання 2	2
	Порядок
Порядок питання 3	3
	Порядок
Порядок питання 4	4

Рисунок 3.18 – Сторінка створення тесту на послідовність

На сторінці тестування на відповідність користувач може створити тест, використовуючи інтерактивний інтерфейс. Спочатку користувач заповнює поля для трьох елементів та відповідних відповідей. Кожне поле для елемента та відповіді обов'язково для заповнення.

Після заповнення елементів та відповідей користувач натискає кнопку «Submit». Введені дані обробляються, і їхній результат надсилається на сервер для подальшої обробки. Після завершення операції користувач отримує відповідне повідомлення: про успіх при створенні тесту або про помилку в разі невідповідності вимогам або технічних проблем.

На сторінці використовується бібліотека MUI для створення компонентів, таких як текстові поля та кнопка. Під час створення тесту

користувач отримує зручний та інтуїтивно зрозумілий інтерфейс, що полегшує взаємодію з програмою.

Заповніть форму для
створення тесту:

Елемент 1
Відповідь 1
Елемент 2
Відповідь 2
Елемент 3
Відповідь 3

Рисунок 3.19 – Сторінка створення тесту на відповідність

Далі, нажавши на кнопку «Тестування» за студента, або кнопку «Тести та статистика» ми повертаємось на компонент головної сторінки і маємо можливість пройти тести. В залежності від виду тесту, який нараз представлено для проходження користувач бачить різні компоненти, які відрізняються своєю структурою. Ці компоненти адаптивно змінюються в залежності від того, яка саме фаза представлена перед користувачем, а саме процес проходження тестування, або перегляд пройденого тестування після його закінчення, через що можна перейти до розгляду цих сторінок одночасно. Сторінка проходження тестів має на меті представити користувачеві питання та варіанти відповідей для тестування його знань. Кожне питання включає в себе текст питання та список варіантів відповідей.

При проходженні тестування користувач здійснює навігацію між тестами завдяки кнопкам «Попереднє питання» або «Наступне питання», при цьому обрана відповідь користувачем зберігається та може бути змінена пізніше.

Компонент одновибіркового тесту відображає питання та варіанти відповідей для тестування типу «одного варіанту відповіді». Кожен варіант відображається у вигляді кнопки, яку користувач може вибрати. Колір кнопки змінюється в залежності від того, чи обрано користувачем цей варіант. Якщо тест вже завершено, правильна відповідь виділяється червоним кольором.

Користувач може вибрати один варіант відповіді для кожного питання. Якщо тест завершено, користувач може переглядати правильну відповідь, яка підкреслена червоним кольором. Це допомагає користувачеві перевірити свої результати та вдосконалити свої знання.

Створено з використанням бібліотеки Redux для управління станом додатка та забезпечення обміну даними між компонентами. Компонент взаємодіє із зовнішнім Redux-стейтом для отримання і збереження відповідей користувача та стану завершення тесту.

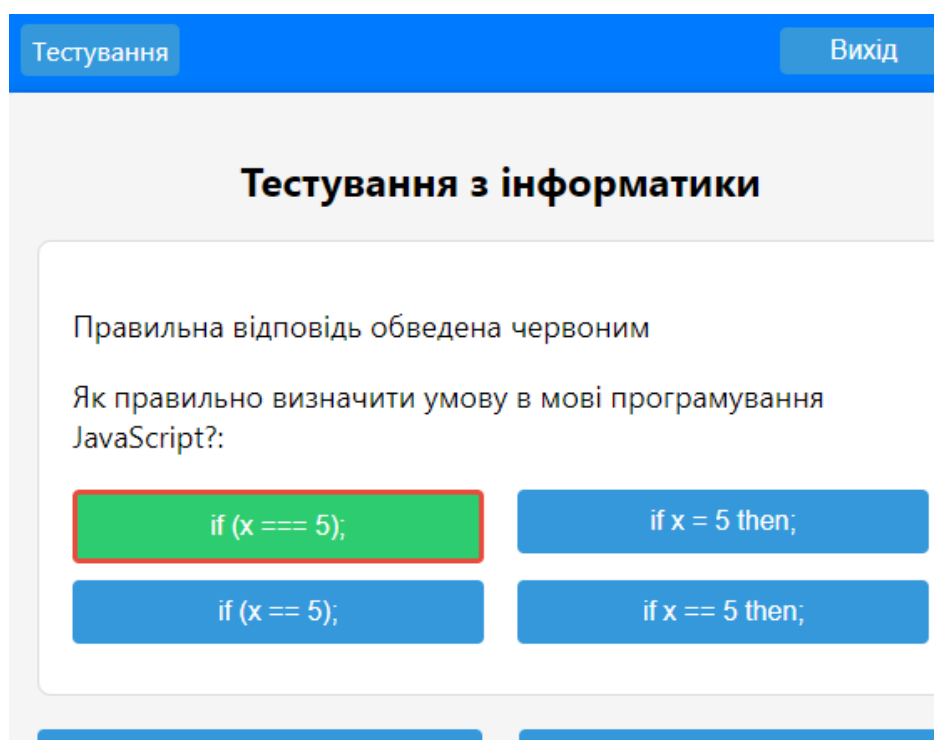
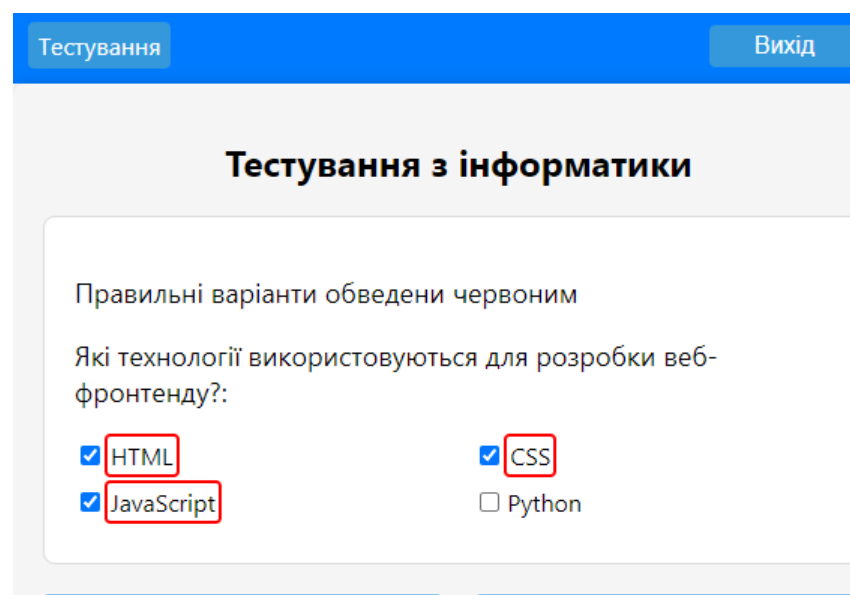


Рисунок 3.20 – Вигляд одновибіркового тесту

У компоненті багатовибіркового тесту користувач повинен вибрати один чи кілька правильних варіантів відповіді. Кожен екземпляр цього компонента отримує дані питання, текстовий опис питання, варіанти відповідей та правильні відповіді та поточний індекс питання в тесті.

В компоненті використовується локальний стан, що відстежує обрані користувачем варіанти відповідей для поточного питання.

Інтерфейс компонента включає блок текстового опису питання, а також блок відповідей, кожна з яких представлена чекбоксом та текстовою міткою. Під час проходження тесту користувач може обирати відповіді, але ця можливість блокується після завершення тесту. У випадку завершення тесту правильні відповіді відзначаються червоним кольором. Компонент також містить умовні перевірки для визначення стану завершення тесту та відображення повідомлень користувачу.



Тестування Вихід

Тестування з інформатики

Правильні варіанти обведені червоним

Які технології використовуються для розробки веб-фронтенду?:

<input checked="" type="checkbox"/> HTML	<input checked="" type="checkbox"/> CSS
<input checked="" type="checkbox"/> JavaScript	<input type="checkbox"/> Python

Рисунок 3.21 – Вигляд багатовибіркового тесту

Наступний компонент створений для відображення тестового питання типу «правда» чи «неправда». Кожен екземпляр цього компонента отримує дані питання, текстовий опис питання, правильна відповідь та поточний індекс питання в тесті.

Компонент використовує локальний стан для відстеження обраної користувачем відповіді. Інтерфейс компонента включає блок текстового опису питання та блок відповідей у вигляді двох кнопок. Під час проходження тесту користувач може обирати відповіді, але ця можливість блокується після завершення тесту. У випадку завершення тесту правильна відповідь відзначається червоним кольором. Компонент також містить умовні перевірки для визначення стану завершення тесту та відображення повідомлень користувачу.

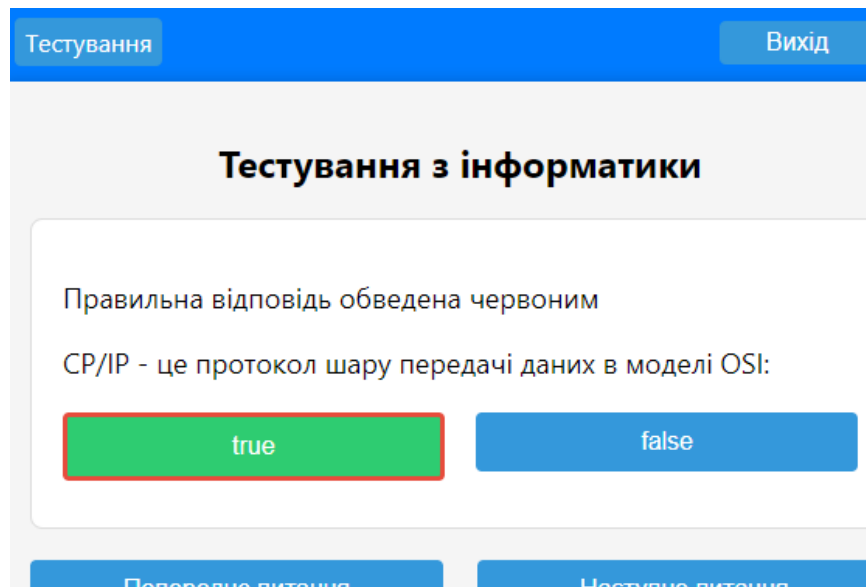


Рисунок 3.22 – Вигляд тесту на правильність

Компонент наступного тесту створено з метою відображення тестового питання, де користувач повинен вставити пропущене слово. Кожен екземпляр цього компонента отримує дані питання (текстовий опис питання та правильна відповідь) та поточний індекс питання в тесті.

Компонент використовує локальний стан для відстеження введеної користувачем відповіді. Зміни у цьому стані спричиняються функцією, яка оновлює введену користувачем відповідь.

Інтерфейс компонента включає блок текстового опису питання та поле вводу, де користувач може вписати свою відповідь. У випадку завершення тесту правильна відповідь виводиться під полем вводу. Компонент також містить умовні перевірки для визначення стану завершення тесту та відображення повідомлень користувачу.

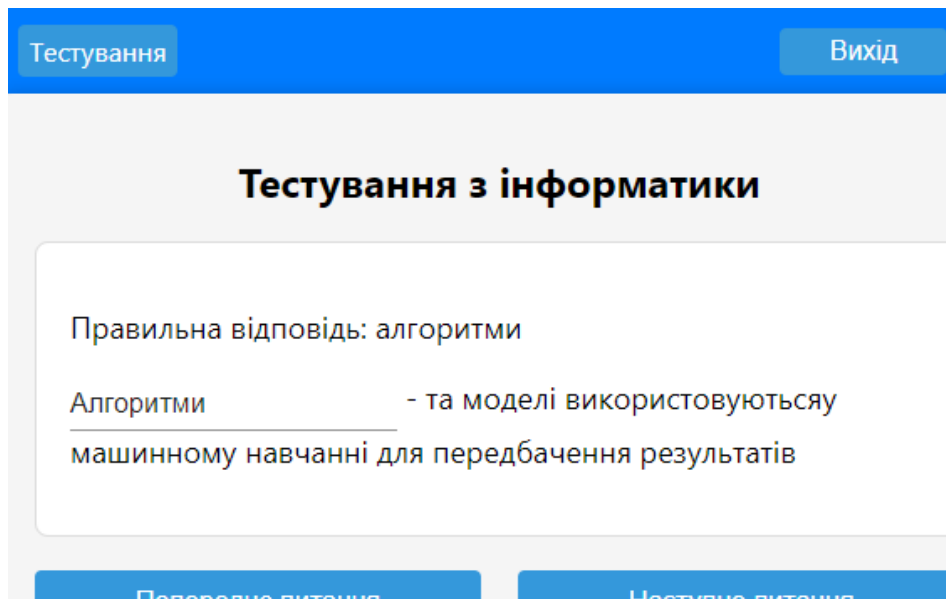


Рисунок 3.23 – Вигляд тесту зі вставками

Для відображення тесту на послідовність створено компонент для відображення тестового питання, де користувач повинен впорядкувати

елементи відповідно до заданого порядку. Кожен екземпляр цього компонента отримує дані питання та поточний індекс питання в тесті.

Компонент також використовує локальний стан для відстеження введеної користувачем відповіді, а також підтримує drag-and-drop для перетягнутого елемента. Зміни у стані спричиняються відразу після завантаження компонента або при перемішуванні елементів.

Інтерфейс компонента включає блок текстового опису питання та список блоків з елементами для сортування. Кожен елемент може бути перетягнутий за допомогою атрибуту «draggable» та розміщений у відповідному порядку. Компонент має кнопку «Перемішати», яка викликає функцію для перемішування порядку елементів, і кнопку «Зберегти відповідь», яка викликає функцію для оновлення та збереження введеної користувачем відповіді. Умовні перевірки використовуються для блокування перетягування та змішування, якщо тест вже завершено.

У випадку завершення тесту виводиться повідомлення про правильну послідовність разом із текстовим описом кожного елемента.

Тестування

Вихід

Тестування з інформатики

Правильна послідовність

- 1: Планування спринту
- 2: Ретроспектива
- 3: Розробка backlog
- 4: Постійне вдосконалення

Планування спринту	1
Ретроспектива	2
Розробка backlog	3
Постійне вдосконалення	4

ПЕРЕМЕШАТЬ

СОХРАНИТЬ ОТВЕТ

Рисунок 3.24 – Вигляд тесту на послідовність

Останнім компонентом для відображення тестового питання, є форма тесту на відповідність, де користувач повинен відповідати на питання, встановлюючи відповіді у відповідному порядку. Кожен екземпляр цього компонента отримує дані питання, а саме список питань, список відповідей та відповідність між ними та поточний індекс питання в тесті.

Компонент використовує локальний стан для відстеження введеної користувачем відповіді для визначення перетягнутого елемента. Зміни у стані також спричиняються відразу після завантаження компонента або при перетягуванні елементів. Інтерфейс компонента включає дві колонки, кожна з яких містить блоки текстового опису питань або відповідей. Кожен елемент може бути перетягнутий за допомогою атрибуту «draggable» та розміщений у відповідному порядку. Всі пари правильних питань та відповідей відзначені, коли тест вже завершено.

Тестування з інформатики

Правильні питання та відповіді повинні відповідати один одному за порядком у списку

Правильна відповідність

Питання 1: ---> Android development

Питання 2: ---> Cross-platform development

Питання 3: ---> iOS development

Питання

Java частіше використовується для

Kotlin частіше використовується для

Swift частіше використовується для

Відповіді

Cross-platform development

Android development

iOS development

СОХРАНИТЬ ОТВЕТ

Рисунок 3.25 – Вигляд тесту на відповідність

Якщо користувач на останньому питанні, кнопка наступного питання змінюється на «Завершити тестування», а якщо тестування завершено, то на «Перейти до результату», залежно від того, чи вже пройдені всі питання.

Після завершення тестування виводяться результати, включаючи кількість правильних відповідей. Кнопка «Переглянути тести» дозволяє переглянути всі питання та відповіді.

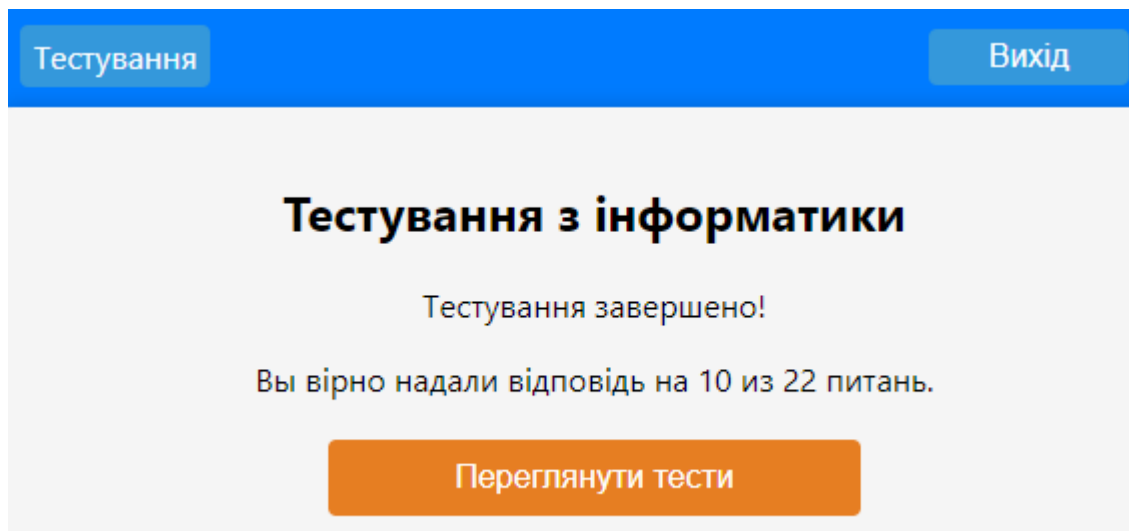


Рисунок 3.26 – Вигляд сторінки результату

За результатом був успішно розроблений та представлений інтерактивний тестовий розділ для мобільних пристроїв, включаючи різноманітні типи тестів, такі як вибір одного варіанту, вибір декількох варіантів, правда/неправда, заповнення пропусків, послідовне розташування та відповідність. Реалізація цього функціоналу дозволила ефективно тестувати користувачів, забезпечуючи їм можливість взаємодії з різними типами завдань.

Забезпечений функціонал адаптований для оптимальної роботи на мобільних пристроях та відповідає усім вимогам, надаючи інтуїтивно зрозумілий та зручний інтерфейс. Розроблений розділ був протестований на основі створених тестових даних, підтверджуючи його коректну роботу та високу ефективність взаємодії з користувачем. Кінцевий продукт дозволяє користувачам успішно проходити тести та переглядати результати, враховуючи різноманітні типи питань і сприяючи позитивному досвіду користувача на мобільних пристроях.

3.5 Аналіз результатів тестування та оцінка ефективності, зіставлення їх з поставленими вимогами

Аналіз результатів тестування та оцінка їх ефективності є ключовим етапом в оцінці здійснення навчального процесу та визначенні його впливу на користувачів. В контексті навчання з використанням інтерактивних тестів для мобільних пристроїв, цей процес дозволяє визначити ступінь успішності студентів у засвоєнні матеріалу та ефективність навчальних тестів як інструменту навчання.

Оцінка результатів тестування відбувається на різних рівнях, починаючи від загального аналізу коректності відповідей і закінчуючи визначенням кількості правильних відповідей для кожного конкретного тесту. Результати порівнюються з попередньо визначеними вимогами та стандартами успішності, що дозволяє визначити, наскільки студент відповідає передбачуваному рівню знань.

У випадку інтерактивних тестів для мобільних пристроїв, оцінка ефективності також включає в себе аналіз взаємодії користувачів із тестами та їхню зручність використання. Застосування статистичних методів дозволяє виявити та виділити найбільш та найменш успішні аспекти тестів, покращення яких може підвищити загальну ефективність навчання.

Загальний підсумок аналізу допомагає визначити сильні та слабкі сторони навчального процесу, а також покращити якість самих тестів для забезпечення оптимального засвоєння матеріалу студентами.

Щодо аналізу результату проведення тестування, то тут головним компонентом є розроблена для відображення статистичних даних тестування таблиця.

Основний функціонал компонента включає в себе формування таблиці (додаток Б), що містить інформацію про кожного користувача, його результати тестів і загальну кількість правильних відповідей. Кількість стовпців у таблиці динамічно визначається на основі отриманих даних, що надає гнучкість у відображенні різноманітних тестових сценаріїв.

Статистика тестування

№	User Name	Test 1	Test 2	Test 3
1	Urii Mihailov	+	+	-
2	Oleha Zadorna	+	+	+

Рисунок 3.27 – Частина таблиці результатів з інформацією про користувачів

Для кожного тесту обчислюється процент прохідності, який далі використовується для визначення кольорів у таблиці. Це сприяє візуальному виділенню тестів, які можуть виявитися або занадто легкими, або занадто складними для студентів.

Компонент також містить інформаційний блок, який розкриває значення кольорів, щоб надати користувачеві зрозумілу інтерпретацію результатів. Загальний функціонал сторінки полягає в забезпеченні користувачеві можливості ефективно вивчати та аналізувати результати

тестування, визначати успішність та виділяти можливі області для подальшого вдосконалення тестів.

% Test Passability	100.00%	100.00%	66.67%	66.67%	100.00%	100.00%	83.33%	100.00%	100.00%	33.33%	83.33%
--------------------	---------	---------	--------	--------	---------	---------	--------	---------	---------	--------	--------

■ – ячейки такого кольору повідомляють про занадто легкі тести. Більше 90% студентів проходить їх вірно. Будь-ласка перегляньте їх.
■ – ячейки такого кольору повідомляють про занадто складні тести. Менше 20% студентів проходить їх вірно. Будь-ласка перегляньте їх

Рисунок 3.28 – Частина таблиці результатів з інформацією про процент прохідності

Сторінка статистики виконує важливу роль в системі інтерактивних тестів, дозволяючи аналізувати результати тестувань та оцінювати ефективність навчання. Компонент отримує статистичні дані, включаючи кількість правильних відповідей, результати тестів та імена користувачів, і представляє їх у вигляді зрозумілої таблиці. Здійснюється розрахунок відсотку прохідності для кожного тесту та загального відсотка успішних відповідей, що допомагає оцінити рівень засвоєння матеріалу студентами.

Особливо корисна є можливість виділення тестів кольоровими маркерами в залежності від їхньої важкості. Зелений колір вказує на тести, які можуть бути занадто легкими, тоді як червоний - на тести з можливою високою складністю. Це дає можливість визначити та виправити можливі проблеми з рівнем складності конкретних тестів.

Компонент також надає пояснення щодо кольорових маркерів, сприяючи зрозумілій інтерпретації результатів тестування. Загалом, сторінка статистики створена для забезпечення об'єктивного аналізу та покращення процесу навчання через ефективне використання інтерактивних тестів.

Що стосується аналізу роботи застосунку в цілому, то тут треба відмітити, що в ході дослідження та розробки інтерактивних тестів для мобільних пристроїв з інформатики було проведено аналіз результатів

тестування та оцінка їх ефективності. Отримані результати були порівняні з встановленими вимогами та критеріями успішності.

На основі визначених вимог та етапів конструювання тестів була розроблена інфраструктура та вибрана методологія розробки інтерактивних тестів для мобільних пристроїв. Було сформовано та проаналізовано правила конструювання та форм тестових завдань, а також проведено аналіз та опрацювання результатів тестування, включаючи методи їх представлення.

Реалізація інтерактивних тестів була здійснена з використанням вибраних технологій та інструментів розробки. Було налаштовано програмне середовище для проведення дослідження та розробки тестів. Також був створений набір тестових даних для проведення дослідження, що дозволило реалізувати інтерактивні тести та проілюструвати їх роботу на мобільних пристроях. В результаті розроблено інтерактивний навчальний застосунок для мобільних пристроїв, спрямований на поліпшення навчального процесу та впровадження його в систему мобільних застосунків

ВИСНОВКИ

У рамках кваліфікаційної роботи було розроблено, реалізовано та протестовано інтерактивні тести з інформатики для мобільних пристроїв.

Аналіз результатів тестування та оцінка ефективності були проведені згідно з поставленими вимогами. Результати тестів порівнювалися з критеріями успішності, що дозволило здійснити об'єктивну оцінку ефективності розроблених інтерактивних тестів. Отримані дані є основою для подальшого вдосконалення та оптимізації системи тестування для мобільних пристроїв з метою поліпшення навчального процесу.

У ході дослідження була визначена інфраструктура та обрана методологія розробки тестів. Проведено аналіз та опрацювання результатів, створено ілюстрації роботи застосунку та проведено аналіз його ефективності.

Об'єктом дослідження була сфера розробки інтерактивних тестів для мобільних пристроїв, а предметом було вдосконалення навчання та оцінка знань з інформатики за допомогою мобільних пристроїв. Мета дослідження була досягнута через вирішення завдань, таких як аналіз тестів, визначення вимог для конструювання та вибір технологій розробки.

Результатом роботи стало створення функціонального застосунку для проведення інтерактивних тестів на мобільних пристроях, який відповідає сучасним вимогам освітнього процесу та технологічним стандартам. Застосунок пройшов тестування на власному датасеті, що дозволило здійснити аналіз результатів тестування та оцінити його ефективність.

Це все свідчить про досягнення поставленої задачі та відповідності застосунку поставленим вимогам.

Результати дослідження апробовано у вигляді 3 тез доповідей під час Міжнародного молодіжного форуму «РАДІОЕЛЕКТРОНІКА І МОЛОДЬ У ХХІ СТОЛІТТІ» [40].

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Smith, A., & Johnson, B. (2019). Interactive Testing in Educational Technology: A Comprehensive Guide. *Educational Technology Research and Development*, 67(5), 1303-1327.
2. Clark, J. S., & Mayer, R. E. (2021). *E-Learning and the Science of Instruction: Proven Guidelines for Consumers and Designers of Multimedia Learning*. Wiley.
3. Frey, B. A., & Birnbaum, M. L. (2019). Interactive Learning Technologies in the Online Classroom: An Evidence-Based Evaluation of the Impact on Student Outcomes. *Information Systems Education Journal*, 17(3), 4-18.
4. Булах, І. (ред.). (2005). *Основи педагогічного оцінювання: Ч. II. Практика: Навчально-методичні та інформаційно-довідкові матеріали для педагогічних працівників*. Київ: Майстер-клас.
5. Нагаєв, В.М. (2012). *Методичні рекомендації по провадженню модульнорейтингової технології навчально-творчої діяльності студентів (із дисципліни управлінського циклу)*. Харків: ХНАУ ім. В.В. Докучаєва, 73.
6. Завальнюк, О. М. (2021). *Тестування як метод педагогічного контролю успішності студентів: науково-методичний збірник*. Кам'янець-Подільський національний університет ім. І. Огієнка. Кам'янець-Подільський: Аксіома, 111.
7. Бойко, Я. В. (2017). *Математичне та програмне забезпечення кластеризації ухвалення рішень*.
8. Гнатієнко, Г. М., & Снитюк, В. Є. (2008). *Експертні технології прийняття рішень*. К.: ТОВ «Маклаут».
9. Гороховатський, В. О., & Творошенко, І. С. (2021). *Методи інтелектуального аналізу та оброблення даних: навч. посібник*.

10. Jelodar, H., Wang, Y., Yuan, C., Feng, X., Jiang, X., Li, Y., & Zhao, L. (2019). Latent Dirichlet allocation (LDA) and topic modeling: models, applications, a survey. *Multimedia Tools and Applications*, 78, 15169-15211.
11. Gorokhovatskyi V., Tvoroshenko I. (2023) Identification of visual objects by the search request. *International scientific symposium «INTELLIGENT SOLUTIONS-S». Computational intelligence (results, problems and perspectives). Decision making theory: proceedings of the international symposium*, September 28, 2023, Kyiv-Uzhorod, Ukraine, pp. 25-27.
12. Творошенко І.С. (2021). Технології прийняття рішень в інформаційних системах: навч. посібник. Харків: ХНУРЕ.
13. Daradkeh Y.I., and Tvoroshenko I. (2020) Technologies for Making Reliable Decisions on a Variety of Effective Factors using Fuzzy Logic, *International Journal of Advanced Computer Science and Applications*, 11(5), pp. 43-50.
14. Gati, I., Levin, N., & Landman-Tal, S. (2019). Decision-making models and career guidance. *International handbook of career guidance*, 115-145.
15. G. Gallardo-Valencia, J. I. Asensio-Pérez, I. J. Ruiz-Cortés, and E. Tovar-Caro. (2017). A Systematic Literature Review on e-Assessment of Programming Assignments. *IEEE Transactions on Education*.
16. S. S. Kamath, A. J. D'Souza, and R. M. Rodrigues (2018). "Developing a Web-Based Mobile Learning Application for Computer Science Courses. *International Journal of Mobile and Blended Learning*, 18-35.
17. R. M. Carvalho, G. Guiné, and L. Morgado (2018). Mobile Technologies in Educational Assessment: A Review of the Literature. *Journal of King Saud University - Computer and Information Sciences*, 28-36.
18. S. Cho and W. Kim. (2018). A Design and Implementation of Mobile-Based Programming Learning Application Using the Zone of Proximal Development. *International Journal of Software Engineering and Its Applications*, 235-248.

19. A. Ramya and T. Vigneshwaran. (2018). Design and Development of a Mobile-Based Learning Application for Computer Science Education. *Journal of King Saud University - Computer and Information Sciences*, 31-37.
20. J. Zhang, D. Zhao, and D. Huang (2017). Mobile Learning in Computer Science Education: A Systematic Review *Computers & Education*, 109, 112-135.
21. D. Gasparini, L. Papazoglou-Chalikias, G. Loiacono, and T. Menapace (2019). Mobile Learning Applications for Computer Science Education: A Systematic Review *Computers in Human Behavior*, 100, 38-57.
22. H. Aljehani, S. Drew, and S. Alhazbi. (2018). A Systematic Review of Mobile Learning in Saudi Arabia: Trends, Challenges, and Opportunities *International Journal of Interactive Mobile Technologies*, 57-77.
23. A. T. Tosho, B. A. Suleiman, and A. B. Yahuza. (2017). Design and Implementation of a Mobile-Based Learning Management System (LMS) for Computer Science Education. *International Journal of Scientific and Research Publications*, 167-172.
24. L. S. de la Fuente Valentín, S. S. Martínez, and E. J. Torres-Coronas. (2018). Mobile Learning and Gamification in Computer Science Higher Education: A Case Study. *Education Sciences*, 190.
25. Tvoroshenko, I., and Kuznetsov, M. (2021). Research Results of Functional, White Box, and Smoke Testing Methods for Mobile Applications.
26. Tvoroshenko, I.S., and Kuznetsov, M. (2021). About the Role of Testing in the Process of Mobile Application Development.
27. Т. В. Лісова. (2022). Моделі та методи сучасної теорії тестів: навчально-методичний посібник, 112.
28. О. І. Локшиної (2021). Моніторинг якості освіти: становлення та розвиток в Україні: Рекомендації з освітньої політики, 160.
29. Н. М. Стеценко. (2013). Основи педагогічних вимірювань та моніторингу якості освіти: навчально-методичний посібник, 257.

30. Я. Я. Болюбаш, І. Є. Булах, М. Р. Мруга, І. В. Філончук. (2017). Педагогічне оцінювання і тестування. Правила, стандарти, відповідність. Наукове видання, 272-273.
31. В.П. Сергієнко, М. П. Малежик, Т. В. Сіткар. (2012). Комп'ютерні технології в тестуванні: навчальний посібник, 270-275.
32. В. П. Сергієнко, Л. О. Кухар. (2011). Методичні рекомендації зі складання тестових завдань, 45-48.
33. Терещук, Г.В. (2017). Удосконалення змісту й технологій оцінювання якості підготовки майбутніх фахівців відповідно до вимог Європейської асоціації якості освіти: Матеріали регіонального науково-практичного семінару, 135-141.
34. Фетісов, В.С. (2011). Комп'ютерні технології в тестуванні: навчально-методичний посібник, 251.
35. Фігурська, Л. В. (2009). Становлення та розвиток тестування як методу педагогічної діагностики. 11-19.
36. Фігурська, Л.В.; (2017). Технологія конструювання педагогічних тестів: науково-методологічні основи, 127–132.
37. Михайлов, К.М. (2000). Моделювання системи рейтингової оцінки знань, 343–346.
38. Михайлов, К.М., Каленбет, Д.В. (2012). Деякі підходи до системи тестування, 503–507.
39. Мойсеюк, Н.Є. Локшина, О.І. (2001). Педагогіка: Навчальний посібник, 311-317.
40. Мягкий Б. (2023) Дослідження та розробка інтерактивних тестів для мобільних пристроїв, методи побудови, проведення тестування та оцінки результатів, *«Розвиток наук в умовах нової реальності: проблеми та перспективи»*, (20 жовтня, 2023). м. Луцьк, України, стр. 72-76.