

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**ХЕРСОНСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ**  
**Факультет комп'ютерних наук, фізики та математики**  
**Кафедра інформатики, програмної інженерії та економічної**  
**кібернетики**

**РОЗРОБЛЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ**  
**ОПРАЦЮВАННЯ ДАНИХ ПЕДАГОГІЧНОГО**  
**ЕКСПЕРИМЕНТУ**

**Кваліфікаційна робота (проект)**  
на здобуття ступеня вищої освіти “бакалавр”

Виконав: студент 4 курсу  
Спеціальності 121 Інженерія  
програмного забезпечення  
Освітньо-професійної програми  
«Інженерія програмного забезпечення»  
першого (бакалаврського) рівня освіти  
Нестерчук Микола Сергійович  
Керівники: доктор педагогічних наук,  
професор Шерман Михайло Ісаакович,  
доктор педагогічних наук, професор  
Співаковський Олександр  
Володимирович  
Рецензент: кандидат педагогічних наук,  
старший викладач  
Григор'єва Валентина Борисівна,

## ЗМІСТ

### ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

	3
<b>ВСТУП</b> .....	5
<b>РОЗДІЛ 1. Аналіз предметної області</b> .....	7
1.1. Online-тестування, та його застосування в дистанційному навчанні .....	7
1.2. Опис предметної області .....	9
1.3. Аналіз аналогів і прототипів .....	12
1.4. Перелік завдань які підлягають вирішенню в процесі розробки .....	20
<b>РОЗДІЛ 2. Проектування Web-застосунку</b> .....	23
2.1. Проектування структурної схеми системи .....	23
2.2. Проектування баз даних .....	25
2.3. Вимоги до систем передавання інформації .....	29
2.4. Вимоги до програми та програмного виробу .....	30
<b>РОЗДІЛ 3. Розробка Web-застосунку</b> .....	32
3.1. Розробка і опис робочих алгоритмів .....	32
3.2. Розробка інтерфейсу взаємодії користувача з системою ....	33
<b>ВИСНОВКИ</b> .....	39
<b>СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ</b> .....	40
<b>ДОДАТКИ</b> .....	43
Додаток А .....	43

### ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

*Технологія розробки програмного забезпечення (ПО) являє собою комплекс організаційних заходів, операцій і прийомів, спрямованих на розробку програмних продуктів високої якості в рамках відведеного бюджету і в строк. Технології включають методики, методології, засоби і процедури розробки ПО.*



**Python** — інтерпретована об'єктно-орієнтована мова програмування високого рівня зі строгою динамічною типізацією.

# django

**Django** (Джанго) — високорівневий відкритий Python-фреймворк (програмний каркас) для розробки веб-систем. Названо його було на честь джазмена Джанго Рейнхардта (відповідно до музичних смаків одного з засновників проекту)



**SQLite** - це вбудована бібліотека, яка реалізує автономний, безсерверний, нульової конфігурації, транзакційний механізм СУБД SQL. Це база даних, яка налаштована на нуль, що означає, як і інші бази даних, які вам не потрібно налаштовувати в вашій системі.



**Bootstrap** - вільний набір інструментів для створення сайтів і веб-додатків. Включає в себе HTML і CSS-шаблони оформлення для типографіки, веб-форм, кнопок, міток, блоків навігації та інших компонентів веб-інтерфейсу, включаючи JavaScript-розширення.



**Vue** (приблизно як view) - це прогресивний фреймворк для створення користувацьких інтерфейсів. На відміну від фреймворків-монолітів, Vue створений придатним для поступового впровадження. Його ядро в першу чергу вирішує завдання рівня уявлення (view), що спрощує інтеграцію з іншими бібліотеками та існуючими проектами. З іншого боку, Vue повністю підходить і для створення складних односторінкових додатків (SPA, Single-Page Applications), якщо використовувати його спільно з сучасними інструментами та додатковими бібліотеками.

## ВСТУП

Бурхливий розвиток промисловості і науки, нестримне зростання обсягів інформації, що надходить призвели до того, що людина опинилася не в змозі сприйняти і переробляти все, йому призначене. Широка автоматизація практично всіх галузей привела до корінної

перебудови вимірювальної техніки: в її завдання поряд з вимірюванням входить інформаційне обслуговування досліджуваного об'єкта, яке включає автоматичний збір, уявлення, доставку, відображення, обробку та аналіз інформації. Тому основою сучасної вимірювальної техніки є не окремий вимірювальний прилад, а інформаційно-вимірювальна система (ІВС), яка і вирішує поставлене завдання.

Проектуванням таких систем займаються в багатьох організаціях, проте необхідні для цього теорія і методи інженерного розрахунку ІВС ще далекі від завершення. Важливим завданням при побудові ІВС є побудова системи збору інформації. При цьому є необхідність використання різноманітних перетворювачів одного виду інформації в інші. Завдання також ускладнюється великою кількістю несуттєвою інформації, тому з'являється необхідність перетворення вимірювального повідомлення в уявлення з адаптацією, що робить його мінімальним за обсягом. За це відповідають системи стиснення даних. Так як в реальних каналах ліній зв'язку інформація передається з перешкодами, то ефективним засобом підвищення достовірності передачі повідомлень є побудова системи кодування.

Залежно від того, з якою якістю повинна передаватися інформація, терміни вживаються в такому перешкодостійкі коди: коригувальний, який виявляє і виправляє. При цьому система збору та обробки інформації повинна бути досить точною і володіти високою швидкістю. Всі елементи системи повинні бути уніфіковані і легко замінені в процесі експлуатації.

На сучасному етапі розвитку науки і техніки до ІВС пред'являються все більш жорсткі вимоги. Це пов'язано зі збільшенням числа джерел інформації і різноманітним завдань, які ставляться при випробуваннях і дослідженнях різних фізичних об'єктів і процесів. Тому сучасні системи збору та передачі інформації повинні володіти високою інформативністю і інформаційної гнучкістю, що дозволяє здійснити

програмний опитування джерел інформації і забезпечити можливість зміни частоти опитування і точності вимірювань по різних каналах.

**Актуальність дослідження** – створення програмного забезпечення, дозволить викладачам за власним бажанням обробляти різноманітні тести.

**Мета дослідження** – створити програмний продукт котрий дозволить викладачам зручно обробляти результати тестів.

**Об'єктом дослідження** є методи інформаційних систем обробки результатів тестів

**Предмет дослідження** – є web-додаток як засіб обробки результатів тестових завдань для викладача.

**Завдання дослідження** полягають у наступному:

1. Дослідити системи зворотного зв'язку, вивчення технології для програмного засобу;
2. Спроекувати структуру програмного засобу обробки результатів тестів;
3. Розробка Web-додатку;
4. Описання методологій по використанню програмного продукту.

## **РОЗДІЛ 1**

### **АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ**

#### **1.1. Online-тестування, його застосування в дистанційному навчанні**

У найпростішому розумінні інтерактивного режиму навчання учень має можливість отримувати (читати, дивитися, слухати) тільки ту інформацію, яку він вибирає для засвоєння з використанням комп'ютера. Ускладнення можливостей і технології здійснення інтерактивного режиму призводить до моделювання навколишнього світу і поведінки

об'єктів в ньому, дозволяючи імітувати реальність.

Звичайно, на сьогоднішній день, в силу багатьох причин, в навчанні використовуються далеко не всі можливості інтерактивного режиму. Зокрема, на думку А. Г: Шмельова, що є найбільшим фахівцем із застосування інтерактивних технологій в освітньому і психологічному тестуванні (система «Телетестинг»), в сучасному Інтернеті переважали не інтерактивні форми підношення освітньої інформації.

Найпростіший інтерактивний режим в локальній мережі і в Інтернеті. Відповідно до класифікації комп'ютерних мереж на локальні і глобальні найпростіший інтерактивний режим, організовується в межах однієї кімнати, або навчального закладу або з використанням Інтернету. Як правило, інтерактивність ґрунтується на асинхронній комунікаційного зв'язку, коли реакція педагога на результати тестування носить відстрочений характер через часу, який необхідний на перевірку тесту в автоматизованому режимі і підрахунок балів учнів за результатами його виконання.

У першому випадку, коли в локальну мережу об'єднано кілька десятків або сотень комп'ютерів, спеціальна програма-реалізатор - інструментальна оболонка - забезпечує видачу заданій online-тесту для всієї групи тестованих, звичайно в індивідуальному й тимчасовому режимі. На екрані кожного комп'ютера з локальної мережі з'являється завдання одного з паралельних варіантів, тесту. При забезпеченні режиму інформаційної безпеки для всієї групи учнів може використовуватися тільки один варіант тесту.

Виконання online-тесту з використанням Інтернету немає принципових відмінностей від випадку застосування локальної мережі при простому рівні інтерактивності без адаптивного режиму, коли всі учні виконують однакові варіанти тесту. Завдання в переважній більшості вимагають від учнів вибору одного або кількох правильних відповідей за допомогою таких відомих діалогових об'єктів, як

«селекторні кнопки» (radio-buttons). Підрахунок тестових балів проводиться шляхом звірення відповідей учнів з ключем і зводиться, найчастіше за все, до простого підсумовування. Передача підсумкового бала по тесту може бути здійснена за допомогою електронної пошти.

Час, витрачений на пред'явлення результату тестування, визначається тривалістю пересилання (зазвичай від декількох секунд до декількох годин) і тим часовим проміжком, який пройде до моменту, коли учень прочитає. В окремих випадках, коли учневі потрібне документальне підтвердження балів, результати тестування можуть бути доставлені offline за допомогою запису на носій інформації. Таким чином, низький рівень інтерактивності цілком придатний для підсумкового тестування поза адаптивного режиму, коли учень повинен працювати без допомоги педагога, а отримання результатів може носити відстрочений у часі характер.

У поточному контролі при дистанційному навчанні зазвичай реалізується середній рівень інтерактивності. Відповідно до можливостей синхронного обміну інформацією в реальному часі за допомогою інтернет-пейджерів учню забезпечуються допомогу і консультації педагога при виконанні завдань коригуючого і діагностичних тестів.

При середньому рівні інтерактивності велика різноманітність набувають форми тестових завдань. Зокрема у школяра з'являється можливість редагування тексту, представленого в завданні, за допомогою введення нових пропозицій або заміни однієї частини тексту на іншу. У завданнях на встановлення правильної послідовності відразу після вибору випробуваним деякого порядку елементів комп'ютер відображає нову послідовність на екрані і т.д. Якщо встановлення синхронного зв'язку не заважають тимчасові пояси, інтерактив негайно забезпечує ефект «педагог поруч», завдяки якому при виконанні завдань поточного контролю учень отримує допомогу, оцінку або підказку



педагога.

Високий рівень інтерактивності в online-тестуванні.

Високий рівень інтерактивності забезпечується в тих випадках, коли при взаємодії з педагогом використовуються звук і відеозображення, що вимагає значних фінансових витрат, але без праці дозволяє ідентифікувати особу учня, що виконує тест в дистанційному контролі.

З педагогічної точки зору високого рівня інтерактивності відповідає адаптивне тестування, що включає розгалужені технології оптимізації складності завдань залежно від відповідей учня на кожне попереднє завдання адаптивного тесту.

## **1.2. Опис предметної області**

В останні роки в сфері освіти спостерігається стрімке збільшення інтересу до автоматизації контролю результатів навчання. Найпопулярнішим видом такого контролю є тестування, засноване на діалозі обчислювальної системи з користувачем.

Перевага web та комп'ютерних систем тестування полягає в тому, що вони дають змогу опитати всіх студентів з різних предметів по всьому матеріалу в однакових умовах за однаковою шкалою оцінок. При цьому значно скорочується час перевірки знань студентів, автоматизується процес обробки результатів тестування, знижується навантаження на викладача

Також варто відзначити, що сьогодні багато компаній і некомерційні організації досить часто стикаються з необхідністю проведення того чи іншого тестування. Наприклад, тестування претендентів на що відкрилася в компанії вакансію, або регулярне психологічне тестування, яке дозволяє виявити "клімат" в офісі і визначити людей, які є джерелами напруги і заважають нормально працювати іншим. А ще ж є тести з техніки безпеки, професійні тести

для підтвердження права займатися тією чи іншою діяльністю, різні атестації і т.д.. Для цього дуже зручно використовувати комп'ютерне тестування. Це дозволяє прискорити проведення опитувань і істотно полегшити обробку результатів.

Метою даної роботи є розробка і реалізація програмного комплексу для забезпечення можливості автоматизованої обробки результатів тестів. Під обробкою результатів мається на увазі підрахунок загальної кількості балів, що нараховуються за кожну правильну відповідь в разі контрольного тестування, або за кожен варіант відповідей, в разі психологічного чи іншого подібного тестування, а також підведення приватної і загальної статистики за результатами. Так, у разі застосування продукту в освітньому середовищі, необхідно мати можливість перегляду статистики по якійсь конкретній групі, курсу або вузу в цілому.

Для обробки результатів найбільш підходить централізована схема організації системи, тобто клієнт-сервер.

Найкращим застосуванням схеми клієнт-сервер буде реалізація її як в локальній, так і в глобальних мережах. Так, в локальній мережі зручно проводити підрахунок відповідей на лабораторному занятті в комп'ютерному класі відразу всієї групи. Варіант з глобальною мережею підійде для віддаленого підрахунку результатів тестів.

Актуальність використання глобальної мережі як середовище передачі даних підтверджує також її повсюдне поширення і доступність.

Статистика підводиться як для конкретного користувача, так і для будь-яких груп користувачів. У разі контрольного тесту в статистиці наводиться відсоткове співвідношення правильних і неправильних відповідей на питання, виходячи з якого викладач робить висновок про оцінку знань.

Взаємодіючими сутностями, в порядку убунання ієрархічної

значущості, є:

1. Користувач (адміністратор);
2. Серверна частина системи;
3. Середовище передачі даних;
4. Клієнтська частина системи;

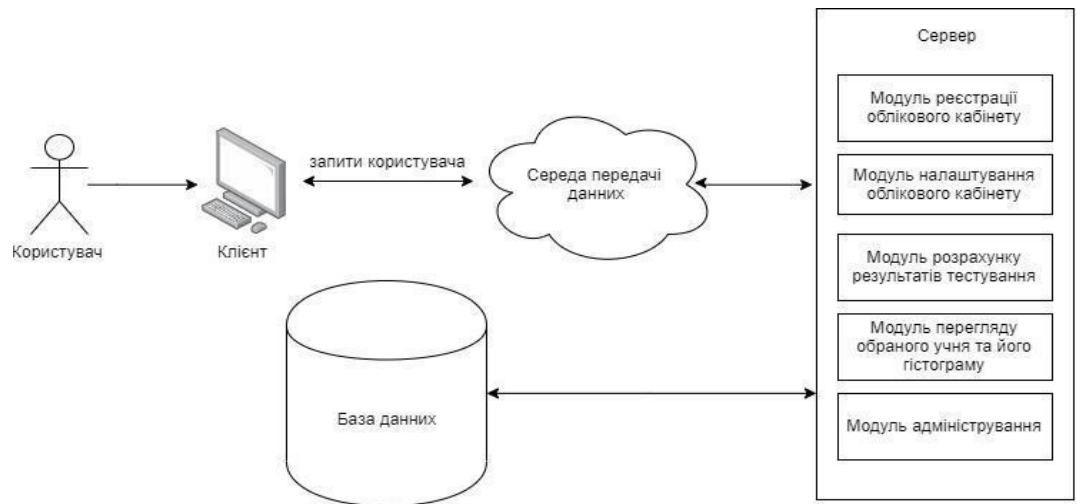


Рис. 1.1 - Схема взаємодії сутностей

Користувач має можливість створити для себе обліковий запис, до якої згодом буде прив'язуватися статистика, вибрати тест для проходження і переглянути свої результати.

Адміністратор, в свою чергу, має можливість редагувати тестові набори, змінювати, тобто налаштовувати, процедуру тестування, вносити зміни в список зареєстрованих користувачів, а також переглядати детальну статистику необхідного виду, отримання якої і є остаточною метою.

### 1.3. Аналіз аналогів і прототипів

Проаналізувавши і з'ясувавши що методів тестування не так вже й багато, з цього має сенс зробити універсальну програму, яка б дала можливість викладачам різних областей створювати свої групи і

заощаджуючи час на проведення та підготовленні тестування і обробку результатів тестування для контролю знань. Зараз ми розглянемо кілька схожих сервісів.

SunRav TestOfficePro

Розробник: SunRav Software (<http://www.sunrav.ru>)

Сайт програми: <http://www.sunrav.ru/srtop/index.php>

Спосіб поширення: shareware

Даний сервіс SunRav TestOfficePro включає в себе такі пакети

які в свою чергу надають такий функціонал як: проведення тестування, створення тестування і обробку результатів тестування. За допомогою даного сервісу можна з легкістю здійснювати тестування та іспити в освітніх установах (вузи, коледжі, школи), а підприємства та організації можуть здійснювати атестацію та сертифікацію своїх співробітників.

Пакет дозволяє створювати тести з предметів шкільної програми, вузівським дисциплін, тести для професійного тестування, психологічні тести тощо

Дані про тестування включаючи абітурієнтів добре захищені методом шифрування який в свою чергу захищений від підробки результатів на виході тестування. На певні частини тесту можна встановити пароль, а саме на: редагування що забезпечує від перегляду XHTML (верстки) для того щоб абітурієнт не зміг переглянути правильні відповіді, на перегляд що забезпечує захист від пробного тестування.

Самі питання і їх відповіді можна редагувати використовуючи текстовий редактор який схожий зі своїми функціями до MS WORD.

У редакторі є все що потрібно для відображення: HTML-документи, таблиці, схеми і формули, а так само аудіо відео файли і будь-які OLE-документи.

У тестах можливе використання п'яти перерахованих вище типів питань.

Створений тест має можливість бути розділеним на стільки тем, скільки буде потрібно. а також при цьому є можливість виставляти для кожного розділу оціночний проміжок яким можна перевірити знання абітурієнта.

Питання можна ставити в хаотичному (випадковому) порядку. Автор тесту має можливість позначити, скільки потрібно питань виставити абітурієнту для тестування. Наприклад, у нас в кожній темі є по 200 питань. Якщо ми вибираємо питання у випадковому порядку тільки 20, то абітурієнти отримають різні масиви питань для однакового тесту. Так само варіанти відповідей можна ставити в хаотичному порядку

Кожен варіант відповіді як і варіант питання може мати свою "важливість". Таким чином нам дається можливість давати користувачеві більше балів за складні питання

Питання може бути закріплений коментарем, який в свою чергу має можливість зберігати в собі відповідь на питання і т.п.

Розглянемо можливі дії на відповіді абітурієнта:

- Запропонувати абітурієнту відповісти на наступне питання;
- Нотифікація абітурієнту про дану відповідь (правильно / неправильно);
- Відобразити документ, додатковий матеріал, інформація або ж пояснення чому відповідь неправильна
- Обмеження за часом для самого тесту або ж для певного питання. Так само можна налаштувати що виділяється час для кожного питання

Сервіс SunRav TestOfficePro має в собі наступні програми:

- t-Maker - під сервіс, редактор тестів який надає можливість створювати і редагувати тести. Включає в себе імпорт

тестів, які були створені в редакторі або в електронній таблиці

- t-Tester - під сервіс для організації тестування;
- t-Admin - під сервіс для редагування групи користувачів і їх обробки результатів тестів. Включає в себе перегляд, роздрук результатів, створення матриць відповідей

#### SunRav TestOfficePro (Технічні дані):

- обсяг питань - необмежена;
- обсяг відповідей - не обмежена
- обсяг користувачів - не обмежена;
- обсяг тестів - не обмежена;
- обсяг тем у тесті - до 512;
- результати і самі тести зберігаються у файлах.

В даний час для даного пакета підготовлена велика кількість тестів. Зокрема, є освітні тести з інформатики, економіки, англійської мови, історії, суспільствознавства.

SunRav TestOfficePro.WEB

Розробник: SunRav Software (<http://www.sunrav.ru>)

Сайт програми: <http://sunrav.ru/webclass.html>

Спосіб поширення: shareware

Вся інформація (тести, розділи тестів, результати і т.д.) міститься в БД і захищено від недозволеного доступу. Якщо користувач не має прав на певне тестування, то він не зможе переглядати інформацію про тестування, не кажучи вже про редагування і видалення.

Сервіс буде корисний для тестування віддалених співробітників, студентів, школярів і т.д. Користувачеві досить браузера тим яким йому більш зручніше користуватись.

Для підготовки експерименту тестування потрібні:

- Web-сервер - в цій іпостасі можна використовувати MS IIS або Apache. Дану програму можна завантажити за посиланням: <http://www.apache.org>;
- MySQL (База даних) - дану програму можна завантажити за посиланням: <http://www.mysql.com>;
- PHP (інтерпретатор) - дану програму можна завантажити за посиланням <http://www.php.net>.

Якщо у користувача немає власного Web-сервера, можна використовувати звичайний хостинг, який надає послуги щодо застосування БД MySQL і PHP.

За допомогою SunRay TestOfficePro.WEB можна:

- організовувати тести в розділи;
- реєструвати імпортувати, редагувати користувачів;
- проводити тестування як зареєстрованих, так і незареєстрованих користувачів;
- переглядати результати тестування по індивідуальному користувачеві;

Keepsoft (Конструктор тестів)

Розробник: Keepsoft

Сайт програми: <http://www.keepsoft.ru/simulator.html>

Спосіб поширення: shareware

«Конструктор тестів» - це програма яка в свою чергу націлена на перевірку знань. Даний сервіс має можливість для проведення

опитування дома та в навчальних закладах. Програма дозволяє використовувати необмежену кількість відповідей, тем і питань.

Програма має такі можливості:

- питання мають можливість мати аудіофайли (формати wav; rmi; mid.),
- картинки (формати jpg, wmf, ico, bmp, emf), відеокліпи (формат avi);
- підтримуються питання всіх п'яти перерахованих вище форматів;
- роздрук та збереження файлу з даними результатів на накопичувачі;
- імпорт експорт відповідей який підтримує перетворення в різні формати файлів
- тестування на одному комп'ютері декількох абітурієнтів. Для абітурієнтів створюється сторінка користувача;
- поділ на групи і установка прав на певні групи людей і на окремих користувачів;
- питання в хаотичному (випадковому);
- "Вага" для кожного питання;
- на кожне питання обмеження за часом;
- можливість "заморозити" тестування для подальшого продовження в зручний час;
- виставлення оцінки після закінчення тестування. Система оцінок налаштовується в «Редакторі». Шкалу оцінок можна налаштувати від 2 до 100-балів;

Keepsoft (Конструктор тестів) надає доступ до скачування демо версії програми яка доступна на офіційному сайті, після закінчення



пробного періоду (30 днів) вам потрібно буде зареєструватися і сплатити вартість програми.

Конструктор тестів «Техносервіс плюс»

Розробник: ТОВ «Техносервіс плюс»

Сайт програми: <http://tsplus.narod.ru/test.html>

Спосіб поширення: замовлення по e-mail

Програма призначена для створення тестів, іспитів, електронних підручників. Для розробки тесту або підручника користувачеві не обов'язково мати навички програмування - програма адаптована і призначена в першу чергу для того щоб користувачеві було легше перевіряти знання певних груп людей. Серед конкурентів дана програма виділяється тим що у неї є можливість додавати матеріали або якусь додаткову інформацію під час тестування.

Дана програма має в собі такі модулі:

- створення, редагування тестів, закріплення різних матеріалів в певному тесті. У конструкторі передбачена можливість використання текстового формату, формул, звукозапису, відеороликів і зображень;
- підручник - він створюється в конструкторі має можливість зберегти в собі безмежну кількість модулів, сторінок і параграфів;
- аналіз, який ще має можливість бути прирівняний до іспиту, тому що ліквідує можливість користуватися підручником під час перевірки знань, як при закріпленні матеріалу. В даному модулі створюються нові сегменти і питання, а ще вказуються відповіді на запитання. При цьому у властивостях питання потрібно показати чисельність балів, що нараховуються за вірну відповідь на нього.

Програма дозволяє:

- автономно робити дослідження для контролю знань студентів. Розробка тесту автоматична і виконується Моторно, враховує ймовірність впровадження НЕ лише тільки слова і малюнків, а й звукозапису, відео, формул, графіків, важких графічних об'єктів і ін .;
- з легкістю імпортувати в конструктор матеріали, за якими побудовані поточні електронні підручники;
- проводити перевірку учнів з метою виявлення їх рівня знань під час вступу до установи (при переході в наступний клас (курс));
- виробляти внутрішні планові / позапланові перевірки знань згодом проходження свіжого матеріалу для оцінки засвоєння матеріалу;
- створювати електронні підручники, інструкції, методичні матеріали;
- в режимі навчання виробляти опитування учня за обраними питань; в разі не вірної відповіді програма вкаже на помилку і порекомендує ще раз прочитати відповідний розділ підручника.

У згорток програм «Конструктор тестів» входять:

- модуль створення досліджень і підручників;
- модуль тестування;
- модуль навчання;
- модуль для перегляду підручників.

Обмежень за кількістю комп'ютерів, на які встановлюється програма, немає.

## 1.4. Перелік завдань підлягають вирішенню в процесі розробки

В освітньому процесі досить велику роль Грають контроль і оцінка знань. Вони в один і той же час виконує функції засвоєння і практичного! Використання знань, які були отриманий студентами і учнями в ході навчального процесу. В даний момент можливо сконцентрувати увагу на 2 типу систем контролю знань:

- спрямовані на проходження досліджень в письмовій або ж усній формі з подальшою перевіркою педагогом. Окремим випадки такого випробування вважається класичним іспит;
- системи комп'ютерного випробування з вірно підібраними тестами;

Системи першого типу є вже довготривалий час і мають ряд дефектів:

- неможливість автоматичної обробки підсумків тестування;
- величезне чисельність рутинної роботи, яка лягає на плечі призвідників тестування;
- мала швидкість обробки результатів;
- ймовірна не об'єктивність оцінки тестування;

Ухвалення укладений в залишкову оцінку зазвичай буквально Абсолютно покладається на педагога, незважаючи на Імовірність вибіркової автоматизації цього процесу. Широке впровадження інструментальних засобів, що дозволяють автоматизувати процес перевірки знань і навичок, допомогло б вирішити частину проблем системи вищої освіти.

Цим чином, автоматизація процесу прийняття висновків в галузі

випробування знань і здібностей в вищій освіті дозволить вивільнити час викладача для творчої діяльності і скоротити витрати вузів.

Найчастіше автоматизація прийняття рішень в області оцінки знань гарантується способи випробування, і найбільшого поширення набули комп'ютерне випробування.

Між перевагами комп'ютерних систем випробування можна відзначити:

- автоматизацію обробки результатів;
- звільнення викладача від проведення традиційних іспитів;
- забезпечення об'єктивності контролю знань; - підвищення оперативності тестування;
- можливість централізованого аналізу якості підготовки тестують по великому колу різних питань;
- зменшення витрат на організацію і проведення тестування;

Створення систем тестування, що використовують в якості середовища передачі інформації глобальну мережу Internet, є крайнім перспективною та актуальною науковою задачею, так як територіальна неухважність вищих навчальних закладів та міжнародна інтеграція навчального процесу призводять до того, що будь-які локальні моделі системи тестування виявляються малоефективними.

Цим чином, актуальність розробки обґрунтована:

- 1) домаганнями підвищення якості вивчення студентів;
- 2) важливим підйомом навантаження на викладацький склад;
- 3) ймовірністю для вдосконалення існуючих ансамблів.

Мета даної роботи полягає в підвищенні ефективності перевірки знань студентів технічних та інших вузів. Поставлена мета досягається за допомогою моделей, алгоритмів і системи автоматизованого тестування знань.

## **РОЗДІЛ 2**

### **ПРОЕКТУВАННЯ WEB-ЗАСТОСУНКУ**

#### **2.1. Проектування структурної схеми системи**

Згідно з технічним завданням програмний ансамбль зобов'язаний гарантувати належні здібності:

- створення і редагування тестових наборів, налаштування обробки результатів тестів;
- адміністрування користувачів і груп користувачів, що працюють з даними юзерів, перегляд підсумків перевірки знань користувачів;
- створення облікового запису тесту користувача є метою подальшого проходження процедури тестування і перегляду результатів сеансу тестування

Таким чином, аналізуючи перелік необхідних можливостей, доцільно провести логічну декомпозицію програмного комплексу на

окремі функціонально завершені програмні модулі:

1. Модуль авторизації
2. Модуль адміністрування і виведення звітів. Створення, редагування і налаштування обробки тестів, налаштування груп та інше - модуль викладача;

Кожен і описаних вище модулів в свою чергу складається з підмодулей, що відповідають за вирішення конкретного завдання.

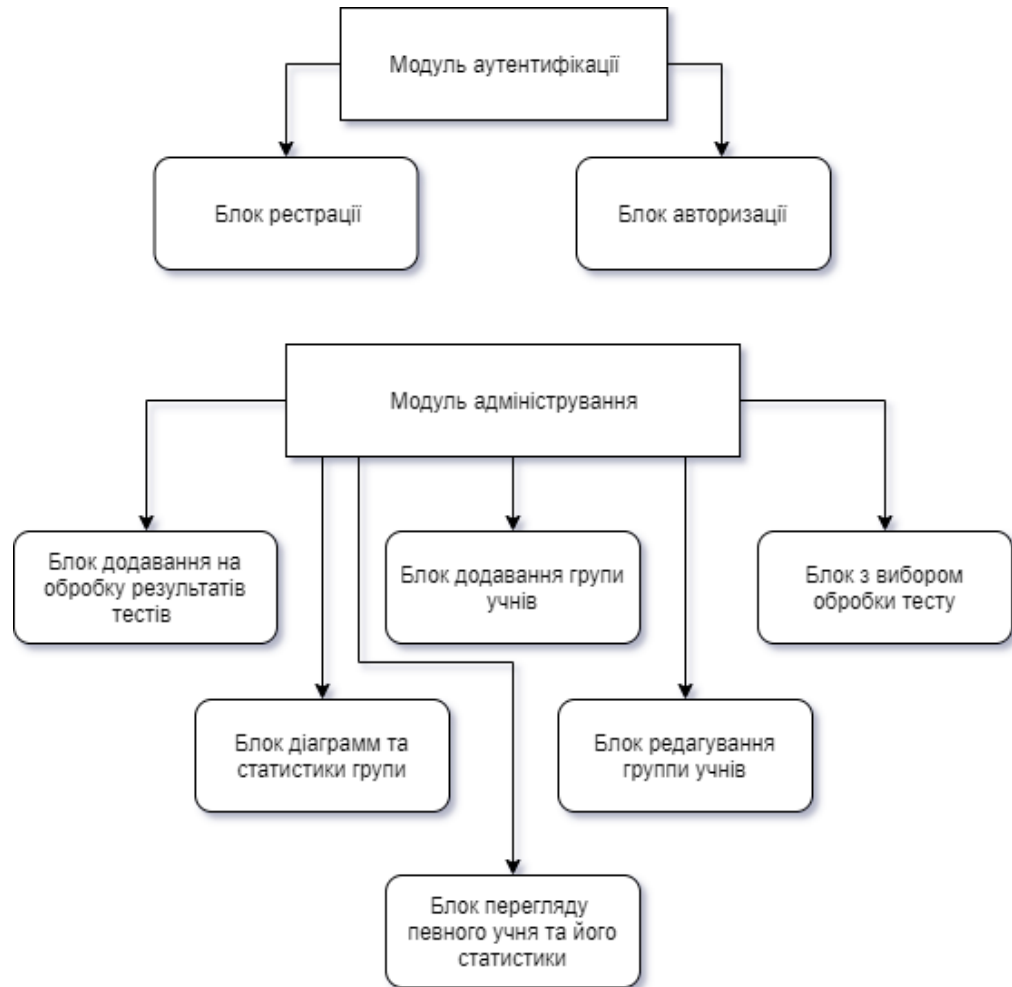
Модуль аутентифікації:

- блок авторизації користувачів. Тут користувач вводить ім'я свого облікового запису з відповідними правами доступу і пароль. У разі якщо введені некоректні дані, доступ до модуля повинен бути закритий.
- блок реєстрації користувачів. Щоб користуватись системою необхідно ввести свою електронну адресу та підтвердити пароль. У разі якщо введені некоректні дані, система не дозволить користувачу зареєструвати особистий кабінет.

Модуль адміністрування складається з наступних функціональних частин:

- блок додавання та обробки результатів тестів
- блок додавання групи учнів
- блок з вибором обробки тестів
- блок діаграм та статистики групи. Дозволяє виводити звіти різноманітного виду як по окремо взятому користувачеві з можливістю фільтрації виведених записів за діапазоном дат, набраними балами і ін.
- блок перегляду певного учня та його статистики

Грунтуючись на описаній вище декомпозиції програмного комплексу на модулі, була розроблена структурна схема комплексу, представлена на рис 2.1



Ри

с 2.1 - Структурна схема програмного комплексу

Як видно з рисунка складові комплекс програмні модулі не мають функціональних зв'язків, завдяки чому спрощується розробка і супровід кінцевого продукту.

## 2.2 Проектування баз даних

Для забезпечення необхідної функціональності, що розробляється

програмний комплекс повинен обробляти гігантські розміри даних, таких як тестові набори даних, дані про групи, користувачів та підсумки проходження досліджень користувачами, всілякі казенні дані. З цієї основи до середовища заощадження даних ставлять високі запити: забезпечені єдності і несуперечливості даних, що зберігаються висока швидкість обробки даних, висока рівень безпеки даних від посягань з боку зловмисників, підтримка багатокористувацького доступу до даних з функціями дозволені або виключені конфліктних ситуацій.

Всім описаним вимогам відповідають реляційні бази даних. Як вмотивованою системою управління базою даних (СКБД) обраний SQLite, який повністю забезпечує виконання поставлених вище вимоги до середовища зберігання даних. Крім цього обраний програмний продукт має досить зручними засобами адміністрування баз даних, що істотно спрощує розробку програмного комплексу в цілому.

Незважаючи на те, що проектувана база даних складається з взаємозв'язаних сутностей, в ній можна виділити дві основні підгрупи сутностей. До першої підгрупи віднесемо суті, описують сховища даних про тести, питаннях і відповідях до цих питань. До другої - сутності користувачів, та користувачів. На рисунку 2.2 представлена ER-модель розробляється бази даних.



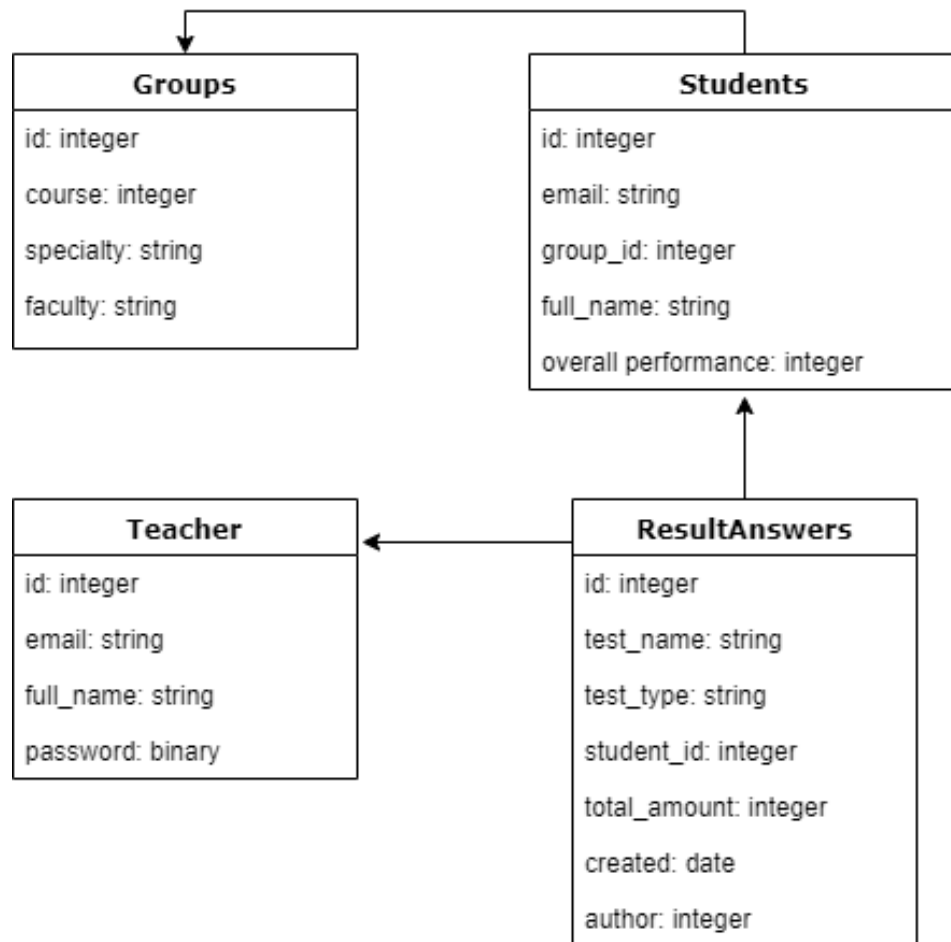


Рис 2.2 - ER-модель бази даних

Сутності та їх атрибути які входять в базу даних:

Сутність Groups. Необхідно для зв'язку студентів, відображення та інш.

Атрибути:

- id. Тип integer. Первинний ключ
- course. Тип integer.
- specialty. Тип string.
- faculty. Тип string.

Сутність Students.

Атрибути:

- id. Тип integer. Первинний ключ
- email. Тип string. Поштова адреса студента
- group\_id. Тип integer. Зберігається особистий номер із суті Group. Визначає аксесуар учня до групи.
- full\_name. Тип string. Ім'я користувача в форматі ПІБ
- overall\_performace. Тип integer. Рівень загальної успішності студента

Сутність ResultAnswers. Зберігає в собі оброблені дані за певним тестом.

Атрибути:

- id. Тип integer. Первинний ключ
- test\_name. Тип string. Зберігає в собі назву тесту
- test\_type. Тип string. Зберігає в собі тип тесту
- student\_id. Тип integer. Зберігається ідентифікатор із сутності Student. Визначає приналежність результатів тесту до студента
- total\_amount. Тип integer. Загальна сума набраних балів у тесті
- created. Тип date. Дата створення
- author. Тип string. Автор. Ким були проведені результати

Сутність Teacher. Зберігає в собі дані користувача

Атрибути:

- id. Тип integer. Первинний ключ
- email. Тип string. Поштова адреса користувача
- full\_name. Тип string. Ім'я користувача в форматі ПІБ

- password. Тип binary. Пароль користувача. Зберігається в закодованому вигляді. Алгоритм кодування - md5.

Підтримка цілісності і несуперечності даних здійснюється засобами цільової СУБД MS SQLite. При виникненні конфліктної ситуації СУБД генерує відповідне повідомлення, яке автоматично буде трансльовано в клієнтську програму і показано користувачеві. Викликала виняткову ситуацію транзакція не буде виконано, а набір даних буде повернений в попереднє несуперечливе стан.

Резервне копіювання даних, налаштування та інші адміністративні функції виконуються засобами адміністрування, такими, як SQL Server Management Studio та SQL Server Business Intelligence Development Studio.

Наявність описаних механізмів підтримки цілісності і несуперечності даних, а так само засобів адміністрування бази даних, робить обґрунтованим використання SQLite в якості цільової СУБД.

### **2.3 Вимоги до систем передавання інформації**

Основний обсяг даних, що пересилаються складають запити клієнтського модуля на вибірку питань, що відносяться до поточного тесту, і результат їх виконання. Пакет відповіді на запит з формулюванням питання в текстовому вигляді з елементами форматування тексту має розмір 2-8 Кб. Розмір пакетів багато в чому залежить від вмісту формулювання питання. Так використання різних мультимедійних об'єктів в формулюванні питання може істотно збільшити час передачі питання в віддалений клієнтський модуль, якщо останній підключений до бази посередством низької модемного з'єднання. Грамотне планування структури і змісту питань дозволяє уникнути подібних проблем.

Для здійснення роботи клієнтського модуля досить модемного

з'єднання на швидкості 33600 Кбіт / сек., Однак, використання в питанні мультимедійних файлів великих розмірів, може уповільнити видачу чергового питання. З точки зору логіки роботи модуля це не критично, оскільки відлік часу відповіді на питання починається з моменту виведення формулювання питання на екран комп'ютера клієнта.

Для комфортної роботи з усіма модулями комплексу досить використання високошвидкісного з'єднання 64 Кб / сек., Що робить можливим використання комплексу, як в локальних, так і в глобальних мережах.

## **2.4. Вимоги до програми та програмного виробу**

### 1. Функції викладача:

- Створення облікового запису для подальшого застосування її в системі;
- Можливість вибору типу обробки тесту;
- Можливість вибору порогу проходження тесту;
- Перегляд гістограму за певним тестом;
- Перегляд результатів одного учня;

### 2. Функції адміністратора:

- Створення, редагування і видалення користувачів і груп;
- Перегляд і друк результатів тестування

Також були наведені вимоги до структури системи. Так, згідно з технічним завданням, система повинна складатися з:

- модуля налаштування облікового кабінету;
- модуля адміністрування;
- модуля розрахунку результатів тестування;

- модуля реєстрації облікового кабінету;
  - модуля перегляду обраного учня та його гістограму;
- Програмні модулі повинні бути легкі в розумінні (user-friendly interface), тексти програм повинні містити всі необхідні коментарі;
- Програмна система повинна включати довідкову інформацію про роботу і підказки користувачеві;
- До складу супроводжуючої документації повинно входити керівництво користувача.

## **РОЗДІЛ 3**

### **РОЗРОБКА WEB-ЗАСТОСУНКУ**

#### **3.1 Розробка і опис робочих алгоритмів**

Оцінювання відбуватиметься від поставлених учителем рівнів проходження (Приклад: E - 30 балів, D - 50 балів, C - 70 балів, B - 80 балів, A - 90 і вище балів). Так само буде можливість обробляти групу

відповідей учня при вказівці типу тесту.

З причини того, що модулі програмного комплексу повинні працювати в багатокористувацької середовищі, алгоритми їх роботи повинні мати низьку ресурсомісткість і малим часом виконання. Необхідно відзначити, що в модулі редагування тестів і модулі адміністрування наявність будь-яких складних обчислювальних алгоритмів НЕ передбачається. По суті, ці модулі є засобами перегляду і редагування вмісту бази даних. Виникаючі в процесі роботи цих модулів затримки обумовлені лише затримкою, пов'язаної з пересилкою результатів запиту від СУБД до програмного модулю, яка в свою чергу може змінюватися в залежності від умов експлуатації модулів: параметрів сервера бази даних і якості лінії зв'язку між програмним модулем і сервером бази даних .

З метою приблизної оцінки затримки при роботі модуля тестування в розрахованому на багато користувачів режимі був проведений експеримент

Тестовий сервер: Intel Core I5 7200u, 8Гб

10 користувачів за 0,013-0,015

20 користувачів за 0,012-0,016

Кількість замірів для обох випадків дорівнює п'яти. Як бачимо, отримані затримки малі, що обумовлено малою вірогідністю виконання сервером одночасно декількох запитів до бази даних. Для цього потрібно було б, щоб запити прийшли одночасно з точністю 0.01с.

### **3.2 Розробка інтерфейсу взаємодії користувача з системою**

Оскільки розроблюваний програмний комплекс в своєму складі має модуль створення, редагування тестів, модуль перегляду результатів тестування і модуль тестування знань, то доцільно поділяти

користувачів, що працюють з різними модулями, різними правами доступу. Це дозволить захистити тестові набори від посягань зловмисників і реалізувати ідеологію чесного тестування, при якій тестується користувач проходить тест, ґрунтуючись тільки на своїх знаннях предмета тестування.

На початковій сторінці нас зустрічає сторінка інфо рисунок 3.1

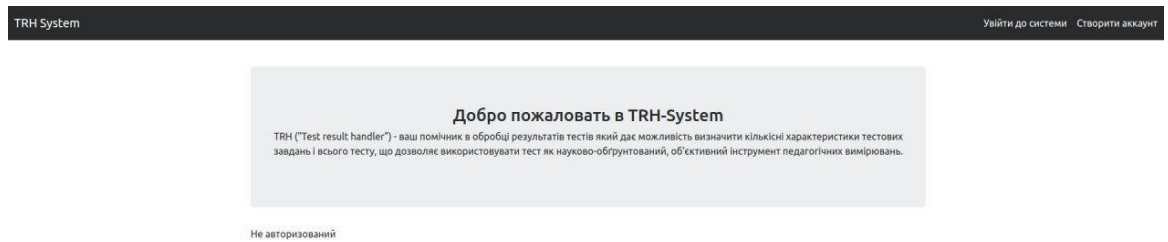


Рис. 3.1 Початкова сторінка

При натисканні на кнопку «Створити аккаунт» ми перейдемо на сторінку реєстрації рис.3.2.

## Створити аккаунт

Ім'я користувача\*

Пароль\*

Повторіть пароль\*

Увійти до системи

Рис. 3.2 Сторінка реєстрації

Після того як ми зареєструвалися і увійдемо в обліковий запис нас буде зустрічати сторінка зі списком результатів тестів (якщо вони є) , рис 3.3:

## Панель управління

Ласкаво просимо, ncstcr

Створити нову обробку тесту

id	название	Редактировать	Удалить
1	Имя теста	Редактировать	Удалить
3	Математика	Редактировать	Удалить

Рис. 3.3 Сторінка зі списком результатів

Якщо у нас є потреба створити нову обробку тестів ми можемо перейти на сторінку створення натиснувши на кнопку "Створити



обробку тесту"

У створенні обробки тексту є редактор де ми можемо вказати: назву тесту, опис тесту, вид тесту. Вибрати тип додавання абітурієнтів. Якщо ми вибрали «Завантажити файл» то нам потрібно щоб файл мав розширення «.excel» та мав колонки з назвою «full\_name» та «rating», а якщо ми вибрали «Додати абітурієнтів вручну» то нам потрібно буде додавати абітурієнта по одному, заповнюючи при цьому поля ПІБ і оцінка рис.3.4, рис.3.5

**Створення обробки тесту**

Назва тесту: \*

Опис:

Вид теста: \*

Файл з даними абітурієнтів: \*  файл не выбран

Примітка: Файл повинен мати розширення файлу ".excel" та мати колонки з назвою full\_name(ФІО) та rating(оцінка)

Рис. 3.4 Сторінка із створенням обробки тесту

### Створення обробки тесту

**Назва тесту:\***

**Опис:**

**Вид тесту:\***

ФІО	оцінка
test 1	: 1 x
test 2	: 22 x
test 3	: 33 x

Рис. 3.5 Сторінка із створенням обробки тесту

Коли ми натиснемо на "Обробити результати" то нам прийде сповіщення що результати теста успішно обробленні. Тепер після цього ми можемо подивитись результати тестування яке нам вирахував алгоритм рис.3.6.



## Математика

Група 345

Иванов Пётр Степанович	: 67 (B)
Степной Людовик Семнадцатов	: 56 (C)
Нионов Андрей Коралович	: 43 (D)

Рис. 3.6 Сторінка з результатами тестування

Користувачі системи - співробітники та студенти вищих навчальних закладів. Всіх користувачів системи можна, розділити на кілька категорій, приналежність до яких конкретизує вимоги. Однак всі користувачі повинні мати навички роботи з персональним комп'ютером. Виділимо наступні категорії користувачів комплексу:

Адміністратор виконує функції управління групами і користувачами, а так само правами в системі, переглядає і роздруковує звіти з результатами тестування окремих користувачів або груп, встановлює відповідність між розділами тестів і групами. Адміністратор може при необхідності виконувати функції всіх нижченаведених типів користувачів. Адміністратор повинен мати глибокі пізнання в сфері установки, конфігурування і роботи з ОС і програмним забезпеченням, використовуваним системою. Також йому необхідно знати основи адміністрування СУБД взагалі і SQLite зокрема. Бажано, щоб адміністратор мав хоча б загальні пізнання в мовах програмування і опису даних, що використовуються в системі (SQL DDL / DML, PL / SQL, і ін.). Одним із завдань адміністрування є консультація користувачів системи в разі виникнення у них будь-яких питань.

Викладач. Вводить в систему нові тестові завдання і редагує існуючі, встановлює часові межі тестування.

Тестований. Ця категорія користувачів добровільно або примусово перевіряє свої знання за допомогою тестів. Перед початком тестування користувач входить під своїм обліковим записом. Потім він вибирає доступний для своєї групи тест і проходить тестування. По завершенню тестування користувач має можливість ознайомитись з результатами сеансу тестування.

Кожній категорії користувачів відповідає окремий програмний модуль. Доступ до модулів редагування тестів і адміністрування

можливий тільки після аутентифікації користувача. Доступ до модуля тестування можливий без аутентифікації, однак, повноцінна робота з модулем можлива тільки після реєстрації користувача в системі.

Інтерфейс всіх модулів має академічний стиль, що відрізняється мінімальними візуальними ефектами і інтуїтивною зрозумілістю кінцевому користувачеві.

## **ВИСНОВКИ**

В результаті виконання курсової роботи були отримані наступні основні результати і висновки:

- Обґрунтовано доцільність та актуальність розробки системи віддаленого тестування знань.
- Вивчено предметна область питання, розглянуті існуючі рішення проблеми.
- Розглянуто і вивчені існуючі аналоги системи, виявлено їх переваги та недоліки, конкретизовані дані технічного завдання

- Розроблено структуру системи в цілому і окремих модулів зокрема.
- Розроблено та описані робочі алгоритми системи
- Розроблено структуру бази даних, використовуваної системою
- Конкретизовано вимоги до середовища передачі інформації
- Описано технологію обробки вхідної інформації
- Розроблено інтерфейс користувача

Отже, можна стверджувати, що мета даної роботи була повністю виконана. Наразі, існує розроблений Web-застосунок, який вже може задовольняти потреби

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Обробка результатів тестів, реалізація [Web ресурс] – Режим доступу до ресурсу: <https://xreferat.com/33/7260-1-obrabotka-rezul-tatov-psihologicheskikh-testov-lisp-realizaciya.html>.
2. Python [Web ресурс] – Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/Python>.
3. Django [Web ресурс] – Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/Django>.
4. Bootstrap [Web ресурс] – Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/Bootstrap>.
5. SQLite [Web ресурс] – Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/SQLite>.

6. Впровадження комп'ютерного тестування знань [Електронний ресурс] – Режим доступу до ресурсу:  
<http://ir.lib.vntu.edu.ua/bitstream/handle/123456789/4826/211-212.pdf?sequence=1&isAllowed=y>.
7. VueJS [Електронний ресурс] – Режим доступу до ресурсу:  
<https://ru.vuejs.org/v2/guide/index.html>.
8. Evans E. Domain-Driven Design: Tackling Complexity in the Heart of Software / Eric Evans., 2011.
9. Філдінг Р. Т. Архітектурні стилі та дизайн архітектури програмного забезпечення / Р. Т. Філдінг. – Ірвін : каліфорнійський університет, 2000.
10. SQLite Documentation [Web ресурс]. – Режим доступу до ресурсу:  
<https://www.sqlite.org/docs.html>.
11. Gheorghiu G. Python for DevOps / G. Gheorghiu, A. Deza, K. Behrman. – California : O'Reilly Media, 2019. – 506 с.
12. Percival H. Test-Driven Development with Python / Harry Percival. – California: O'Reilly Media, 2016. – 624 с.
13. Уязвимость CSRF. Введение [Електронний ресурс]. – Режим доступу до ресурсу: <https://intsystem.org/security/learn-about-csrf-intro>.
14. HTML/CSS [Web ресурс] // Вікіпедія. – Режим доступу до ресурсу: [https://en.wikibooks.org/wiki/HyperText\\_Markup\\_Language/CSS](https://en.wikibooks.org/wiki/HyperText_Markup_Language/CSS).
15. JavaScript [Web ресурс] // Вікіпедія. – Режим доступу до ресурсу: <https://ru.wikipedia.org/wiki/JavaScript>.
16. Git [Електронний ресурс] // Вікіпедія. – Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/Git>.
17. Pycharm [Електронний ресурс] // Вікіпедія. – Режим доступу до ресурсу: <https://ru.wikipedia.org/wiki/PyCharm>.
18. Документація Django 3.0 на руском языке [Електронний ресурс]. – Режим доступу до ресурсу: <https://django.fun/docs/django/ru/3.0/>.
19. Nield T. Getting Started with SQL: A Hands-On Approach for

- Beginners / Thomas Nield., 2016. – 134 с.
20. DRF Authentication [Электронный ресурс]. – Режим доступа до ресурсу:  
<https://www.django-rest-framework.org/api-guide/authentication/>.
  21. AJAX [Электронный ресурс] // Вікіпедія. – Режим доступа до ресурсу: <https://uk.wikipedia.org/wiki/AJAX>.
  22. DataTables [Электронный ресурс] // Вікіпедія. – Режим доступа до ресурсу: <https://datatables.net>.
  23. Gunicorn - Python WSGI HTTP Server for UNIX [Электронный ресурс]. – Режим доступа до ресурсу: <https://gunicorn.org>.
  24. Getting Started on Heroku with Python [Электронный ресурс]. – Режим доступа до ресурсу:  
<https://devcenter.heroku.com/articles/getting-started-with-python>.
  25. Gheorghiu G. Python for DevOps / G. Gheorghiu, A. Deza, K. Behrman. – California : O'Reilly Media, 2019. – 506 с.
  26. Percival H. Test-Driven Development with Python / Harry Percival. – California: O'Reilly Media, 2016. – 624 с.
  27. Дронов В. А. Django: практика создания Web-сайтов на Python / В. А. Дронов. – Петербург: Профессиональное программирование, 2016. – 528 с.
  28. Уязвимость CSRF. Введение [Электронный ресурс]. – Режим доступа до ресурсу: <https://intsystem.org/security/learn-about-csrf-intro>.
  29. Библиотека Django [Электронный ресурс]. – Режим доступа до ресурсу: <https://github.com/w3prog/GoodTravel/wiki/Библиотека-Django>.
  30. Введение · Django Girls Tutorial [Электронный ресурс]. – Режим доступа до ресурсу: <https://tutorial.djangogirls.org/ru/>.

## ДОДАТКИ

### Додаток А

#### КОДЕКС АКАДЕМІЧНОЇ ДОБРОЧЕСНОСТІ ЗДОБУВАЧА ВИЩОЇ ОСВІТИ ХЕРСОНЬКОГО ДЕРЖАВНОГО УНІВЕРСИТЕТУ

Я, Нестерчук Микола Сергійович, учасник освітнього процесу Херсонського державного університету, **УСВІДОМЛЮЮ**, що академічна доброчесність – це фундаментальна етична цінність усієї академічної спільноти світу.

**ЗАЯВЛЯЮ**, що у своїй освітній і науковій діяльності **ЗОБОВ'ЯЗУЮСЯ**:

– дотримуватися:

- вимог законодавства України та внутрішніх нормативних документів університету, зокрема Статуту Університету;
- принципів та правил академічної доброчесності;
- нульової толерантності до академічного плагіату;
- моральних норм та правил етичної поведінки;
- толерантного ставлення до інших;
- дотримуватися високого рівня культури спілкування;

– надавати згоду на:

- безпосередню перевірку курсових, кваліфікаційних робіт тощо на ознаки наявності академічного плагіату за допомогою спеціалізованих програмних продуктів;
- оброблення, збереження й розміщення кваліфікаційних робіт у відкритому доступі в інституційному репозитарії;
- використання робіт для перевірки на ознаки наявності академічного плагіату в інших роботах виключно з метою виявлення можливих ознак академічного плагіату;

– самостійно виконувати навчальні завдання, завдання поточного й підсумкового контролю результатів навчання;

– надавати достовірну інформацію щодо результатів власної навчальної (наукової, творчої) діяльності, використаних методик досліджень та джерел інформації;

– не використовувати результати досліджень інших авторів без використання покликань на їхню роботу;

– своєю діяльністю сприяти збереженню та примноженню традицій університету, формуванню його позитивного іміджу;

– не чинити правопорушень і не сприяти їхньому скоєнню іншими особами;

– підтримувати атмосферу довіри, взаємної відповідальності та співпраці в освітньому середовищі;

– поважати честь, гідність та особисту недоторканність особи, незважаючи на її стать, вік, матеріальний стан, соціальне становище, расову належність, релігійні й політичні переконання;

– не дискримінувати людей на підставі академічного статусу, а також за національною, расовою, статевою чи іншою належністю;

– відповідально ставитися до своїх обов'язків, вчасно та сумлінно виконувати необхідні навчальні та науково-дослідницькі завдання;

– запобігати виникненню у своїй діяльності конфлікту інтересів, зокрема не використовувати службових і родинних зв'язків з метою отримання нечесної переваги в навчальній, науковій і трудовій діяльності;

– не брати участі в будь-якій діяльності, пов'язаній із обманом, нечесністю, списуванням, фабрикацією;



- не підроблювати документи;
- не поширювати неправдиву та компрометуючу інформацію про інших здобувачів вищої освіти, викладачів і співробітників;
- не отримувати і не пропонувати винагород за несправедливе отримання будь-яких переваг або здійснення впливу на зміну отриманої академічної оцінки ;
- не залякувати й не проявляти агресії та насильства проти інших, сексуальні домагання;
- не завдавати шкоди матеріальним цінностям, матеріально-технічній базі університету та особистій власності інших студентів та/або працівників;
- не використовувати без дозволу ректорату (деканату) символіки університету в заходах, не пов'язаних з діяльністю університету;
- не здійснювати і не заохочувати будь-яких спроб, спрямованих на те, щоб за допомогою нечесних і негідних методів досягати власних корисних цілей;
- не завдавати загрози власному здоров'ю або безпеці іншим студентам та/або працівникам.

**УСВІДОМЛЮЮ**, що відповідно до чинного законодавства у разі недотримання Кодексу академічної доброчесності буду нести академічну та/або інші види відповідальності й до мене можуть бути застосовані заходи дисциплінарного характеру за порушення принципів академічної доброчесності.

09.04.2020  
(дата)

 (підпис)

Микола Нестерчук  
(ім'я, прізвище)