

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ХАРЧОВИХ ТЕХНОЛОГІЙ

Інститут (факультет) Факультет автоматизації і комп'ютерних систем

Кафедра Інформаційних систем

Освітній ступінь бакалавр

Спеціальність 122 «Комп'ютерні науки»

(код і назва)

Освітньо-професійна програма «Комп'ютерні науки»

(назва)

ЗАТВЕРДЖУЮ

Завідувач

кафедри Інформаційних систем

« » _____ 20 року

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧА

Ширшова Івана Олеговича

(прізвище, ім'я, по батькові)

1. Тема Розроблення online тестів для контролю знань з дисциплін в режимі Дистанційного навчання

керівник роботи Горлова Тетяна Михайлівна к.т.н., с.н.с.

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від «29» квітня 2021 р. № 248-кв

2. Строк подання здобувачем роботи: 28.05.2021 р.

3. Вихідні дані до роботи: Форма проведення контролю знань, перелік навчальних дисциплін, перелік академічних груп

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити):

Системний аналіз предметної області, функціональні моделі «Організація навчального процесу в режимі дистанційного навчання», моделі бази даних, технічне завдання на розробку системи, опис комплексу задач автоматизації, висновки.

5. Перелік графічного матеріалу:

Функціональні моделі «AS-IS» та «TO-BE», логічна та фізична модель бази даних, діаграма бази даних в середовищі СУБД, знімки інтерфесу системи, фрагменти програмного коду

6. Консультанти розділів роботи:

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1.	Горлова Т.М., доцент	30.04.21	01.05.21
2.	Горлова Т.М., доцент	30.04.21	01.05.21
3.	Горлова Т.М., доцент	30.04.21	01.05.21
4.	Горлова Т.М., доцент	30.04.21	01.05.21

7. Дата видачі завдання: 30.04.2021

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів виконання кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Перший етап – системний аналіз процесу контроль знань студентів	03.05-07.05	Виконано
2	Другий етап – розробка технічного завдання на проектування	08.05-09.05	Виконано
3	Третій етап – реалізація задач автоматизації процесу контролю знань	10.05-21.05	Виконано
4	Четвертий етап – опис охорони праці	22.05-24.05	Виконано
5	П'ятий етап – оформлення пояснювальної записки та створення презентації	25.05-28.05	Виконано

Здобувач

Керівник роботи

_____ (підпис)

_____ (підпис)

Ширшов І. О.

_____ (прізвище та ініціали)

Горлова Т. М.

_____ (прізвище та ініціали)

АНОТАЦІЯ

Метою даної кваліфікаційної роботи є розробка online тестів для контролю знань з дисциплін в режимі дистанційного навчання. Програмний засіб дозволить забезпечити комфортні умови для проведення тестування здобувачів вищої освіти в умовах дистанційного навчання.

Для реалізації інформаційної системи розроблено функціональні схеми організації навчального процесу в режимі дистанційного навчання, структуру бази даних.

Об'єктом дослідження є діяльність відділу дистанційного навчання університету.

Предметом дослідження є процес організації проведення контролю знань у тестовій формі.

Бакалаврська робота містить 98 сторінок, 24 таблиці, 58 рисунків, 9 додатків та 10 літературних джерел.

КЛЮЧОВІ СЛОВА: КОНТРОЛЬ ЗНАНЬ, ДИСТАНЦІЙНЕ НАВЧАННЯ, БАЗА ДАНИХ, ІНФОРМАЦІЙНА СИСТЕМА.

ANNOTATION

The central purpose of this qualification work is to develop online tests to control knowledge of disciplines in distance learning. The software will provide comfortable conditions for testing higher education in distance learning.

To implement the information system, functional schemes of the organization of the educational process in the mode of distance learning, the structure of the database have been developed.

The object of research is the activity of the distance learning department of the university.

The subject of the study is the process of organizing the control of knowledge in test form.

The bachelor's thesis contains 98 pages, 24 tables, 58 figures, 9 appendices and 10 literature sources.

KEY WORDS: KNOWLEDGE CONTROL, DISTANCE LEARNING,
DATABASE, INFORMATION SYSTEM.

ЗМІСТ

ВСТУП	8
РОЗДІЛ 1. СИСТЕМНИЙ АНАЛІЗ ОБ'ЄКТА ДОСЛІДЖЕННЯ ТА ВИЯВЛЕННЯ ЗАДАЧ АВТОМАТИЗАЦІЇ.....	9
1.1. Загальна характеристика Національного університету харчових технологій.....	9
1.2. Організаційна структура Національного університету харчових технологій, роль і взаємодія підрозділів.....	10
1.2.1. Загальна схема організаційної структури	10
1.2.2. Структура відділу дистанційного навчання.....	15
1.3. Аналіз нинішнього стану комп'ютеризації відділу дистанційного навчання	17
1.4. Розроблення функціональної моделі та аналіз існуючих бізнес-процесів.....	18
1.4.1. Функціональна модель відділку дистанційного навчання	18
1.4.2. Виявлені проблеми.....	22
1.4.3. Задачі автоматизації.....	24
1.5. Огляд існуючих рішень для розв'язання виявлених проблем.	25
1.5.1. «ClassMarker».....	25
1.5.2. «Online Test Pad»	26
1.5.3. «Let's test»	27
1.5.4. Порівняльна характеристика доступних рішень	27
1.6. Обґрунтування доцільності проектування й розроблення онлайн-тестів для контролю знань з дисциплін в режимі дистанційного навчання.....	29
1.7. Концептуальна модель системи.....	29
РОЗДІЛ 2. ТЕХНІЧНЕ ЗАВДАННЯ НА ПРОЕКТУВАННЯ.....	31
2.1. Загальні положення.....	31
2.2. Призначення і цілі створення системи.....	31
2.3. Характеристика об'єкта автоматизації.....	32
2.4. Вимоги до системи.....	32
2.5. Склад і зміст робіт по створенню системи	42
2.6. Порядок контролю і приймання системи.....	42
2.7. Вимоги до складу і змісту робіт із підготовки до введення системи в дію.	43
2.8. Вимоги до документації.....	43
2.9. Джерело розробки	43
РОЗДІЛ 3. ОПИС КОМПЛЕКСУ ЗАДАЧ АВТОМАТИЗАЦІЇ	44
3.1. Інформаційне забезпечення системи.....	44
3.2. Алгоритмізація та реалізація комплексу задач автоматизації	48
3.2.1. Реалізація зв'язку з базою даних.....	48
3.1.2. Реалізація інтерфейсу користувача.....	56

3.3. Інструкція користувача.....	65
3.4. Технічне та системне забезпечення розробки	75
3.5. Розрахунок економічного ефекту від впровадження системи.	78
РОЗДІЛ 4. ОХОРОНА ПРАЦІ	83
4.1. Служба охорони праці в установі	83
4.2. Мікроклімат	84
4.3. Шум та вібрації.....	85
4.4. Вимоги до освітленості приміщень	86
4.5. Вимоги пожежної безпеки.....	87
4.6. Електробезпека	88
ВИСНОВКИ.....	89
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	90
ДОДАТКИ.....	91

ВСТУП

Етап оцінювання рівня знань студента, незалежно від способу його реалізації, є ключовим в процесі всього навчання. Контроль знань, окрім того, що дозволяє вирішувати різного роду організаторські задачі, наприклад, такі важливі питання як призначення стипендії чи переведення на наступний курс, допомагає викладачам перевірити якість викладеного навчального матеріалу та ефективність різних підходів та способів подачі його студентам. Самим студентам надається можливість самостійної організації навчального процесу у відповідності до графіку та способу проведення зрізів знань. До того ж, вони, на основі аналізу результатів оцінювання, отримують інформацію про ті теми навчальної дисципліни, вивченню яких слід приділяти більше уваги. Систематичний контроль примушує до постійного повторення навчального матеріалу, надає студентам можливість для саморозвитку, розробці способів презентації здобутих знань та навичок.

Традиційно контроль знань студентів проводиться у формі усного опитування, виконання індивідуального завдання на задану тему або типового тестування чи написання розгорнутої відповіді. Все переліченні види передбачають безпосередню взаємодію екзаменатора та студента в реальному часі. В умовах дистанційного навчання організація подібної взаємодії є досить складним завданням, що в свою чергу породжує запит на створення спеціалізованої інформаційної системи контролю знань. Сучасний рівень інформаційних технологій дозволяє створити такі програмні засоби, які можна використовувати для контролю знань студента з усіх предметів, які він вивчає протягом семестру чи півріччя. Причому, нині існують всі необхідні умови для успішної адаптації подібних систем для будь-яких типів пристроїв: від стаціонарних комп'ютерів до мобільних телефонів.

РОЗДІЛ 1. СИСТЕМНИЙ АНАЛІЗ ОБ'ЄКТА ДОСЛІДЖЕННЯ ТА ВИЯВЛЕННЯ ЗАДАЧ АВТОМАТИЗАЦІЇ.

1.1. Загальна характеристика Національного університету харчових технологій

Національний університет харчових технологій (НУХТ) – вищий навчальний заклад інженерно-технологічного профілю, де здійснюється підготовка фахівців трьох освітньо-кваліфікаційних рівнів «бакалавр», «спеціаліст» та «магістр» за двомастами п'ятнадцятьма навчальними програмами для різних напрямлень харчової, фармацевтичної і мікробіологічної промисловості, а також інших галузей агропромислового комплексу та харчового машинобудування України. У ВУЗі передбачено дві форми навчання: денну та заочну.

НУХТ – навчальний заклад четвертого рівня акредитації. Навчально-виховний процес, науково-дослідна та науково-методична робота забезпечується в межах шести факультетів та інститутів: Навчально-науковому інституті харчових технологій, Навчально-науковому інженерно-технічному інституті ім. акад. І. С. Гулого, Навчально-науковому інституті економіки і управління, факультеті біотехнології та екологічного контролю, факультеті автоматизації і комп'ютерних систем, факультеті готельно-ресторанного та туристичного бізнесу. Крім того, до складу комплексу входять два інститути післядипломної освіти, шість філій: дві у Львові та по одній у Смілі, Сумах, Кам'янець-Подільському та Полтаві, а також одинадцять коледжів, які знаходяться у різних областях країни.

В університеті функціонують сорок дві кафедри, які мають двадцять чотири філії та вісім навчально-науково-виробничих комплексів на передових підприємствах, у проектних і науково-дослідних установах. Окрім цього, є три навчально-виробничих центри, редакційно-видавничий центр,

центр інформаційних технологій, який об'єднує факультетські комп'ютерні центри, комп'ютерні класи та п'ятдесят дві локальні мережі.

У навчально-науковому комплексі «Національний університет харчових технологій» працюють близько п'яти тисяч співробітників, серед яких: понад ста двадцяти професорів, докторів наук; близько восьми сотень доцентів, кандидатів наук. Серед них двадцять один академік, шістнадцять лауреатів Державної премії України, заслужені діячі науки і техніки України, заслужені працівники вищої школи, відмінники освіти України.

Фахівців вищої кваліфікації готують в аспірантурі та докторантурі. В університеті функціонують сім спеціалізованих вчених рад із захисту дисертацій за тринадцятьма науковими спеціальностями [6].

1.2. Організаційна структура Національного університету харчових технологій, роль і взаємодія підрозділів

1.2.1. Загальна схема організаційної структури

Організаційна структура університету представлена навчальними підрозділами, науково-технічними центрами, відділами, службами і т. д., що створюються в цілях забезпечення освітнього процесу та виконання наукових робіт.

Основним організаційним і навчально-науковим структурним підрозділом Національного університету харчових технологій є факультет (в окремих випадках – навчально-науковий інститут), що об'єднує в собі відповідні кафедри, кількість яких не менше трьох. В свою чергу, кафедра є базовим структурним підрозділом, що проводить освітню і методичну діяльність з відповідних спеціальностей, освітніх дисциплін, а також здійснює наукову діяльність за певним напрямом. Керівником факультету (або інституту), що видає розпорядження та несе відповідальність за результати функціонування підпорядкованого підрозділу, є декан факультету (директор інституту).

До територіально відокремлених відносяться такі типи структурних підрозділів як філія, коледж та інститут післядипломної освіти. Філія університету створюється з метою задоволення потреб регіонального ринку праці та наближення місця навчання здобувачів вищої освіти до їх місця проживання. Філію очолює директор, підпорядкований ректору і який діє на підставі відповідного доручення.

Фаховий коледж – підрозділ, основним напрямом діяльності є освітня діяльність у сфері фахової освіти. Інститут післядипломної освіти створюється з метою проведення освітньої діяльності за програмою підвищення кваліфікації працівників та спеціалістів харчової та переробної промисловості.

До повного переліку навчальних підрозділів відносяться:

- Інститути:
 - Інститут харчових технологій;
 - Інженерно-технічний інститут;
 - Інститут економіки і управління;
- Факультети:
 - Факультет біотехнології та екологічного контролю;
 - Факультет автоматизації і комп'ютерних систем;
 - Факультет готельно-ресторанного та туристичного бізнесу;
- Коледжі:
 - Вінницький фаховий коледж НУХТ;
 - Волинський фаховий коледж НУХТ;
 - Івано-Франківський фаховий коледж ресторанного сервісу і туризму НУХТ;
 - Кам'янець-Подільський фаховий коледж харчової промисловості НУХТ;
 - Київський фаховий коледж ресторанного господарства НУХТ;
 - Львівський фаховий коледж харчової і переробної промисловості НУХТ;

- Львівський фаховий коледж харчових технологій та бізнесу НУХТ;
- Полтавський фаховий коледж НУХТ;
- Свалявський технічний фаховий коледж НУХТ;
- Смілянський технологічний фаховий коледж НУХТ;
- Сумський фаховий коледж НУХТ.
- Філії:
 - Філія Національного університету харчових технологій у м. Львові;
 - Смілянська філія Національного університету харчових технологій;
 - Сумська філія Національного університету харчових технологій;
 - Кам'янець-Подільська філія Національного університету харчових технологій;
 - Львівська філія Національного університету харчових технологій;
 - Полтавська філія Національного університету харчових технологій.
- Інститути післядипломної освіти;
- Підготовче відділення.

Слід зазначити, що безпосереднє управління дальністю університету здійснює ректор. Ця фігура виступає в ролі офіційного представника університету у відносинах з державними органами, зокрема Міністерством освіти України, органами місцевого самоврядування, різними юридичними та фізичними особами.

Ректор університету організовує його діяльність, затверджує його структуру, відповідає за результати його діяльності, видає накази та розпорядження, що є обов'язковими для виконання всіма учасниками освітнього процесу, призначає та звільняє працівників, визначає їх функціональні обов'язки, формує контингент осіб, які навчаються в університеті, та має повноваження на відрахування та поновлення здобувачів вищої освіти. Частина свої повноважень він може делегувати проректорам і керівникам структурних підрозділів. Також ректор входить до складу Вченої ради.

Вчена рада – це колегіальний орган управління, членами якого, окрім вищезгаданого ректора, є перший проректор, проректори, директори навчально-наукових інститутів, декани факультетів, учений секретар, директор науково-технічної бібліотеки, головний бухгалтер, голова профспілкового комітету. Інші члени є виборними представниками з числа наукових, виробничих працівників та педагогів. Ключовими повноваженнями Вченої ради є ухвалення рішень ректора про утворення, розформування та зміну структурних підрозділів, обрання на посади деяких відокремлених підрозділів та можливість відкликання ректора з посади.

Далі в цьому розділі наведений короткий опис основних служб та відділів університету.

Навчально-методичне управління – самостійний структурний підрозділ, завдання якого є удосконалення організації процесу підготовки фахівців відповідно до вимог державних та відомчих стандартів освіти [4]. Воно забезпечує інформаційну підтримку освітньої діяльності університету на основі координації роботи факультетів і кафедр. До підрозділу входять навчально-методичний та навчальний відділи.

Центр моніторингу якості та координації освітньої діяльності університету – підрозділ, що включає в себе відділ адміністрування університету та відокремлених структурних підрозділів НУХТ з питань ЄДЕБО, відділ аналітично-інформаційного моніторингу освітньої діяльності, відділ координації освітньої діяльності університету та навчального комплексу НУХТ, відділ рекламно-виставкової та інформаційної діяльності, відділ працевлаштування студентів і випускників, підготовче відділення.

Центр метрології та стандартизації – підрозділ, що здійснює державний нагляд за метрологічним забезпеченням, розробкою, станом, використанням і ремонтом засобів вимірювання, додержання метрологічних правил, вимог і норм, а також за метрологічною службою університету під керівництвом відомчої метрологічної служби Міністерства освіти і науки України.

Структура центру метрології складається з сектору метрологічного нагляду і ремонту засобів вимірювання та сектору стандартизації і норм контролюю

Редакційно-видавничий центр – підрозділ, що спеціалізується на виданні навчальної, навчально-методичної, наукової, науково-популярної та довідкової літератури та рекламно-виставкової поліграфічної продукції, наданні різноманітних поліграфічних послуг. До складу редакційно-видавничого центру входить друкарсько-виробничий сектор, сектор реалізації та маркетингової діяльності, сектор комп'ютерного набору та макетування і редакційно-видавничий сектор.

Центр інформаційних технологій – підрозділ, що займається забезпеченням факультетів, кафедр та інших структурних підрозділів і служб Університету технологічною платформою інформатизації навчального процесу на базі інформаційної комп'ютерної мережі університету та комп'ютерних мультимедійних систем, вирішенню задач внутрішньо-університетської інформатизації, роботи в галузі забезпечення інформаційними технологіями і ресурсами навчального процесу, наукової і адміністративно-господарської діяльності університету, міжвузівської діяльності в галузі інформаційних технологій і ресурсів, міжнародної діяльності в галузі інформатизації. До складу центру інформаційних технологій входять відділ дистанційного навчання, відділ мережевих технологій та експлуатації інформаційних систем, відділ технічного супроводу комп'ютерної техніки [4].

Серед інших структурних підрозділів представлені такі відділи як:

- Відділ міжнародних зв'язків;
- Відділ докторантури і аспірантури;
- Відділ наукової роботи студентів;
- Юридичний відділ;
- Відділ кадрів;
- Відділ документального забезпечення;

- Архів;
- Музей;
- Експлуатаційно-технічний відділ;
- Відділ матеріального та технічного постачання кафедр та структурних підрозділів;
- Планово-фінансовий відділ;
- Бухгалтерія;
- Відділ інтелектуальної власності;
- Відділ охорони праці і техніки безпеки;
- Відділ майнових питань навчального комплексу;
- Відділ ліцензування та акредитації.

1.2.2. Структура відділу дистанційного навчання

Відділ дистанційного навчання існує як окремий структурний підрозділ Центру інформаційних технологій. Відділ працює згідно з «Положенням про використання технологій дистанційного навчання в Національному університеті харчових технологій» (далі – Положення).

Завданням відділу дистанційного навчання є визначення і закріплення принципів організації і функціонування системи дистанційного навчання; формування матеріально-технічної бази системи дистанційного навчання; формування системи дистанційного навчання і впровадження технологій дистанційного навчання для підвищення якості освіти і розширення спектру освітніх послуг, що надаються ВНЗ; організація системи підготовки і підвищення кваліфікації науково-педагогічних працівників університету.

Відділ дистанційного навчання займається підготовкою та підтримкою баз даних інформаційно-освітніх ресурсів, необхідних для функціонування системи дистанційного навчання. В його обов'язки також водить планування і організація навчального процесу в системі дистанційного навчання, що має на увазі залучення науково-педагогічних працівників для забезпечення

навчального процесу та розробку змісту документації для забезпечення навчального процесу за дистанційною формою. Відділ також займається навчанням студентів і науково-педагогічних працівників роботі із інформаційними освітніми технологіями. Відділ адаптує навчальні матеріали, що надаються кафедрами, для використання їх в процесі дистанційного навчання, зокрема займається їх публікацією в системі дистанційного навчання.

Відповідно до Положення виділяють наступні ролі учасників дистанційного навчання:

- Асистент – науково-педагогічний працівник, який несе відповідальність за проведення практичних чи лабораторних занять. Асистент зобов'язаний проводити оцінювання здобувачів за результатами виконаних завдань і тестів, однак сам не має права на публікацію та редагування електронних навчальних курсів.

- Викладач - науково-педагогічний працівник, що відповідальний за проведення лекційних, практичних чи лабораторних занять, може виступати керівником курсових чи кваліфікаційних робіт. Викладач може створювати, редагувати та викладати електронні навчальні ресурси, оцінювати здобувачів за результатами виконаних завдань та тестів.

- Тьютер – працівник, що несе відповідальність за дистанційне навчання на кафедрі. Він відслідковує підготовку веб-матеріалів та наповнення електронних навчальних курсів дисциплін. У разі необхідності, тьютер проводить консультацію викладачів.

- Менеджер – відповідальний за дистанційне навчання на факультеті (інституті). Менеджер співпрацює з тьютерами та відслідковує підготовку матеріалів кафедрами. У разі необхідності менеджер надає консультації для тьютерів кафедр.

Відділ дистанційного навчання взаємодіє з кафедрами факультетів (інститутів) з питань надання доступу науково-педагогічним працівникам та

здобувачам освіти доступу до системи дистанційного навчання. Він надає кафедрам шаблони анкет викладача та студента, отримує інформацію у вигляді заповнених бланків анкет, після чого проводить реєстрацію нових користувачів та передачу даних для входу (логін та пароль) назад до кафедр.

1.3. Аналіз нинішнього стану комп'ютеризації відділу дистанційного навчання

На момент написання роботи функціонує сайт дистанційного навчання Національного університету харчових технологій, що виступає у ролі основного і ключового веб-середовища дистанційного навчання. Сайт є основною платформою для розміщення матеріалів електронного навчального курсу, до яких відносяться мультимедійні лекційні матеріали у формі презентацій, аудіо- та відеофайлів, документи планування навчального процесу, бібліографії, практичні завдання та методичні рекомендації до них, текстові видання у електронному форматі, що представлені як у вигляді файлів веб-публікацій, так і звичайних текстових документах, призначених для перегляду з використанням програм із досить розповсюдженого офісного пакету Microsoft Office та його аналогів.

Заявлені віртуальні лабораторні роботи, що подаються у вигляді відносно стислої викладки завдання з прикріпленими посиланнями на відповідні розділи лабораторних практикумів; захист їх проходить шляхом завантаження файлу звіту або заповнення відповідної форми на сайті дистанційного навчання. В деяких випадках присутня інтерактивна частина, що реалізована з використанням нині застарілої технології Adobe Flash. На жаль, дана технологія є не тільки морально застарілою, але й офіційно більше не підтримується з кінця 2020-го року.

Поточний та підсумковий контроль знань студентів передбачено проводити трьома можливими способами:

- усне опитування в онлайн-режимі із застосуванням сторонніх телекомунікаційних технологій (в зв'язку з тим, що такі не реалізовані на платформі дистанційного навчання);
- виконання індивідуального завдання або контрольної роботи, що реалізовано аналогічно до захисту лабораторних та практичних робіт;
- виконання тестів із застосуванням наявних автоматизованих комплексів.

Автоматизований тестувальний комплекс – система онлайн-тестів, що самостійно оцінює студента на основі порівняння кількості правильних відповідей до загальної кількості запитань. Система тестів пов'язана із системою подій сайту, що дозволяє встановити терміни виконання завдання.

До характерних особливостей системи тестування можна віднести наступні:

- можливість встановлення часових рамок виконання одного тесту;
- випадковий порядок відображення запитань для кожної нової спроби;
- можливість встановлення доступу студента до перегляду результатів тестування;
- можливість встановлення кількості дозволених спроб на проходження тесту;
- можливість встановлення методу оцінювання;
- одночасне відображення на сторінці лише одного (або групи) питання (нь) тесту;
- як наслідок попереднього пункту, посторінкова навігація по завданням тесту.

1.4. Розроблення функціональної моделі та аналіз існуючих бізнес-процесів

1.4.1. Функціональна модель відділку дистанційного навчання

Функціональна «AS-IS» модель організації процесу дистанційного навчання представлена у додатках «А-С».

Опис моделі поданий з точки зору безпосереднього учасника процесу дистанційного навчання (під цей термін підпадають як особи, що навчаються, так і особи, що забезпечують процес навчання).

Більш детальне роз'яснення функціональних блоків та стрілок подано у таблицях 1.1. – 1.2.

Таблиця 1. 1. Опис функціональних блоків IDEF0-діаграми.

№	Назва	Опис
1	Організація навчального процесу в режимі дистанційного навчання	Основний блок, що розміщений на першому рівні декомпозицій функціональної моделі та представляє собою роботу по організації процесу навчання в дистанційному режимі.
2	Читання лекцій в режимі відеоконференції	Блок, що розміщений на другому рівні декомпозиції, представляє собою один із основних видів навчальних занять, що проводяться в синхронному режимі.
3	Самостійне вивчення лекційного та додаткового матеріалу	Блок, що розміщений на другому рівні декомпозиції, представляє процес самостійного ознайомлення з веб-ресурсами навчальної дисципліни.
4	Проведення лабораторних, практичних та семінарських занять	Блок, що розміщений на другому рівні декомпозиції, представляє собою другий із основних видів навчальних занять, де студенти набувають практичних навичок на основі вивченого теоретичного матеріалу.
5	Контроль знань	Блок, що розміщений на другому рівні декомпозиції, представляє собою процес оцінювання рівня засвоєння студентом матеріалу навчальної дисципліни.
6	Поточний контроль знань	Блок, що розміщений на третьому рівні декомпозиції, логічно пов'язаний із блоком «Контроль знань», представляє собою процес оцінювання поточного рівня засвоєння студентом навчального матеріалу дисципліни. Поточний контроль знань проводиться за допомогою: <ul style="list-style-type: none"> - прямого спілкування з викладачем у режимі відеоконференції; - онлайн-тестування; - виконання індивідуального завдання.
7	Підсумковий контроль	Блок, що розміщений на третьому рівні

	знань	<p>декомпозиції, логічно пов'язаний із блоком «Контроль знань», представляє собою процес оцінювання рівня підготовки студента в кінці кожного тематичного блоку (модулю). Підсумковий контроль проводиться за допомогою:</p> <ul style="list-style-type: none"> - усного опитування; - тестування; - виконання контрольних робіт. <p>На результат підсумкового контролю впливає виконання студентом лабораторних, практичних робіт, індивідуальних завдань.</p>
8	Атестація	<p>Блок, що розміщений на третьому рівні декомпозиції, логічно пов'язаний із блоком «Контроль знань», представляє собою процес встановлення відповідності результатів навчання студента вимогам освітньої програми. У відповідності до Закону України «Про вищу освіту» проводиться виключно в синхронному режимі, за умови виникнення надзвичайної ситуації (умова актуальна тільки для дистанційної форми проведення атестації).</p>

Таблиця 1. 2. Опис стрілок IDEF0-діаграми.

№	Назва	Тип	Опис
1	Студент	Вхід	Учасник процесу дистанційного навчання.
2	Студент, що ознайомлений з теоретичним матеріалом	Вихід-Вхід	Студент, що потенційно готовий до проведення практичних занять.
3	Студент, що засвоїв певну частину навчальної програми	Вихід-Вхід	Студент, що потенційно готовий до проведення оцінювання.
4	Веб-ресурси навчальних дисциплін	Вхід	Сукупність електронної навчальної інформації та засобів навчально-методичного характеру, необхідних для засвоєння дисципліни.
5	Закон України «Про вищу освіту»	Управління	Закон, що визначає правові засади організації та

			діяльності вищих навчальних закладів.
6	Положення про використання технологій дистанційного навчання в НУХТ	Управління	Положення, ухвалене Вченою радою НУХТ в 2020-му році, що визначає порядок організації навчального процесу з використанням сучасних інформаційних технологій дистанційного навчання в НУХТ.
7	Навчальний план	Управління	Нормативний документ закладу освіти, за допомогою якого здійснюється організація навчального процесу.
8	Інформаційно-телекомунікаційні засоби	Механізм	В контексті функціональної моделі – це веб-платформи для організації відеоконференцій.
9	Веб-середовище дистанційного навчання	Механізм	В контексті функціональної моделі – сайт дистанційного навчання.
10	Працівники університету	Механізм	Учасники навчального процесу з правом викладати, редагувати навчальні курси, оцінювати студентів за результатами виконаних завдань і тестів.
11	Викладач	Механізм	Працівник університету, відповідальний за проведення лекційних, практичних чи лабораторних занять, є керівником курсових і кваліфікаційних проектів.
12	Асистент	Механізм	Працівник університету, відповідальний за проведення практичних чи лабораторних занять.
13	Лекційний матеріал	Вихід-Вхід	Мультимедійний

			лекційний матеріал, як от презентації, відео- та аудіофайли, що даються студенту для більш детального ознайомлення в ході самостійного вивчення дисципліни.
14	Науково-педагогічний працівник	Механізм	В контексті діаграми – працівник університету, що виступає у ролі лектора.
15	Педагогічний працівник	Механізм	В контексті діаграми – працівник університету, що виступає у ролі консультанта.
16	Навчально-методична література	Управління	В контексті діаграми – література, якою повинен керуватися студент під час лабораторних, практичних або семінарських занять.
17	Автоматизований комплекс тестування	Механізм	В контексті діаграми – система онлайн тестування на сайті дистанційного навчання.
18	Веб-платформа для завантаження виконаного індивідуального завдання (виконаної контрольної роботи)	Механізм	Веб-форма на сайті дистанційного навчання для завантаження текстових документів.
19	Результати проведення навчальних занять у формі звітів	Вихід	Результати проведення лабораторних, практичних, семінарських занять.
20	Результати оцінювання успішності студента	Вихід	Результати виконання тестів, індивідуальних, контрольних робіт.

1.4.2. Виявлені проблеми

Основною проблемою існуючої платформи дистанційного навчання, а точніше системи онлайн-тестування, була і залишається проблема відслідковування діяльності студента під час виконання тестового завдання.

Наразі немає можливості відстежити те, чи користується студент сторонньою допомогою для пошуку правильних відповідей.

Кількість запитань одного тесту фіксована, що створює можливість для студентів заздалегідь підготувати правильні відповіді у тому разі, коли відводиться кілька спроб на проходження тесту.

Також слід звернути увагу на те, як реалізований сам інтерфейс системи тестування (рис. 1.1).

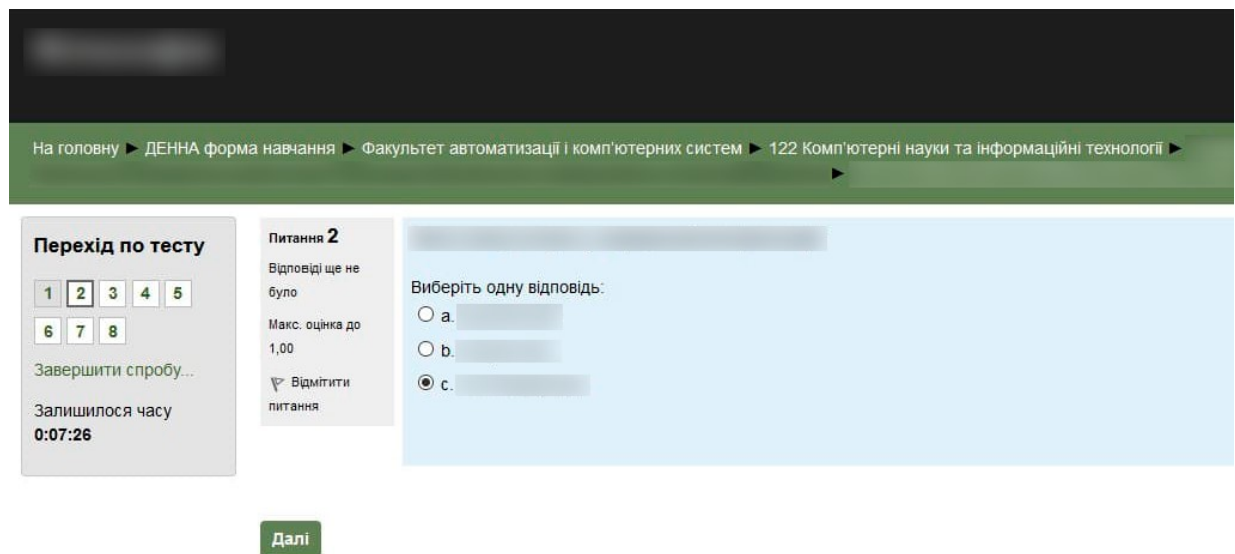


Рисунок 1. 1. Інтерфейс системи онлайн-тестування платформи дистанційного навчання.

Серед виявлених недоліків можна виділити наступні:

- одночасне відображення на сторінці лише одного запитання;
- не адаптованість інтерфейсу сайту до екранів мобільних пристроїв.

Перший недолік провокує появу ситуації, коли при переході між запитаннями, необхідно щоразу заново завантажувати всі графічні елементи сторінки. В залежності від швидкості передачі даних в мережі та конфігурації комп'ютера цей процес може тривати до кількох секунд. При чому, таймер, що відраховує час, який залишився до кінця спроби, продовжує свою роботу.

Другий недолік значно звужує вибір апаратури, яку можна використовувати для доступу до сайту дистанційного навчання. Ще в

середині минулого десятиліття було офіційно встановлено, що більшу частину пристроїв, які використовуються для перегляду сторінок в інтернеті, становлять різного роду портативні пристрої, типу мобільних телефонів, планшетів, наручних годинників, тощо. Однак, на рисунку 1. 2. можна спостерігати, як саме виглядає інтерфейс сайту на екрані мобільного телефону.

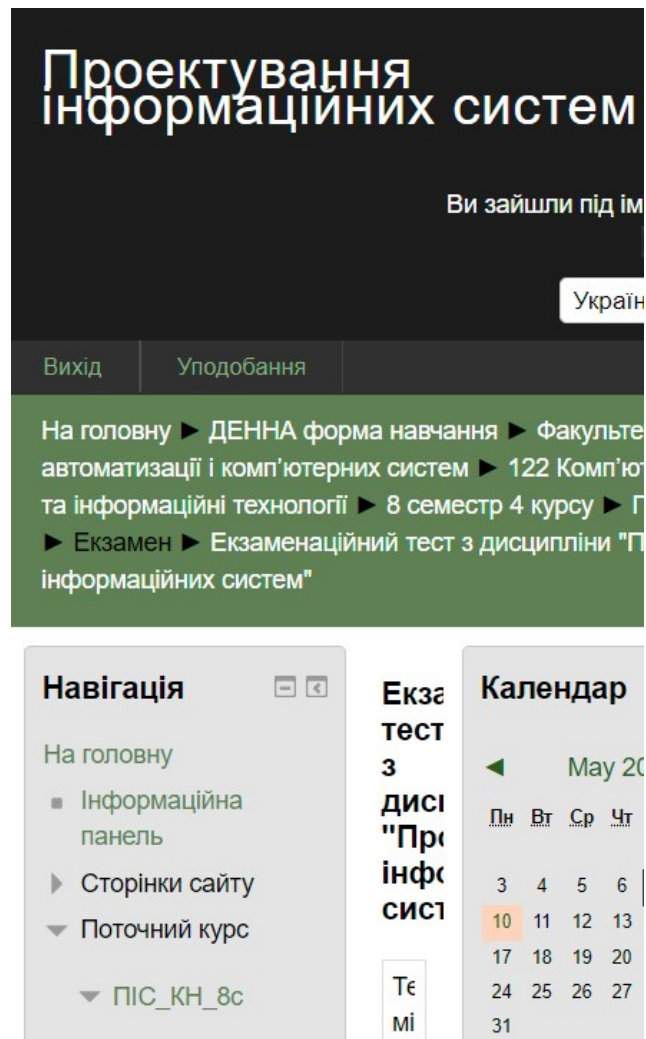


Рисунок 1. 2. Приклад відображення інтерфейсу сайту дистанційного навчання на екрані мобільного телефону.

1.4.3. Задачі автоматизації

Для усунення виявлених проблем пропонується створити нову систему для організації дистанційного контролю знань студентів, яка повинна виконувати наступні задачі:

- надати користувачеві можливість комфортної роботи інформації з будь-якого пристрою, з доступом до мережі Інтернет;
- надавати користувачу можливість самостійної розробки тестів;
- реалізовувати перегляд списку тестів, з можливістю фільтрації по певним параметрам;
- надавати користувачу можливість проходити тестування та переглядати результати цього тестування.

1.5. Огляд існуючих рішень для розв'язання виявлених проблем.

1.5.1. «ClassMarker»

ClassMarker – це професійний конструктор онлайн-тестів для бізнесу або навчання. Використовується у всьому світі для організації опитувань в сфері бізнесу та на підприємствах, тестування перед працевлаштуванням, онлайн-сертифікації, проведення опитувань з питань охорони праці, проведення тестувань в школах і університетах, під час дистанційного навчання, на онлайн-курсах і т. д.

Конструктор відомий своїм простим для засвоєння інтерфейсом. Тут є 7 типів питань, спеціальний банк, куди зберігаються всі створені питання для повторного використання в інших тестах. Для класифікації анкет можна створювати власні категорії. Також можна відкрити доступ до тесту, для групового редагування та використання.

Зібрати відповіді можна за допомогою розміщення HTML-коду тесту на сайті, відправки посилань учасникам. Можна контролювати доступ до тесту, створивши індивідуальні реєстраційні коди або паролі для користувачів. Статистика результатів відображається в особистому кабінеті, можна переглядати відповіді як по тесту в цілому, так і окремо по кожному учаснику, однак вивантажити їх не дозволяється.

Наявна 30-денна пробна версія. Далі необхідно вибрати один з безлічі тарифів, ціни на які варіюються від 20 до 1000 доларів. Для оплати можна використовувати банківські картки.

Плюси: приємний і зручний дизайн, можливість проведення закритих тестувань.

Мінуси: англійська мова інтерфейсу, висока ціна, відсутність можливості експорту результатів тесту.

1.5.2. «Online Test Pad»

Сервіс дозволяє створити як прості тести, так і опитування, кросворди, «діалоги» - тести, оформлені у вигляді спілкування з віртуальним екзаменатором. Також всередині сервісу є система дистанційного навчання і тестування, де можна створювати групи учнів, давати їм уроки і завдання, вести журнал успішності.

При створенні тесту потрібно вибрати його тип: психологічний, особистісний або освітній. У конструкторі 17 типів питань, сервіс дозволяє провести навіть інтерактивний диктант.

Доступ до тесту можна надати за посиланням, його можна вбудувати на сайт за допомогою HTML-коду, розіслати запрошення по електронній пошті або опублікувати для загального доступу.

Послуги сервісу надаються безкоштовно, однак при проходженні тесту учасники бачитимуть рекламу. Її відключення коштує 0,01 долара для кожного проходження тесту.

Плюси: багато типів питань, чат з техпідтримкою в особистому кабінеті, інтуїтивно зрозумілий інтерфейс.

Мінуси: застарілий дизайн тестів, відсутність повного перегляду тесту, проблеми з відображенням на різних типах девайсів.

1.5.3. «Let's test»

Сервіс спеціалізується виключно на онлайн-тестах. Дозволяє налаштувати доступ до тестування по паролю або попередньої реєстрації та контролювати, хто буде його проходити.

Конструктор містить всього шість типів питань, проте самостійно розібратися в функціоналі і інтерфейсі не так просто. За допомогою можна звернутися в службу підтримки і переглянути інтерактивну презентацію роботи сервісу. Відсутня можливість самостійного налаштування дизайну тесту, для цього необхідно оформити заявку.

Сервіс дозволяє завантажувати питання з файлу, надавати доступ до тесту за спеціальним паролем.

Сервіс платний. Є пробний період - 5 днів, а далі необхідно вибрати найбільш підходящий тариф. Ціна мінімального тарифу - 27 доларів на місяць.

Плюси: ізольована система тестування, допомога в налаштуванні тесту.

Мінуси: застарілий дизайн конструктора, неінтуїтивний інтерфейс, немає можливості самостійно налаштувати дизайн тесту, немає функції попереднього перегляду, доступна лише російська мова інтерфейсу.

1.5.4. Порівняльна характеристика доступних рішень

Характеристика онлайн-конструкторів тестів наведена у табл. 1. 3.

Таблиця 1. 3. Порівняльна характеристика сервісів онлайн-тестування.

Характеристика	Сервіс		
	«ClassMarker»	«Online Test Pad»	«Let's test»
Мова інтерфейсу	Англійська	Англійська, російська, українська, білоруська	Російська
Вартість	20-1000 дол./м.	Безкоштовний (реклама)	27 дол./м
Пробна версія (для платних сервісів)	30 днів	-	5 днів

Медіа додатки до запитань	Файли, в тому числі зображення та відео.	Зображення, таблиці, спец. символи, мат. формули, аудіо файли.	Зображення, відео, посилання, таблиці.
Експорт результатів	Доступний тільки для платної версії (відсутній у пробній)	Доступний тільки перегляд статистики на сайті	Доступний у форматі pdf, xls,xlsx, rtf, csv.
Імпорт тестів	Доступний	Відсутній	Імпорт запитань із xls файлів.
Експорт шаблонів	Відсутній	Доступний у форматі html, pdf.	Відсутній
Розміщення тестів на інших сайтах	Посилання на сторінку тесту; інтеграція віджету на сайт	Посилання на сторінку тесту	Посилання на сторінку тесту; інтеграція віджету на сайт
Адаптованість для мобільних пристроїв	Хороша, за умови використання сервісу напряду.	Можлива проява графічних «артефактів» на мобільних телефонах	Відсутня
Інші особливості	Можливість встановити часові обмеження; потужні інструменти аналітики; можливість повторного використання окремих запитань з тестів; можливість встановити обмеження доступу до тесту	Коментарі до тестів; можливість встановити часові обмеження.	Детальна довідникова документація; банк запитань; можливість встановити обмеження доступу до тесту; потужні інструменти аналітики та збору статистики.

1.6. Обґрунтування доцільності проектування й розроблення онлайн-тестів для контролю знань з дисциплін в режимі дистанційного навчання

Очевидно, що готові рішення для проведення онлайн тестування, наявні на українському секторі веб-простору, мають досить широкий вибір інструментів як для організації опитувань, так і для збору та аналізу результатів цих опитувань. Однак у більшості доступних сервісів бізнес-модель розповсюдження – підписка, що передбачає щомісячне вливання коштів за можливість користування повним арсеналом функціональних можливостей. Передбачені пробні періоди не є виходом із ситуації, оскільки призначені лише для ознайомлення з можливостями тієї чи іншої платформи. Безумовно, зустрічаються безкоштовні аналоги, що володіють лише базовим набором для створення тестів, однак їх головним недоліком є наявність реклами, яка потенційно може заважати студенту і відволікати його.

Серед наведених у пункті 1.5.4 аналогів діючої автоматизованої системи тестування сайту дистанційного навчання НУХТу немає тих, які в повній мірі вирішують проблеми, заявлені в пункті 1.4.2. «ClassMarker» хоч і володіє найбільшим переліком якісних переваг, однак є надто дорогим варіантом, щоб пропонувати його в якості заміни. «Let's test» - другий в черзі кандидат, однак досить складний для оперативного освоєння через свій комплексний інтерфейс, який до того ж взагалі не підходить для мобільних пристроїв. «Online Test Pad» - найбільш привабливе рішення, однак доволі незручний у використанні, через наявність реклами, неможливість експорту результатів тестування та проблеми із роботою на мобільних пристроях.

1.7. Концептуальна модель системи

Функціональна «ТО-ВЕ» модель організації процесу дистанційного навчання представлена у додатках «D-F».

Модель розроблена з урахуванням впровадження на об'єкт системи контролю знань у тестовій формі. Модель не відображає та не передбачає

можливості проведення поточного та підсумкового контролю знань з дисциплін в усній формі, з використанням сторонніх інформаційно-телекомунікаційних засобів. Форма проведення атестації, у відповідності до діючого законодавства, не може бути представлена у вигляді тестів, в зв'язку з чим відповідна частина функціональної моделі залишається незмінною.

Детальне роз'яснення нових функціональних блоків та стрілок подане у таблицях 1.4 – 1.5.

Таблиця 1. 4. Опис нових функціональних блоків IDEF0-діаграми.

№	Назва	Опис
1	Розробка тестової форми	Блок, що розміщений на третьому рівні декомпозиції, представляє собою процес розробки тестового бланку із визначеним переліком запитань та можливих варіантів відповідей, з метою подальшого його використання для проведення контролю знань студентів.

Таблиця 1. 5. Опис нових стрілок IDEF0-діаграми.

№	Назва	Тип	Опис
1	Перелік груп студентів	Вхід	Повний список академічних груп студентів, що навчається у закладі вищої освіти.
2	Перелік дисциплін, що викладається	Вхід	Повний перелік навчальних дисциплін, що викладається у закладі вищої освіти.
3	Форма тесту зі списком запитань та варіантами відповідей	Вихід-Вхід	Електронна форма тестового бланку тесту, що може бути використана для проведення поточного або підсумкового контролю знань.
4	Тестова форма проведення контролю знань	Управління	Форма проведення контролю знань у вигляді тестування.
5	Система онлайн-тестів	Механізм	Розроблена система створення та проведення тестів з метою отримати дані про рівень знань студентів.

РОЗДІЛ 2. ТЕХНІЧНЕ ЗАВДАННЯ НА ПРОЕКТУВАННЯ

2.1. Загальні положення

2.1.1. Найменування системи: «Онлайн-тести для контролю знань з дисциплін в режимі дистанційного навчання»

2.1.2. Результати робіт зі створення системи оформлюються згідно з вимогами ДСТУ на відповідні етапи розробки. Порядок оформлення і передачі результатів у даному випадку визначається змістом і календарним планом виконання розробки.

2.1.3. У випадку необхідності на наступних стадіях робіт по створенню системи окремі положення можуть уточнюватися і розвиватися.

2.2. Призначення і цілі створення системи

2.2.1. Призначення системи.

Система призначена для організації процесу контролю знань у режимі дистанційного навчання. Контроль знань проводиться у форматі онлайн-тестів. Система автоматизує процес тестування, збору та підрахунку результатів. Система містить дані про користувачів, академічні групи та навчальні дисципліни; дозволяє створювати електронні бланки тестів, проходити тестування; надає додаткові функції, які дозволяють виконувати зручний пошук за різними властивостями.

2.2.2. Цілі створення системи.

Основною метою створення системи є забезпечення оперативного отримання об'єктивної інформації про поточний рівень знань студентів, автоматизація процесу проведення зрізу знань. Це забезпечить створення умов для комфортного проведення поточного та підсумкового контроль знань з навчальних дисциплін в режимі дистанційного навчання.

2.3. Характеристика об'єкта автоматизації

2.3.1. Короткі відомості про об'єкт автоматизації

Об'єктом автоматизації є діяльність відділу дистанційного навчання Національного університету харчових технологій, а саме той її напрям, що напряму стосується процесу організації контролю знань.

2.4. Вимоги до системи

2.4.1. Вимоги до системи в цілому.

2.4.1.1. Вимоги до структури і функціонування системи.

2.4.1.1.1. Система повинна мати клієнт-серверну архітектуру, що використовує єдину базу даних (надалі — БД).

2.4.1.1.2. Діагностування системи має передбачати виявлення відхилень від нормального процесу розв'язання задач, програмних помилок, забезпечувати користувачів відповідними повідомленнями.

Взаємозв'язок між окремими модулями системи має здійснюватися на інформаційному рівні через загальну БД.

2.4.1.1.3. Розвиток та модернізація системи повинні проводитися шляхом уточнення, розширення та зміни виконуваних функцій, модернізації програмних засобів протягом всього періоду експлуатації. Програмне забезпечення повинно бути розроблене із задіянням таких технологій та мати таку структуру, що забезпечуватимуть простоту модифікації в ході розвитку проекту. Повинна бути передбачена можливість збільшення інформаційних масивів та розмірності задач.

2.4.1.1.4. Функціонування системи має забезпечувати діалогову та мережну обробку даних.

2.4.1.2. Вимоги до чисельності і кваліфікації персоналу.

2.4.1.2.1. Персонал, який використовує дану систему, повинен відповідати наступним вимогам:

- мати навички роботи з браузером та мережею Інтернет;

- дотримуватись технологічних інструкцій при роботі з системою в діалоговому режимі;

2.4.1.2.2. Користувачами системи може виступати персонал відділу дистанційного навчання університету, викладачі та студенти. Вхід у систему повинен здійснюватися із використання логіну та паролю користувача. Функціональні можливості, що надаються користувачеві повинні залежати від його ролі. Передбачається створення трьох функціональних рівнів: адміністратор, викладач та студент.

2.4.1.3. Показники призначення

2.4.1.3.1. Система повинна передбачати можливість масштабування підвищення продуктивності і обсягу оброблюваної інформації без модифікації її програмного забезпечення шляхом модернізації використовуваного комплексу технічних засобів. Можливості масштабування повинні забезпечуватися засобами використовуваного базового програмного забезпечення.

2.4.1.4. Вимоги до надійності

2.4.1.4.1. Система є поліфункціональною та призначена для використання протягом робочого дня. Всі її функції виконуються дискретно.

Система повинна зберігати працездатність і забезпечувати відновлення своїх функцій при виникненні наступних позаштатних ситуацій:

- при збоях в системі електропостачання апаратної частини, що призводять до перезавантаження ОС, відновлення програми має відбуватися після перезапуску ОС і запуску виконуваного файлу системи;
- при помилках в роботі апаратних засобів (крім носіїв даних і програм) відновлення функції системи покладається на ОС;
- при помилках, пов'язаних з програмним забезпеченням (ОС і драйвери пристроїв), відновлення працездатності покладається на ОС.

Для гарантування надійності програмного та інформаційного забезпечення системи необхідно передбачити використання:

- модульного, структурного і об'єктно-орієнтованого програмування;
- засобів контролю вхідної інформації з видачою користувачу попереджувальних повідомлень про виявлені помилки;
- засобів захисту від збоїв, спроб отримання несанкціонованого доступу, помилкових дій користувачів і т. д.;
- резервних дамів БД.

2.4.1.5. Вимоги до безпеки.

Всі зовнішні елементи технічних засобів системи, що знаходяться під напругою, повинні мати захист від випадкового дотику, а самі технічні засоби мати занулення або захисне заземлення відповідно до ГОСТ 12.1.030-81.

Система електроживлення повинна забезпечувати захисне відключення при перевантаженнях і коротких замиканнях в ланцюгах навантаження, а також аварійне ручне відключення.

Загальні вимоги пожежної безпеки повинні відповідати нормам на побутове електрообладнання. У разі загоряння не повинно виділятися отруйних газів і димів. Після зняття електроживлення повинно бути допустимо застосування будь-яких засобів пожежогасіння.

Фактори, що шкідливі впливи на здоров'я з боку всіх елементів системи (в тому числі інфрачервоне, ультрафіолетове, рентгенівське і електромагнітне випромінювання, вібрація, шум, електростатичні поля, ультразвук малої частоти і т. д.), не повинні перевищувати діючих норм.

2.4.1.6. Вимоги до ергономіки та технічної естетики.

Взаємодія користувачів з прикладним програмним забезпеченням, що входить до складу системи має здійснюватися за допомогою візуального графічного інтерфейсу (GUI). Інтерфейс системи повинен бути зрозумілим і зручним, не повинен бути перевантажений графічними елементами.

Навігаційні елементи повинні бути виконані в зручній для користувача формі. Введення-виведення даних системи, прийом керуючих команд і відображення результатів їх виконання повинні виконуватися в інтерактивному режимі. Інтерфейс повинен відповідати сучасним ергономічним вимогам і забезпечувати зручний доступ до основних функцій та операцій системи.

Інтерфейс повинен бути розрахований на використання базових пристроїв введення інформації, що характерні для стаціонарних машин, якщо доступ до системи здійснюється із стаціонарного ПК. Для можливості доступу із портативного пристрою, інтерфейс повинен бути розрахований на введення даних з допомогою тачскріну.

Система повинна забезпечувати коректну обробку аварійних ситуацій, викликаних неправильними діями користувачів, невірним форматом або неприпустимими значеннями вхідних даних. У зазначених випадках система повинна видавати користувачу відповідні повідомлення, після чого повертатися в робочий стан, що передував невірної (неприпустимою) команді або некоректного вводу даних.

Всі написи, а також повідомлення, що видаються користувачеві (крім системних повідомлень) повинні бути на українській мові.

Форми введення інформації повинні проектуватися з урахуванням вимог уніфікації:

- всі елементи користувальницького інтерфейсу повинні бути виконані в єдиному графічному дизайні, з однаковим розташуванням основних елементів керування та навігації;
- для позначення подібних операцій повинні використовуватися подібні графічні значки, кнопки та інші керуючі (навігаційні) елементи;
- зовнішня поведінка подібних елементів інтерфейсу повинно реалізовуватися однаково для однотипних елементів.

2.4.1.7. Вимоги по експлуатації, технічного обслуговування, ремонту і зберігання компонентів системи.

Система повинна бути розрахована на експлуатацію в складі програмно-технічного комплексу замовника і враховувати поділ ІТ інфраструктури замовника на внутрішню і зовнішню. Технічна і фізичний захист апаратних компонентів системи, носіїв даних, безперебійне енергопостачання, резервування ресурсів, поточне обслуговування реалізується технічними та організаційними засобами, передбаченими в ІТ інфраструктурі Замовника.

Для нормальної експлуатації розроблюваної системи має бути забезпечено безперебійне живлення ПЕОМ. При експлуатації система повинна бути забезпечена відповідна стандартам зберігання носіїв і експлуатації ПЕОМ температура і вологість повітря.

Періодичне технічне обслуговування технічних засобів повинно проводитися відповідно до вимог технічної документації виробників, але не рідше одного разу на рік.

Періодичне технічне обслуговування і тестування технічних засобів повинні включати в себе обслуговування і тестування всіх використовуваних засобів, включаючи робочі станції, сервери, кабельні системи та мережеве обладнання, пристрої безперебійного живлення.

В процесі проведення періодичного технічного обслуговування повинні проводитися зовнішній і внутрішній огляд і чистка технічних засобів, перевірка контактних з'єднань, перевірка параметрів налаштувань працездатності технічних засобів і тестування їх взаємодії.

На підставі результатів тестування технічних засобів повинні проводитися аналіз причин виникнення виявлених дефектів і вживати заходів щодо їх ліквідації.

Відновлення працездатності технічних засобів повинно проводитися відповідно до інструкцій розробника і постачальника технічних засобів і документами по відновленню працездатності технічних засобів і

завершуватися проведенням їх тестування. При введенні системи в дослідну експлуатацію повинен бути розроблений план виконання резервного копіювання програмного забезпечення і оброблюваної інформації. Під час експлуатації системи, персонал, відповідальний за експлуатацію системи повинен виконувати розроблений план.

Розміщення приміщень і їх обладнання повинні виключати можливість безконтрольного проникнення в них сторонніх осіб і забезпечувати збереження що знаходяться в цих приміщеннях конфіденційних документів і технічних засобів.

Розміщення обладнання, технічних засобів повинно відповідати вимогам техніки безпеки, санітарним нормам і вимогам пожежної безпеки.

Всі користувачі системи повинні дотримуватися правил експлуатації електронної обчислювальної техніки.

Кваліфікація персоналу та його підготовка повинні відповідати технічній документації.

2.4.1.8. Вимоги до захисту інформації від несанкціонованого доступу.

Компоненти підсистеми захисту від несанкціонованого доступу повинні забезпечувати:

- ідентифікацію користувача;
- перевірку повноважень користувача при роботі з системою;
- розмежування доступу користувачів на рівні завдань та інформаційних масивів.

Захищена частина системи повинна використовувати "сліпі" паролі (при наборі пароля його символи не відображаються на екрані або замінюються одним типом символів; кількість символів не відповідає довжині пароля).

2.4.1.9. Вимоги щодо збереження інформації при аваріях.

Повинна бути передбачена можливість організації резервного копіювання даних системи засобами системного і базового програмного

забезпечення (ОС, СУБД), що входить до складу програмно-технічного комплексу.

2.4.1.10. Вимоги по захисту від впливу зовнішніх діянь.

Захист від впливу зовнішніх впливів повинна забезпечуватися засобами програмно технічного комплексу замовника.

2.4.1.11. Вимоги до патентної чистоти.

При створенні даної системи патентні дослідження не проводяться.

2.4.1.12. Вимоги по стандартизації і уніфікації.

У системі кодування інформації необхідно проводити за світовим класифікатором і стандартом.

2.4.2. Вимоги до функцій.

2.4.2.1. Перелік функцій із зазначенням вхідної та вихідної інформації наведено в таблиці 2.1.

Функції системи повинні бути поділені на модулі, доступ до яких мають користувачі з відповідним рівнем доступу:

- Модуль «Адміністрування» (доступ мають користувачі із роллю «адміністратор»)
- Модуль «Конструктор» (доступ мають користувачі із роллю «викладач»)
- Модуль «Тестер» (доступ мають користувачі із роллю «студент»)

Таблиця 2. 1. Перелік функцій, вхідної та вихідної інформації.

№ п/п	Найменування функції	Вхідна інформація	Вихідна інформація
Модуль «Адміністрування»			
1	Безпосередній доступ та можливість редагування інформації з таблиць БД.	Всі таблиці БД	Внесені зміни до таблиць БД
Модуль «Конструктор»			
1	Формування та виведення списку тестів	Таблиці БД «Quiz», «User»	Форма каталогу розроблених тестів

2	Фільтрація тестів у відповідності до навчальної дисципліни	Таблиці БД «Quiz», «Discipline», «User»	Форма каталогу розроблених тестів із відфільтрованими записами
3	Створення та редагування бланку тесту	Таблиці БД «Quiz», «User»	Форма редагування бланку тесту
4	Створення та редагування запитань	Таблиці БД «Quiz», «Question»	Форма редагування запитань
5	Створення та редагування відповідей	Таблиці БД «Question», «Answer»	Форма редагування відповідей
6	Перегляд готового бланку тесту із повним переліком запитань	Таблиці БД «Quiz», «Question», «Answer»	Форма перегляду інформації про тест
7	Відкриття доступу до тесту для окремої академічної групи	Таблиці БД «Quiz», «Group»	Форма налаштування доступу до тесту
8	Перегляд та експорт результатів тестування	Таблиці БД «Quiz», «Group», «User Attempt»	Форма перегляду результатів тестування
Модуль «Тестер»			
1	Формування та виведення списку доступних тестів	Таблиці БД «Quiz»	Форма каталогу доступних тестів
2	Перегляд інформації про окремий тест зі списком спроб тестування	Таблиці БД «Quiz», «User Attempt», «Attempt»	Форма перегляду інформації про тест
3	Проходження тесту	Таблиці БД «Quiz», «Question», «Answer», «Attempt»	Форма проходження тесту

2.4.3. Вимоги до видів забезпечення.

2.4.3.1. У вимогах до математичного забезпечення (МЗ) система не вимагає спеціального математичного забезпечення для реалізації покладених на неї функцій. Достатньо можливостей обраної СУБД.

2.4.3.2. Вимоги до інформаційного забезпечення (ІЗ).

2.4.3.2.1. Інформаційне забезпечення системи повинно містити дані, достатні для виконання всіх покладених на систему функцій. ІЗ повинно гарантувати раціональну організацію зберігання інформації та доступу до неї.

Заповнення БД інформацією покладається на замовника за методиками і формами, створеними розробниками системи.

Склад, структура і спосіб організації інформації представляються у логічній моделі БД і можуть уточнюватись на етапі технічного проектування.

2.4.3.2.2. Слід передбачити захист даних від руйнування при аваріях і порушеннях у енергоживленні системи — використання резервних копій БД.

2.4.3.3. Вимоги до лінгвістичного забезпечення (ЛЗ).

2.4.3.3.1. Для розробки програмних засобів, які реалізують виконання функцій і забезпечують сервіс користувачів повинні використовуватися мови високого рівня, які забезпечують створення структурних програм, а також мова обраної СУБД для здійснення доступу та маніпулювання даними.

2.4.3.3.2. Графічний інтерфейс користувача системи повинен бути виконаний державною мовою. Допускається при виконанні регламентних процедур використання англійської мови з вказанням підстав для її використання в технічній документації проекту.

2.4.3.4. Вимоги до програмного забезпечення (ПЗ).

2.4.3.4.1. Загальносистемне ПЗ має забезпечувати надійне і якісне виконання функціональних завдань системи. До загальносистемного ПЗ належить:

- сучасний веб-браузер – Google Chrome, Opera, Firefox, Microsoft Edge, Safari та ін.
- система управління БД (далі СУБД) — MySQL.

2.4.3.4.2. Загальні вимоги до системного ПЗ можна сформулювати так:

- мінімальні вимоги до ресурсів технічних засобів;
- максимальна швидкодія;
- повне задоволення потреб функціональних завдань системи.

2.4.3.4.3. Вимоги до СУБД:

- максимальне задоволення потреб функціональних задач;
- надійність;
- швидкість виконання запитів користувачів;
- можливість розширення та модифікації даних без збитку для системи загалом;
- мінімальні вимоги до ТЗ.

2.4.3.4.4. Програмні засоби введення та виведення даних і ведення діалогу повинні забезпечувати:

- виведення необхідних даних на екран у формі веб-документів із усіма;
- контроль введених даних та інформування користувача про виявлені помилки;
- виведення даних у формі файлу документа за запитом користувача.

2.4.3.4.5. При розробленні спеціального ПЗ слід виконати наступні вимоги:

- Використовувані програмні рішення не повинні конфліктувати між собою та мають бути повністю сумісні із загальносистемним ПЗ;
- розробка ПЗ повинна проводитись із використанням засобів ООП;
- ПЗ повинно мати модульну структуру;
- повинна бути передбачена можливість розширення складу задач у відповідності з новими функціональними потребами;
- ПЗ повинно гарантувати незалежність від окремого типу апаратної периферії;
- взаємодія із користувачем повинна проводитись в супроводі з поясненнями виконуваних дій та можливістю отримання підказок.

2.4.3.5. Вимоги до технічного забезпечення.

2.4.3.5.1. Технічні засоби системи повинні забезпечувати виконання функцій, перерахованих в таблиці 2.1.

2.4.3.5.2. Засоби обчислювальної техніки повинні забезпечувати обмін інформації в об'ємах, приведених в п. 4.3.2.

2.4.3.6. Вимоги до метрологічного забезпечення.

В системі не передбачене використання вимірювальних приладів та відповідного обладнання, на цій підставі вимоги до метрологічного забезпечення не висуваються.

2.4.3.7. Вимоги до організаційного забезпечення.

2.4.3.7.1. Організаційне забезпечення системи розробляється в відповідності з вимогами державного стандарту по АСУП.

2.4.3.7.2. При впровадженні системи на об'єкт не передбачається збільшення штату працівників. Територіальне розташування робочих місць, де буде встановлена система, визначається замовником.

2.5. Склад і зміст робіт по створенню системи

2.5.1. Стадії створення системи і терміни виконання робіт наведені в таблиці

2.2.

Таблиця 2. 2. Роботи по створенню системи та терміни їх виконання.

№ п/п	Найменування робіт	Терміни виконання робіт
1	Передпроектне дослідження об'єкта автоматизації	03.05-07.05
2	Технічне завдання	08.05-09.05
3	Технічний проект	10.05-21.05
4	Оформлення документації	22.05-28.05

2.6. Порядок контролю і приймання системи

2.6.1. Система вводиться на потужностях Центру інформаційних технологій НУХТ. При введенні в дію система повинна пройти приймальні випробування згідно з ДСТУ 3974-2000.

2.6.2. Замовник та розробники проводять випробування для оцінки працездатності системи. На основі випробувань приймається рішення про переведення системи в постійну експлуатацію.

Розробник складає програму випробувань, а замовник затверджує її.

2.6.3. Здача системи в доцільну експлуатацію здійснюється на базі технічного завдання на розробку та інструкцій користувача. За результатами

випробувань та тестової експлуатації формується перелік допрацювань та рекомендовані терміни їх виконання.

2.6.4. Введення в дію системи оформлюється актом здачі-прийому.

2.7. Вимоги до складу і змісту робіт із підготовки до введення системи в дію.

Для введення в дію ІС замовник проводить наступний ряд підготовчих робіт:

- проводить укомплектування технічних засобів;
- проводить дослідну експлуатацію і вводить систему в дію.

2.8. Вимоги до документації

2.8.1. На систему розробляється комплекс документації у складі: технічне завдання та технічний проект.

2.8.2. Документація на систему розробляється у відповідності з вимогами Державних стандартів серії 19 «Єдина система програмної документації» та серії 24 «Єдина система стандартів автоматизованих систем управління».

2.9. Джерело розробки

2.9.1. При розробленні технічного завдання на систему використано наступні документи:

- ДСТУ 3008-2015. Документація. Звіти у сфері науки і техніки. Структура та правила оформлювання;
- ДСТУ 3973-2000 Система розроблення та поставлення продукції на виробництво;

РОЗДІЛ 3. ОПИС КОМПЛЕКСУ ЗАДАЧ АВТОМАТИЗАЦІЇ

3.1. Інформаційне забезпечення системи

Діаграми логічної та фізичної моделі бази даних представлені, відповідно, у додатку G та додатку H.

3.1.1. Таблиці бази даних:

Таблиця «User» містить дані про зареєстрованих користувачів, а саме інформацію про повне ім'я користувача, адрес електронної пошти, логін та пароль, поле з відомостями про те, чи є даний користувач активним користувачем системи.

Структура даної таблиці подано у табл. 3.1.

Таблиця 3. 1. Структура таблиці «User».

№	Назва атрибуту	Тип даних
1	id	integer
2	email	varchar(256)
3	username	varchar(256)
4	name	varchar(256)
5	surname	varchar(256)
6	patronymic	varchar(256)
7	password	varchar(256)
8	active	bit
9	group_id	integer

Таблиці «Role» та «Group» містять у собі дані відповідно про ролі користувачів системи та академічні групи. Структура таблиць викладена відповідно у табл. 3.1. – 3.2.

Таблиця 3. 2. Структура таблиці «Role».

№	Назва атрибуту	Тип даних
1	id	integer
2	name	varchar(128)
3	description	varchar(256)

Таблиця 3. 3. Структура таблиці «Group».

№	Назва атрибуту	Тип даних
1	id	integer
2	name	varchar(128)

Таблиця «Quiz» містить інформацію про електронний бланк тесту, а саме відомості про його назву, опис, інструкцію, часове обмеження, обмеження в кількості дозволених спроб, кількість запитань для одночасного відображення, відомості про порядок відображення запитань, спосіб визначення результату, відомості про доступ студентів до перегляду вірних відповідей. Структура таблиці викладена в табл. 3. 4.

Таблиця 3. 4. Структура таблиці «Quiz».

№	Назва атрибуту	Тип даних
1	id	integer
2	caption	varchar(256)
3	description	text
4	instruction	text
5	time_limit	integer
6	attempt_limit	integer
7	question_cut	integer
8	random_order	bit
9	result_type	varchar(128)
10	show_answers	bit
11	author_id	integer
12	image_id	integer
13	dscipline_id	integer

Таблиця «Discipline» містить дані про навчальні дисципліни, відомості про повну та скорочену назву. Структура таблиці викладена в табл. 3. 5.

Таблиця 3. 5. Структура таблиці «Discipline».

№	Назва атрибуту	Тип даних
1	id	integer
2	fullname	varchar(256)
3	shortname	varchar(128)

Таблиця «Question» містить дані про запитання, відомості про текст запитання, коментар викладача, тип відповіді. Структура таблиці викладена в табл. 3. 6.

Таблиця 3. 6. Структура таблиці «Question».

№	Назва атрибуту	Тип даних
1	id	integer
2	text	text
3	comment	text
4	type	varchar(128)
5	quiz_id	integer
6	image_id	integer

Таблиця «Answer» містить інформацію про варіанти відповідей на певне запитання, відомості про текст відповіді, вартість (у балах) відповіді, відомості про те, чи є даний варіант відповіді правильним, коефіцієнт відхилення від вказаного варіанту відповіді. Структура таблиці викладена в табл. 3. 7.

Таблиця 3. 7. Структура таблиці «Answer».

№	Назва атрибуту	Тип даних
7	id	integer
8	text	text
9	cost	integer
10	correct	bit
11	deviation	int
12	quiz_id	integer
13	image_id	integer

Таблиця «Image» містить інформацію про зображення, що можуть бути приєднані в якості медіа додатків до відповідей, запитань або заголовків тестів. Таблиця містить у собі сам файл зображення у вигляді послідовності байтів, його закодований варіант, придатний для розміщення на веб-сторінці, та інформацію про тип файлу.

Структура таблиці викладена в табл. 3. 8.

Таблиця 3. 8. Структура таблиці «Image».

№	Назва атрибуту	Тип даних
1	id	integer
2	raw_data	image
3	rendered_data	text
4	filetype	varchar(64)

Таблиця «User_answer» містить відомості про відповідь користувача на запитання під час проходження тесту. Вона містить інформацію про текст відповіді, відомості про те, чи є відповідь користувача правильною. Структура таблиці викладена в табл. 3. 9.

Таблиця 3. 9. Структура таблиці «User_answer».

№	Назва атрибуту	Тип даних
1	id	integer
2	text	text
3	correct	bit
4	answer_id	integer
5	question_id	integer
6	attempt_id	integer

Таблиця «Attempt» містить дані про окрему спробу користувача пройти тестування. В ній розміщені дані про результат спроби (оцінку) та витрачений час. Структура таблиці викладена в табл. 3. 10.

Таблиця 3. 10. Структура таблиці «Attempt».

№	Назва атрибуту	Тип даних
1	id	integer
2	user_attempt_id	integer
3	Result	integer
4	time_spended	varchar(32)

Таблиця «User_attempt» містить інформацію про фінальний результат проходження тесту, після завершення всіх дозволених спроб. Структура таблиці викладена в табл. 3. 11.

Таблиця 3. 11. . Структура таблиці «User_attempt».

№	Назва атрибуту	Тип даних
1	id	integer
2	quiz_id	integer
3	user_id	integer
4	new_attempt_allowed	bit
5	final_result	integer

3.2. Алгоритмізація та реалізація комплексу задач автоматизації

3.2.1. Реалізація зв'язку з базою даних

Зв'язок з базою даних реалізовано із використанням можливостей бібліотеки SQLAlchemy, а точніше із використанням можливостей технології ORM(Object-Relational Mapping), що пов'язує БД з концепцією об'єктно-орієнтованих мов програмування. Бібліотека дозволяє описувати структуру БД та способи взаємодії з нею у вигляді класів. Власне, повний перелік класів, що є представленнями таблиць із створеної БД, викладено далі в цьому розділі.

На рис 3.1. показано допоміжні об'єкти, які використовуються для організації зв'язку між таблицями «User» та «Role», «Quiz» та «Group». В першому випадку, це необхідно для реалізації механізму присвоєння певному користувачу певної ролі (один користувач може мати кілька ролей, одна роль може бути присвоєна кільком користувачам), в другому – для реалізації можливості відкривати доступ до тесту для групи користувачів-студентів.

```
# допоміжні таблиці для реалізації зв'язку типу "many-to-many"
roles_users = db.Table("roles_users",
    db.Column("user_id", db.Integer, db.ForeignKey("user.id")),
    db.Column("role_id", db.Integer, db.ForeignKey("role.id")))

quizes_groups = db.Table("quizes_groups",
    db.Column("quiz_id", db.Integer, db.ForeignKey("quiz.id")),
    db.Column("group_id", db.Integer, db.ForeignKey("group.id")))
```

Рисунок 3. 1. Допоміжні таблиці для реалізації зв'язку «many-to-many»

На рис. 3.2.-3.0. зображено опис всіх класів таблиць БД.

```
class User(db.Model, UserMixin):
    id = db.Column(db.Integer, primary_key=True)
    email = db.Column(db.String(256), unique=True)
    username = db.Column(db.String(256), unique=True, index=True)
    name = db.Column(db.String(256))
    surname = db.Column(db.String(256))
    patronimic = db.Column(db.String(256))
    password = db.Column(db.String(256))
    active = db.Column(db.Boolean)
    group_id = db.Column(db.Integer, db.ForeignKey("group.id"))
    roles = db.relationship("Role", secondary=roles_users, backref=db.backref("users", lazy="dynamic"))
    tests = db.relationship("Quiz", backref=db.backref("author"), lazy="dynamic")
    attempts = db.relationship("UserAttempt", backref=db.backref("user"), lazy="dynamic")
```

Рисунок 3. 2. Опис класу представлення таблиці «User».

```
class Role(db.Model, RoleMixin):
    id = db.Column(db.Integer(), primary_key=True)
    name = db.Column(db.String(128), unique=True)
    description = db.Column(db.String(255))
```

Рисунок 3. 3. Опис класу представлення таблиці «Role».

```
class Discipline(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    fullname = db.Column(db.String(256), nullable=False)
    shortname = db.Column(db.String(128), nullable=False)
    tests = db.relationship("Quiz", backref=db.backref("discipline"), lazy="dynamic")

    def __init__(self, *args, **kwargs):
        super(Discipline, self).__init__(*args, **kwargs)
        self.generate_shortname()

    def generate_shortname(self):
        if self.fullname:
            target_str = self.fullname
            target_str.strip()
            self.shortname = "".join([s[0] for s in target_str.split(' ')]).upper()
```

Рисунок 3. 4. Опис класу представлення таблиці «Discipline»

Метод «generate_shortname» класу «Discipline» використовується для автоматичної генерації скороченого запису повної назви навчальної дисципліни при створенні нового запису в таблиці.

```
class Group(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    name = db.Column(db.String(256), nullable=False)
    students = db.relationship("User", backref=db.backref("group"), lazy="dynamic")
    tests = db.relationship("Quiz", secondary=quizes_groups, backref=db.backref("groups", lazy="dynamic"))
```

Рисунок 3. 5. Опис класу представлення таблиці «Group»

```
class Image(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    raw_data = db.Column(db.LargeBinary, nullable = False)
    rendered_data = db.Column(db.Text)
    filetype = db.Column(db.String(64), nullable = False)
    answers = db.relationship("Answer", backref="image", lazy="dynamic")
    questions = db.relationship("Question", backref="image", lazy="dynamic")
    quizzes = db.relationship("Quiz", backref="image", lazy="dynamic")

    def __init__(self, *args, **kwargs):
        super(Image, self).__init__(*args, **kwargs)
        self.render_picture()

    def render_picture(self):
        if self.raw_data:
            self.rendered_data = base64.b64encode(self.raw_data).decode('ascii')
```

Рисунок 3. 6. Опис класу представлення таблиці «Image»

Метод «render_picture» класу «Image» використовується для кодування файлу зображення в тестовий рядок із символів латинського алфавіту, з використанням алгоритму Base64. Даний текстовий рядок використовується для розміщення та відображення зображення на веб-сторінці.

```
class Answer(db.Model): # варіант відповіді
    id = db.Column(db.Integer, primary_key=True)
    text = db.Column(db.Text, nullable=False)
    cost = db.Column(db.Integer, nullable=False)
    correct = db.Column(db.Boolean, nullable=False)
    deviation = db.Column(db.Integer)
    question_id = db.Column(db.Integer, db.ForeignKey("question.id"), nullable=False)
    image_id = db.Column(db.Integer, db.ForeignKey("image.id"))
```

Рисунок 3. 7. Опис класу представлення таблиці «Answer»

```
class UserAnswer(db.Model): # відповідь користувача
    id = db.Column(db.Integer, primary_key=True)
    text = db.Column(db.Text)
    correct = db.Column(db.Boolean, nullable=False, default=False)
    answer_id = db.Column(db.Integer, db.ForeignKey("answer.id"))
    question_id = db.Column(db.Integer, db.ForeignKey("question.id"), nullable=False)
    attempt_id = db.Column(db.Integer, db.ForeignKey("attempt.id"), nullable=False)
```

Рисунок 3. 8. Опис класу представлення таблиці «UserAnswer»

```
class Question(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    text = db.Column(db.Text, nullable=False)
    comment = db.Column(db.Text)
    type = db.Column(db.String(128), nullable=False) # тип запитання: radio - одна відповідь, check
    - ввід користувача
    quiz_id = db.Column(db.Integer, db.ForeignKey("quiz.id"), nullable=False)
    answers = db.relationship("Answer", backref=db.backref("question"), lazy="dynamic", cascade="save")
```

Рисунок 3. 9. Опис класу представлення таблиці «Question»

```
class Quiz(db.Model):
    id = db.Column(db.Integer, primary_key=True)
    caption = db.Column(db.String(256), nullable=False)
    description = db.Column(db.Text)
    instruction = db.Column(db.Text)
    time_limit = db.Column(db.Integer, default=0)
    attempt_limit = db.Column(db.Integer, default=0)
    question_cut = db.Column(db.Integer, default=0) # кількість запитань, що відображаються під час к
    random_order = db.Column(db.Boolean, default=False) # випадковий порядок відображення питань
    result_type = db.Column(db.String(128), nullable=False, default="max") # спосіб підрахунку результ
    max - найвища оцінка, average - середнє значення, last - результат останньої спроби
    show_answers = db.Column(db.Boolean) # дозвіл на перегляд результатів
    author_id = db.Column(db.Integer, db.ForeignKey("user.id"), nullable=False)
    questions = db.relationship("Question", backref=db.backref("quiz"), lazy="dynamic", cascade="save")
```

Рисунок 3. 10. Опис класу представлення таблиці «Quiz»

```
class Attempt(db.Model): # спроба проходження тесту
    id = db.Column(db.Integer, primary_key=True)
    user_attemps_id = db.Column(db.Integer, db.ForeignKey("user_attempt.id"), nullable=False)
    result = db.Column(db.Integer)
    time_spended = db.Column(db.String(32))
    user_answers = db.relationship("UserAnswer", backref=db.backref("attempt"), lazy="dynamic")
```

Рисунок 3. 11. Опис класу представлення таблиці «Attempt»

```
class UserAttempt(db.Model): # поєднання всіх спроб проходження тесту
    id = db.Column(db.Integer, primary_key=True)
    quiz_id = db.Column(db.Integer, db.ForeignKey("quiz.id"), nullable=False)
    user_id = db.Column(db.Integer, db.ForeignKey("user.id"), nullable=False)
    new_attempt_allowed = db.Column(db.Boolean, nullable=False, default=True)
    final_result = db.Column(db.Integer, default=0) # фінальна оцінка за пр
```

Рисунок 3. 12. Опис класу представлення таблиці «UserAttempt»

Пов'язування класів представлень із екземпляром класу відбувається шляхом передачі даного екземпляру в конструктор кожного об'єкту представлення.

Ініціалізація екземпляру класу SQLAlchemy відбувається в головному модулі. В подальшому він використовується для організації базових операцій маніпулювання даними, а саме вибірка, створення, оновлення та видалення записів з таблиць БД.

Для отримання даних з таблиці БД використовується атрибут «query». Даний атрибут містить кожен клас моделі таблиці із описом вище і є аналогом SQL-команди SELECT. При зверненні до даного атрибуту він повертає об'єкт запиту, з допомогою якого можна отримати доступ до записів з таблиць. На рис. 3.13 показана робота даного механізму на прикладі таблиці «User».

```
>>> from app import db
>>> from models import *
>>> User.query
<flask_sqlalchemy.BaseQuery object at 0x000001CE204514F0>
>>> print(User.query)
SELECT [user].id AS user_id, [user].email AS user_email, [user].username AS user_username, [u
[user].surname AS user_surname, [user].patronymic AS user_patronymic, [user].password AS use
ve AS user_active, [user].group_id AS user_group_id
```

Рисунок 3. 13. Результат звернення до атрибуту query класу «User».

Об'єкт, що повертає атрибут, дозволяє використовувати наступні методи:

- get() – повертає запис, ключ якого передається в якості параметру;
- all() – повертає всі записи з таблиці;
- first() – повертає перший запис з таблиці;

- `filter_by()` – повертає всі записи, що відповідають вказаному параметру (аналогічно до SQL-команди `SELECT` із оператором `WHERE`). Даний метод може використовуватися в комбінації із двома попередніми.

Приклад роботи цих методів показаний на рис. 3.14 – 3.17.

```

from app import db
from models import *

disciplines = Discipline.query.all()

print("Об'єкти запитів")
for discipline in disciplines:
    print("Об'єкт: {} Тип: {}".format(discipline, type(discipline)))

print("Отримання даних із об'єтів (повна назва дисципліни)")

```

Рисунок 3. 14. Приклад виконання звичайного запиту, код програми для демонстрації.

```

Об'єкти запитів
Об'єкт: <Discipline 1> Тип: <class 'models.Discipline'>
Об'єкт: <Discipline 2> Тип: <class 'models.Discipline'>
Об'єкт: <Discipline 3> Тип: <class 'models.Discipline'>
Об'єкт: <Discipline 4> Тип: <class 'models.Discipline'>
Об'єкт: <Discipline 5> Тип: <class 'models.Discipline'>
Отримання даних із об'єтів (повна назва дисципліни)
1 Проектування інформаційних систем
2 Тестовий запис дисципліни
3 Інтелектуальний аналіз даних

```

Рисунок 3. 15. Приклад виконання звичайного запиту, результат виконання програми для демонстрації.

В даному прикладі виконується звичайний запит до таблиці «Discipline». Для отримання даних про повну назву навчальної дисципліни, виконується звернення до параметру класу моделі, що відповідає атрибуту таблиці з бази даних.

Приклад виконання параметричного запиту наведено на рис. 3.16 - 3.17.

```

from app import db
from models import *

discipline = Discipline.query.filter_by(shortname="
print("Приклад параметричного запиту:", discipline)
print("Результат виконання запиту:")
discipline = discipline.first()

```

Рисунок 3. 16. Приклад виконання параметричного запиту, код програми для демонстрації.

```

Приклад параметричного запиту:
SELECT discipline.id AS discipline_id, discipline.fu
discipline_fullname, discipline.shortname AS disciplin
FROM discipline
WHERE discipline.shortname = ?
Результат виконання запиту:

```

Рисунок 3. 17. Приклад виконання параметричного запиту, результат виконання програми для демонстрації.

Методи, які використовуються для створення та видалення записів:

- `add()` – додає новий запис в таблицю, екземпляр моделі якої передається в якості параметру (аналог SQL-команди INSERT);
- `delete()` – видаляє запис з таблиці, екземпляр моделі якої передається в якості параметру (аналог SQL-команди DELETE).

Демонстрація створення, оновлення та видалення записів з БД показана на рис. 3.18 – 3.20.

```

from app import db
from models import *
def print_list():
    disciplines = Discipline.query.all()
    for d in disciplines:
        print(d.id, d.shortname, d.fullname)
print("Список дисциплін до створення нового запису:")
print_list()
# оголошення нового запису таблиці Discipline
discipline = Discipline(
    fullname = "Приклад нового запису",
    shortname = "Приклад")
db.session.add(discipline) # створення нового запису
db.session.commit() # підтвердження виконання поперед
print("Список дисциплін після створення нового запису")
print_list()
# модифікація існуючого запису
discipline.fullname = "Філософія"

```

Рисунок 3. 18. Приклад створення та оновлення записів в таблиці БД, код програми для демонстрації.

```

Список дисциплін до створення нового за
1 ПІС Проектування інформаційних систем
2 ТЗД Тестовий запис дисципліни
3 ІАД Інтелектуальний аналіз даних
4 УІТП Управління ІТ-проектами
5 МКІТ Методи комп'ютерних інформаційни
Список дисциплін після створення нового
1 ПІС Проектування інформаційних систем
2 ТЗД Тестовий запис дисципліни
3 ІАД Інтелектуальний аналіз даних
4 УІТП Управління ІТ-проектами
5 МКІТ Методи комп'ютерних інформаційни
6 ПНЗ Приклад нового запису
Список дисциплін після модифікації запи
1 ПІС Проектування інформаційних систем
2 ТЗД Тестовий запис дисципліни

```

Рисунок 3. 19. Приклад створення та оновлення записів в таблиці БД, результат виконання програми для демонстрації.

```

from flask.globals import session
from app import db
from models import *
def print_list():
    disciplines = Discipline.query.all()
    for d in disciplines:
        print(d.id, d.shortname, d.fullname)
print("Список дисциплін до видалення запису:")
print_list()
discipline = Discipline.query.get(6)
db.session.delete(discipline) # видалення запису

```

Рисунок 3. 20. Приклад видалення запису з БД, код програми для демонстрації.

```

Список дисциплін до видалення запису:
1 ПІС Проектування інформаційних систем
2 ТЗД Тестовий запис дисципліни
3 ІАД Інтелектуальний аналіз даних
4 УІТП Управління ІТ-проектами
5 МКІТ Методи комп'ютерних інформаційних технологій
6 Філ. Філософія
Список дисциплін після видалення запису:
1 ПІС Проектування інформаційних систем
2 ТЗД Тестовий запис дисципліни
3 ІАД Інтелектуальний аналіз даних

```

Рисунок 3. 21. Приклад видалення запису з БД, результат виконання програми для демонстрації.

3.1.2. Реалізація інтерфейсу користувача

3.1.2.1. Реалізація форм введення інформації

Форми для введення інформації реалізовано із використанням можливостей бібліотеки WTForms. Кожна форма, що відображається на сторінці пов'язана із класом-представленням, де міститься перелік полів з їх атрибутами та валідаторами. На рис. 3.22 показаний приклад такого класу, що є

представленням форми для створення нового бланку тесту. До речі, він використовується також для редагування вже існуючого тесту.

```
class TestForm(FlaskForm):
    caption = StringField("Заголовок", validators=[DataRequired()])
    description = TextAreaField("Опис")
    instruction = TextAreaField("Інструкція")
    image = FileField("Зображення", validators=[FileAllowed(IMAGES, message=f"Дозволені тільки такі форм")])
    time_limit = IntegerField("Часове обмеження", default=0, validators=[NonNegativeValue])
    question_cut = IntegerField("Запитань для спроби", default=0, validators=[NonNegativeValue])
    attempt_limit = IntegerField("Кількість спроб", default=0, validators=[NonNegativeValue])
    result_type = SelectField("Спосіб визначення оцінки", choices=[("max", "Найкращий результат"), ("ave", "Арифметична середня"), ("last", "Результат останньої спроби)], default="max")
    discipline = SelectField("Дисципліна", coerce=int)
```

Рисунок 3. 22. Приклад класу форми для створення нового тесту.

Атрибути класу містять об'єкти полів із базовим набором параметрів, як от: текст заголовка, список функцій-валідаторів, що викликаються при натисненні кнопки для відправки даних на сервер, значення форми по замовчуванню та ін.

Коли користувач виконує послідовність команд «Конструктор→Новий тест», виконується функція, зображена на рис. 3.23.

```
@constructor.route("/create/test", methods=["GET", "POST"])
@roles_required("teacher")
def tCreate():
    tForm = TestForm()
    tForm.discipline.choices = [(discipline.id, discipline.fullname) for discipline in Discipline]
    if tForm.validate_on_submit():
        image = None
        if tForm.image.data:
            image = Image(
                raw_data = request.files[tForm.image.name].read(),
                filetype = tForm.image.data.mimetype
            )
            db.session.add(image)
            db.session.commit()
        test = Quiz(
            caption = tForm.caption.data,
            description = tForm.description.data,
            instruction = tForm.instruction.data,
            time_limit = tForm.time_limit.data,
            attempt_limit = tForm.attempt_limit.data,
            question_cut = tForm.question_cut.data,
            random_order = tForm.random_order.data,
            result_type = tForm.result_type.data,
            show_answers = tForm.show_answers.data,
            author_id = current_user.id,
            image_id = image.id if image else None
```

Рисунок 3. 23. Функція відображення сторінки створення нового тесту.

Спочатку відбувається ініціалізація класу TestForm. Якщо сторінка перезавантажується користувачем, або внаслідок виникнення помилки вводу, в клас автоматично передаються раніше введені дані, якщо такі виявлено, і, відповідно, автоматично підставляються в поля форми. На другому рядку відбувається підстановка в поле вибору дисципліни списку дисциплін із таблиці БД. Спосіб отримання даних з БД аналогічний до того, який був показаний в п. 3.2.1.

На третьому рядку виконується перевірка результату валідації форми. Якщо валідація проходить успішно, то в БД додається нове зображення в тому разі, коли користувач прикріпив файл до форми. Далі ініціюється та додається новий запис до таблиці «Quiz». Під кінець, користувача перенаправляють на сторінку перегляду деталей створеного бланку.

У випадку, коли валідація форми невдала (якщо виявлено помилки вводу або сторінка завантажується вперше), відбувається відображення шаблону сторінки з формою «eTest.html». Код шаблону сторінки продемонстрований на рис. 3.24.

```
{% extends 'base.html' %}

{% block title %}
Новий тест
{% endblock title %}

{% from "constructor/_macros.html" import render_form_field, render_checkbox_field, render_form_group_field, render_select_field %}

{% block content %}
<div class="container-fluid m-0 p-3 row justify-content-center">
  <div class="col-lg-8">
    <form id="qForm1" method="POST" enctype="multipart/form-data">
      {{ form.hidden_tag() }}
      {{ render_form_field(form.caption, form.caption.label.text) }}
      {{ render_form_field(form.description, form.description.label.text) }}
      {{ render_form_field(form.instruction, form.instruction.label.text) }}
      {{ render_select_field(form.result_type, form.result_type.label.text) }}
      {{ render_select_field(form.discipline, form.discipline.label.text) }}
      {{ render_form_group_field(form.time_limit, form.time_limit.label.text, postfix="хв", title="Часове обмеження (0 - без обмежень)") }}
      {{ render_form_group_field(form.attempt_limit, form.attempt_limit.label.text, title="Кількість дозволених спроб (0 - необмежена кількість)") }}
      {{ render_form_group_field(form.question_cut, form.question_cut.label.text, title="Кількість запитань для відображення (0 - відобразити весь список запитань)") }}
      {{ render_checkbox_field(form.random_order, form.random_order.label.text, title="Запитання повинні відображатися в випадковому порядку") }}
      {{ render_checkbox_field(form.show_answers, form.show_answers.label.text, title="Дозволити переглядати результати тесту") }}
      {{ render_file_field(form.image, form.image.label.text, image) }}
    <div class="mb-3">
      {{ form.submit(class="btn btn-primary w-75 d-block mx-auto", value="Зберегти") }}
    </div>
  </div>
</div>
{% endblock content %}
```

Рисунок 3. 24. Код шаблону «eTest.html»

Для відображення кожного поля форми для вводу використовується окремий макрос, який імпортується із шаблону «_macros.html».

Сторінка, яку бачить користувач, після завантаження шаблону та всіх вкладених макросів, зображена на рис. 3.25.

Контроль знань Вихід Конструктор Адміністрування Тестування

Заголовок

Опис

Інструкція

Спосіб визначення оцінки
Найкращий результат

Дисципліна
Проектування інформаційних систем

Часове обмеження 0

Кількість спроб 0

Запитань для спроби 0

Випадковий порядок запитань

Перегляд результатів тесту

Зображення
Выберите файл | Файл не выбран

Рисунок 3. 25. Сторінка із формою для введення даних про тест.

Інформування користувача відбувається через спливаючі підказки, що з'являються, якщо навести курсор на відповідне поле. Приклад такої підказки показаний на рис. 3.26.

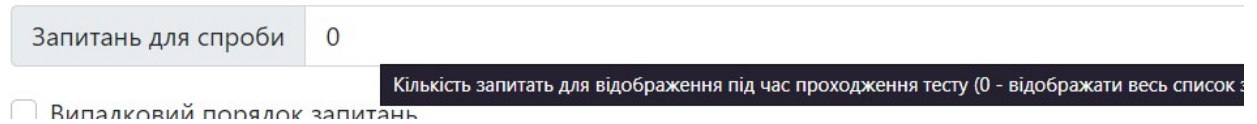


Рисунок 3. 26. Приклад спливаючої підказки.

Якщо користувач робить помилку при введенні даних, йому виводиться попередження у вигляді спливаючого повідомлення під полем, де знайдена помилка. Даний сценарій показаний на рис. 3.27.

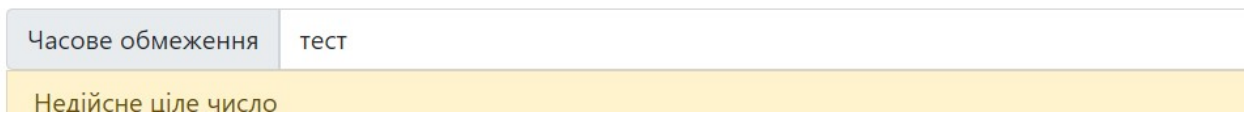


Рисунок 3. 27. Приклад попередження про допущену помилку.

Якщо користувач завантажує файл зображення, його вміст автоматично відображається під відповідним полем. Даний механізм реалізовано із використанням мови JavaScript. Код JS-скрипта «fileInput», який відображає зображення на сторінці, наведений на рис. 3. 28.

```

const fileInputs = document.querySelectorAll("input[type=file]");
for(let fileInput of fileInputs) {
  let img = fileInput.parentNode.querySelector("img");
  let svg = fileInput.parentNode.querySelector("svg");
  if (img.getAttribute("src")== "") {
    img.classList.add("visually-hidden");
    if (svg.classList.contains("visually-hidden")) {svg.classList.remove("visually-hidden")}
  }
  fileInput.onChange = function(){
    let svg = this.parentNode.querySelector("svg");
    let img = this.parentNode.querySelector("img");

    if (this.files.length > 0) {
      let file = this.files[0];

      let fileReader = new FileReader();
      fileReader.readAsDataURL(file);
      fileReader.onload = function(evt){
        if( evt.target.readyState == FileReader.DONE) {img.setAttribute("src",
        fileReader.result)}
      };

      if (!svg.classList.contains("visually-hidden")) {svg.classList.add("visually-hidden")}
      if (img.classList.contains("visually-hidden")) {img.classList.remove("visually-hidden")}
    }
    else {

```

Рисунок 3. 28. Код JS-скрипта «fileInput» для динамічного відображення зображення на сторінці.

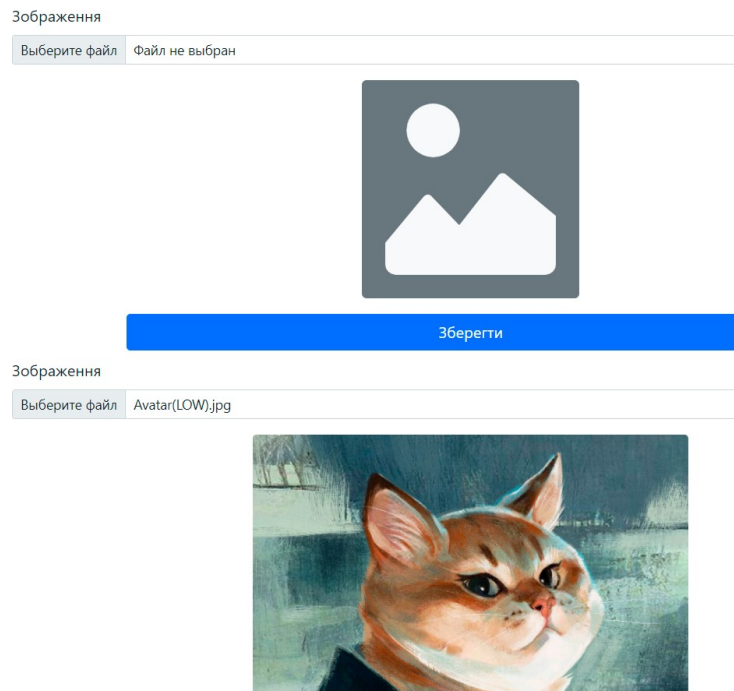


Рисунок 3. 29. Результат роботи JS-скрипта «fileInput».

Аналогічно до сторінки створення нового тесту реалізовано наступні сторінки:

- Форма авторизації користувача;

- Форма реєстрації користувача;
- Форма створення запитання;
- Форма створення варіанту відповіді;
- Форма налаштування доступу до тесту.

Для створення нових та редагування існуючих записів, використовується один і той же набір шаблонів форм. Функція для відображення сторінки редагування подібна до тієї, що використовується для відображення сторінки створення. Приклад функції рендеру сторінки редагування для форми TestForm показаний на рис. 3.30.

```
@constructor.route("/edit/test-<tid>", methods=["GET", "POST"])
@roles_required("teacher")
def tEdit(tid):
    test = Quiz.query.get(tid)
    tImage = Image.query.get(test.image_id)
    if request.method == "GET":
        tForm = TestForm(obj=test)
        tForm.discipline.choices = [(discipline.id, discipline.fullname) for discipline in D
        tForm.discipline.data = test.discipline.id
    if request.method == "POST":
        tForm = TestForm()
        tForm.discipline.choices = [(discipline.id, discipline.fullname) for discipline in D
    if tForm.validate_on_submit():
        if tForm.image.data:
            image = Image(
                raw_data = request.files[tForm.image.name].read(),
                filetype = tForm.image.data.mimetype
            )
            db.session.add(image)
            db.session.commit()
        try:
            test.image_id = image.id
        except NameError:
            pass
        test.caption = tForm.caption.data
        test.description = tForm.description.data
        test.instruction = tForm.instruction.data
        test.time_limit = tForm.time_limit.data
        test.question_cut = tForm.question_cut.data
        test.attempt_limit = tForm.attempt_limit.data
```

Рисунок 3. 30. Код функції відображення форми редагування тесту.

Послідовність дій схожа із функцією з рис. 3.23. Ключова відмінність полягає в механізмі оновлення запису в таблиці БД, робота якого була показана в п. 3.2.1.

3.1.2.2. Реалізація сторінок-каталогів.

Механізм відображення списку тестів реалізований схожим чином для функціональних модулів «Конструктор» і «Тестер». В зв'язку з цим, фільтрація відбувається за ідентичним сценарієм для обох випадків.

Коли користувач переходить на сторінку каталогу, відбувається виконання функції для відображення списку. В якості прикладу, на рис . 3.31. показана функція відображення із модуля «Тестер».

```
@constructor.route("/test-list", methods=["GET", "POST"])
@roles_required("teacher")
def tests():
    if request.method == "POST":
        dfilterValue = request.form.get("filter")
        tlist = Quiz.query.filter_by(author_id=current_user.id, discipline_id = int(dfilterValue)
                                   if dfilterValue else Quiz.query.filter_by(author_id=current_user.id).all())
    else:
        dfilterValue = None
        tlist = Quiz.query.filter_by(author_id=current_user.id).all()
    ilist = Image.query
```

Рисунок 3. 31. Функція відображення каталогу тестів.

В залежності від того, чи вибраний певний параметр фільтрації, виконується звичайний або параметричний запит на вибірку даних з БД. В шаблон передається результат даного запиту, який потім відображається на сторінці. Код шаблону сторінки каталогу наведений на рис. 3.32.

```
{% block content %}
<div class="container-fluid m-0 p-3 row justify-content-center">
  <div class="col-lg-8">
    {% if tests %}
    {% for test in tests %}
    <div class="input-group mb-3">
      <h5 class="input-group-text m-0">{{ loop.index }}</h5>
      {{ render_test_card(test, images.filter_by(id=test.image
    </div>
    {% endfor %}
    {% else %}
    <p class="text-center fs-4 text-muted">Список тестів по
    {% endif %}
```

Рисунок 3. 32. Фрагмент шаблону для відображення каталогу тестів.

3.1.2.3. Реалізація експорту звітів.

Експорт звіту по результатам проходження доступний із сторінки для перегляду деталей тесту. На сторінці перегляду звіту користувач може переглянути таблицю з результатами тестування. На рис. 3.33 показаний код функції для відображення шаблону сторінки, а на рис. 3.34 – фрагмент самого шаблону для побудови таблиці.

```
@constructor.route("/test-<tid>/report", methods=["GET", "POST"])
@roles_required("teacher")
def report(tid):
    test = Quiz.query.get(tid)
    if request.method == "GET":
        param = None
    else:
        param = request.form.get("paramGroup")
    groups = Quiz.query.get(tid).groups.all()
```

Рисунок 3. 33. Функція для відображення сторінки перегляду звіту.

```
{% for group in test.groups %}
<tr class="table-primary">
    <td colspan="4">Група "{{ group.name }}"</td>
</tr>
{% for user in group.students.all() %}
<tr
    class="{% if user.attempts.filter_by(quiz_id=test.id).first() %}table-
    table-secondary{% endif %}">
    <th scope="row">{{ loop.index }}</th>
    <td>{{ "{} {} {}".format(user.name, user.surname, user.patronymic) }}
    <td>
        {% if user.attempts.filter_by(quiz_id=test.id).first() %}
        {{ user.attempts.filter_by(quiz_id=test.id).first().attempts.co
        {% else %}
        0
        {% endif %}
    </td>
```

Рисунок 3. 34. Фрагмент код шаблону для побудови звіту.

Фрагмент коду вміщує два вкладені цикли, які відображають інформацію по результатам тестування окремого студента, окремої групи.

Коли користувач натискає на кнопку «Експорт», викликається функція, що формує файл звіту у форматі файлу електронних таблиць Excel. Для цих

цілей використовується бібліотека «Oreproxl». В залежності від того, чи був ввімкнений фільтр груп, експортується таблиця, де:

- 1) наявні результати тестування всіх груп, при чому для кожної групи відводиться свій лист;
- 2) наявні результати тестування тієї групи, по якій відбувалась фільтрація.

Фрагмент коду формування файлу звіту наведений на рис. 3.35.

```
groups = test.groups.all()
for group in groups:
    sheet = book.create_sheet(title=group.name)
    sheet.append(['#', 'Студент', 'Кількість спроб', 'Оцінка'])
    students = group.students.all()
    for student in enumerate(students):
        fullname = f"{student[1].name} {student[1].surname} {student[1].patronymic}"
        user_attempt = student[1].attempts.filter_by(quiz_id=test.id).first()
        attempt_count = user_attempt.attempts.count() if user_attempt else 0
        final_result = user_attempt.final_result if user_attempt else None
        sheet.append([student[0], fullname, attempt_count, final_result])
    book.remove(book['Sheet'])
    output = BytesIO()
    book.save(output)
```

Рисунок 3. 35. Фрагмент коду функції експорту звіту – формування звіту по всім групам.

В результаті користувачу надсилається готовий файл, який можна зберегти в постійній пам'яті пристрою.

3.3. Інструкція користувача

При переході на головну сторінку, вгорі, на панелі навігації, не буде жодних посилань, окрім посилання на сторінку із формою для авторизації. Слід звернути увагу на те, що, в залежності від розміру екрану девайсу, панель навігації може відрізнятись. На рис. 3.36-3.37 продемонстровано обидва випадки.

Контроль знань Вхід

Головна сторінка сайту

Рисунок 3. 36. Відображення панелі навігації на стаціонарному комп'ютері.

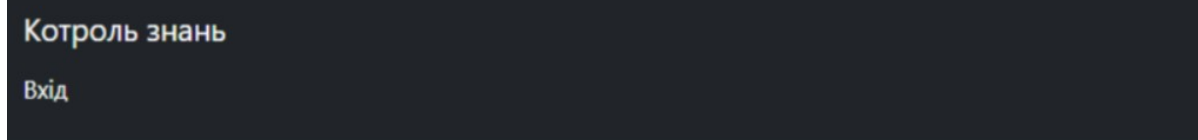
Контроль знань

Головна сторінка сайту

Зараз Ви перебуваєте на головній сторінці

Рисунок 3. 37. Відображення панелі навігації на мобільному телефоні.

Для розкриття панелі, на мобільному пристрої, слід натиснути на піктограму в правому верхньому куті, після чого будуть доступні всі пункти меню, як на рис. 3.37.



Головна сторінка сайту

Рисунок 3. 38. Панель навігації у розгорнутому вигляді.

При переході на сторінку авторизації (рис. 3.39), потрібно ввести свої дані в форму та натиснути на кнопку «Увійти». Для того, що залишатись в системі, після завершення сесії, треба поставити галочку навпроти «Запам'ятати мене».

Рисунок 3. 39. Форма авторизації.

Якщо ви не зареєстровані в системі, перейдіть на форму реєстрації, натиснувши на посилання «Реєстрація». Заповніть форму реєстрації (рис. 3.40) та надішліть дані на сервер, натиснувши «Зареєструватися».

Реєстрація

Справжнє ім'я

Прізвище

По-батькові

Ім'я користувача

Email Address

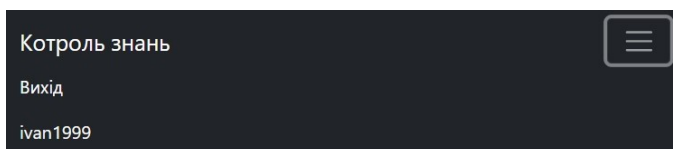
Пароль

Підтвердження паролю

[Зареєструватися](#)

Рисунок 3. 40. Приклад заповненої форми реєстрації.

Після успішного завершення авторизації відбудеться переправлення на головну сторінку, а на панелі навігації з'явиться ім'я користувача (рис. 3.41.).

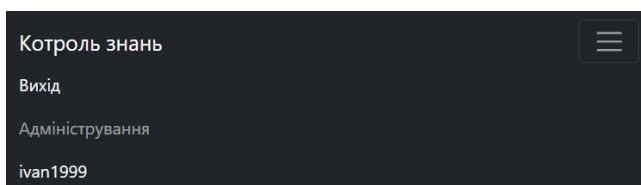


Головна сторінка сайту

Зараз Ви перебуваєте на головній сторінці

Рисунок 3. 41. Результат авторизації.

Якщо користувач має роль «administrator», на панелі навігації з'явиться пункт «Адміністрування», як показано на рис. 3.42.



Головна сторінка сайту

Зараз Ви перебуваєте на головній сторінці

Рисунок 3. 42. Панель навігації користувача з роллю "administrator"

При переході на присилання відкриється сторінка адміністрування із списком таблиць БД. Робота з таблицями подібна до роботи в будь-якій іншій СУБД. Приклад форми для редагування таблиці показаний на рис. 3.43.

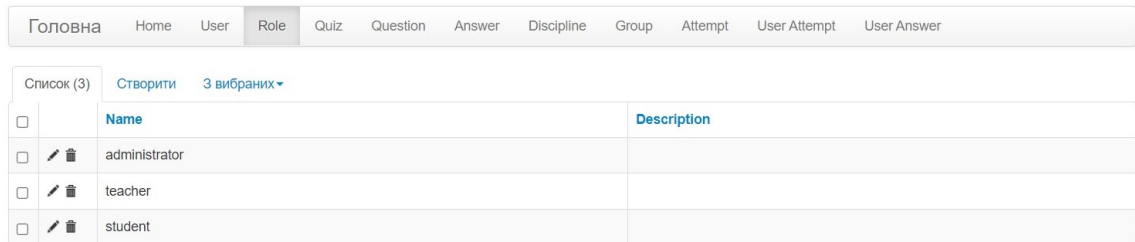
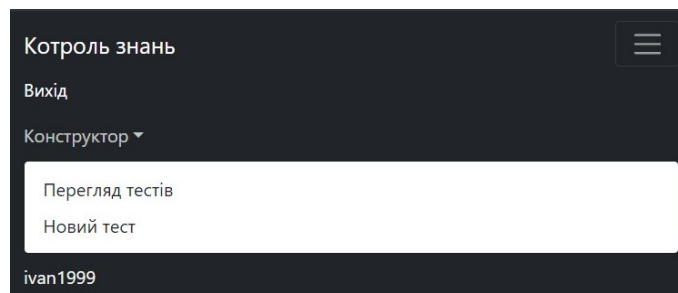


Рисунок 3. 43. Приклад форми для редагування таблиці «Role».

Якщо користувач має роль «teacher», на панелі навігації з'явиться пункт «Конструктор», як показано на рис. 3.44.



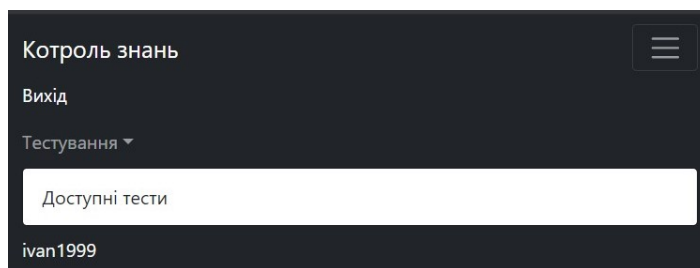
Головна сторінка сайту

Зараз Ви перебуваєте на головній сторінці

Рисунок 3. 44. Панель навігації користувача з роллю "teacher"

При переході за посиланням «Перегляд тестів» відкриється список доступних тестів. Пункт «Новий тест» переправить користувача на сторінку створення нового тесту.

Якщо користувач має роль «student», на панелі навігації з'явиться пункт «Тестування», як показано на рис. 3.44.



Головна сторінка сайту

Зараз Ви перебуваєте на головній сторінці

Рисунок 3. 45. Панель навігації користувача з роллю "student"

Якщо перейти за посиланням «Перегляд тестів» або «Доступні тести», відкриється сторінка із списком усіх тестів. Для викладачів будуть відображатися всі створені тести, для студентів – список доступних для їхньої групи тестів. На сторінці є палень, де розміщена кнопка для швидкого переходу на початок списку та поле фільтру. Поле вміщує випадаючий список навчальних дисциплін із бази даних. Для ввімкнення фільтру потрібно обрати один із пунктів списку та натиснути «Фільтрувати». Послідовність описаних дій наведена на рис. 3.46 – 3.48.

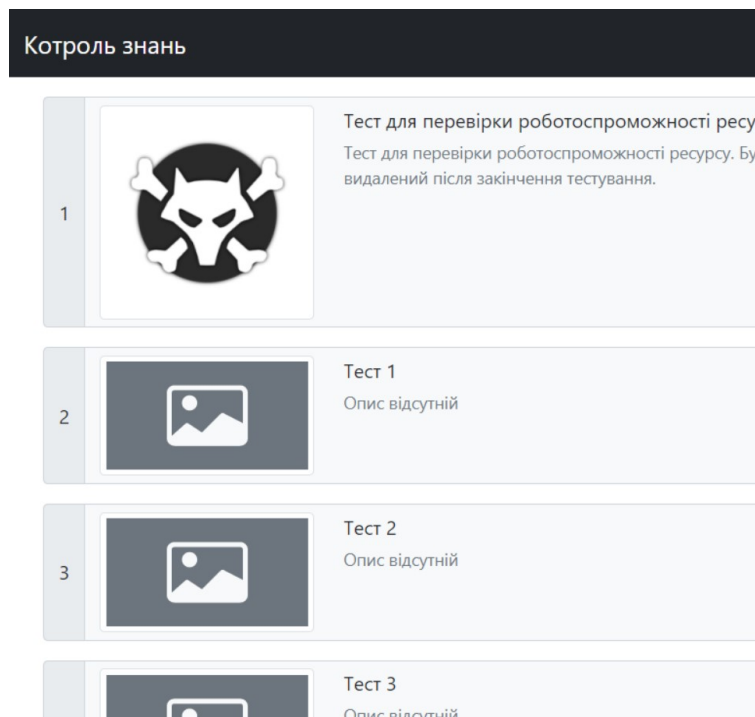


Рисунок 3. 46. Сторінка перегляду каталогу тестів.

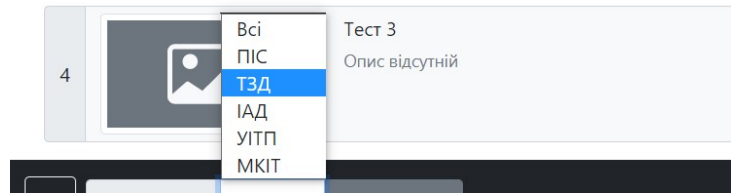


Рисунок 3. 47. Вибір дисципліни із списку для подальшої фільтрації.

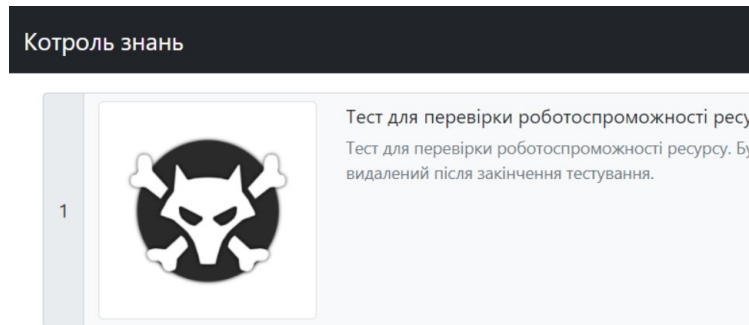


Рисунок 3. 48. Результат фільтрації.

Для викладачів, щоб переглянути деталі тесту, слід натиснути на кнопку справа від запису тесту. Відкриється відповідна сторінка із всією інформацією про тест (рис. 3.49).

Якщо натиснути на кнопку «Показати зображення тесту», відкриється панель із зображенням, що пов'язане із даним тестом.

При натисканні кнопки «Показати список запитань» (число біля назви – це загальна кількість запитань для даного тесту), відкривається інтерактивний перелік запитань із варіантами відповідей. Перелік організований у формі картотеки, де кожна картка – це відповідь із варіантами запитань. Приклад такої картки представлений на рис. 3.50.

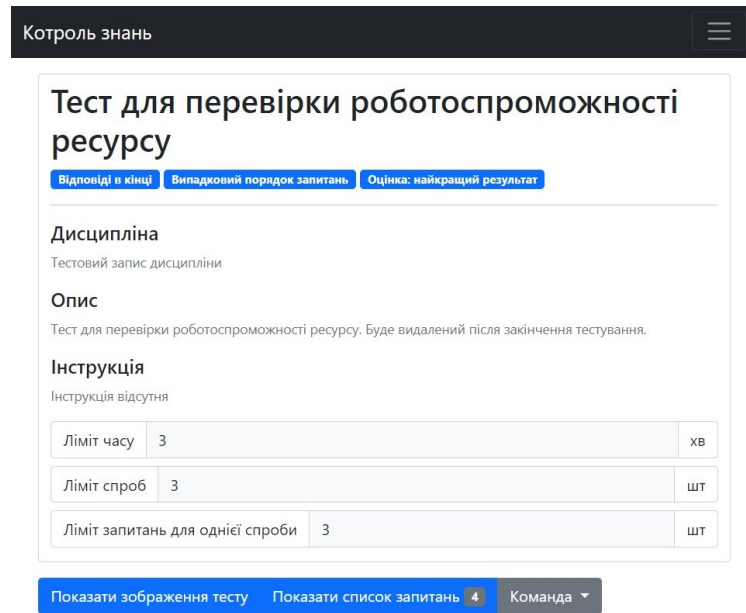


Рисунок 3. 49. Сторінка для перегляду інформації про тест.

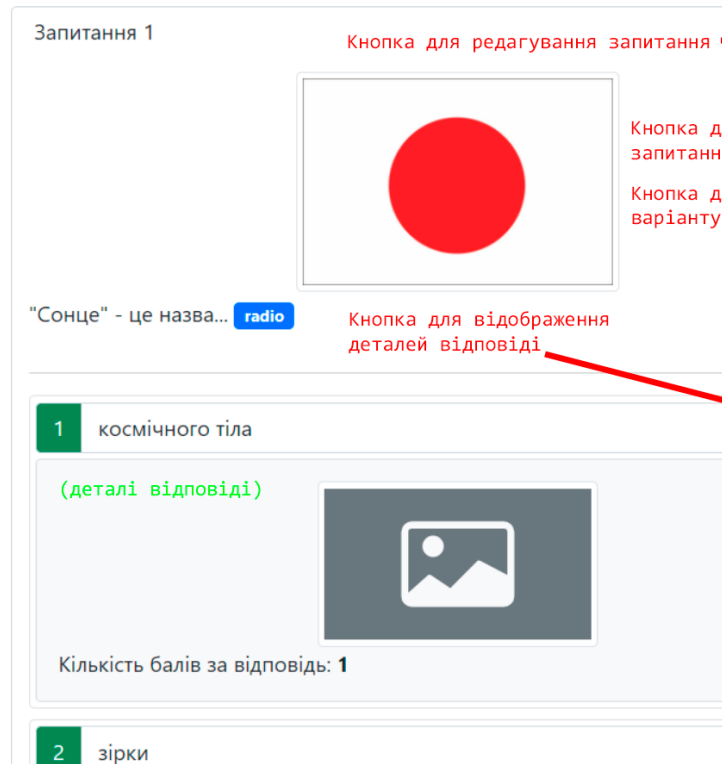


Рисунок 3. 50. Запитання із загального списку запитань тесту.

Кнопка «Команда» відкриває випадаючий список команд, що складається із наступних пунктів:

- «Додати запитання»
- «Редагувати тест»
- «Видалити тест»

- «Налаштування доступу»
- «Звіт по результатам»

Команди «Додати запитання» та «Редагувати тест» переправляють користувача на відповідні сторінки із формати для створення запитання та редагування тесту.

Сторінка із запитанням та варіантами відповідей подібна до сторінки із тестом та списком запитань.

Команда «Видалити тест» видаляє бланк тесту.

Команда «Налаштування доступу» відкриває сторінку налаштування доступу до тесту (рис. 3.51.), де знаходиться список всіх академічних груп та таблиця, де міститься інформація про групи, для яких відкритий доступ до тесту. Для того, що відкрити доступ до тесту, необхідно обрати одну або кілька груп із загального переліку та натиснути «Підтвердити». Щоб закрити доступ для групи, потрібно натиснути на піктограму із знаком «мінус» навпроти рядку таблиці.

Відкрити доступ для

Оберіть групу(и)

АК-4-7
АК-1-1
АК-7-4
ЦК-3-1с

Підтвердити Повернутися до тесту

Доступ відкритий для груп

#	ID	Назва групи
1	1	АК-4-7
2	2	АК-1-1

Рисунок 3. 51. Сторінка налаштування доступу.

Команда «Звіт по результатам» відкриває сторінку для перегляду звіту по результатам тестування (рис. 3.52). Є можливість відфільтрувати дані подібно до того, як це робиться на сторінці каталогу. Якщо натиснути «Експорт», відкриється вікно для збереження файлу звіту. Приклад вмісту файлу наведений на рис. 3.53.

Контроль знань

Звіт по тесту "Тест для перевірки робото спроможності ресурсу"

#	Студент	Кількість спроб
Група "АК-4-7"		
1	Петро Петренко Петрович	0
2	Маргарита Іващенко Петрівна	0
3	Тарас Бондар Тарасович	0
4	Ісай Сахаров Мойсейович	1
Група "АК-1-1"		
1	Мойсей Брутт Олександрович	2
2	Анна Кареніна Петрівна	0
3	Інокентій Тичина Йосипович	0
4	Антон Чехов Павлович	0

Рисунок 3. 52. Сторінка перегляду звіту по результатам тестування.

report (1).xlsx - Excel

ФАЙЛ ГЛАВНАЯ ВСТАВКА РАЗМЕТКА СТРАНИЦЫ ФОРМУЛЫ ДАННЫЕ РЕЦЕНЗИРО

Вставить Шрифт Выравнивание Число Условное форматирование Форматировать как таблицу Ячейки Редактирова

Буфер обмена fx #

#	Студент	Кількість спроб	Оцінка
0	Петро Петренко Петрович	0	
1	Маргарита Іващенко Петрівна	0	
2	Тарас Бондар Тарасович	0	
3	Ісай Сахаров Мойсейович	1	4

Рисунок 3. 53. Вміст xlsx-файлу звіту.

Як зазначалося раніше, на сторінці каталогу, студенти можуть переглядати доступні їм тести. Так само як і викладач, студент може перейти на сторінку із деталями тесту (рис. 3.54.), де будуть відображена загальна інформація та відомості про попередні спроби пройти тестування.

Кнопка «Почати тестування» відкриває сторінку із тестом (рис. 3.55.). Якщо кнопка червоного кольору – це означає, що вичерпано ліміт спроб.

Кнопка в кінці рядка таблиці «Результати попередніх спроб» відкриває сторінку для ознайомлення із відповідями користувача (рис. 3.56) (можна переглянути правильні відповіді та помилки, які допустив студент).

Контроль знань
☰

Тест для перевірки роботоспроможності ресурсу

Опис тесту
Тест для перевірки роботоспроможності ресурсу. Буде видалений після закінчення тестування.

Інструкція
Інструкція відсутня

Обмеження в часі: 3хв
Залишилося спроб: 2
Поточна оцінка: 4

Почати тестування

Результати попередніх спроб


	Затрачений # час	Набраних балів	Всього відповідей	Правильних відповідей	
1	00:18	4	3	2	i

Рисунок 3. 54. Сторінка для перегляду інформації про тест.

Залишилося часу: 02:56

Увага! Після зупинки таймера, дані будуть автоматично відправлені на о

Запитання 1



"Сонце" - це назва...

- космічного тіла
- зірки
- планети
- комети

Рисунок 3. 55. Сторінка проходження тесту.

Запитання 2

Скільки буде 2+2?

Введіть відповідь у вигляді цілого числа.

Текст відповіді

4

Варіанти відповідей: "4";

✓ Відповідь правильна

Запитання 3

RGB?

Red **Правильна відповідь**

Blue **Правильна відповідь**

Green **Правильна відповідь**

Purple

Рисунок 3. 56. Сторінка перегляду відповідей.

3.4. Технічне та системне забезпечення розробки

3.4.1. Обґрунтування вибору ОС та протоколу обміну даними.

Під час розробки програмного забезпечення використовувалася високорівнева кросплатформна мова програмування Python. Особливістю даної мови є те, що вона – інтерпретуєма. Це означає, що код програми не компілюється в машинний код заздалегідь, а виконується «на льоту» програмою-інтерпретатором. В зв'язку з цим, серверна частина інформаційної системи буде працювати на будь-якій операційній системі, на якій встановлений даний інтерпретатор.

Графічний інтерфейс користувача розроблений із застосуванням можливостей фреймворку Bootstrap до складу якого входить набір HTML та CSS шаблонів, JS сценаріїв. Використання даного інструменту гарантує коректне відображення та роботу сайту на всіх підтримуємих цим фреймворком браузерах і операційних системах, до списку яких входять:

- MS Windows
- Windows Mobile
- Android
- IOS

- Mac OS
- ОС сімейства Linux та ін.

Основним протоколом для обміну даними є протокол передачі гіпертексту HTTP, що використовується для передачі довільних даних. HTTP в даний час повсюдно використовується у всесвітній павутині для отримання інформації з веб-сайтів.

3.4.2. Розробка і обґрунтування стратегії адміністрування системи.

Створена ІС не вимагає розробки специфічної стратегії адміністрування. Для управління базою даних достатньо застосування типових рішень, розроблених автором СУБД – компанією Microsoft. Вся необхідна технічна документація розміщена на офіційному сайті Microsoft.

Єдиним зауваженням є те, що для забезпечення зручності доступу до даних з таблиць БД, для адміністраторів доступний окремий модуль, який, між усім, також є реалізацією типового рішення, що пропонується бібліотекою-розширенням Flask-Admin. Технічна документація для даної бібліотеки також знаходиться у відкритому доступі, а інтерфейс модулю спроектований із прицілом на мімікрію під найбільш широко використовуванні системи адміністрування БД, як от phpMyAdmin та подібні.

3.4.3. Заходи захисту від несанкціонованого доступу до системи.

Захист від несанкціонованого доступу реалізовано через систему реєстрації користувачів, яка побудована на базі бібліотеки Flask-Security.

На основі даної бібліотеки побудований контроль доступу на основі ролей. Це означає, що коли користувача переходить на сторінку одного із функціональних модулів, робиться перевірка на відповідність назви присвоєних йому ролей до вказаного параметру. В тому разі, якщо користувач не авторизований або не проходить перевірку по іншим причинам, замість переправлення на сторінку він отримує відповідне попереджувальне повідомлення.

Інший передбачений захід захисту, реалізований на базі даної бібліотеки – це хешування паролю користувача. В зв'язку з цим, навіть адміністратор системи, що має доступ до бази даних не може заволодіти інформацією користувачів системи, відповідно авторизуватися під чужим ім'ям.

Ще один засіб захисту – використання бібліотеки SQLAlchemy в ролі основного інструменту для забезпечення доступу до бази даних. Можливості даної бібліотеки дозволяють уникнути використання «сирих» SQL-запитів (запис SQL-команд у формі текстових літералів), що унеможлиблює застосування SQL-ін'єкцій для отримання несанкціонованого доступу до БД.

3.5. Розрахунок економічного ефекту від впровадження системи.

Вхідні дані для розрахунку:

- 1) Ступінь новизни розроблюваних задач - В - «Розробка проекту з використання типових проектних рішень за умови їх змін».
- 2) Група складності алгоритму - 3 - «Алгоритми, що реалізують стандартні методи рішень і не передбачають використання складних чисельних і логічних методів».
- 3) Узагальнені дані вхідної та вихідної інформації для системи за видами інформації табл. 3.12.

Таблиця 3. 12. Узагальнені дані для вхідної та вихідної інформації системи.

Вид інформації	Позначення	Кількість наборів даних
Змінна інформація	ЗІ	m=5
Нормативно-довідкова інформація	НДІ	n=4
Банк (база) даних	БД	p=1
Обробка в режимі реального часу	РЧ	Так
Забезпечення телекомунікаційної обробки даних і управління віддаленими об'єктами	ТОУ	Так

- 4) Витрати часу на систему, а саме на розробку ескізного проекту T_1 і технічного завдання T_2 , будуть такими, як показано в табл. 3.13.

Таблиця 3. 13. Визначення витрат часу для системи

Вид системи	Стадія розробки системи	
	Передпроектне дослідження, T_1	Технічне завдання, T_2
	В	В
Управління науково-технічною інформацією.	67	24

- 5) Розрахунки значень витрат часу для стадій «Технічний проект», «Робочий проект» і «Впровадження».

Перелік вхідних даних для розрахунку:

- Кількість форм вхідної інформації $B_1 = 4$

- Кількість форм вихідної інформації $B_2 = 2$
- базове значення витрат часу для стадій «Технічний проект»: $T_{Б3} = 73$;
- базове значення витрат часу для стадій «Робочий проект»: $T_{Б4} = 125$;
- базове значення витрат часу для стадій «Впровадження»: $T_{Б5} = 36$

Таблиця 3. 14. Коефіцієнт ступеню новизни проекту, k_o

Стадія розробки проекту	Вид обробки	Ступінь новизни - В
Технічний проект	РЧ	1,26
Робочий проект	РЧ	1,32
Впровадження	РЧ	1,21

Таблиця 3. 15. Коефіцієнти k_1, k_2, k_3 для стадії «Технічний проект»

Вид використаної інформації	Ступінь новизни
	В
k_1 (ЗІ)	1.0
k_2 (НДІ)	0.72
k_3 (БД)	2.08

Таблиця 3. 16. Коефіцієнти k_1, k_2, k_3 для стадії «Робочий проект»

Вид використаної інформації	Група складності алгоритму	Ступінь новизни
		В
k_1 (ЗІ)	3	1.0
k_2 (НДІ)	3	0.48
k_3 (БД)	3	0.40

Коефіцієнт складності контролю вхідної та вихідної інформації $k_C = 1,08$.

Визначення коефіцієнту трудомісткості робіт на стадії «Технічний проект»:

$$k_{\Pi} = \frac{1 * 5 + 0,72 * 4 + 2,08 * 1}{5 + 4 + 1} = 0,996$$

Витрати часу на розробку «Технічного проекту» T_3 :

$$T_3 = 73 * 0,996 * 1,26 = 92$$

Визначення коефіцієнту трудомісткості робіт на стадії «Робочий проект» та «Впровадження»:

$$k_{\Pi} = \frac{1 * 5 + 0,48 * 4 + 0,4 * 1}{5 + 4 + 1} = 0,732$$

Витрати часу для стадії «Робочий проект»:

$$T_4 = 125 * 0,732 * 1,32 * 1,08 = 130$$

Витрати часу для стадії «Впровадження»:

$$T_5 = 36 * 0,732 * 1,21 * 1,08 = 34$$

Загальні витрати людської праці на проектування системи складають:

$$T_{\Sigma} = 67 + 24 + 92 + 130 + 34 = 347 \text{ (люд-дн)}$$

Для кваліфікаційної роботи кількість робочих годин складає 530 із 7-годинним робочим днем, тому на розробку проекту виділено Φ днів:

$$\Phi = \frac{530}{7} = 75$$

Для дипломного проекту $\Phi = 75$ дн. Тоді визначаємо кількість місяців із розрахунку 25 робочих днів. Кількість місяців на розробку, M :

$$M = \frac{75}{25} = 3$$

Отже, для виконання такого проекту потрібна наступна чисельність виконавців:

$$Ч = \frac{347}{75} = 5$$

Якщо прийняти, що мінімальна заробітна плата програміста становить 11550 грн., то оплата праці всіх виконавців становить:

$$V'_1 = 3 * 5 * 11550 = 173250 \text{ (грн.)}$$

б) Витрати, пов'язані із розробкою програми на ПК
Розрахунок річного фонду часу роботи ПК в годинах.

Дійсний річний фонд часу ЕОМ у годинах рівний кількості робочих годин у році оператора, за винятком часу, затраченого на технічне обслуговування та ремонт обчислювальної техніки.

Число робочих годин у році для оператора становить: 2400 год.

$$T_{\text{ПК}} = 2400 - (6 * 8 + 5 * 12) = 2292$$

Так як, під час виконання кваліфікаційної роботи студент в середньому витрачає близько 450 год. машинного часу, то величина фонду часу ПК дорівнює:

$$T'_{\text{ПК}} = 2292 * \frac{450}{2400} = 430 \text{ (год.)}$$

Розрахунок балансової вартості ПК:

$$Ц_{\text{ПК}} = 4200 * (1 - 0,12) = 3696 \text{ (грн.)}$$

Розрахунок амортизаційних відрахувань використання ПК:

$$З_{\text{АМ}} = \frac{3696}{5} = 739 \text{ (грн.)}$$

Розрахунок витрат на електроенергію, споживану ПК:

$$З_{\text{ЕЛ}} = 0,6 * 430 * 0,9 * 0,9 = 209 \text{ (грн.)}$$

Розрахунок витрат на ремонт і технічне обслуговування ПК:

$$З_{\text{Р}} = 3696 * 0,06 = 222 \text{ (грн.)}$$

Розрахунок непрямих витрат, пов'язаних з експлуатацією ПК:

$$З_{\text{МАТ}} = 3696 * 0,05 = 185 \text{ (грн.)}$$

Поточні витрати на експлуатацію (з урахуванням умови, що роботи виконуються на власному ПК):

$$V_1'' = 739 + 209 + 222 + 185 = 1355 \text{ (грн.)}$$

Розрахунок загальних витрат на розробку програмного забезпечення:

$$V_1 = 173250 + 1355 = 174605 \text{ (грн.)}$$

7) Витрати на придбання і установку ПК, підготовку приміщення і навчання персоналу

Оскільки проект створюється на базі відділу дистанційного навчання, потреби в придбанні ПК, підготовки приміщення і навчанні персоналу немає, тому:

$$V_2 = V_3 = V_4 = 0$$

8) Загальна вартість розробки і впровадження системи

Загальна вартість розробки і впровадження системи становить:

$$V_{\Sigma} = 174605 \text{ (грн.)}$$

Річний економічний ефект становить:

$$V_p = \frac{174605}{5} = 34921 \text{ (грн.)}$$

Річний прибуток буде досягнуто за рахунок оптимізації процесу оцінювання студентів, що, в свою чергу, повинно призвести до зниження кількості відрахувань через академічну заборгованість. Орієнтовно він складатиме $\Pi_p = 37500$ грн.

В такому разі, коефіцієнт економічної ефективності розробки становить:

$$K_{EF} = \frac{37500}{34921} = 1,07$$

Термін окупності:

$$T_{OK} = \frac{1}{1,07} = 0,93$$

Отже, термін окупності інформаційної системи приблизно становить 11 місяців.

РОЗДІЛ 4. ОХОРОНА ПРАЦІ

4.1. Служба охорони праці в установі

Ключові вимоги по забезпеченню незагрозливих умов праці з інформаційною системою повинні виконуватися в межах допустимих норм, що передбачені нормативно-технічною документацією: ДСанПІН 3.3.2.007-98 «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» та НПАОП 0.00-1.28.10 «Правила охорони праці під час експлуатації електронно-обчислювальних машин».

Робота з інформаційною системою цілком побудована на взаємодії з обчислювальною технікою і, в зв'язку з цим, супроводжується таким набором шкідливих факторів як: електромагнітне випромінювання, електричний струм, статика, шум, вібрація, пожежна небезпека, постійне навантаження очей та ін.

3.1. Позичіювання технічного обладнання

У відповідності до НПАОП 0.00-1.28-10 забороняється розміщувати ЕОМ у приміщеннях, що розташовані у підвалах будинків. Для уникнення провокації можливих аварійних ситуацій внаслідок короткого замикання, поряд з комп'ютерними приміщеннями заборонене проведення ремонтних та іншого виду робіт, які потребують здійснення надмірно вологих технологічних процесів.

Згідно НПАОП 0.00-1.28-10 кількість приміщень зобов'язана бути такою, щоб на кожне робоче місце припадало не менше шести квадратних метрів площі, причому висота стелі повинна бути не менше трьох цілих та двох десятих метра, загальний об'єм – не менше двадцяти метрів кубічних.

Організація робочого місця при роботі з візуальним дисплейним терміналом (далі – ВДТ) – це одна з умов для безпечної роботи члена персоналу та повинна забезпечувати відповідність усіх елементів робочого

місця та їх розміщення до вимог ергономічності, описаних в ГОСТ 12.2.032-78 «ССБТ. Рабочее место при выполнении работ сидя. Общие эргономические требования».

При організації робочого місця, що передбачає роботу з електронним обчислювальним пристроєм, слід передбачити наступне:

- достатній для комфортної роботи простір для оператора;
- розміщення екрана терміналу у зоні, що забезпечує комфорт зорового спостереження у вертикальній площині під кутом близько тридцяти градусів від лінії зору оператора;
- можливість обертання дисплея навколо двох осей;
- екран монітору повинен бути розташований на відстані від очей оператора, що не менша шестидесяти сантиметрів;
- базові пристрої вводу інформації повинні бути розташовані на поверхні близько десяти сантиметрів від краю, зокрема повинна бути передбачена можливість її переміщення;

Організація робочого місця працівника, як користувача стаціонарного комп'ютера, гарантує відповідність усіх елементів робочого місця та їх позиціонування до характеру та особливостей трудової діяльності.

4.2. Мікроклімат

У приміщення та робочих місцях з ВДТ повинні бути забезпечені оптимальні значення параметрів мікроклімату, а саме: температури, вологості та рухливості повітря (ГОСТ 12.1.005-88, ДСН 3.3.6-042-99). Оптимальними повинні бути і категорії робіт по величинам сумарних витрат електроенергії та від теплових характеристик приміщення (ДСН 3.3.6-042-99).

Робота із інформаційною системою класифікується як категорія робіт 1А – легкі роботи, що виконуються переважно в сидячому положенні та не вимагають від виконавця значного м'язового напруження (ДСН 3.3.4.042-99).

Оптимальні параметри мікроклімату для комп'ютерних приміщень з урахуванням категорії робіт за енергозатратами організму людини та періоду року наведено в табл. 4. 1.

Таблиця 4. 1. Нормовані параметри мікроклімату для приміщень з ПК.

Період року	Категорія робіт	Температура повітря	Відносна вологість повітря	Швидкість руху повітря
Відносно холодний	Легка 1А	~23°C	~50%	0.1 м/с
Відносно теплий	Легка 1А	~24°C	~50%	0.1 м/с

4.3. Шум та вібрації

Негативний вплив вібрації та шуму на працівників може мати наступні проявлення:

- Погіршення уваги;
- Посилення відчуття втоми;
- Сповільнення реакції.

Вище перераховані фактори призводять до загального зниження працездатності та підвищення ймовірності появи аварійних ситуацій, нещасних випадків.

На робочих місцях основними джерелами шуму та вібрації можуть бути як внутрішні елементи системного блоку ПК, наприклад кулер центрального процесора, відеокарти, несправний жорсткий диск, так і периферійні пристрої вводу-виводу інформації, наприклад принтер, сканер і т. д.

У приміщеннях рівень звукового тиску та рівень звуку повинні відповідати вимогами ГОСТ 12.1.003-83 «ССБТ. Шум. Общие требования безопасности» та ДСНЗ.3.6-037-99 «Державні санітарні норми виробничого шуму, ультразвуку та інфразвуку» і не мають перевищувати шестидесяти децибел.

4.4. Вимоги до освітленості приміщень

Для створення сприятливих умов для зорової роботи, що нівелюють швидку втомлюваність органів зору, виникненню професійних захворювань, нещасних випадків, виробниче освітлення приміщень повинно відповідати вимогам:

- забезпечувати таку освітленість робочих поверхонь, що відповідає особливостям зорової роботи та не є нижчою від встановлених норм;
- забезпечувати постійне та рівномірне освітлення у робочих приміщеннях, що понизить рівень навантаження очей внаслідок ліквідації необхідності частої зорової адаптації;
- гарантувати усунення засліплюваної дії від всіх джерел світла, що перебувають в полі зору робітників;
- гарантувати усунення різких та рухомих тіней на робочій поверхні;
- гарантувати забезпечення достатнього рівня контрастності елементів робочої поверхні;
- не створювати загрозливих або шкідливих для здоров'я виробничих чинників (шум, теплові випромінювання, небезпека уражений струмом, пожежо- та вибухонебезпека світильників):
- освітлення повинно бути надійним, економічним та простим в експлуатації.

Приміщення, де розташоване робоче місце, повинно бути забезпечене хорошим природним та якісним штучним освітленням у відповідності до ДБН В.2.5-28-2006 «Інженерне обладнання будинків та споруд. Природне та штучне освітлення». Штучна система освітлення повинна бути загальною та обладнаною світильниками з хорошою світловіддачею.

Вважається, що при роботі з ІС виконується робота, що відповідає третьому розряду зорових робіт – роботи високої точності. Виходячи з цього, рівень освітленості у приміщенні повинен становити близько трьохсот Люкс.

Джерела штучного освітлення повинні бути вмонтовані в стелю. Висота їх підвісу повинна знаходитися в межах від двох до чотирьох метрів.

4.5. Вимоги пожежної безпеки

Пожежна безпека організовується відповідно до вимог ГОСТ 12.1.004-85 «ССБТ. Пожарная безопасность. Общие требования», НПАОП 0.01-1.01-95 «Правила пожежної безпеки в Україні» та інших нормативних актів.

Відповідно до НАПБ Б.03.002-2007 «Нормы определения категорий помещений, зданий и наружных установок по взрывопожарной и пожарной опасности» приміщення для експлуатації персональних ЕОМ відносяться до пожежобезпечної категорії «Д».

Будівлі та приміщення в них повинні бути не нижче другого ступеня вогнестійкості згідно з ДБН В.1.1.7-2002.

Приміщення, де розміщені робочі місця операторів, крім тих, де розташовані ЕОМ загального призначення, мають бути оснащені системою автоматичної пожежної сигналізації.

Для уникнення появи пожежі, необхідно:

- дотримуватися правил пожежної безпеки при роботі з комп'ютером та елементами освітлення;
- проводити періодичний контроль цілісності і надійності електроізоляції на вузлах живлення;
- забезпечити наявність та легкий доступ до інструкцій з пожежної безпеки;
- періодично проводити зняття зарядів статичної електрики;
- використання важко- або негорючих матеріалів в інтер'єрах приміщень;
- забезпечити наявність пристроїв автоматичного вимкнення техніки на випадок пожежі;
- забезпечення наявності первинних засобів пожежогасіння (вогнегасників у кількості не менше двох штук).

4.6. Електробезпека

Вимоги електробезпеки у приміщеннях, де встановленні ЕОМ і ПК, відображенні у НПАОП 0.00-1.28-10.

Приміщення з ЕОМ, за ступенем ураження електричним струмом, відносяться до класу приміщень з підвищеною небезпекою.

Заземлені конструкції в приміщеннях, де розташовані робочі місця операторів, повинні бути захищені діелектричними щитками або сітками.

Обчислювальна техніка, елементи освітлення, проводи живлення та кабелі за технічними характеристиками і рівнем захисту мають відповідати класу зони за НПАОП 40.1-1.01-97, мати апаратуру захисту від струму короткого замикання та інших аварійних режимів.

Під час прокладання та експлуатації ліній електромережі потрібно повністю унеможливити появу електричного джерела загоряння внаслідок короткого замикання чи перевантаження проводів.

ВДТ є джерелом статичної електрики. Тривале перебування людини в електростатичному полі призводить до появи захворювань леденів, порушень у роботі серцево-судинної та нервової систем і т. п.

Напруженість електростатичного поля на робочих місцях не повинна перевищувати показника в двадцять кіловольт на метр, відповідно до ГОСТ 12.1.045-84 «ССБТ. Электростатические поля. Допустимые уровни на рабочих местах и требования к проведению контроля».

Для захисту від статички потрібно проводити регулярні вологі прибирання приміщення; провітрення приміщення; очищення робочого місця та ПК від пилу.

Із ціллю підвищення рівня електробезпеки все обладнання в приміщенні повинно мати заземлення. Відповідно до вимог ГОСТ 12.1.030-81 електричний опір системи захисного заземлення не повинен перевищувати показника в чотири Оми.

ВИСНОВКИ

У процесі виконання кваліфікаційної роботи було проведене дослідження процесу організації навчання в умовах дистанційного навчання, зокрема організацію контролю знань з навчальних дисциплін.

Був проведений системний аналіз даного виду діяльності, на основі якого були побудовані функціональні моделі «AS-IS» та «TO-BE», що відображають поточний спосіб організації дистанційного навчання і контролю знань до та після впровадження розробленого програмного забезпечення.

Підставною для розробки ІС став незадовільний стан користувацького інтерфейсу платформи дистанційного навчання та деяких функціональних аспектів.

Серед особливостей розробленої інформаційної системи можна виділити наступні: новий порядок реєстрації користувачів, адаптивний інтерфейс сайту, можливість створення та редагування тестових форм, можливість проходження тестування та отримання звіту по результатам тестування.

Online тести реалізовано у форматі веб-сайту, що дозволяє отримати доступ до них з будь-якого пристрою, на якому встановлений веб-браузер та є доступ до інтернету.

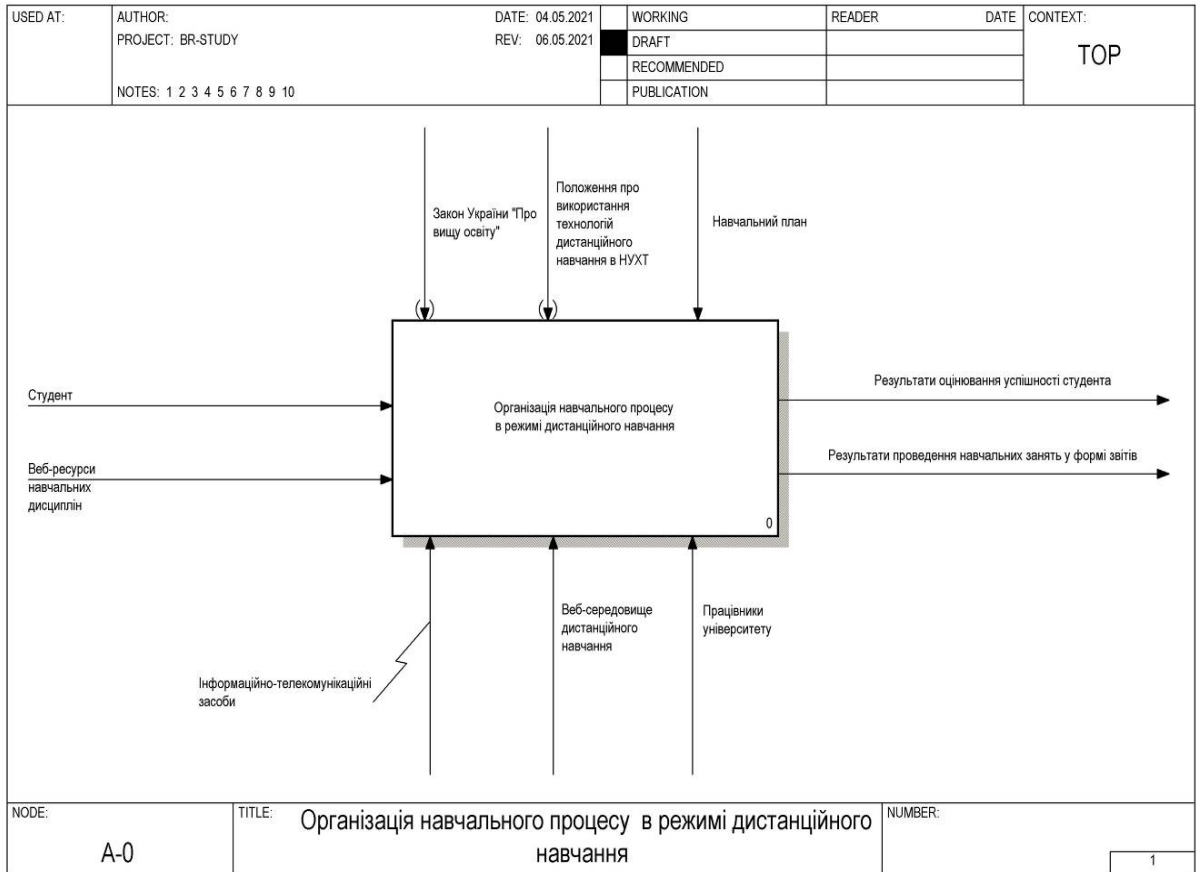
Програмне рішення створює зручні умови для проведення контролю знань студентів. Його використання викладачами університету дозволить отримати доступ до актуальної та об'єктивної інформації про поточний рівень знань студентів, а студентам – проходити тестування однаково зручно як на стаціонарному комп'ютері, так і на мобільному телефоні.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

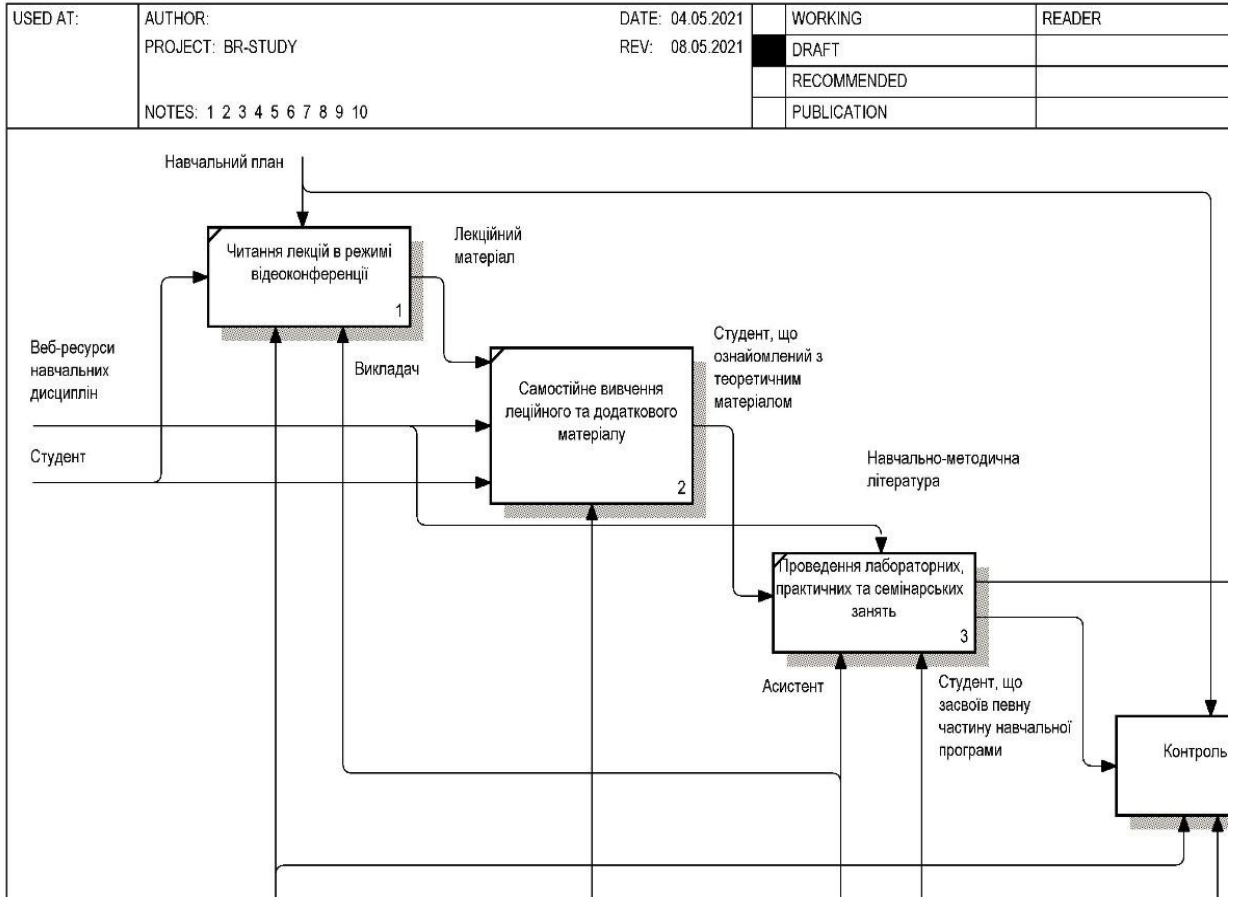
1. Статут Національного університету харчових технологій / [Ідентифікаційний код 2070938] – [Наказ МОН України від 11.11.2020. № 1397]
2. Наказ МОН України 25.04.2013 №466 «Про затвердження Положення про дистанційне навчання».
3. МЕТОДИЧНА РОЗРОБКА «Форми і методи контролю знань в умовах сучасних навчальних технологій»/ [Розробив: вик. Янченко О.І.] – [Інгулецький технікум Криворізького технічного університету]
4. НУХТ: Національний університет харчових технологій [Електронний ресурс] : [Веб-сайт] – Режим доступу: <https://nuft.edu.ua>
5. Обзор сервисов для онлайн-тестирования [Електронний ресурс] : [Веб-сайт] – Режим доступу: <https://spark.ru/startup/anketolog/blog/64613/obzor-servisov-dlya-onlajn-testirovaniya>
6. Національний університет харчових технологій [Електронний ресурс] : [Матеріал з Вікіпедії - вільної енциклопедії] – Режим доступу: https://uk.wikipedia.org/wiki/Національний_університет_харчових_технологій
7. Положення про використання технологій дистанційного навчання в Національному університеті харчових технологій / [Додаток № 3 до наказу № 100 від 03.09.2020]
8. Bootstrap Documentation [Електронний ресурс] : [Веб-сайт] - Режим доступу: <https://getbootstrap.com/docs/5.0/getting-started/introduction/>
9. Flask Documentation [Електронний ресурс] : [Веб-сайт] - Режим доступу: <https://flask.palletsprojects.com/en/2.0.x>
10. Закон України «Про вищу освіту»

ДОДАТКИ

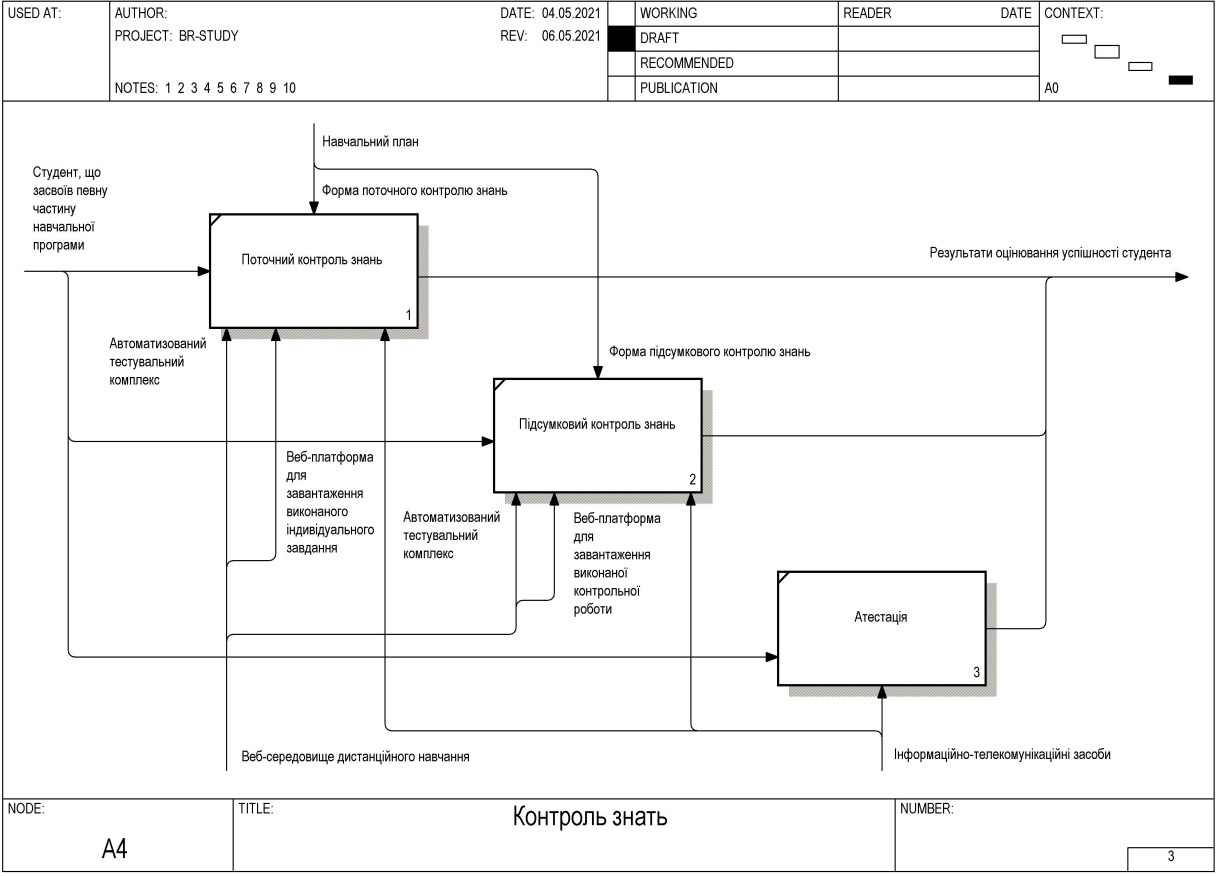
Додаток А Функціональна модель організації процесу дистанційного навчання.



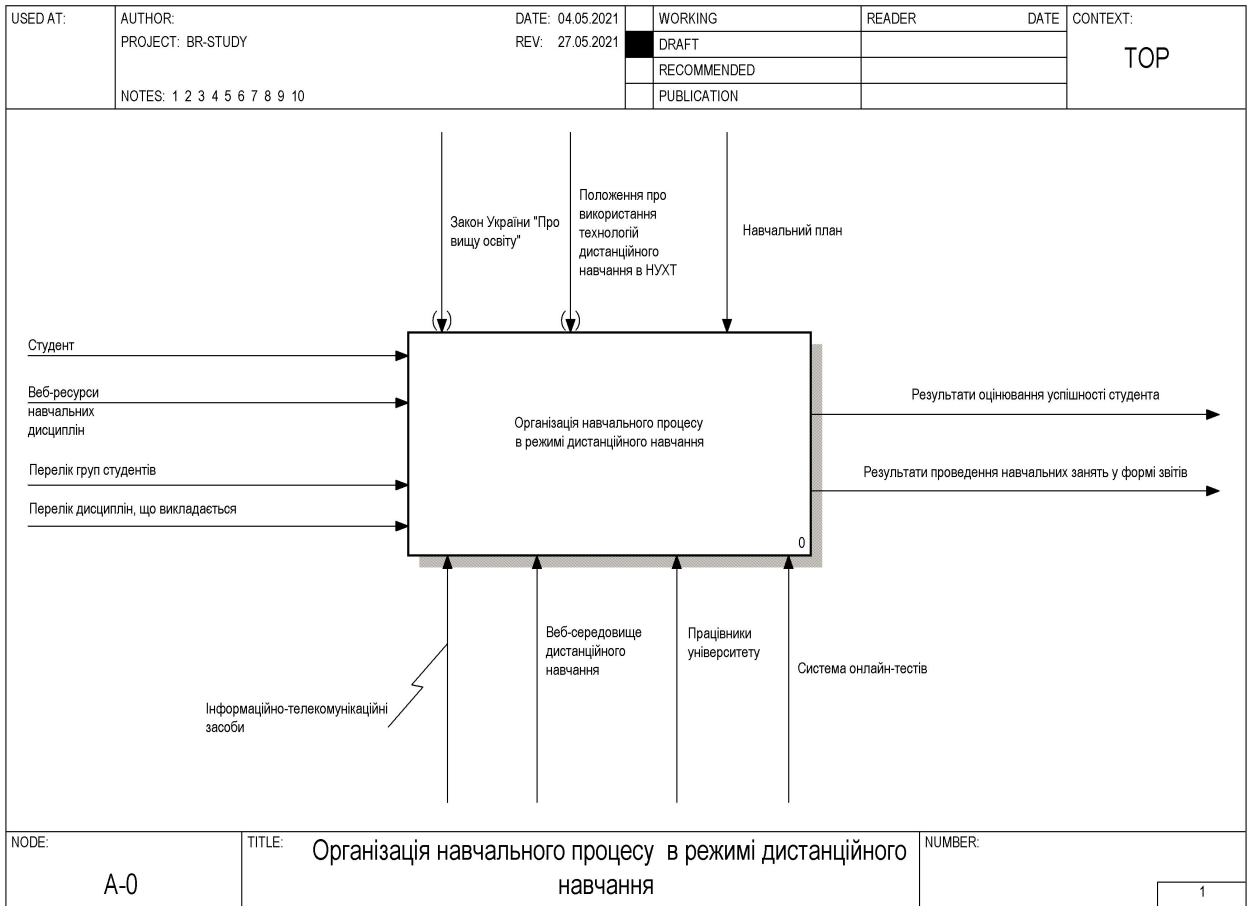
Додаток В Функціональна модель організації процесу дистанційного навчання, перший рівень декомпозиції.



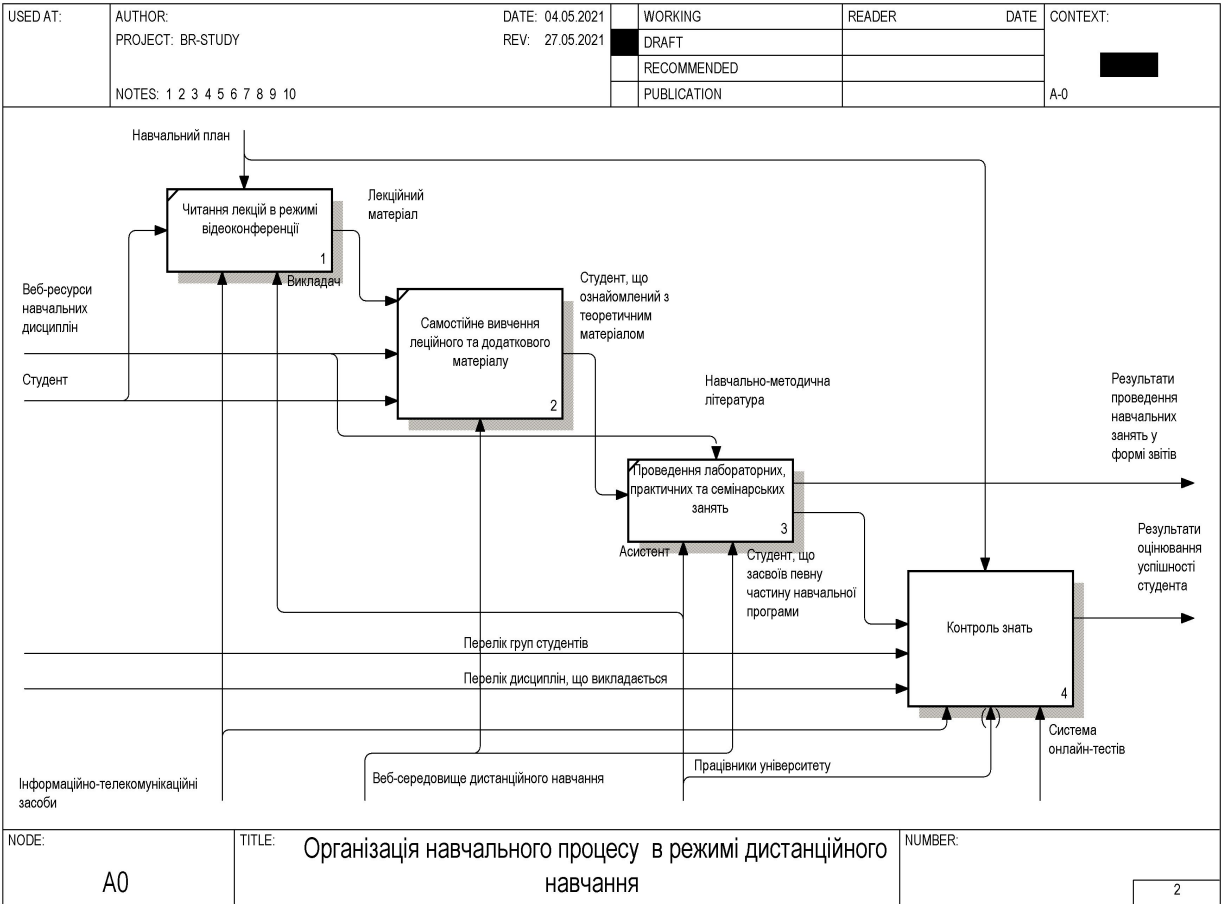
Додаток С Функціональна модель організації процесу дистанційного навчання, другий рівень декомпозиції.



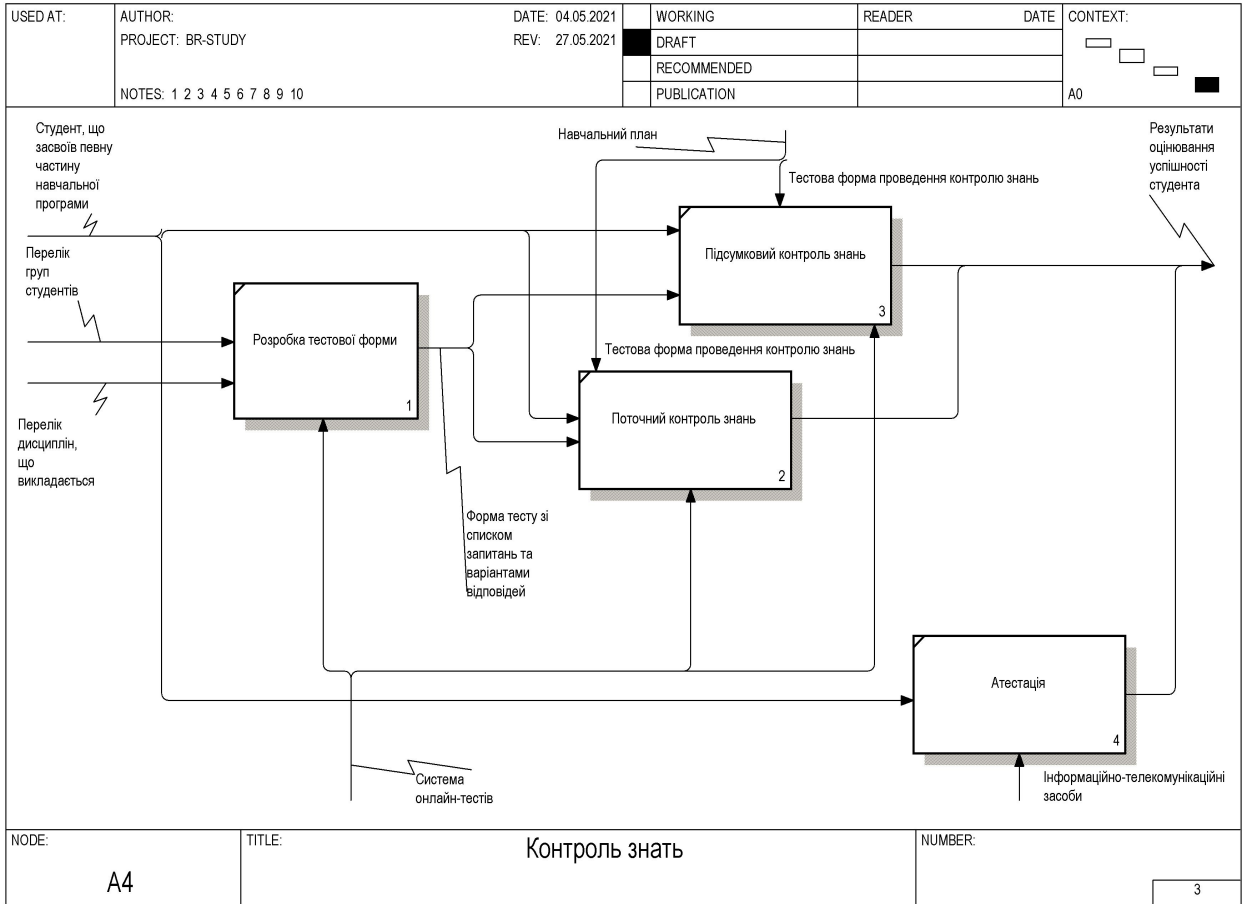
Додаток D Функціональна модель організації процесу дистанційного навчання, з урахуванням використання розроблюваної системи, перший рівень декомпозиції.



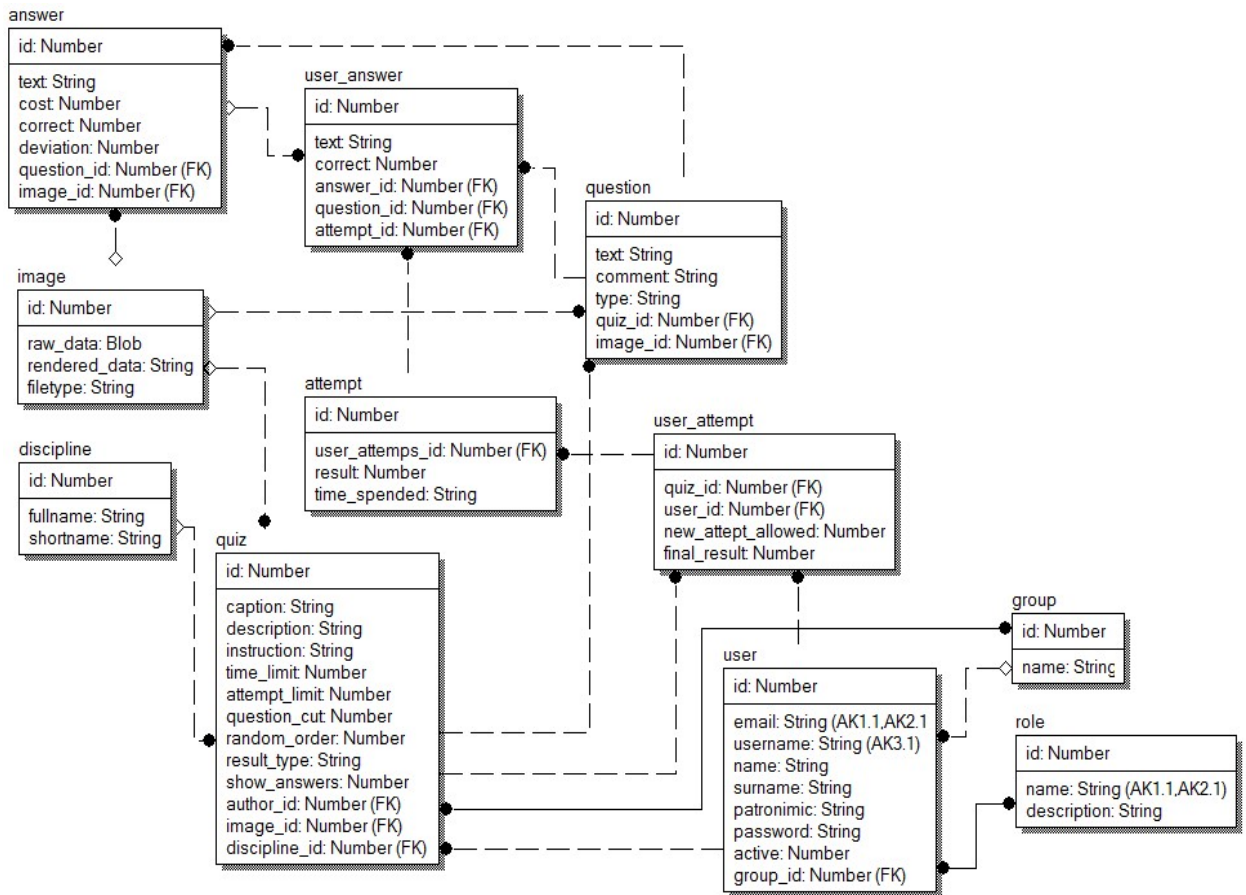
Додаток Е Функціональна модель організації процесу дистанційного навчання, з урахуванням використання розробленої системи, другий рівень декомпозиції.



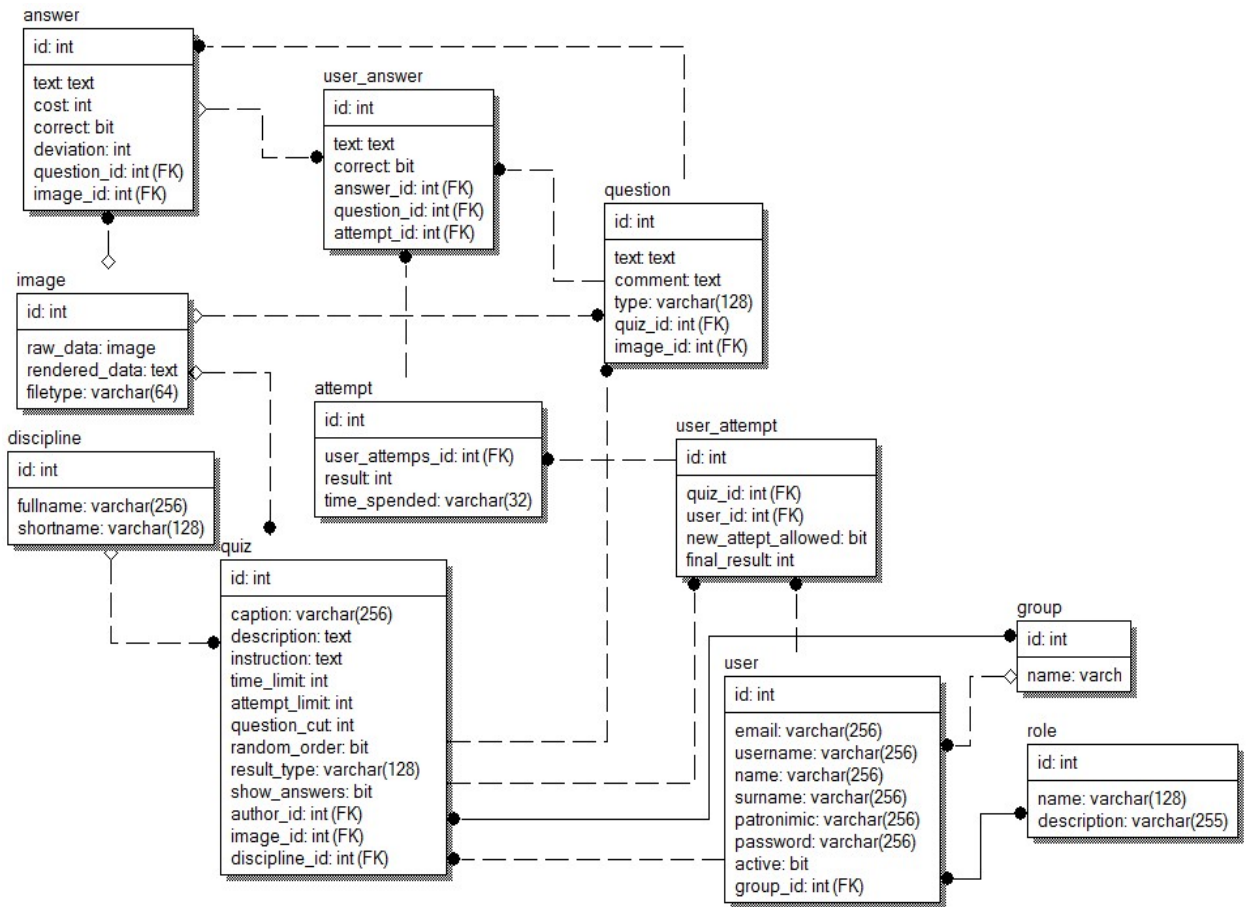
Додаток F Функціональна модель організації процесу дистанційного навчання, з урахуванням використання розроблюваної системи, третій рівень декомпозиції.



Додаток G Логічна модель бази даних.



Додаток Н Фізична модель бази даних.



Додаток J Згенерована модель даних в СУБД MS SQL Server

