

Київ – 2022 р.

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ХАРЧОВИХ ТЕХНОЛОГІЙ
Інститут (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

ЗАТВЕРДЖУЮ

Завідувач кафедри Чумаченко С.М.

“ ___ ” _____ 20__ року

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧА

Нижнику Олександрю Олександровичу

(прізвище, ім'я, по батькові)

1. Тема роботи Розроблення модуля формування анкет в системі анкетування НУХТ

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 “ ___ ” _____ року № _____

2. Строк подання здобувачем роботи 12 червня

3. Вихідні дані до роботи 1. Дані про освітньо-професійні програми

2. Дані про групи здобувачів освіти

3. Інформація про дисципліни в поточному семестрі

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

1. Системний аналіз НУХТ

2. Моделі бази даних ІС

3. Інтерфейс ІС

5. Перелік графічного матеріалу

1. Організаційна структура університету

2. Функціональна та концептуальна моделі

3. Логічна та фізична моделі бази даних

4. Приклади роботи ІС (інтерфейс користувача)

5. Фрагменти коду

6. Консультанти розділів роботи

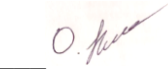
Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	Горлова Т.М., доцент	15.05.22	02.06.22
2	Горлова Т.М., доцент	02.06.22	12.06.22

7. Дата видачі завдання 15.05.2022

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів виконання кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Системний аналіз підприємства та постановка задачі на проектування	15.05.2022 - 19.05.2022	Виконано
2	Проектування БД	19.05.2022 - 25.05.2022	Виконано
3	Створення ІС	25.05.2022 - 02.06.2022	Виконано
4	Написання інструкцій користувача	02.06.2022 - 07.06.2022	Виконано
5	Оформлення пояснювальної записки	07.06.2022 - 09.06.2022	Виконано
6	Оформлення презентації	09.06.2022-12.06.2022	Виконано

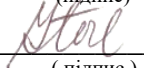
Здобувач


(підпис)

Нижник О.О.

(прізвище та ініціали)

Керівник роботи


(підпис)

Горлова Т.М.

(прізвище та ініціали)

АНОТАЦІЯ

Нижник О.О. розроблення модуля формування анкет в системі анкетування НУХТ

Дипломний проект складається з 78 сторінок, 9 джерел та 6 додатків.

В дипломному проекті проведено системний аналіз Національного університету харчових технологій та модуль формування анкет, використовуючи актуальні технології для створення функціональної моделі, проектування бази даних та створення інтерфейсу, а саме: AllFusion Process Modeler r7, ERwin Data Modeler, MY SQL.

Метою проекту є опанування методів розв'язання задач прикладного характеру на основі отриманих знань, умінь та навичок відповідно до стандартів вищої освіти.

Ключові слова: системний аналіз, функціональна схема, база даних, інформаційна система, інтерфейс, SQL, таблиці, форма, запит, звіт, анкетування, опитування.

SUMMARY

Nyzhnyk O.O. development of the module of questionnaire formation in the NUHT questionnaire system

The diploma project consists of 78 pages, 9 sources and 6 appendices.

The diploma project conducted a systematic analysis of the National University of Food Technology and a module of questionnaires, using current technologies to create a functional model, database design and interface, namely: AllFusion Process Modeler r7, ERwin Data Modeler, MY SQL.

The aim of the project is to master the methods of solving problems of applied nature on the basis of acquired knowledge, skills and abilities in accordance with the standards of higher education.

Keywords: system analysis, functional diagram, database, information system, interface, SQL, tables, form, query, report.

ЗМІСТ

ВСТУП.....	8
РОЗДІЛ 1. СИСТЕМНИЙ АНАЛІЗ РОБОТИ ВІДДІЛУ ЯКОСТІ ОСВІТИ НУХТ	10
1.1 Загальна характеристика НУХТ	10
1.2 Організаційна структура НУХТ, роль і взаємодія підрозділів.....	11
1.2.1. Загальна схема організаційної структури НУХТ	11
1.2.2. Структура відділу контролю якості освіти.....	13
1.2.3. Структура системи внутрішнього забезпечення якості вищої освіти.....	14
1.2.4. Основні функції та завдання відділу контролю якості	18
1.3. Аналіз нинішнього стану комп'ютеризації відділу контролю якості освіти НУХТ	21
1.4. Розроблення функціональної моделі та аналіз існуючих бізнес – процесів	21
1.4.1. Функціональна модель проведення анкетування НУХТ	21
1.4.2. Виявлені проблеми.....	23
1.4.3. Задачі автоматизації.....	24
1.5. Огляд існуючих рішень для розв'язання виявлених проблем.	24
1.5.1. Google Forms.....	24
1.5.2. Survio.....	24
1.5.3. Simpoll.....	25
1.5.4. Порівняння систем – аналогів	25
1.6. Обґрунтування доцільності проектування й розроблення системи анкетування НУХТ	26
1.7. Концептуальна модель системи	27

РОЗДІЛ 2. ОПИС КОМПЛЕКСУ ЗАДАЧ АВТОМАТИЗАЦІЇ	29
2.1. Інформаційне забезпечення системи.....	29
2.2. Алгоритмізація та реалізація комплексу задач автоматизації.....	35
2.3. Інструкція користувача	53
3.4. Технічне та системне забезпечення розробки.....	55
3.4.1. Обґрунтування вибору технічних засобів	55
3.4.2. Обґрунтування вибору ОС та протоколу обміну даними	57
3.4.3. Заходи захисту від несанкціонованого доступу до систему	58
ВИСНОВКИ.....	60
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	61
ДОДАТКИ.....	62
Додаток А	62
Додаток Б	65
Додаток В	68
Додаток Г	69
Додаток Д	69
Додаток Е	74

ВСТУП

На теперішній час дуже важливим фактором освіти є її якість. Забезпечення якості вищої освіти передбачає створення належних умов для особистісного і професійного розвитку здобувача (студента, слухача), успішної реалізації ним освітньо-виховних цілей і досягнення запланованих результатів. Місія сучасної вищої школи полягає в тому, щоб створити для кожного студента сприятливе освітнє середовище, спрямувати його на самостійну навчально-пізнавальну, науково-дослідницьку, професійно-практичну діяльність. А також зорієнтуватися в конкурентному середовищі, визначитися у власних освітньо-професійних пріоритетах, побудувати індивідуальну траєкторію навчання і виховання, виробити дієвий алгоритм реалізації поставлених цілей. Таким чином, стратегія «забезпечення якості вищої освіти» зумовлює взаємопов'язану і взаємоузгоджену діяльність усіх суб'єктів освітнього процесу, що підпорядковується досягненню спільної мети.

Провідна роль у забезпеченні якості вищої освіти належить ЗВО, на які покладено завдання формування власної стратегії якісної освітньої діяльності, впровадження і постійного вдосконалення внутрішніх механізмів її забезпечення, реалізації просвітницької місії, спрямованої на пропагування загальноєвропейських академічних ідеалів і цінностей, виховання культури якості, залучення до співпраці всіх зацікавлених сторін.

До основних чинників, які обумовлюють ефективність системи забезпечення якості вищої освіти, відносяться:

1. Ставлення до вищої освіти як найвищої цінності, запоруки суспільного прогресу, національної безпеки, конкурентоспроможності держави, вишів, окремої особистості.

2. Належне фінансування закладів вищої освіти, гідна оплата праці науково-педагогічних працівників, які реалізують освітньо-виховні цілі та забезпечують якість викладання.

3. Просвітницька діяльність ЗВО, постійне інформування здобувачів, громадськості, працедавців, інших зацікавлених сторін про власну стратегію забезпечення якості, спільні завдання і заходи щодо підвищення ефективності освітнього процесу.

4. Пропагування академічної доброчесності, нетерпимість до будь-яких форм її порушення (копіляція, плагіат, фальсифікація), практичне впровадження загальнонаціональної системи антиплагіату та покарання за різні прояви академічної недоброчесності (держслужбовців і чиновників усіх рангів, керівників ЗВО, викладачів, науковців, здобувачів вищої освіти).

5. Особистісна відповідальність стейкхолдерів за результати освітньої діяльності (учіння, викладання, організацію професійно-практичної підготовки, ухвалення управлінських рішень).

6. Постійний моніторинг ефективності систем зовнішнього і внутрішнього забезпечення якості вищої освіти з боку держави, суспільства, громадських організації, незалежних агенцій.

7. Прозорість механізмів і заходів щодо забезпечення якості вищої освіти, відкритий доступ до інформації про діяльність відповідних органів управління, закладів вищої освіти, наукових установ, пов'язану із удосконаленням систем зовнішнього і внутрішнього забезпечення якості [1].

Тому актуальною задачею є створення системи анкетування студентів для оцінки якості викладання навчальних дисциплін.

РОЗДІЛ 1. СИСТЕМНИЙ АНАЛІЗ РОБОТИ ВІДДІЛУ ЯКОСТІ ОСВІТИ НУХТ

1.1 Загальна характеристика НУХТ

Історія університету – це яскраве відображення економічного розвитку країни, її науково-технічного прогресу, суспільно-політичного та культурного життя.

Виникнення спеціалізованого навчального закладу пов'язане, з одного боку, з розвитком освіти в Російській імперії, створенням Київського відділення Російського імператорського технічного товариства, а з іншого – необхідністю підготовки інженерно-технічних кадрів вищої і середньої ланки для бурякоцукрового виробництва, галузі, що на той час найбільш інтенсивно розвивалася.

Таким чином, історія Національного університету харчових технологій, витоки якої сягають другої половини ХІХ століття, тісно пов'язана з історією розвитку цукрової промисловості.

Прийняття Положення про училища дозволило Київському відділенню Російського імператорського технічного товариства на своєму засіданні 15 лютого 1884 р. ухвалити рішення про відкриття у місті Сміла технічних класів на базі училища графів Бобринських. З цієї дати починається історія національного університету харчових технологій.

Сьогодні Національний університет харчових технологій – провідний вищий навчальний заклад, у якому здійснюється підготовка висококваліфікованих фахівців для всіх галузей господарства.

Зберігаючи та розвиваючи позитивний досвід вітчизняної освіти, університет сприймає і запроваджує кращі світові стандарти підготовки фахівців. Це забезпечує соціальну захищеність випускників, підвищений попит на них у промисловості та бізнесі. Постійно зростає також конкурс абітурієнтів при щорічному наборі студентів [2].

Метою університету є формування інноваційного, освітнього і наукового простору для забезпечення розвитку потенціалу та можливостей самореалізації університетської спільноти в процесі їх спільної освітньої, наукової, інноваційної та організаційної діяльності; підготовка висококваліфікованих, конкурентоспроможних фахівців для забезпечення потреб суспільства, ринку праці та держави, здійснення культурно-просвітницької діяльності, утвердження культурних і загальнолюдських цінностей.

1.2 Організаційна структура НУХТ, роль і взаємодія підрозділів.

1.2.1. Загальна схема організаційної структури НУХТ

Політика Університету щодо організації внутрішнього забезпечення якості вищої освіти передбачає спрямування діяльності учасників освітнього процесу на досягнення кількісних індикаторів якості, які є складовими переліку ключових показників ефективності для зазначених в пункті 1.2.1 суб'єктів другого, третього, четвертого та п'ятого рівнів системи.

Відповідно до схеми наведеної на рис. 1.1. можна зрозуміти що управління університетом відбувається згори донизу, тобто ректор видає певний наказ, далі він розповсюджується по факультетах, які в свою чергу передають його на свої кафедри, після чого викладачі цих кафедр доносять його до студентів.



Рис 1.1. Організаційна структура НУХТ

1.2.2. Структура відділу контролю якості освіти

На рисунку 1.2 представлена структура відділу контролю якості.

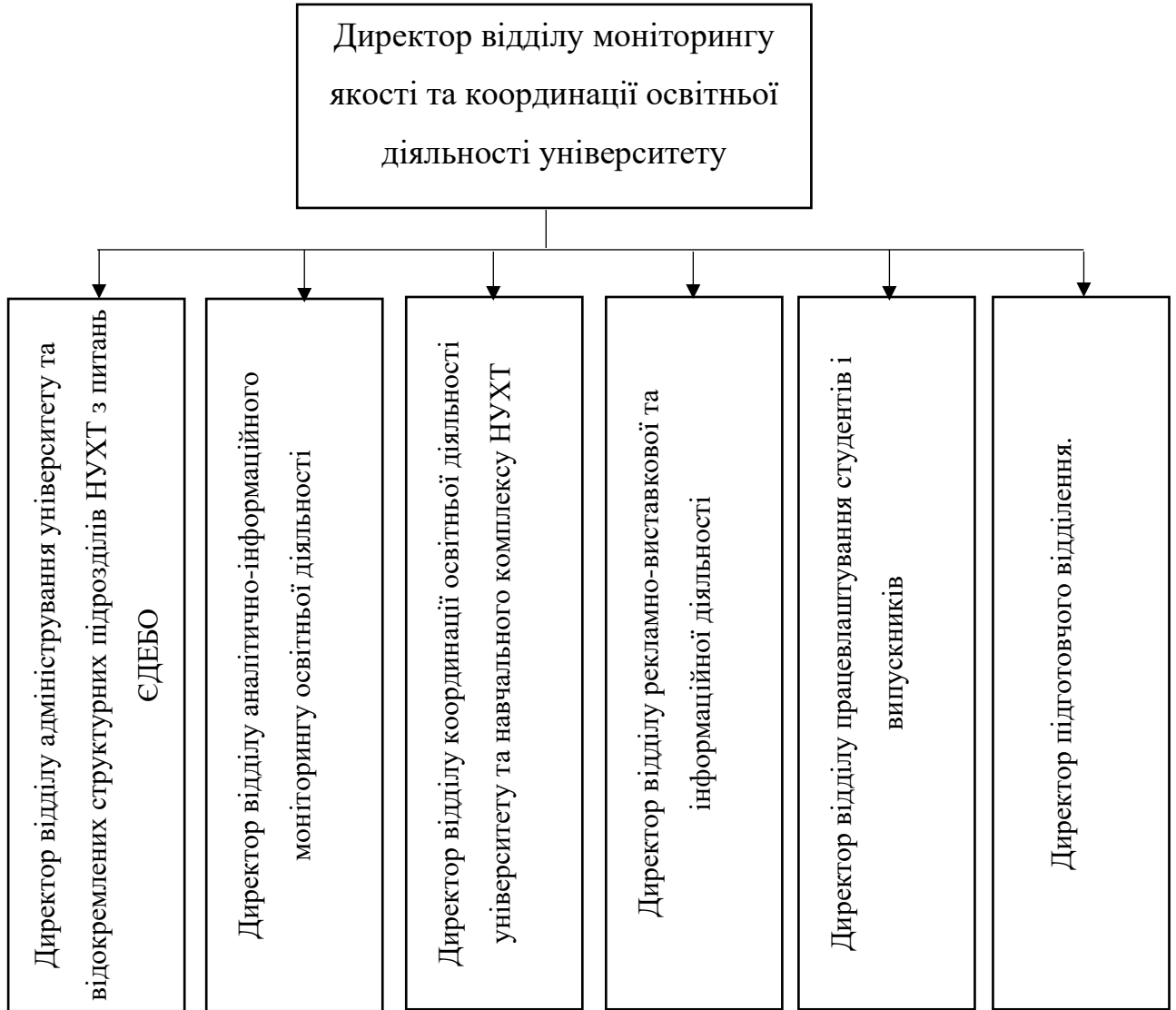


Рис. 1.2. Організаційна структура відділу контролю якості освіти

1.2.3. Структура системи внутрішнього забезпечення якості вищої освіти

Система внутрішнього забезпечення якості вищої освіти в Університеті функціонує на таких рівнях:

- перший - здобувачі вищої освіти;
- другий - кафедра (завідувачі кафедр, гаранті освітніх програм, науково-педагогічні працівники, куратори академічних груп);
- третій - факультет/інститут (декан факультету/директор інституту, заступники декана/директора, члени науково-методичної комісії факультету/інституту, Вчена рада факультету, студентське самоврядування факультету/інституту);
- четвертий - загальноуніверситетські структурні підрозділи (Центр моніторингу якості та координації освітньої діяльності, навчальний відділ навчально-методичного управління, навчально-методичний відділ навчально-методичного управління, центр інформаційних технологій, науково-технічна бібліотека, відділ міжнародних зв'язків, студентська рада тощо);
- п'ятий - ректорат, науково-методична рада Університету, Вчена рада Університету.

На першому рівні організації системи внутрішнього забезпечення якості здійснюються опитування здобувачів вищої освіти щодо:

- змісту освітніх програм та їх компонентів за спеціальностями, якості організації викладання дисциплін науково-педагогічними працівниками Університету;
- якості функціонування освітнього середовища, у тому числі щодо забезпечення можливості формування індивідуальної освітньої траєкторії, задоволеності організацію практичної підготовки тощо;
- ефективності діяльності кафедр, деканатів/дирекцій, студентської ради Університету (факультету), науково-технічної бібліотеки тощо).

Другий рівень організації системи внутрішнього забезпечення якості здійснюється науково-педагогічними працівниками кафедри за безпосереднім керівництвом завідувачів кафедри та гарантів освітніх програм за такими напрямками:

- напрацювання власної практики якісного забезпечення освітніх компонентів освітніх програм, що реалізуються кафедрою;
- контроль виконання вимог і встановлених нормативів щодо якісної організації освітньої діяльності ;
- моніторинг поточних і підсумкових результатів навчання здобувачів вищої освіти;
- забезпечення наявності необхідних навчальних і навчально-методичних ресурсів, баз практики для організації освітнього процесу, у тому числі самостійної роботи здобувачів вищої освіти;
- дотримання принципів академічної доброчесності, запровадження системи заходів запобігання наявності та виявлення плагіату на всіх рівнях і у будь-яких формах;
- залучення до освітнього процесу провідних фахівців галузей, роботодавців, які мають авторитет у відповідній галузі або регіоні та є лідерами ринку для проведення теоретичних і практичних занять на відповідних освітніх програмах;
- напрацювання ефективної системи взаємозв'язків із випускниками та потенційними роботодавцями, що дозволять створити надійний зв'язок з ринком праці.

Третій рівень організації системи внутрішнього забезпечення якості вищої освіти реалізується на факультеті/інституті під безпосереднім керівництвом декана/директора та полягає в:

- внутрішній перевірці якості підготовки здобувачів вищої освіти, системному моніторингу результатів навчання;

- забезпеченні функціонування інформаційної системи факультету/інституту (ПС-деканат) для ефективного управління освітнім процесом;
- управлінні якістю, у тому числі на основі контролю досягнень за освітніми програмами на усіх рівнях вищої освіти;
- організація моніторингу освітніх програм на предмет їх відповідності вимогам ринку праці та тенденціям розвитку галузі й спеціальності, із залученням усіх учасників освітнього процесу та різних категорій стейкхолдерів;
- контролі підготовки та проходження процедур зовнішнього оцінювання якості вищої освіти (через акредитацію освітніх програм).

На четвертому рівні системи внутрішнього забезпечення якості структурними підрозділами Університету здійснюються процедури та заходи, які підтверджують, що усі вимоги до якості вищої освіти виконуються:

- узагальнення та поширення кращих практик формування освітніх програм у межах Університету, своєчасне виявлення негативних тенденцій, допомога у формуванні відомостей самооцінювання для акредитації освітніх програм;
- планування та організація заходів щодо підвищення якості вищої освіти (діяльність, спрямована на реалізацію стратегії, політики, цілей та вимог до якості вищої освіти);
- нормативно-правовий супровід процедур внутрішнього забезпечення якості в Університеті;
- щорічне оцінювання здобувачів вищої освіти (за підсумками складання екзаменаційних сесій), науково-педагогічних і педагогічних працівників Університету (за підсумками рейтингування) та регулярне оприлюднення результатів таких оцінювань на офіційному вебсайті;
- забезпечення підвищення кваліфікації та професійного розвитку педагогічних, наукових і науково-педагогічних працівників;

- удосконалення системи зворотнього зв'язку з випускниками Університету, роботодавцями, дослідження ринку праці;
- створення умов для формування індивідуальної освітньої траєкторії здобувача вищої освіти, забезпечення прозорості процедур визнання результатів навчання, отриманих в інших закладах вищої освіти та неформальній освіті;
- сприяння факультетам/інститутам і кафедрам у підвищенні якості вищої освіти за спеціальностями та надання необхідної фактологічної аналітичної інформації у проведенні моніторингу для вдосконалення освітніх програм, навчальних планів, робочих програм навчальних дисциплін щодо їх відповідності вимогам ринку праці, потребам здобувачів вищої освіти за участі учасників освітнього процесу;
- моніторинг іміджу Університету в освітньому просторі регіону, України, світу тощо.

На п'ятому рівні системи внутрішнього забезпечення якості ректоратом, науково-методичною радою Університету, Вченою радою Університету забезпечується :

- напрацювання університетської філософії впровадження культури якості на всіх рівнях;
- формування політики забезпечення якості вищої освіти в Університеті;
- моніторинг світових та загальноукраїнських трендів ринку праці, у тому числі з врахуванням регіонального контексту;
- затвердження локальних нормативних актів щодо забезпечення реалізації цієї політики;
- контроль процедур і заходів щодо виконання вимог до якості вищої освіти в Університеті [3].

1.2.4. Основні функції та завдання відділу контролю якості

Відділ контролю якості виконує такі завдання:

1. Реалізація політики Університету щодо забезпечення якості освітніх послуг, практична реалізація внутрішньої системи забезпечення якості освітньої діяльності Університету.
2. Вдосконалення системи забезпечення якості освітньої діяльності з урахуванням завдань Стратегії розвитку та Програми інноваційного розвитку Університету, визначення системи кількісно-якісних показників діяльності Університету згідно з європейськими стандартами якості, пріоритетами національних та міжнародних рейтингів закладів вищої освіти.
3. Координація роботи з періодичного перегляду освітніх програм, відповідно до стандартів вищої освіти в Україні, особливостей забезпечення якості в Європейському просторі вищої освіти та підготовка відповідних матеріалів для розгляду на Науково-методичній та Вченій радах Університету.
4. Узагальнення й аналіз результатів діяльності структурних підрозділів Університету із забезпечення якості освітньої діяльності, формування культури якості, впровадження принципів академічної доброчесності.
5. Організація та проведення науково-методичних заходів (семінарів, тренінгів, круглих столів) з актуальних питань забезпечення якості освітньої діяльності, впровадження освітніх інновацій, новацій в ЄДЕБО.
6. Розроблення та/або впровадження в освітній процес нормативних документів (внутрішніх нормативних документів та наказів, розпоряджень МОН України тощо), що регламентують питання, пов'язані з якістю освітньої діяльності та вищої освіти.
7. Контроль виконання Указів Президента, Постанов та рішень Кабінету Міністрів України, наказів та розпоряджень Міністерства освіти і науки України, Національного агентства із забезпечення якості вищої освіти, Вченої ради Університету з питань запровадження та моніторингу освітніх програм

підготовки здобувачів вищої освіти, ліцензування та акредитації освітньої діяльності Університету.

8. Взаємодія та співпраця з Національним агентством із забезпечення якості вищої освіти, провідними закладами вищої освіти, вивчення та узагальнення їх досвіду щодо діяльності експертних служб та моніторингових досліджень.

9. Взаємодія та співпраця з національним технічним комітетом стандартизації (ТК 163 «Якість освітніх послуг») щодо впровадження стандартів і рекомендацій по забезпеченню якості в Європейському просторі вищої освіти (ESG).

10. Забезпечення підтримки високого іміджу Університету. Надання консультативної та методичної допомоги учасникам освітнього процесу з питань забезпечення якості освіти [4].

Виконання даних завдань є неможливим без взаємодії між іншими відділами. Розглянемо яку інформацію відділ отримує, а яку передає іншим (Табл.1.1.)

Таблиця 1.1. Взаємозв'язки різних відділів університету з відділом контролю якості освіти

№	Підрозділ	Отримання	Надання
1	Факультети університету	Звіти про роботу факультетів	Рекомендації щодо покращення навчальної діяльності на факультетах Організація та проведення науково-методичних заходів з актуальних питань забезпечення якості освітньої діяльності
2	Дирекція університету	Звіти про роботу університету	Кількісно-якісні показники діяльності Університету згідно з європейськими стандартами якості
3	Учасники освітнього процесу		Надання консультативної та методичної допомоги з питань забезпечення якості освіти
4	Вчена рада університету	Поточні освітні програми	Кориговані освітні програми

1.3. Аналіз нинішнього стану комп'ютеризації відділу контролю якості освіти НУХТ

На даний момент у відділі функціонує система проведення анкетування. Система передбачає, що користувач повинен зареєструватися в ній, після чого буде доступна можливість створення нових анкет. Після створення анкети викладач отримує посилання на анкету, яке направляє студентів на гугл форму анкети. Через 7 днів анкетування автоматично припиняється, після чого доступний перегляд результатів.

Головним недоліком існуючої системи є те що вона направлена на створення 1 анкети, а необхідно формувати різні анкети і проводити опитування. Ще одним недоліком є те, що при створенні анкети викладач сам повинен ввести назви груп студентів, таким чином він може допустити помилку. Враховуючи що вона написана з використанням nodejs та mongodb її складно підтримувати та не можливо розгорнути на серверах НУХТ.

Крім того, що необхідно забезпечити інтеграцію даних з інших інформаційних систем, що забезпечить вірне проведення опитування усіх здобувачів освіти, а також обробку результатів.

1.4. Розроблення функціональної моделі та аналіз існуючих бізнес – процесів

1.4.1. Функціональна модель проведення анкетування НУХТ

Важливим етапом дослідження роботи відділу контролю якості освіти є побудова функціональної моделі «AS-IS»(як зараз є). Побудова моделі була виконана за допомогою програми AllFusion Process Modeler r7 в нотації IDEF0. Функціональна модель відділу контролю якості освіти показує та детально описує основні процеси діяльності відділу. Всі роботи, вхідні та вихідні величини, механізми та алгоритми обробки даних у відділі позначаються за допомогою стрілок та блоків. Настільки детальний розгляд системи дозволяє не тільки дослідити логіку функціонування системи але й дослідити низку питань

відносно системи, наприклад: дослідити ефективність роботи відділу, сформувані ідеї для підвищення ефективності та швидкості роботи відділу.

Функціональна модель відділу описує процес проведення опитування так як воно проводиться в даний момент (AS-IS) з точки зору викладача. Контекстна діаграма у нотації IDEF0 утворена з наступних елементів (Додаток А Рис.1.):

- Основний блок «Організація проведення опитування НУХТ»;
- Три вхідні стрілки – «Дані про освітньо-професійні програми», «Дані про групи здобувачів», «Інформація про дисципліни в поточному семестрі»;
- Чотири стрілки контролю – «Вимоги контролю якості освіти», «Накази МОН», «Статут НУХТ» «Нормативні документи університету»;
- Чотири стрілки механізмів – «Керівництво НУХТ», «Працівники відділу якості освіти», «Викладачі», «Здобувачі освіти»;
- Чотири стрілки виходу – «Затверджена анкета», «Наказ про проведення опитування», «Результат опитування», «Звіт про проведення опитування».

Діаграма першого рівня декомпозиції моделі (Додаток А Рис.2.) складається з:

- П'яти блоків процесу «Формування та завантаження анкети», «Організація опитування», «Формування опитування для навчальних груп», «Проведення опитування», «Обробка результатів опитування та формування звіту»;
- Стрілки контролю, механізмів, входів та виходів аналогічні, проте додалися проміжні стрілки «Обрана анкета», «Посилання на форму анкетування», «Сировина», які перетворюються у стрілки виходу, що були по замовчуванню.

Далі йде діаграма декомпозиції процесу «Формування та завантаження анкети» (Додаток А Рис.3.), яка складається з:

— П'яти блоків процесу – «Визначення типу респондентів», «Визначення мети опитування», «Формування переліку питань», «Формування анкети», «Затвердження анкети»;

— Стрілки контролю, механізмів, входів та виходів аналогічні, проте додалися проміжні стрілки «Визначений тип респондентів», «Мета опитування», «Перелік питань», «Проект анкети», «Незатверджена анкета», які перетворюються у стрілки виходу, що були по замовчуванню.

Далі йде діаграма декомпозиції процесу «Обробка результатів опитування та формування звіту» (Додаток А Рис.4.), яка складається з:

— Чотирьох блоків процесу – «Формування звітів по кожній дисципліні», «Вивантаження звіту по факультетах», «Аналіз результатів по кожній обраній ОПП», «Формування звіту»;

— Стрілки контролю, механізмів, входів та виходів аналогічні, проте додалися проміжні стрілки «Звіти по кожній дисципліні», «Сумарна інформація по кожному факультету», «Результат аналізу», які перетворюються у стрілки виходу, що були по замовчуванню.

1.4.2. Виявлені проблеми

Під час створення функціональної моделі було виявлено наступні проблеми та недоліки:

- анкетування проводиться тільки за однією анкетною;
- немає можливості формування та редагування анкет;
- кожен раз необхідно створювати опитування для кожної групи, при цьому йде навантаження на викладача;
- обробка результатів направлена тільки на одну анкету;
- результати представляються одним масивом даних, що вивантажується у excel;
- звіт формується тільки по кожній окремій групі;
- призначення анкети для груп прописується вручну;

— Анкети існують не довгий період часу, кожен рік потрібно створювати наново.

1.4.3. Задачі автоматизації

Зробивши функціональну схему та провівши її аналіз було виявлено що для усунення наявних проблем потрібно створити нову систему що буде реалізована за допомогою PHP та MYSQL. Необхідно, щоб у системі можливо було: завантажувати інформацію про викладачів, групи, дисципліни; можливість створювати та постійно редагувати різні анкети кожен рік; можливість формування звітності. Необхідно завантажувати інформацію у систему про викладацький склад, та поєднувати функції попередньої системи, але повинна бути нова система, яку можна буде розташувати на сервері НУХТ.

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

1.5.1. Google Forms

Google Форми (англ. Google Forms) — програмне забезпечення для адміністрування опитування. Google Forms має свої особливості:

- Не потрібно тиражувати матеріал опитування на всіх респондентів;
- Немає потреби особисто спілкуватися з респондентами, збирати їх всіх в одному місці в один час;
- Результати опитування зберігаються і постійно доступні в інтернеті;
- Результати опитування автоматично опрацьовуються і представляються і вигляді наочних узагальнених діаграм;
- Безкоштовно;
- Існує анонімне та не анонімне анкетування.

1.5.2. Survio

Survio — це онлайн-система опитувань для підготовки анкет, збору та аналізу даних та обміну результатами. Також має свої особливості:

- Є безкоштовний тариф і платний;
- Має ряд готових шаблонів опитування;
- Підтримує 19 видів питань;

— Широка підтримка звітності(PDF, DOCx, PPTx, PNG, XLS, CSV, XML);

— Анонімне анкетування.

1.5.3. Simpoll

Simpoll — це онлайн-система опитувань для підготовки анкет, збору та аналізу даних та обміну результатами. Також має свої особливості:

— Є безкоштовний тариф і платний;

— Існує можливість додати співавторів для опитування(створювати опитування разом);

— Немає анонімності;

— Підтримує багато видів питань та має шаблони анкет.

1.5.4. Порівняння систем – аналогів

Зрівняння систем – аналогів виконано за критеріями:

— Ціна;

— Анонімність:

— Шаблони анкет;

— Зручні звіти по результатах опитування;

— Можливість додати співавторів.

Порівняння систем – аналогів наведено у таблиці 1.2.

Таблиця 1.2. Порівняння систем – аналогів

Критерій	Google Forms	Survio	Simpoll
Ціна	Безкоштовно	Безкоштовно-900 грн\місяць	Безкоштовно-300 грн\місяць
Анонімність	Є можливість вибрати	Лише анонімно	Лише не анонімно
Шаблони анкет	-	+	+

Зручні звіти по результатах опитування	+	+	+
Можливість додати співавторів	-	-	+

Проаналізувавши таблицю і порівнявши системи за різними критеріями, можна сказати, що більшість систем аналогів справляються з більшістю виявлених проблем, але при цьому також мають ряд недоліків. Основними недоліками Survio та Simpoll є те, що для наших потреб вони доволі дорогавартісні. І всі вони не справляються з нашою проблемою, що потрібно формувати не одну а декілька анкет одразу. Виходячи з усіх вищезгаданих особливостей систем, їх впровадження у відділі є недоречним.

1.6. Обґрунтування доцільності проектування й розроблення системи анкетування НУХТ

При проведенні аналізу нинішньої системи анкетування та системного аналізу відділу в цілому було виявлено ряд недоліків які не задовольняють всіх потреб відділу.

Під час створення функціональної моделі було виявлено наступні проблеми та недоліки:

- анкетування проводиться тільки за однією анкетною;
- немає можливості формування та редагування анкет;
- кожен раз необхідно створювати опитування для кожної групи, при цьому йде навантаження на викладача;
- обробка результатів направлена тільки на одну анкету;
- результати представляються одним масивом даних, що вивантажується у excel;

- звіт формується тільки по кожній окремій групі;
- призначення анкети для груп прописується вручну;
- анкети існують не довгий період часу, кожен рік потрібно створювати наново.

Було виявлено, що існуючі системи аналоги не задовольняють всіх вимог, які ставляться до системи. Існуюча система анкетування застаріла і не здатна виконати функції які від неї вимагаються.

Враховуючи всі дані та висновки було вирішено спроектувати та розробити нову систему проведення анкетування Національного університету харчових технологій, що буде реалізована за допомогою PHP та MYSQL.

Необхідно, щоб у системі можливо було: завантажувати інформацію про викладачів, групи, дисципліни; можливість створювати та постійно редагувати різні анкети кожен рік; можливість формування звітності. Необхідно завантажувати інформацію у систему про викладацький склад, та поєднувати функції попередньої системи, але повинна бути нова система, яку можна буде розташувати на сервері НУХТ.

1.7. Концептуальна модель системи

Проектування, створення та впровадження нової системи анкетування повинно призвести до спрощення або змін у виконанні деяких задач та процесів. Доцільно буде створити концептуальну модель (TO-BE), яка схожа на функціональну модель, проте буде відображати зміни та скоротить час впровадження системи, оскільки наочно покаже в яких процесах буде застосовуватись нова розробка. Її метою є чітке визначення розподілу ресурсів між наявним процесами, що дає можливість оцінити ефективність використання нової системи.

Концептуальна модель що розміщена в додатку Б на противагу функціональній моделі містить в собі нові елементи. Стрілка виходу – «Ефективно сформовані анкети», Замість блоку процесу «Формування анкети» тепер блок «Формування анкет».

Розглянувши концептуальну модель, бачимо, що створення нової системи анкетування дасть можливість значно ефективніше, швидше та простіше створювати опитування, цим самим усунувши недоліки попередньої системи. Оскільки система вирішує проблеми які були визначені, то вона є актуальною для розробки.

РОЗДІЛ 2. ОПИС КОМПЛЕКСУ ЗАДАЧ АВТОМАТИЗАЦІЇ

2.1. Інформаційне забезпечення системи

Створення системи розпочинається з проектування та створення логічної та фізичної схеми бази даних. Як результат проектування ми отримаємо готову структуру бази даних, тобто склад таблиць та стовпців в ній, їх типи даних та розміри, зв'язки між таблицями та інше. На логічному рівні проектування відбувається відображення даних концептуальної моделі, яка була створена раніше в логічну модель, підтримувану СУБД. Тому виходячи з висновків та сформованих задач у першому розділі можна створити список сутностей, які будуть фундаментом нашої бази даних:

- Дисципліна – містить загальну інформацію дисципліни в поточному семестрі;
- Факультет – містить інформацію про факультети університету;
- Науковий ступінь – містить інформацію про науковий ступінь викладачів;
- Викладач – містить інформацію про викладачів;
- Дисципліна – містить інформацію про дисципліни в поточному семестрі;
- Анкетування – містить інформацію проведення анкетування;
- Відповідь на анкету – зберігає відповіді на анкети;
- Анкета – містить інформацію про анкету;
- Питання – містить питання на варіанти відповідей (якщо це потрібно);
- Склад анкети – містить набір питань на анкету;
- Освітній ступінь – містить інформацію про освітній ступінь викладачів;
- Навчальна група – містить інформацію про навчальні групи студентів;
- Освітньо-професійна програма – містить інформацію про програму навчання для студентів;
- Галузь знань – містить інформацію про галузь знань до якої належить та чи інша навчальна програма;

- Код спеціальності – містить назву спеціальності до якої належить навчальна програма;
- Наукове звання – містить інформацію про наукове звання яке може мати викладач;
- Посада – містить інформацію про посаду, яку займає викладач;
- Вибіркова дисципліна – вказує чи можна вибирати дану дисципліну;
- Обов’язкова дисципліна – вказує чи є дисципліна обов’язковою до вивчення.

Відповідно до даного списку можна будувати логічну та фізичну моделі даних (Додаток В) за допомогою CASE – засобу AllFusion ERWin Data Modeler. Після визначення та створення основних сутностей в ERWin Data Modeler, потрібно задати їм атрибути(поля) та визначити домени(властивості). Згодом ми можемо спроектувати логічну модель бази даних, визначивши первинні ключі, та зв’язки між сутностями. Варто зазначити, що при створенні фізичної моделі, сутності іменуються таблицями, атрибути - стовпчиками, а домени - типами даних. Перелік таблиць їх стовпчиків та типів даних наведені у Таблиці 3.1.

Таблиця. 2.1. Перелік таблиць, стовпчиків та типів даних

Таблиця	Перелік стовпчиків та типів даних
Факультет	Код факультету: integer Назва факультету: varchar(100)
Кафедра	Код кафедри: integer Назва кафедри: varchar(100) Код факультету: integer
Науковий ступінь	Код наукового ступеня: integer Назва наукового ступеня: varchar(50)
Наукове звання	Код звання: integer Назва звання: varchar(50)
Посада	Код посади: integer

Таблиця. 2.1. Перелік таблиць, стовпчиків та типів даних (продовження)

Таблиця	Перелік стовпчиків та типів даних
	Назва посади: varchar(100)
Викладач	Код викладача: integer ПІБ викладача: varchar(100) Дата народження: datetime Дата заключення договору: datetime Дата закінчення договору: datetime Код кафедри: integer Код наукового ступеня: integer Код звання: integer Код посади: integer Пароль: varchar(100) email: varchar(100)
Освітній ступінь	Код освітнього ступеня: integer Назва освітнього ступеня: varchar(50)
Галузь знань	Код галузі знань: integer Назва галузі знань: varchar(200)
Спеціальність	Код спеціальності: integer Назва спеціальності: varchar(100) Код галузі знань: integer
Освітньо-професійна програма	Код ОПП: integer Назва ОПП: varchar(100) Рік затвердження: integer Код освітнього ступеня: integer Код спеціальності: integer Код викладача: integer

Таблиця. 2.1. Перелік таблиць, стовпчиків та типів даних (продовження)

Таблиця	Перелік стовпчиків та типів даних
Дисципліна	Код дисципліни: integer Назва дисципліни: varchar(100) Кількість лекційних годин: integer Кількість лабораторних годин: integer Кількість практичних годин: integer Код ОПП: integer Тип дисципліни: varchar(50)
Навчальна група	Код групи: integer Назва групи: varchar(10) Курс: integer Форма навчання: varchar(10) Код ОПП: integer Кількість здобувачів: integer
Дисципліна в семестрі	Код дисципліни в семестрі: integer Код групи: integer Код дисципліни: integer Код викладача: integer Семестр: integer Рік навчання: integer Викладає лекції: integer Викладає лабораторні: integer Викладає практичні: integer
Анкета	Код анкети: integer Назва анкети: varchar(100)

Таблиця. 2.1. Перелік таблиць, стовпчиків та типів даних (продовження)

Таблиця	Перелік стовпчиків та типів даних
	Опис анкети: varchar(2000) Статус: varchar(20) Призначення: varchar(200) Дата створення: datetime Дата затвердження: datetime
Питання	Код питання: integer Текст питання: varchar(200) Тип питання: varchar(20) Варіант відповіді 1: varchar(100) Варіант відповіді 2: varchar(100) Варіант відповіді 3: varchar(100) Варіант відповіді 4: varchar(100) Варіант відповіді 5: varchar(100) Варіант відповіді 6: varchar(100) Варіант відповіді 7: varchar(100) Варіант відповіді 8: varchar(100) Варіант відповіді 9: varchar(100) Варіант відповіді 10: varchar(100)
Склад анкети	Код складу анкети: integer Код анкети: integer Код питання: integer
Обов'язкова дисципліна	Код дисципліни: integer Наявність курсової роботи: integer
Вибіркова дисципліна	Код дисципліни: integer Тип переліку: integer

Таблиця. 2.1. Перелік таблиць, стовпчиків та типів даних (продовження)

Таблиця	Перелік стовпчиків та типів даних
Анкетування	Код анкетування: integer Дата початку анкетування: datetime Дата закінчення анкетування: datetime Код дисципліни семестру: integer Код анкети: integer Посилання на анкету: varchar(100)
Відповідь на анкету	Код відповіді: integer Дата відповіді: datetime Код анкетування: integer Код питання: integer Відповідь: varchar(1000)

Варто також зазначити, що при створенні фізичної моделі бази даних важливим етапом є створення правил валідації:

- Всі дати ставляться сьогоднішнім числом по – замовчуванню, але також користувач зможе сам ставити дату на свій розсуд;
- В тип даних який integer, значення поля не повинно бути менше 1;

Також важливо встановити для всіх первинних ключів властивість IDENTITY, задля автоматичного інкрименту ключів при створенні нових записів.

Після створення логічної та фізичної моделі бази даних та її перевірки на неточності чи певні помилки можна генерувати базу даних. Генерація бази даних відбувається за допомогою спеціального інструменту середовища AllFusion ERWin Data Modeler – «Schema Generation», який дає можливість

генерації бази даних на основі створених моделей в передбачену середовищем СКБД. Для генерації нашої бази даних було обрано СКБД – MySQL. Перед генерацією потрібно виконати певні дії: створити порожню базу даних в СКБД та під'єднати AllFusion ERWin Data Modeler до неї. Після генерації, відкривши базу даних в My SQL можна побачити створені таблиці, які проектувались у AllFusion ERWin Data Modeler, побудувати схему бази даних (Додаток Г) – вона також буде відповідати фізичній моделі бази даних. Коли база даних готова можна переходити до створення її інтерфейсу.

2.2. Алгоритмізація та реалізація комплексу задач автоматизації

При створенні інтерфейсу користувача та реалізації задач автоматизації було використано фреймворк Symfony. Перш за все потрібно під'єднати базу даних до проекту. Для цього необхідно налаштувати рядок DATABASE_URL, який в нашому випадку буде мати вигляд: DATABASE_URL="mysql://db_user:db_password@127.0.0.1:3306/db_name?serverVersion=5.7".

Спочатку потрібно розробити форму входу в модуль, якій буде відбуватись авторизація викладача. Для цього на форму на Рис. 2.1. додали два поля введення, два елементи типу Label, для розуміння, що куди вводити, і кнопку для авторизації. Передбачено, що дані для входу користувачі отримають заздалегідь.

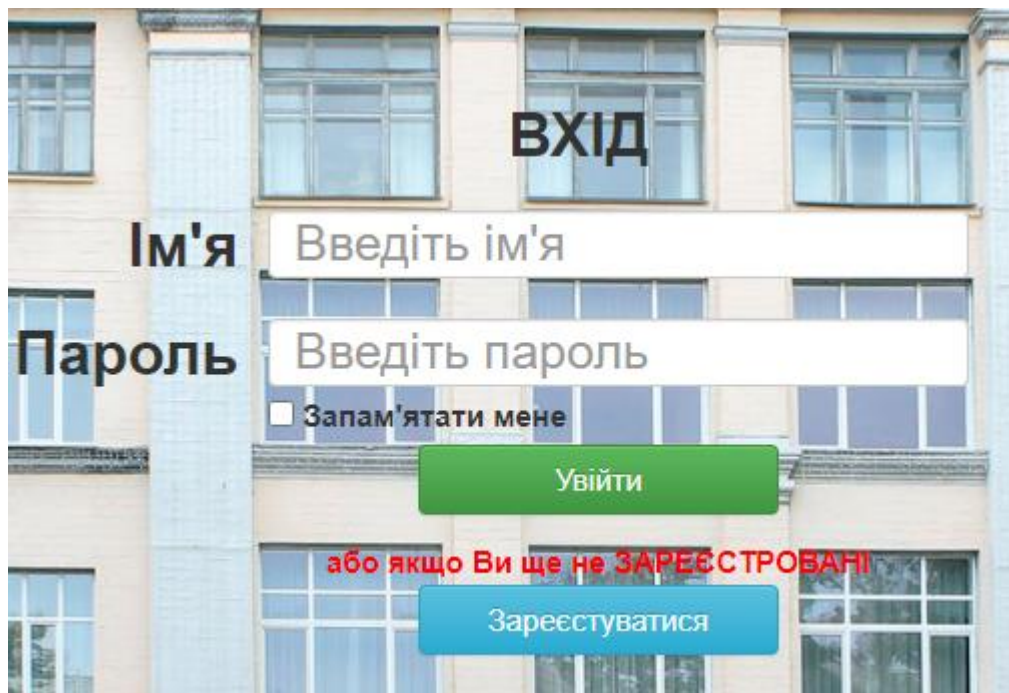


Рис. 2.1. Форма авторизації

При спробі авторизуватися не вводячи даних в поля отримуємо помилку:

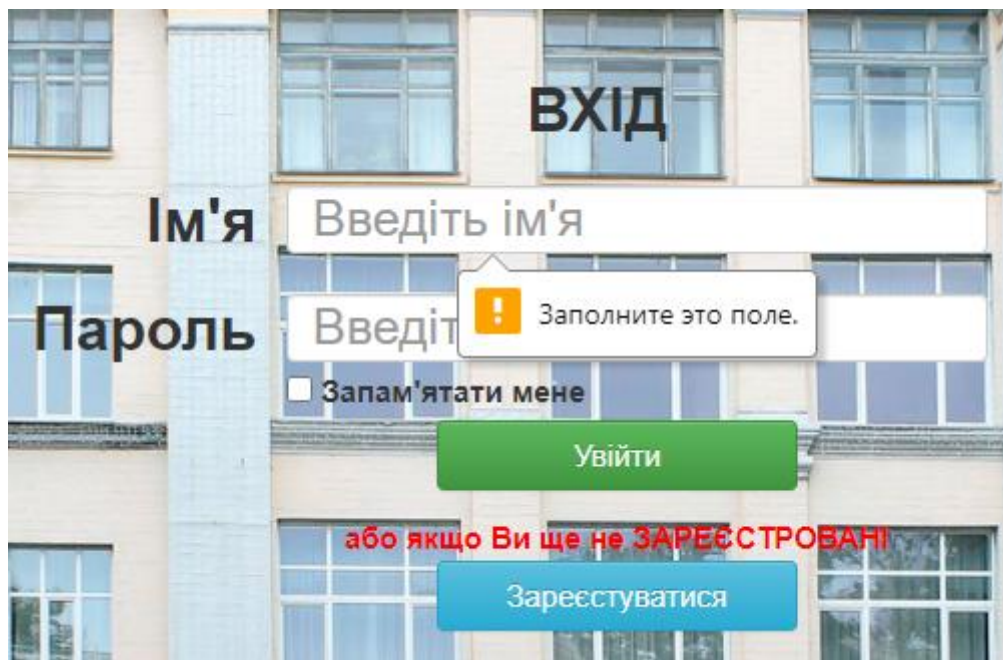


Рис. 2.2. Помилка при авторизації

Якщо при авторизації вказати дані користувача якого не існує, то отримаємо попередження:

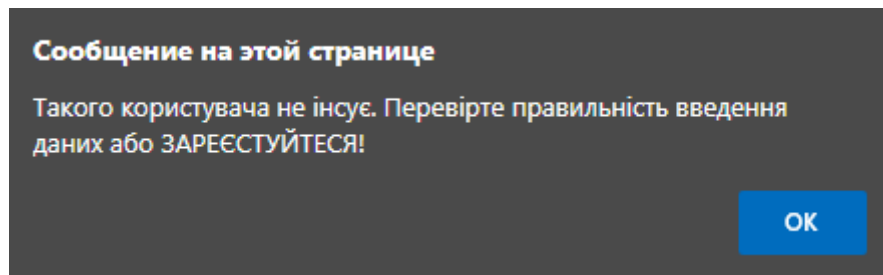


Рис. 2.3. Повідомлення при авторизації

При авторизації поле пароля приховує введені дані, щоб сторонні їх не бачили, але є можливість їх переглянути:



Рис. 2.4. Поле «Пароль» приховане



Рис. 2.5. Поле «Пароль» відкрите

За авторизацію користувача відповідає наступний блок коду:

```
<?php
namespace Phppot;
use \Phppot\DataSource;

$message = "";
if (count($_POST) > 0) {
    $isSuccess = 0;
    require_once __DIR__ . '/DataSource.php';
    $conn = new DataSource();
    $query = 'SELECT * FROM Викладач WHERE ПІБ= ?';
    $paramType = 's';
    $paramValue = array(
        $_POST["ПІБ"]
    );
    $result = $conn->select($query, $paramType, $paramValue);

    if (! empty($result)) {

        $hashedPassword = $result[0]["Пароль"];
        if (Password_verify($_POST["Пароль"], $hashedPassword)) {
            $isSuccess = 1;
        }
    }
}
```

```

    }
  }
  if ($isSuccess == 0) {
    $message = "Неправильне ім'я чи пароль!";
  } else {
    header("Location: ./success-message.php");
  }
}
?>

```

Після отримання інформації про автентифікацію користувача в PHP запит порівнює дані форми з базою даних користувача, виконуючи запит за допомогою об'єкта з'єднання.

Запит пов'язує ім'я користувача, введене користувачем через форму HTML. Потім він перевіряє хеш пароля з введеним паролем, щоб повернути результати автентифікації.

Якщо збіг знайдено, це означає, що користувач справжній, який уже зареєструвався в системі. Отже, код автентифікації дозволить користувачеві продовжити далі.

Якщо користувач ще не зареєстрований в системі, йому потрібно пройти реєстрацію, натиснувши на відповідну кнопку. Вікно реєстрації має наступний вигляд:

РЕЄСТРАЦІЯ

Ім'я та прізвище

Електронна адреса

Пароль

Повторіть пароль

або якщо Ви вже **ЗАРЕЄСТРОВАНІ**

Рис. 2.6. Форма реєстрації нового користувача

При спробі ввести данні(ім'я та прізвище чи email), які вже існують в системі, користувач побачить повідомлення:

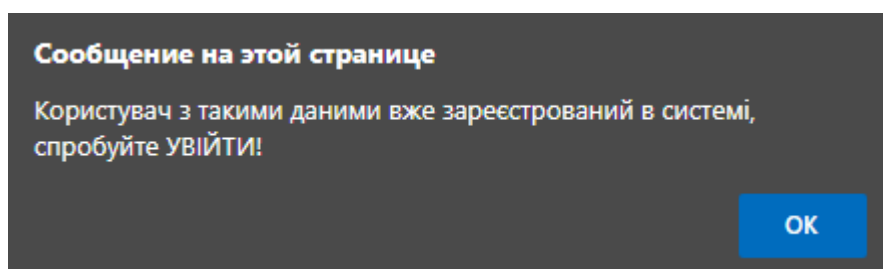


Рис. 2.7. Спроба зареєструватися з даними які вже є в базі даних

Аналогічно до форми авторизації, на формі реєстрації передбачено обов'язкове заповнення всіх полів, та приховування паролів. Окрім того якщо при натисканні на кнопку реєстрації введені дані про паролі не будуть збігатися, то користувач отримає наступне повідомлення:

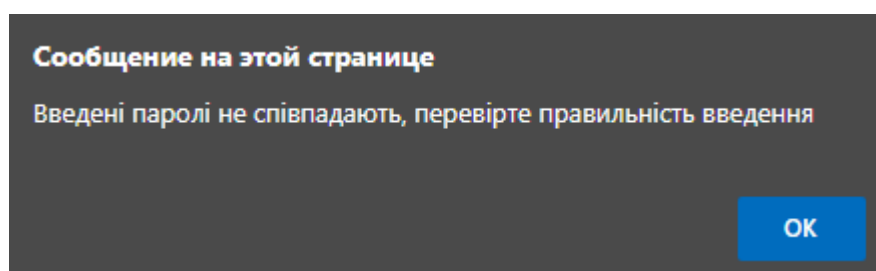


Рис. 2.8. Повідомлення при пробі авторизуватися з різними паролями

За реєстрацію нового користувача відповідає запит на додавання, який реалізується таким PHP скриптом:

```
<?php
include_once 'db.php';
if(isset($_POST['submit']))
{
    $name = $_POST['ПІБ'];
    $email = $_POST['email'];
    $password = $_POST['Пароль'];
    $sql = "INSERT INTO Викладач (ПІБ,email, пароль)
    VALUES ('$name','$email', '$password)";
    if (mysqli_query($conn, $sql)) {
        echo "Успіх";
    } else {
        echo "Помилка: " . $sql . ":-" . mysqli_error($conn);
    }
}
mysqli_close($conn);
?>
```

Якщо всі дані введені коректно тоді користувачу відкривається головне меню:



Рис. 2.9. Головне меню

В головному меню присутня можливість перейти до однієї з 6 наступних функцій:

- Адміністрування – дозволяє проводити налаштування системи;
- Анкети – дозволяє переглянути всі анкети були створені раніше, створити нову анкету(нову чи на основі іншої);
- Опитування – дозволяє налаштувати анкету для певної аудиторії, вибрати дати проведення опитувань, та запустити їх;
- Налаштування – дає можливість налаштування профіля;
- Звіт за період – дає можливість переглянути звітність за певний період часу за певною освітньою програмою;
- Вихід – реалізує вихід з модуля.

Також в основній частині екрану головного меню міститься інформація про авторизованого користувача: ПІБ, телефон, email.

При виборі «Анкети», відкривається форма (Рис. 2.10.), яка дає змогу вибрати одне з наступного:

- Переглянути всі анкети;

- Створити нову анкету;
- Редагувати анкету.



Рис. 2.10. Меню «Анкети»

Для того, щоб перейти до одного з вищенаведених пунктів, потрібно натиснути відповідну кнопку. При натисканні на «Перегляд анкет» маємо наступне вікно:

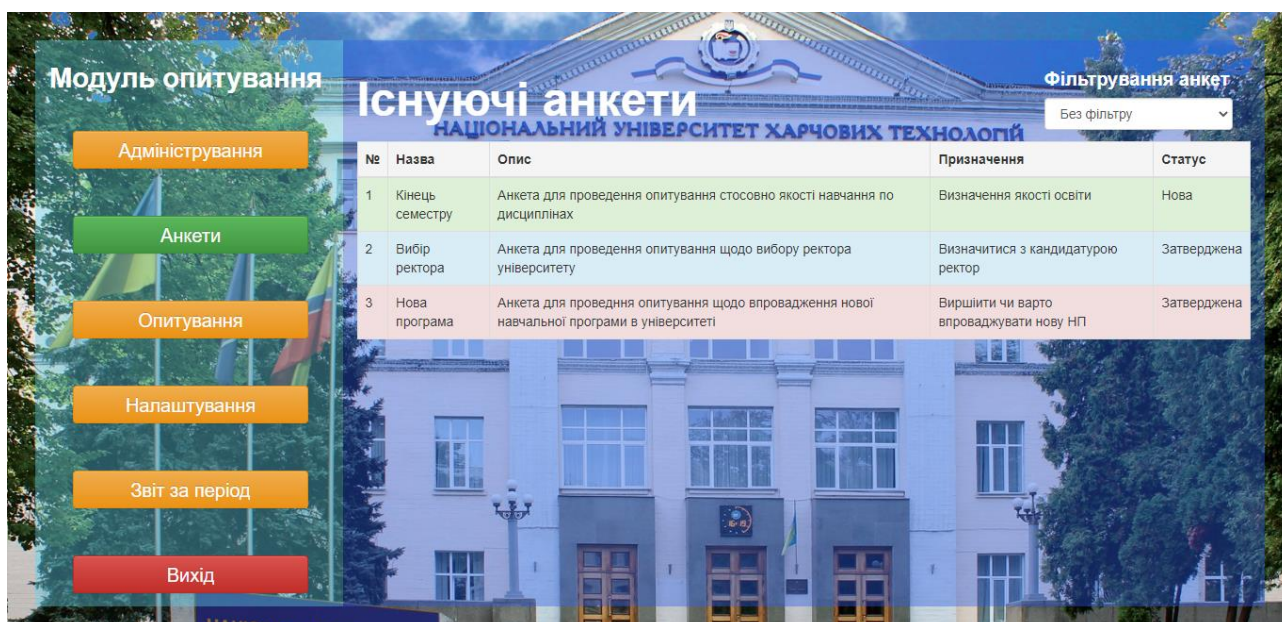


Рис. 2.11. Перегляд анкет

Форма дозволяє переглянути інформацію про всі створені анкети, та за необхідності відфільтрувати їх за одним з наступних параметрів:

- Без фільтру(за змовчуванням);

- За назвою;
- За назвою(реверс);
- Нові;
- Затверджені.

Вікно фільтрування виглядає так:

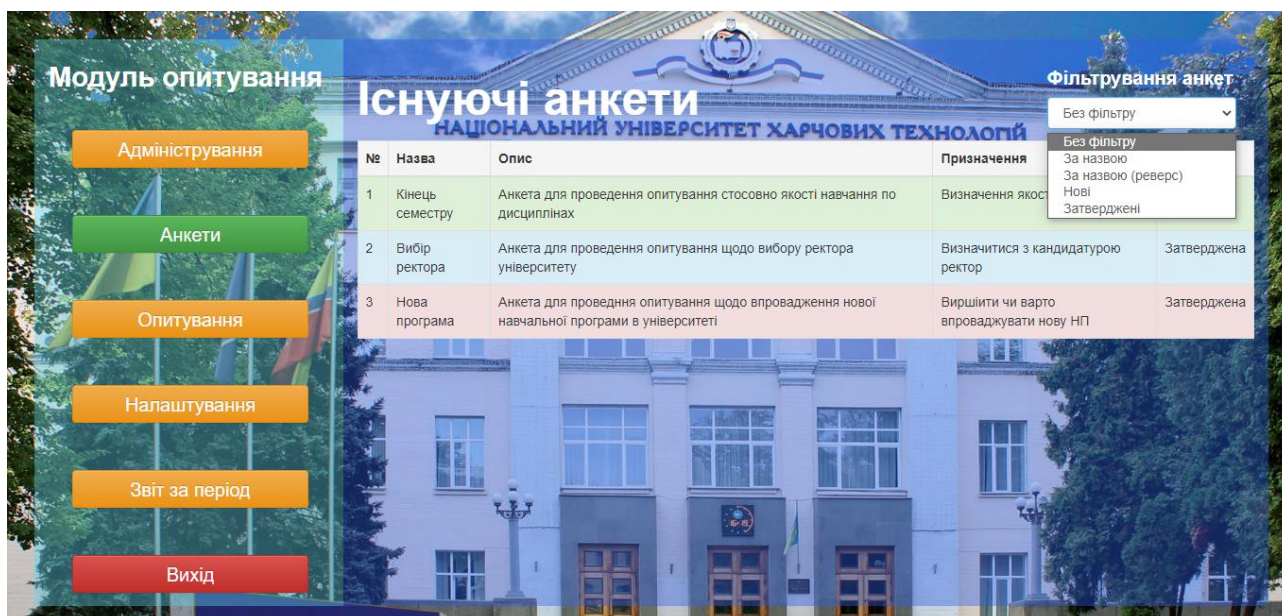


Рис. 2.12. Вікно фільтрування

При виборі параметра фільтрування виконується запит на вибірку даних, який реалізує такий PHP скрипт(приклад для параметру фільтрування «Затверджена»):

```
<?php
$query = sprintf("SELECT Анкета.Назва, Анкета.Опис, Анкета.Призначення, Анкета.Статус
FROM Анкета
WHERE Анкета.Статус='Затверджена'",
mysql_real_escape_string($Анкета.Статус);
$result = mysql_query($query);
// Тут перебираємо всі значення які відповідають запиту і записуємо в таблицю
while ($row = mysql_fetch_assoc($result)) {
    echo $row['Анкета.Назва'];
    echo $row['Анкета.Опис'];
    echo $row['Анкета.Призначення'];
    echo $row['Анкета.Статус'];
}
```

```

}
mysql_free_result($result);
?>

```

Результатом фільтрування буде:



Рис. 2.13. Результат фільтрування анкет

Щоб повернутися знову до всіх анкет обираємо пункт «Без фільтру». Повернення реалізує такий скрипт:

```

<?php
$query = sprintf("SELECT Анкета.Назва, Анкета.Опис, Анкета.Призначення, Анкета.Статус
FROM Анкета;
$result = mysql_query($query);
// Тут перебираємо всі значення які відповідають запиту і записуємо в таблицю
while ($result) {
    echo $row['Анкета.Назва'];
    echo $row['Анкета.Опис'];
    echo $row['Анкета.Призначення'];
    echo $row['Анкета.Статус'];
}
mysql_free_result($result);
?>

```

Для додавання нової анкети повертаємося в попереднє меню, та обираємо «Нова анкета». Нас зустрічає вікно:

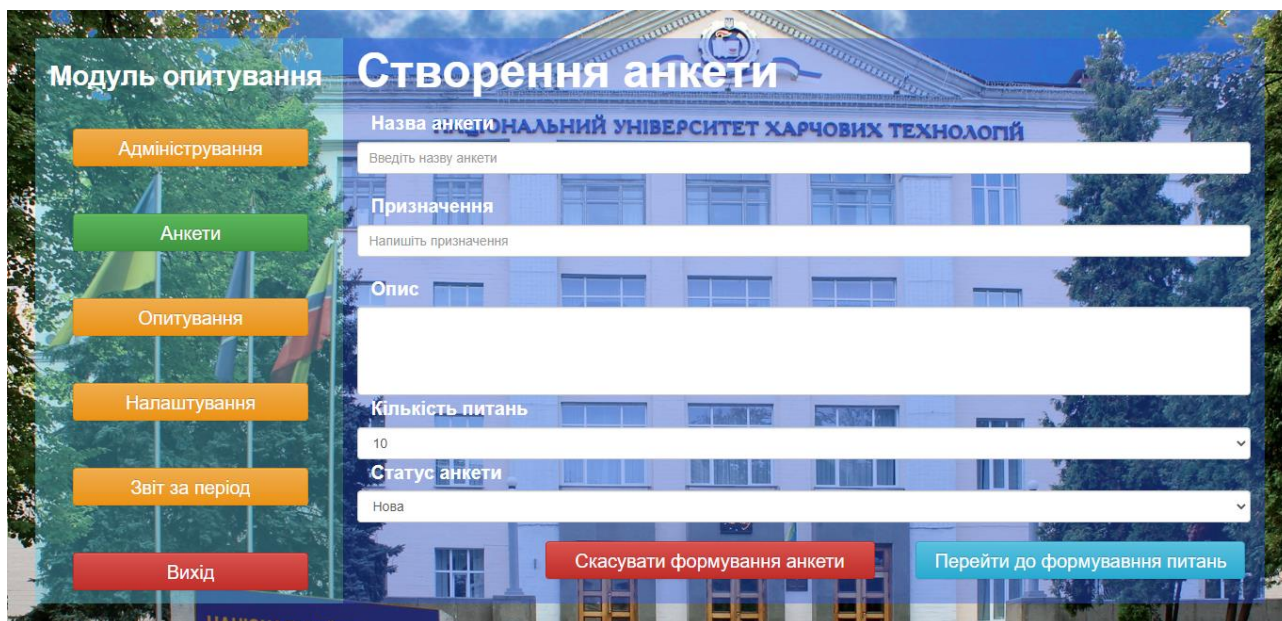


Рис. 2.14. Вікно створення нової анкети

При створенні анкети обов'язково потрібно вказати всі дані. Кількість питань в анкеті за замовчуванням 10, проте є можливість встановити їх аж до 100. Для того щоб мати можливість редагувати потім анкету потрібно вибрати її статус «Нова». При «Затверджена» інформується про те, що не буде можливості редагувати анкету:



Рис. 2.15. Попередження про неможливість редагування

Якщо ви хочете відмінити формування анкети, потрібно натиснути відповідну кнопку. Після натиснення буде попередження чи дійсно ви хочете це зробити (на випадок випадкового натиснення), при натисненні «ОК» відбудеться вихід в попереднє меню, в іншому випадку ви залишитися на сторінці формування анкети, вікно має вигляд:

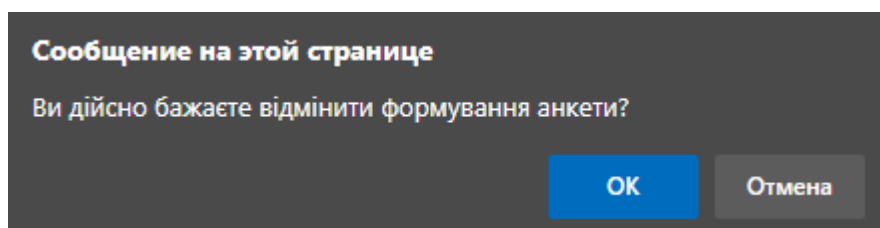


Рис. 2.16. Сповідження при спробі скасувати формування анкети

При переході до формування питань користувач побачить вікно:

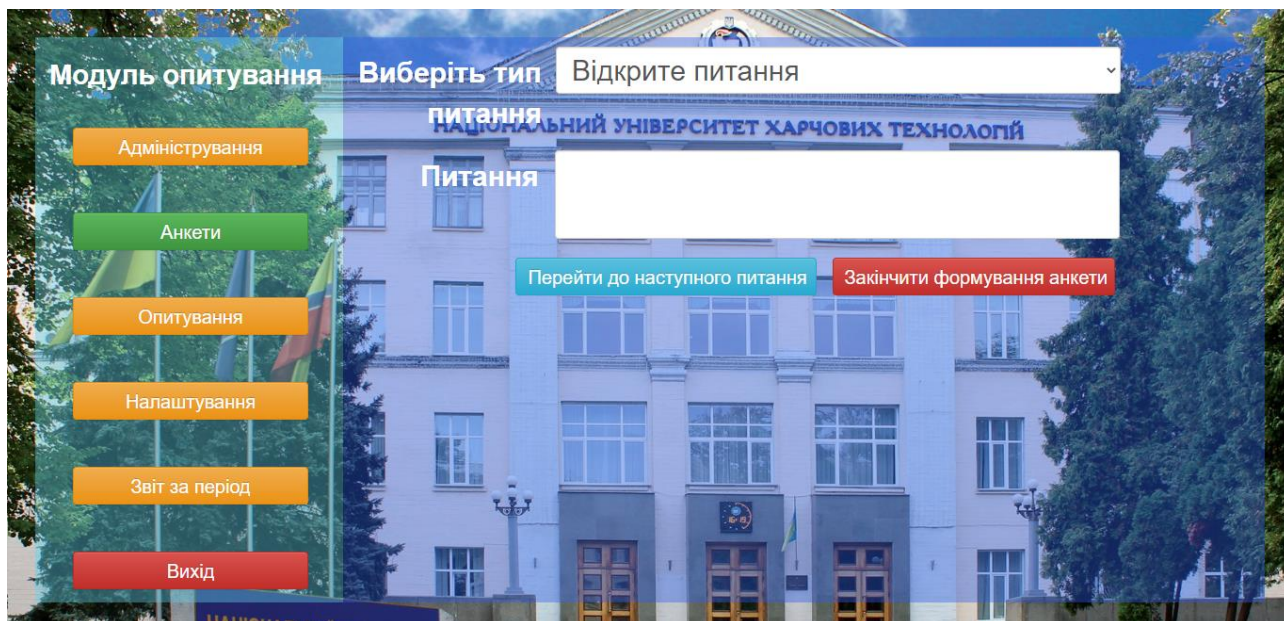


Рис. 2.17. Вікно формування питань

У цьому вікні користувач має змогу додавати питання до анкети. Питання можуть бути двох типів:

- Відкрите питання;
- Питання з варіантами відповіді.

За замовчуванням встановлено «відкрите питання», проте користувач може це змінити вибравши інший тип зі списку. Також, не дивлячись на кількість питань, яка була вибрана до того, користувач може завершити формування анкети, не заповнивши всі питання за допомогою відповідної кнопки, при цьому він отримає попередження чи дійсно він бажає це зробити.

Відкрите питання буде зберігатися в базі даних за допомогою запиту на додавання, який буде реалізувати наступний PHP скрипт:

```
<?php
include_once 'db.php';
if(isset($_POST['submit']))
{
    $text = $_POST['Текст_питання'];
    $type = $_POST['Тип_питання'];
    $sql = "INSERT INTO Питання (Текст_питання,Тип_питання)
VALUES ('$text','$type)";
    if (mysqli_query($conn, $sql)) {
        echo "Успіх";
    }
}
```

```

} else {
    echo "Помилка: " . $sql . ":-" . mysqli_error($conn);
}
mysqli_close($conn);
}
?>

```

При виборі питання з варіантами відповіді інтерфейс вікна зміниться на наступний:

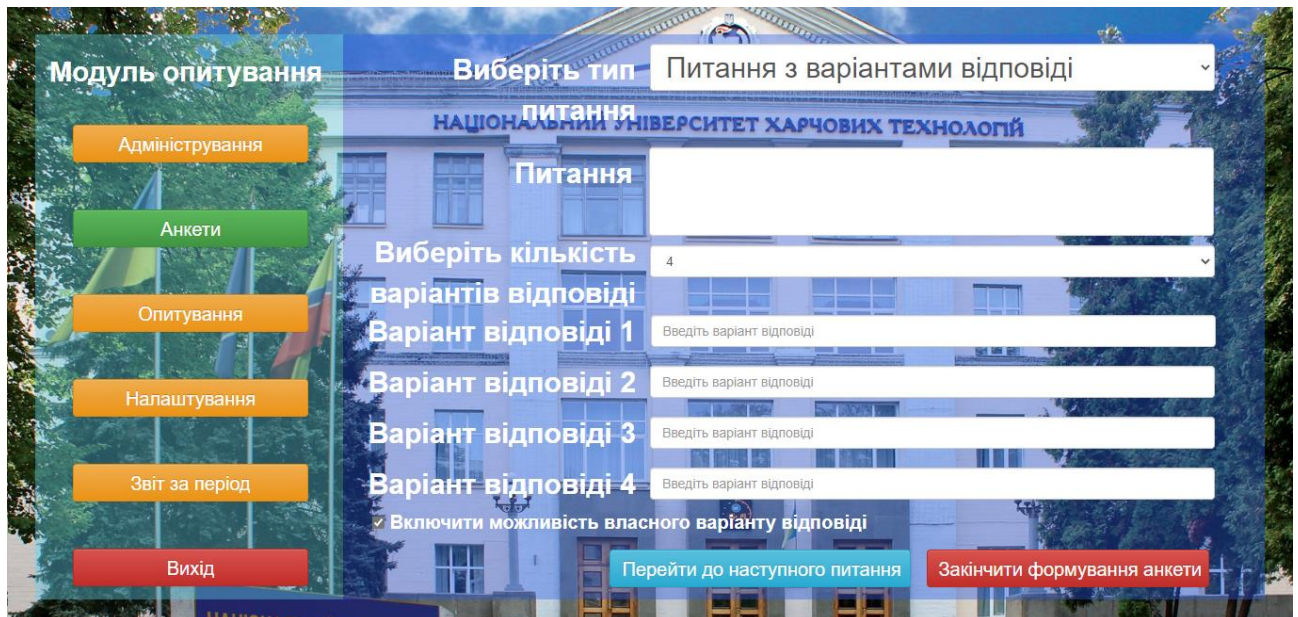


Рис. 2.18. Вигляд вікна додавання питання з варіантами відповіді

При виборі цієї опції, з'являються поля, де потрібно ввести варіанти відповіді, за замовчуванням їх 4, але є можливість вибрати від 2 до 10. Також є можливість включити можливість власного варіанту відповіді.

Запис питань та варіантів відповіді відбувається за допомогою запиту на додавання, який в PHP реалізується наступним скриптом:

```

<?php
include_once 'db.php';
if(isset($_POST['submit']))
{
    $text = $_POST['Текст_питання'];
    $type = $_POST['Тип_питання'];
    $answer_1 = $_POST['Варіант_відповіді_1'];
    $answer_2 = $_POST['Варіант_відповіді_2'];
    $answer_3 = $_POST['Варіант_відповіді_3'];
    $answer_4 = $_POST['Варіант_відповіді_4'];
    $sql = "INSERT INTO Питання (Текст_питання,Тип_питання,
Варіант_відповіді_1,Варіант_відповіді_2,Варіант_відповіді_3,Варіант_відповіді_4)

```

```

VALUES ('$text','$type', '$answer_1','$answer_2', '$answer_3', '$answer_4");
if (mysqli_query($conn, $sql)) {
    echo "Успіх";
} else {
    echo "Помилка: " . $sql . ":-" . mysqli_error($conn);
}
mysqli_close($conn);
}
?>

```

Кількість полів для варіантів відповіді змінюється в залежності від обраного значення кількості варіантів відповіді.

Також при формуванні анкети передбачено, що всі поля питань та відповідей повинні бути заповнені:

Рис. 2.19. Передбачено обов'язкове заповнення полів(Питання)

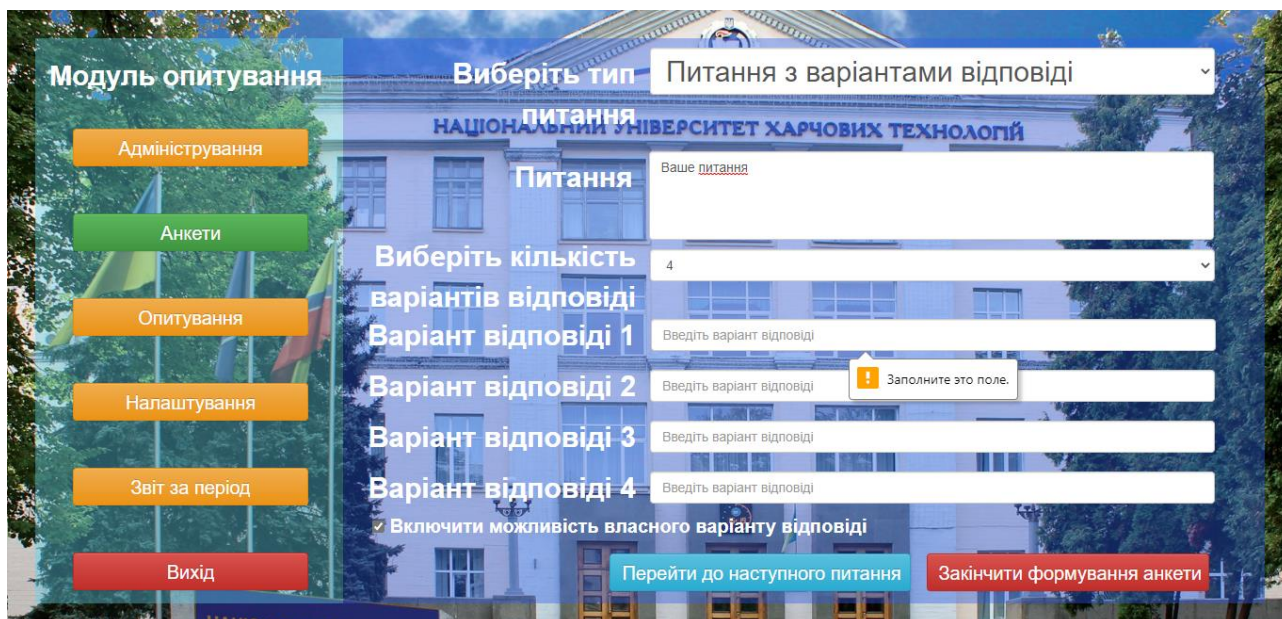


Рис. 2.20. Передбачено обов'язкове заповнення полів(Варіанти відповіді)

Після завершення формування анкети, за умови що її статус відповідає «Нова» є можливість її редагувати, для цього потрібно повернутися в попереднє меню, та обрати відповідний пункт меню, після чого нас зустрічає вікно:

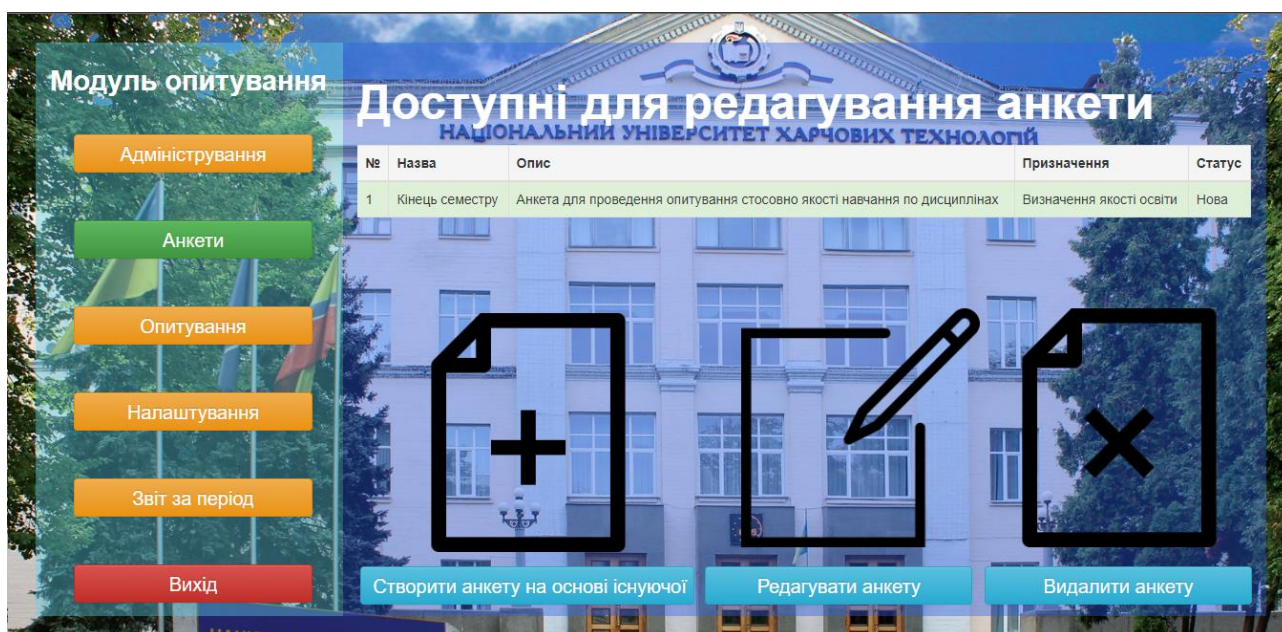


Рис. 2.21. Меню редагування анкет

Наступне меню дає можливість:

- Створити нову анкету на основі вже існуючої(редагуючи її питання, їх кількість);

- Редагувати анкету(додати нові питання чи змінити існуючі);
- Видалити анкету;

При виборі видалення анкети, аналогічно до попередніх повідомлень користувачу, модуль анкетування запитає його, чи дійсно він бажає видалити анкету. Два інші пункти меню мало чим відрізняються від звичайного додавання питань до анкети, єдина різниця що поля будуть вже заповнені і їх потрібно за необхідності редагувати, та є можливість додати нове питання. При редагування анкети буде спрацьовувати запит на оновлення який реалізується наступним чином для питання з відкритою відповіддю:

```
<?php
include_once 'db.php';
if(isset($_POST['submit']))
{
    $text = $_POST['Текст_питання'];
    $type = $_POST['Тип_питання'];
    $sql = "UPDATE Питання (Текст_питання,Тип_питання)
    SET (Текст_питання = '$text', Тип_питання = '$type')
    WHERE Текст_питання = ?";
    if (mysqli_query($conn, $sql)) {
        echo "Успіх";
    } else {
        echo "Помилка: " . $sql . ":-" . mysqli_error($conn);
    }
    mysqli_close($conn);
}
?>
```

Запит працює наступним чином: значення питання береться з таблиці і знаходиться питання, яке йому відповідає в базі даних, потім, після того як користувач ввів нові дані, значення полів оновлюються.

Для питань з варіантами відповіді запит буде мати такий вигляд:

```
<?php
include_once 'db.php';
if(isset($_POST['submit']))
{
    $text = $_POST['Текст_питання'];
    $type = $_POST['Тип_питання'];
    $answer_1 = $_POST['Варіант_відповіді_1'];
    $answer_2 = $_POST['Варіант_відповіді_2'];
    $answer_3 = $_POST['Варіант_відповіді_3'];
    $answer_4 = $_POST['Варіант_відповіді_4'];
    $sql = " UPDATE Питання (Текст_питання,Тип_питання,
    Варіант_відповіді_1,Варіант_відповіді_2,Варіант_відповіді_3,Варіант_відповіді_4)
```

```

SET (Текст_питання ='$text', Тип_питання ='$type', Варіант_відповіді_1='$answer_1',
Варіант_відповіді_2='$answer_2', Варіант_відповіді_3='$answer_3',
Варіант_відповіді_4='$answer_4')
WHERE Текст_питання = ?";
if (mysqli_query($conn, $sql)) {
    echo "Успіх";
} else {
    echo "Помилка: " . $sql . ":-" . mysqli_error($conn);
}
mysqli_close($conn);
}
?>

```

Для того, щоб запустити опитування по створених анкетах, потрібно перейти в пункт меню «Опитування»:

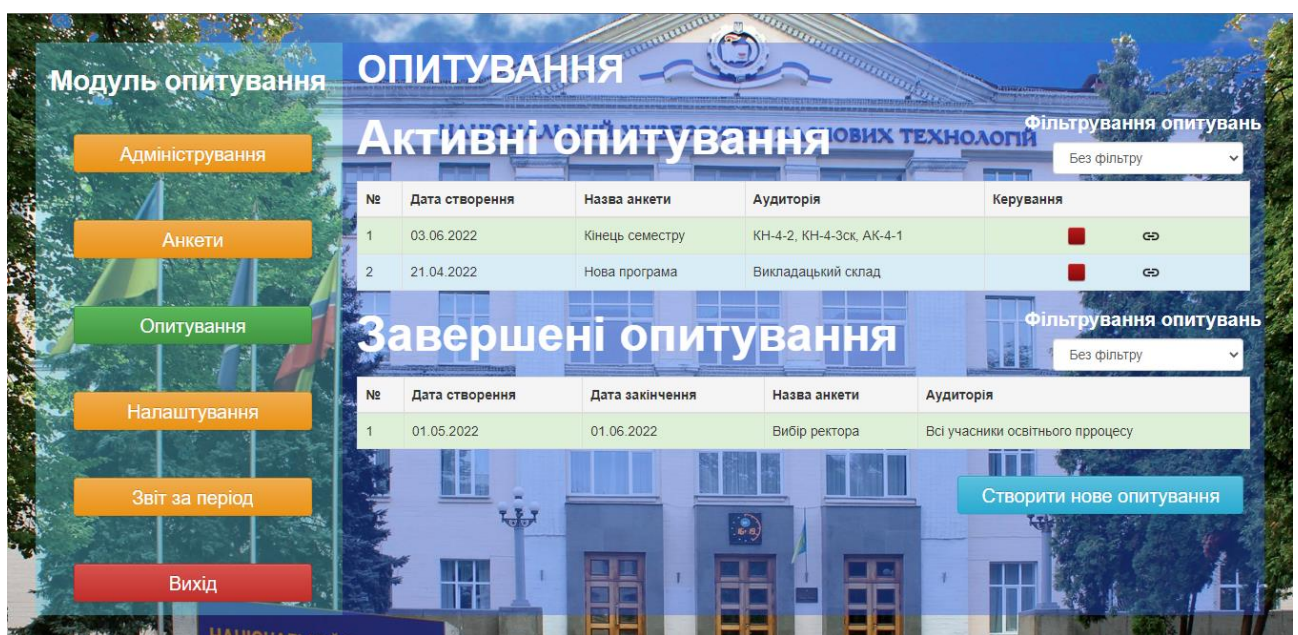


Рис. 2.22. Вигляд вікна «Опитування»

У вікні можна бачити активні та завершені опитування, для обох доступне фільтрування за такими властивостями:

- Без фільтру(за замовчуванням);
- За назвою анкети;
- За назвою анкети(реверс);
- За датою;
- За датою(реверс).

Для завершених опитувань додатково є фільтр по завершених опитуваннях «за датою завершення».

Для активних опитувань є можливість скопіювати посилання на опитування та завершити опитування натиснувши на відповідні значки в стовпці «Керування». При натисненні на завершення опитування користувач буде повідомлений чи дійсно він бажає його завершити задля запобігання випадкового натиску, це сповіщення має вигляд:

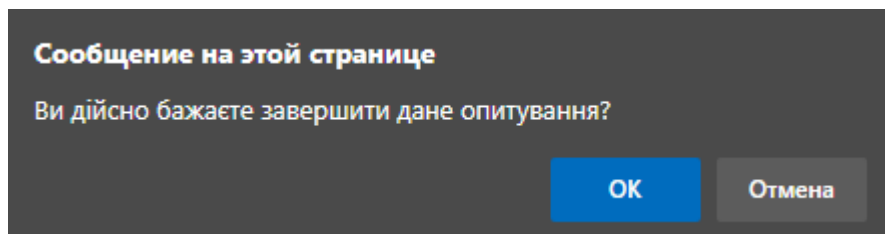


Рис. 2.23. Повідомлення при завершенні опитування

При копіюванні посилання на опитування буде наступний результат:

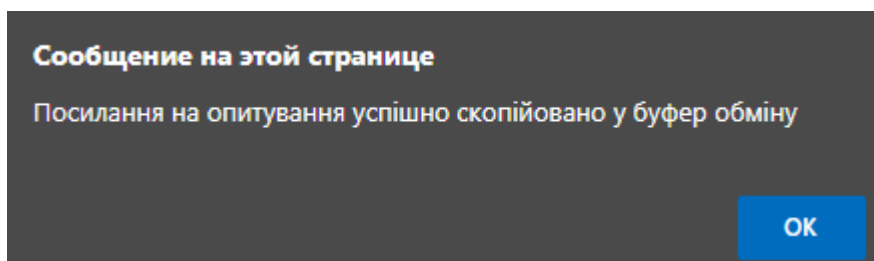


Рис. 2.24. Повідомлення при копіювання посилання на опитування

Якщо користувач бажає створити нове опитування, то він повинен натиснути на відповідну кнопку, після чого перейде на таке вікно:

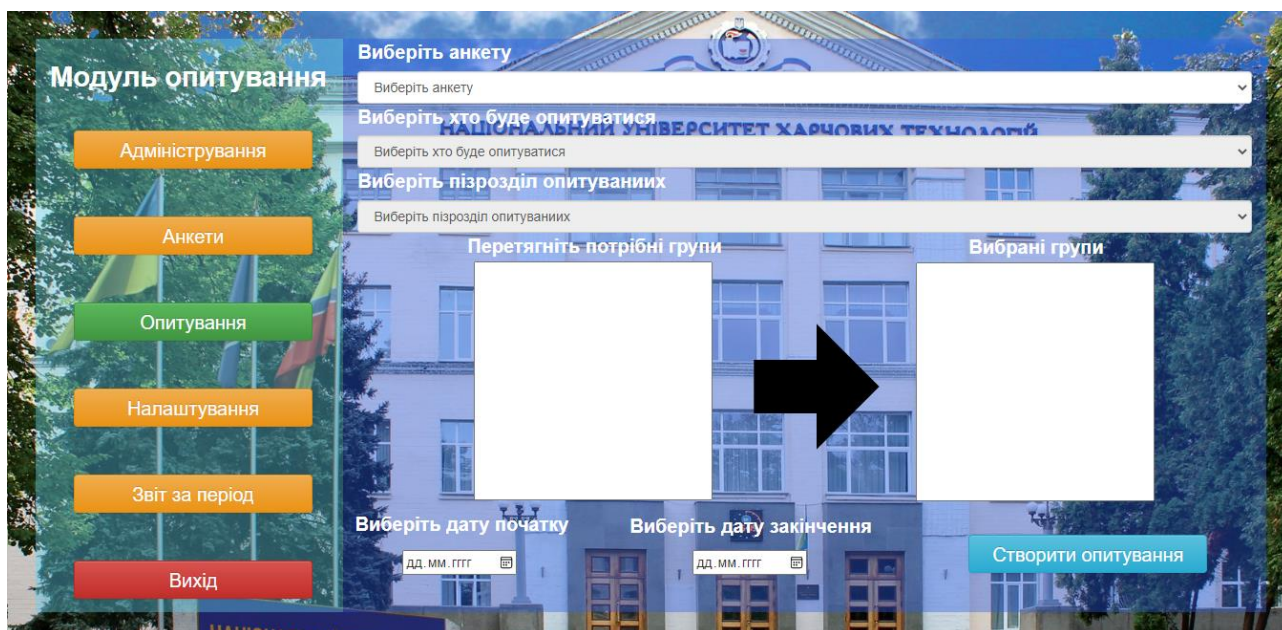


Рис. 2.25. Вигляд вікна створення нової анкети

На цій формі для того щоб створити опитування потрібно вибрати:

- Анкету, на основі якої буде створене опитування(в випадяючому списку містяться всі доступні анкети);
- Аудиторію опитування(зі списку потрібно вибрати одне з: всі учасники освітнього процесу, викладацький склад, студенти);
- Підрозділ університету, для якого буде опитування(є можливість вибрати всіх або окремих факультет).

Важливим є те що допоки невибраний попередній пункт, наступний буде недоступний для вибору.

Далі в лівому списку підтягнуться групи для яких створюється опитування, серед яких за допомогою стрілки потрібно вибрати ті які вам потрібні(обираєте групу і клікаєте на стрілку, після чого група з'являється у списку напроти).

Далі потрібно встановити дати опитування і створити його.

Приклад заповнених налаштувань опитування:

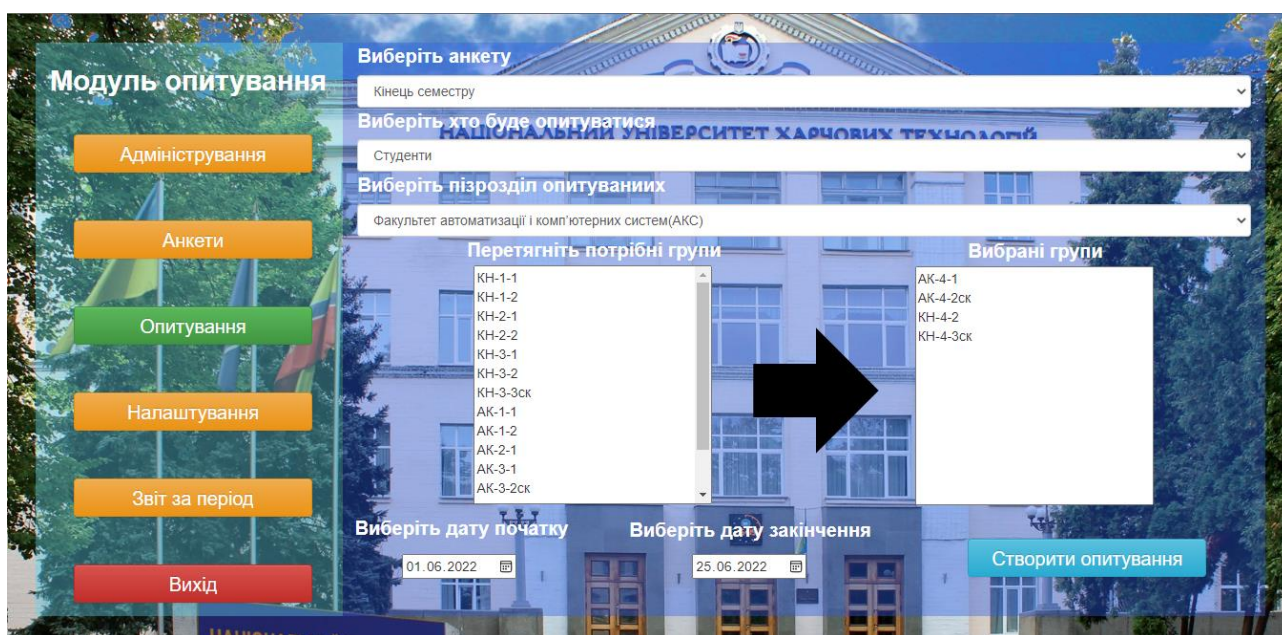


Рис. 2.26. Приклад налаштувань опитування

Після цього анкета з'явиться в попередньому меню

Також передбачена перевірка при виході з додатку, коли тиснемо на кнопку виходу отримуємо наступне повідомлення:

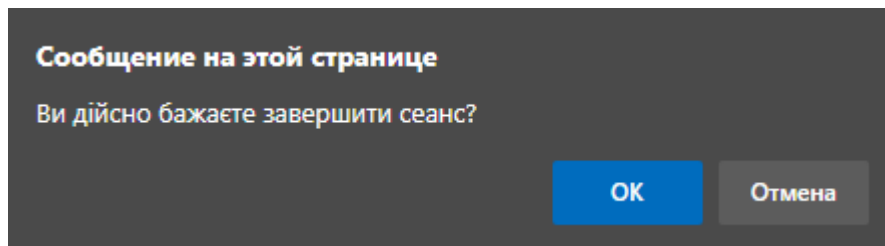


Рис. 2.27. Перевірка на вихід з додатку

Коли тиснемо «ОК» то потрапляємо на сторінку авторизації (Рис.2.1), якщо тиснемо відміну то залишаємося на сторінці.

2.3. Інструкція користувача

Дана система анкетування була розроблена для використання у відділі контролю якості освіти. Можливості даної системи передбачають створення анкети, редагування та видалення анкет, створення опитувань на основі створених анкет, перегляд результатів опитувань.

Для початку роботи з системою потрібно перейти на сайт анкетування, після чого відкриється вікно авторизації(Рис. 2.1). Для подальшої роботи потрібно ввести ім'я користувача та пароль(якщо ви вже зареєстровані) або ж зареєструватися в системі. Далі, якщо все введено коректно, буде відкрите головна сторінка, де користувач зможе обрати потрібну йому функцію, при невдалій спробі авторизації користувач побачить повідомлення що дані введені невірно.

На головному екрані доступні 6 функцій:

- Адміністрування – дозволяє проводити налаштування системи;
- Анкети – дозволяє переглянути всі анкети були створені раніше, створити нову анкету(нову чи на основі іншої);
- Опитування – дозволяє налаштувати анкету для певної аудиторії, вибрати дати проведення опитувань, та запустити їх;
- Налаштування – дає можливість налаштування профіля;

- Звіт за період – дає можливість переглянути звітність за певний період часу за певною освітньою програмою;
- Вихід – реалізує вихід з модуля.

Для початку роботи потрібно перейти на вкладку «Анкети», після чого вибрати створення нової анкети з меню:



Рис. 2.28. Меню «Анкети»

Далі відкриється вікно, де потрібно вказати налаштування анкети, і заповнити її питання. Коли анкета буде сформована, слід перейти на меню «Опитування», де можна побачити та керувати вже існуючими опитуваннями чи створити нове. При створенні нового опитування слід вказати на основі якої анкети буде створене опитування, вибрати аудиторію опитування, та вказати терміни проведення опитування. Після створення опитування воно з'явиться в таблиці активних опитувань на вкладці меню «Опитування». Потім існує можливість поділитися посиланням на опитування, або ж завершити його. Після завершення опитування можна переглянути результати у вигляді звіту.

3.4. Технічне та системне забезпечення розробки

3.4.1. Обґрунтування вибору технічних засобів

При створенні функціональної та концептуальної моделей ІС було використано CASE – засіб AllFusion Process Modeler r7, який призначений для моделювання, аналізу, оптимізації бізнес – процесів в нотаціях IDEF0, DFD, IDF3. Даний програмний засіб є ефективним інструментом для опису, реорганізації та створення нових бізнес процесів підприємств та інших установ.

Користувачі можуть використовувати erwin Data Modeler як спосіб створення концептуальної моделі даних та створення логічної моделі даних, яка не залежить від конкретної технології бази даних. Ця схематична модель може бути використана для створення фізичної моделі даних. Потім користувачі можуть направити інженерну мову визначення даних, необхідний створення схеми для низки систем управління базами даних. Програмне забезпечення включає функції для графічної модифікації моделі, у тому числі діалогові вікна для вказівки кількості взаємозв'язків сутностей, обмежень бази даних, індексів і унікальності даних. Ервін підтримує три мови моделювання даних: IDEF1X, варіант розробки інформаційних технологій, розроблений Джеймсом Мартіном, та форму позначення розмірного моделювання.

Програмне забезпечення дозволяє користувачам створювати моделі даних шляхом зворотного проектування вже існуючих баз даних, заснованих на декількох різних форматах. Ще одна включена функція - це здатність Ервіна створювати багаторазово використовувані стандарти проектування: "включаючи стандарти іменування, стандарти типів даних, шаблони моделей. Програмне забезпечення включає кілька функцій для зміни способу відображення моделі даних, включаючи параметри для декількох кольорів, шрифти, схеми, тематичні області та макети.

Функція повного порівняння erwin дозволяє користувачеві порівнювати дві версії моделі, відображає відмінності та дозволяє об'єднувати та оновлювати дані у будь-якому напрямку. [5].

В якості системи управління базами даних в даному проекті було обрано MySQL, що була розроблена корпорацією «ТсХ» для підвищення швидкодії обробки великих баз даних. СКБД має ряд суттєвих переваг, за що її і обрали:

1. Використовуваний основний потік є багатопоточним і підтримує кілька процесорів.

2. Існує кілька типів стовпців: 1, 2, 3, 4 та 8-байтове ціле число без знака / зі знаком, FLOAT, DOUBLE, CHAR, VARCHAR, TEXT, BLOB, DATE, TIME, DATETIME, TIMESTAMP, YEAR та Тип ENUM .

3. Він реалізує бібліотеку функцій SQL через високооптимізовану бібліотеку класів і працює настільки швидко, наскільки це можливо, зазвичай після ініціалізації запиту не повинно бути виділення пам'яті. Жодних витоків пам'яті.

4. Він повністю підтримує пропозиції SQL GROUP BY та ORDER BY та підтримує агреговані функції (COUNT(), COUNT(DISTINCT), AVG(), STD(), SUM(), MAX() та MIN()). Ви можете змішувати таблиці різних баз даних в одному запиті.

5. Підтримка LEFT OUTER JOIN та ODBC ANSI SQL.

6. Усі стовпці мають значення за промовчанням. Ви можете використовувати INSERT для вставки підмножини стовпця таблиці, і для тих стовпців, які не потрібно вказувати явно, встановлюються стандартні значення.

7. MySQL може працювати на різних платформах. Підтримка C, C++, Java, Perl, PHP, Python та TCL API [6].

Для створення інтерфейсу користувача ІС було обрано фреймворк Symfony. Symfony надає всі інструменти, які уам потрібні для використання баз даних у уашій програмі завдяки Doctrine, кращому набору бібліотек PHP для роботи з базами даних. Symfony спрямований на прискорення створення та підтримки вебзастосунків, а також для уникнення витрат часу для розв'язування тривіальних задач у розробці (наприклад, написання валідаторів форм). Symfony використовує шаблон проектування модель-вид-контролер. За допомогою

акселератора PHP Symfony збільшує продуктивність та зменшує навантаження на сервер.

Symfony ставить за мету дати розробникам повний контроль над конфігурацією: переважну більшість можливостей можливо налаштувати, від структури каталогів до сторонніх бібліотек [7].

3.4.2. Обґрунтування вибору ОС та протоколу обміну даними

Розроблюваний проект буде розташовуватися на хостингу під лінукс. Лінукс був обраний через: безкоштовність(встановивши Linux, ви отримаєте набір із тисяч безкоштовні програми. Хоч вони й не настільки звичні як Windows-програми, але абсолютно функціональні), надійність(коректна робота апаратної частини вашого ПК, дозволить Linux'у працювати роки без перезавантаження та зависань), безпеку(У Linux практично немає вірусів. Саме побудова операційної системи виключає роботу шкідливих програм. І тому ви можете обійтися без антивірусних програм, що гальмують комп'ютер і заважає працювати. Не потрібно весь час оновлювати антивірусні бази та перевіряти жорсткий диск на віруси, втрачаючи безцінний час), відкритий вихідний код(Це дає можливість використовувати і модифікувати код за власним бажанням. Можна будь-якої миті виправити якісь помилки або недоліки системи, а також розширити її функціональність, шляхом написання доповнень або програм, що працюють під її керуванням).

Дистрибутиви Linux вже давно використовуються в як серверні операційні системи і зайняли значну частку цього ринку; за даними компанії Netcraft на лютий 2014 року, сім з десяти найнадійніших інтернет-компаній, надають хостинг, використовують Linux на своїх веб-серверах. Linux є ключовим компонентом комплексу серверного комплексу програмного забезпечення LAMP(Linux, Apache, MariaDB/MySQL, Perl /PHP/Python), який набув популярності серед веб-розробників і став однією з найбільш найпоширеніших платформ для хостингу веб-сайтів [8].

Отже з огляду на представлені вище позитивні властивості цієї операційної системи, вибір даної платформи для розробки програмних засобів, вважається досить обґрунтованим.

3.4.3. Заходи захисту від несанкціонованого доступу до систему

На сьогодні, веб-вразливості справді перевершують по кількості і можливій шкоді будь-які інші проблеми інформаційної безпеки. Більшість зовнішніх атак на корпоративні інформаційні системи націлені саме на вразливості веб-додатків.

Серед основних загроз на рівні з використанням програмних вразливостей, зловмисники активно використовують і фішингові інструменти, які побудовані на помилках людей. Від цього нам нікуди не втекти, однак спеціалісти з інформаційної безпеки також не сидять склавши руки. Сьогодні на ринку вже доступні ефективні інструменти захисту веб-додатків.

Один з основних, це – WAF (Web Application Firewall). Суть якого полягає в захисті WEB ресурсів, що виставлені назовні.

Це рішення дозволяє запобігати багатьом видам атак. Серед них SQL і PHP-ін'єкції, cross-site scripting (XSS), підбір пароля, експлуатація вразливостей zero-day в веб-додатках, DDoS-атаки рівня додатків і атаки на ресурсомісткі сторінки.

Для атак на веб-додатки зловмисники активно використовують вразливості нульового дня (0-day), що робить марними сигнатурні методи аналізу. Замість цього потрібно аналізувати мережевий трафік і системні журнали для створення моделі нормального функціонування програми, і на основі цієї моделі виявляти аномальну поведінку системи. WAF у силу своєї архітектури може розібрати весь сеанс зв'язку користувача, і тому здатний на більш поглиблений поведінковий аналіз, ніж NGFW. Такий підхід дозволяє виявляти атаки з використанням автоматичних засобів (сканування, підбір паролів, DDoS, фрод, залучення в ботнети).

Існує також клас атак, наприклад CSRF, що спрямовані саме на клієнта веб-додатку. Оскільки трафік атаки не проходить через захисний периметр, на перший погляд захистити користувача доволі складно, однак все ж можливо [9].

ВИСНОВКИ

Всі функції в розробленому модулі анкетування ефективно виконують всі визначені задачі автоматизації, що були представлені в першому розділі. Щоб продемонструвати користь від створення, було проведено аналіз діяльності відділу «до» та «після» впровадження системи.

До впровадження системи був ряд проблем:

- анкетування проводилося тільки за однією анкетною;
- не було можливості формування та редагування анкет;
- кожен раз необхідно було створювати опитування для кожної групи, при цьому йшло навантаження на викладача;
- обробка результатів була направлена тільки на одну анкету;
- результати представлялися одним масивом даних, що вивантажувався у excel;
- звіт формувался тільки по кожній окремій групі;
- призначення анкети для груп прописувалося вручну;
- анкети існували не довгий період часу, кожен рік потрібно було створювати наново.

Після розробки даного модуля всі ці проблеми були враховані та виправлені. Тепер анкети можна редагувати, і вони існують стільки часу скільки потрібно. Завжди можна створити нову анкету на основі іншої. Забезпечений вибір аудиторії опитуваних за списками всіх існуючих в університеті груп які знаходяться в БД. Все це покращує та пришвидшує роботу відділу в рази.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Якість вищої освіти: теорія і практика: навчально-методичний посібник
URL: https://lib.iitta.gov.ua/717657/1/Посібник_якість%20вищої%20освіти.pdf
2. Історія НУХТ | НУХТ. *Національний університет харчових технологій* | НУХТ. URL: <https://nuft.edu.ua/istoriya-nuft> .
3. Положення про ВСЯО URL: [Положення про ВСЯО.pdf - Google Диск](#).
4. Центр моніторингу якості та координації освітньої діяльності університету | НУХТ. *Національний університет харчових технологій* | НУХТ. URL: <https://nuft.edu.ua/sluzhby-ta-viddily/czentr-monitoringu-yakosti-ta-koordinacziyi-osvitnoyi-diyaln/>.
5. Erwin Data Modeler. URL: [ERwin Data Modeler — Вікіпедія \(wikipedia.org\)](#)
6. переваги та недоліки mysql, а також концепція горячого і холодного резервування. URL: <https://russianblogs.com/article/7589764876/>.
7. Учасники проєктів Вікімедіа. Symfony – Вікіпедія. *Вікіпедія*. URL: <https://uk.wikipedia.org/wiki/Symfony>.
8. Linux под разные задачи - презентація онлайн. URL: <https://ppt-online.org/481749>.
9. Захист WEB-додатків. Чому це актуально? - Octava Capital. *Octava Capital*. URL: <https://octavacapital.ua/zahyst-web-dodatkiv-chomu-ce-aktualno/>.

ДОДАТКИ

Додаток А

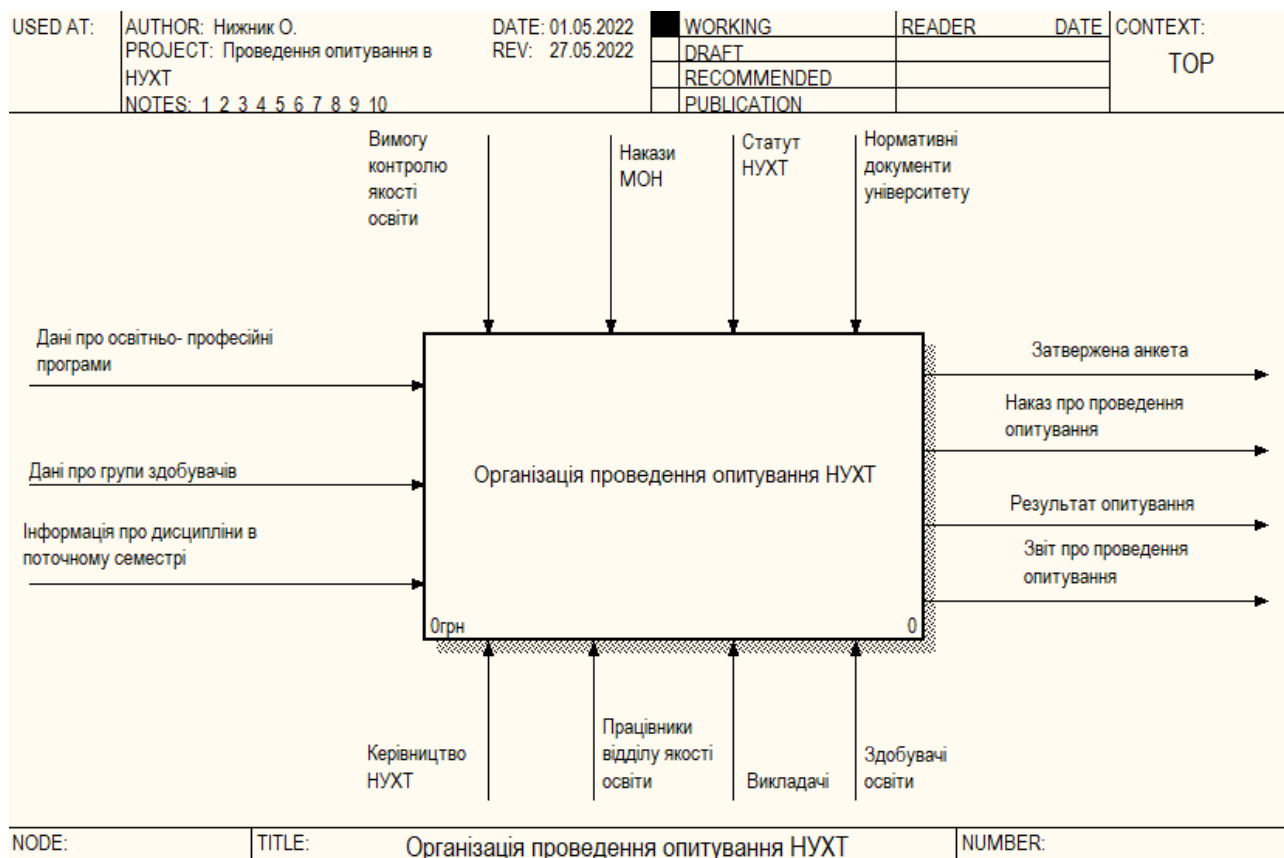


Рис. А.1. Концептуальна модель «Організація проведення опитування НУХТ»

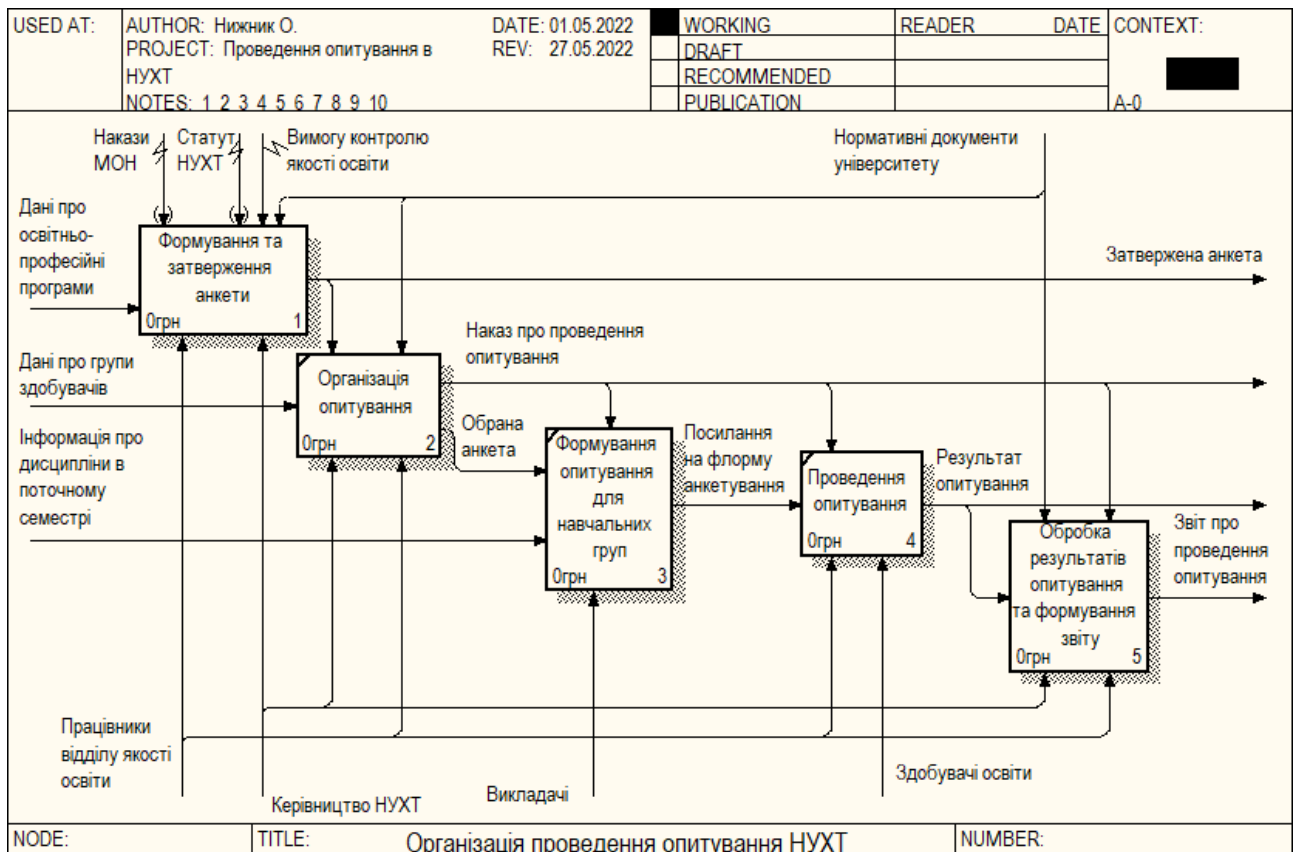


Рис. А.2. Діаграма першого рівня декомпозиції моделі

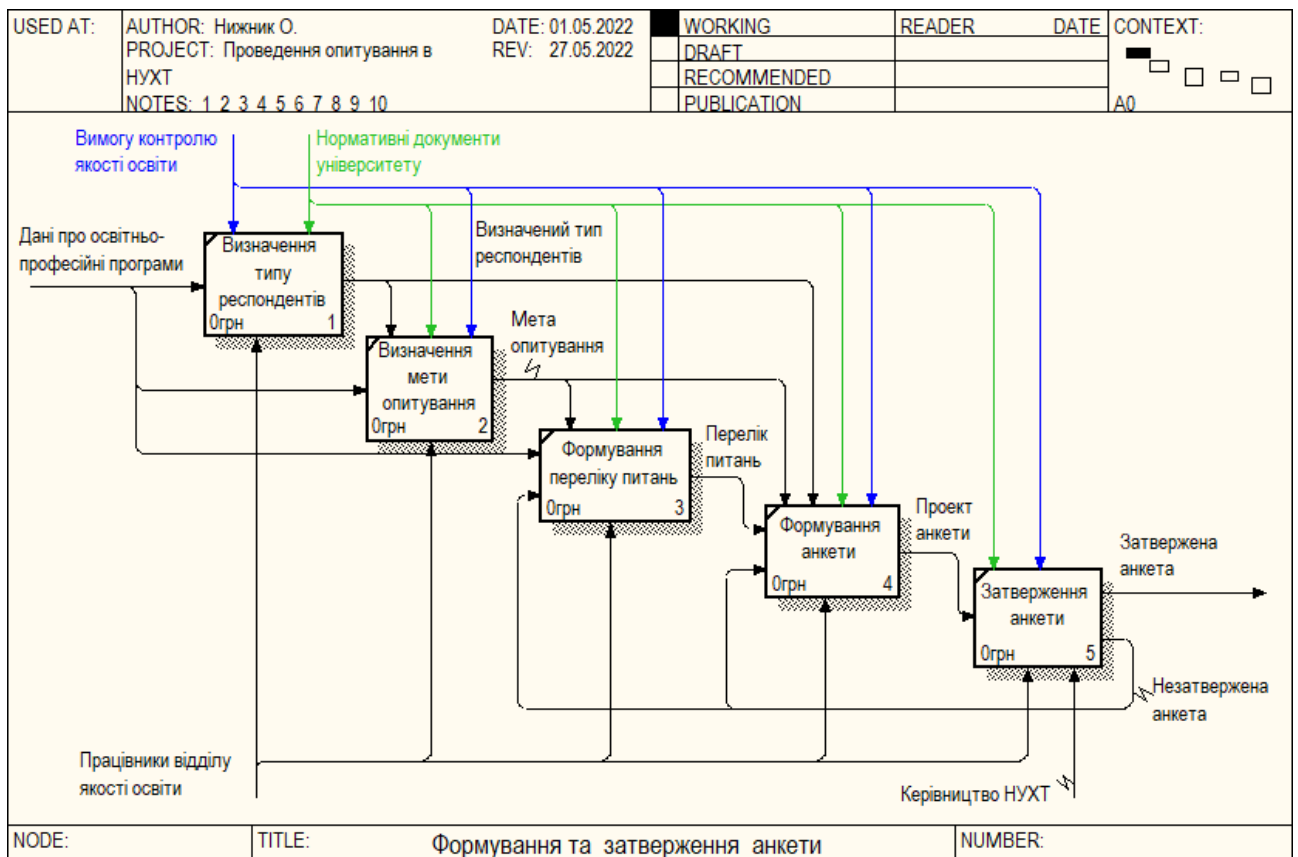


Рис. А.3. Діаграма декомпозиції процесу «Формування та затвердження анкети»

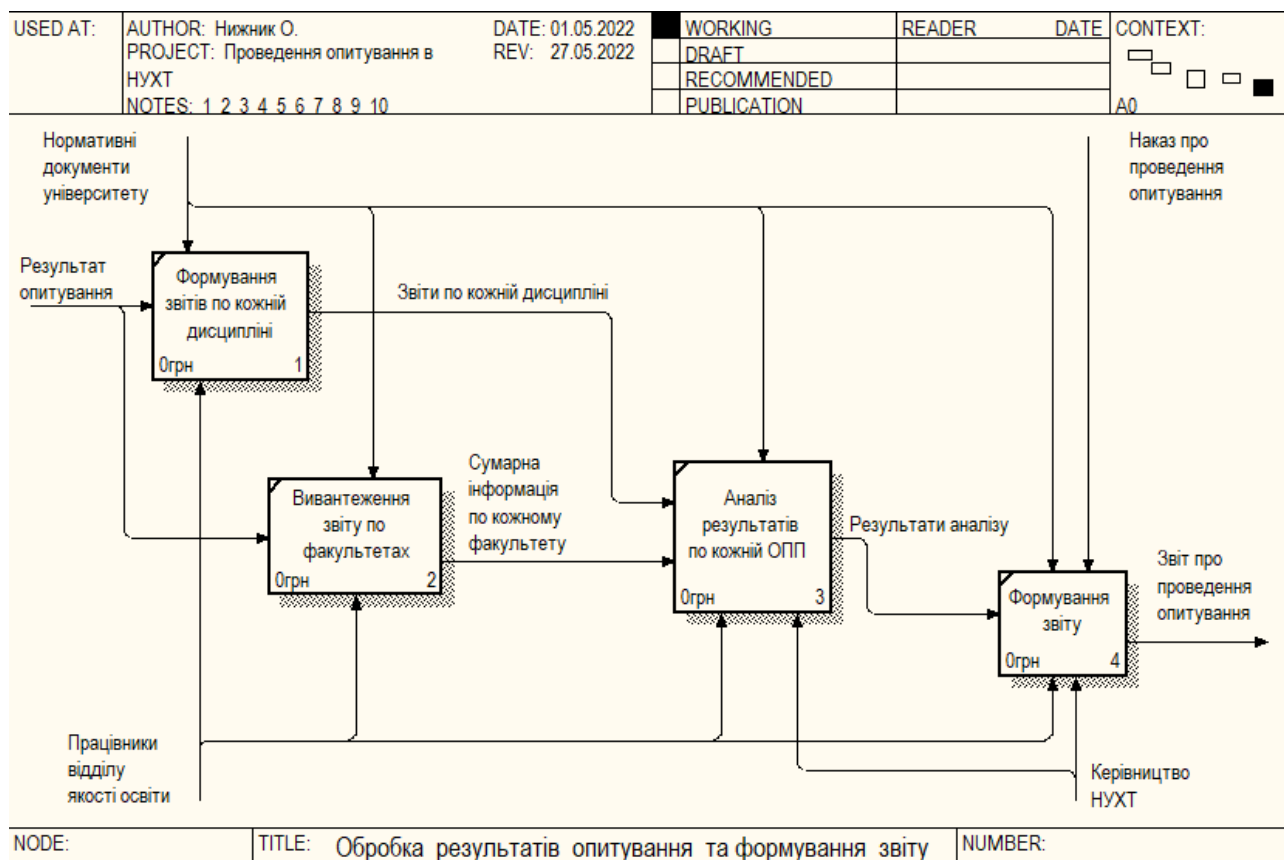


Рис. А.4. Діаграма декомпозиції процесу «Обробка результатів опитування та формування звіту»

Додаток Б

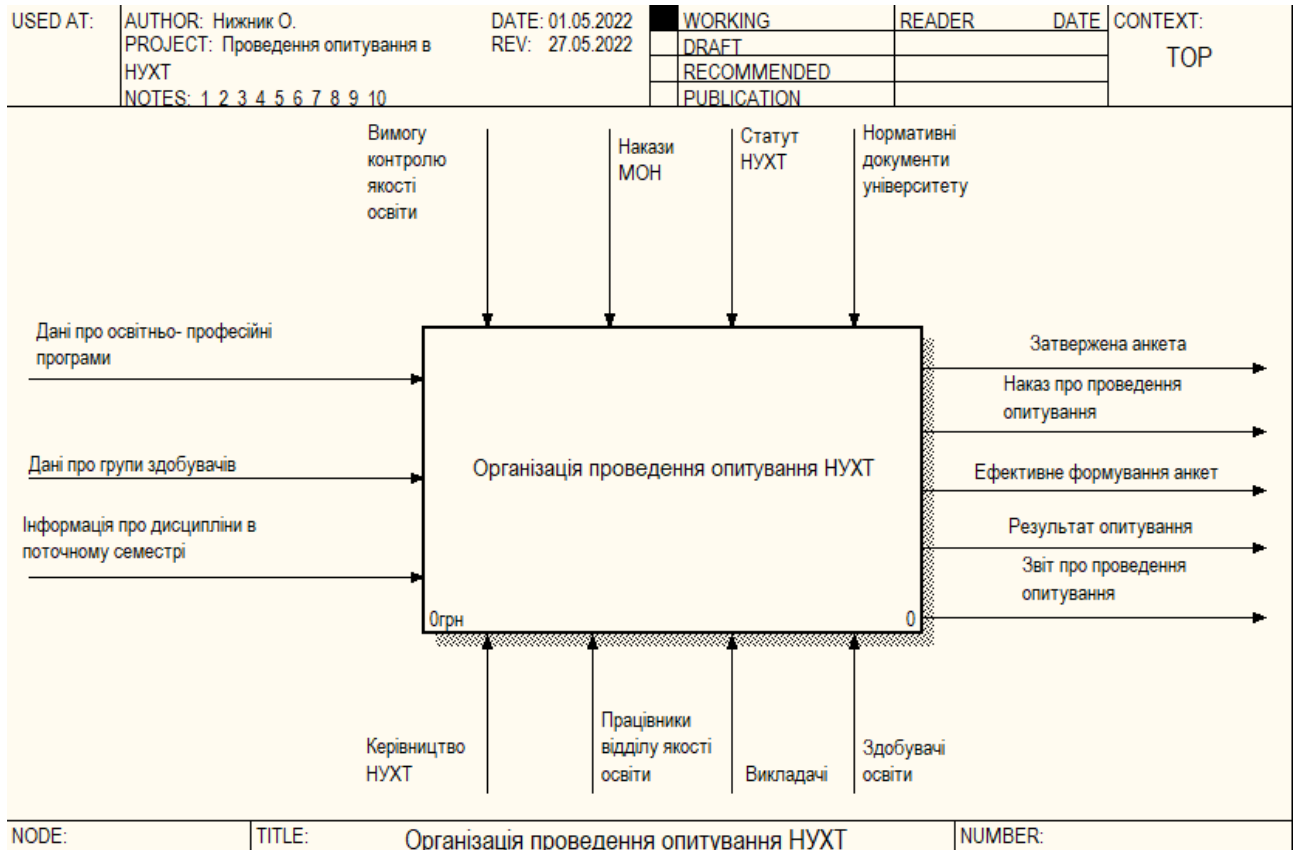


Рис. Б.5. Концептуальна модель «Організація проведення опитування НУХТ»

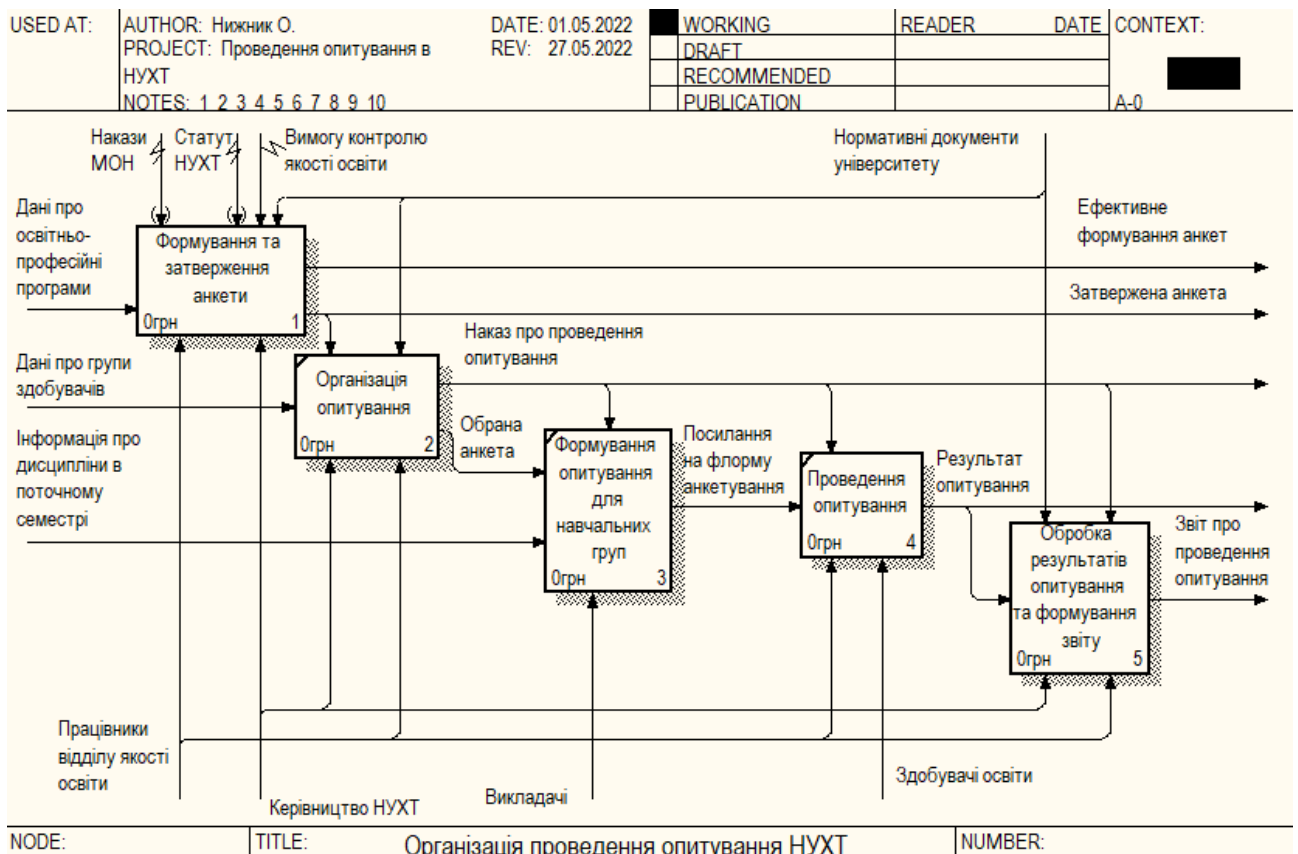


Рис. Б.6. Діаграма першого рівня декомпозиції моделі

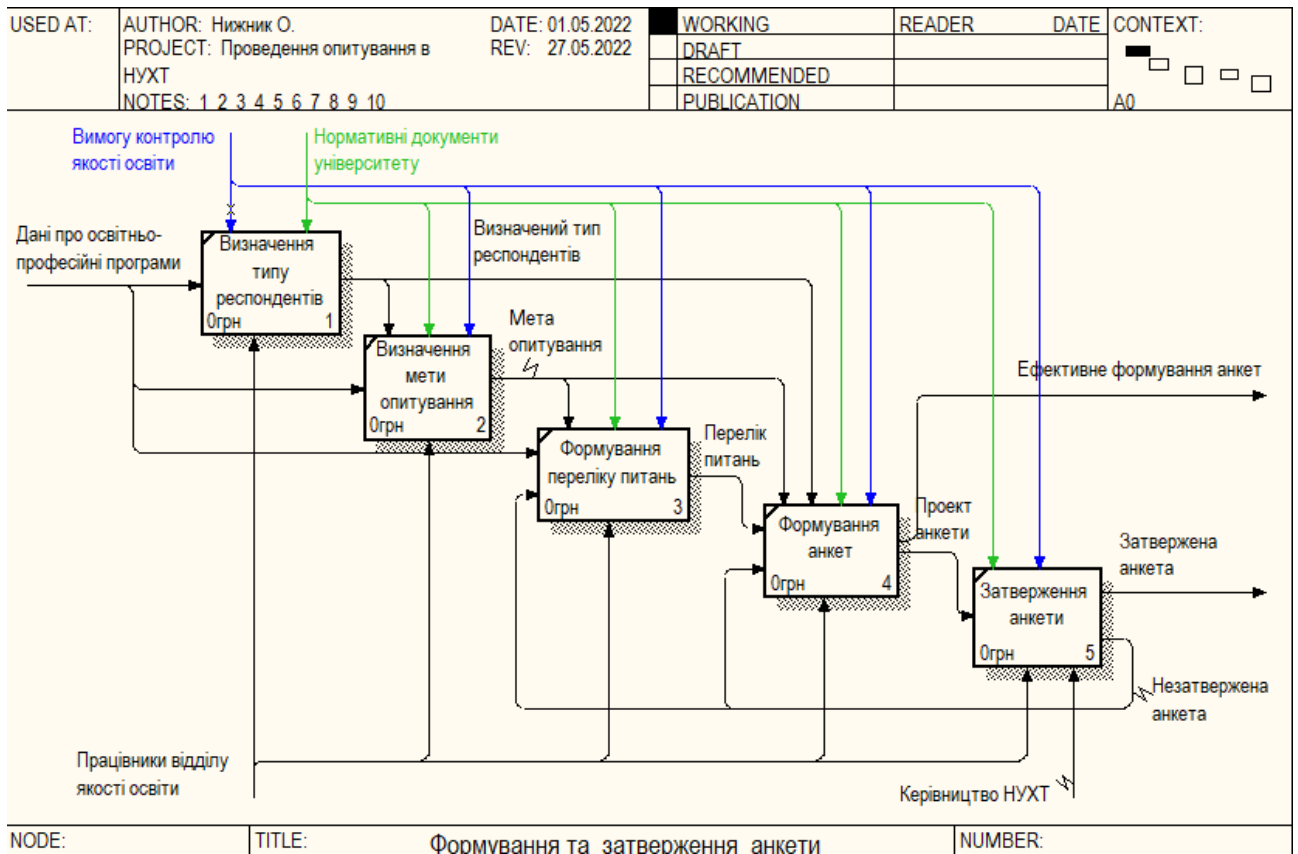


Рис. Б.7. Діаграма декомпозиції процесу «Формування та затвердження анкети»

Додаток Г

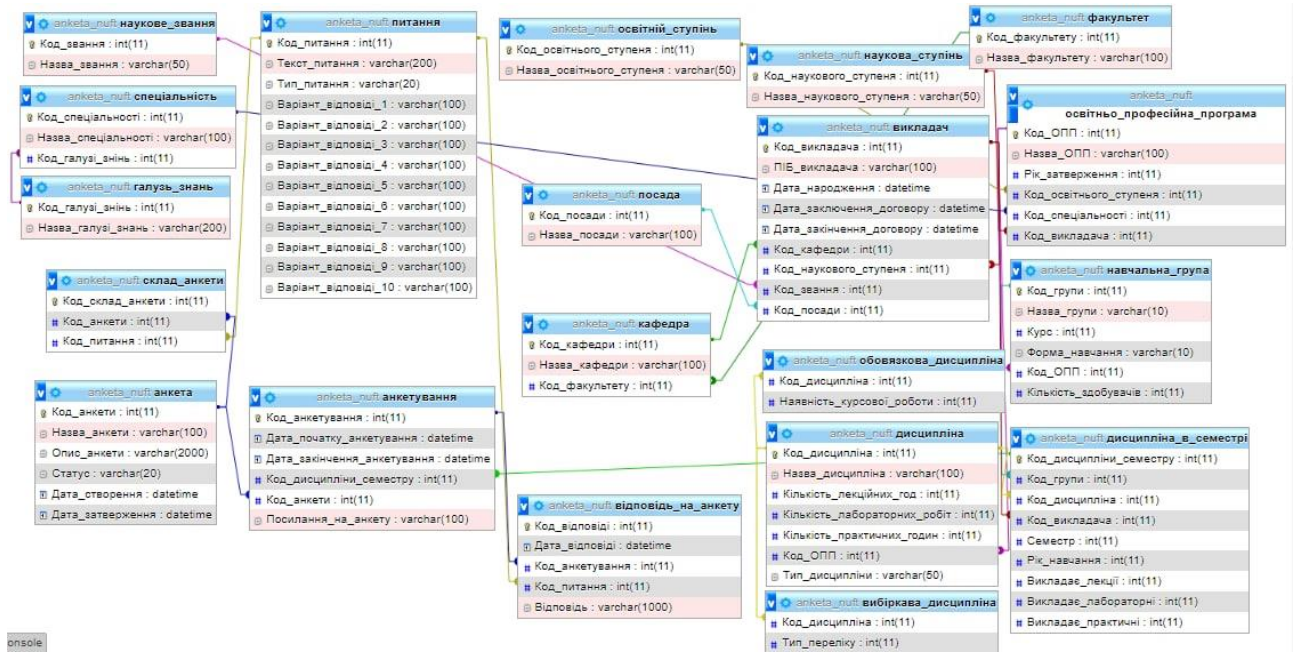


Рис. Г.11. Схема бази даних в MY SQL

Додаток Д

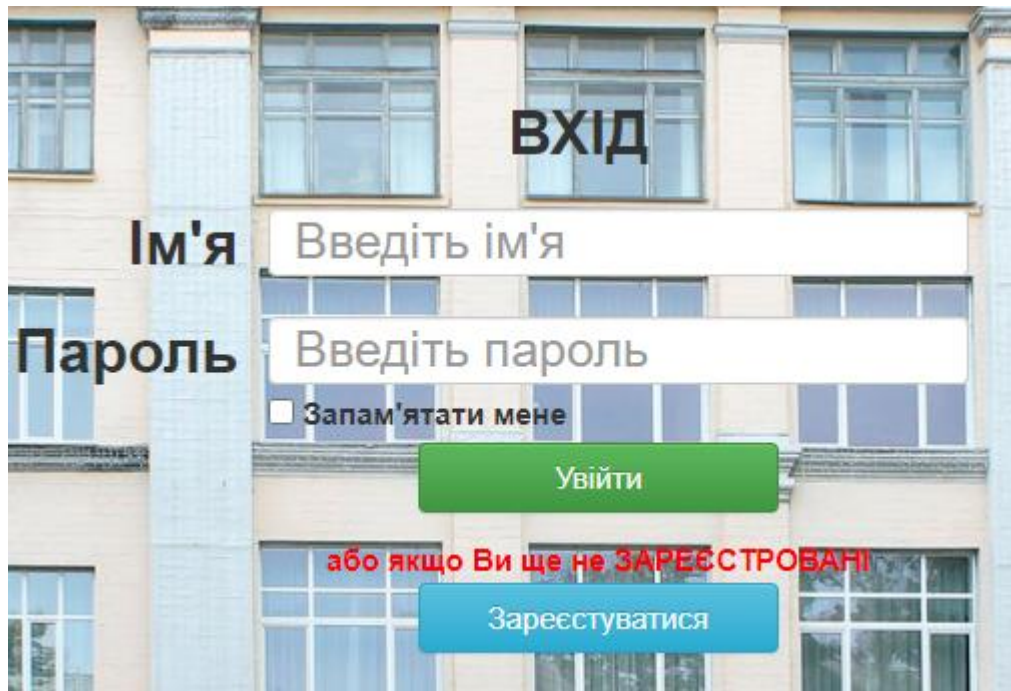


Рис. Д.12. Форма авторизації

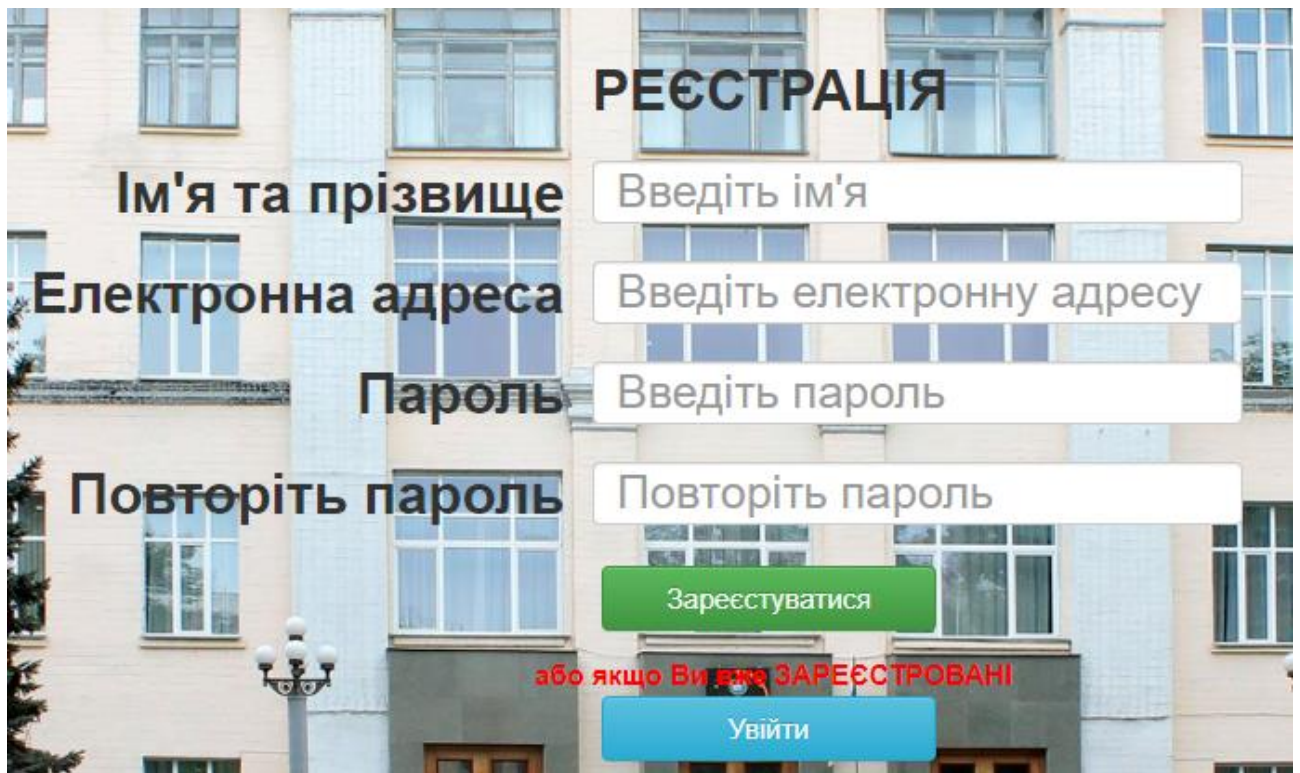


Рис. Д.13. Форма реєстрації нового користувача



Рис. Д.14. Головне меню



Рис. Д.15. Меню «Анкети»

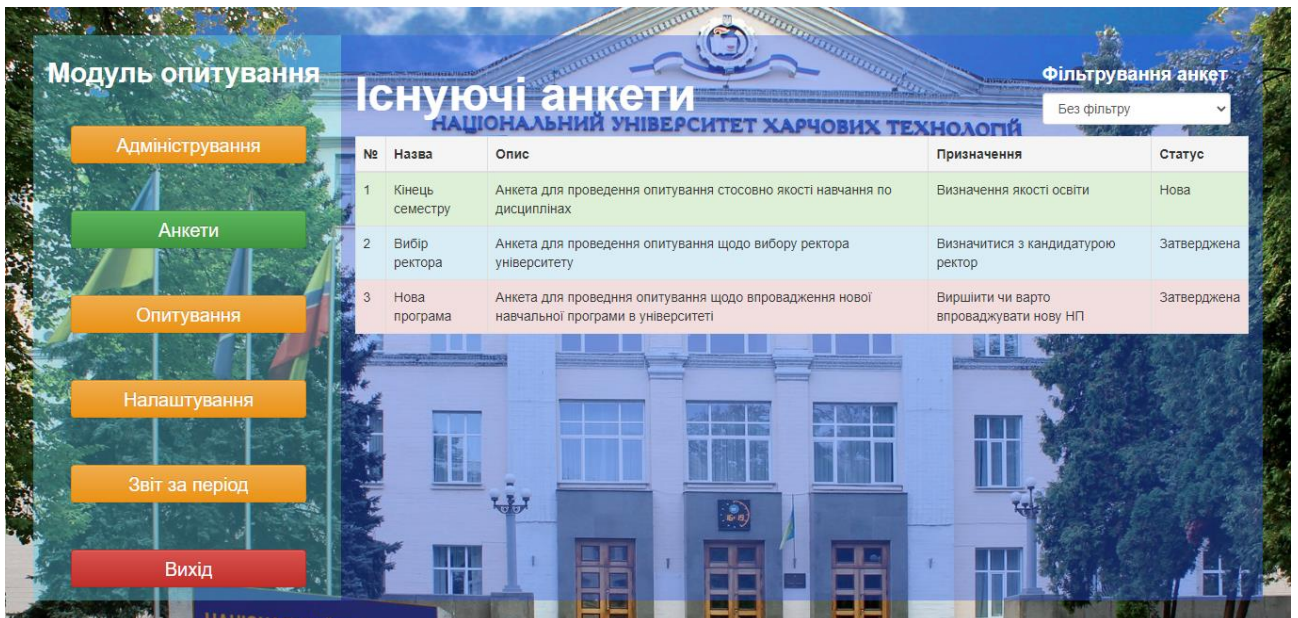


Рис. Д.16. Перегляд анкет

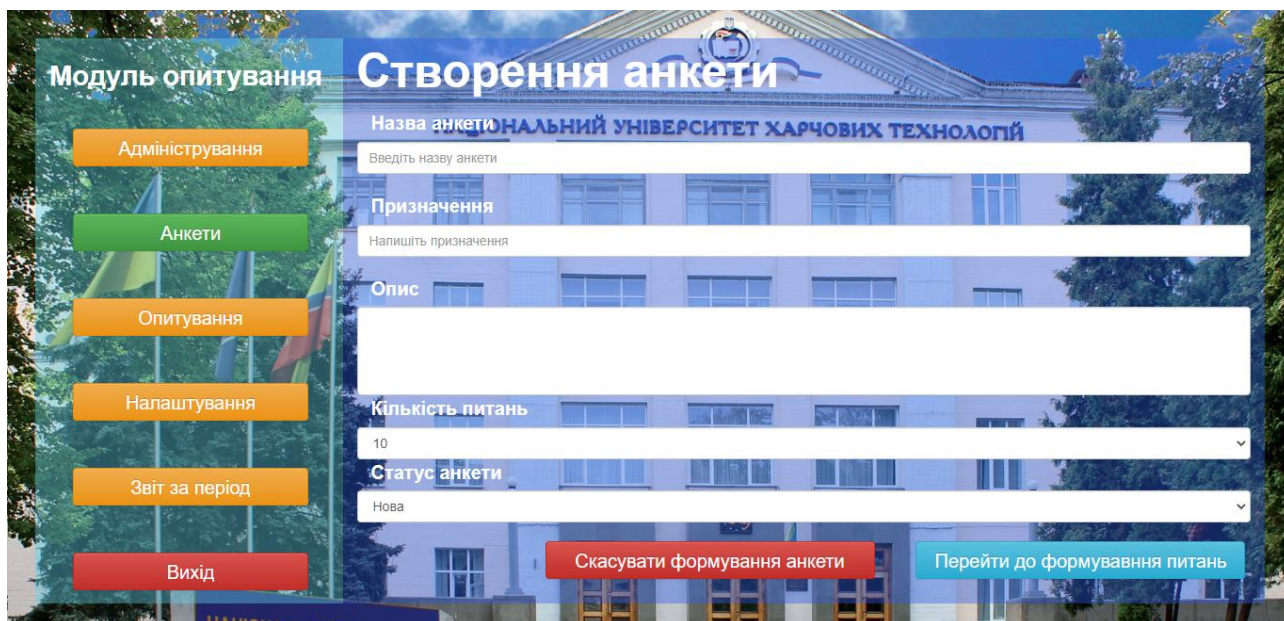


Рис. Д.17. Вікно створення нової анкети

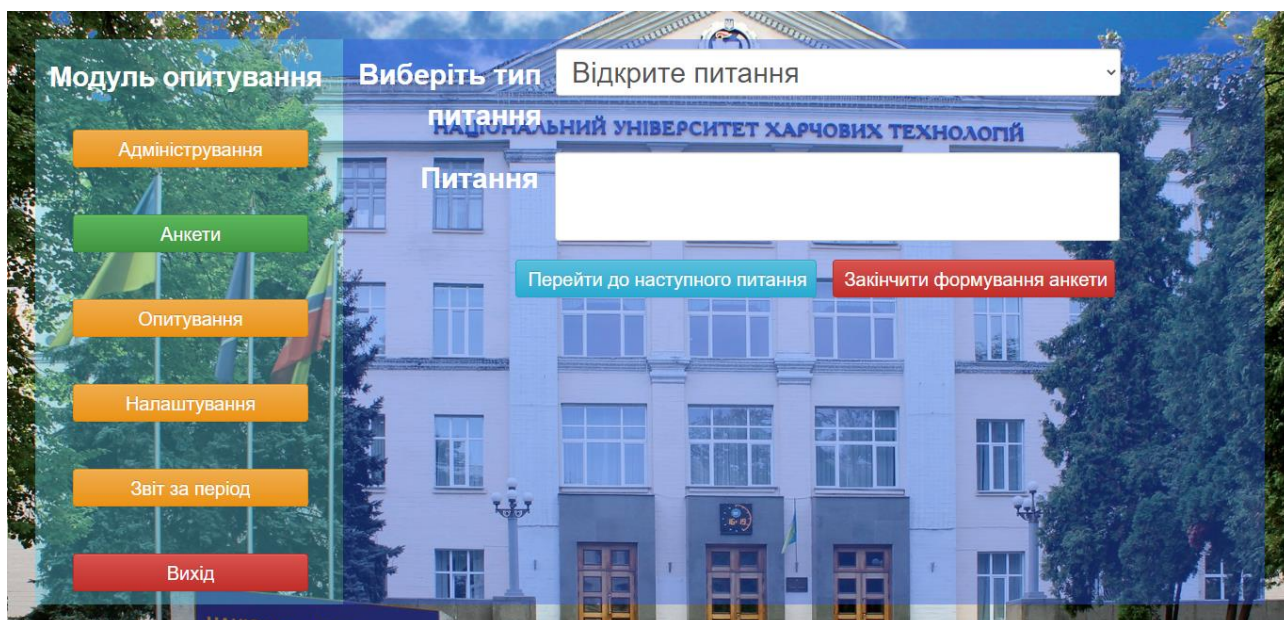


Рис. Д.18. Вікно формування питань

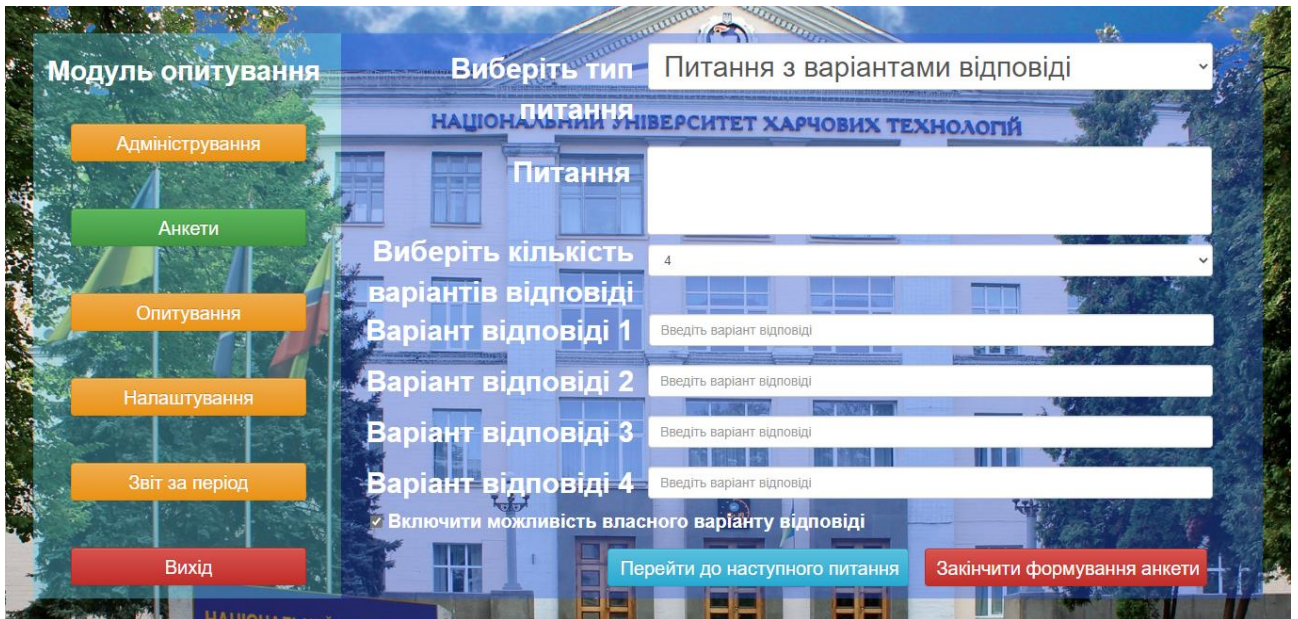


Рис. Д.19. Вигляд вікна додавання питання з варіантами відповіді

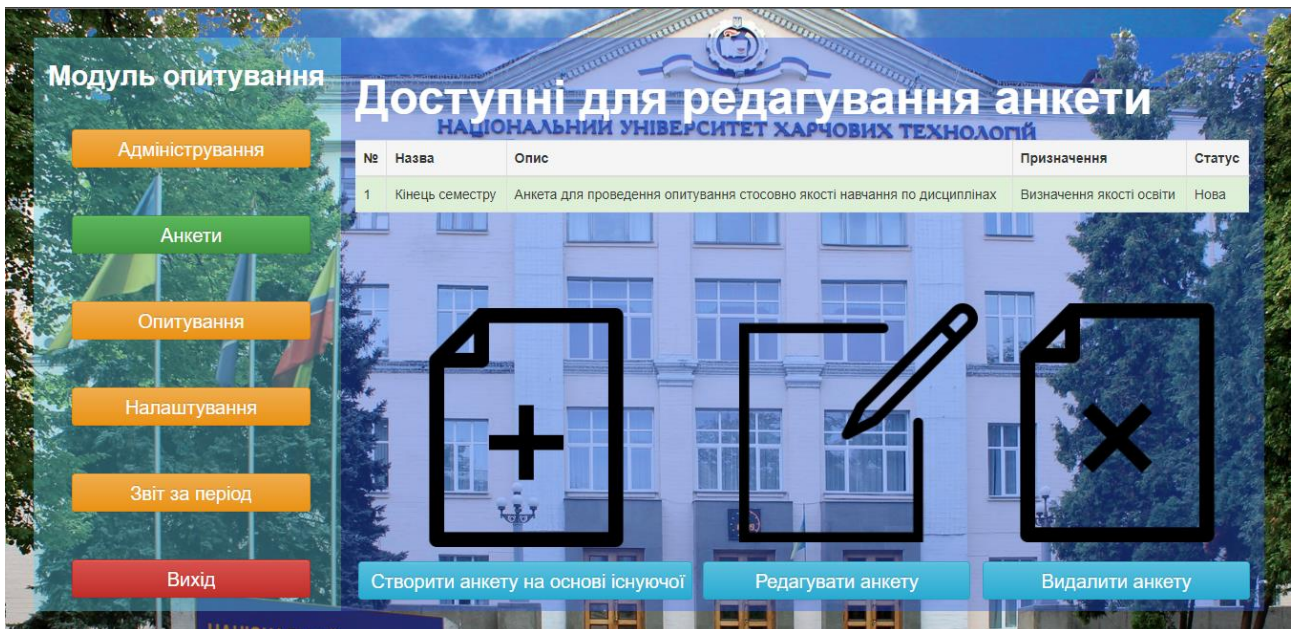


Рис. Д.20. Меню редагування анкет

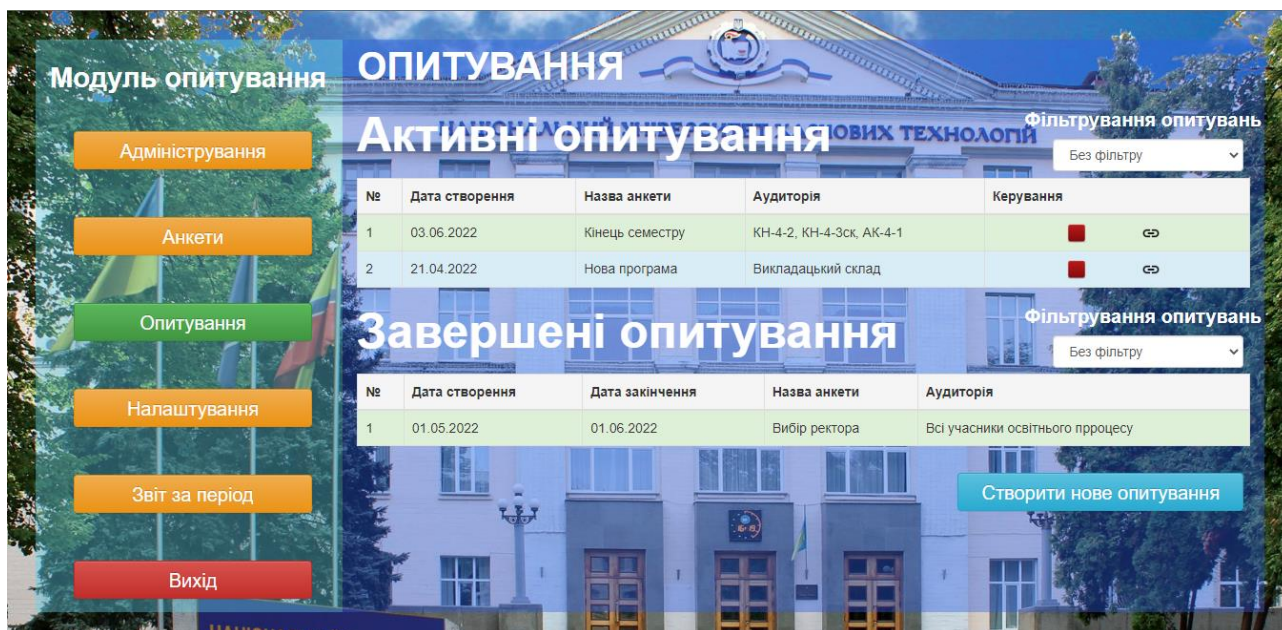


Рис. Д.21. Вигляд вікна «Опитування»

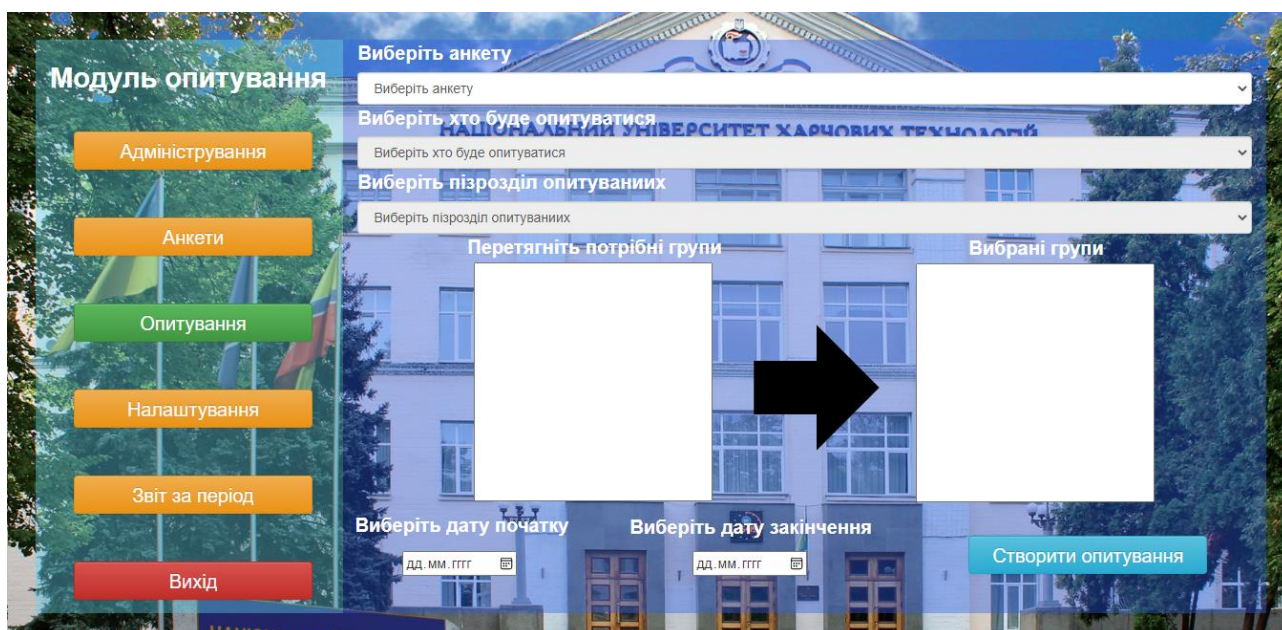


Рис. Д.22. Вигляд вікна створення нової анкети

Додаток Е

Авторизація користувача:

```
<?php
namespace Phppot;
use \Phppot\DataSource;

$message = "";
if (count($_POST) > 0) {
    $isSuccess = 0;
    require_once __DIR__ . '/DataSource.php';
    $conn = new DataSource();
```

```

$query = 'SELECT * FROM Викладач WHERE ПІБ= ?';
$params = array(
    $_POST["ПІБ"]
);
$result = $conn->select($query, $params);

if (! empty($result)) {

    $hashedPassword = $result[0]["Пароль"];
    if (Password_verify($_POST["Пароль"], $hashedPassword)) {
        $isSuccess = 1;
    }
}
if ($isSuccess == 0) {
    $message = "Неправильне ім'я чи пароль!";
} else {
    header("Location: ./success-message.php");
}
?>

```

Реєстрація користувача:

```

<?php
include_once 'db.php';
if(isset($_POST['submit']))
{
    $name = $_POST['ПІБ'];
    $email = $_POST['email'];
    $password = $_POST['Пароль'];
    $sql = "INSERT INTO Викладач (ПІБ,email, пароль)
    VALUES ('$name','$email', '$password')";
    if (mysqli_query($conn, $sql)) {
        echo "Успіх";
    } else {
        echo "Помилка: " . $sql . ":-" . mysqli_error($conn);
    }
    mysqli_close($conn);
}
?>

```

Фільтрування анкет:

```

<?php
$query = sprintf("SELECT Анкета.Назва, Анкета.Опис, Анкета.Призначення, Анкета.Статус
FROM Анкета

```

```

WHERE Анкета.Статус='Затверджена',
mysql_real_escape_string($Анкета.Статус);
$result = mysql_query($query);
// Тут перебираємо всі значення які відповідають запиту і записуємо в таблицю
while ($row = mysql_fetch_assoc($result)) {
    echo $row['Анкета.Назва'];
    echo $row['Анкета.Опис'];
    echo $row['Анкета.Призначення'];
    echo $row['Анкета.Статус'];
}
mysql_free_result($result);
?>

```

Зняття фільтру:

```

<?php
$query = sprintf("SELECT Анкета.Назва, Анкета.Опис, Анкета.Призначення, Анкета.Статус
FROM Анкета;
$result = mysql_query($query);
// Тут перебираємо всі значення які відповідають запиту і записуємо в таблицю
while ($result) {
    echo $row['Анкета.Назва'];
    echo $row['Анкета.Опис'];
    echo $row['Анкета.Призначення'];
    echo $row['Анкета.Статус'];
}
mysql_free_result($result);
?>

```

Додавання відкритого питання:

```

<?php
include_once 'db.php';
if(isset($_POST['submit']))
{

```

```

$text = $_POST['Текст_питання'];
$type = $_POST['Тип_питання'];
$sql = "INSERT INTO Питання (Текст_питання,Тип_питання)
VALUES ('$text','$type)";
if (mysqli_query($conn, $sql)) {
    echo "Успіх";
} else {
    echo "Помилка: " . $sql . ":-" . mysqli_error($conn);
}
mysqli_close($conn);
}
?>

```

Додавання питання з варіантами відповіді:

```

<?php
include_once 'db.php';
if(isset($_POST['submit']))
{
    $text = $_POST['Текст_питання'];
    $type = $_POST['Тип_питання'];
    $answer_1 = $_POST['Варіант_відповіді_1'];
    $answer_2 = $_POST['Варіант_відповіді_2'];
    $answer_3 = $_POST['Варіант_відповіді_3'];
    $answer_4 = $_POST['Варіант_відповіді_4'];
    $sql = "INSERT INTO Питання (Текст_питання,Тип_питання,
Варіант_відповіді_1,Варіант_відповіді_2,Варіант_відповіді_3,Варіант_відповіді_4)
VALUES ('$text','$type', '$answer_1','$answer_2', '$answer_3', '$answer_4)";
    if (mysqli_query($conn, $sql)) {
        echo "Успіх";
    } else {
        echo "Помилка: " . $sql . ":-" . mysqli_error($conn);
    }
    mysqli_close($conn);
}
?>

```

Редагування питання:

```

<?php
include_once 'db.php';
if(isset($_POST['submit']))
{
    $text = $_POST['Текст_питання'];
    $type = $_POST['Тип_питання'];
    $sql = "UPDATE Питання (Текст_питання,Тип_питання)
SET (Текст_питання = '$text', Тип_питання = '$type')
WHERE Текст_питання = ?";
}

```

```

if (mysqli_query($conn, $sql)) {
    echo "Успіх";
} else {
    echo "Помилка: " . $sql . ":-" . mysqli_error($conn);
}
mysqli_close($conn);
}
?>

```

Або:

```

<?php
include_once 'db.php';
if(isset($_POST['submit']))
{
    $text = $_POST['Текст_питання'];
    $type = $_POST['Тип_питання'];
    $answer_1 = $_POST['Варіант_відповіді_1'];
    $answer_2 = $_POST['Варіант_відповіді_2'];
    $answer_3 = $_POST['Варіант_відповіді_3'];
    $answer_4 = $_POST['Варіант_відповіді_4'];
    $sql = " UPDATE Питання (Текст_питання,Тип_питання,
Варіант_відповіді_1,Варіант_відповіді_2,Варіант_відповіді_3,Варіант_відповіді_4)
    SET (Текст_питання ='$text', Тип_питання ='$type', Варіант_відповіді_1='$answer_1',
Варіант_відповіді_2='$answer_2', Варіант_відповіді_3='$answer_3',
Варіант_відповіді_4='$answer_4')
    WHERE Текст_питання = ?";
    if (mysqli_query($conn, $sql)) {
        echo "Успіх";
    } else {
        echo "Помилка: " . $sql . ":-" . mysqli_error($conn);
    }
    mysqli_close($conn);
}
?>

```