

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ХАРЧОВИХ ТЕХНОЛОГІЙ**

**Інститут (факультет) Автоматизації і комп'ютерних систем
Кафедра Інформаційних технологій, штучного інтелекту і кібербезпеки**

«До захисту в ЕК»
Директор інституту(декан факультету)
Андрій ФОРСЮК
(підпис) (ім'я та прізвище)

« 02 » червня 2025р.

«До захисту допущено»
Завідувач кафедри
Сергій ГРИБКОВ
(підпис) (ім'я та прізвище)

« 02 » червня 2025р.

**КВАЛІФІКАЦІЙНА РОБОТА
НА ЗДОБУТТЯ ОСВІТНЬОГО СТУПЕНЯ БАКАЛАВРА**

зі спеціальності 122 «Комп'ютерні науки»

(код та назва спеціальності)

освітньо-професійної програми Комп'ютерні науки

на тему: «Розроблення вебсайту проблемної науково-дослідної лабораторії
НУХТ»

Виконав: здобувач 4 курсу, групи КН-4-2

Сиротенко Дмитро Олександрович

(прізвище, ім'я по батькові повністю)

(підпис)

Керівник Харкянен Олена Валеріївна

(прізвище, ім'я та по батькові повністю)

(підпис)

Консультанти

(ім'я та прізвище)

(підпис)

(ім'я та прізвище)

(підпис)

(ім'я та прізвище)

(підпис)

Рецензент

(ім'я та прізвище)

(підпис)

Я як здобувач(ка) Національного університету харчових технологій розумію і підтримую політику університету з академічної доброчесності. Я не надавав(-ла) і не одержував(-ла) недозволеної допомоги під час підготовки цієї роботи. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело

Здобувач

(підпис)

Київ – 2025 р.

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ХАРЧОВИХ ТЕХНОЛОГІЙ

Інститут(факультет) Автоматизації і комп'ютерних систем
 Кафедра Інформаційних технологій, штучного інтелекту і кібербезпеки
 Освітній ступінь бакалавр
 Спеціальність 122 «Комп'ютерні науки»
 Освітньо-професійна програма Комп'ютерні науки

ЗАТВЕРДЖУЮ

Завідувач кафедри
Інформаційних технологій, штучного
інтелекту і кібербезпеки
Сергій ГРИБКОВ

“ 28 ” квітня 2025 року

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧА

Сиротенка Дмитра Олександровича

(прізвище, ім'я, по батькові)

1. Тема роботи Розроблення вебсайту проблемної науково-дослідної лабораторії НУХТ
 керівник роботи Харкянен Олена Валеріївна, к.т.н., доцент
 (прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від 28 квітня 2025 року № 254-кв

2. Строк подання здобувачем роботи 30.04.2025 р.

3. Вихідні дані до роботи

Первинна інформація про ПНДЛ НУХТ, штатно-посадові інструкції ПНДЛ,
інформація про науково-дослідні проекти

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

Розділ 1. Аналіз діяльності підприємства та обґрунтування необхідності
розробки інформаційної системи

Розділ 2. Технічне завдання на створення системи

Розділ 3. Проектування, створення та апробація програмного забезпечення

5. Перелік графічного матеріалу

Організаційна структура ПНДЛ НУХТ

Моделі бізнес-процесів діяльності ПНДЛ НУХТ

Структура бази даних

Фрагменти інтерфейсу веб-застосунку

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	Харкянен О. В., доцент НУХТ	28.04.2025	08.05.2025
2	Харкянен О. В., доцент НУХТ	28.04.2025	16.05.2025
3	Харкянен О. В., доцент НУХТ	28.04.2025	19.05.2025

7. Дата видачі завдання 28.04.2025**КАЛЕНДАРНИЙ ПЛАН**

№	Назва етапів виконання кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Вибір теми та затвердження завдання	29.04.2025 - 30.04.2025	Виконано
2	Збір і аналіз інформації про об'єкт дослідження	01.05.2025 - 02.05.2025	Виконано
3	Формування технічного завдання	03.05.2025 - 14.05.2025	Виконано
4	Проектування архітектури та структури БД	05.05.2025 - 16.05.2025	Виконано
5	Реалізація інформаційної системи	07.05.2025 - 20.05.2025	Виконано
6	Розробка графічного інтерфейсу користувача	11.05.2025 - 22.05.2025	Виконано
7	Тестування функціональності та безпеки ІС	13.05.2025 - 25.05.2025	Виконано
8	Оформлення пояснювальної записки	16.05.2025 - 27.05.2025	Виконано
9	Підготовка презентації	28.05.2025 - 29.05.2025	Виконано

Здобувач

(підпис)

Сиротенко Д.О.

(прізвище та ініціали)

Керівник роботи

(підпис)

Харкянен О.В.

(прізвище та ініціали)

АНОТАЦІЯ

Сиротенко Д.О. «Розроблення вебсайту проблемної науково-дослідної лабораторії НУХТ». Кваліфікаційна робота бакалавра присвячена розробленню інформаційної системи у формі веб-сайту для підтримки функціонування Проблемної науково-дослідної лабораторії НУХТ. В основі проєкту — аналіз внутрішньої організації лабораторії, виявлення актуальних потреб у цифровізації та удосконаленні процесів обміну інформацією між замовниками та виконавцями досліджень.

У першому розділі проведено дослідження предметної області, описано структуру установи, напрями її діяльності, визначено ключові проблеми, що потребують технічного вирішення. У другому розділі сформовано технічне завдання: наведено перелік функціональних і нефункціональних вимог, зазначено програмні інструменти розробки, серед яких HTML, CSS, JavaScript, PHP, MySQL та Bootstrap. Третій розділ присвячено безпосередній реалізації: створено базу даних, розроблено інтерфейс користувача, реалізовано модулі реєстрації, оформлення замовлень, персональні кабінети, систему адміністрування, календар подій і чат. Також проведено тестування функціональності системи та здійснено оцінку її ефективності.

Пояснювальна записка містить 63 сторінки, 46 рисунків, 10 таблиць, 5 додатків та 27 літературних джерел.

Ключові слова: ВЕБРЕСУРС, ІНФОРМАЦІЙНА СИСТЕМА, ПНДЛ, НУХТ, PHP, MySQL, ЦИФРОВІЗАЦІЯ, НАУКОВІ ДОСЛІДЖЕННЯ.

SUMMARY

Syrotenko D.O. “Development of a Website for the Problem Scientific Research Laboratory of NUFT”.

The bachelor’s qualification thesis is dedicated to the development of an information system in the form of a website designed to support the operation of the Problem Scientific Research Laboratory of the National University of Food Technologies (NUFT). The project is based on an analysis of the laboratory's internal structure, identification of current needs for digital transformation, and improvement of information exchange processes between clients and research performers.

The first section investigates the subject area, describes the structure of the institution, its fields of activity, and highlights the key challenges requiring technical solutions. The second section defines the technical specifications, including a list of functional and non-functional requirements, and specifies the development tools used: HTML, CSS, JavaScript, PHP, MySQL, and Bootstrap. The third section is devoted to practical implementation: a database was designed, a user interface developed, and core modules implemented, including user registration, order submission, personal accounts, an administration system, an event calendar, and a messaging module. Functional testing was conducted, and the system’s efficiency was evaluated.

The explanatory note contains 63 pages, 46 figures, 10 tables, 5 appendices, and 27 references.

Keywords: WEB RESOURCE, INFORMATION SYSTEM, PROBLEM SCIENTIFIC RESEARCH LABORATORY, NUFT, PHP, MySQL, DIGITALIZATION, SCIENTIFIC RESEARCH.

ЗМІСТ

ВСТУП.....	8
РОЗДІЛ 1. АНАЛІЗ ДІЯЛЬНОСТІ ПНДЛ НУХТ ТА ОБҐРУНТУВАННЯ НЕОБХІДНОСТІ РОЗРОБКИ ІНФОРМАЦІЙНОЇ СИСТЕМИ	9
1.1. Загальна характеристика ПНДЛ НУХТ	9
1.2. Основні напрями діяльності та функціональні завдання ПНДЛ НУХТ	11
1.3. Аналіз проблем взаємодії ПНДЛ НУХТ із користувачами та зацікавленими сторонами	13
1.4. Постановка задачі та обґрунтування необхідності розробки веб-сайту.....	15
1.5. Аналіз існуючих аналогів інформаційних систем	17
1.6. Техніко-економічне обґрунтування впровадження інформаційної системи.....	19
РОЗДІЛ 2. ТЕХНІЧНЕ ЗАВДАННЯ НА СТВОРЕННЯ САЙТУ	23
2.1. Загальні положення.....	23
2.2. Мета, завдання та призначення системи.....	24
2.3. Об'єкт автоматизації.....	25
2.4. Основні вимоги до системи.....	26
2.5. Етапи розробки інформаційної системи	27
2.6. Контроль якості реалізації.....	28
2.7. Умови впровадження системи	29
2.8. Документація до системи	30
РОЗДІЛ 3. ПРОЄКТУВАННЯ, СТВОРЕННЯ ТА АПРОБАЦІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	32
3.1. Обґрунтування вибору програмно-технічних засобів розроблення програмного продукту	32
3.2. Проєктування та створення бази даних	35
3.3. Реалізація функціональних модулів системи	38
3.4. Інструкція користувача.....	50
3.5. Захист та безпека даних.....	56
3.6. Тестування інформаційної системи.....	57
3.7. Результати реалізації програмного продукту.....	61
ВИСНОВКИ.....	63
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	64

	7
ДОДАТКИ.....	65
Додаток А. Організаційна структура ПНДЛ НУХТ	67
Додаток Б. Моделі бізнес-процесів	68
Додаток В. Структура бази даних	69
Додаток Г. Фрагменти інтерфейсу вебсайту	70
Додаток Д. Програмний код.....	80

ВСТУП

У сучасних умовах цифрової трансформації та підвищення вимог до ефективності наукових досліджень, виникає необхідність впровадження інформаційних рішень, здатних забезпечити швидку та зручну взаємодію з користувачами. Проблемна науково-дослідна лабораторія Національного університету харчових технологій (ПНДЛ НУХТ), яка активно функціонує у сфері прикладної науки, на момент початку дослідження не мала інтегрованої веб-платформи для організації роботи із запитами, комунікації з користувачами та публічного представлення результатів.

Серед основних груп користувачів лабораторії — як внутрішні (викладачі, студенти), так і зовнішні (партнерські установи, представники бізнесу). Усі вони потребують доступу до інформаційних ресурсів у зручній електронній формі, без надмірного паперового документообігу. Таким чином, актуальним є створення веб-сайту, який поєднуватиме інформаційну функцію з інтерактивними можливостями: подання заявок на дослідження, ведення внутрішньої звітності, адміністрування замовлень та інше.

Метою кваліфікаційної роботи є проектування та створення веб-сайту для ПНДЛ НУХТ, що забезпечить автоматизацію ключових процесів — від приймання заявок до керування дослідницькою діяльністю та налагодження комунікації з користувачами.

Для досягнення поставленої мети передбачено виконання таких завдань:

- дослідити організаційну структуру та функціонування лабораторії;
- визначити основні проблеми цифрової взаємодії з користувачами;
- сформулювати технічне завдання на розробку інформаційної системи;
- розробити структуру бази даних та інтерфейс веб-застосунку;
- реалізувати функціональні модулі для користувача та адміністратора;
- провести тестування розробленої системи та оцінити її ефективність.

РОЗДІЛ 1. АНАЛІЗ ДІЯЛЬНОСТІ ПНДЛ НУХТ ТА ОБҐРУНТУВАННЯ НЕОБХІДНОСТІ РОЗРОБКИ ІНФОРМАЦІЙНОЇ СИСТЕМИ

1.1. Загальна характеристика ПНДЛ НУХТ

Проблемна науково-дослідна лабораторія (ПНДЛ) функціонує як структурний підрозділ Національного університету харчових технологій, створений з метою впровадження прикладних досліджень у сфері харчової промисловості, автоматизації технологічних процесів та суміжних галузей. Ініціатива створення ПНДЛ виникла у відповідь на зростаючу потребу закладу в практико-орієнтованих наукових розробках, тісно пов'язаних з навчальним процесом.

Головною метою діяльності ПНДЛ є підтримка та реалізація наукових ініціатив, що виникають у межах освітнього середовища університету. Лабораторія покликана забезпечити умови для проведення досліджень викладачами, аспірантами й студентами, а також сприяти трансформації результатів наукової діяльності в прикладні рішення, що мають практичне значення для української харчової промисловості.

Діяльність лабораторії охоплює такі напрямки:

- дослідження технологій переробки харчової сировини;
- розробка методів автоматизації та контролю процесів;
- застосування інформаційних систем у виробничій сфері;
- впровадження екологічно орієнтованих рішень;
- підтримка студентських ініціатив у рамках курсових, дипломних і дослідницьких проєктів.

ПНДЛ функціонує як майданчик для реалізації міждисциплінарних і внутрішньокафедральних ініціатив, підтримуючи зв'язок між теорією та практикою у професійній підготовці.

Організаційна структура ПНДЛ є динамічною та адаптивною. У штатному складі лабораторії працюють науково-педагогічні працівники, а також можуть залучатися фахівці з кафедр та факультетів університету. За потреби ректором

університету формуються тимчасові групи або сектори, що спеціалізуються на конкретних дослідницьких задачах.

ПНДЛ активно співпрацює з кафедрами та іншими підрозділами університету у форматі спільних проєктів, консультацій, проведення наукових експериментів і апробації нових навчальних курсів.

У центрі діяльності ПНДЛ перебуває інтеграція наукової та освітньої функцій. Студенти старших курсів, магістри та аспіранти залучаються до досліджень у рамках індивідуальних і групових проєктів. Лабораторія також підтримує викладачів, які розробляють нові наукові теми або оновлюють зміст освітніх компонентів.

Такий формат співпраці дозволяє формувати середовище, в якому результати досліджень стають елементом навчального процесу, а молоді науковці — активними учасниками інноваційних змін.

ПНДЛ виконує функцію внутрішньої наукової лабораторії при університеті, орієнтованої на підтримку наукових пошуків викладачів та молодих дослідників. Вона не діє як окремий інститут або центр, а функціонує у складі університету, забезпечуючи дослідницьке середовище для реалізації актуальних науково-технічних задач, що виникають на кафедрах.

У найближчій перспективі ПНДЛ планує:

- оновлення матеріально-технічної бази;
- розширення тематики досліджень у межах внутрішніх університетських ініціатив;
- підтримку міжкафедральної кооперації;
- посилення участі студентів у науковій роботі через створення лабораторних практикумів і хакатонів.

Таким чином, ПНДЛ зберігає свій фокус як навчально-науковий осередок, що об'єднує інженерну думку, освітній підхід і дослідницьку практику в межах НУХТ.

1.2. Основні напрями діяльності та функціональні завдання ПНДЛ НУХТ

Проблемна науково-дослідна лабораторія здійснює прикладні дослідження, пов'язані з актуальними завданнями харчової промисловості, зокрема в галузях технологій переробки сировини, автоматизації процесів, удосконалення засобів контролю якості та впровадження екологічно орієнтованих рішень. Особливу увагу приділяють технологічним інноваціям, які можуть бути безпосередньо використані на підприємствах харчової галузі України.

ПНДЛ є активним учасником освітнього процесу, виконуючи функції методичної платформи для практичної підготовки студентів. Лабораторія забезпечує базу для виконання курсових, дипломних і дисертаційних робіт, а також надає доступ до сучасного обладнання та програмного забезпечення, що дозволяє студентам реалізовувати інженерні ідеї на практиці.

Крім того, співробітники лабораторії беруть участь у розробці нових освітніх компонентів і науково-методичного забезпечення дисциплін технічного спрямування.

Однією з ключових функцій ПНДЛ є підтримка молодих дослідників, що включає:

- можливість участі в наукових дослідженнях у складі тимчасових груп;
- виконання індивідуальних експериментальних завдань;
- консультування під час підготовки до наукових конференцій;
- апробацію результатів досліджень у межах навчального процесу.

Студенти старших курсів, магістри та аспіранти активно залучаються до реалізації проєктів, що формує середовище професійного зростання і дозволяє закладати основи майбутніх наукових кар'єр.

Управління ПНДЛ здійснюється під керівництвом призначеного координатора, як правило — з числа науково-педагогічного складу університету.

Організаційна модель лабораторії побудована на принципах гнучкості, академічної автономії та міжкафедральної взаємодії.

ПНДЛ не функціонує як окремий науковий інститут, а тісно інтегрована в діяльність факультетів і кафедр, які формують тематику досліджень, надають персонал і використовують результати роботи лабораторії в освітньому процесі.

Структурна схема взаємодії між основними учасниками дослідницького процесу наведена на рисунку 1.1.



Рисунок 1.1 – Організаційна структура
Проблемної науково-дослідної лабораторії НУХТ

З метою реалізації конкретних завдань або проведення експериментальних досліджень у межах ПНДЛ можуть створюватися тимчасові наукові групи. Їх склад формується з числа викладачів, аспірантів і студентів, які мають відповідну фахову підготовку та зацікавленість у тематиці досліджень.

Такі групи мають змінний склад, обмежені строки реалізації проекту й чітко визначену мету. Це дозволяє лабораторії оперативно реагувати на внутрішні потреби кафедр та індивідуальні ініціативи, підтримуючи наукову активність у масштабах університету.

1.3. Аналіз проблем взаємодії ПНДЛ НУХТ із користувачами та зацікавленими сторонами

Діяльність Проблемної науково-дослідної лабораторії пов'язана з постійною взаємодією із внутрішніми та зовнішніми користувачами. До ключових категорій користувачів належать:

- студенти та аспіранти, які беруть участь у дослідницьких проєктах;
- викладачі та науково-педагогічні працівники, що ініціюють або координують проєкти;
- адміністративний персонал НУХТ, відповідальний за погодження та звітність;
- потенційні замовники досліджень з числа представників кафедр або факультетів.

На момент проведення аналізу більшість процесів взаємодії з користувачами здійснювалася вручну. Подання заявок на виконання наукових досліджень відбувається переважно через електронну пошту або паперові документи. Зберігання результатів, погодження тем та передача проміжних даних — усе це не має централізованої цифрової платформи.

Інформування про хід виконання досліджень, зміни статусу заявок або зв'язок між учасниками здійснюється переважно через особисту комунікацію або месенджери, що ускладнює документування та підвищує ризик втрати інформації.

У результаті аналізу було зафіксовано такі основні проблеми:

- відсутність єдиного цифрового середовища, що призводить до дублювання даних;
- низький рівень автоматизації при поданні та обробці заявок;
- ускладнення контролю за виконанням завдань і координації між учасниками;
- обмеженість звітності та відсутність централізованого архіву заявок і результатів;

- підвищене навантаження на персонал, зокрема на етапі обробки документів;
- неможливість аналітичного моніторингу — складно порахувати кількість заявок, тривалість, затримки.

Для візуалізації поточного стану інформаційної взаємодії було побудовано модель бізнес-процесів виконання заявки на проведення наукового дослідження (статус “AS IS”), наведену на рисунку 1.2. Модель демонструє, що всі основні етапи — від ініціювання заявки до погодження і архівації — виконуються у фрагментарному, неструктурованому форматі.

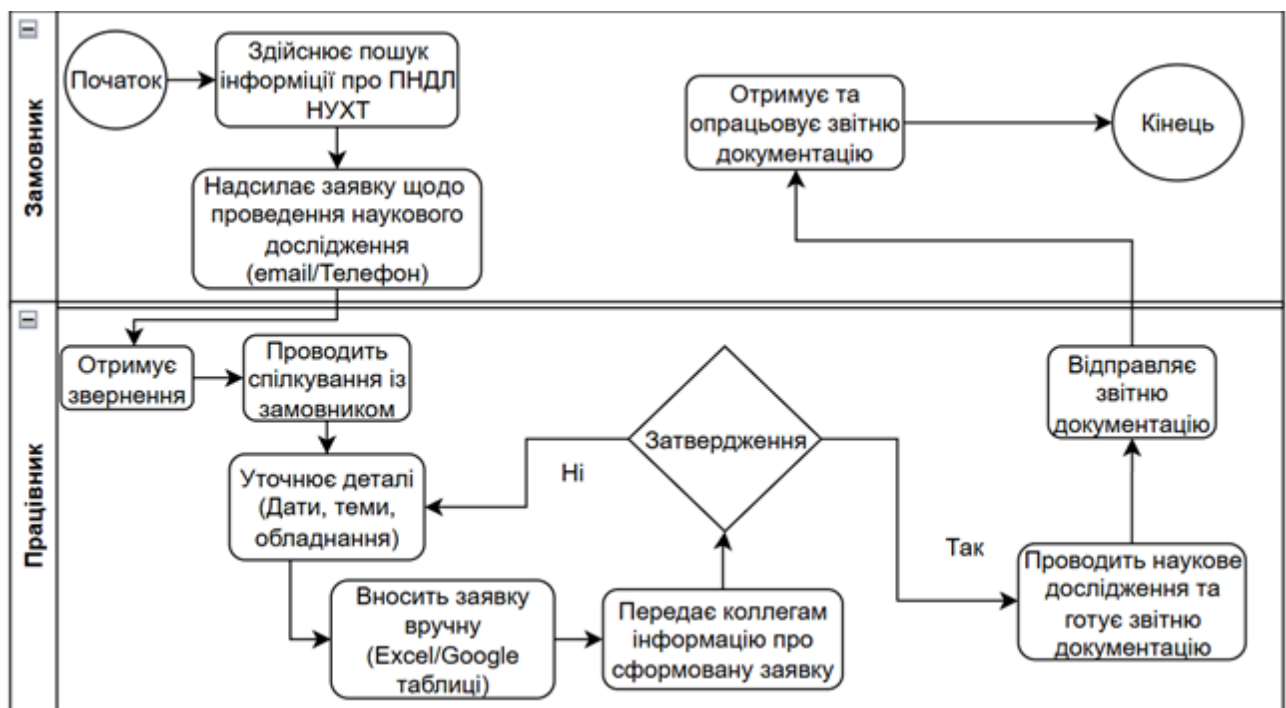


Рисунок 1.2 – Модель бізнес процесів виконання заявки в нотації BPMN статус “AS IS”

Аналіз чинного стану взаємодії свідчить про значну фрагментацію процесів та втрати ефективності на всіх етапах. Відсутність інформаційної системи ускладнює як внутрішню, так і зовнішню комунікацію лабораторії, а також гальмує аналітичну звітність, що є необхідною умовою сучасного управління дослідженнями.

У цьому контексті розробка єдиної інформаційної системи подання заявок, відстеження статусу, координації учасників і архівації результатів є критично важливою для підвищення прозорості, якості та продуктивності наукової

діяльності ПНДЛ. Модель бізнес-процесів виконання заявки в нотації BPMN статусу “ТО ВЕ” наведена на рисунку 1.3.

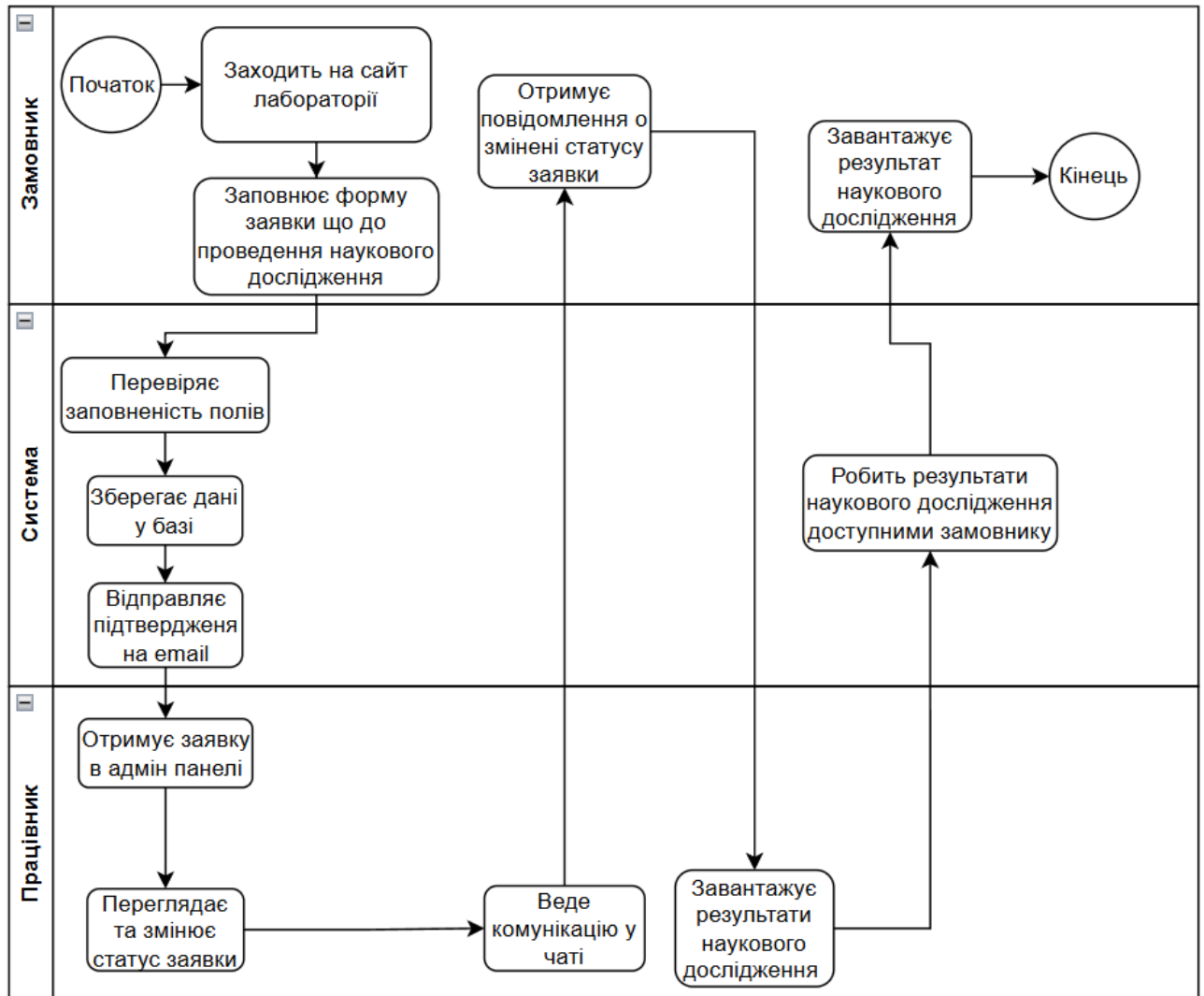


Рисунок 1.3 – Модель бізнес процесів виконання заявки в нотації BPMN статус “ТО ВЕ”

1.4. Постановка задачі та обґрунтування необхідності розробки веб-сайту

На основі проведеного аналізу функціонування Проблемної науково-дослідної лабораторії НУХТ, було встановлено, що ключові процеси, пов’язані з поданням, опрацюванням і супроводом заявок на дослідження, здійснюються вручну або з використанням загальних інструментів електронної пошти й офісного ПЗ. Такий формат взаємодії:

- не дозволяє централізовано контролювати документообіг,
- не забезпечує можливості аналітичної обробки даних,

- ускладнює координацію між працівниками, студентами та адміністрацією.

Крім того, зростання кількості заявок, підвищення ролі студентів у наукових розробках і загальна потреба в прозорості процесів — формують об'єктивну потребу у цифровій трансформації комунікаційної та адміністративної взаємодії лабораторії.

Метою створення інформаційної системи є розробка веб-орієнтованого середовища, яке дозволить:

- автоматизувати подання та обробку заявок на виконання наукових досліджень;
- забезпечити персоналізовану взаємодію між користувачами системи;
- підвищити оперативність прийняття рішень у межах ПНДЛ;
- централізувати зберігання, доступ і обробку інформації;
- зменшити адміністративне навантаження на працівників;
- покращити якість звітності та аналітичного моніторингу.

Таким чином, система має не лише розв'язувати прикладні задачі, а й стати цифровим інструментом підтримки наукової діяльності університету.

Інформаційна система повинна реалізовувати наступний мінімальний набір функціональних модулів:

- кабінет користувача: інтерфейс подання заявок, перегляд статусу, завантаження документів, зв'язок із координатором;
- кабінет працівника (адміністратора): обробка заявок, зміна статусу, комунікація з користувачем, створення звітів;
- форма подання заявки: структурована форма з фіксованими обов'язковими полями (контактна інформація, опис дослідження, терміни тощо);
- система повідомлень: автоматичне сповіщення про зміну статусу заявки;
- модуль архіву: зберігання попередніх заявок із можливістю пошуку;
- модуль звітності: фільтрація заявок за періодом, статусом, користувачем; експорт у табличний формат;
- модуль календаря: візуальне представлення запланованих та активних досліджень;

- механізм ролей і прав доступу: обмеження функцій залежно від типу користувача (користувач, працівник, адміністратор);

Запровадження системи дозволить досягти таких результатів:

- зменшення часу обробки однієї заявки (із 30 хв до 5–10 хв);
- зменшення кількості помилок, пов'язаних із людським фактором;
- зростання прозорості взаємодії між усіма учасниками процесу;
- оперативний доступ до архівних даних, включно з результатами попередніх досліджень;
- можливість аналітичного аналізу навантаження, динаміки заявок, термінів виконання;
- поліпшення цифрової репутації лабораторії як сучасного науково-освітнього підрозділу.

Запропоноване рішення прямо відповідає специфіці діяльності ПНДЛ:

- структура заявок адаптована до форми, яка використовується в ручному форматі;
- ролі системи базуються на реальних учасниках процесу (студент, координатор, керівник);
- вся архітектура проекту враховує обмеженість ресурсів та потребу у простому, але функціональному інтерфейсі;
- система має бути розгорнута на локальному сервері університету з перспективою подальшої інтеграції в загальну ІТ-інфраструктуру.

1.5. Аналіз існуючих аналогів інформаційних систем

Для формування уявлення про можливості автоматизації процесів подання заявок і супроводу досліджень було проаналізовано декілька доступних типових систем, які використовуються в закладах вищої освіти або дослідницьких організаціях.

Основними критеріями відбору стали:

- наявність веб-інтерфейсу для подання заявок;
- можливість відстеження статусу;
- функції адміністрування;

- відкритість або доступність системи для аналізу;
- наявність підтримки звітності або експорту.

Порівняльний аналіз систем наведено у таблиці 1.1, в якій розглянуто найбільш уживані інструменти (Google Forms, HelpDesk, Moodle, WordPress CMS, OpenConf, ін.), що використовуються в навчальному середовищі та для автоматизації заявок.

Таблиця 1.1. Аналіз аналогів інформаційних систем

№	Назва системи	Сфера використання	Основні функції	Недоліки в контексті ПНДЛ
1	Google Forms + Sheets	Освіта, опитування	Прийом форм, збереження у таблицях	Відсутність авторизації, не персоналізовано
2	Наукова CRM “Сфера”	Деякі ЗВО України	Реєстрація заявок, статуси, документообіг	Недоступна для кастомізації, складна інфраструктура
3	OpenConf	Конференції, подання тез	Подання заявок, перегляд рецензій, статусів	Орієнтовано на разові події, не університетське середовище
4	HelpDesk-системи	Бізнес, техпідтримка	Заявки, статуси, повідомлення	Немає наукового контексту, складна адаптація
5	Moodle + модулі	Навчальні платформи	Подання файлів, перевірка, коментарі	Не призначено для адміністрування наукових заявок
6	Розроблені CMS на WordPress	Різні сфери	Базова логіка взаємодії користувачів і форм	Потребує багато доопрацювань, нестабільна логіка прав доступу

Аналіз дозволив виявити сильні й слабкі сторони кожної системи в контексті вимог ПНДЛ.

Аналіз засвідчив, що наявні рішення:

- не охоплюють повністю весь цикл роботи ПНДЛ (від заявки до архіву),
- є занадто складними для впровадження в умовах навчального закладу.

Зокрема, системи HelpDesk і OpenConf побудовані для зовсім інших сценаріїв (техпідтримка або наукові конференції), а Google Forms взагалі не забезпечує ніякого контролю доступу й не гарантує безпеки даних.

Moodle та подібні освітні платформи хоча і мають інструменти подання інформації, проте не адаптовані для адміністративної моделі ПНДЛ, яка вимагає:

- багаторівневої взаємодії,
- фіксації статусів,
- звітності.

Беручи до уваги особливості організаційної структури, специфіку наукових досліджень і потребу в адаптації до внутрішніх процесів, створення власної інформаційної системи виглядає єдиним доцільним варіантом.

Таке рішення дозволяє:

- реалізувати лише потрібний функціонал без перевантаження;
- забезпечити простоту в адмініструванні та підтримці;
- уникнути залежності від зовнішніх платформ;
- адаптувати інтерфейси під реальні процеси роботи ПНДЛ.

1.6. Техніко-економічне обґрунтування впровадження інформаційної системи

Розрахунок економічного ефекту впровадження інформаційної системи здійснюється з урахуванням поточних умов діяльності Проблемної науково-дослідної лабораторії, обмеженості бюджетного фінансування та потреби в оптимізації витрат людських і часових ресурсів.

Через обмежений доступ до первинної фінансової звітності, для розрахунків були використані усереднені показники, характерні для закладів вищої освіти України: середня вартість години праці адміністративного працівника — 120 грн, середня кількість заявок на місяць — 100 одиниць.

До впровадження системи:

один цикл обробки заявки: ~30 хв (0,5 год)

$100 \text{ заявок} * 0,5 \text{ год} = 50 \text{ год/міс}$

$50 * 120 \text{ грн} = 6\,000 \text{ грн/міс}$

Після впровадження системи:

автоматизовані процеси: подання, повідомлення, архівація, статуси

середній час обробки однієї заявки: ~5 хв (0,083 год)

$$100 \text{ заявок} * 0,083 \text{ год} = 8,3 \text{ год/міс}$$

$$8,3 * 120 \text{ грн} \approx 1\,000 \text{ грн/міс}$$

$$\text{Економія: } 5\,000 \text{ грн/міс або } 60\,000 \text{ грн/рік}$$

Вартість розробки: інформаційну систему було створено в рамках бакалаврської роботи. Проте в умовах ринку орієнтовна вартість аналогічного проекту складає 2 000 грн, що включає:

- проектування БД;
- розробку бекенду;
- UI/UX інтерфейс;
- тестування;
- налаштування безпеки;
- підключення поштового сервісу.

Формула розрахунку терміну окупності:

$$T_{\text{окупн.}} = \frac{\text{Вартість розробки}}{\text{Місячна економія}} = \frac{25\,000}{5\,000} = 5 \text{ місяців}$$

Тобто проєкт повністю окупається вже через 5 місяців експлуатації, далі генерує чистий економічний ефект.

Додатково:

розрахунок економічної ефективності

Для оцінки рентабельності інвестицій застосовано показник ROI (Return on Investment) — коефіцієнт повернення вкладень:

$$\text{ROI} = (\text{Економічний ефект} - \text{Вартість впровадження}) / \text{Вартість впровадження} * 100\%$$

$$\text{ROI} = (60\,000 - 25\,000) / 25\,000 * 100\% = 140\%$$

Таким чином, інвестиції окупаються з надлишком, а прибутковість перевищує 140% за перший рік, що свідчить про високу економічну ефективність рішення.

Крім того, розраховано коефіцієнт економічної ефективності (КЕЕ):

$$\text{КЕЕ} = \text{Ефект} / \text{Витрати} = 60\,000 / 25\,000 = 2,4$$

Значення КЕЕ > 1 підтверджує доцільність впровадження, а 2,4 — свідчить про високу віддачу на кожен гривню інвестиції.

Окрім безпосередньої фінансової економії, впровадження системи створює низку соціальних і організаційних переваг, які позитивно впливають на функціонування ПНДЛ:

- зниження навантаження на співробітників — завдяки автоматизації рутинних операцій (до -80% часу на одну заявку);
- підвищення якості обслуговування — швидкий зворотний зв'язок, інформування користувачів;
- покращення умов праці — зменшення стресу, відсутність паперових черг, структурована робота;
- зростання професійної залученості — працівники можуть сфокусуватись на експертизі, а не на паперовій бюрократії;
- формування сучасної цифрової культури в межах лабораторії.

Ці ефекти не мають прямого грошового еквіваленту, проте істотно впливають на загальний рівень продуктивності.

Окрім прямої економії витрат на обробку заявок, система створює низку додаткових переваг, що позитивно впливають на загальну ефективність діяльності лабораторії. У таблиці 1.2 наведено ключові з них:

Таблиця 1.2. Додаткові переваги впровадженої системи

№	Перевага	Короткий опис
1	Централізація даних	Вся інформація про заявки, користувачів і результати зберігається в єдиній БД
2	Прозора статистика	Можливість фільтрації, аналізу, формування звітів за критеріями
3	Масштабованість під майбутні потреби	Структура системи дозволяє розширення функціоналу без зміни архітектури
4	Простота адаптації до інших підрозділів	Можливість використання системи в інших лабораторіях або кафедрах

№	Перевага	Короткий опис
5	Підвищення відповідальності користувачів	Персональні кабінети фіксують дії кожного користувача

Проведений аналіз підтверджує економічну доцільність, швидку окупність і позитивний соціальний вплив від впровадження вебсистеми. Інвестиції в її розробку, навіть у комерційних умовах, повністю себе виправдовують за кілька місяців. А подальша експлуатація дозволяє зменшити навантаження на персонал і підвищити якість роботи ПНДЛ без додаткових ресурсів.

РОЗДІЛ 2. ТЕХНІЧНЕ ЗАВДАННЯ НА СТВОРЕННЯ САЙТУ

2.1. Загальні положення

Дане технічне завдання визначає загальні вимоги до функціональності, структури, умов реалізації та впровадження інформаційної системи *NaukaLab*. Документ слугує основою для всіх етапів проєктування, розробки, тестування й оцінки готового продукту. Його метою є уніфікація очікувань між замовником та розробником, а також фіксація ключових технічних параметрів майбутньої системи.

Повна назва розроблюваної системи — інформаційна система “NaukaLab”. Вона реалізується у вигляді вебзастосунку з клієнт-серверною архітектурою, модульною побудовою та централізованою базою даних. Система включає багаторівневу систему доступу, що забезпечує персоналізовану взаємодію з користувачами та можливість адміністрування внутрішніх процесів.

Замовник: проблемна науково-дослідна лабораторія, структурний підрозділ, діяльність якого пов’язана з організацією й супроводом наукових досліджень, що потребують цифровізації документообігу та оптимізації взаємодії з клієнтами.

Розробник: здобувач вищої освіти — студент 4 курсу, спеціальність 122 «Комп’ютерні науки», група КН-4-2.

Розробка здійснюється в рамках освітнього процесу, без залучення зовнішнього фінансування. Проєкт реалізується як навчальне індивідуальне завдання на базі локального середовища розробки з відкритими інструментами (PHP, MySQL, HTML, CSS, JavaScript, Bootstrap, phpMyAdmin). Тестування та розгортання виконуються у межах локального серверного середовища типу XAMPP або еквівалентного.

Процес реалізації охоплює дев’ять ключових етапів, що виконуються послідовно у визначений календарний період. Поетапний розподіл робіт представлено на діаграмі Ганта (Рис.. 2.1), яка слугує візуалізацією часових меж проєкту.

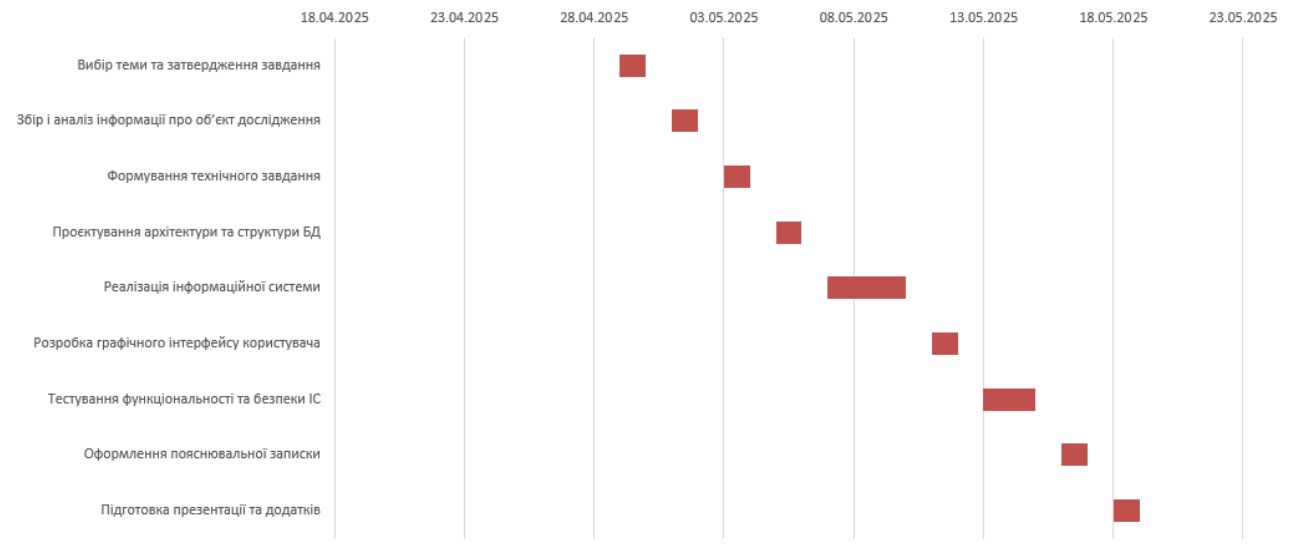


Рисунок 2.1 – Діаграма Ганта

У разі виявлення нових вимог з боку замовника або появи технічних обмежень, допускається коригування даного технічного завдання за взаємною згодою сторін. Усі зміни фіксуються документально та враховуються в подальшому процесі розробки.

2.2. Мета, завдання та призначення системи

Інформаційна система *NaukaLab* призначена для автоматизації ключових процесів, пов'язаних із поданням, обробкою та супроводом заявок на проведення наукових досліджень у межах діяльності лабораторії. Система виступає єдиним цифровим інструментом для взаємодії між користувачами (замовниками досліджень) та співробітниками, забезпечуючи фіксацію, збереження, контроль і архівацію даних у структурованому форматі.

Рішення реалізується як веб-застосунок з багаторівневою системою авторизації, що дозволяє:

- створювати та редагувати заявки;
- переглядати статус їх обробки;
- організовувати внутрішню комунікацію між сторонами;
- формувати аналітичні звіти про виконану роботу.

Метою розробки інформаційної системи *NaukaLab* є створення інструменту, що забезпечує реалізацію таких функціональних цілей:

- створення веб-інтерфейсу для подання та редагування заявок на дослідження;
- впровадження авторизації з розподілом прав доступу за ролями (гість, користувач, адміністратор);
- забезпечення персоналізованого доступу до інформації через особисті кабінети;
- зберігання заявок та супровідної інформації у структурованій базі даних;
- реалізація системи комунікації між користувачем та працівником лабораторії;
- створення модуля аналітики для перегляду звітів та формування статистики;
- забезпечення адаптивності інтерфейсу для мобільних та десктопних пристроїв;
- підтримка архівації завершених заявок і можливість їх пошуку за фільтрами.

Ці технічні цілі визначають функціональний каркас майбутньої системи та слугують основою для формування вимог до архітектури, бази даних і логіки взаємодії користувачів із системою.

2.3. Об'єкт автоматизації

Об'єктом автоматизації є процеси обробки заявок на проведення наукових досліджень у Проблемній науково-дослідній лабораторії. До складу об'єкта входять такі компоненти:

- прийом заявок від замовників;
- зберігання даних про дослідження;
- супровід комунікації між користувачами та працівниками лабораторії;
- зміна статусу заявок на різних етапах виконання;
- формування звітності та перегляд архіву виконаних досліджень;

- адміністрування користувачів та керування доступом до функціоналу системи.

Автоматизація зазначених процесів дозволяє забезпечити централізований облік, підвищити ефективність взаємодії між учасниками, зменшити ризики втрати або дублювання інформації та спростити загальне управління виконанням дослідницьких запитів.

2.4. Основні вимоги до системи

Система має відповідати сучасним вимогам до інформаційної безпеки, доступності й сумісності. До ключових вимог належать:

- цілодобовий доступ до функціоналу через веб-інтерфейс;
- використання захищеного з'єднання за протоколом HTTPS;
- сумісність з мобільними пристроями та десктопними браузерами;
- інтерфейс, реалізований українською мовою;
- підтримка системи автентифікації й авторизації з розмежуванням ролей.

У системі реалізовано низку модулів, що відповідають за конкретні групи функціональності. Їх короткий опис наведено у таблиці 2.1.

Таблиця 2.1. Функціональні модулі інформаційної системи

№	Модуль	Функціональність
1	Авторизація	Вхід до системи, контроль доступу за ролями
2	Інфо-сторінки	Надання довідкової інформації про діяльність ПНДЛ
3	Замовлення	Створення, редагування та відправлення заявок
4	Архів	Пошук та перегляд завершених досліджень
5	Комунікація	Чат між користувачами та співробітниками лабораторії
6	Кабінет	Персоналізоване управління файлами та заявками

Система передбачає розмежування прав доступу відповідно до типу користувача, що описано у таблиці 2.2.

Таблиця 2.2. Типи користувачів та доступ до функцій

Роль	Опис	Доступ
Гість	Незареєстрований користувач	Перегляд інформаційних сторінок
Користувач	Зареєстрований клієнт	Повний доступ до кабінету та подання заявок
Адміністратор	Співробітник ПНДЛ	Повний доступ до управління системою

2.5. Етапи розробки інформаційної системи

Процес створення інформаційної системи *NaukaLab* складався з послідовних етапів, кожен з яких відіграє ключову роль у забезпеченні повного циклу розробки: від формування вимог до тестування та оформлення документації. Календарне планування виконання етапів наведено у діаграмі Ганта (Рис. 2.1).

На першому етапі проводився збір та аналіз інформації про діяльність лабораторії, її завдання, типові сценарії взаємодії з клієнтами. Визначалися ключові проблеми, що підлягають вирішенню, та формулювалися функціональні очікування користувачів. Результатом цього етапу стала специфікація вимог до системи.

Було розроблено концептуальну модель бази даних, що включає основні сутності (користувачі, заявки, повідомлення, документи) та зв'язки між ними. Після цього виконано логічне і фізичне проєктування з урахуванням нормалізації, уніфікації типів даних та забезпечення цілісності. Структура моделі оформлена у вигляді ER-діаграм.

Функціональну логіку реалізовано засобами мови PHP із використанням засобів обробки запитів, взаємодії з базою даних через PDO, а також реалізації сценаріїв авторизації, обробки форм та обліку даних. У якості середовища розробки застосовано локальний сервер XAMPP, який забезпечує роботу PHP-інтерпретатора та СУБД MySQL.

Інтерфейс користувача реалізовано за допомогою HTML5, CSS3 та JavaScript, з використанням фреймворку Bootstrap для створення адаптивного дизайну. Реалізовано навігацію між сторінками, інтуїтивну взаємодію з формами, валідацію введених даних та відображення повідомлень про результат операцій.

Після завершення модульної розробки здійснено з'єднання клієнтської та серверної частин, з налагодженням логіки запит–відповідь, перевіркою коректності взаємодії з базою даних, а також перевіркою сценаріїв переходу між ролями користувачів.

Виконано повний цикл перевірки працездатності ключових модулів системи, включаючи:

- модульне тестування окремих функцій;
- інтеграційне тестування взаємодії модулів;
- навантажувальне тестування з симуляцією багатьох одночасних запитів;
- функціональне тестування користувацьких сценаріїв.

Під час завершального етапу було підготовлено пояснювальну записку, розроблено презентаційні матеріали та інструкції користувача. Проведено внутрішню перевірку відповідності реалізованого функціоналу до технічного завдання. У разі виявлення розбіжностей — внесено відповідні коригування.

2.6. Контроль якості реалізації

Для забезпечення відповідності реалізованої інформаційної системи *NaukaLab* встановленим технічним і функціональним вимогам, у рамках проєкту передбачено проведення комплексного контролю якості. Перевірка має охоплювати окремі модулі, взаємодію між компонентами, відповідність структурі даних, а також повну функціональність із позиції користувача.

Кожен функціональний компонент системи повинен перевірятись в ізольованому середовищі. Тестуванню підлягають:

- модуль реєстрації та автентифікації;
- створення, редагування та перегляд заявок;

- обробка повідомлень у чаті;
- перегляд архіву та завантаження результатів.

Ціль — виявити логічні помилки на рівні окремих елементів програмного коду.

Після перевірки модулів повинна бути забезпечена перевірка їхньої взаємодії:

- синхронізація клієнтської та серверної частин;
- правильність оновлення бази даних;
- передача параметрів між формами, запитами та відповідями;
- обробка виключних ситуацій.

Система підлягає перевірці на предмет повної реалізації передбаченого функціоналу. До ключових критеріїв належать:

- відповідність структур бази даних логічній моделі;
- реалізація всіх описаних функціональних модулів;
- підтримка розмежування прав доступу;
- відповідність інтерфейсу вимогам до адаптивності.

З метою підтвердження виконання функціональних вимог, повинна бути проведена демонстрація роботи системи за ключовими сценаріями:

подання та обробка заявки;

зміна статусу дослідження;

передача повідомлень між користувачем і адміністратором;

архівування завершених досліджень.

2.7. Умови впровадження системи

Для коректного розгортання та експлуатації інформаційної системи *NaukaLab* повинно бути забезпечено відповідне технічне середовище, яке гарантує її стабільну роботу, інформаційну безпеку та доступність:

- наявність вебсервера з підтримкою php 7.4 або вище;
- підтримка СУБД MySQL 5.7 або сумісної версії;

- платформа має забезпечувати коректну обробку запитів post/get, сесій та підключення до бази даних через pdo;
- рекомендовані середовища для запуску: apache+mysql на базі локального або хмарного хостингу (ХАМРР, LAMP, LEMP тощо);
- система повинна бути розгорнута з підтримкою https-з'єднання, що забезпечує шифрування трафіку;
- необхідна наявність ssl-сертифіката, виданого сертифікаційним центром або самопідписаного у разі тестового впровадження;
- сервер повинен мати механізми захисту від sql-ін'єкцій, xss-атак та несанкціонованого доступу;
- повинна бути налаштована регулярна архівація бази даних;
- файли резервних копій повинні зберігатись у захищеному каталозі, з обмеженим доступом;
- рекомендовано зберігати принаймні 3 останні резервні копії із можливістю автоматичного відновлення;
- повинен бути забезпечений доступ до інтерфейсу адміністратора через обліковий запис з правами доступу;
- реалізація журналу подій для фіксації ключових операцій (вхід, зміна статусу заявки, оновлення даних);
- забезпечення можливості моніторингу навантаження, журналів помилок і активності користувачів.

2.8. Документація до системи

Інформаційна система *NaukaLab* повинна супроводжуватись комплектом технічної документації, що забезпечує можливість розгортання, експлуатації та обслуговування програмного забезпечення відповідно до функціонального призначення.

Містить стислий опис структури системи, вказівки щодо налаштування локального або віддаленого середовища, а також інструкції для початкового запуску. Повинен включати:

- перелік необхідного ПЗ та версій;
- шаблон конфігураційного файлу;
- порядок ініціалізації бази даних.

Має описувати покрокову взаємодію з веб-інтерфейсом з позиції кінцевого користувача:

- реєстрація облікового запису;
- створення, перегляд та редагування заявок;
- перегляд статусу виконання;
- отримання результатів досліджень.
- включає технічні аспекти адміністрування системи:
- управління обліковими записами користувачів;
- перегляд, зміна та видалення заявок;
- публікація новин або повідомлень;
- завантаження готових результатів та контроль за архівом.

Якщо система передбачає використання внутрішніх або зовнішніх API, повинна бути створена окрема специфікація методів із зазначенням: HTTP-методів (GET, POST тощо); параметрів запити; форматів відповіді; прикладів виклику та обробки помилок.

Таким чином, наявність повного комплексу документації є обов'язковою умовою для ефективного використання, масштабування та супроводу системи *NaukaLab* в експлуатаційному середовищі.

РОЗДІЛ 3. ПРОЄКТУВАННЯ, СТВОРЕННЯ ТА АПРОБАЦІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

3.1. Обґрунтування вибору програмно-технічних засобів розроблення програмного продукту

У процесі проєктування та реалізації інформаційної системи було визначено низку критеріїв, що стали основою для вибору технологічного стеку.

Основними з них стали:

- сумісність компонентів – здатність технологій ефективно взаємодіяти між собою;
- підтримка спільноти – наявність активної бази знань, документації та форумів;
- вільна ліцензія – використання безкоштовного або відкритого програмного забезпечення;
- простота розгортання – можливість налаштування в умовах локального середовища без потреби в складних конфігураціях;
- безпека та масштабованість – здатність системи адаптуватись до зміни навантаження та захист даних;
- сумісність з сучасними браузером та пристроями – кросплатформеність інтерфейсу.

На етапі відбору було проведено порівняння найбільш поширених технологій для веб-розробки. Наведений у таблиці 3.1 аналіз можливих варіантів демонструє ключові параметри:

Таблиця 3.1. Аналіз можливих варіантів

Технологія	Переваги	Недоліки
PHP	Простий синтаксис, велика кількість готових бібліотек, сумісність з Apache/MySQL	Старіші версії мають проблеми з безпекою
Python (Django)	Сучасний підхід, ORM, інтеграція з ML	Вища складність налаштування для новачків
Node.js	Асинхронність, одна мова на фронті і бекенді	Потребує більше налаштування безпеки
MySQL	Легкість, простота інтеграції з PHP, ACID-сумісність	Обмежені можливості у складних запитах
PostgreSQL	Висока продуктивність, підтримка JSON та розширень	Вища складність для початківців

У результаті аналізу було обрано стек, що найкраще поєднує швидкість розробки, безпеку та простоту у використанні.

Мова програмування: PHP 8.1 обрана за свою стабільність, підтримку з боку спільноти та відповідність вимогам до серверної обробки даних. Завдяки інтеграції з MySQL забезпечується швидка розробка без складних конфігурацій.

СУБД: MySQL 8.0: реляційна СУБД із підтримкою зовнішніх ключів, транзакцій і масштабування. Вона чудово поєднується з PHP, забезпечуючи швидку обробку запитів та цілісність даних.

Фреймворк інтерфейсу: Bootstrap 5. Адаптивна верстка, кросбраузерність і велика бібліотека компонентів дозволили створити сучасний, зручний інтерфейс без необхідності глибокого занурення у CSS.

Фронтенд-технології: HTML5, CSS3, JavaScript. Забезпечують стандартну структурування та візуалізацію вмісту сторінок, інтерактивність форм і клієнтську валідацію.

Для локальної розробки було обрано XAMPP — універсальний стек, що містить:

- Apache — HTTP-сервер для обробки веб-запитів;
- PHP — інтерпретатор для обробки серверного коду;
- MySQL — сервер баз даних.
- XAMPP забезпечив миттєвий запуск сервера, без потреби в додатковому налаштуванні. Водночас, логіка системи розроблена таким чином, що допускає перенесення на хмарний сервер (наприклад, з підтримкою HTTPS та резервного копіювання).

Приклад робочого середовища наведено на рисунку 3.1.

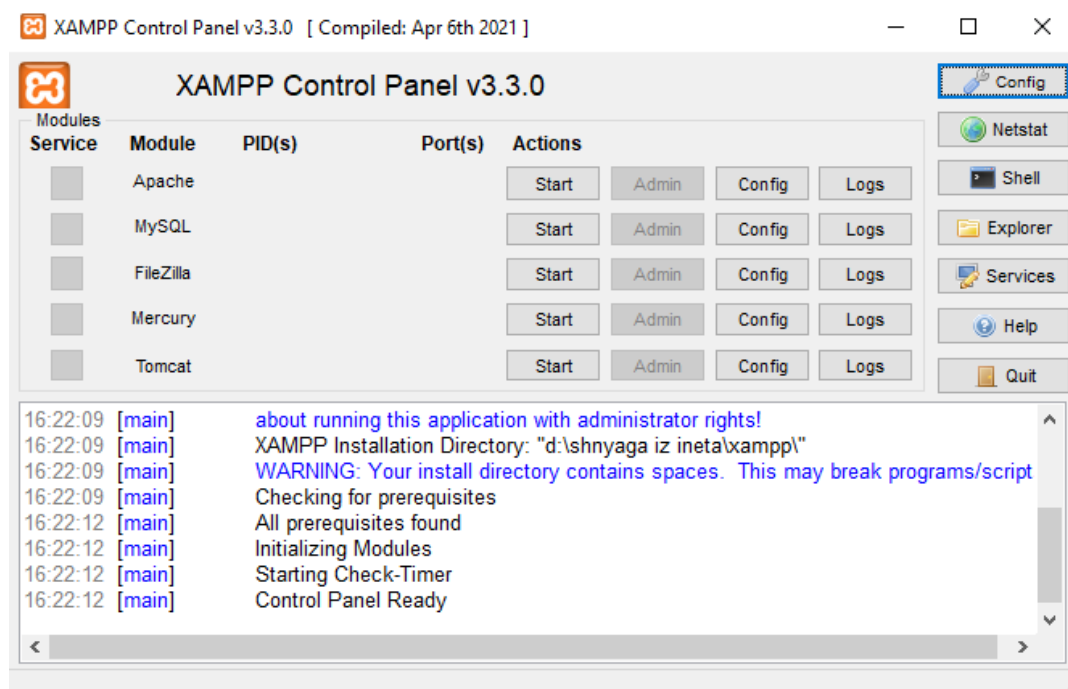


Рисунок 3.1 – Приклад робочого середовища

phpMyAdmin – інтерфейсна оболонка для управління базою даних, яка дозволяє зручно перевіряти, оновлювати та експортувати дані.

PHPMailer – бібліотека для безпечної та стабільної реалізації відправлення email-листів з підтвердженням реєстрації.

Обраний набір інструментів дозволив:

- мінімізувати час розробки завдяки простоті налаштування;
- забезпечити безпеку за рахунок підтримки сучасних засобів захисту (PDO, password_hash, SSL);
- гарантувати адаптивність інтерфейсу під мобільні пристрої;

- залишити простір для масштабування, у тому числі розгортання в хмарному середовищі, інтеграції з іншими API та підтримки нових мов інтерфейсу.

Таким чином, технологічна база системи є гнучкою, надійною і відповідає як навчальним, так і реальним прикладним потребам.

3.2. Проектування та створення бази даних

База даних інформаційної системи розроблена у відповідності до реляційної моделі та принципів нормалізації, що дозволило досягти логічної структурованості, відсутності надмірності та високої зв'язаності між сутностями. Усі таблиці пов'язані зовнішніми ключами, що забезпечує узгодженість інформації при обробці запитів, зміні чи видаленні записів. Кожна таблиця відповідає окремому функціональному модулю системи.

Таблиця «Users» зберігає всі зареєстровані облікові записи користувачів та співробітників лабораторії.

Структура таблиці «Users»:

- `client_id` – первинний ключ, унікальний ідентифікатор користувача;
- `fullname` – повне ім'я користувача, відображається в інтерфейсі;
- `email` – унікальна адреса електронної пошти, використовується для входу;
- `password` – хешований пароль, захищений алгоритмом `password_hash()`;
- `role` – роль користувача (`user` або `worker`), що визначає рівень доступу;
- `created_at` – дата створення облікового запису;
- `is_verified` – булеве поле, яке вказує, чи підтверджено email;
- `verify_token` – токен, що генерується для підтвердження реєстрації.

Використовується для автентифікації, авторизації та фільтрації доступу у модулях системи.

Таблиця «research_requests» містить усі запити на проведення наукових досліджень.

Структура таблиці «research_requests»:

- research_id – первинний ключ;
- client_ref – зовнішній ключ на users.client_id;
- email, client_origin, organization_name, position, full_name – детальна інформація про замовника;
- request_date, due_date – дата подачі заявки та бажаний дедлайн;
- grant_info, sample_count, apparatus_choice, sample_description, comments – дані щодо дослідження;
- created_at – мітка створення;
- status – поточний стан виконання;
- file_path – шлях до прикріплених матеріалів.

Таблиця «documents» призначена для зберігання інформації про наукове обладнання лабораторії.

Структура таблиці «documents»:

- document_id – унікальний ідентифікатор;
- title – назва документа;
- description – короткий опис;
- file_path – фізичне розташування файлу на сервері;
- created_at – дата завантаження.

Таблиця «messeges» реалізує систему обміну повідомленнями між адміністратором та користувачами.

Структура таблиці «messeges»:

- message_id – первинний ключ;
- sender_ref, receiver_ref – зовнішні ключі на users.client_id;
- message_text – вміст повідомлення;
- created_at – час відправлення.

Таблиця «messegas» використовується у двосторонньому чаті, дозволяє вести листування між клієнтом та підтримкою.

Таблиця «calendar_blacklist» призначена для блокування дат для замовлення наукових робіт (через наприклад, свята, ремонти, перевантаження обладнання).

Структура таблиці «calendar_blacklist»:

- calendar_entry_id – ідентифікатор запису;
- date – заблокована дата;
- reason – причина блокування (наприклад, "технічне обслуговування").

Функціонал блокування дат застосовується у календарі адміністратора при поданні нових заявок.

Таблиця «news» використовується інформування користувачів про події в лабораторії.

Структура таблиці «news»:

- news_entry_id / announcement_id – унікальні ідентифікатори;
- title, content – назва та текст новини/анонсу;
- type – тип (news або announcement);
- created_at – дата публікації.

Новини виводяться на головній сторінці у спеціальному розділі.

Таблиця «orders» може використовуватись як додатковий механізм для внутрішнього обліку.

Структура таблиці «orders»:

- order_ref – ідентифікатор;
- client_ref – користувач;
- title, details, status, created_at, due_date – деталі замовлення.

Може бути інтегрована з зовнішніми CRM або API.

Таблиця «achievements» зберігає список доступного обладнання.

Структура таблиці «achievements»:

- achievement_id – ідентифікатор;

- name, description – назва та опис;
- created_at – дата додавання;
- file_path – зображення або технічний документ.

На рисунку 3.2 наведено структуру бази даних. Вона демонструє зв'язки типу один-до-багатьох між користувачами та їх заявками, документами, повідомленнями. Зовнішні ключі дозволяють зберігати референційну цілісність при виконанні будь-яких CRUD-операцій.

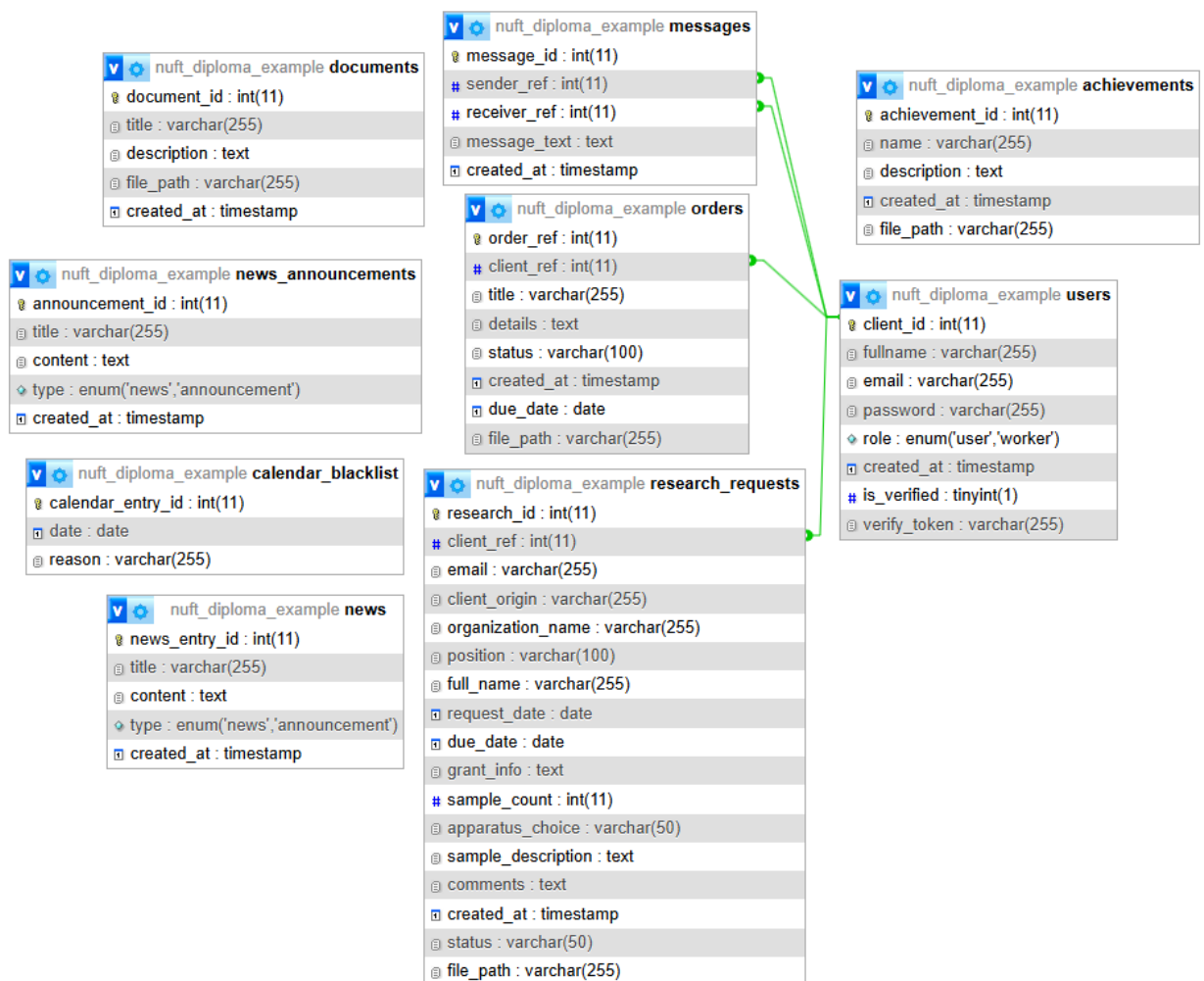


Рисунок 3.2 – ER-діаграма бази даних системи NaukaLab

3.3. Реалізація функціональних модулів системи

Інформаційна система побудована за модульним принципом, що передбачає відокремлення кожного функціонального компонента в окремий

логічний блок. Це дозволяє забезпечити підтримуваність, розширюваність і повторне використання коду. Усі модулі взаємодіють через спільну базу даних та реалізовані з використанням принципів MVC (Model–View–Controller) у спрощеній формі.

Інформаційна система реалізована за принципом клієнт-серверної архітектури з класичним розподілом на фронтенд, бекенд і базу даних. Веб-застосунок обробляє HTTP-запити через Apache-сервер, серверна логіка реалізована мовою PHP, а всі дані зберігаються у базі даних MySQL.

Компоненти архітектури:

- Клієнтська частина: HTML5 + CSS3 + JavaScript + Bootstrap 5;
- Серверна частина: PHP 8.1, логіка авторизації, обробки заявок, листування;
- Система управління базою даних: MySQL 8.0, використання зовнішніх ключів, транзакцій;
- Сервер середовища: XAMPP (локально), із подальшою можливістю хмарного розгортання.

Фрагмент коду PHP для підключення до бази даних:

```
$pdo = new PDO("mysql:host=localhost;dbname=lab_site", "root", "");
$pdo->setAttribute(PDO::ATTR_ERRMODE,
PDO::ERRMODE_EXCEPTION);
```

Сторінка register.php реалізує механізм створення нового облікового запису користувача з підтвердженням електронної пошти. Після заповнення форми дані перевіряються, хешуються і надсилається лист з підтвердженням через PHPMailer.

Процес:

- Заповнення форми;
- Валідація email та пароля;
- Генерація verify-токена;
- Надсилання листа;
- Активація облікового запису через verify.php.

Приклад фрагменту PHP-коду:

```

$hashedPassword = password_hash($password, PASSWORD_DEFAULT);
$token = bin2hex(random_bytes(32));
$stmt = $pdo->prepare("INSERT INTO users (fullname, email, password,
verify_token) VALUES (?, ?, ?, ?)");
$stmt->execute([$fullname, $email, $hashedPassword, $token]);

```

Головна
Обладнання
Заявка
Реєстрація
Вхід

[Головна](#) > [Реєстрація](#)

Реєстрація нового акаунту

ПІБ:

Email:

Пароль:

Підтвердіть пароль:

Вже маєте акаунт? [Увійти](#)

Адреса: м. Київ, вул. Володимирська, 68

Телефон: +38 (044) 123-45-67

Пошта: pndl@nuft.edu.ua

Рисунок 3.3 – Форма реєстрації нового користувача

Модуль авторизації (login.php) перевіряє дані для входу та відкриває відповідний кабінет згідно з роллю користувача. Реалізовано перевірку підтвердження пошти, хеш-паролів та сесійний захист.

Процес:

- Введення email і пароля;
- Перевірка через password_verify;
- Перевірка підтвердження пошти;

- Встановлення сесії;
- Редірект до відповідного профілю.

Приклад фрагмент коду PHP:

```
$stmt = $pdo->prepare("SELECT * FROM users WHERE email = ?");
$stmt->execute([$email]);
$user = $stmt->fetch();
if (password_verify($password, $user['password'])) {
    $_SESSION['user_id'] = $user['client_id'];
    $_SESSION['role'] = $user['role'];
}
```

Рисунок 3.4 – Сторінка входу до системи

Форма `submit-order-form.php` є однією з ключових у інформаційній системі — дозволяє користувачеві подати заявку з описом дослідження, обрати обладнання, зазначити дедлайн та додати коментарі.

Компоненти форми:

- Тип замовника (внутрішній/зовнішній);

- ПІБ, email, посада;
- Опис дослідження;
- Обране обладнання;
- Коментарі;
- Дата подачі та бажаний дедлайн.

Приклад фрагменту PHP-коду для збереження заявки:

```
$stmt = $pdo->prepare("INSERT INTO research_requests (client_ref, title, type, description, apparatus_choice, deadline, comment, created_at) VALUES (?, ?, ?, ?, ?, ?, ?, NOW())");
```

```
$stmt->execute([$userId, $title, $type, $desc, $equipment, $deadline, $comment]);
```

Рисунок форми реєстрації можна подивитись у додатку Б.

Сторінка `profile-user.php` призначена для взаємодії користувача з системою після авторизації. Вона реалізує функціонал перегляду поданих заявок, завантаження результатів та спілкування з представником лабораторії.

Функціональні елементи:

- таблиця із заявками (назва, дата, статус);
- кнопки перегляду деталей та скачування документів;
- вивід повідомлень за кожною заявкою.

Приклад фрагменту коду PHP:

```
$stmt = $pdo->prepare("SELECT * FROM research_requests WHERE client_ref = ?");
```

```
$stmt->execute($_SESSION['user_id']);
```

```
while ($row = $stmt->fetch()) {
    echo "<tr><td>{$row['title']}</td><td>{$row['status']}</td></tr>";
}
```

Головна Обладнання Заявка Особистий кабінет Вийти

Головна > Особистий кабінет

Вітаємо, Дмитро123!

Мої заявки на наукові дослідження

У вас поки що немає заявок. [Подати заявку](#)

Календар зайнятості лабораторії

● — ваші заявки, ✗ — зайнято іншими

< > today травень 2025 р.

нд	пн	вт	ср	чт	пт	сб
27	28	29	30	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	✗ Зайнято	✗ Зайнято
1	2	3	4	5	6	7

Чат з адміністрацією

Адреса: м. Київ, вул. Володимирська, 68
 Телефон: +38 (044) 123-45-67
 Пошта: pndl@nuff.edu.ua
 © 2025 ПНДЛ НУХТ. Всі права захищені.

Рисунок 3.5 – Інтерфейс кабінету користувача

Сторінка `profile-worker.php` реалізує функціонал адміністратора: перегляд, редагування, фільтрація та видалення заявок, а також чат і експорт.

Основні можливості:

- пошук за статусом та датою;
- кнопки “редагувати”, “видалити”, “додати файл”;
- таблиця з усіма заявками;
- вивід повідомлень та відповіді адміністратору.

Приклад фрагменту коду PHP:

php

Копировать Редактировать

```
$stmt = $pdo->prepare("UPDATE research_requests SET status = ? WHERE
research_id = ?");
```

```
$stmt->execute([$newStatus, $id]);
```

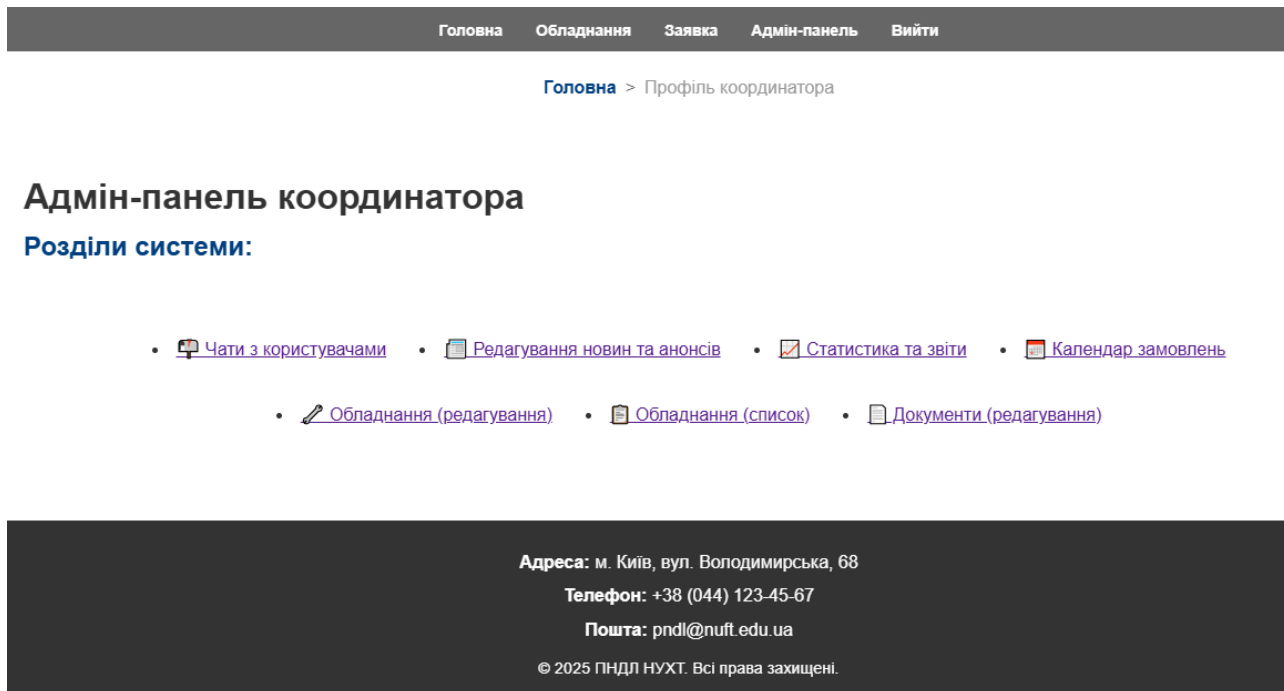


Рисунок 3.6 – Панель керування заявками для працівника

Форма реалізована на сторінці `calendar.php`, з інтеграцією бібліотеки FullCalendar для зручного відображення даних у вигляді календаря.

Можливості календаря:

- відображення поданих заявок;
- блокування нових дат;
- коментар до кожної заблокованої дати;
- маркування статусів кольорами.

Приклад фрагменту коду PHP для блокування дати:

```
$stmt = $pdo->prepare("INSERT INTO calendar_blacklist (date, reason) VALUES (?,
?);");
```

```
$stmt->execute([$date, $reason]);
```

31 Календар заявок на дослідження

Клікніть на дату для блокування або зняття блокування.

<
>
today
травень 2025 р.
month
week

нд	пн	вт	ср	чт	пт	сб
27	28	29	30	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
				Заблоковані		
25	26	27	28	29	30	31
				Заблоковані	#9: Сиротенко Д	#10: Маринін
1	2	3	4	5	6	7
						Заблоковані

Адреса: м. Київ, вул. Володимирська, 68
 Телефон: +38 (044) 123-45-67
 Пошта: pndl@nuft.edu.ua
 © 2025 ПНДЛ НУХТ. Всі права захищені.

Рисунок 3.7 – Інтерфейс календаря замовлень

Модуль чату реалізований як асинхронна переписка без AJAX, на основі форми з POST-запитом. Повідомлення додаються до таблиці messages, і виводяться за sender_ref та receiver_ref.

Ключові особливості:

- повідомлення містить текст, дату та ID автора;
- адміністратор бачить чат з усіма користувачами;

- кожна заявка має свій ID для фільтрації діалогу.

Приклад фрагменту коду PHP:

```
$stmt = $pdo->prepare("INSERT INTO messages (sender_ref, receiver_ref, message_text, created_at) VALUES (?, ?, ?, NOW())");
```

```
$stmt->execute([$sender, $receiver, $text]);
```

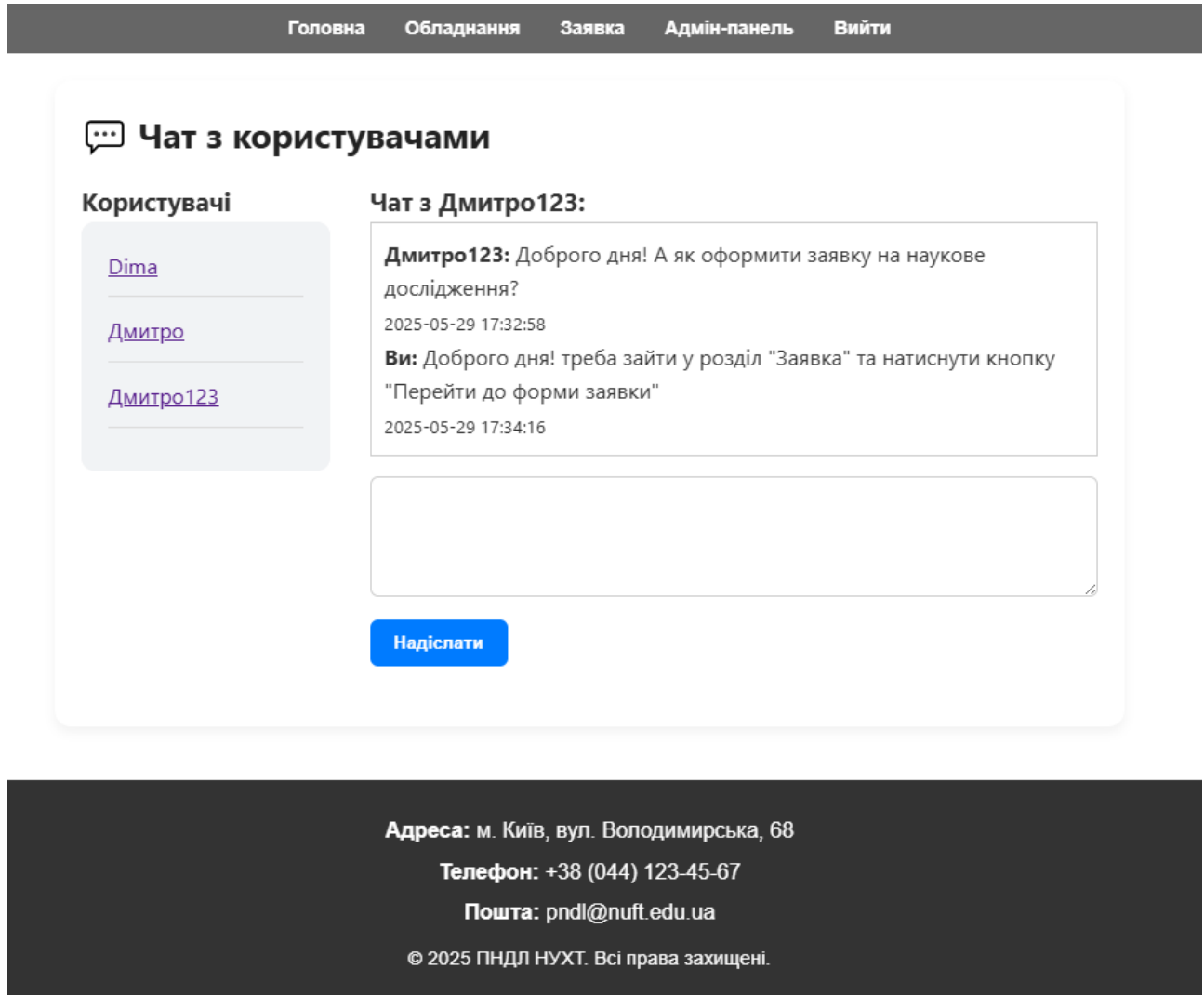


Рисунок 3.8 – Вікно чату між користувачем і працівником

Функціонал управління документами реалізовано на сторінках `documents.php` (для користувача) та `documents-admin.php` (для працівника). Він забезпечує перегляд, завантаження, додавання та видалення документів, пов'язаних із заявками.

Можливості:

- перегляд документів за заявками;

- завантаження результатів досліджень;
- додавання нових файлів через форму (PDF, DOCX, PNG тощо);
- видалення або оновлення файлу.

Приклад фрагменту PHP-коду завантаження файлу:

```
$filename = $_FILES['file']['name'];
```

```
$path = "uploads/" . basename($filename);
```

```
move_uploaded_file($_FILES['file']['tmp_name'], $path);
```

```
$stmt = $pdo->prepare("INSERT INTO documents (title, description, file_path,
created_at) VALUES (?, ?, ?, NOW())");
```

```
$stmt->execute([$title, $desc, $path]);
```

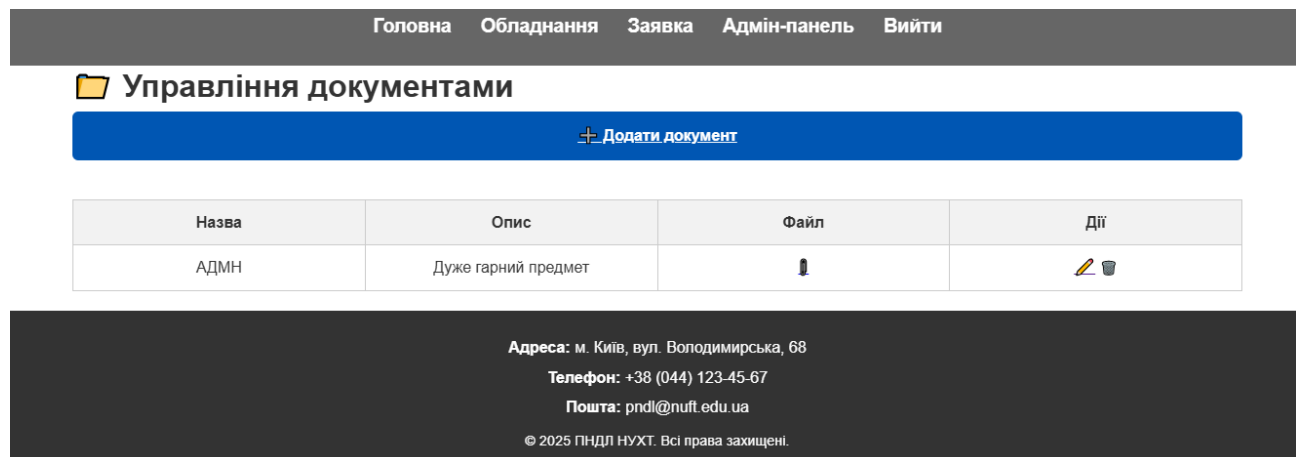


Рисунок 3.9 – Інтерфейс управління документами (admin)

На сторінці `news-editor.php` реалізовано інтерфейс для створення, оновлення та видалення новин або анонсів. Дані зберігаються в таблицях `news` і `news_announcements`.

Ключовий функціонал:

- додавання заголовку, тексту, статусу активності;
- встановлення типу (анонс або новина);
- автоматичне збереження дати публікації;
- виведення на головній сторінці у блоці "Новини ПНДЛ".

Приклад фрагменту PHP-коду додавання новини:

```
$stmt = $pdo->prepare("INSERT INTO news (title, content, is_active, date_posted)
VALUES (?, ?, ?, CURDATE());");$stmt->execute([$title, $content, $isActive]);
```

Рисунок 3.10 – Форма редагування новин

Сторінка `reports.php` дозволяє адміністраторам аналізувати статистику поданих заявок:

Можливості:

- фільтрація за місяцем, кварталом, роком;
- підрахунок кількості заявок;
- побудова графіків (`chart.js`);
- експорт у Excel (через `PHPExcel`);
- порівняльний аналіз періодів.

Приклад фрагменту коду для експорту в Excel:

```
$spreadsheet = new Spreadsheet();
$sheet = $spreadsheet->getActiveSheet();
$sheet->setCellValue('A1', 'Назва заявки');
```

// ...

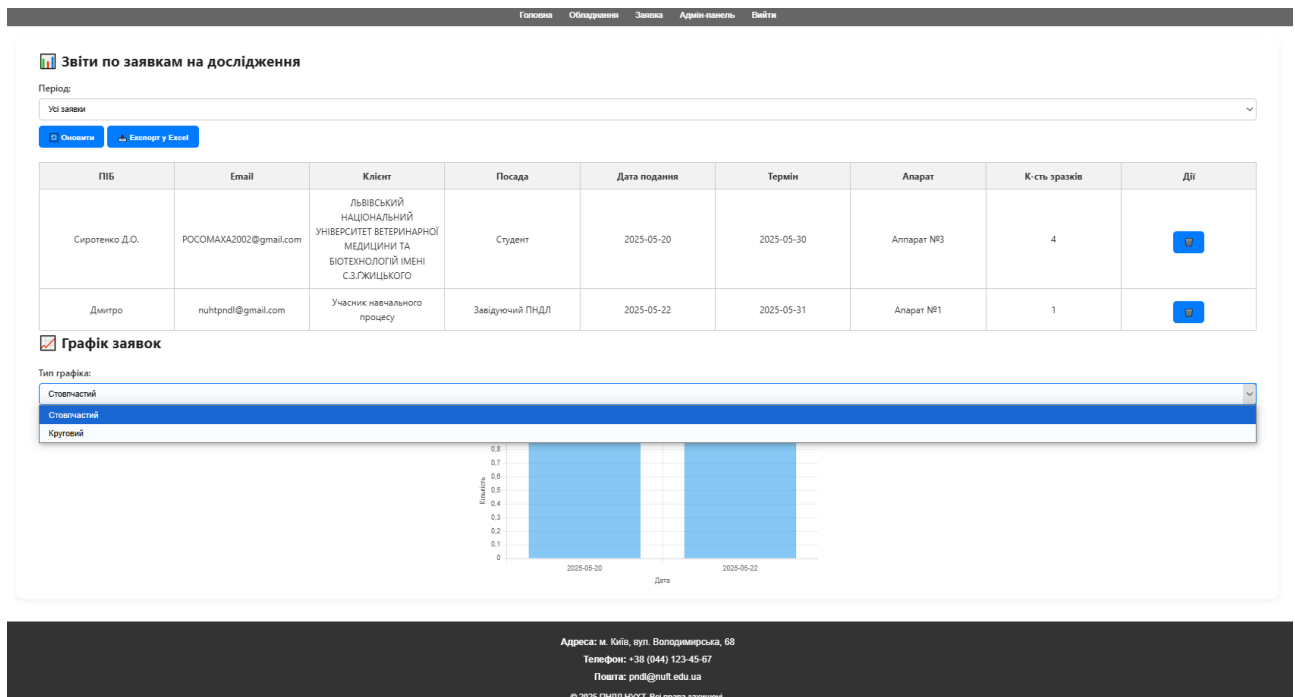
`$writer = new Xlsx($spreadsheet);``$writer->save('звіти.xlsx');`

Рисунок 3.11 – Графік кількості заявок за місяць

Модуль `achievements.php` реалізує функцію додавання, редагування та видалення одиниць обладнання, які користувач може вибрати під час подання заявки.




Інтерфейс передбачає:

- введення назви, опису;
- додавання зображення;
- вивід у вигляді сітки (grid) з картками;
- збереження шляху до зображення в таблиці `achievements`.

Приклад фрагменту PHP-коду для додавання обладнання:

```
$imgPath = "uploads/images/" . $_FILES['image']['name'];
move_uploaded_file($_FILES['image']['tmp_name'], $imgPath);
$stmt = $pdo->prepare("INSERT INTO achievements (name, description, file_path,
created_at) VALUES (?, ?, ?, NOW())");
$stmt->execute([$name, $desc, $imgPath]);
```

Обладнання — список

Назва	Опис	Фото	Дії
РОТАЦІЙНИЙ РЕОМЕТР "Kinexus Pro+"	-прямий контроль деформації, контроль швидкості...		Редагувати Видалити
Комплект лабораторного обладнання для вимірювання нітрогену та протеїну в харчових продуктах та їх складових	-Діапазон вимірювання: 0,1-200 мг нітрогену ...		Редагувати Видалити
ЛАЗЕРНИЙ ДИФРАКЦІЙНИЙ АНАЛІЗАТОР РОЗМІРІВ ЧАСТИНОК "Mastersizer 3000"	- розмір частинок сухих та мокрих дисперсій у д...		Редагувати Видалити

Адреса: м. Київ, вул. Володимирська, 68

Телефон: +38 (044) 123-45-67

Пошта: pndl@nuft.edu.ua

© 2025 ПНДЛ НУХТ. Всі права захищені.

Рисунок 3.12 – Каталог технічного обладнання

3.4. Інструкція користувача

Інформаційна система розроблена з урахуванням принципів інтуїтивної взаємодії, що забезпечує простоту та доступність навіть для нових користувачів. Інтерфейс підтримує логічну послідовність дій, а всі основні функції розташовані у зрозумілих та структурованих блоках.

Нижче подано покроковий сценарій роботи користувача з системою.

Користувач переходить на сторінку register.php, де вводить обов'язкові дані:

- Прізвище, ім'я, по батькові;
- Адресу електронної пошти;
- Обрану роль (студент, викладач, зовнішній клієнт);
- Посаду;
- Пароль.

Після натискання кнопки «Зареєструватися», система:

1. Перевіряє унікальність email;
2. Хешує пароль;
3. Генерує verify-токен;

4. Надсилає лист на пошту.

Користувач активує обліковий запис, перейшовши за посиланням у листі.

The screenshot shows a web interface for user registration. At the top, there is a dark navigation bar with links: Головна, Обладнання, Заявка, Реєстрація, Вхід. Below the navigation bar, the breadcrumb trail reads: Головна > Реєстрація. The main content area features a white registration form titled "Реєстрація нового акаунту". The form contains the following fields and elements:

- PIB: [text input field]
- Email: [text input field]
- Пароль: [text input field]
- Підтвердіть пароль: [text input field]
- A blue button labeled "Зареєструватися"
- A link below the button: "Вже маєте акаунт? [Увійти](#)"

At the bottom of the page, there is a dark footer bar containing contact information:

- Адреса: м. Київ, вул. Володимирська, 68
- Телефон: +38 (044) 123-45-67
- Пошта: pndl@nuft.edu.ua

Рисунок 3.13 – Форма реєстрації нового користувача

Після активації користувач може увійти через `login.php`, ввівши email та пароль. Якщо дані введені правильно — система встановлює сесію й перенаправляє користувача до відповідного кабінету:

- `profile-user.php` — для звичайного користувача;
- `profile-worker.php` — для працівника.

У разі помилки або непідтвердженої пошти відображається відповідне повідомлення.

Головна Обладнання Заявка Реєстрація Вхід

Головна > Вхід

Вхід до акаунту

Email:

Пароль:

Увійти

Немає акаунту? [Реєстрація](#)

Адреса: м. Київ, вул. Володимирська, 68
Телефон: +38 (044) 123-45-67
Пошта: pndl@nuft.edu.ua
© 2025 ПНДЛ НУХТ. Всі права захищені.

Рисунок. – 3.14 Сторінка входу до системи

На сторінці `submit-order-form.php` користувач заповнює заявку:

- Тип замовника;
- Особисті дані;
- Назва та опис дослідження;
- Вибір обладнання;
- Кількість зразків;
- Дата завершення;
- Коментарі.

Заявка на проведення науково-дослідних робіт з використанням можливостей ЦККНО

Email (не змінюється):

POCOMAXA2002@gmail.com

Від кого заявка?

Університети

- Одеський національний технологічний університет
- Державний біотехнологічний університет
- Львівський НУВМ та БТ ім. С.З.Гжицького
- Херсонський НТУ
- КНУТД
- ОДЕУ

Учасник навчального процесу

Учасник навчального процесу

Інше

Інше

Посада:

Рисунок 3.15 – Форма подання заявки (submit-order-form.php) (Частина 1)

ПІБ:

Дата подання заявки:

22.05.2025

Дата, до якої потрібен результат:

Джерело фінансування:

Кількість зразків:

Оберіть прилад:

- №1
- №2
- №3

Рисунок 3.16 – Форма подання заявки (submit-order-form.php) (Частина 2)

Опис зразків / умови досліджень:

Коментарі:

[← Назад на головну](#)

Рисунок 3.17 – Форма подання заявки (submit-order-form.php) (Частина 3)
 Користувач переходить у profile-user.php, де може:

- переглянути список заявок;
- перевірити статус виконання;
- завантажити результати;
- перейти до чату з працівником.

Вітаємо, Дмитро!

Мої заявки на наукові дослідження

Дата подання	Термін	Прилад	К-сть зразків	Коментар	Статус	Файл
2025-05-22	2025-05-31	Апарат №1	11214	41241	Очікується	—

Рисунок 3.18 – Список поданих заявок у кабінеті користувача

У кожній заявці доступний окремий чат, через який користувач може:

- поставити запитання;
- отримати уточнення;
- надіслати додаткову інформацію.

Повідомлення зберігаються в таблиці messages, і доступні для обох сторін.

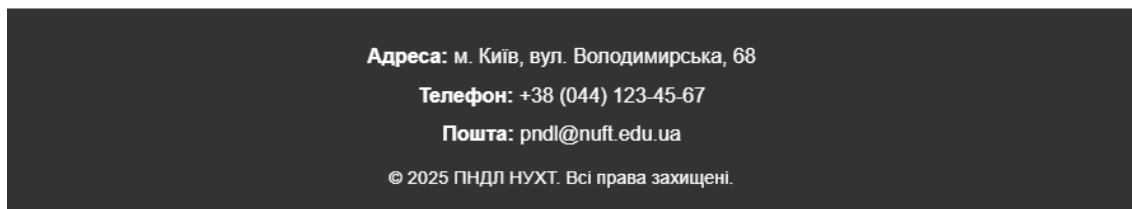
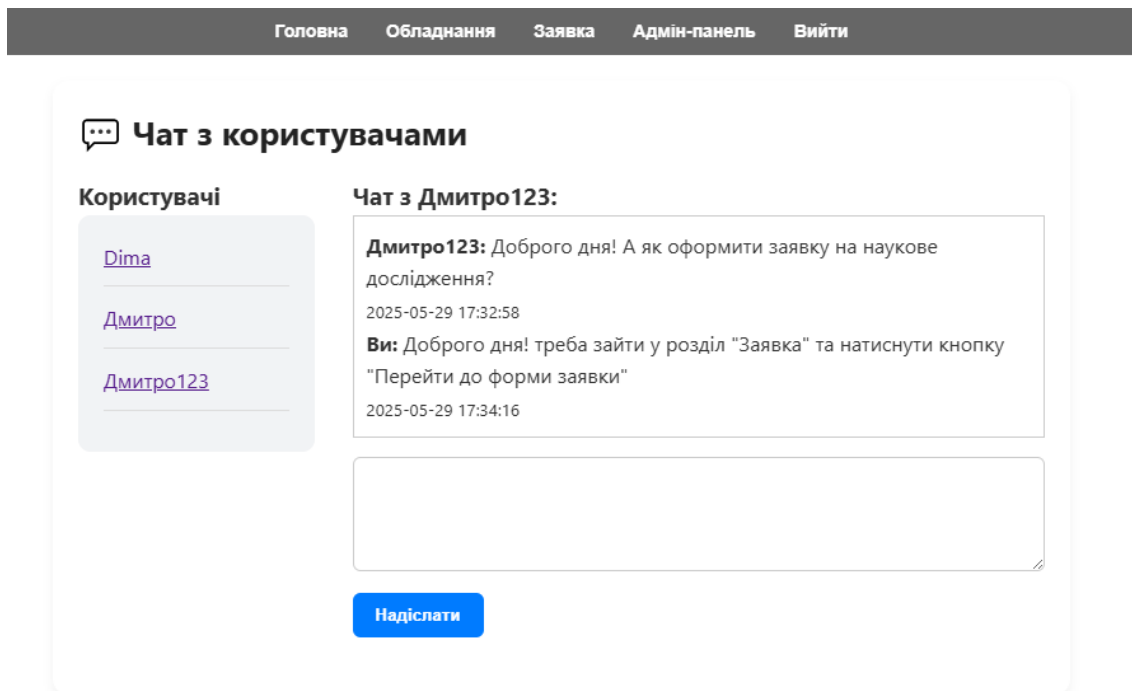


Рисунок. 3.19 – Вікно чату між користувачем і працівником

Після завершення обробки заявки працівник завантажує документ (звіт, протокол), який користувач може переглянути або завантажити у своєму кабінеті. Доступ до документів мають лише авторизовані користувачі.

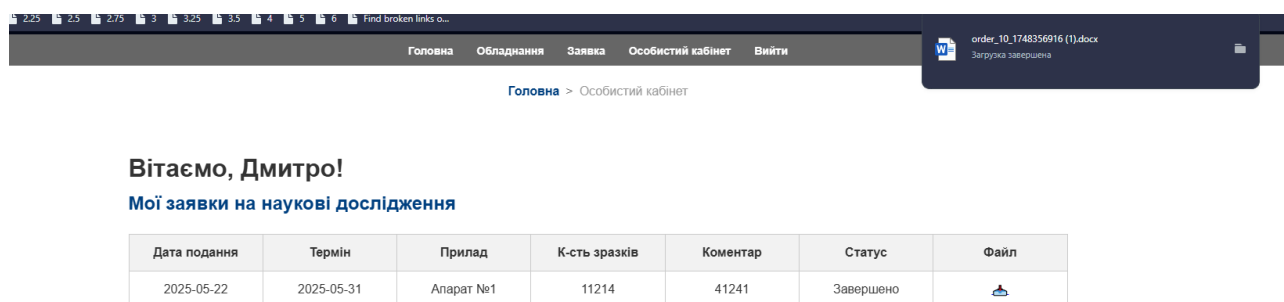


Рисунок 3.20 – Кнопка завантаження результату дослідження

Після завершення роботи користувач натискає кнопку «Вийти», яка призводить до знищення сесії та повернення на головну сторінку.

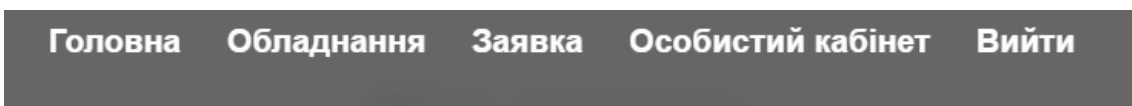


Рисунок 3.21 – Кнопка виходу з системи

3.5. Захист та безпека даних

Інформаційна система реалізована з урахуванням сучасних вимог до безпеки веб-додатків. Захист здійснюється на декількох рівнях: під час аутентифікації, обробки запитів, збереження даних і передачі інформації. Окрему увагу приділено протидії типових атак, таких як SQL-ін'єкції, XSS, крадіжка сесій тощо.

Паролі зберігаються у базі даних у вигляді хешів за допомогою алгоритму `password_hash()`, який є стійким до перебору та не дозволяє відновити пароль у зворотному порядку.

Приклад фрагменту коду:

```
$hash = password_hash($password, PASSWORD_DEFAULT);
```

Усі запити до бази даних реалізовані через механізм підготовлених запитів (Prepared Statements) бібліотеки PDO. Це виключає можливість вставки шкідливого SQL-коду з боку користувача.

Приклад фрагменту коду:

```
$stmt = $pdo->prepare("SELECT * FROM users WHERE email = ?");
$stmt->execute([$email]);
```

Усі захищені сторінки починаються з перевірки активної сесії та ролі користувача. У разі відсутності автентифікації — здійснюється автоматичне перенаправлення на `login.php`.

Приклад фрагменту коду:

```
session_start();
if (!isset($_SESSION['user_id']) || $_SESSION['role'] !== 'user') {
    header("Location: login.php");
    exit;
}
```

Перед збереженням будь-яких даних до бази перевіряється коректність введення на двох рівнях:

Клієнтська перевірка через JavaScript (`required`, `pattern`);

Серверна перевірка через PHP (`filter_var`, `!empty`).

Це дозволяє не лише підвищити UX, але й запобігти масовим автоматизованим запитам.

Приклад фрагменту перевірки email:

```
if (!filter_var($_POST['email'], FILTER_VALIDATE_EMAIL)) {
    $errors[] = "Некоректна адреса електронної пошти.";
}
```

Усі дані, що виводяться на сторінки (особливо коментарі та повідомлення), проходять фільтрацію за допомогою функції htmlspecialchars(), що запобігає виконанню JavaScript-коду.

Приклад фрагменту коду:

```
echo htmlspecialchars($message['message_text'], ENT_QUOTES, 'UTF-8');
```

Таким чином, інформаційна система побудована з урахуванням багаторівневого захисту: від безпечного зберігання до надійного відображення та валідації даних.

3.6. Тестування інформаційної системи

З метою перевірки функціональності, надійності та стійкості інформаційної системи було проведено ручне тестування основних модулів. Для кожної ключової функції сформовано набір тест-кейсів, які охоплюють як позитивні, так і негативні сценарії використання.

Тестування проводилось у локальному середовищі XAMPP, з використанням реальних даних користувачів і симульованих атак.

Обрана методика тестування — мануальне функціональне тестування на основі тест-кейсів. Для кожної функції системи було визначено:

- очікувану поведінку (expected result);
- умови тестування (preconditions);
- кроки виконання (steps);
- фактичний результат (actual result).

Таблиця 3.2 відображає приклади тестових сценаріїв, пов'язаних з функціональністю реєстрації користувачів.

У цьому переліку охоплено як ситуації успішного створення облікового запису, так і перевірку реакції системи на поширені помилки вводу. Наприклад, спроби зареєструвати вже існуючу електронну пошту або відправити форму без заповнення обов'язкових полів. Метою є підтвердження стійкості модуля до помилок користувача та забезпечення коректної обробки введених даних.

Таблиця 3.2. Тест-кейси для модуля реєстрації

№	Назва тесту	Умова	Кроки	Очікуваний результат	Фактичний результат
ТС-1	Реєстрація з валідними даними	Email не зареєстровано	Ввести всі поля, натиснути "Зареєструвати ся"	Лист підтвердження на email	Успішно
ТС-2	Реєстрація з уже існуючим email	Email вже у БД	Ввести той самий email	Повідомлення "Email вже використовується"	Успішно
ТС-3	Некоректний email	Email без "@"	Ввести email: usergmail.com	Помилка валідації	Успішно
ТС-4	Порожній пароль	Поле залишене порожнім	Відправити форму	Повідомлення "Пароль обов'язковий"	Успішно

Таблиця 3.3 містить тест-кейси, які стосуються процедури входу до системи.

Тут перевірено кілька варіантів поведінки: авторизація з валідними обліковими даними, введення неправильного пароля, а також ситуація, коли користувач не підтвердив email. Таким чином, забезпечується перевірка коректності доступу до функціоналу системи відповідно до логіки автентифікації.

Таблиця 3.3. Тест-кейси для входу в систему

№	Назва тесту	Умова	Кроки	Очікуваний результат	Фактичний результат
ТС-5	Вхід з правильними даними	Користувач підтверджений	Ввести правильний email та пароль	Перехід до профілю	Успішно
ТС-6	Вхід з неправильним паролем	Email існує	Ввести email + неправильний пароль	Повідомлення про помилку	Успішно
ТС-7	Вхід без підтвердження email	Користувач не натиснув verify	Ввести логін і пароль	Блокування входу	Успішно

Таблиця 3.4 описує основні перевірки, що проводяться під час подання заявки через сайт. У переліку враховані як базові сценарії (успішне надсилання повної заявки), так і нестандартні (пропуск обов'язкового поля, спроба SQL-ін'єкції). Такий підхід дозволяє оцінити стабільність обробки даних і наявність захисту від потенційних загроз безпеці.

Таблиця 3.4. Тест-кейси для подання заявки

№	Назва тесту	Умова	Кроки	Очікуваний результат	Фактичний результат
ТС-8	Коректна заявка	Усі поля заповнено	Ввести всю інформацію, натиснути "Надіслати"	Заявка збережена в БД	Успішно
ТС-9	Пропущено обов'язкове поле	Не введено тему	Відправити без title	Повідомлення про помилку	Успішно
ТС-10	SQL-ін'єкція в полі	Ввести ' OR 1=1; --	Відправити заявку	Відхилення запиту	Успішно

Таблиця 3.5 охоплює тестування доступу до адміністративної частини сайту. Тут відтворено ситуації, коли користувач намагається відкрити адмін-панель без авторизації або з невідповідною роллю. Таким чином перевіряється реалізація механізмів контролю доступу та відповідність логіці безпеки веб-додатку.

Таблиця 3.5. Тест-кейси для доступу до адмінпанелі

№	Назва тесту	Умова	Кроки	Очікуваний результат	Фактичний результат
ТС-11	Перехід без входу	Сесія відсутня	Відкрити profile-worker.php напряду	Редірект на login.php	Успішно
ТС-12	Перехід під користувачем	Сесія role = user	Відкрити адмінпанель	Блокування доступу	Успішно

Усі основні функціональні модулі пройшли тестування успішно. Виявлені помилки були несуттєвими та усунуті під час розробки. Таблиці вище підтверджують відповідність поведінки системи очікуваним результатам.

3.7. Результати реалізації програмного продукту

У результаті реалізації інформаційної системи було досягнуто ключові функціональні та технічні цілі, визначені на етапі технічного завдання. Система повністю виконує свої задачі у сфері автоматизації замовлення та обробки досліджень у рамках діяльності науково-дослідної лабораторії.

Під час розробки було реалізовано:

- побудову веб-інтерфейсу з адаптивним дизайном для користувачів і працівників;
- стабільну та захищену систему реєстрації та авторизації;
- повний цикл обробки заявки: подання, збереження, перегляд, статус, результат;
- внутрішню систему обміну повідомленнями (чат);
- модуль новин та документів;
- експорт звітів у excel, візуалізацію статистики;
- інтерфейс управління обладнанням та календар замовлень.

Усі реалізовані модулі успішно протестовані й функціонують у локальному середовищі без критичних помилок.

Основні функціональні можливості, що були розроблені:

- система облікових записів з підтвердженням пошти;
- форма подання заявки з розширеним описом;
- чат між користувачем та працівником;
- модуль завантаження документів;
- генерація звітів і графіків;
- календар недоступних дат;
- управління списком обладнання;
- додавання та редагування новин.

У таблиці 3.6 наведений перелік відповідних функціональних елементів технічному завданню на розробку ІС.

Таблиця 3.6 містить порівняльний аналіз між вимогами, сформульованими у технічному завданні, та фактично реалізованими елементами системи. Для кожної функціональної вимоги зазначено, чи було її виконано, а також коротко наведено технічні деталі або реалізовані засоби, які підтверджують відповідність. Такий підхід дозволяє швидко оцінити повноту впровадження функціоналу, зберігаючи зв'язок між початковими вимогами та результатами розробки.

Таблиця 3.6. Відповідність функціоналу технічному завданню

№	Функціональна вимога з ТЗ	Стан реалізації	Коментар
1	Реєстрація з підтвердженням email	Реалізовано	PHPMailer, verify-токен
2	Авторизація з розмежуванням ролей	Реалізовано	user / worker
3	Подання заявки на дослідження	Реалізовано	Збереження в БД
4	Перегляд та управління заявками	Реалізовано	profile-user, profile-worker
5	Система повідомлень	Реалізовано	Модуль чату
6	Завантаження результатів	Реалізовано	Документи прив'язуються до заявки
7	Статистика та звітність	Реалізовано	Chart.js, PHPExcel
8	Управління обладнанням	Реалізовано	CRUD-операції
9	Система новин	Реалізовано	news-editor.php
10	Захист даних та сесій	Реалізовано	password_hash, PDO, session-check

ВИСНОВКИ

У ході виконання кваліфікаційної роботи було розроблено інформаційну веб-систему «NAUKALAB», яка забезпечує автоматизацію основних процесів діяльності Проблемної науково-дослідної лабораторії НУХТ.

Реалізована система повністю відповідає початковим вимогам технічного завдання та досягла таких результатів:

- автоматизовано прийом та обробку заявок на проведення наукових досліджень, що значно скорочує час на їх розгляд і спрощує комунікацію між замовниками та науковцями.
- забезпечено ефективне зберігання та облік інформації про замовлення, документи, дослідження, що дозволяє впорядкувати діяльність лабораторії та уникнути втрат даних.
- реалізовано систему багаторівневого доступу з чітким розподілом ролей (гість, зареєстрований користувач, адміністратор), що забезпечує захист та конфіденційність інформації.
- створено інтуїтивно зрозумілий, адаптивний веб-інтерфейс, доступний з різних пристроїв, з підтримкою темної та світлої тем оформлення.
- забезпечено необхідний рівень безпеки даних (хешування паролів, захист від SQL-ін'єкцій, XSS-атак, HTTPS-з'єднання).
- проведено повне тестування системи, що підтвердило її працездатність та стабільність при високих навантаженнях.

Подальший розвиток системи передбачає впровадження нових функціональних модулів, використання штучного інтелекту для автоматизації обробки заявок, та створення мобільного застосунку, що дозволить розширити аудиторію користувачів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. ДСТУ 3008:2015. Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлювання. Чинний від 01.07.2016. Київ : ДП «УкрНДНЦ», 2015. 32 с.
2. ДСТУ 3918:1999 (ISO/IEC 12207:2008). Інформаційні технології. Процеси життєвого циклу програмного забезпечення. Київ : Держспоживстандарт України, 1999. 57 с.
3. ДСТУ ISO/IEC TR 15504:2006. Інформаційні технології. Оцінювання процесів життєвого циклу програмного забезпечення. Київ : ДП «УкрНДНЦ», 2006. 315 с.
4. ДСТУ ISO/IEC/IEEE 29119-1:2017. Інженерія програмного забезпечення. Тестування програмних засобів. Частина 1. Чинний від 01.01.2018. Київ : ДП «УкрНДНЦ», 2017. 45 с.
5. ДСТУ ISO/IEC 27000:2015. Система управління інформаційною безпекою. Основні положення та словник термінів. Чинний від 01.11.2016. Київ : ДП «УкрНДНЦ», 2015. 84 с.
6. ДСТУ ISO/IEC 15910:2012. Інформаційні технології. Документація користувача. Чинний від 01.07.2013. Київ : ДП «УкрНДНЦ», 2012. 60 с.
7. ДСТУ 2941:1994. Системи оброблення інформації. Терміни та визначення. Київ : Держстандарт України, 1994. 39 с.
8. ДСТУ 1.0:2003. Національна стандартизація. Основні положення. Чинний від 01.01.2004. Київ : Держстандарт України, 2003. 24 с.
9. ДСТУ 2226:1993. Автоматизовані системи. Загальні вимоги. Київ : Держстандарт України, 1993. 55 с.
10. ДСТУ ISO/IEC 29155-1:2015. Інформаційні технології. Оцінювання проектів інформаційних систем. Частина 1. Чинний від 01.03.2016. Київ : ДП «УкрНДНЦ», 2015. 73 с.
11. ДСТУ ISO/IEC 12207:2014. Інженерія програмного забезпечення. Процеси життєвого циклу. На заміну ДСТУ 3918:1999 ; чинний від 01.01.2015. Київ : ДП «УкрНДНЦ», 2014. 120 с.

12. ДСТУ 3321:2003. Системи комп'ютерного дизайну. Терміни та визначення понять. Київ : Держспоживстандарт України, 2003. 48 с.
13. ДСТУ ISO 6309:2007. Пожежна безпека. Загальні вимоги. Чинний від 01.07.2008. Київ : ДП «УкрНДНЦ», 2007. 44 с.
14. М'якшило О. М., Харкянен О. В. Організація баз даних та знань: конспект лекцій. – Київ : НУХТ, 2023. – 147 с.
15. М'якшило О. М., Загоровська Л. Г. CASE-технології у проектуванні інформаційних систем: навч. посібник. – Київ : НУХТ, 2017. – 190 с.
16. Мартін Р. Чиста архітектура. Мистецтво розробки програмного забезпечення. – Київ : Наш Формат, 2019. – 416 с.
17. PHP Manual [Електронний ресурс]. – Режим доступу: <https://www.php.net/docs.php> [Дата звернення: 13.05.2025].
18. MySQL Documentation [Електронний ресурс]. – Режим доступу: <https://dev.mysql.com/doc/> [Дата звернення: 16.05.2025].
19. Bootstrap Documentation [Електронний ресурс]. – Режим доступу: <https://getbootstrap.com/docs/> [Дата звернення: 04.05.2025].
20. PHPMailer Documentation [Електронний ресурс]. – Режим доступу: <https://github.com/PHPMailer/PHPMailer> [Дата звернення: 17.05.2025].
21. Apache HTTP Server Documentation [Електронний ресурс]. – Режим доступу: <https://httpd.apache.org/docs/> [Дата звернення: 14.05.2025].
22. phpMyAdmin Documentation [Електронний ресурс]. – Режим доступу: <https://docs.phpmyadmin.net> [Дата звернення: 15.05.2025].
23. GitHub. Репозиторії з прикладами авторизації на PHP [Електронний ресурс]. – Режим доступу: <https://github.com> [Дата звернення: 08.05.2025].
24. Інструкція користувача XAMPP [Електронний ресурс]. – Режим доступу: <https://www.apachefriends.org/> [Дата звернення: 06.05.2025].
25. OpenWebDocs: Mozilla Developer Network (MDN) [Електронний ресурс]. – Режим доступу: <https://developer.mozilla.org> [Дата звернення: 16.05.2025].
26. Stack Overflow – Спільнота програмістів [Електронний ресурс]. – Режим доступу: <https://stackoverflow.com> [Дата звернення: 09.05.2025].

27. Методичні рекомендації до виконання кваліфікаційної роботи на здобуття освітнього ступеня «бакалавр» спеціальності 122 «Комп'ютерні науки» освітньо-професійної програми «Комп'ютерні науки» денної та заочної форми здобуття освіти [Електронний ресурс] / уклад. С. В. Грибков, Н. В. Ліманська, М. П. Костіков. – Київ : НУХТ, 2025. – 43 с. [Дата звернення: 17.05.2025].

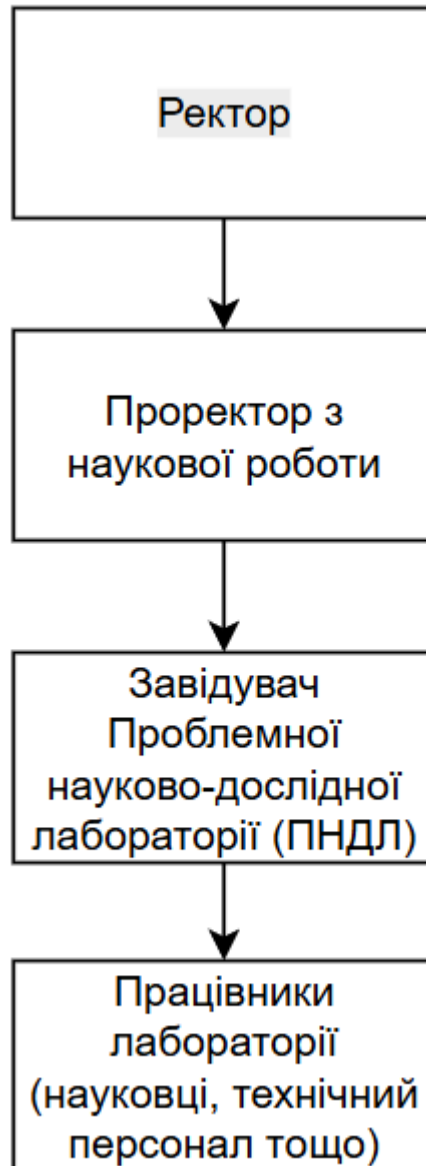
ДОДАТКИ**Додаток А. Організаційна структура ПНДЛ НУХТ**

Рисунок А.1 – Організаційна структура
Проблемної науково-дослідної лабораторії НУХТ

Додаток Б. Моделі бізнес-процесів

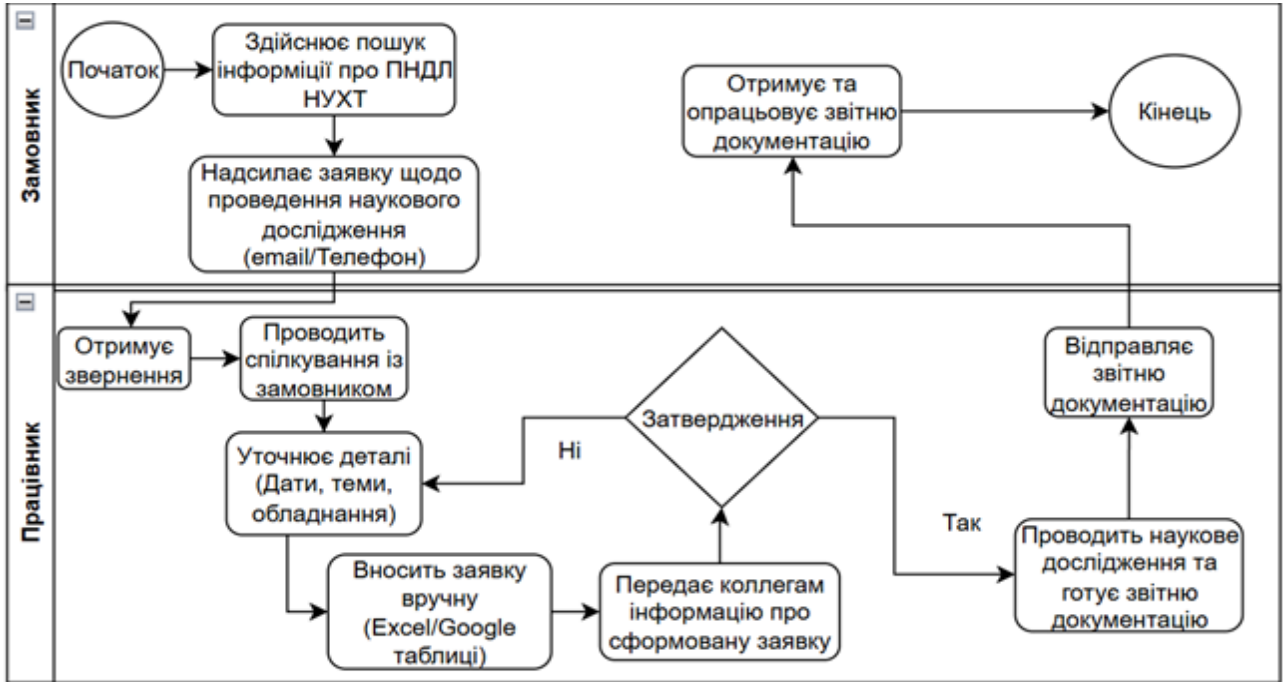


Рисунок Б.1 – Модель бізнес процесів виконання заявки в нотації BPMN статус “AS IS”

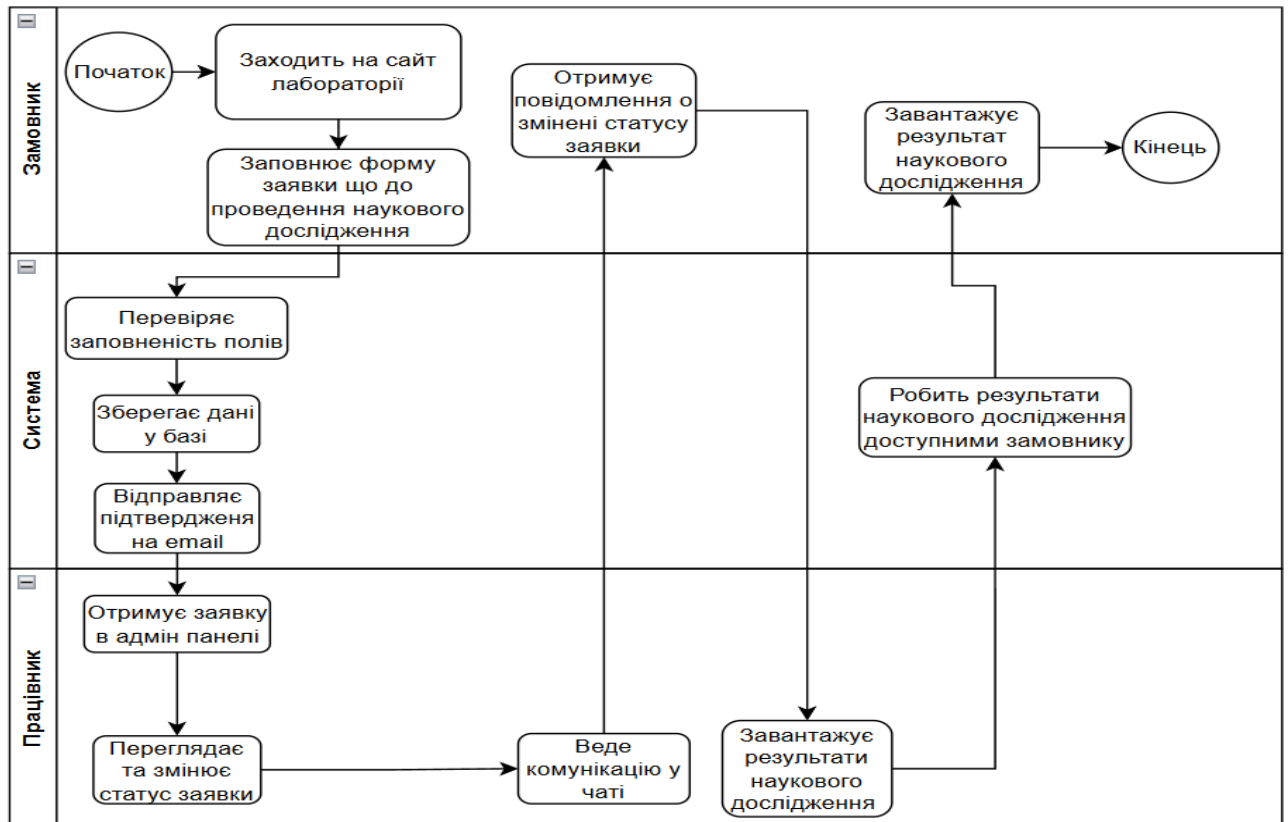


Рисунок Б.2 – Модель бізнес процесів виконання заявки в нотації BPMN статус “TO BE”

Додаток В. Структура бази даних

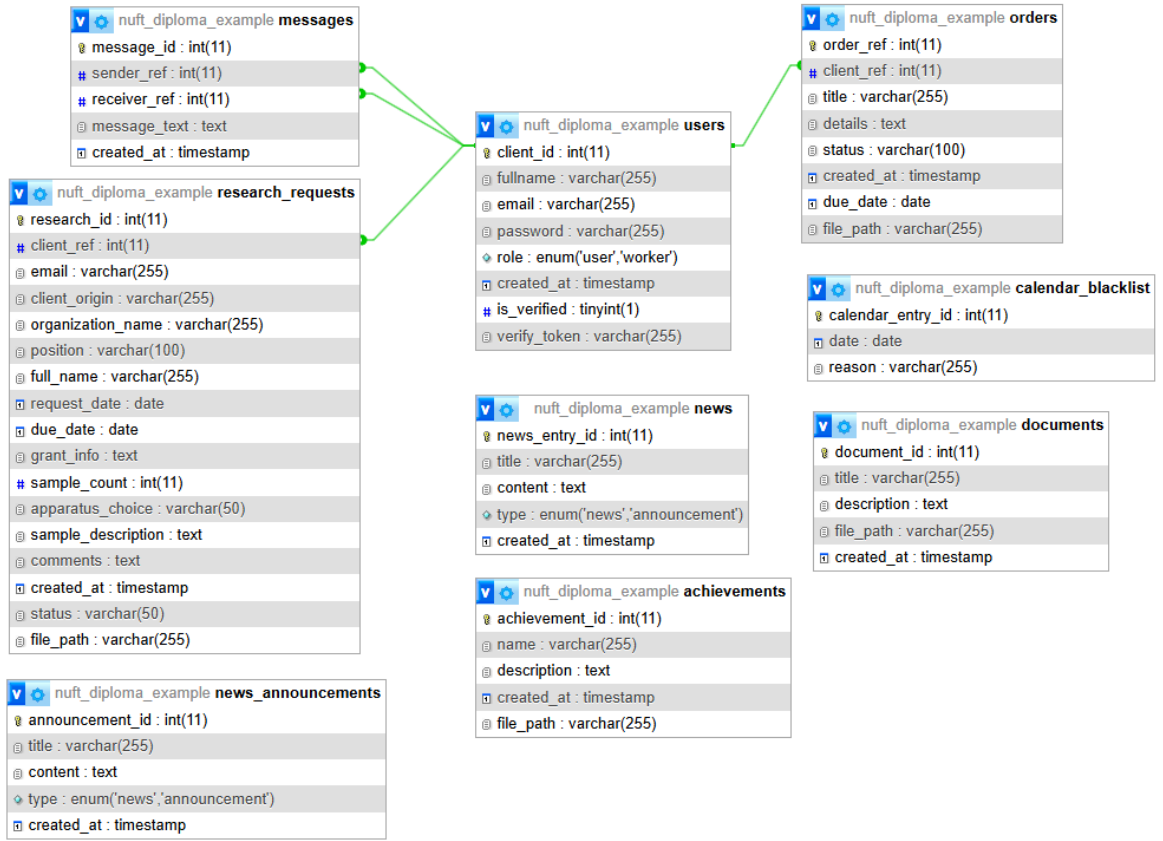


Рисунок В.1 – ER-діаграма бази даних системи NaukaLab

Головна Обладнання Заявка Реєстрація Вхід

Головна > Реєстрація

Реєстрація нового акаунту

ПІБ:

Email:

Пароль:

Підтвердіть пароль:

[Зареєструватися](#)

[Вже маєте акаунт? Увійти](#)

Адреса: м. Київ, вул. Володимирська, 68
 Телефон: +38 (044) 123-45-67
 Пошта: rndl@nuht.edu.ua
 © 2025 ПНДЛ НУХТ. Всі права захищені.

Рисунок Г.3 – Форма реєстрації користувача (register.php)

Головна Обладнання Заявка Реєстрація Вхід

Головна > Вхід

Вхід до акаунту

Email:

Пароль:

[Увійти](#)

[Немає акаунту? Реєстрація](#)

Адреса: м. Київ, вул. Володимирська, 68
 Телефон: +38 (044) 123-45-67
 Пошта: rndl@nuht.edu.ua
 © 2025 ПНДЛ НУХТ. Всі права захищені.

Рисунок Г.4 – Форма входу в систему (login.php)

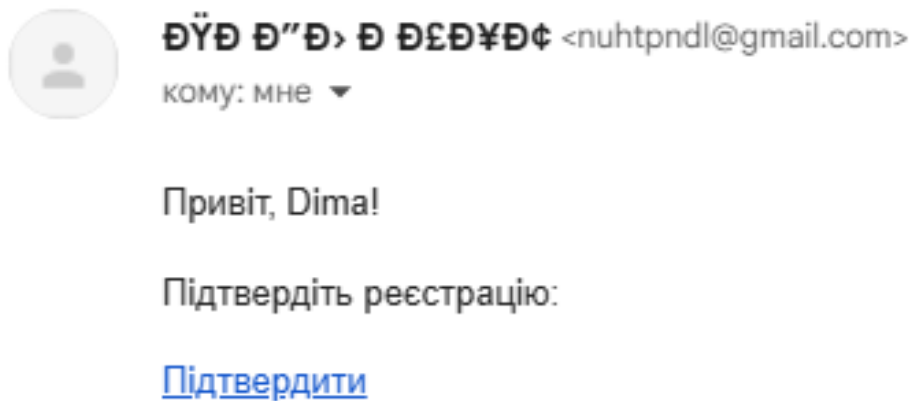


Рисунок Г.5 – Повідомлення про підтвердження пошти (verify.php)

Заявка на проведення науково-дослідних робіт з використанням можливостей ЦККНО

Email (не змінюється):

РОСОМАХА2002@gmail.com

Від кого заявка?

🏛️ Університети

- Одеський національний технологічний університет
- Державний біотехнологічний університет
- Львівський НУВМ та БТ ім. С.З.Гжицького
- Херсонський НТУ
- КНУТД
- ОДЕУ

👤 Учасник навчального процесу

- Учасник навчального процесу

📅 Інше

- Інше

Посада:

Рисунок Г.6 – Форма подання заявки (submit-order-form.php) (Частина 1)

ПІБ:

Дата подання заявки:

22.05.2025

Дата, до якої потрібен результат:

Джерело фінансування:

Кількість зразків:

Оберіть прилад:

- №1
- №2
- №3

Рисунок Г.7 – Форма подання заявки (submit-order-form.php) (Частина 2)

Опис зразків / умови досліджень:

Коментарі:

Надіслати заявку

[← Назад на головну](#)

Рисунок Г.8 – Форма подання заявки (submit-order-form.php) (Частина 3)

[Головна](#) [Обладнання](#) [Заявка](#) [Особистий кабінет](#) [Вийти](#)

Головна > Особистий кабінет

Вітаємо, Дмитро123!
Мої заявки на наукові дослідження

У вас поки що немає заявок. [Подати заявку](#)

Календар зайнятості лабораторії

● — ваша заявка, ✖ — зайнято іншими

< > today травень 2025 р.

нд	пн	вт	ср	чт	пт	сб
27	28	29	30	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30 ✖ Зайнято	31 ✖ Зайнято
1	2	3	4	5	6	7

Чат з адміністрацією

Надіслати

Рисунок Г.9 – Кабінет користувача (profile-user.php)

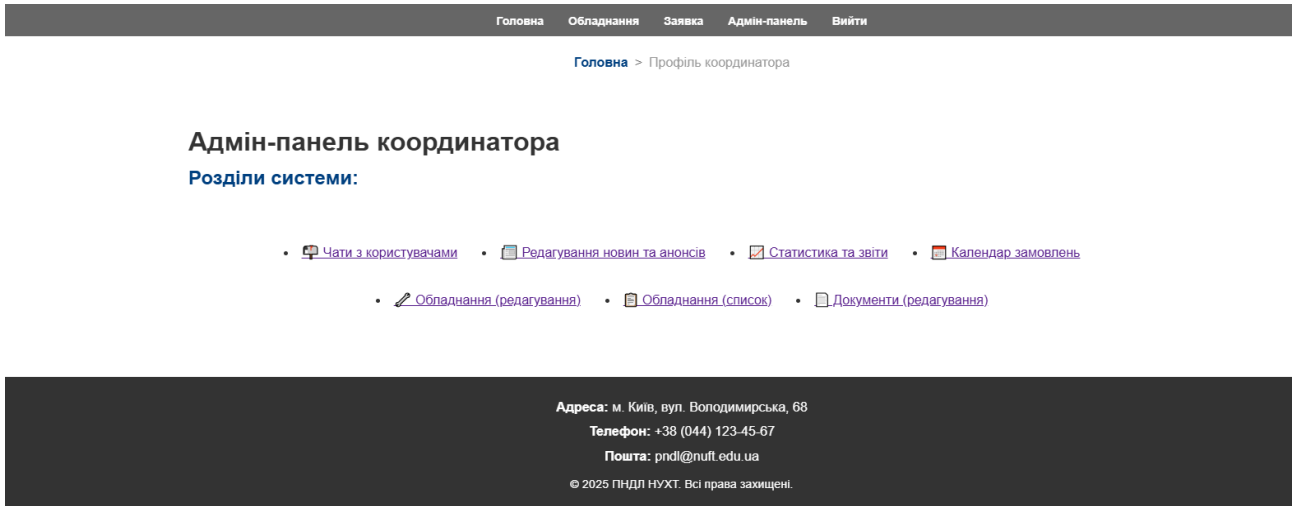


Рисунок Г.10 – Кабінет працівника (profile-worker.php)

Чат з адміністрацією

Рисунок Г.11 – Інтерфейс чату з підтримкою (Вид користувача) (chat-admin.php)

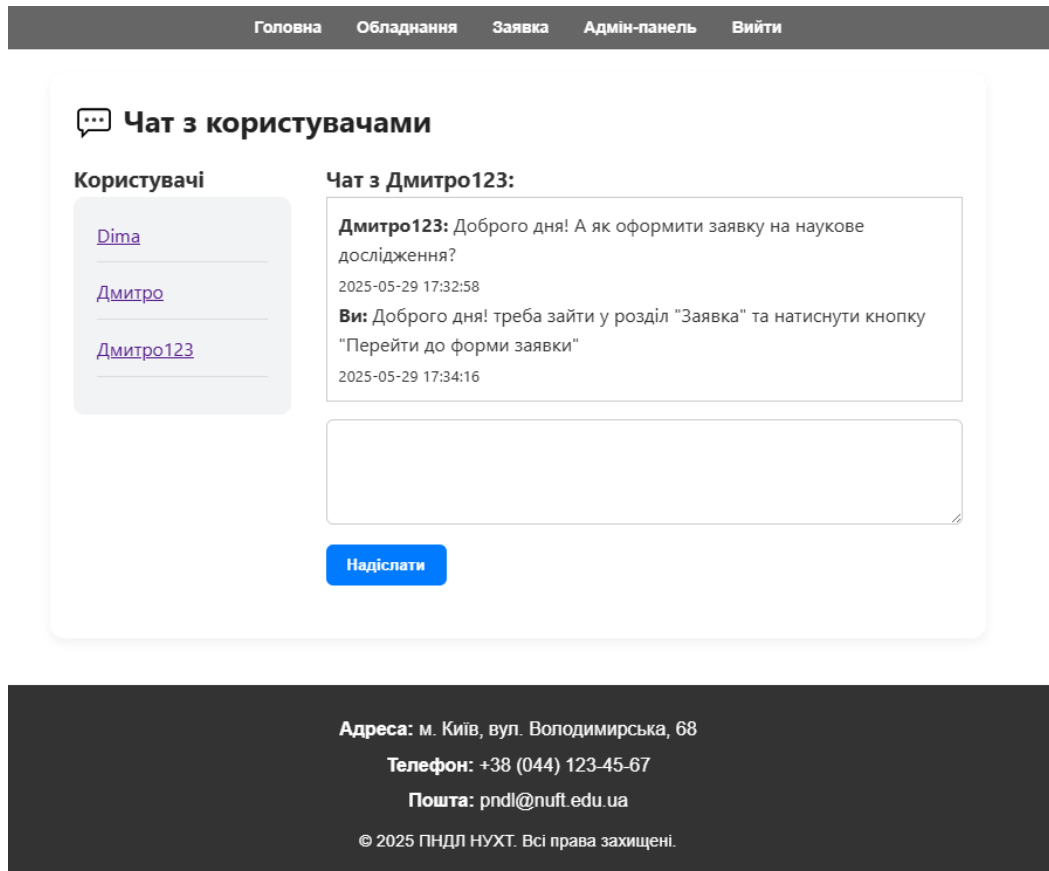


Рисунок Г.12 – Інтерфейс чату з підтримкою (Вид адміністратора) (chat-admin.php)

Головна Обладнання Заявка Адмін-панель Вийти

31 Календар заявок на дослідження

Клікніть на дату для блокування або зняття блокування.

< > today **травень 2025 р.** month week

нд	пн	вт	ср	чт	пт	сб
27	28	29	30	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
				Заблоковані		
25	26	27	28	29	30	31
				Заблоковані	#9: Сиротенко Д	#10: Маринін
1	2	3	4	5	6	7
						Заблоковані

Адреса: м. Київ, вул. Володимирська, 68
 Телефон: +38 (044) 123-45-67
 Пошта: pndl@nuf.edu.ua
 © 2025 ПНДЛ НУХТ. Всі права захищені.

Рисунок Г.13 – Календар замовлень із заблокованими датами (calendar.php)

Завантажити файл

Выберите файл Файл не выбран

Оновити заявку

Видалити заявку

[← Назад до календаря](#)

Рисунок Г.14 – Завантаження готової роботи до заявки (upload-request-work.php)

Управління документами

[+ Додати документ](#)

Назва	Опис	Файл	Дії
АДМН	Дуже гарний предмет		

Адреса: м. Київ, вул. Володимирська, 68

Телефон: +38 (044) 123-45-67

Пошта: pndl@nuff.edu.ua

© 2025 ПНДЛ НУХТ. Всі права захищені.

Рисунок Г.15 – Панель керування документами (documents.php)

Головна
Обладнання
Заявка
Адмін-панель
Вийти

Редагування новин та анонсів

Додати новину / анонс

Заголовок:

Текст:

Тип:

Новина
▼

[Додати](#)

Список записів

[Новина] '33 — 2025-05-22 09:20:08	Редагувати Видалити
[Новина] гфарфашфыаыпывпе — 2025-05-10 20:42:49	Редагувати Видалити
[Новина] Новий проект з АДМН — 2025-05-01 04:01:54	Редагувати Видалити
[Новина] Участь у виставці FoodTech 2025 — 2025-05-01 04:01:54	Редагувати Видалити
[Новина] Грант на дослідження від ЄС — 2025-05-01 04:01:54	Редагувати Видалити
[Анонс] Зустріч випускників 2025 — 2025-05-01 04:01:54	Редагувати Видалити
[Анонс] Конференція "Технології майбутнього" — 2025-05-01 04:01:54	Редагувати Видалити
[Анонс] Відкрита лекція професора Гнатюка — 2025-05-01 04:01:54	Редагувати Видалити
[Анонс] У ПНДЛ ПОЯВИТСЯ САЙТ — 2025-05-01 02:07:06	Редагувати Видалити
[Новина] ЛУЧШИЙ САЙТ ДЛЯ ПНДЛ — 2025-05-01 02:06:41	Редагувати Видалити

Адреса: м. Київ, вул. Володимирська, 68

Телефон: +38 (044) 123-45-67

Пошта: pndl@nuff.edu.ua

© 2025 ПНДЛ НУХТ. Всі права захищені.

Рисунок Г.17 – Інтерфейс редагування новин (news-editor.php)

Звіти по заявкам на дослідження

Період:

ПІБ	Email	Клієнт	Посада	Дата подання	Термін	Апарат	К-сть зразків	Дії
Сирогенко Д.О.	ROSSMAHA2002@gmail.com	Львівський національний університет ветеринарної медицини та біотехнологій імені С.З.Жижельського	Студент	2025-05-20	2025-05-30	Апарат №3	4	<input type="button" value="Ді"/>
Дмитро	nuhtpnd@gmail.com	Учасник навчального процесу	Завідуючий ПНДЛ	2025-05-22	2025-05-31	Апарат №1	1	<input type="button" value="Ді"/>

Графік заявок

Тип графіка:

Адреса: м. Київ, вул. Володимирська, 68
Телефон: +38 (044) 123-45-67
Почта: rpl@npl.gov.ua
© 2025 ПНДЛ ІНХС. Всі права захищені.

Рисунок Г.17 – Сторінка формування звітів (reports.php)

 Графік заявок

Тип графіка:

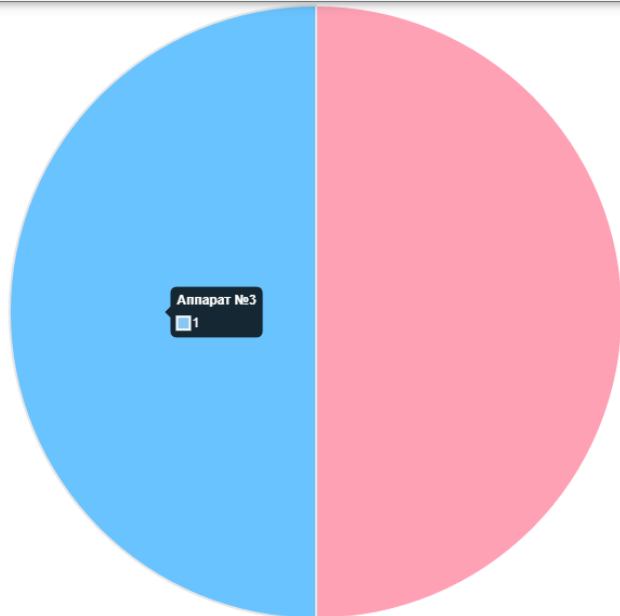
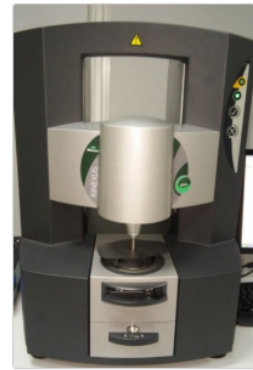


Рисунок Г.18 – Діаграма, що відображує кількість заявок (Chart.js)

Обладнання ЦККНО

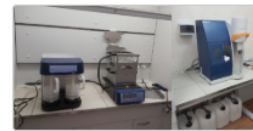
РОТАЦІЙНИЙ РЕОМЕТР "Kinexus Pro+"

- прямий контроль деформації, контроль швидкості зсуву, контроль напруження зсуву;
- режим дослідження (з ручним встановленням параметрів методу аналізу) для розробки методик;
- автоматичний режим (створення стандартних операційних процедур (СОП) процесу вимірювання, створення послідовностей вимірювань за набором СОП);
- діапазон обертових моментів (віскозиметрія, контроль швидкості і напруження зсуву): 5 нНм - 220 мНм;
- діапазон обертового моменту (осциляція, контроль деформації): 1 нНм - 220 мНм;
- роздільна здатність по моменту: не більше 0,1 нНм;
- роздільна здатність по положенню: не більше 10 нрад;
- діапазон кутових швидкостей: не вужче ніж 10 нрад/с - 500 рад/с;
- діапазон частот: 6,28 мкрад/с - 942 рад/с;
- діапазон нормальної сили: 0,001 - 20 Н ;
- швидкість дії (нормальна сила) не більше 10 мс;
- діапазон вертикального переміщення: не менше 230 мм;
- діапазон швидкості вертикального переміщення не вужче ніж 0,1 мкм/с - 35 мм/с;
- роздільна здатність вертикального положення (у всьому діапазоні) : не більше 0,1 мкм.



Комплект лабораторного обладнання для вимірювання нітрогену та протеїну в харчових продуктах та їх складових

- Діапазон вимірювання: 0,1-200 мг нітрогену
- Час дистиляції: 3,5 хв. для 30 мг N (6,5 хв. для 200 мг N)
- Продуктивність дистиляції: приблизно 40 мл/хв
- Об'єм реагенту, що подається насосом: 0-150 мл.
- Відтворюваність: ±1 % від галузевого стандартного зразку
- Відновлення: 99,5 % при 1-200 мг N.



ЛАЗЕРНИЙ ДИФРАКЦІЙНИЙ АНАЛІЗАТОР РОЗМІРІВ ЧАСТИНОК "Mastersizer 3000"

- розмір частинок сухих та мокрих дисперсій у діапазоні 10 нм – 3500 мкм;
- розрахунок питомої поверхні;
- диспергування зразків з використанням відцентрового насоса, мішалки і ультразвукового зонда з автоматизованим управлінням параметрами диспергування через програмне забезпечення;
- програмне регулювання і контроль параметрів диспергування;
- автоматичний модуль повітряного диспергування для крихких, твердих та когезивних матеріалів.



Рисунок Г.19 – Сторінка обладнання (achievements-list.php)

Обладнання — список

Назва	Опис	Фото	Дії
РОТАЦІЙНИЙ РЕОМЕТР "Kinexus Pro+"	-прямий контроль деформації, контроль швидкості...		Редагувати Видалити
Комплект лабораторного обладнання для вимірювання нітрогену та протеїну в харчових продуктах та їх складових	-Діапазон вимірювання: 0,1-200 мг нітрогену ...		Редагувати Видалити
ЛАЗЕРНИЙ ДИФРАКЦІЙНИЙ АНАЛІЗАТОР РОЗМІРІВ ЧАСТИНОК "Mastersizer 3000"	- розмір частинок сухих та мокрих дисперсій у д...		Редагувати Видалити

Адреса: м. Київ, вул. Володимирська, 68

Телефон: +38 (044) 123-45-67

Пошта: pndi@nuft.edu.ua

© 2025 ПНДЛ НУХТ. Всі права захищені.

Рисунок Г.20 – Форма редагування елементів обладнання (achievements-editor.php)

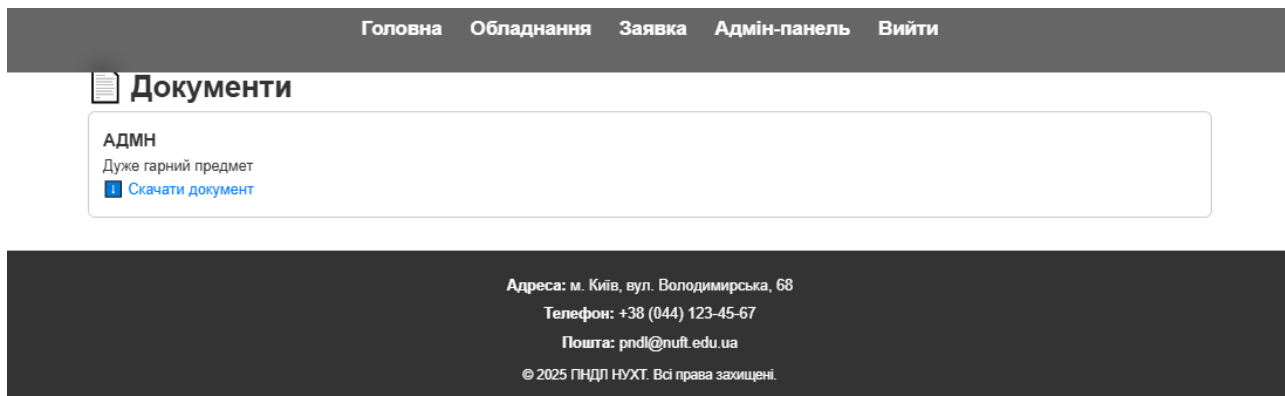


Рисунок Г.21 – Сторінка Документів (documents.php)

Реєстрація користувача (register.php)

HTML:

```
<form method="POST" action="register.php">
  <input type="email" name="email" placeholder="Email" required>
  <input type="password" name="password" placeholder="Пароль" required>
  <button type="submit">Зареєструватися</button>
</form>
```

PHP:

```
<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $email = $_POST['email'];
    $password = password_hash($_POST['password'],
PASSWORD_DEFAULT);
    $stmt = $pdo->prepare("INSERT INTO users (email, password) VALUES (?,
?);");
    $stmt->execute([$email, $password]);
    echo "Реєстрація успішна!";
}
?>
```

Авторизація користувача (login.php)

HTML:

```
<form method="POST" action="login.php">
  <input type="email" name="email" placeholder="Email" required>
  <input type="password" name="password" placeholder="Пароль" required>
  <button type="submit">Увійти</button>
</form>
```

PHP:

```
<?php
session_start();
```

```

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $stmt = $pdo->prepare("SELECT * FROM users WHERE email = ?");
    $stmt->execute([$POST['email']]);
    $user = $stmt->fetch();
    if ($user && password_verify($POST['password'], $user['password'])) {
        $_SESSION['user_id'] = $user['id'];
        header("Location: profile-user.php");
    } else {
        echo "Невірний логін або пароль";
    }
}
?>

```

Обробка авторизації (process_login.php)

```

<?php
require_once 'config.php';
session_start();
?>

```

Головна сторінка (index.php)

```

<?php
require_once 'includes/header.php';
echo "<h1>Ласкаво просимо до NaukaLab</h1>";
require_once 'includes/footer.php';
?>

```

Підтвердження пошти (verify.php)

```

<?php
if (isset($_GET['token'])) {
    $stmt = $pdo->prepare("UPDATE users SET verified = 1 WHERE token =
?");
    $stmt->execute($_GET['token']);
    echo "Поштова адреса підтверджена.";
}

```

```
}
?>
```

Подання заявки (submit-order-form.php) HTML:

```
<form method="POST" action="process-research-request.php">
  <input type="text" name="title" placeholder="Назва дослідження" required>
  <textarea name="description" placeholder="Опис"></textarea>
  <button type="submit">Подати заявку</button>
</form>
```

Обробка заявки (process-research-request.php)

```
<?php
$stmt = $pdo->prepare("INSERT INTO research_requests (title, description)
VALUES (?, ?)");
$stmt->execute([$POST['title'], $POST['description']]);
echo "Заявка збережена!";
?>
```

Перегляд заявок користувача (profile-user.php)

```
<?php
$stmt = $pdo->prepare("SELECT * FROM research_requests WHERE user_id
= ?");
$stmt->execute([$SESSION['user_id']]);
foreach ($stmt as $row) {
  echo
"<div><strong>{$row['title']}</strong><p>{$row['description']}</p></div>";
}
?>
```

Кабінет працівника (profile-worker.php)

```
<?php
$stmt = $pdo->query("SELECT * FROM research_requests");
foreach ($stmt as $row) {
```

```

        echo                                "<div><h3>{$row['title']}</h3><p>Статус:
{$row['status']}</p></div>";
    }
?>

```

Форма заявки (order-submit.php)

Форма Аналогічна “Подання заявки (submit-order-form.php) HTML”

Перегляд заявки (order-details.php)

```

<?php
$stmt = $pdo->prepare("SELECT * FROM research_requests WHERE id = ?");
$stmt->execute([$ _GET['id']]);
$req = $stmt->fetch();
echo "<h2>{$req['title']}</h2><p>{$req['description']}</p>";
?>

```

Звіти та експорт (reports.php, export-orders.php)

```

// reports.php
<?php
$stmt = $pdo->query("SELECT COUNT(*) AS total FROM
research_requests");
$row = $stmt->fetch();
echo "<p>Заявок у системі: {$row['total']}</p>";
?>

// export-orders.php
<?php
header("Content-Type: text/csv");
header("Content-Disposition: attachment; filename=export.csv");
$fp = fopen("php://output", "w");
fputcsv($fp, ["ID", "Назва"]);
$stmt = $pdo->query("SELECT id, title FROM research_requests");
foreach ($stmt as $row) {
    fputcsv($fp, [$row['id'], $row['title']]);
}

```

```
}
fclose($fp);
?>
```

Зміна статусу заявки

```
<?php
$stmt = $pdo->prepare("UPDATE research_requests SET status = ? WHERE id
= ?");
$stmt->execute([$ _POST['status'], $ _POST['id']]);
?>
```

Деталі дослідження (research-details.php)

```
<?php
$stmt = $pdo->prepare("SELECT * FROM research_requests WHERE id = ?");
$stmt->execute([$ _GET['id']]);
$request = $stmt->fetch();
echo "<h1>{$request['title']}</h1><p>{$request['description']}</p>";
?>
```

Оновлення заявки (update-research-request.php)

```
<?php
$stmt = $pdo->prepare("UPDATE research_requests SET title = ?, description
= ? WHERE id = ?");
$stmt->execute([$ _POST['title'], $ _POST['description'], $ _POST['id']]);
?>
```

Видалення заявки (delete-research-request.php)

```
<?php
$stmt = $pdo->prepare("DELETE FROM research_requests WHERE id = ?");
$stmt->execute([$ _POST['id']]);
?>
```

Видалення дослідження (delete-research.php)

```
<?php
$stmt = $pdo->prepare("DELETE FROM research_requests WHERE id = ?");
$stmt->execute([$ _POST['id']]);
?>
```

Модуль новин (news-editor.php, news-edit.php)

```
<!-- news-editor.php -->
<form method="POST">
  <input type="text" name="title" placeholder="Заголовок новини">
  <textarea name="content" placeholder="Текст новини"></textarea>
  <button type="submit">Опублікувати</button>
</form>
// PHP обробка
<?php
$stmt = $pdo->prepare("INSERT INTO news (title, content) VALUES (?, ?)");
$stmt->execute([$ _POST['title'], $ _POST['content']]);
?>
// news-edit.php
<?php
$stmt = $pdo->prepare("UPDATE news SET title = ?, content = ? WHERE id =
?");
$stmt->execute([$ _POST['title'], $ _POST['content'], $ _POST['id']]);
?>
```

Модуль документів (documents.php, add-document.php)

```
// documents.php
<?php
$stmt = $pdo->query("SELECT * FROM documents");
foreach ($stmt as $doc) {
  echo "<a href='{ $doc['path'] }'>{ $doc['title'] }</a><br>";
}
}
```

```

?>
// add-document.php
<?php
    move_uploaded_file($_FILES['file']['tmp_name'],          "uploads/"
$_FILES['file']['name']);
?>

Завантаження результатів дослідження (upload-work.php)
<?php
    move_uploaded_file($_FILES['file']['tmp_name'],          "works/"
$_FILES['file']['name']);
?>

Чат з підтримкою
// chat-send.php
<?php
    $stmt = $pdo->prepare("INSERT INTO messages (sender_id, receiver_id,
message) VALUES (?, ?, ?)");
    $stmt->execute([0, $_POST['receiver_id'], $_POST['message']]);
?>

// chat-send-user.php
<?php
    $stmt = $pdo->prepare("INSERT INTO messages (sender_id, receiver_id,
message) VALUES (?, 0, ?)");
    $stmt->execute([$_SESSION['user_id'], $_POST['message']]);
?>

// chat-admin.php
<?php
    $stmt = $pdo->query("SELECT * FROM messages ORDER BY created_at
DESC");
    foreach ($stmt as $msg) {
        echo "<p><strong>{$msg['sender_id']}:</strong> {$msg['message']}</p>";

```

```
}
```

```
?>
```

Календар замовлень

```
// block-date.php
```

```
<?php
```

```
$stmt = $pdo->prepare("INSERT INTO blocked_dates (date) VALUES (?");
```

```
$stmt->execute([$_POST['date']]);
```

```
?>
```

```
// unblock-date.php
```

```
<?php
```

```
$stmt = $pdo->prepare("DELETE FROM blocked_dates WHERE date = ?");
```

```
$stmt->execute([$_POST['date']]);
```

```
?>
```

Додавання обладнання (achievements.php)

```
?php
```

```
$stmt = $pdo->prepare("INSERT INTO equipment (name, description)
VALUES (?, ?");
```

```
$stmt->execute([$_POST['name'], $_POST['description']]);
```

```
?>
```

Д.24. Видалення обладнання (delete-equipment.php)

```
<?php
```

```
$stmt = $pdo->prepare("DELETE FROM equipment WHERE id = ?");
```

```
$stmt->execute([$_POST['id']]);
```

```
?>
```

Видалення замовлення (delete-order.php)

```
<?php
```

```
$stmt = $pdo->prepare("DELETE FROM orders WHERE id = ?");
```

```
$stmt->execute([$_POST['id']]);
```

```
?>
```

Службові скрипти

```
// config.php
<?php
$pdo = new PDO('mysql:host=localhost;dbname=lab', 'root', '');
$pdo->setAttribute(PDO::ATTR_ERRMODE,
PDO::ERRMODE_EXCEPTION);
?>

// auth.php
<?php
session_start();
if (!isset($_SESSION['user_id'])) {
    header("Location: login.php");
    exit;
}
?>

// logout.php
<?php
session_start();
session_destroy();
header("Location: index.php");
?>
```