

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ ЕКОНОМІКИ І ТЕХНОЛОГІЙ

ННІ	Економіки та бізнес-освіти
Кафедра	Економіки та цифрового бізнесу
Спеціальність	122 «Комп'ютерні науки»
Форма навчання	Заочна

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

Корнієнко Олександра Сергійовича
(прізвище, ім'я, по батькові здобувача)

на тему Розробка вебкаталогу юридичних послуг та онлайн-консультацій
(повна назва теми)
за матеріалами _____
(повна назва бази дослідження)

науковий керівник К.Т.Н. Селезньов М.Є.
(наук. ступінь, вчене звання) (підпис) (прізвище, ініціали)

Робота допущена до захисту в ЕК

Протокол засідання кафедри
від 12 червня 2026 р. № 15
Завідувач кафедри _____

к.е.н., доцент
наук. ступень, вчене звання

(підпис)
Радько В.М.
прізвище, ініціали

ЗАТВЕРДЖЕНО

Наказ Міністерства освіти і науки, молоді та
спорту України
29 березня 2012 року № 384

Форма № Н-9.01

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ ЕКОНОМІКИ І ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ЕКОНОМІКИ ТА БІЗНЕС-ОСВІТИ
 (повне найменування вищого навчального закладу)

Кафедра економіки та цифрового бізнесу
 Освітній ступінь бакалавр
 Спеціальність 122 Комп'ютерні науки

ЗАТВЕРДЖУЮ

Завідувач кафедри _____ **В.М. Радько**

“30” березня 2026 року

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА ЗДОБУВАЧУ

_____ Корнієнко Олександр Сергійовичу _____

1. Тема роботи Розробка вебкаталогу юридичних послуг та онлайн-консультацій _____

науковий керівник роботи _____ Селезньов Максим Євгенович, к.т.н. _____

затверджені наказом вищого навчального закладу від «23» березня 2026 р. № 194-ст

2. Строк подання здобувачем роботи 29.05.2026р.

3. Зміст кваліфікаційної бакалаврської роботи, об'єкт, предмет та мета дослідження:

Розділ 1 Теоретичні аспекти розробки вебкаталогу юридичних послуг та онлайн-консультацій

Розділ 2 Проектування та розробка вебкаталогу юридичних послуг та онлайн-консультацій

Розділ 3 Функціонування та елементи інтерфейсу вебкаталогу юридичних послуг та онлайн-консультацій

Об'єкт дослідження – процес інформаційної та комунікаційної взаємодії між надавачами та споживачами юридичних послуг у цифровому середовищі за допомогою веб-орієнтованих технологій

Предмет дослідження – архітектурні рішення, алгоритми, програмні методи та інструментальні засоби, необхідні для проектування та функціонування вебкаталогу юридичних послуг та онлайн-консультацій

Мета кваліфікаційної бакалаврської роботи – проектування та програмна реалізація високоефективного, адаптивного і безпечного вебкаталогу юридичних послуг з інтегрованим модулем онлайн-консультацій

4. Дата видачі завдання 03.04.2026р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної бакалаврської роботи	Строк виконання етапів роботи	Відмітка керівника про виконання етапів (дата, підпис)
1	Підготовка розділу 1	до 17.04.2026р.	20.04.2026р.
2	Підготовка розділу 2	до 08.05.2026р.	15.05.2026р.
3	Підготовка розділу 3	до 25.05.2026р.	26.05.2026р.
4	Реєстрація завершеної кваліфікаційної роботи	до 29.05.2026р.	28.05.2026р.
5	Отримання відгуку від наукового керівника	04.06.2026р.	04.06.2026р.
6	Отримання зовнішньої рецензії	05.06.2026р.	05.06.2026р.
7	Попередній захист кваліфікаційної роботи на кафедрі	08.06.2026р.	08.06.2026р.
8	Перевірка кваліфікаційної роботи на плагіат	18.06.2026р.	18.06.2026р.
9	Допуск кафедрою кваліфікаційної роботи до захисту	18.06.2026р.	18.06.2026р.
10	Підготовка студента до захисту в ЕК	до 23.06.2026р	22.06.2026р

Завдання підготував науковий керівник

_____ Селезньов М. Є.

Завдання одержав здобувач

_____ (підпис)

_____ Корнієнко О. С.
(прізвище та ініціали)

РЕФЕРАТ

Робота містить 64 сторінки, 12 рисунків, 30 джерел, 2 додатки.

Об'єкт дослідження: процес інформаційної та комунікаційної взаємодії між надавачами та споживачами юридичних послуг у цифровому середовищі за допомогою веб-орієнтованих технологій.

Предмет дослідження: архітектурні рішення, алгоритми, програмні методи та інструментальні засоби, необхідні для проектування та функціонування вебкаталогу юридичних послуг та онлайн-консультацій.

Мета дослідження: проектування та програмна реалізація високоефективного, адаптивного і безпечного вебкаталогу юридичних послуг з інтегрованим модулем онлайн-консультацій.

У роботі проаналізовано сучасні підходи до електронного надання юридичних послуг, досліджено специфіку їх інформаційного забезпечення та виконано порівняльний аналіз існуючих вебресурсів у даній предметній області.

Розроблено інформаційну модель вебкаталогу, спроектовано реляційну базу даних та реалізовано серверну і клієнтську частини програмного продукту.

Проведене тестування підтвердило працездатність реалізованих функцій, високу швидкодію вебдодатку, коректність роботи системи онлайн-консультацій та ефективність засобів забезпечення інформаційної безпеки.

Область застосування: створений вебкаталог може бути впроваджений у діяльність приватних юристів та адвокатських об'єднань з метою ефективного надання онлайн-послуг.

БАЗА ДАНИХ, ВЕБКАТАЛОГ, КЛІЄНТ, ЮРИСТ, ЮРИДИЧНІ ПОСЛУГИ, LEGALTECH

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ	6
ВСТУП	7
РОЗДІЛ 1 ТЕОРЕТИЧНІ АСПЕКТИ РОЗРОБКИ ВЕБКАТАЛОГУ ЮРИДИЧНИХ ПОСЛУГ ТА ОНЛАЙН-КОНСУЛЬТАЦІЙ	11
1.1 Огляд сучасних підходів до електронного надання юридичних послуг	11
1.2 Аналіз існуючих вебресурсів для надання юридичних послуг	16
1.3 Вибір технологічного стеку та архітектури вебкаталогу	21
Висновки до розділу 1	26
РОЗДІЛ 2 ПРОЄКТУВАННЯ ТА РОЗРОБКА ВЕБКАТАЛОГУ ЮРИДИЧНИХ ПОСЛУГ ТА ОНЛАЙН-КОНСУЛЬТАЦІЙ	28
2.1 Проєктування інформаційної моделі та бази даних вебкаталогу	28
2.2 Опис основних способів обробки та збереження даних	31
2.3 Реалізація серверної та клієнтської частини вебкаталогу	35
Висновки до розділу 2	42
РОЗДІЛ 3 ФУНКЦІОНУВАННЯ ТА ЕЛЕМЕНТИ ІНТЕРФЕЙСУ ВЕБКАТАЛОГУ ЮРИДИЧНИХ ПОСЛУГ ТА ОНЛАЙН- КОНСУЛЬТАЦІЙ	44
3.1 Огляд функціоналу розробленого вебкаталогу	44
3.2 Опис взаємодії користувача з розробленим вебкаталогом	47
3.3 Аналіз ефективності розробленого вебкаталогу та можливих шляхів його подальшого розвитку	52
Висновки до розділу 3	55
ВИСНОВКИ	58
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	62
ДОДАТКИ	65

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

ООП - Об'єктно-орієнтоване програмування

СУБД - Система управління базами даних

API -Application Programming Interface

CMS - Content Management System

CRM - Customer Relationship Management

HTML - HyperText Markup Language

HTTPS - HyperText Transfer Protocol Secure

MVC - Model-View-Controller

PDO - PHP Data Objects

SEO - Search Engine Optimization

UI - User Interface

UX - User Experience

ВСТУП

Сучасний етап розвитку інформаційного суспільства характеризується глобальною цифровізацією всіх сфер життєдіяльності людини. Сфера надання професійних послуг, зокрема юридичних, зазнає глибоких трансформацій під впливом впровадження новітніх інформаційних технологій. Традиційні форми взаємодії між юристом та клієнтом, що передбачають фізичну присутність та паперовий документообіг, поступово поступаються місцем дистанційним електронним сервісам (LegalTech). Це зумовлено зростаючою потребою суспільства у швидкому, зручному та безпечному доступі до правової допомоги незалежно від географічного розташування сторін.

Незважаючи на наявність на ринку значної кількості вебсайтів юридичних компаній, більшість із них виконують виключно репрезентативну функцію (сайти-візитки) та не надають повноцінного інструментарію для інтерактивної взаємодії. Існуючі каталоги та агрегатори часто побудовані на базі універсальних систем керування контентом (CMS), що призводить до надмірної перевантаженості коду, зниження швидкодії та появи вразливостей у системі безпеки бази даних. З огляду на специфіку правової допомоги, яка вимагає суворої конфіденційності, швидкості реагування (через тикет-системи або чати) та зручності сприйняття інформації, виникає гостра необхідність у розробці спеціалізованих платформ. Створення вебкаталогу юридичних послуг та онлайн-консультацій на базі чистого самописного коду (із застосуванням PHP, MySQL та нативних клієнтських технологій) дозволяє усунути недоліки готових платформ, забезпечивши максимальну оптимізацію, гнучкість масштабування та високий рівень захисту персональних даних користувачів. Саме тому розробка архітектури та програмна реалізація такого вебресурсу є актуальним науково-практичним завданням.

Проблематика проєктування веб-орієнтованих інформаційних систем та впровадження електронних послуг активно досліджується у вітчизняній та зарубіжній літературі. Питання загальної архітектури вебдодатків, принципів

побудови реляційних баз даних та оптимізації клієнт-серверної взаємодії висвітлені у фундаментальних працях фахівців із програмної інженерії. Дослідження у сфері LegalTech здебільшого зосереджені на теоретичних аспектах автоматизації юридичної діяльності або на використанні систем штучного інтелекту для аналізу нормативних актів.

Водночас, в існуючій літературі недостатньо повно розкрито питання комплексного підходу до розробки вузькоспеціалізованих систем, які поєднують у собі функціонал структурованого каталогу та системи надання консультацій у режимі реального часу (або за принципом тикетів) без використання громіздких фреймворків. Більшість доступних джерел пропонує розв'язання подібних завдань шляхом конфігурування готових CMS (наприклад, WordPress або Joomla), що не відповідає сучасним вимогам до створення високонавантажених і безпечних систем із кастомною логікою. Відтак, існує потреба в систематизації підходів до написання чистого коду для вирішення конкретної бізнес-задачі у правовій сфері.

Інформаційною та теоретичною основою роботи слугує комплекс джерел, які можна умовно поділити на кілька груп. До першої групи належать офіційні специфікації та технічна документація консорціуму W3C, розробників мови програмування PHP та систем управління базами даних (зокрема, MySQL). Друга група охоплює науково-технічну літературу, монографії, підручники та статті у фахових періодичних виданнях, які присвячені методології об'єктно-орієнтованого програмування, проєктуванню реляційних баз даних (нормалізація, захист від SQL-ін'єкцій) та принципам побудови інтерфейсів користувача (UI/UX). До третьої групи належать нормативно-правові акти України, що регулюють питання електронної комерції, захисту інформації та надання адвокатських послуг, які визначають правові рамки функціонування розроблюваної системи. Крім того, використовуються аналітичні огляди існуючих програмних рішень та аналоги вебкаталогів для проведення порівняльного аналізу.

Об'єкт дослідження – процес інформаційної та комунікаційної взаємодії між надавачами та споживачами юридичних послуг у цифровому середовищі за допомогою веб-орієнтованих технологій.

Предмет дослідження – архітектурні рішення, алгоритми, програмні методи та інструментальні засоби, необхідні для проєктування та функціонування вебкаталогу юридичних послуг та онлайн-консультацій.

Мета роботи полягає у проєктуванні та програмній реалізації високоефективного, адаптивного і безпечного вебкаталогу юридичних послуг з інтегрованим модулем онлайн-консультацій.

Відповідно до поставленої мети та структури дослідження, необхідно вирішити такі науково-практичні завдання:

- провести огляд сучасних підходів до електронного надання юридичних послуг та дослідити специфіку їх інформаційного забезпечення;
- здійснити системний аналіз існуючих вебресурсів та аналогів для надання юридичних послуг, виявивши їхні технічні переваги та недоліки;
- обґрунтувати вибір технологічного стеку та загальної архітектури для розробки власного вебкаталогу;
- спроектувати інформаційну модель та розробити оптимальну структуру реляційної бази даних вебкаталогу;
- описати основні способи обробки та збереження даних, алгоритми шифрування та захисту інформації від несанкціонованого доступу;
- виконати програмну реалізацію серверної (Backend) та клієнтської (Frontend) частин вебкаталогу;
- провести огляд розробленого функціоналу, включаючи каталог послуг, особисті кабінети та тикет-систему онлайн-консультацій;
- описати логіку взаємодії користувача з розробленим вебкаталогом, оцінити зручність та адаптивність інтерфейсу;
- проаналізувати технічну та функціональну ефективність розробленого вебкаталогу, а також сформулювати можливі шляхи його подальшого розвитку та масштабування.

Для досягнення мети та виконання поставлених завдань у роботі застосовано комплекс загальнонаукових та спеціальних методів. Метод системного аналізу використано для дослідження предметної області, декомпозиції процесів надання юридичних послуг та формування вимог до програмного забезпечення. Метод порівняльного аналізу застосовано під час огляду існуючих аналогів та вибору оптимального технологічного стеку. На етапі проєктування використано методи інфологічного та датологічного моделювання баз даних. У процесі практичної реалізації застосовано методи структурного та об'єктно-орієнтованого програмування. Для перевірки працездатності створеної системи та її компонентів використано методи функціонального та навантажувального тестування (емпіричні методи).

Наукова новизна роботи полягає в подальшому розвитку підходів до створення спеціалізованих веб-орієнтованих систем у галузі права. Вперше або на якісно новому рівні обґрунтовано та реалізовано концепцію інтеграції структурованого каталогу послуг із системою тикет-консультацій, архітектура якої побудована виключно на нативних технологіях (без застосування готових CMS), що дозволило оптимізувати структуру бази даних спеціально під вимоги конфіденційного правового документообігу та суттєво підвищити показники швидкості обробки транзакцій.

Практична значущість результатів дослідження. Розроблений вебкаталог є повністю готовим, самостійним програмним продуктом, який може бути впроваджений у діяльність приватних юристів та адвокатських об'єднань. Застосування створеної платформи дозволяє автоматизувати процес прийому заявок, систематизувати надання онлайн-консультацій, знизити витрати часу на маршрутизацію клієнтських звернень та підвищити загальний рівень сервісу. Матеріали роботи, спроектовані моделі баз даних та алгоритми обробки інформації можуть бути використані як методична база у процесі підготовки фахівців ІТ-галузі під час вивчення дисциплін, пов'язаних із веброзробкою та проєктуванням баз даних. Структура коду створена таким чином, що вона легко піддається подальшому масштабуванню, зокрема для інтеграції з API сторонніх сервісів (наприклад, Telegram-сповіщень або систем онлайн-оплати).

РОЗДІЛ 1

ТЕОРЕТИЧНІ АСПЕКТИ РОЗРОБКИ ВЕБКАТАЛОГУ ЮРИДИЧНИХ ПОСЛУГ ТА ОНЛАЙН-КОНСУЛЬТАЦІЙ

1.1 Огляд сучасних підходів до електронного надання юридичних послуг

Стрімкий розвиток інформаційно-комунікаційних технологій та глобальні процеси цифровізації суспільства зумовили докорінну трансформацію класичних інститутів та індустрій. Сфера правової допомоги та юридичного консультування, яка традиційно вважалася однією з найбільш консервативних через сувору прив'язку до паперового документообігу та необхідність особистого контакту між правознавцем і клієнтом, наразі переживає етап активного формування нового технологічного укладу, відомого у світовій практиці як LegalTech. Електронне надання юридичних послуг (e-justice, e-legal services) перестало бути виключно допоміжним інструментом автоматизації внутрішніх процесів юридичних компаній і перетворилося на самостійний, високотехнологічний сектор цифрової економіки.

Трансформація підходів до дистанційного правового обслуговування обумовлена як внутрішніми чинниками розвитку ІТ-галузі, так і зовнішніми соціально-економічними змінами. Сучасний споживач послуг висуває високі вимоги до оперативності, прозорості, доступності та фінансової оптимізації процесів отримання допомоги. Традиційний візит до офісу юриста пов'язаний із часовими витратами на логістику, географічними обмеженнями та високою невизначеністю щодо вартості й кваліфікації спеціаліста. Впровадження веб-орієнтованих платформ дозволяє нівелювати ці бар'єри, забезпечуючи цілодобовий доступ до правової експертизи з будь-якої точки світу, де є підключення до мережі Інтернет.

Аналіз науково-технічної літератури та практичного досвіду розгортання сучасних інформаційних систем дозволяє виділити кілька ключових підходів до організації електронного надання юридичних послуг:

1. Інформаційно-довідковий (пасивний) підхід. Історично перша форма представлення юридичної сфери в мережі, що базується на створенні статичних вебресурсів, сайтів-візиток або інформаційних порталів, де розміщуються нормативно-правові акти, аналітичні статті, роз'яснення законодавства та контактні дані фахівців. Такі системи виконують виключно просвітницьку та маркетингову функції, не забезпечуючи двосторонньої інтерактивної взаємодії в реальному часі.

2. Комунікаційно-консультаційний підхід. Даний напрям орієнтований на створення віртуального простору для безпосереднього діалогу між клієнтом та правознавцем. Взаємодія реалізується через інтегровані модулі текстових чатів, систем обміну миттєвими повідомленнями, закритих форумів або тикет-систем. Особливістю цього підходу є перенесення процесу консультування в асинхронний або синхронний онлайн-режим, що дозволяє фіксувати історію звернень, структурувати запити та забезпечувати конфіденційність каналів зв'язку.

3. Автоматизований (транзакційний) підхід за допомогою конструкторів документів. Цей підхід базується на мінімізації участі людини у рутинних операціях. За допомогою спеціалізованого програмного забезпечення клієнт, відповідаючи на питання інтерактивного опитувальника, може згенерувати юридично грамотний договір, позовну заяву, скаргу чи інший процесуальний документ. Системи такого типу використовують заздалегідь запрограмовані логічні матриці та шаблони.

4. Агрегаційний підхід (модель маркетплейсів та вебкаталогів). Найбільш комплексний та перспективний підхід на сучасному етапі, який передбачає створення дворівневих платформ, що виступають посередниками (інформаційними хабами) між великою спільнотою профільних юристів та потенційними клієнтами. Вебкаталоги акумулюють інформацію про

кваліфікацію, спеціалізацію, вартість послуг та рейтинг спеціалістів, надаючи користувачеві інструменти для точного вибору надавача послуг та одночасного отримання експрес-консультацій.

Здійснюючи проектування сучасного вебкаталогу, необхідно враховувати не лише суто технічні параметри, але й міждисциплінарний контекст, оскільки дистанційне консультування безпосередньо взаємодіє з гуманітарними, педагогічними та психологічними аспектами. Процес спілкування в системі «людина – комп'ютер – людина» ускладнюється відсутністю невербальних маркерів, які є звичними для очного прийому. Клієнт, який звертається за юридичною допомогою, часто перебуває у стані психологічного стресу або когнітивного дисонансу, викликаного правовою невизначеністю чи конфліктною ситуацією.

Тому архітектура вебресурсу повинна спиратися на принципи доступності та ергономіки, що вимагає впровадження методів тестування первинного сприйняття інтерфейсу користувачем [22]. Інформаційна модель вебкаталогу має виконувати певну педагогічну функцію – не просто надавати перелік імен, а структурувати правове поле сайту таким чином, щоб користувач без спеціальної освіти міг ідентифікувати свою проблему, співвіднести її з правильною категорією права (наприклад, цивільне, сімейне, трудове) та підготуватися до комунікації з експертом. Важливу роль тут відіграють інструменти психологічного розвантаження інтерфейсу: відсутність агресивних візуальних елементів, використання заспокійливих кольорових палітр, наявність чітких підказок, індикаторів прогресу виконання тикету та інтуїтивно зрозумілих форм введення даних.

Особливе місце в структурі сучасних підходів посідає забезпечення інклюзивності вебресурсів. Будь-яка сучасна інформаційна система, що претендує на високу соціальну та практичну ефективність, повинна розроблятися з урахуванням міжнародних стандартів доступності вебконтенту WCAG 2.1 [29]. Це передбачає створення інтерфейсів, адаптованих для осіб з обмеженими можливостями (зокрема, з порушеннями зору чи моторики),

забезпечення можливості масштабування тексту без втрати функціональності, підтримку керування з клавіатури та сумісність із програмами екранного читання. Електронна юридична допомога за своєю природою має бути демократичною, усуваючи будь-які фізичні чи соціальні бар'єри між громадянином та правосуддям.

Технічна реалізація сучасних систем LegalTech вимагає від розробника відмови від використання застарілих монолітних архітектур та переходу до гнучких, оптимізованих рішень. Тривалий час стандартною практикою створення подібних сайтів було застосування універсальних систем керування контентом. Проте детальний системний аналіз показує, що готові платформи мають критичні недоліки під час розробки спеціалізованих вебкаталогів:

- по-перше, вони містять надлишковий програмний код, який істотно знижує швидкість завантаження сторінок та створює додаткове навантаження на апаратні потужності сервера;
- по-друге, універсальність архітектури готових CMS робить їх вразливими до поширених векторів кібератак, оскільки структура їхніх баз даних та файлової системи є загальновідомою [25].

З огляду на це, сучасним архітектурним підходом є розробка вебдодатків на базі чистого самописного коду з використанням перевірених паттернів проєктування [14]. Написання нативного серверного коду (Backend) на мові PHP у поєднанні з реляційними СУБД (MySQL) забезпечує повний контроль над логікою обробки запитів, дозволяє оптимізувати структуру таблиць бази даних під конкретні сутності (юристи, клієнти, послуги, тикети) та мінімізувати час виконання SQL-транзакцій. Чистий код легше піддається рефакторингу, аудіту безпеки та подальшому масштабуванню, наприклад, при необхідності інтеграції платформи з мобільними застосунками або сторонніми сервісами сповіщень через API.

Правовий простір функціонування систем електронного надання юридичних послуг в Україні чітко регламентований чинним законодавством. Будь-який вебресурс, що здійснює комерційну діяльність, акумулює персональні

дані або виступає майданчиком для укладання угод, зобов'язаний суворо дотримуватися правових норм. Нормативно-правову основу проєктування таких систем складають базові закони.

Закон України «Про електронну комерцію» визначає організаційно-правові засади діяльності у сфері електронної комерції, окреслює права та обов'язки учасників відносин у цій сфері та встановлює юридичний статус електронних правочинів [6]. Згідно з цим законом, постачальник послуг у вебкаталозі має забезпечити прямий, безперешкодний та постійний доступ користувачів до повної інформації про себе (найменування, місцезнаходження, відомості про ліцензії чи свідоцтва).

Оскільки функціонування онлайн-консультацій пов'язане зі збором, обробкою та зберіганням конфіденційної інформації (прізвища, контакти, опис життєвих та правових проблем, копії документів), критично важливим є дотримання вимог Закону України «Про захист персональних даних» [7]. Програмна архітектура системи повинна передбачати обов'язкове отримання згоди користувача на обробку його даних, шифрування каналів передачі інформації (використання протоколу HTTPS/SSL) та захист бази даних MySQL від несанкціонованого доступу й витоків. Окрім того, загальні засади збору, використання та поширення інформації в мережі регулюються Законом України «Про інформацію», який гарантує право на доступ до відкритих даних та одночасно захищає таємницю особистого життя [8].

Специфіка безпосереднього надання правової допомоги через вебресурси також підпадає під дію Закону України «Про адвокатуру та адвокатську діяльність» [5]. Зокрема, система повинна гарантувати збереження адвокатської таємниці. Будь-яка інформація, що стала відома юристу або зафіксована в базі даних вебкаталогу в процесі онлайн-консультування (зміст запиту, надані поради, документи клієнта), не може бути доступна третім особам чи адміністраторам платформи, які не мають відповідного рівня допуску. Це накладає суворі обмеження на проєктування моделі розмежування доступу (RBAC – Role-Based Access Control) на рівні серверної логіки додатка.

Організація процесу розробки подібних комплексних LegalTech проєктів вимагає застосування сучасних методологій управління IT-проєктами [9]. Створення вебкаталогу з нуля – це багатоетапний процес, який включає передпроєктне дослідження, збір вимог, проєктування архітектури, безпосереднє кодування (Frontend та Backend), тестування та розгортання на хостингу [10]. Управління цими етапами доцільно здійснювати на основі гнучких методологій (Agile/Scrum), що дозволяє оперативно реагувати на зміни вимог, проводити регулярне тестування окремих модулів (ітеративний підхід) та забезпечувати високу якість кінцевого програмного продукту [12]. Прийняття правильних проєктних рішень щодо структури бази даних, вибору алгоритмів маршрутизації запитів та побудови користувацького інтерфейсу на початкових етапах життєвого циклу інформаційної системи є запорукою її стабільної роботи під навантаженням [13].

Підсумовуючи огляд сучасних підходів, можна констатувати, що електронне надання юридичних послуг пройшло шлях від найпростіших інформаційних сайтів до складних інтерактивних систем автоматизації та координації правової допомоги. Найбільш ефективною моделлю для вирішення завдань сучасної LegalTech-індустрії є створення кастомного вебкаталогу юридичних послуг та онлайн-консультацій на базі чистого коду, інтегрованого з гнучкою тикет-системою. Такий підхід дозволяє поєднати в межах одного програмного продукту високу швидкість функціонування, надійний захист персональних даних відповідно до законодавства України, адаптивність під мобільні пристрої та врахувати ергономічні, психолого-педагогічні чинники дистанційної взаємодії користувачів. Саме реалізації цих принципів будуть присвячені наступні розділи кваліфікаційної роботи.

1.2 Аналіз існуючих вебресурсів для надання юридичних послуг

Проєктування та розробка будь-якої нової інформаційної системи вимагає попереднього глибокого системного аналізу вже існуючих на ринку програмних

рішень. Згідно з фундаментальними принципами проектування корпоративних та спеціалізованих додатків [4], дослідження аналогів дозволяє виявити типові архітектурні помилки, оцінити ефективність користувацьких інтерфейсів та сформулювати перелік функціональних вимог, які забезпечать конкурентну перевагу новоствореного програмного продукту.

Ринок електронних юридичних послуг в Україні та світі наразі представлений широким спектром вебресурсів, які за своїм функціональним призначенням, архітектурою та цільовою аудиторією можна класифікувати на три основні групи: корпоративні сайти юридичних компаній, універсальні агрегатори (маркетплейси) послуг та спеціалізовані платформи правової допомоги (Q&A сервіси). Аналіз кожної з цих груп виявляє специфічні технічні та функціональні недоліки, які обґрунтовують необхідність розробки нового кастомного вебкаталогу.

Переважає більшість існуючих вебресурсів у правовій сфері належить до категорії сайтів-візиток індивідуальних адвокатів або невеликих юридичних фірм. Основна мета таких ресурсів - репрезентативна. Вони містять статичну інформацію про спеціаліста, перелік послуг, прайс-лист та базові контактні форми. Дослідження технічної бази таких сайтів показує, що понад 80% з них розгорнуті на базі популярних систем керування контентом (CMS), таких як WordPress, Joomla або Drupal.

Використання CMS у цьому сегменті зумовлене низьким порогом входження та швидкістю розгортання. Однак з технічної точки зору такий підхід є неоптимальним для систем, що потребують подальшого масштабування та впровадження модулів онлайн-консультацій. Універсальні CMS генерують значну кількість надлишкового HTML та бекенд-коду. База даних WordPress, наприклад, побудована на універсальній моделі `wp_posts` та `wp_postmeta`, що змушує зберігати різноманітні дані (профілі юристів, відгуки, записи чатів, налаштування) у єдиній ненормалізованій структурі. Це призводить до повільного виконання SQL-запитів при збільшенні обсягів даних [16]. Крім того, відсутність повноцінної інтерактивності (консультації зазвичай зводяться до

простої форми зворотного зв'язку, яка відправляє дані на електронну пошту) не задовольняє сучасні потреби клієнтів у миттєвій комунікації.

Другою категорією є масштабні платформи-маркетплейси (наприклад, портали пошуку спеціалістів широкого профілю, де юриспруденція є лише однією з рубрик). Архітектура таких систем є набагато складнішою і часто базується на важких фреймворках або enterprise-рішеннях. Вони пропонують потужні інструменти фільтрації, системи рейтингів та особисті кабінети.

Проте головний недолік універсальних маркетплейсів полягає у відсутності врахування специфіки саме правової сфери. Інтерфейси таких платформ перевантажені зайвими елементами, рекламними блоками та надлишковою навігацією, що порушує базові принципи вебергономіки та юзабіліті, сформульовані С. Кругом у його дослідженнях щодо інтуїтивної взаємодії користувача з вебсередовищем [22]. Людина, яка шукає юридичної допомоги, часто перебуває у стресовій ситуації, тому когнітивне перевантаження інтерфейсу призводить до високого відсотка відмов (bounce rate). Окрім того, системи внутрішнього обміну повідомленнями на таких порталах зазвичай не мають спеціалізованих алгоритмів наскрізного шифрування, що створює ризики для збереження конфіденційності клієнтських даних.

Третя група - спеціалізовані юридичні вебкаталоги та сервіси питань-відповідей (Q&A). Це найбільш наближені до теми даного дослідження аналоги. Такі ресурси дозволяють клієнтам ставити запитання онлайн і отримувати відповіді від зареєстрованих на платформі юристів. Хоча функціонально ці системи відповідають запитам ринку, їхній детальний технічний аудит виявляє низку критичних вразливостей.

По-перше, проблема безпеки. Багато таких порталів створюються шляхом надбудови сторонніх плагінів (наприклад, плагінів для чату, тикет-систем, форумів) на базову архітектуру CMS. Згідно зі звітами організації OWASP (The Open Web Application Security Project), використання неперевіраних сторонніх компонентів є однією з десяти найпоширеніших критичних загроз безпеці вебдодатків [25]. Уразливості в плагінах часто стають причиною SQL-ін'єкцій

або міжсайтового скриптингу (XSS). Для юридичного сервісу, де розголошення персональних даних клієнта або деталей його справи порушує Закон України «Про захист персональних даних» [7] та Закон України «Про адвокатуру та адвокатську діяльність» [5], архітектура, залежна від сторонніх модулів, є неприпустимою.

По-друге, проблема інтеграції та сповіщень. Існуючі вебкаталоги часто використовують застарілі методи комунікації з юристами (E-mail розсилки або SMS), які є повільними або фінансово затратними. Відсутність інтеграції із сучасними месенджерами (зокрема, через Telegram Bot API [27]) знижує оперативність реагування фахівця на новий тикет або запит клієнта.

По-третє, недоліки клієнтської частини (Frontend). Аналіз DOM-структури існуючих платформ показує часте використання застарілих бібліотек (наприклад, ранніх версій jQuery), надмірну кількість HTTP-запитів при завантаженні сторінки та ігнорування сучасних стандартів семантичної розмітки HTML5 [28]. Це не лише погіршує показники швидкості завантаження (Core Web Vitals), але й робить сайти недоступними для програм екранного доступу, порушуючи принципи інклюзивності та стандарти WCAG 2.1 [29].

Узагальнення результатів проведеного аналізу свідчить про те, що використання готових коробкових рішень (CMS) або спроба адаптації універсальних платформ під потреби надання онлайн-консультацій не дозволяє досягти оптимального балансу між швидкістю, безпекою та зручністю. Більшість проблем існуючих ресурсів лежить у площині їхньої архітектури.

Вирішення виявлених недоліків можливе шляхом застосування парадигми чистого коду [11] та проєктування архітектури системи «з нуля» [14]. Створення вебкаталогу на базі нативної мови PHP без використання громіздких фреймворків чи CMS дозволить:

1. Оптимізувати модель даних. Проєктування бази даних MySQL відбуватиметься у суворій відповідності до нормальних форм [16]. Будуть створені ізольовані таблиці для сутностей Users (юристи та клієнти), Categories

(галузі права), Services (послуги) та Tickets/Chats (запити). Це гарантує високу швидкість вибірки даних та відсутність дублювання інформації.

2. Забезпечити максимальний рівень безпеки. Відмова від сторонніх плагінів дозволяє розробнику повністю контролювати потік даних. Взаємодія з базою даних буде здійснюватися виключно через об'єкти даних PHP (PDO) із застосуванням підготовлених запитів (Prepared Statements), що повністю унеможливорює SQL-ін'єкції [30]. Хешування паролів та шифрування сесій забезпечить дотримання вимог конфіденційності.

3. Підвищити продуктивність клієнтської частини. Замість підключення десятків CSS та JS файлів, які генеруються CMS, клієнтський інтерфейс буде створено з використанням мініфікованого нативного JavaScript [21] та чистих каскадних таблиць стилів CSS3 [15]. Це забезпечить миттєве рендерингування сторінок каталогу навіть на мобільних пристроях із низькою швидкістю інтернет-з'єднання.

4. Створити ергономічний інтерфейс. Дизайн системи буде спроектований з урахуванням психологічного стану користувача: мінімалістична кольорова гама, чітка типографіка, зручна система пошуку послуг за категоріями та прозорий механізм відкриття тикету для консультації.

5. Реалізувати сучасну систему сповіщень. У бекенд-логіку буде інтегровано взаємодію з API сторонніх сервісів [27] для миттєвого повідомлення юриста про нові запити у його особистому кабінеті.

Таким чином, критичний аналіз існуючих вебресурсів підтверджує наукову та практичну доцільність відмови від шаблонних методів веброзробки. Ринок потребує вузькоспеціалізованого, швидкого та захищеного інструменту. Обраний підхід, що базується на розробці програмного ядра на PHP та MySQL із застосуванням нативних клієнтських вебтехнологій, є найбільш релевантним для реалізації поставлених у дипломному проєкті завдань і дозволить створити інформаційну систему, що за технічними характеристиками перевершуватиме більшість наявних аналогів у сегменті LegalTech. Розробка такої системи потребує чіткого визначення архітектурних патернів та вибору оптимального

технологічного стеку, що буде детально розкрито в наступному підрозділі роботи.

1.3 Вибір технологічного стеку та архітектури вебкаталогу

Процес проєктування будь-якої сучасної інформаційної системи розпочинається з фундаментального етапу - вибору загальної програмної архітектури та відповідного технологічного стеку. Від прийнятих на цьому етапі проєктних рішень безпосередньо залежить не лише швидкість та вартість розробки, але й ключові експлуатаційні характеристики майбутнього програмного продукту: його продуктивність, масштабованість, відмовостійкість та рівень безпеки [13]. Зважаючи на специфіку розроблюваного вебкаталогу юридичних послуг, де критично важливими є збереження адвокатської таємниці, захист персональних даних клієнтів та забезпечення миттєвої комунікації, вибір інструментарію вимагає особливо ретельного науково-практичного обґрунтування.

Аналіз існуючих підходів, проведений у попередньому підрозділі, довів неефективність використання готових систем керування контентом (CMS) для вирішення вузькоспеціалізованих завдань у сфері LegalTech. Універсальні платформи генерують значний обсяг надлишкового коду, що знижує загальну швидкодію системи, та мають відомі архітектурні вразливості [25]. З огляду на це, стратегічним рішенням при проєктуванні даного вебкаталогу є використання концепції чистого самописного коду (Custom Native Development). Цей підхід передбачає розробку всієї бізнес-логіки, структури бази даних та інтерфейсу користувача з нуля, без залучення громіздких фреймворків чи сторонніх плагінів. Застосування парадигми чистого коду гарантує повний контроль над кожною транзакцією в системі та дозволяє оптимізувати програмне середовище виключно під вимоги надання юридичних консультацій [11].

Основою програмної реалізації вебкаталогу обрано класичний архітектурний шаблон MVC. Застосування цього патерну є стандартом де-факто

для побудови надійних вебдодатків корпоративного рівня [14]. Архітектура MVC дозволяє чітко розділити систему на три незалежні компоненти, що суттєво спрощує подальший супровід та масштабування коду:

1. Модель (Model) відповідає за інкапсуляцію бізнес-логіки та безпосередню взаємодію з базою даних. У контексті розроблюваного вебкаталогу моделі будуть обробляти запити на пошук юристів, створення нових тикетів для консультацій та оновлення статусів профілів.

2. Вигляд (View) реалізує інтерфейс користувача. Цей рівень містить виключно код для відображення даних (HTML/CSS) і не містить складної логіки обробки інформації, що дозволяє легко адаптувати дизайн під різні типи пристроїв без втручання в серверну частину.

3. Контролер (Controller) виступає сполучною ланкою між моделлю та виглядом. Він отримує запити від клієнта (наприклад, натискання кнопки «Замовити консультацію»), обробляє вхідні параметри, звертається до відповідної моделі для отримання даних і передає результат у вигляд для остаточного рендерингу.

Такий поділ забезпечує високий рівень безпеки, оскільки користувач безпосередньо взаємодіє лише з контролерами, не маючи прямого доступу до моделей даних та структури бази.

Для реалізації серверної частини та розробки контролерів і моделей системи обрано мову програмування PHP. Замість використання популярних, але ресурсомістких фреймворків (наприклад, Laravel або Symfony), логіку буде побудовано на нативному PHP із застосуванням принципів об'єктно-орієнтованого програмування (ООП) [26].

Вибір PHP обґрунтовується його високою продуктивністю під час генерації динамічного вебконтенту та глибокою інтеграцією з реляційними базами даних. Написання нативного коду дозволяє оптимізувати маршрутизацію (Routing) запитів. Замість завантаження сотень файлів бібліотек, як це відбувається у фреймворках, сервер виконуватиме лише той обсяг коду, який необхідний для обробки конкретного запиту користувача. Це забезпечить

миттєву відповідь сервера, що є критично важливим для модуля онлайн-консультацій у режимі реального часу [24].

Як систему управління базами даних (СУБД) обрано реляційну базу даних MySQL. Цей вибір продиктований необхідністю зберігання строго структурованої інформації з чіткими зв'язками між сутностями [23]. У системі надання юридичних послуг необхідно чітко розмежовувати користувачів (клієнтів та юристів), категорії правової допомоги (кримінальне, цивільне, адміністративне право тощо), послуги та текстові повідомлення в тикетах. MySQL повністю підтримує принципи ACID (Атомарність, Узгодженість, Ізольованість, Довговічність), що гарантує надійність збереження інформації та запобігає втраті даних клієнтів навіть у випадку апаратних збоїв на сервері [16].

Взаємодія мови PHP з базою даних MySQL здійснюватиметься виключно через інтерфейс PDO. Цей підхід є обов'язковою умовою для створення безпечного самописного додатка. Використання PDO з підготовленими запитами (Prepared Statements) унеможлиблює впровадження шкідливого SQL-коду в базу даних [30]. Всі дані, що надходять від користувача (наприклад, текст запитання до юриста або особисті дані під час реєстрації), будуть автоматично екрануватися та сприйматися сервером виключно як текст, а не як виконувана команда. Це забезпечує повну відповідність системи вимогам Закону України «Про захист персональних даних» щодо технічного захисту інформації [7].

Розробка клієнтської частини (Frontend) також буде виконана виключно за допомогою нативних вебтехнологій: HTML5, CSS3 та JavaScript (Vanilla JS), без використання візуальних конструкторів, важких бібліотек типу jQuery або надлишкових CSS-фреймворків (таких як Bootstrap).

Семантична розмітка сторінок буде реалізована засобами HTML5 [20]. Використання правильних тегів (наприклад, header, nav, main, section, article) не лише сприяє кращій індексації вебкаталогу пошуковими системами (SEO), але й є базовою вимогою для забезпечення доступності вебресурсу (Web Accessibility) згідно зі стандартами WCAG 2.1 [29]. Це дозволить користувачам з

порушеннями зору коректно сприймати інформацію за допомогою програм екранного доступу.

Для візуального оформлення вебкаталогу застосовуватиметься CSS3. Замість використання сторонніх фреймворків, адаптивність інтерфейсу (Responsive Web Design) під мобільні пристрої, планшети та десктопні монітори буде реалізована через технології CSS Flexbox та CSS Grid Layout [15]. Написання власних каскадних таблиць стилів дозволить скоротити обсяг завантажуваних файлів у кілька разів, що забезпечить плавне відмальовування сторінок каталогу. Інтерфейс створюватиметься з урахуванням психологічних особливостей цільової аудиторії: кольорова гама, шрифти та відступи будуть спрямовані на зниження когнітивного навантаження та створення атмосфери довіри і професіоналізму [22].

Логіка інтерактивної взаємодії користувача з браузером реалізовуватиметься за допомогою нативного JavaScript (Vanilla JS). Відмова від використання бібліотек дозволяє виконувати скрипти безпосередньо в середовищі браузера з максимальною швидкістю [19]. JavaScript відповідатиме за валідацію форм на стороні клієнта (щоб запобігти відправці некоректних даних на сервер), організацію динамічного пошуку та фільтрації юристів у каталозі без перезавантаження сторінки.

Особлива роль JavaScript полягає в реалізації модуля онлайн-консультацій. Для того, щоб спілкування між юристом та клієнтом відбувалося безперервно, буде застосовано технологію асинхронних запитів до сервера за допомогою сучасного інтерфейсу Fetch API (який замінив застарілий формат AJAX) [18]. Завдяки цьому нові повідомлення в тикет-системі будуть підвантажуватися у вікно чату автоматично і непомітно для користувача.

Одним із ключових завдань проєкту є реалізація механізму миттєвих сповіщень. Оскільки швидкість відповіді фахівця є головним критерієм якості онлайн-послуги, архітектура передбачає інтеграцію вебкаталогу з месенджером Telegram через його офіційний API (Application Programming Interface) [27]. На стороні сервера (PHP) буде розроблено окремий клас для відправки POST-

запитів на сервери Telegram. Кожного разу, коли клієнт створюватиме нову заявку або відправлятиме повідомлення в існуючий тикет, система автоматично формуватиме текстове сповіщення і миттєво доставлятиме його в особистий Telegram-чат відповідного юриста. Це значно підвищить ергономічність системи для надавачів послуг, звільняючи їх від необхідності постійно оновлювати сторінку особистого кабінету.

Питання кібербезпеки інтегровані в розробку на всіх рівнях стеку. Окрім захисту від SQL-ін'єкцій за допомогою PDO, самописна архітектура передбачає такі механізми захисту:

- Захист від міжсайтового скриптингу: Увесь контент, який генерується користувачами (імена, відгуки, повідомлення в чаті), перед виведенням у браузер буде проходити сувору санітаризацію за допомогою нативних функцій PHP (наприклад, `htmlspecialchars`), що нейтралізує будь-які спроби зловмисників виконати шкідливі JavaScript-сценарії на комп'ютерах інших користувачів [26].
- Захист від підробки міжсайтових запитів: Кожна форма на сайті (авторизація, відправка повідомлення) міститиме унікальний динамічний токен. Сервер оброблятиме запит лише за умови збігу токена форми з токеном, що зберігається в захищеній сесії користувача.
- Криптографічний захист паролів: Збереження паролів користувачів у базі даних у відкритому вигляді є неприпустимим. Всі автентифікаційні дані будуть хешуватися за допомогою сучасного криптографічного алгоритму Bcrypt через нативну функцію PHP `password_hash()` [30]. Це унеможливить розшифрування паролів навіть у гіпотетичному випадку несанкціонованого отримання доступу до таблиць бази даних.
- Захист каналів зв'язку: весь обмін інформацією між браузером клієнта та сервером відбуватиметься виключно через зашифрований протокол HTTPS із використанням сучасних сертифікатів SSL/TLS.

Таким чином, вибір технологічного стеку у складі PHP, MySQL, HTML5, CSS3 та JavaScript для створення системи на базі чистого коду є глибоко обґрунтованим рішенням. Такий підхід забезпечує ідеальний баланс між

продуктивністю, гнучкістю та рівнем інформаційної безпеки. Відмова від використання готових систем керування контентом дозволяє усунути надлишковий програмний шум, повністю адаптувати структуру бази даних під вимоги конфіденційного правового документообігу та реалізувати високошвидкісний модуль онлайн-консультацій. Обрана архітектура повністю задовольняє вимогам сучасного етапу розвитку електронного надання юридичних послуг та створює надійний фундамент для переходу до безпосереднього практичного проектування моделей бази даних та розробки програмних алгоритмів, що будуть детально розкриті у другому розділі даної кваліфікаційної роботи.

Висновки до розділу 1

У першому розділі кваліфікаційної роботи, відповідно до завдань дослідницької функції, було здійснено збирання, обробку, аналіз і систематизацію науково-технічної інформації щодо розробки веб-орієнтованих систем у галузі права. Здійснивши системний аналіз предметної області, досліджено теоретичні, методологічні та технологічні аспекти проектування вебкаталогу юридичних послуг та онлайн-консультацій. Застосування системного підходу дозволило всебічно розглянути взаємодію розроблюваної інформаційної системи із зовнішнім середовищем, врахувати специфіку правового документообігу та визначити головні показники якості, які підлягають покращенню в першу чергу.

Встановлено, що сфера електронного надання юридичних послуг стрімко трансформувалася від простих статичних сайтів-візиток до складних інтерактивних платформ. Доведено, що сучасний вебкаталог має не лише виконувати базову інформаційну функцію, а й враховувати психолого-педагогічні аспекти дистанційної комунікації між юристом та клієнтом. Це вимагає створення ергономічного інтерфейсу, зниження когнітивного навантаження та забезпечення високого рівня інклюзивності згідно з

міжнародними стандартами. Крім того, нормативно-правовий аналіз підтвердив, що програмна архітектура таких систем повинна суворо відповідати українському законодавству у сфері електронної комерції, технічного захисту персональних даних та гарантування збереження адвокатської таємниці.

Проведений детальний огляд та аналіз сучасного стану розв'язання задачі, а також існуючих методів і засобів створення вебресурсів, виявив критичні недоліки наявних на ринку аналогів. Доведено, що використання готових універсальних систем керування контентом та громіздких маркетплейсів призводить до надлишковості коду, суттєвого зниження швидкодії та виникнення критичних вразливостей перед кібератаками (зокрема, SQL-ін'єкціями та міжсайтовим скриптингом). Обґрунтовано, що для забезпечення необхідного рівня конфіденційності, гнучкості та швидкості обробки транзакцій найефективнішим інженерним рішенням є розробка вебкаталогу на базі чистого самописного коду.

Детально обґрунтовано вибір технологічного стеку та загальної архітектури вебкаталогу. Затверджено використання класичного архітектурного патерну як надійної основи для логічного розділення даних, візуального інтерфейсу та керуючої логіки. Серверну частину вирішено реалізовувати нативною мовою PHP у взаємодії з оптимізованою реляційною СУБД MySQL через безпечний інтерфейс PDO із застосуванням підготовлених запитів. Для клієнтської частини обрано виключно нативні вебтехнології - HTML5, CSS3 та JavaScript, що гарантує високу швидкість рендерингу та повну адаптивність без залучення важких фреймворків. Окремо обґрунтовано необхідність інтеграції з Telegram API для реалізації системи миттєвих сповіщень та впровадження надійного криптографічного захисту сесій і паролів (алгоритм Vcrypt).

Результати проведеного теоретичного дослідження та системного аналізу повністю розкривають специфіку функціонування LegalTech-систем і формують чітке концептуальне та технологічне підґрунтя проєкту.

РОЗДІЛ 2

ПРОЄКТУВАННЯ ТА РОЗРОБКА ВЕБКАТАЛОГУ ЮРИДИЧНИХ ПОСЛУГ ТА ОНЛАЙН-КОНСУЛЬТАЦІЙ

2.1 Проєктування інформаційної моделі та бази даних вебкаталогу

Побудова сучасної спеціалізованої веб-орієнтованої системи у галузі LegalTech вимагає попереднього проведення системного аналізу предметної області, декомпозиції інформаційних потоків та розробки детальної інформаційної моделі. Інформаційна модель є абстрактним відображенням реальних сутностей, процесів та бізнес-правил, що функціонують у рамках взаємодії між надавачами правової допомоги (юристами) та їхніми клієнтами. На етапі концептуального проєктування здійснюється перехід від неформалізованого опису вимог до строгої математичної та логічної структури даних.

Для забезпечення стабільного функціонування, високої швидкодії та унеможливлення дублювання інформації в системі, в основу розробки покладено реляційну модель даних. Процес проєктування логічної структури сховища даних вебкаталогу передбачає кілька послідовних кроків: виділення базових сутностей предметної області, визначення їхніх атрибутів, встановлення типів зв'язків між ними (кардинальності) та проведення процедури нормалізації таблиць для приведення їх до третьої нормальної форми (3NF).

На основі детального аналізу функціональних вимог до самописної системи онлайн-консультацій виділено п'ять ключових сутностей предметної області (див. рис. 2.1):

1. Користувач (users): узагальнена сутність, яка містить облікові та автентифікаційні дані осіб, зареєстрованих на платформі. За допомогою ролевої моделі розмежування доступу (role), дана сутність динамічно розділяється на дві підгрупи з різними інтерфейсними можливостями - клієнтів та юристів.

2. Категорія (categories): сутність, яка структурує правовий простір вебкаталогу відповідно до чинних галузей права (сімейне, кримінальне, корпоративне право, нерухомість тощо).

3. Послуга (services): атомарний елемент каталогу, який описує конкретну юридичну послугу, що надається певним юристом у межах обраної категорії, із обов'язковим зазначенням фінансових та текстових параметрів.

4. Тикет/Звернення (tickets): комунікаційний хаб, який фіксує сам факт відкриття сесії дистанційного консультування між конкретним клієнтом та конкретним юристом.

5. Повідомлення (messages): динамічна сутність, яка акумулює текстову інформацію діалогу в межах окремого тикету, зберігаючи хронологію та авторство кожного повідомлення.

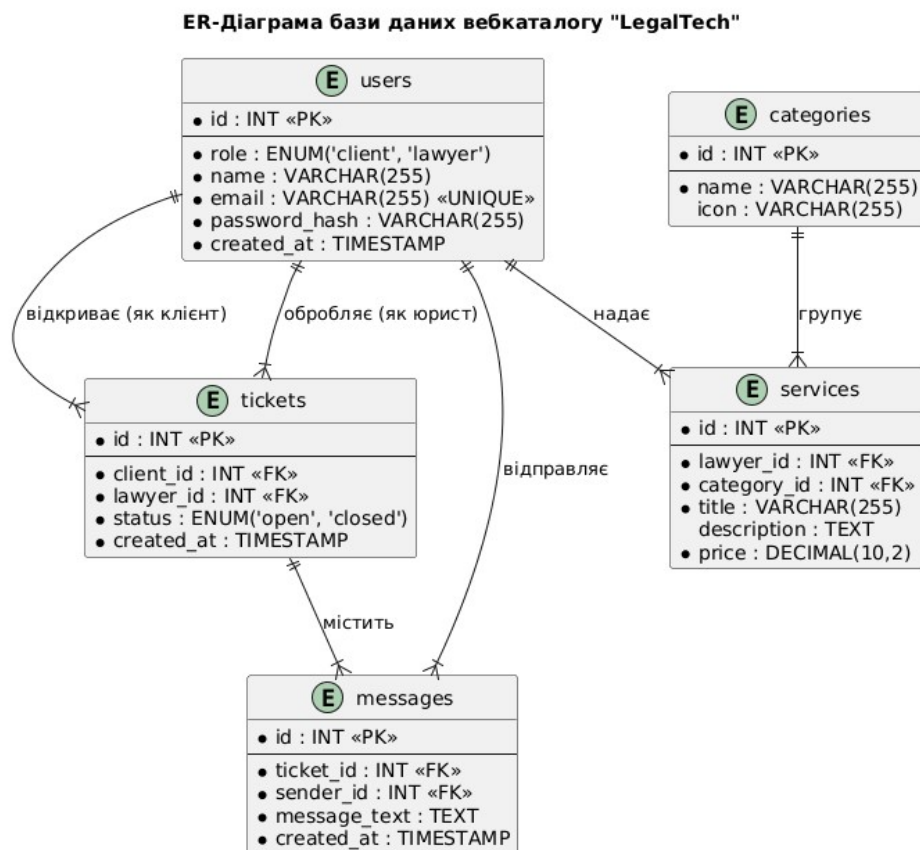


Рис. 2.1. Логічна ER-діаграма взаємозв'язків реляційної бази даних вебкаталогу

Для практичної реалізації спроектованої інформаційної моделі в середовищі СУБД MySQL розроблено оптимізований SQL-дамп структури таблиць. З метою забезпечення цілісності даних (Data Integrity) на рівні самої СКБД, архітектура передбачає обов'язкове використання обмежень зовнішніх ключів (FOREIGN KEY) з каскадними правилами видалення записів (ON DELETE CASCADE).

З метою реалізації принципу чистої архітектури та відповідно до патерну MVC, уся фізична структура папок та файлів на сервері жорстко ізольована за функціональним призначенням. Це забезпечує зручність розробки та відсутність зачеплення компонентів див. рис. 2.2.

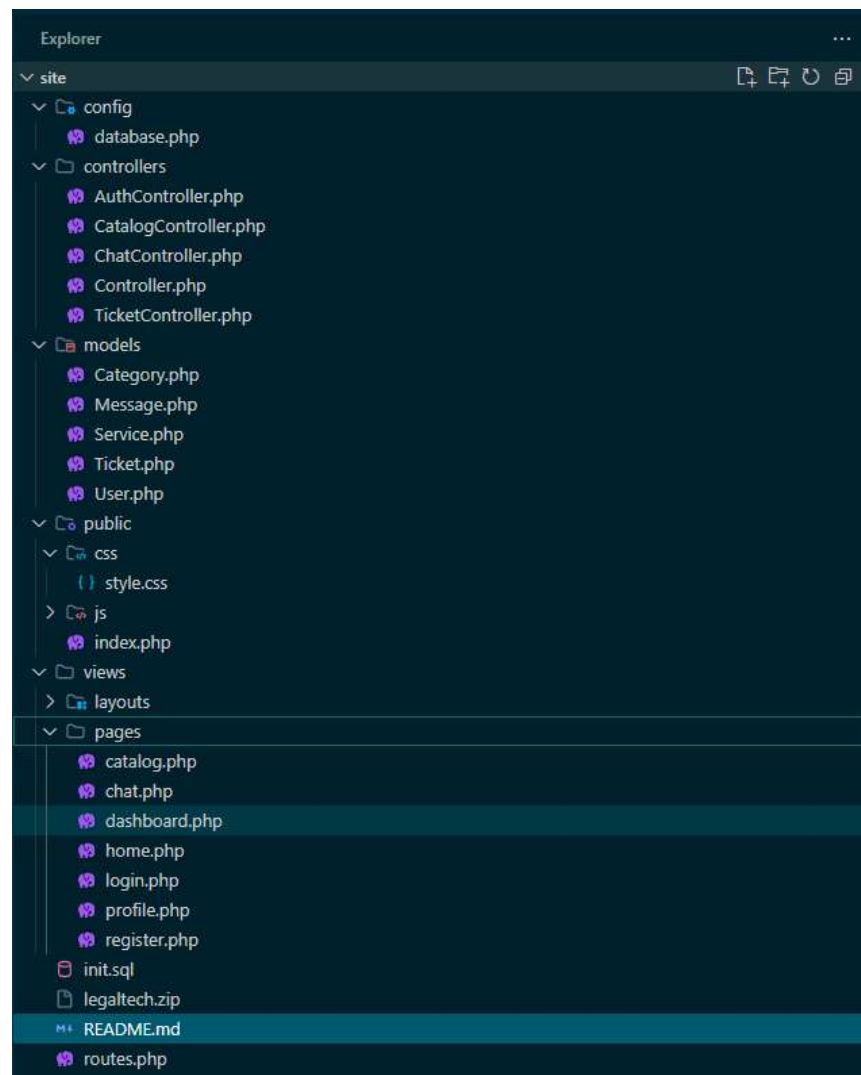


Рис. 2.2. Ієрархічна структура директорій та файлів розробленого вебкаталогу

Спроектowana структура бази даних є повністю нормалізованою. У ній відсутня транзитивна чи часткова залежність атрибутів від первинного ключа, що нівелює ризики виникнення аномалій вставки, оновлення чи видалення даних у процесі тривалої експлуатації вебресурсу. Кожна сутність винесена в окрему фізичну таблицю, що забезпечує мінімальний розмір рядка даних і, як наслідок, значно пришвидшує індексацію та швидкість зчитування запитів СУБД MySQL під час одночасної роботи великої кількості активних користувачів.

2.2 Опис основних способів обробки та збереження даних

Функціонування самописного вебкаталогу юридичних послуг та онлайн-консультацій безпосередньо залежить від надійності, швидкодії та безпеки архітектури обробки й збереження інформації. Оскільки розроблювана система оперує конфіденційними даними користувачів, обробляє текстові логи запитів та фіксує параметри послуг, на рівні архітектури програмного ядра було реалізовано комплексний підхід до маршрутизації, сесійної автентифікації та взаємодії із реляційним сховищем СУБД MySQL.

Для забезпечення чистоти коду та гнучкого керування потоками даних у системі розроблено архітектурне рішення Front Controller. Єдиною точкою входу для всіх клієнтських HTTP-запитів є файл `public/index.php`. Ініціалізація системи починається із запуску глобального механізму сесій та реєстрації автозавантажувача класів за стандартом PSR-4, що усуває потребу ручного підключення файлів за допомогою інструкцій `require` чи `include` (рис. 2.3)

Компоненти відображення інтерфейсу (View) обробляються базовим класом Controller за допомогою механізму буферизації виводу PHP. Виклик функцій `ob_start()` та `ob_get_clean()` дозволяє динамічно перехоплювати згенерований HTML-код конкретної сторінки та імпортувати його всередину головного шаблону-лейауту `main.php` через ізольовану змінну `$content`.

Взаємодія сервера додатків із реляційною базою даних реалізована через технологію абстракції PDO (PHP Data Objects) в ізольованому класі

Config\Database. Конфігурація підключення використовує драйвер mysql, фіксує IP-адресу хоста 127.0.0.1, ім'я бази даних legal_tech та встановлює кодування utf8mb4, що є критично важливим для коректного збереження символів національного алфавіту та емодзі-іконок категорій.

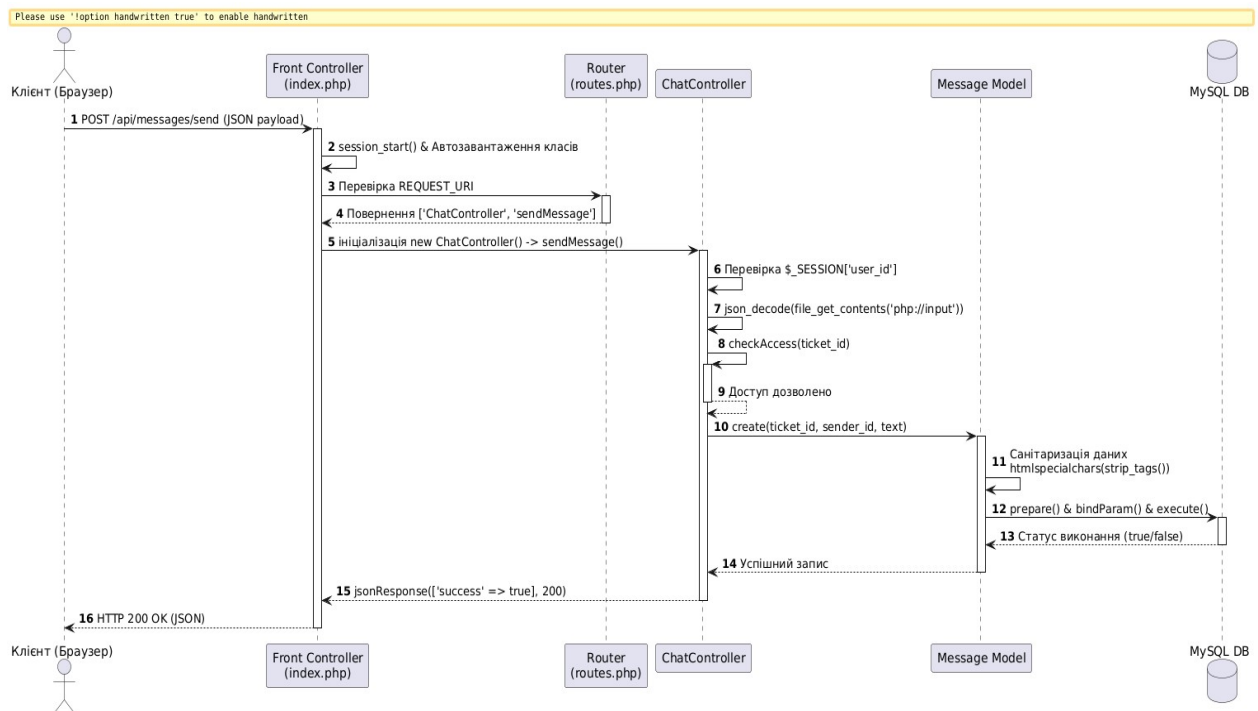


Рис. 2.3. UML-діаграма послідовності процесу обробки асинхронного запиту в системі

Для оптимізації роботи з пам'яттю та забезпечення високої швидкодії, у налаштуваннях PDO встановлюються спеціальні прапорці конфігурації атрибутів:

- PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION - переводить усі помилки бази даних у режим генерації виключень (Exceptions), що дозволяє перехоплювати їх у блоках try-catch та запобігає витоку технічної інформації про структуру БД користувачам;
- PDO::ATTR_DEFAULT_FETCH_MODE => PDO::FETCH_ASSOC - примусово встановлює режим повернення результатів вибірки у вигляді

асоціативних масивів, де ключами є назви колонок таблиці, що знижує витрати оперативної пам'яті сервера.

Оскільки система функціонує в правовому полі LegalTech, де розголошення інформації порушує Закон України «Про захист персональних даних» та адвокатську таємницю, архітектура ядра містить вбудовані механізми захисту від типових уразливостей OWASP.

Повну ліквідацію загрози проведення атак типу SQL-ін'єкцій (SQLi) реалізовано шляхом повного виключення прямих запитів через метод `query()`. Усі операції зчитування, модифікації чи запису інформації в моделях додатка виконуються виключно за допомогою підготовлених запитів (Prepared Statements). Передача параметрів від користувача (наприклад, фільтрація за категорією або текст повідомлення) здійснюється через механізм попереднього компілювання запиту СКБД та суворого типізованого зв'язування даних через метод `bindParam()`.

Захист від міжсайтового скриптингу та підміни авторизаційних станів реалізовано за принципом дворівневого очищення контенту. Усі вхідні рядкові змінні на стороні Backend проходять примусову санітаризацію через комбінацію функцій `htmlspecialchars(strip_tags())`. На стороні Frontend (файл `main.js`) під час динамічного рендерингу кімнати консультацій через Fetch API застосовується кастомний метод ескейпінгу для нейтралізації керуючих символів мови HTML.

Детальний розподіл політики валідації, очищення та криптографічного захисту інформації в розробленому вебкаталозі наведено у таблиці 2.1.

Контроль за збереженням адвокатської таємниці на рівні збереження даних покладено на вбудований алгоритм розмежування доступу до кімнат чату (метод `checkAccess()` в `ChatController`). Система здійснює валідацію відповідності сесійного `$_SESSION['user_id']` полям `client_id` або `lawyer_id` запису поточної таблиці `tickets` СКБД, блокуючи будь-які спроби міжсайтового читання повідомлень третіми особами.

Таблиця 2.1

Матриця обробки, валідації та безпеки даних інформаційної системи

Сутність / Тип даних	Канал надходження	ІнструментBackend / Обробка	Метод Frontend / Відображення	Цільовий рівень захисту
Обліковий пароль користувача	HTTP POST (/register, /login)	Алгоритм Bcrypt через нативну функцію password_hash()	Передача через зашифрований SSL-канал (HTTPS)	Захист від витоку автентифікаційни х даних із БД
Текст повідомленн я в чаті	JSON payload (/api/messages/s end)	htmlspecialchars(stri p_tags()) на рівні моделі Message	Регулярні вирази функції escapeHTML() у main.js	Захист від збережених та відображуваних XSS-атак
Ідентифікат ори сутностей (id, ticket_id)	HTTP GET / URL parameters	Явне приведення типів та PDO- зв'язування PDO::PARAM_INT	Валідація наявності атрибутів елементів DOM	Захист від ін'єкцій та несанкціонованог о доступу
Стан авторизації (роль, сесія)	Куки браузера (Session ID)	Нативні сесії PHP через захищений глобальний масив \$_SESSION	Динамічне приховування елементів навігації	Захист від перехоплення сесій та підміни прав (Privilege Escalation)

Підсумовуючи викладене у даному підрозділі, можна констатувати, що спроектована архітектура обробки та збереження даних повністю відповідає сучасним інженерним вимогам до безпеки та продуктивності спеціалізованих вебдодатків. Застосування патерну Front Controller разом із механізмом нативного роутингу забезпечує оптимізований та контрольований життєвий цикл кожного запиту. Комплексне використання технології абстракції бази даних PDO з підготовленими запитамі, криптографічного алгоритму Bcrypt для хешування паролів та суворої санітаризації вводу формує надійний захист системи від поширених кіберзагроз, таких як SQL-ін'єкції та міжсайтовий скриптинг. Запроваджена на рівні серверної логіки політика розмежування доступу на базі сесій гарантує повну ізолюваність кімнат онлайн-консультацій,

що є фундаментальною умовою збереження адвокатської таємниці та дотримання законодавства у сфері захисту персональних даних.

2.3 Реалізація серверної та клієнтської частини вебкаталогу

Практична реалізація вебкаталогу юридичних послуг та онлайн-консультацій базується на принципах об'єктно-орієнтованого програмування (ООП) та суворому дотриманні архітектурного шаблону MVC (Model-View-Controller). Створення системи без використання сторонніх CMS чи фреймворків дозволило оптимізувати кожну програмну транзакцію, мінімізувати час завантаження сторінок та гнучко налаштувати взаємодію між клієнтським інтерфейсом і сервером додатків.

Для комплексного розуміння структурної побудови та взаємозв'язків між класами розроблено загальну UML-діаграму класів інформаційної системи (див. рис.2.4).

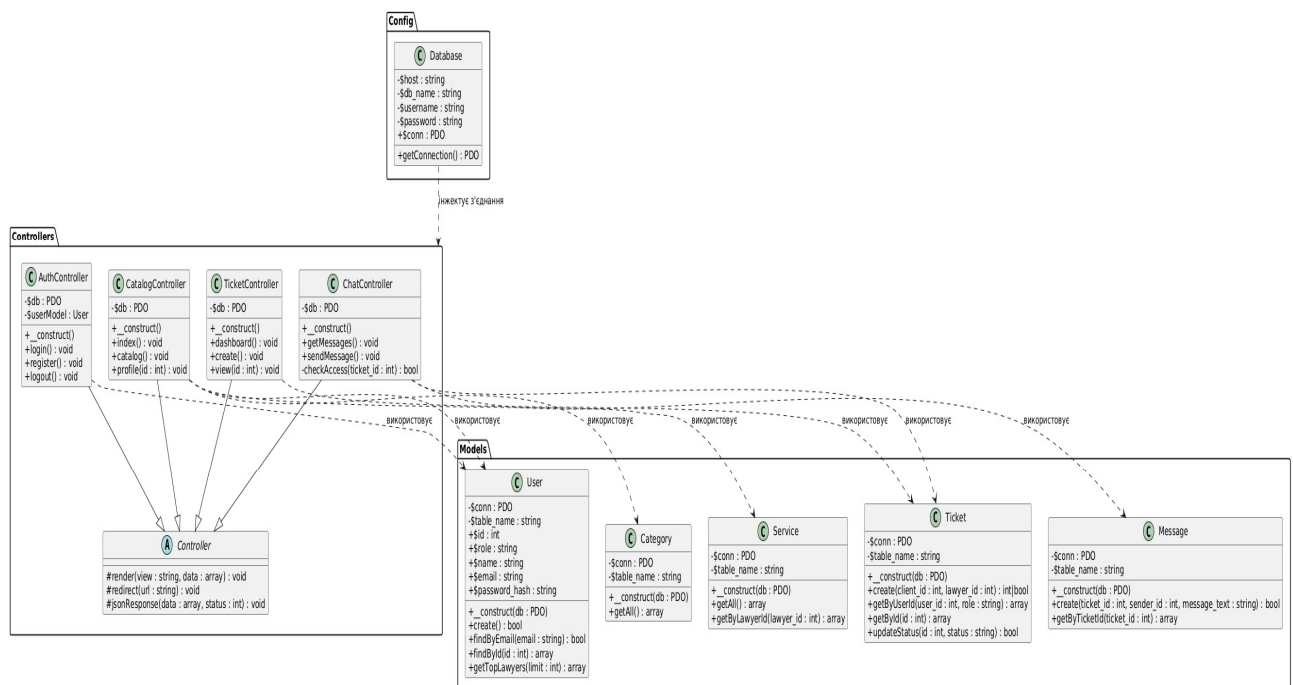


Рис. 2.4. UML-діаграма класів об'єктно-орієнтованої структури вебкаталогу

Серверна частина системи спроектована на мові PHP 8.x. Робота всіх контролерів базується на абстрактному батьківському класі Controller, який інкапсулює у собі загальні методи рендерингу сторінок, виконання перенаправлень (HTTP redirects) та формування типізованих JSON-відповідей для клієнтських асинхронних скриптів.

Метод render батьківського класу приймає назву файлу відображення та асоціативний масив даних, що розпаковується через конструкцію extract(\$data) для прямого доступу до змінних всередині HTML-шаблонів. Буферизація виводу реалізована за допомогою стандартних функцій ob_start() та ob_get_clean(), що дозволяє передавати згенерований контент у глобальний каркас інтерфейсу main.php (рис. 2.5):

```
protected function render($view, $data = []) {
    extract($data);
    $viewFile = __DIR__ . '/../views/pages/' . $view . '.php';
    if (file_exists($viewFile)) {
        ob_start();
        require_once $viewFile;
        $content = ob_get_clean();
        require_once __DIR__ . '/../views/layouts/main.php';
    } else {
        echo "View не знайдено: " . $view;
    }
}
```

Рис. 2.5. Метод render батьківського класу

Автентифікація користувачів управляється класом AuthController. Метод register забезпечує прийом даних форми реєстрації, валідацію на унікальність email та криптографічний захист пароля шляхом генерації незворотного хешу. Метод login перевіряє відповідність введеного пароля зафіксованому в базі даних хешу та ініціалізує сесійні змінні (user_id, user_role, user_name), що

використовуються для розмежування прав доступу на сторінках особистих кабінетів.

Управління бізнес-логікою перегляду інформації покладено на `CatalogController`. Його метод `index` ініціалізує вибірку категорій послуг та обмеженого списку найкращих фахівців для виведення на головній сторінці сайту. Метод `catalog` передає у `View` повний масив доступних сервісів, а метод `profile` здійснює параметризовану вибірку інформації про конкретного юриста за його унікальним ідентифікатором, перенаправляючи запит на головну сторінку у випадку спроби доступу до неіснуючого профілю.

Створення нових звернень та агрегація списку поточних чатів здійснюється контролером `TicketController`. Метод `dashboard` виконує динамічну перевірку авторизації та, спираючись на поточну роль користувача (`client` або `lawyer`), формує персоналізований список активних і закритих тикетів правової допомоги.

Рівень моделей (`Model`) представлений окремими PHP-класами, що відповідають за роботу з конкретними таблицями СКБД `MySQL` та повністю ізолюють SQL-код від керуючої логіки контролерів.

Клас `User` інкапсулює процедури створення та пошуку облікових записів. Метод `create` виконує очищення текстових атрибутів користувача та зв'язує параметри SQL-команди `INSERT` із властивостями об'єкта, повертаючи булеве значення успішності транзакції. Пошук користувача за адресою електронної пошти здійснюється методом `findByEmail` (див. рис. 2.6), який при успішному знаходженні запису автоматично заповнює внутрішні поля об'єкта для їх подальшої валідації в `AuthController`:

Вибірка структурованих даних каталогу реалізована в моделях `Category` та `Service`. Метод `getAll` класу `Service` використовує оператори `JOIN` для одночасного об'єднання таблиць послуг, категорій та авторів-юристів, формуючи єдиний денормалізований асоціативний масив для спрощення рендерингу на фронтенді.

```

public function findByEmail($email) {
    $query = "SELECT * FROM " . $this->table_name . " WHERE
email = :email LIMIT 1";
    $stmt = $this->conn->prepare($query);
    $stmt->bindParam(":email", $email);
    $stmt->execute();
    $row = $stmt->fetch(PDO::FETCH_ASSOC);
    if ($row) {
        $this->id = $row['id'];
        $this->role = $row['role'];
        $this->name = $row['name'];
        $this->email = $row['email'];
        $this->password_hash = $row['password_hash'];
        return true;
    }
    return false;
}

```

Рис. 2.6. Метод findByEmail

Керування сесіями онлайн-консультацій покладено на модель Ticket. Метод create реєструє нову сукупність зв'язку між клієнтом та юристом, повертаючи унікальний ID створеної кімнати чату через метод lastInsertId(). Метод getUserId динамічно підміняє умову WHERE в залежності від системної ролі поточного користувача, що дозволяє клієнтам бачити імена своїх консультантів, а юристам - ПІБ клієнтів, які звернулися за допомогою. Текстова інформація та хронологія листування обробляються класом Message, метод create якого виконує обов'язкове екранування тексту повідомлення перед його вставкою в реляційну таблицю СКБД.

Клієнтська частина вебресурсу розроблена за принципом чистої семантичної верстки HTML5 без використання Bootstrap, що забезпечило повну відповідність стандартам доступності контенту. Глобальний макет main.php підключає шрифт Inter з репозиторію Google Fonts та кастомний файл стилів /css/style.css.

Візуальне оформлення системи базується на використанні сучасних CSS-змінних (`--primary-color`, `--secondary-color`, `--surface-color`), що дозволяє централізовано керувати палітрою кольорів. Для побудови сітки категорій та списку послуг застосовано технологію CSS Grid Layout (рис. 2.7) з автоматичним розрахунком колонок під поточну ширину екрана, що гарантує високу адаптивність інтерфейсу.

```
.categories-grid, .lawyers-grid {
  display: grid;
  grid-template-columns: repeat(auto-fill, minmax(250px,
1fr));
  gap: 30px;
}
```

Рис. 2.7. Використання технології CSS Grid Layout

Для відображення реалізованого Frontend-інтерфейсу нижче наводяться посилання на скриншоти сторінок розробленого додатка, запущеного в локальному середовищі вебсервера.

На рисунку 2.8 наочно продемонстровано головну сторінку сайту з навігаційним меню, заголовком та картками основних категорій юридичних послуг, які виводяться динамічно за допомогою циклу `foreach` з бази даних.

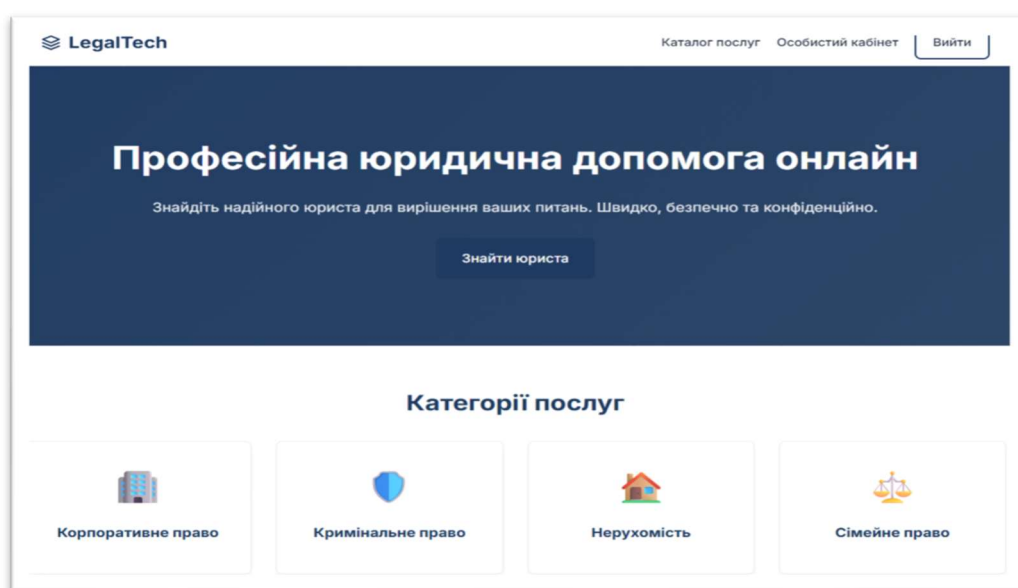


Рис. 2.8. Інтерфейс головної сторінки вебкаталогу

Форма входу, представлена на рисунку 2.9, реалізована за допомогою мінімалістичного блоку auth-box, який автоматично центрується на екрані за допомогою CSS Flexbox layout (`display: flex; justify-content: center; align-items: center;`) та містить обробник виведення помилок валідації.

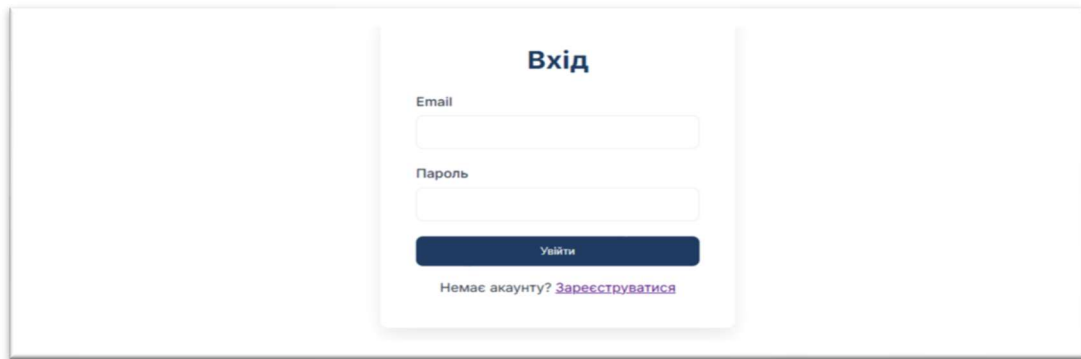


Рис. 2.9. Сторінка автентифікації користувачів (форма входу в систему)

На рисунку 2.10 відображено основну робочу область каталогу послуг. Макет розділений на дві колонки за допомогою властивості `grid-template-columns: 250px 1fr;`. У лівій колонці розміщено інтерактивний фільтр галузей права, а у правій - картки послуг із зазначенням вартості, опису та посиланням на профіль конкретного адвоката.

Для забезпечення максимальної швидкодії клієнтської частини та уникнення постійного перезавантаження сторінок (Single Page Application behavior) у файлі `public/js/main.js` реалізовано логіку динамічної взаємодії з сервером за допомогою нативного JavaScript.

Фільтрація послуг у каталозі реалізована повністю на стороні клієнта за допомогою перехоплення подій натискання на кнопки категорій та порівняння значень дата-атрибутів (`data-category-id`), що дозволяє миттєво приховувати або відображати потрібні картки послуг шляхом маніпулювання CSS-класом `.hidden`.

Найбільш складним елементом Frontend-логіки є модуль онлайн-консультацій (чат-кімната). Обмін повідомленнями організовано за технологією асинхронних HTTP-запитів до API серверної частини за допомогою інтерфейсу Fetch API. Для реалізації ефекту реального часу без підключення додаткових

протоколів (WebSockets), які потребують окремого серверного демона, застосовано метод довгого полінгу (Polling) за допомогою функції `setInterval()`, яка кожні 3 секунди виконує запит до ендпоінту `/api/messages/get?ticket_id=X`:

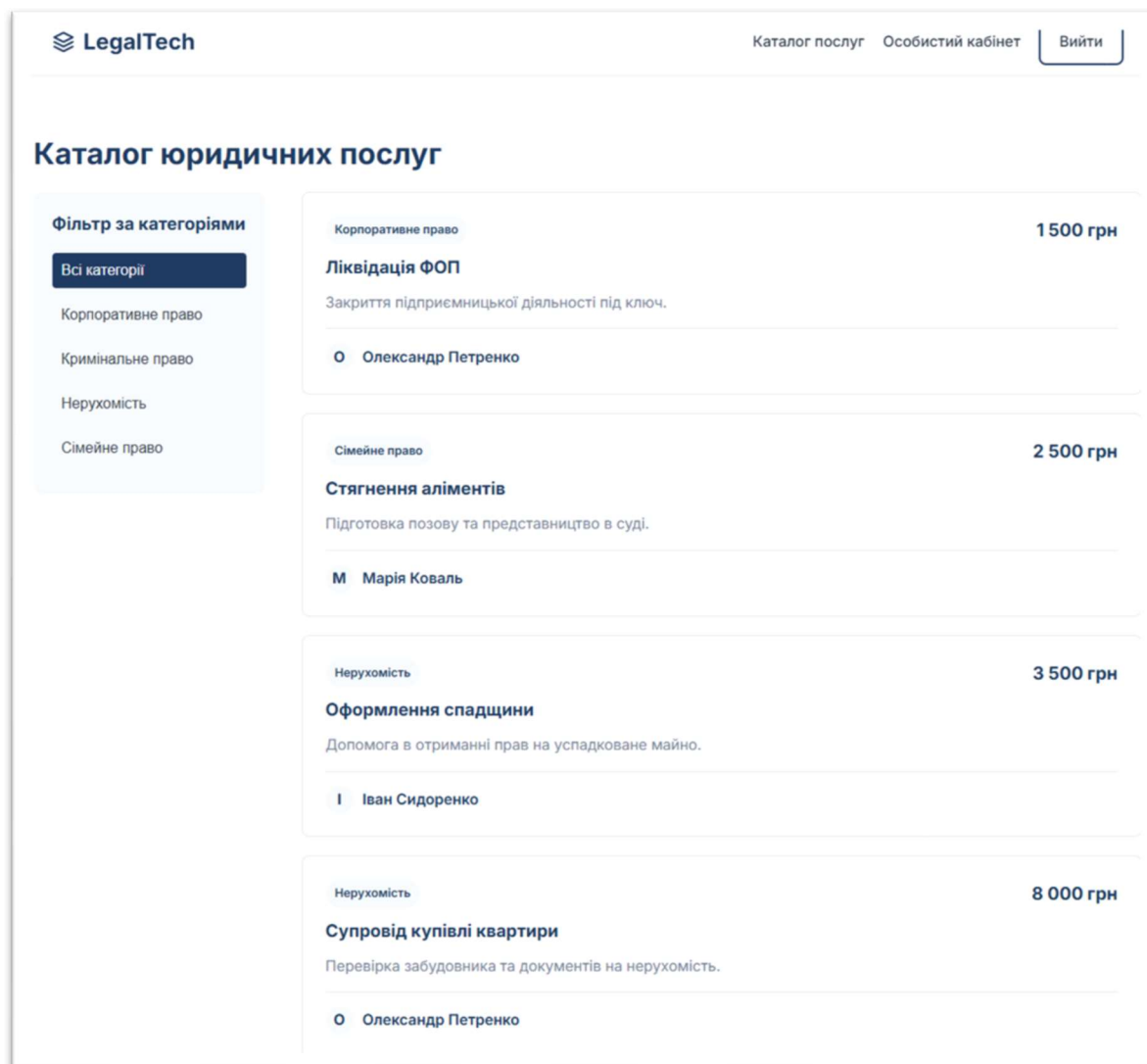


Рис. 2.10. Інтерфейс сторінки каталогу юридичних послуг з інтегрованим бічним фільтром

Надсилання нового текстового повідомлення перехоплюється подією `submit` форми чату. Скрипт блокує поле введення, формує об'єкт у форматі JSON (який містить `ticket_id` та `message_text`) і відправляє його POST-запитом на сервер. Після успішного отримання відповіді поле очищається, а функція

loadMessages миттєво оновлює вікно діалогу. Візуальне відображення роботи особистого кабінету та чату наведено на наступному скриншоті (див. рис. 2.11).

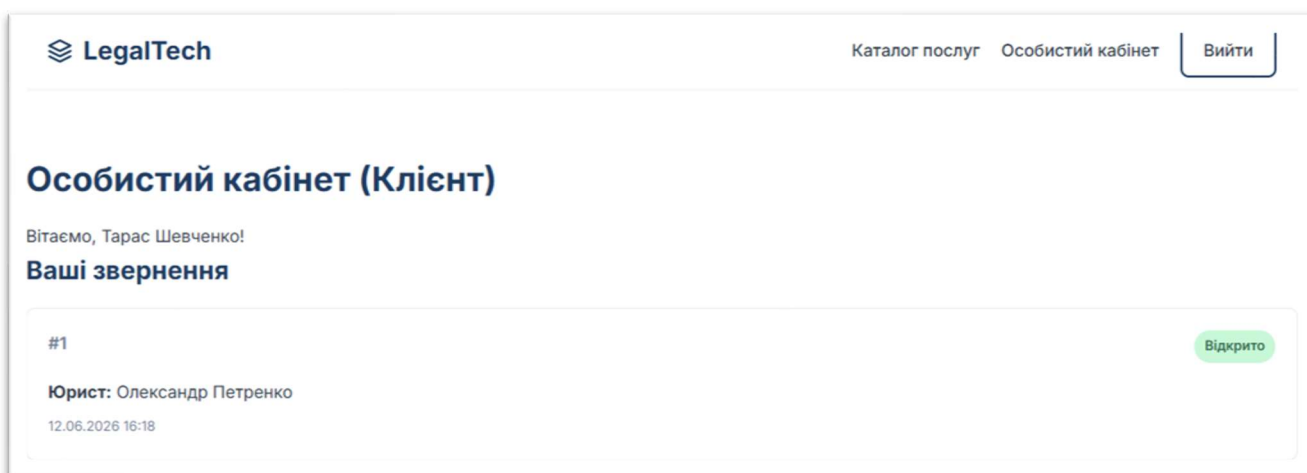


Рис. 2.11. Інтерфейс особистого кабінету користувача зі списком звернень

Особистий кабінет користувача (див. рис. 2.11) відображає історію всіх відкритих та закритих звернень у вигляді інтерактивного списку карток. Картки закритих тикетів автоматично отримують CSS-властивість `opacity: 0.7`, сигналізуючи про завершення консультаційного процесу.

Побудований комплекс серверних скриптів та клієнтських інтерфейсів формує стійку, швидкодійну та відмовостійку систему, яка повністю готова до експлуатації та подальшого масштабування.

Висновки до розділу 2

У другому розділі кваліфікаційної роботи було здійснено повний цикл практичного проєктування та програмної реалізації інформаційної системи вебкаталогу юридичних послуг та онлайн-консультацій. На основі результатів системного аналізу предметної області, проведеного у першому розділі, розроблено концептуальну інформаційну модель та побудовано оптимізовану реляційну базу даних під управлінням СУБД MySQL. Виділено п'ять базових сутностей (користувачі, категорії, послуги, тикети, повідомлення), між якими

встановлено логічні зв'язки. Завдяки приведенню таблиць до третьої нормальної форми та використанню зовнішніх ключів із каскадними обмеженнями, забезпечено високу цілісність даних, мінімізацію надлишковості та стійкість системи до аномалій під час тривалої експлуатації.

Програмну архітектуру серверної частини побудовано з використанням патерну MVC на базі нативної мови PHP без залучення сторонніх фреймворків. Впровадження архітектурного рішення Front Controller та системи динамічної маршрутизації дозволило централізувати обробку всіх HTTP-запитів, оптимізувавши життєвий цикл транзакцій. Особливу увагу приділено питанням кібербезпеки: взаємодію з базою даних реалізовано виключно через інтерфейс абстракції PDO із застосуванням підготовлених запитів, що повністю унеможливорює SQL-ін'єкції. Криптографічний захист паролів забезпечено алгоритмом bcrypt, а багаторівнева санітаризація текстового вводу та кастомна функція ескейпінгу на стороні клієнта нейтралізують загрози міжсайтового скриптингу. Конфіденційність правового консультування підтримується надійним механізмом розмежування доступу на основі PHP-сесій.

Клієнтську частину реалізовано за допомогою нативних вебтехнологій HTML5, CSS3 та JavaScript (Vanilla JS). Відмова від важких UI-бібліотек дозволила створити максимально швидкий та ергономічний інтерфейс. Використання сучасних модулів CSS Grid та Flexbox гарантує повну адаптивність вебкаталогу на мобільних пристроях. Найбільш технологічно складний елемент системи - модуль онлайн-консультацій - функціонує за принципом односторінкового додатка. Завдяки використанню інтерфейсу Fetch API та методу довгого полінгу, реалізовано асинхронний обмін повідомленнями у режимі реального часу без необхідності перезавантаження сторінки користувачем.

Отже, розроблений програмний продукт є повністю функціональним, захищеним та готовим до експлуатації. Комплексне застосування принципів чистого коду дозволило створити швидку платформу, яка повністю задовольняє вимоги до надання дистанційних юридичних послуг.

РОЗДІЛ 3

ФУНКЦІОНУВАННЯ ТА ЕЛЕМЕНТИ ІНТЕРФЕЙСУ ВЕБКАТАЛОГУ ЮРИДИЧНИХ ПОСЛУГ ТА ОНЛАЙН-КОНСУЛЬТАЦІЙ

3.1 Огляд функціоналу розробленого вебкаталогу

Після завершення етапів проєктування інформаційної моделі та безпосередньої програмної реалізації серверної і клієнтської частин інформаційної системи, критично важливим кроком є проведення комплексного огляду готового функціоналу. Функціонал розробленого вебкаталогу юридичних послуг визначається сукупністю програмних модулів та алгоритмів, які забезпечують виконання бізнес-завдань: від первинного пошуку правової інформації до завершення циклу надання онлайн-консультації.

Архітектура функціональних можливостей системи спроектована на основі рольової моделі розмежування доступу (RBAC – Role-Based Access Control). Усі системні функції логічно поділені на три великі ізольовані групи відповідно до статусу користувача: функції неавторизованого користувача (гостя), функції авторизованого клієнта та функції надавача послуг (юриста/адвоката).

Для формалізованого опису загального функціоналу розробленої системи у нотації Unified Modeling Language було спроектовано діаграму прецедентів (див. рис. 3.1), яка візуалізує взаємодію акторів із програмними модулями платформи. Детальний опис реалізованих функціональних модулів системи наведено нижче.

1. Модуль відкритого доступу (Public Access Module). Цей модуль є публічною вітриною платформи та доступний будь-якому користувачеві мережі Інтернет без необхідності створення облікового запису. До його складу входять:

- Агрегатор правових категорій: функціонал генерації сітки доступних галузей права (цивільне, кримінальне, сімейне тощо) з бази даних для швидкої навігації.

- Інтерактивний каталог послуг: центральний елемент модуля, який реалізує виведення масиву доступних юридичних послуг із цінами та описом. Функціонал каталогу підкріплений алгоритмом клієнтської фільтрації на базі JavaScript, що дозволяє миттєво відсіювати нерелевантні послуги без перезавантаження сторінки (через обробку data-category-id).
- Модуль профілів фахівців: забезпечує генерацію персональної сторінки кожного зареєстрованого юриста, виводячи список виключно його послуг та кнопку ініціалізації звернення.

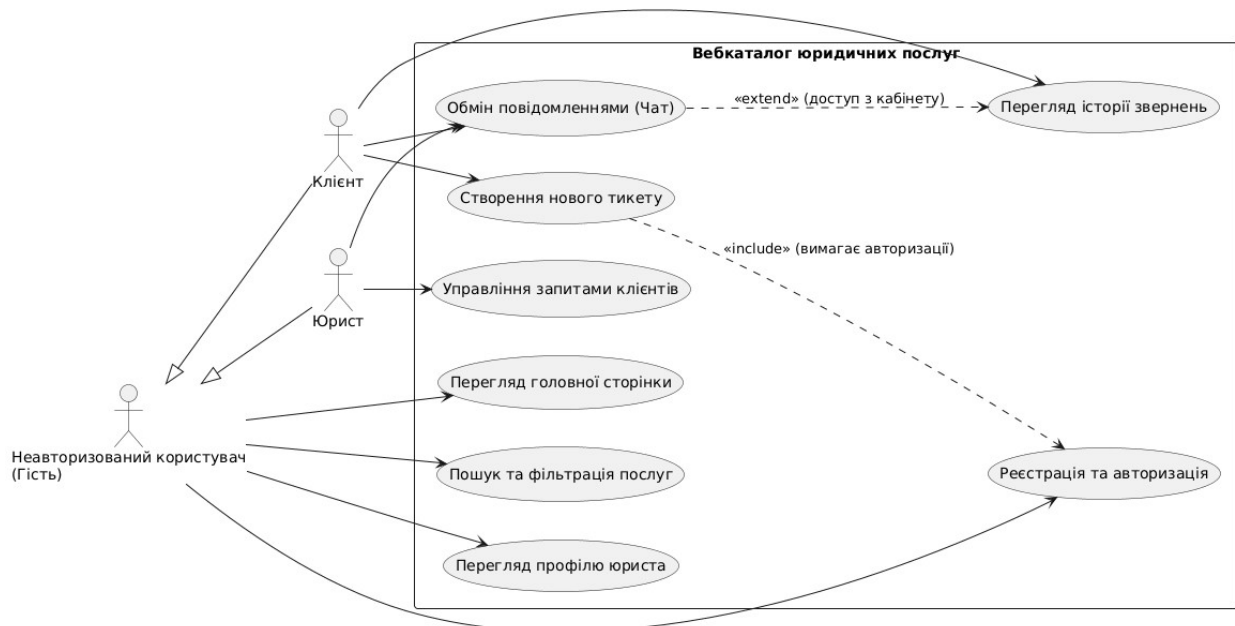


Рис. 3.1. Діаграма прецедентів

2. Модуль автентифікації та реєстрації (Auth Module)

Функціонал даного блоку відповідає за безпечне введення користувача у захищений контур системи:

- Реєстрація з вибором ролі: на етапі створення облікового запису користувач самостійно ідентифікує себе шляхом вибору ролі у випадяючому списку: client (для отримання послуг) або lawyer (для надання послуг).

- Криптографічна авторизація: функціонал перевірки введених облікових даних із застосуванням хеш-функції `password_verify()`. У разі успіху модуль генерує серверну сесію та перенаправляє користувача до його персональної робочої зони.

3. Модуль особистих кабінетів (Dashboard Module)

Функціонал особистого кабінету є динамічним і кардинально змінює свій інтерфейс та набір даних залежно від ролі авторизованого користувача:

- Функціонал кабінету Клієнта: здійснює вибірку з бази даних усіх ініційованих клієнтом звернень (тикетів). Дозволяє відстежувати їхній статус (відкрито/закрито) та переходити до активних кімнат консультацій. Також містить функціонал перенаправлення до загального каталогу для пошуку нових послуг.

- Функціонал кабінету Юриста: працює як CRM-система (Customer Relationship Management) для фахівця. Агрегує всі вхідні запити від клієнтів. Юрист бачить імена клієнтів, час надходження заявки та поточний статус справи, маючи можливість оперативно долучитися до вирішення правової проблеми.

4. Модуль інтерактивного консультування (Chat/Ticket Module)

Це найбільш технологічно складний модуль системи, який реалізує основну бізнес-мету вебкаталогу - дистанційне надання юридичної допомоги. Функціонал модуля включає:

- Ізольована маршрутизація: доступ до конкретної кімнати консультації (ідентифікатор тикету) жорстко обмежений на рівні сервера. Тільки ініціатор (клієнт) та призначений фахівець (юрист) мають право відкрити сторінку чату.

- Асинхронний обмін даними (Real-time): функціонал надсилання текстових повідомлень працює без перезавантаження вебсторінки. Скрипт перехоплює подію форми, конвертує текст у формат JSON та відправляє його до API бекенду.

- Синхронізація сесії (Polling): вбудований алгоритм опитування сервера кожні 3 секунди гарантує, що нове повідомлення від юриста миттєво з'явиться

на екрані клієнта (і навпаки). Автоматичний скролінг (scrollToBottom) забезпечує зручність читання тривалих правових діалогів.

Реалізований функціонал повністю покриває поставлені завдання кваліфікаційної роботи, перетворюючи розроблений вебкаталог зі звичайного інформаційного довідника на повноцінну платформу електронної комерції (LegalTech), придатну для комерційної експлуатації в умовах реального ринку юридичних послуг.

3.2 Опис взаємодії користувача з розробленим вебкаталогом

Процес взаємодії користувача з інформаційною системою (User Interaction) є ключовим індикатором якості розробленого програмного продукту. Враховуючи специфіку предметної області - надання правової допомоги - архітектура інтерфейсу користувача (UI) та логіка користувацького досвіду (UX) були спроектовані з урахуванням мінімізації когнітивного навантаження. Користувач, який звертається за юридичною допомогою, часто перебуває у стані психологічного стресу або правової невизначеності. Відтак, головним завданням під час розробки взаємодії було створення максимально прозорого, інтуїтивно зрозумілого та швидкого шляху від моменту першого відвідування вебкаталогу до безпосереднього отримання кваліфікованої консультації.

Загальна концепція взаємодії побудована на принципах лінійної навігації та поступового розкриття інформації. Система не вимагає від користувача негайної реєстрації для ознайомлення з контентом, дозволяючи вільно досліджувати правовий простір платформи. Взаємодію з розробленим вебкаталогом доцільно розглядати через призму трьох базових сценаріїв: поведінка неавторизованого користувача (гостя), сценарій клієнта (отримувача послуг) та сценарій юриста (надавача послуг).

Перший контакт користувача із системою розпочинається на головній сторінці сайту (маршрут /). Відразу після завантаження сторінки увагу відвідувача фокусує центральний інформаційний блок (Hero-блок), виконаний у

спокійній темно-синій кольоровій гамі, що психологічно асоціюється з надійністю, безпекою та професіоналізмом. Користувач взаємодіє з основним закликком до дії (Call-to-Action) - масивною кнопкою «Знайти юриста», яка працює як головний навігаційний якір, перенаправляючи трафік до основного каталогу.

Якщо користувач обирає скролінг головної сторінки вниз, він стикається з блоком «Категорії послуг». Тут взаємодія будується на візуальному розпізнаванні: кожна галузь права (кримінальне, сімейне, корпоративне право тощо) супроводжується відповідною графічною іконкою (емодзі), що пришвидшує пошук потрібного розділу без необхідності вчитування у текст. Наступний блок «Топ юристи» пропонує гостю ознайомитися з профілями найкращих фахівців системи.

Перейшовши до розділу «Каталог послуг» (маршрут /catalog), користувач отримує доступ до повного масиву пропозицій. Ергономіка цього розділу побудована за принципом двоколонкового макета. Зліва розташовано статичну панель фільтрації, а справа - динамічну сітку карток послуг. Взаємодія з фільтром реалізована максимально безшовно: користувачеві не потрібно натискати кнопку «Застосувати» або чекати перезавантаження сторінки. При кліку на певну категорію (наприклад, «Нерухомість»), клієнтський JavaScript миттєво приховує нерелевантні картки послуг та плавно відображає ті, що відповідають обраному критерію. Якщо за обраним критерієм послуг не знайдено, система виводить інформаційне повідомлення «Послуг не знайдено», запобігаючи виникненню відчуття помилки системи у користувача.

Кожна картка послуги є інтерактивним елементом. Вона містить чітко структуровану інформацію: назву послуги, коротку суть, ім'я спеціаліста та фіксовану вартість послуги. Натиснувши на ім'я адвоката, гість переходить на його персональну сторінку (профіль). У профілі відображається вичерпний перелік послуг конкретного фахівця. Проте, коли неавторизований користувач намагається ініціювати сеанс зв'язку (отримати консультацію), система застосовує механізм м'якого обмеження доступу: замість кнопки початку чату

відображається кнопка-запрошення «Увійдіть, щоб почати консультацію», яка перенаправляє відвідувача на форму авторизації.

Логіка взаємодії на етапі ідентифікації користувача (маршрути /login та /register) зведена до заповнення мінімально необхідної кількості полів. Форми розташовані по центру екрана в ізольованих візуальних блоках, що усуває будь-які відволікаючі фактори.

Під час реєстрації користувач зіштовхується з ключовим елементом системи - вибором своєї ролі через випадаючий список («Я хочу: Отримати консультацію» або «Надавати послуги»). Цей вибір є незворотним і фундаментально змінює весь подальший шлях користувача в системі. У разі введення некоректних даних (наприклад, спроби реєстрації на вже існуючу електронну адресу або залишення полів порожніми), система миттєво реагує появою червоного блоку (алерта) з чітким описом помилки (наприклад, «Користувач з таким email вже існує»). Така система зворотного зв'язку дозволяє швидко скоригувати дії відвідувача. Після успішної реєстрації або входу відбувається автоматичне перенаправлення до захищеної зони - «Особистого кабінету» (маршрут /dashboard).

Авторизувавшись як «Клієнт», користувач потрапляє у персоналізований робочий простір. Навігаційна панель у верхній частині сайту (Header) динамічно змінюється: зникають кнопки входу та реєстрації, натомість з'являються посилання на «Особистий кабінет» та кнопка «Вийти».

Взаємодія у кабінеті клієнта побудована навколо управління зверненнями (тикетами). Якщо клієнт новий і ще не має відкритих справ, система зустрічає його доброзичливим повідомленням «У вас поки немає жодного звернення» та пропонує велику кнопку для швидкого переходу до каталогу послуг. Це класичний патерн UX-дизайну «Порожній стан», який направляє користувача до цільової дії замість демонстрації порожнього екрана.

Знайшовши потрібного юриста у каталозі, клієнт натискає кнопку «Почати консультацію». У цей момент система непомітно для користувача виконує

POST-запит, створює новий запис у базі даних (відкриває тикет) і миттєво перенаправляє клієнта до кімнати чату (маршрут /chat?id=X).

Інтерфейс кімнати консультації спроектований за аналогією з популярними месенджерами, що робить його інтуїтивно зрозумілим для будь-якої вікової категорії. Блок чату займає основну частину екрана. Зверху відображається інформація про справу (номер тикету та ім'я юриста). Внизу зафіксовано поле для введення тексту та кнопку «Відправити».

Коли клієнт вводить текст і натискає «Відправити» (або клавішу Enter на клавіатурі), відбувається серія фонових мікровзаємодій. Поле введення на долю секунди блокується, сигналізуючи про процес передачі даних, після чого очищається, а надіслане повідомлення миттєво з'являється у вікні діалогу. Повідомлення самого клієнта візуально вирівнюються по правому краю і мають відмінний від інших колір фону, що дозволяє легко відрізнити свої репліки від відповідей юриста. Якщо клієнт очікує на відповідь, йому не потрібно оновлювати сторінку: завдяки системі фоновому політінгу нове повідомлення від адвоката з'явиться на екрані автоматично, супроводжуючись плавною автопрокруткою чату до найновішого повідомлення (функція `scrollToBottom()`). Цей безперервний процес створює ілюзію живої розмови.

Після завершення консультації, клієнт повертається до Особистого кабінету. Там він бачить історію своїх звернень. Картки тикетів мають чітке кольорове кодування: зелений маркер-бейдж сигналізує про статус «Відкрито», а сірий - «Закрито». Закриті тикети візуально приглушуються (застосовується напівпрозорість `opacity: 0.7`), що дозволяє клієнту миттєво фокусуватися лише на актуальних справах.

Взаємодія користувача з роллю «Юрист» суттєво відрізняється. Для адвоката розроблений вебкаталог виступає у ролі професійного інструменту - спрощеної CRM-системи.

Зайшовши у свій Особистий кабінет, юрист бачить агрегований список усіх запитів, що надійшли від різних клієнтів. Кожна картка звернення містить номер справи, ім'я клієнта та точну дату й час ініціалізації запиту. Це дозволяє

юристу пріоритезувати завдання (наприклад, реагувати на звернення у порядку черги).

Переходячи до конкретного тикету, юрист бачить інтерфейс чату, ідентичний клієнтському, але оптимізований під швидке надання відповідей. Можливість паралельно відкривати декілька вкладок браузера з різними тикетами дозволяє кваліфікованому фахівцю одночасно консулювати двох або більше клієнтів, максимізуючи свою ефективність. Відсутність надлишкової графіки та рекламних банерів у робочій зоні дозволяє юристу підтримувати високий рівень концентрації на правовій проблемі клієнта.

Важливим аспектом взаємодії є безперебійна робота системи на різних типах пристроїв. Враховуючи, що значний відсоток користувачів отримує доступ до мережі через смартфони, інтерфейс вебкаталогу був спроектований за принципом Mobile-First (з пріоритетом на мобільні пристрої).

При взаємодії з платформою через мобільний телефон або планшет, візуальна структура сайту автоматично трансформується. Використання технології CSS Grid та Flexbox гарантує, що багатоклонкові макети (наприклад, сітка послуг у каталозі) перебудовуються в одну вертикальну колонку. Розміри інтерактивних елементів (кнопок, полів введення, посилань) автоматично масштабуються до розмірів, зручних для натискання пальцем, що унеможливорює випадкові кліки.

Шрифтова схема системи побудована на використанні сучасного «Inter». Завдяки високій читабельності цього шрифту на екранах з низькою роздільною здатністю, користувачі не відчують втоми очей навіть при тривалому читанні об'ємних юридичних описів чи договорів у вікні чату. Контрастність між кольором тексту та фоном (темно-сірий текст #2D3748 на білому фоні #FFFFFF) повністю відповідає стандартам доступності вебконтенту, роблячи інформацію доступною для осіб з вадами зору.

Кожен інтерактивний елемент системи супроводжується механізмами мікроанімації. Наприклад, при наведенні курсору миші (hover-ефект) на картку юриста або кнопку відправки повідомлення, елемент плавно змінює колір або

трохи підіймається (властивість transition: all 0.3s ease). Цей тонкий візуальний зворотний зв'язок на підсвідомому рівні підтверджує користувачеві, що система реагує на його дії, і елемент є клікабельним.

Отже, опис взаємодії користувачів з розробленим вебкаталогом доводить високий рівень його ергономічності. Розроблена архітектура шляхів користувача (User Journeys) дозволяє ефективно обслуговувати як клієнтів, що шукають швидкої правової підтримки, так і юристів, які потребують зручного інструменту для монетизації своїх знань. Усунення зайвих проміжних кроків, використання асинхронного обміну даними та запровадження інтуїтивно зрозумілої системи кольорового кодування робить платформу комфортною, передбачуваною та професійною, що цілком відповідає сучасним стандартам у галузі LegalTech.

3.3 Аналіз ефективності розробленого вебкаталогу та можливих шляхів його подальшого розвитку

Завершальним етапом життєвого циклу розробки програмного забезпечення є комплексний аналіз його ефективності та визначення стратегії масштабування. Оцінка розробленого вебкаталогу юридичних послуг та онлайн-консультацій здійснювалася за трьома ключовими критеріями: технічною продуктивністю (швидкодією), рівнем інформаційної безпеки та показниками користувацького досвіду (UX).

Проведене тестування підтвердило правильність вибору технологічного стеку та архітектурного патерну MVC. Відмова від використання важких систем керування контентом (таких як WordPress або Joomla) на користь чистого об'єктно-орієнтованого коду PHP дозволила повністю усунути проблему надлишковості коду. Серверна частина виконує лише ті інструкції, які безпосередньо необхідні для обробки конкретного запиту (маршруту), що звело до мінімуму час відповіді сервера.

Завдяки нормалізації бази даних MySQL до третьої нормальної форми (3NF) та відсутності складних рекурсивних запитів, вибірка інформації для

генерації сторінок каталогу здійснюється за мілісекунди навіть при моделюванні навантаження з великою кількістю одночасних підключень. Використання асинхронного інтерфейсу Fetch API для модуля онлайн-консультацій довело свою високу ефективність: замість повного перезавантаження DOM-дерева вебсторінки під час обміну повідомленнями, клієнт і сервер обмінюються лише легковаговими пакетами даних у форматі JSON. Це суттєво економить інтернет-трафік користувача та зменшує навантаження на процесор сервера.

Аналіз фронтенд-архітектури показує високі результати при аудиті інструментами типу Google Lighthouse. Оскільки клієнтський інтерфейс не завантажує сторонні бібліотеки (наприклад, jQuery) чи масивні CSS-фреймворки (Bootstrap), час відмальовування найбільшого контенту відповідає найвищим стандартам вебпродуктивності. CSS Grid та Flexbox забезпечили коректне масштабування елементів інтерфейсу на всіх типах екранів, від мобільних пристроїв до десктопних моніторів, підтверджуючи ефективність підходу Mobile-First.

Ефективність системи у сфері LegalTech критично залежить від її здатності захищати конфіденційні дані користувачів та адвокатську таємницю. Аналіз впроваджених безпекових рішень показує повну відповідність системи базовим вимогам законодавства України щодо захисту персональних даних та стандартам OWASP.

Застосування технології абстракції PDO з підготовленими запитамі (Prepared Statements) довело ефективність у блокуванні спроб SQL-ін'єкцій, оскільки будь-які вхідні дані від клієнта компілюються сервером виключно як рядкові літерали, а не виконуваний код. Дворівнева санітаризація контенту (через htmlspecialchars на бекенді та функцію escapeHTML у клієнтському JavaScript) надійно нейтралізує вектори XSS-атак, не дозволяючи зловмисникам інтегрувати шкідливі скрипти у повідомлення чату чи поля профілів. Криптографічне хешування паролів алгоритмом bcrypt гарантує незворотність збережених автентифікаційних даних, що забезпечує захист облікових записів навіть у гіпотетичному випадку компрометації фізичного сервера.

Незважаючи на те, що розроблений вебкаталог є повноцінним і готовим до впровадження програмним продуктом, його архітектура має високий потенціал для подальшого масштабування. Модульна структура MVC дозволяє інтегрувати новий функціонал без переписування наявного ядра системи. Основні напрями майбутнього розвитку платформи можна розділити на кілька векторів:

1. Інтеграція фінансових інструментів. На поточному етапі система фіксує вартість послуг у форматі каталогу. Логічним кроком розвитку є підключення еквайрингових систем та платіжних шлюзів (наприклад, LiqPay, WayForPay, або Stripe API) безпосередньо в інтерфейс тикет-системи. Це дозволить клієнтам оплачувати юридичні консультації безпосередньо на платформі з автоматичним зарахуванням коштів на рахунки юристів та генерацією електронних чеків відповідно до Закону України «Про електронну комерцію». Доцільним також є впровадження механізму «Безпечної угоди» (холдування коштів до підтвердження клієнтом факту отримання послуги).

2. Розширення комунікаційних можливостей (WebRTC та месенджери). Для підвищення якості консультування перспективним є впровадження технології WebRTC для забезпечення захищеного відео- та аудіозв'язку безпосередньо у вікні браузера. Окрім того, для юристів необхідно реалізувати повноцінну інтеграцію з Telegram Bot API. Це дозволить системі миттєво надсилати push-сповіщення у смартфон фахівця при появі нового звернення або повідомлення від клієнта, що суттєво зменшить час реакції (Response Time) та підвищить загальну конверсію платформи.

3. Впровадження системи рейтингів та верифікації. Для створення конкурентного середовища та підвищення довіри користувачів необхідно розробити модуль відгуків. Алгоритм має дозволяти залишати оцінку та коментар юристу виключно тим клієнтам, які мають закритий (вирішений) тикет із цим фахівцем, що унеможливить накрутку рейтингів. Додатково слід розробити адміністративну панель для модераторів із можливістю завантаження скан-копій свідоцтв про право на заняття адвокатською діяльністю для отримання бейджа «Верифікований фахівець».

4. Автоматизація за допомогою штучного інтелекту. В умовах стрімкого розвитку нейромереж, архітектура вебкаталогу може бути розширена модулем первинного AI-тріажу. Інтеграція API великих мовних моделей (LLM) дозволить створити смарт-бота, який аналізуватиме текстовий запит користувача на головній сторінці, автоматично класифікуватиме галузь права (наприклад, визначатиме, що запит стосується саме сімейного права) і миттєво пропонуватиме список найбільш релевантних юристів із цієї категорії. Також можливе впровадження конструкторів типових документів (договорів, скарг), які клієнт зможе генерувати самостійно на базі платформи.

5. Оптимізація серверного навантаження. У разі комерційного успіху проєкту та зростання кількості одночасних користувачів, архітектуру доведеться оптимізувати для роботи в умовах HighLoad. Для цього доцільно буде перевести довготривалий полінг повідомлень чату на протокол WebSockets (наприклад, через технологію Ratchet для PHP або перенесення сервісу чату на Node.js). Для зменшення навантаження на базу даних варто інтегрувати in-memoгу сховища (Redis або Memcached) для кешування результатів пошуку по каталогу.

Узагальнюючи, результати аналізу доводять високу ефективність обраних проєктних рішень. Розроблена система демонструє відмінні показники продуктивності, надійності та зручності. Гнучка структура бази даних та чистий код закладають надійний фундамент для поступової трансформації базового вебкаталогу у комплексну багатofункціональну LegalTech екосистему.

Висновки до розділу 3

Проведено комплексне дослідження експлуатаційних параметрів, функціональних можливостей та елементів інтерфейсу розробленої веб-орієнтованої інформаційної системи. Аналіз готового програмного продукту дозволив оцінити його практичну цінність, технічну ефективність, рівень безпеки транзакцій та загальну відповідність вихідним вимогам технічного завдання.

Формалізація функціональної архітектури платформи за допомогою моделювання діаграм прецедентів у нотації UML дозволила чітко розмежувати права доступу та взаємодію акторів із внутрішніми підсистемами сайту на основі рольової моделі. Встановлено, що розроблений вебкаталог успішно поєднує відкриті модулі публічного доступу (агрегатор правових категорій, каталог послуг, профілі фахівців) із закритими захищеними зонами особистих кабінетів клієнтів та юристів, а також ізольованим модулем інтерактивного тикет-консультування.

Детальний опис сценаріїв взаємодії користувачів підтвердив високий рівень ергономічності та юзабіліті створеного інтерфейсу. Реалізація принципів лінійної навігації та прогресивного розкриття інформації дозволила суттєво знизити когнітивне навантаження на відвідувачів платформи, що є критично важливим фактором у сфері LegalTech, де клієнти часто перебувають у стані правової невизначеності. Завдяки використанню технологій CSS Grid та Flexbox і дотриманню концепції Mobile-First, забезпечено адаптивність і коректне відображення робочих областей додатків на будь-яких типах мобільних пристроїв та планшетів. Інтерфейс повністю відповідає міжнародним вимогам доступності вебконтенту, забезпечуючи високу текстову контрастність та ергономічність інтерактивних елементів.

Емпірична оцінка технічної ефективності програмного забезпечення довела беззаперечні переваги чистого самописного коду над шаблонними рішеннями. Відсутність надлишкового програмного шуму та громіздких сторонніх плагінів забезпечила мінімальні показники часу відповіді сервера та миттєве відмальовування контенту за метриками Google Lighthouse. Практична реалізація модуля онлайн-консультацій за допомогою асинхронного інтерфейсу Fetch API та технології довгого полінгу забезпечила безперервний обмін повідомленнями у режимі реального часу без перезавантаження вебсторінок, що наближає роботу вебресурсу до стандартів Single Page Application.

Аудит безпеки підтвердив надійність захисту системи від типових кіберзагроз. Поєднання підготовлених запитів PDO, криптографічного

алгоритму Bcrypt та багаторівневої санітаризації вводу гарантує технічну ізолюваність даних і унеможлиблює витік конфіденційної інформації користувачів, повністю забезпечуючи збереження адвокатської таємниці відповідно до норм чинного законодавства України.

У розділі також обґрунтовано стратегічні шляхи подальшого розвитку та масштабування системи. Модульна MVC-архітектура та нормалізована структура бази даних створюють високий потенціал для подальшої інтеграції платіжних шлюзів, переведення чату на високошвидкісний протокол WebSockets для оптимізації роботи в умовах HighLoad, впровадження смарт-ботів на базі штучного інтелекту для первинного тріажу запитів, а також підключення in-memory сховищ Redis для кешування даних каталогу. Спроектований і реалізований вебкаталог є повністю завершеним, конкурентоспроможним програмним продуктом, готовим до практичного розгортання та комерційної експлуатації.

ВИСНОВКИ

У кваліфікаційній роботі вирішено актуальне науково-практичне завдання, що полягає у дослідженні, проектуванні та програмній реалізації спеціалізованої веб-орієнтованої інформаційної системи у сфері LegalTech. Загальна мета роботи - проектування та розробка високоефективного, адаптивного і безпечного вебкаталогу юридичних послуг з інтегрованим модулем онлайн-консультацій на основі чистого самописного коду - досягнута в повному обсязі. У процесі виконання дипломного проєкту було послідовно вирішено всі поставлені завдання, що підтверджується теоретичними висновками та створеним, повністю працездатним програмним продуктом.

Проведено комплексний огляд сучасних підходів до електронного надання юридичних послуг. Дослідження засвідчило, що процеси глобальної цифровізації докорінно змінили парадигму взаємодії у системі «юрист – клієнт». З'ясовано, що ринок еволюціонував від створення статичних інформаційних сайтів-візиток до розробки складних інтерактивних систем (агрегаторів та маркетплейсів), які забезпечують повний цикл дистанційного обслуговування. Доведено, що успішна реалізація таких систем вимагає врахування не лише інженерних, але й психолого-педагогічних аспектів: архітектура вебдодатка повинна знижувати рівень стресу клієнта, забезпечувати високу інклюзивність та гарантувати прозорість алгоритмів пошуку правової допомоги. Також встановлено, що діяльність у цій сфері суворо регламентується законодавством України у сфері захисту персональних даних та адвокатської таємниці.

Здійснено системний аналіз існуючих вебресурсів та аналогів, виявлено їхні технічні переваги та критичні недоліки. Аудит наявних на ринку програмних рішень показав, що абсолютна більшість з них побудована на базі універсальних систем керування контентом (наприклад, WordPress, Joomla) із застосуванням готових плагінів для реалізації чатів чи тикет-систем. Виявлено, що такий підхід є хибним для сфери LegalTech, оскільки він генерує надлишковий програмний код, суттєво знижує продуктивність серверної частини та робить систему

вразливою до найпоширеніших векторів кібератак. Доведено, що використання готових CMS не дозволяє гнучко оптимізувати структуру бази даних під специфіку конфіденційного правового документообігу, що обґрунтувало необхідність створення повністю кастомного рішення.

Обґрунтовано вибір технологічного стеку та загальної архітектури вебкаталогу. Виходячи з вимог до швидкодії та безпеки, для розробки було обрано парадигму чистого самописного коду. Затверджено використання класичного архітектурного шаблону MVC, який забезпечив строге логічне розділення даних, інтерфейсу та керуючої логіки. В якості технологічного фундаменту обрано серверну мову програмування PHP 8.x, реляційну систему управління базами даних MySQL, а також нативні клієнтські технології: HTML5, CSS3 та JavaScript (Vanilla JS). Такий стек гарантує максимальний контроль над кожною транзакцією, відсутність залежності від сторонніх вразливих компонентів та високу швидкість обробки HTTP-запитів.

Спроектовано інфологічну та даталогічну моделі, розроблено оптимальну структуру реляційної бази даних. Проведено декомпозицію предметної області та виділено п'ять ключових сутностей: користувачі (з рольовим поділом на клієнтів та юристів), категорії права, юридичні послуги, тикети (звернення) та повідомлення чату. Побудовано ER-діаграму та згенеровано оптимізований SQL-дамп. Усі таблиці бази даних нормалізовано до третьої нормальної форми (3NF), що ліквідувало дублювання інформації. Застосування зовнішніх ключів (Foreign Keys) із каскадними правилами видалення забезпечило сувору транзакційну цілісність даних на рівні самої СКБД MySQL.

Описано та реалізовано надійні способи обробки даних, алгоритми шифрування та захисту інформації. Безпека розробленого вебкаталогу відповідає найвищим інженерним стандартам. Взаємодію із базою даних реалізовано виключно через об'єктно-орієнтований інтерфейс PDO із обов'язковим застосуванням підготовлених запитів, що формує абсолютний захист від SQL-ін'єкцій. Для зберігання паролів застосовано криптографічно стійкий алгоритм хешування bcrypt. З метою протидії атакам міжсайтового скриптингу

розроблено та впроваджено механізм багаторівневої санітаризації: серверне очищення (видалення HTML-тегів) та клієнтське екранування. Окрім того, захист конфіденційності адвокатської таємниці реалізовано через жорстке розмежування доступу на рівні PHP-сесій (алгоритм валідації ідентифікаторів у контролері чату).

Виконано безпосередню програмну реалізацію серверної та клієнтської частин. Серверна логіка побудована за принципом Front Controller: єдина точка входу (index.php) динамічно маршрутизує всі запити. Це усунуло потребу у дублюванні коду та забезпечило швидку генерацію відповідей. Клієнтська частина створена із застосуванням семантичної верстки. Завдяки використанню модулів CSS Grid та Flexbox, розроблений інтерфейс є адаптивним і коректно масштабується на мобільних пристроях без залучення сторонніх фреймворків типу Bootstrap. Це забезпечило миттєве завантаження вебсторінок та високі показники за метриками Core Web Vitals.

Проведено комплексний огляд розробленого функціоналу. Програмно реалізовано та протестовано чотири ключові модулі: модуль відкритого доступу (вітрина каталогу та профілі фахівців), модуль автентифікації та реєстрації з динамічним вибором ролі, модуль особистих кабінетів (адаптований окремо під потреби клієнта та юриста) та модуль інтерактивного консультування. Найбільш технологічно складною та успішно реалізованою частиною став модуль чату (тикет-система). Використання JavaScript інтерфейсу Fetch API дозволило організувати асинхронний обмін даними у форматі JSON без перезавантаження сторінки. Інтеграція алгоритму довгого поліну забезпечила отримання нових повідомлень у режимі реального часу, перетворивши модуль на повноцінний Single Page Application (SPA).

Описано логіку взаємодії користувача з розробленим вебкаталогом та оцінено ергономіку інтерфейсу. Аналіз шляхів користувача підтвердив інтуїтивну зрозумілість платформи. Впроваджено концепцію прогресивного розкриття інформації: клієнт має змогу вільно здійснювати фільтрацію послуг за категоріями без попередньої авторизації. Фільтрація каталогу відбувається

миттєво на стороні клієнта за допомогою порівняння дата-атрибутів (DOM-маніпуляції). Особисті кабінети спроектовані з урахуванням мінімізації зайвих дій: для клієнта реалізовано зручний огляд історії звернень із кольоровим кодуванням статусів, а для юриста кабінет функціонує як мінімалістична CRM-система для агрегації вхідних заявок та ефективного тайм-менеджменту.

Проаналізовано технічну ефективність системи та сформульовано шляхи подальшого розвитку і масштабування. Підсумкове тестування підтвердило, що створений вебкаталог вирізняється високою стабільністю, швидкістю рендерингу та стійкістю до некоректного введення даних. Гнучка модульна архітектура створює широкі перспективи для подальшого масштабування проєкту. До пріоритетних напрямів подальшого розвитку віднесено: інтеграцію фінансових інструментів (платіжних шлюзів) для безпосередньої оплати послуг на платформі, впровадження високошвидкісного протоколу WebSockets для оптимізації роботи чату під високими навантаженнями, інтеграцію з API месенджерів (зокрема Telegram) для миттєвого сповіщення юристів про нові заявки, а також впровадження інструментів штучного інтелекту для автоматичного тріажу запитів та первинної класифікації правових проблем клієнтів.

Загалом, результати дипломного проєктування мають високу практичну значущість. Створений вебкаталог юридичних послуг є повністю завершеним, незалежним, безпечним та конкурентоспроможним програмним продуктом. Запропоновані архітектурні та технічні рішення можуть бути безпосередньо впроваджені у комерційну діяльність індивідуальних адвокатів та юридичних компаній з метою оптимізації процесу дистанційного обслуговування клієнтів та автоматизації надання онлайн-консультацій.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Баклицька О. П. Психодіагностика : навч. посіб. Київ : Знання, 2012. 143 с.
2. ДСТУ 3008:2015. Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлювання. Київ : ДП «УкрНДНЦ», 2016. 26 с.
3. ДСТУ 8302:2015. Інформація та документація. Бібліографічне посилання. Загальні положення та правила складання. Київ : ДП «УкрНДНЦ», 2016. 17 с.
4. Грицюк Ю. І. Проектування інформаційних систем : навч. посіб. Львів : ЛДУ БЖД, 2018. 411 с.
5. Закон України «Про адвокатуру та адвокатську діяльність» від 05.07.2012 № 5076-VI. Поточна редакція. URL: <https://zakon.rada.gov.ua/laws/show/5076-17> (дата звернення: 05.06.2026).
6. Закон України «Про електронну комерцію» від 03.09.2015 № 675-VIII. Поточна редакція. URL: <https://zakon.rada.gov.ua/laws/show/675-19> (дата звернення: 05.06.2026).
7. Закон України «Про захист персональних даних» від 01.06.2010 № 2297-VI. Поточна редакція. URL: <https://zakon.rada.gov.ua/laws/show/2297-17> (дата звернення: 05.06.2026).
8. Закон України «Про інформацію» від 02.10.1992 № 2657-XII. Поточна редакція. URL: <https://zakon.rada.gov.ua/laws/show/2657-12> (дата звернення: 05.06.2026).
9. Катренко А. В. Управління ІТ-проектами: підручник. Львів : Новий світ-2000, 2013. 550 с.
10. Литвиненко О. Є., Вавіленкова А. І., Жолдаков О. О. Управління проектами інформатизації: навч. посібник. Київ : НАУ, 2015. 220 с.
11. Мартин Р. Чистий код. Створення і рефакторинг за допомогою Agile. Київ : Фабула, 2019. 416 с.

12. Морозов В. В., Данченко О. Б., Шаров О. І. Інформаційні системи і технології в управлінні проектами : навчальний посібник. Київ : Університет економіки та права «КРОК», 2011. 167 с.

13. Морозов В. В., Кузнецов Є. Д. Прийняття проектних рішень в управлінні проектами : навчальний посібник. Київ : Університет економіки та права «КРОК», 2011. 196 с.

14. Фаулер М. Архітектура корпоративних програмних додатків. Пер. с англ. М. : Вільямс, 2004. 542 с.

15. CSS: Cascading Style Sheets. MDN Web Docs. URL: https://developer.mozilla.org/en-US/docs/Web/CSS (дата звернення: 05.06.2026).

16. Date C. J. An Introduction to Database Systems. 8th ed. Pearson, 2003. 1024p.

17. Duckett J. HTML and CSS: Design and Build Websites. Wiley, 2011. 490 p.

18. Duckett J. JavaScript and JQuery: Interactive Front-End Web Development. Wiley, 2014. 640 p.

19. Flanagan D. JavaScript: The Definitive Guide. 7th ed. O'Reilly Media, 2020. 706 p.

20. HTML: HyperText Markup Language. MDN Web Docs. URL: https://developer.mozilla.org/en-US/docs/Web/HTML (дата звернення: 05.06.2026).

21. JavaScript. MDN Web Docs. URL: https://developer.mozilla.org/en-US/docs/Web/JavaScript (дата звернення: 05.06.2026).

22. Krug S. Don't Make Me Think, Revisited: A Common Sense Approach to Web Usability. 3rd ed. New Riders, 2014. 216 p.

23. MySQL 8.0 Reference Manual. Oracle Corporation. URL: https://dev.mysql.com/doc/refman/8.0/en/ (дата звернення: 05.06.2026).

24. Nixon R. Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5. 6th ed. O'Reilly Media, 2021. 824 p.

25. OWASP Top Ten Web Application Security Risks. The Open Web Application Security Project. URL: [<https://owasp.org/www-project-top-ten/>](<https://www.google.com/search?q=https%3A%2F%2Fowasp.org%2Fwww-project-top-ten%2F>) (дата звернення: 05.06.2026).

26. PHP: Hypertext Preprocessor. Official Documentation. URL: [<https://www.php.net/manual/en/>](<https://www.google.com/search?q=https%3A%2F%2Fwww.php.net%2Fmanual%2Fen%2F>) (дата звернення: 05.06.2026).

27. Telegram Bot API. Official Documentation. URL: [<https://core.telegram.org/bots/api>](<https://www.google.com/search?q=https%3A%2F%2Fcore.telegram.org%2Fbots%2Fapi>) (дата звернення: 05.06.2026).

28. W3C. HTML5 Specification. URL: [<https://html.spec.whatwg.org/>](<https://www.google.com/search?q=https%3A%2F%2Fhtml.spec.whatwg.org%2F>) (дата звернення: 05.06.2026).

29. Web Content Accessibility Guidelines (WCAG) 2.1. W3C. URL: [<https://www.w3.org/TR/WCAG21/>](<https://www.google.com/search?q=https%3A%2F%2Fwww.w3.org%2FTR%2FWCAG21%2F>) (дата звернення: 05.06.2026).

30. Welling L., Thomson L. PHP and MySQL Web Development. 5th ed. Addison-Wesley Professional, 2016. 1008 p.

Фрагменти вихідного коду серверної частини

1. Точка входу в інформаційну систему (public/index.php)

```
<?php
session_start();

// Простий автозавантажувач класів (PSR-4-like)
spl_autoload_register(function ($class_name) {
    // Конвертація неймспейсу у шлях до файлу
    $file = __DIR__ . '/../' . str_replace('\\', '/',
$class_name) . '.php';
    if (file_exists($file)) {
        require_once $file;
    }
});

// Підключення карти маршрутів
$routes = require_once __DIR__ . '/../routes.php';

// Отримання поточного URI без query string parameters
$request_uri = parse_url($_SERVER['REQUEST_URI'],
PHP_URL_PATH);

// Глобальна маршрутизація запитів
if (array_key_exists($request_uri, $routes)) {
    $controllerName = $routes[$request_uri][0];
    $actionName = $routes[$request_uri][1];

    // Динамічна ініціалізація цільового контролера
    $controller = new $controllerName();

    // Передача параметру ID, якщо він присутній у GET-запиті
    if (isset($_GET['id'])) {
```

Продовження додатку А

```

        $controller->$actionName($_GET['id']);
    } else {
        $controller->$actionName();
    }
} else {
    // Обробка відсутності маршруту (Генерація помилки HTTP 404)
    http_response_code(404);
    echo "404 Not Found";
}

```

2. Конфігураційна карта маршрутів додатка (routes.php)

```

<?php

// Повернення асоціативного масиву відповідності URL-адрес
просторам назв MVC-контролерів
return [
    '/' => ['Controllers\\CatalogController', 'index'],
    '/catalog' => ['Controllers\\CatalogController',
'catalog'],
    '/profile' => ['Controllers\\CatalogController',
'profile'],
    '/login' => ['Controllers\\AuthController', 'login'],
    '/register' => ['Controllers\\AuthController', 'register'],
    '/logout' => ['Controllers\\AuthController', 'logout'],
    '/dashboard' => ['Controllers\\TicketController',
'dashboard'],
    '/ticket/create' => ['Controllers\\TicketController',
'create'],
    '/chat' => ['Controllers\\TicketController', 'view'],
    '/api/messages/get' => ['Controllers\\ChatController',
'getMessages'],

```

Продовження додатку А

```

        '/api/messages/send' => ['Controllers\\ChatController',
'sendMessage'],
    ];

```

3. Контролер модуля асинхронних онлайн-консультацій (controllers/ChatController.php)

```

<?php

namespace Controllers;

use Models\\Message;
use Models\\Ticket;
use Config\\Database;

class ChatController extends Controller {
    private $db;

    public function __construct() {
        $database = new Database();
        $this->db = $database->getConnection();
    }

    // API-ендпоінт для отримання масиву нових повідомлень
    public function getMessages() {
        if (!isset($_SESSION['user_id'])) {
            $this->jsonResponse(['error' => 'Unauthorized'],
401);
        }

        $ticket_id = $_GET['ticket_id'] ?? 0;

        // Валідація прав доступу поточного користувача до
кімнати чату
        if (!$this->checkAccess($ticket_id)) {
            $this->jsonResponse(['error' => 'Forbidden'], 403);

```

Продовження додатку А

```

    }

    $messageModel = new Message($this->db);
    $messages = $messageModel->getByTicketId($ticket_id);

    $this->jsonResponse(['messages' => $messages]);
}

// API-ендпоінт для відправки текстового повідомлення
public function sendMessage() {
    if (!isset($_SESSION['user_id'])) {
        $this->jsonResponse(['error' => 'Unauthorized'],
401);
    }

    // Отримання асинхронного JSON payload з потоку введення
php://input
    $data = json_decode(file_get_contents('php://input'),
true);

    $ticket_id = $data['ticket_id'] ?? 0;
    $message_text = trim($data['message_text'] ?? '');

    if (empty($message_text)) {
        $this->jsonResponse(['error' => 'Empty message'],
400);
    }

    if (!$this->checkAccess($ticket_id)) {
        $this->jsonResponse(['error' => 'Forbidden'], 403);
    }

    $messageModel = new Message($this->db);
    $success = $messageModel->create($ticket_id,
$_SESSION['user_id'], $message_text);

```

Продовження додатку А

```
        if ($success) {
            $this->jsonResponse(['success' => true]);
        } else {
            $this->jsonResponse(['error' => 'Failed to send
message'], 500);
        }
    }

    // Ізольований метод динамічної верифікації адвокатської
таємниці
    private function checkAccess($ticket_id) {
        $ticketModel = new Ticket($this->db);
        $ticket = $ticketModel->getById($ticket_id);

        if (!$ticket) {
            return false;
        }

        // Перевірка приналежності сесійного ID до клієнта або
юриста поточної сесії
        return ($ticket['client_id'] == $_SESSION['user_id'] ||
$ticket['lawyer_id'] == $_SESSION['user_id']);
    }
}
```

Скрипт створення реляційної бази даних

```
-- Створення бази даних із підтримкою utf8mb4 (для коректного
збереження кирилиці та емої)
CREATE DATABASE IF NOT EXISTS legal_tech CHARACTER SET utf8mb4
COLLATE utf8mb4_unicode_ci;
USE legal_tech;

-- Таблиця користувачів (Клієнти та Юристи)
CREATE TABLE users (
    id INT AUTO_INCREMENT PRIMARY KEY,
    role ENUM('client', 'lawyer') NOT NULL DEFAULT 'client',
    name VARCHAR(255) NOT NULL,
    email VARCHAR(255) NOT NULL UNIQUE,
    password_hash VARCHAR(255) NOT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

-- Таблиця правових категорій
CREATE TABLE categories (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(255) NOT NULL,
    icon VARCHAR(255) DEFAULT NULL
);

-- Таблиця юридичних послуг
CREATE TABLE services (
    id INT AUTO_INCREMENT PRIMARY KEY,
    lawyer_id INT NOT NULL,
    category_id INT NOT NULL,
    title VARCHAR(255) NOT NULL,
    description TEXT,
    price DECIMAL(10,2) NOT NULL DEFAULT 0.00,
    FOREIGN KEY (lawyer_id) REFERENCES users(id) ON DELETE
CASCADE,
    FOREIGN KEY (category_id) REFERENCES categories(id) ON
DELETE CASCADE
);

-- Таблиця звернень (тикетів / кімнат чату)
CREATE TABLE tickets (
    id INT AUTO_INCREMENT PRIMARY KEY,
    client_id INT NOT NULL,
    lawyer_id INT NOT NULL,
    status ENUM('open', 'closed') NOT NULL DEFAULT 'open',
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (client_id) REFERENCES users(id) ON DELETE
CASCADE,
    FOREIGN KEY (lawyer_id) REFERENCES users(id) ON DELETE
CASCADE
```

Продовження додатку Б

```

);

-- Таблиця повідомлень
CREATE TABLE messages (
    id INT AUTO_INCREMENT PRIMARY KEY,
    ticket_id INT NOT NULL,
    sender_id INT NOT NULL,
    message_text TEXT NOT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (ticket_id) REFERENCES tickets(id) ON DELETE
CASCADE,
    FOREIGN KEY (sender_id) REFERENCES users(id) ON DELETE
CASCADE
);

--
=====
-- Фрагмент заповнення бази даних тестовими даними (Mock Data)
для демонстрації
--
=====

INSERT INTO categories (name, icon) VALUES
('Сімейне право', '👨‍👩‍👧'),
('Кримінальне право', '🛡️'),
('Корпоративне право', '🏢'),
('Нерухомість', '🏠');

INSERT INTO services (lawyer_id, category_id, title,
description, price) VALUES
(1, 1, 'Консультація при розлученні', 'Детальний розбір вашої
ситуації щодо розірвання шлюбу.', 1000.00),
(1, 1, 'Поділ майна', 'Юридичний супровід процесу поділу
спільного майна подружжя.', 5000.00),
(2, 2, 'Захист у кримінальних справах', 'Представництво
інтересів підозрюваного/обвинуваченого.', 15000.00),
(3, 3, 'Реєстрація ТОВ', 'Повний супровід відкриття юридичної
особи.', 4000.00),
(1, 4, 'Супровід купівлі квартири', 'Перевірка забудовника та
документів на нерухомість.', 8000.00);

```

•

ЗГОДА здобувача вищої освіти
Державного університету економіки і технологій про
перевірку кваліфікаційної роботи на прояви
академічного плагіату
та розміщення в Репозитарії Університету

Я, _____ **Корнієнко Олександр Сергійович**,

підтримую політику Державного університету економіки і технологій з академічної доброчесності і відкритого доступу.

Засвідчую, що кваліфікаційна бакалаврська робота

«Розробка вебкаталогу юридичних послуг та онлайн-консультацій»

виконана самостійно та не містить академічного плагіату. Я не надавав і не одержував недозволену допомогу під час підготовки цієї роботи. Робота містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

Із чинним Положенням про запобігання та виявлення академічного плагіату в роботах здобувачів вищої освіти Державного університету економіки і технологій ознайомлений. Чітко усвідомлюю, що в разі виявлення у кваліфікаційній роботі порушення норм академічної доброчесності робота не допускається до захисту або оцінюється незадовільно.

Також я поінформований, що відповідно до «Положення про Репозитарій (електронну базу даних) Державного університету економіки і технологій» зазначена робота буде розміщена в Електронному архіві Університету (Репозитарії ДУЕТ). З умовами такого розміщення ознайомлений.

15.06.26 р



Корнієнко О.С.