

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Західноукраїнський національний університет
Факультет комп'ютерних інформаційних технологій
Кафедра інформаційно-обчислювальних систем і управління

Прончук Дарина Сергіївна

**Інформаційна система надання онлайн консультацій для осіб, які
постраждали від війни / Information System for Providing Online Consultations
to War-Affected Individuals**

Спеціальність 122 – Комп'ютерні науки
Освітньо-професійна програма – Комп'ютерні науки

Кваліфікаційна робота

Виконала студентка групи КН-42
Д. С. Прончук

Науковий керівник:
к.т.н., доцент, О.Р. Осолінський

Кваліфікаційну роботу допущено до
захисту

«___»_____ 2026 р.

Завідувач кафедри

_____ Н.В. Дзюбановська

Тернопіль – 2026

Факультет комп'ютерних інформаційних технологій
Кафедра інформаційно-обчислювальних систем і управління
Освітній ступінь «бакалавр»
спеціальність 122 – Комп'ютерні науки
освітньо-професійна програма – Комп'ютерні науки

ЗАТВЕРДЖУЮ
В.о. завідувача кафедри
_____ Н.В. Дзюбановська
«_____» _____ 20__ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ
Прончук Дарина Сергіївна
(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи: Інформаційна система надання онлайн консультацій для осіб, які постраждали від війни / Information System for Providing Online Consultations to War-Affected Individuals

керівник роботи О.Р. Осолінський, к.т.н., доцент

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

затверджений наказом по університету від 01 грудня 2025 року № 894.

2. Строк подання студентом закінченої кваліфікаційної роботи 25 травня 2026 р.

3. Вихідні дані до кваліфікаційної роботи: завдання на кваліфікаційну роботу студента, наукові статті, технічна література.

4. Основні питання, які потрібно розробити:

- провести аналіз організації консультаційної допомоги особам, які постраждали від війни;
- провести аналіз існуючих платформ дистанційної консультаційної підтримки;
- сформулювати постановку задачі розробки інформаційної системи онлайн-консультацій;
- розробити архітектуру та функціональну структуру інформаційної системи онлайн-консультування;
- розробити алгоритми обробки звернень, запису та проведення онлайн-консультацій;
- розробити інформаційне забезпечення та модель бази даних системи;
- провести вибір та аналіз програмно-апаратних засобів для інформаційної системи;
- реалізувати модулі Django-застосунку онлайн-консультацій;
- провести тестування функціональних сценаріїв роботи системи;

5. Перелік графічного матеріалу в роботі:

- деталізована архітектура інформаційної системи онлайн-консультування;
- узагальнена схема основних процесів інформаційної системи;

6. Консультанти розділів кваліфікаційної роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання 01 грудня 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів кваліфікаційної роботи	Примітка
1	Огляд літературних джерел відповідно до предметної області та складання плану роботи	до 01.01.2026 р.	
2	Написання 1 розділу кваліфікаційної роботи	до 01.02.2026 р.	
3	Написання 2 розділу кваліфікаційної роботи	до 15.03.2026 р.	
4	Написання 3 розділу кваліфікаційної роботи	до 01.05.2026 р.	
5	Представлення попереднього варіанту кваліфікаційної роботи, перевірка та внесення змін керівником	до 15.05.2026 р.	
6	Опрацювання зауважень та представлення завершеного варіанту кваліфікаційної роботи. Підготовка супроводжуючих документів	до 25.05.2026 р.	
7	Перевірка кваліфікаційної роботи на оригінальність тексту	до 30.05.2026 р.	
8	Оформлення кваліфікаційної роботи та отримання допуску до захисту	до 10.06.2026 р.	
9	Подання кваліфікаційної роботи до захисту на засіданні атестаційної комісії	до 10.06.2026 р.	

Студент _____ Д. С. Прончук
(підпис) (прізвище та ініціали)

Керівник кваліфікаційної роботи _____ О.Р. Осолінський
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Кваліфікаційна робота на тему «Інформаційна система надання онлайн консультацій для осіб, які постраждали від війни» на здобуття освітнього ступеня «бакалавр» зі спеціальності 122 «Комп'ютерні науки» освітньої програми «Комп'ютерні науки» написана обсягом в 68 сторінок і містить 37 ілюстрацій, 1 таблицю, 4 додатки та 39 використаних джерел.

Метою кваліфікаційної роботи є розробка інформаційної системи надання онлайн-консультацій для осіб, які постраждали від війни.

В роботі використано методи системного аналізу, структурно-функціонального моделювання, логіко-алгоритмічного опису процесів, проектування баз даних, веб програмування та тестування програмного забезпечення.

В роботі було проаналізовано предметну область онлайн-консультування для осіб, які постраждали від війни, та розглянуто існуючі платформи дистанційної консультаційної підтримки. Запропоновано архітектуру інформаційної системи. Розроблено алгоритми створення звернення, первинної маршрутизації, підбору консультанта, запису на консультацію, проведення онлайн-сесії та завершення взаємодії. Побудовано інфологічну модель і ER-діаграму бази даних. Реалізовано модульний Django-застосунок із функціями авторизації, обробки звернень, бронювання консультацій, обміну повідомленнями, завантаження файлів і журналювання дій користувачів.

Практична цінність роботи полягає у створенні програмної основи для організації дистанційної консультаційної допомоги особам, які постраждали від війни.

Ключові слова: ІНФОРМАЦІЙНА СИСТЕМА, ОНЛАЙН-КОНСУЛЬТАЦІЇ, ДИСТАНЦІЙНА ДОПОМОГА, DJANGO, POSTGRESQL, BOOTSTRAP, ВЕБЗАСТОСУНОК, БАЗА ДАНИХ, ЗВЕРНЕННЯ, КОНСУЛЬТАЦІЯ, СПОВІЩЕННЯ, ЖУРНАЛ ПОДІЙ.

ANNOTATION

The qualification thesis on the topic “Information System for Providing Online Consultations to War-Affected Individuals” for obtaining the Bachelor’s degree in specialty 122 “Computer Science” of the educational program “Computer Science” consists of 68 pages and includes 37 illustrations, 1 table, 4 annexes, and 39 references.

The purpose of the qualification thesis is to develop an information system for providing online consultations to individuals affected by the war.

The work uses methods of system analysis, structural and functional modeling, logical and algorithmic description of processes, database design, web programming, and software testing.

The work analyzes the subject area of online consulting for individuals affected by the war and examines existing platforms for remote consultation support. The architecture of the information system is proposed. Algorithms for creating a request, primary routing, selecting a consultant, booking a consultation, conducting an online session, and completing the interaction are developed. An infological model and an ER diagram of the database are built. A modular Django application is implemented with functions for authorization, request processing, consultation booking, message exchange, file uploading, and logging user actions.

The practical value of the work lies in creating a software basis for organizing remote consultation support for individuals affected by the war.

Keywords: INFORMATION SYSTEM, ONLINE CONSULTATIONS, REMOTE SUPPORT, DJANGO, POSTGRESQL, BOOTSTRAP, WEB APPLICATION, DATABASE, REQUEST, CONSULTATION, NOTIFICATION, EVENT LOG.

ЗМІСТ

Вступ.....	7
1 Аналіз предметної області надання онлайн-консультацій особам, які постраждали від війни	10
1.1 Особливості організації консультаційної допомоги особам, які постраждали від війни	10
1.2 Аналіз існуючих платформ дистанційної консультаційної підтримки	13
1.3 Постановка задачі розробки інформаційної системи онлайн-консультацій .	18
2 Проектування архітектури, алгоритмічного та інформаційного забезпечення системи	20
2.1 Архітектура та функціональна структура інформаційної системи онлайн-консультавання.....	20
2.2 Алгоритми обробки звернень, запису та проведення онлайн-консультацій .	24
2.3 Інформаційне забезпечення та модель бази даних системи	30
3 Програмно-технологічна реалізація та тестування інформаційної системи.....	34
3.1 Вибір Та аналіз програмно-апаратних засобів для інформаційної системи ..	34
3.2 Реалізація модулів Django-застосунку онлайн-консультацій	37
3.3 Тестування функціональних сценаріїв роботи системи	46
Висновки	50
Список використаних джерел	52
Додаток А Деталізована архітектура інформаційної системи онлайн-консультавання.....	56
Додаток Б Узагальнена схема основних процесів інформаційної системи	57
Додаток В Код програмної реалізації інформаційної системи онлайн-консультацій	58
Додаток Г Апробація отриманих результатів	64

ВСТУП

Останні роки суттєво змінили способи отримання допомоги та комунікації між людьми і фахівцями. Значна частина послуг поступово переходить у дистанційний формат. Тому зростає потреба в інформаційних системах, які дають змогу організувати звернення, запис і подальшу взаємодію з фахівцем без особистого відвідування установи. Однією з таких сфер є надання онлайн-консультацій, яке стало важливим засобом підтримки населення в умовах кризових ситуацій, соціальної нестабільності та обмеженого доступу до очних послуг. Вебзастосунок у такому випадку може стати зручним каналом зв'язку, через який людина подає звернення, отримує відповідь і домовляється про консультацію незалежно від свого місця перебування.

В умовах війни потреба у доступній та безпечній консультативній допомозі для осіб, які постраждали від бойових дій, вимушеного переселення, втрати житла, джерел доходу або близьких людей, істотно зростає. Багато громадян стикаються не лише з матеріальними труднощами, а й із психологічним напруженням, тривогою та потребою в своєчасній інформаційній, соціальній або психологічній підтримці. За таких умов очне консультування не завжди є можливим, оскільки люди можуть перебувати в різних регіонах країни або за її межами, мати труднощі з пересуванням чи обмежений доступ до відповідних установ. Тому онлайн-консультування є доцільною формою організації допомоги постраждалим особам.

Інформаційна система надання онлайн-консультацій повинна підтримувати повний цикл взаємодії користувача із сервісом. До основних функцій такої системи належать реєстрація та авторизація, подання звернення, вибір напрямку допомоги, планування консультації, обмін повідомленнями та файлами, надсилання нагадувань і сповіщень, а також збереження історії взаємодії. Окреме значення мають конфіденційність персональних даних, розмежування прав доступу, можливість роботи за нестабільного інтернет-з'єднання та наявність засобів адміністрування і модерації. Це особливо важливо через чутливий характер

інформації, пов'язаної зі станом користувачів та обставинами їхнього звернення по допомогу.

Наявні засоби веб-розробки та хмарного розгортання дають змогу створювати такі сервіси без складної локальної інфраструктури, що особливо важливо для соціальних і волонтерських ініціатив. Такі системи дають змогу централізувати обробку звернень, підвищити оперативність надання консультаційних послуг, зменшити організаційне навантаження на персонал і зробити допомогу доступнішою для користувачів. Отже, розробка такої системи має практичне значення, тому що допомагає впорядкувати процес звернення по допомогу та зробити консультації доступнішими для постраждалих осіб.

Метою кваліфікаційної роботи є розробка інформаційної системи надання онлайн-консультацій для осіб, які постраждали від війни, що забезпечує створення звернень, підбір консультанта, запис на консультацію, проведення дистанційної взаємодії, обмін повідомленнями і файлами, формування сповіщень та адміністрування основних процесів.

Для досягнення поставленої мети необхідно виконати такі завдання:

1. Проаналізувати предметну область надання онлайн-консультацій для осіб, які постраждали від війни.

2. Розглянути існуючі рішення дистанційної консультаційної підтримки та визначити їхні переваги й обмеження.

3. Сформулювати постановку задачі та визначити основні функціональні вимоги до інформаційної системи.

4. Розробити загальну структуру системи та визначити її основні функціональні модулі.

5. Побудувати алгоритмічне забезпечення для створення звернення, маршрутизації запиту, підбору консультанта, запису та проведення консультації.

6. Розробити інформаційне забезпечення системи, визначити основні сутності предметної області та побудувати ER-діаграму бази даних.

7. Обґрунтувати вибір програмно-технологічних засобів для реалізації системи.

8. Реалізувати основні модулі вебзастосунку, зокрема модулі користувачів, звернень, консультацій, розкладу, повідомлень, файлів, сповіщень та журналювання подій.

9. Провести тестування основних сценаріїв роботи системи та перевірити коректність її функціонування.

Об'єктом дослідження є процес організації дистанційного надання консультаційної допомоги особам, які постраждали від війни.

Предметом дослідження є методи, моделі та програмні засоби побудови інформаційної системи онлайн-консультацій з підтримкою користувацьких ролей, обробки звернень, планування консультацій, обміну повідомленнями, роботи з файлами та журналювання подій.

В роботі використано системний аналіз, моделювання інформаційних процесів, опис алгоритмів роботи системи, проектування бази даних, вебпрограмування та тестування програмного забезпечення. Для реалізації системи використано фреймворк Django, систему керування базами даних PostgreSQL, шаблонний механізм HTML, CSS і Bootstrap для побудови інтерфейсу, а також засоби інтеграції із сервісами відеозв'язку та хмарного розгортання.

Результати кваліфікаційної роботи апробовані та опубліковані в матеріалах XI Міжнародної наукової конференції «Традиційні та інноваційні підходи до наукових досліджень» (м. Луцьк, Україна).

Практичне значення роботи полягає у створенні програмної основи для інформаційної системи, яка може бути використана для організації дистанційних консультацій, планування зустрічей, фіксації історії взаємодії між користувачем і консультантом та адміністрування консультаційного процесу. Запропонована система може бути адаптована для установ, які надають підтримку людям, постраждалим від війни.

Кваліфікаційна робота складається із вступу, трьох розділів, висновків, списку використаних джерел та додатків.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ НАДАННЯ ОНЛАЙН-КОНСУЛЬТАЦІЙ ОСОБАМ, ЯКІ ПОСТРАЖДАЛИ ВІД ВІЙНИ

1.1 Особливості організації консультаційної допомоги особам, які постраждали від війни

Під час організації допомоги людям, які постраждали від війни, однією з перших проблем є не сама консультація, а можливість швидко й безпечно до неї дістатися. Людина відчуючи гострий стрес, тривогу, виснаження, втрату контролю, порушення сну, або має складні переживання після травматичних подій, не має зрозумілого і безпечного способу швидко отримати фахову підтримку. Таке трапляється коли частина людей переміщена або можуть жити в умовах небезпеки чи нестабільного транспорту, також великою проблемою є страх та неохочість розкривати особисті дані, а інколи людина просто не має ресурсу ходити по установах і пояснювати свою історію кільком різним спеціалістам. В гуманітарній практиці це описується як потреба в організованій системі психічного здоров'я та психосоціальної підтримки, де є кілька рівнів допомоги та зрозумілий маршрут клієнта від першого контакту до потрібного формату підтримки [1]. В таких умовах онлайн-консультування можна розглядати як один з основних каналів допомоги, оскільки воно зменшує бар'єри звернення та підтримує зв'язок між людиною і фахівцем навіть за складних обставин. Водночас онлайн-формат потребує чіткої організації: необхідно захищати персональні дані, визначати межі відповідальності фахівців, передбачати дії у кризових ситуаціях і контролювати, щоб звернення не втрачалися в процесі обробки [2].

Тому організація психологічної допомоги в умовах війни вимагає надійної структури, де професійна етика поєднується з технологічною безпекою. Коли така підтримка переходить в онлайн-формат вона включає не лише розмову з психологом, а й організований процес прийому звернення, первинної оцінки, підбору спеціаліста, планування консультацій, супроводу і завершення випадку з дотриманням етичних та правових норм [4]. Робота такого сервісу ґрунтується на взаємодії трьох ролей: клієнта, адміністратора та консультанта. Кожна з них

відповідає за окремий етап надання допомоги. Центральною фігурою в цьому процесі є клієнт. Для нього важливі простий шлях до фахівця, швидкий зворотний зв'язок і можливість звернутися з мінімальним обсягом персональних даних. Це допомагає подолати бар'єри пов'язані зі страхом стигми або небажанням розкривати особисту історію кільком різним фахівцям в умовах нестабільності. При цьому клієнт повинен зберігати відчуття суб'єктності та контролю над власною інформацією, чітко розуміючи, як саме вона буде використана сервісом.

Після створення звернення його опрацьовує адміністратор. Ця роль полягає в тому, щоб жодне звернення не залишилося поза увагою. Він реєструє запити, проводить первинну маршрутизацію за рівнем терміновості та підбирає відповідного спеціаліста. В гуманітарних підходах ця функція є ключовою, адже психосоціальна допомога часто нероздільна від інших потреб людини. Адміністратор забезпечує зрозумілий маршрут клієнта, допомагаючи йому знайти доступ до необхідних ресурсів чи комплексного супроводу, якщо запит виходить за межі простої консультації, що відповідає міжнародним стандартам та рекомендаціям. Третьою ключовою роллю є психолог або психотерапевт, який безпосередньо надає професійну підтримку та дотримується етичних вимог роботи з клієнтом. Особливість дистанційної роботи полягає в тому, що частина невербальних сигналів через екран може бути недоступною. Саме тому міжнародні стандарти телепсихології наполягають на обов'язковому попередньому узгодженні правил спілкування, наявності резервного каналу зв'язку та чіткого плану дій на випадок ризиків. Психолог має провести інтервенцію, оцінити стан людини та задокументувати необхідну інформацію, забезпечуючи безпеку процесу у непередбачуваному середовищі. Лише узгоджена робота клієнта, адміністратора і фахівця дає змогу зробити онлайн-консультування безпечним та керованим процесом [3].

Така надійна структура дозволяє системі ефективно реагувати на конкретні людські потреби, які в умовах війни мають свої особливі сценарії. Такі ситуації важливо описувати як реальні життєві кейси, а не як абстрактні діагнози, оскільки за кожним зверненням стоїть жива історія адаптації до кризи. Часто перше

звернення має просту побутову форму, коли людина пише, що не може спати, постійно відчуває тривогу, гостро реагує на сирени, не може працювати або не витримує потоку новин. Це сценарії де потрібна коротка стабілізація, навички саморегуляції, підтримка а за потреби, перехід до більш структурованої допомоги. Підхід першої психологічної допомоги добре описує логіку такого первинного контакту де спочатку треба побачити та зрозуміти ситуацію після чого вислухати і підтримати [5].

Окрема група сценаріїв безпосередньо відображає важкі рани, завдані війною. Досвід втрати близьких, дому та руйнування базового відчуття безпеки викликають травматичні спогади часто переростають у симптоми ПТСР, що можуть проявлятися через виснажливі флешбеки коли пережите повертається з новою силою, або через гостру реакцію на звукові чи візуальні тригери, які нагадують про небезпеку. Часто спостерігається стратегія уникання або емоційне оніміння, коли людина втрачає здатність відчувати, що є захисною реакцією психіки на нестерпний біль. Для таких звернень онлайн-формат є надзвичайно ефективним, оскільки дозволяє фахівцю залишатися на зв'язку з клієнтом навіть під час його переміщень. Проте робота з травмою війни в онлайн-середовищі вимагає особливої структурованості та стабільності контакту, оскільки саме ці фактори створюють необхідну опору для людини в умовах зовнішнього хаосу [6].

Також існують кризові сценарії, в яких онлайн-сервіс повинен швидко передати звернення на відповідний рівень реагування. Особливо в моменти коли появляються суїцидальні думки, самопошкодження, насильство це все несе небезпеку для клієнта або інших. Для цього предметна область повинна прямо включати протоколи оцінки ризику, мінімальні питання для безпеки, правила ескалації та взаємодію з відповідними службами, бо без цього сервіс може нашкодити або бути юридично й етично вразливим [7].

1.2 Аналіз існуючих платформ дистанційної консультативної підтримки

Через зростання кількості дистанційних звернень звичайного відеодзвінка вже недостатньо. Потрібна система, яка зберігає звернення, керує записами, підтримує історію взаємодії та контролює доступ до даних. В ситуаціях, де клієнт перебуває в небезпеці або має нестабільний зв'язок, система повинна забезпечувати безперервність супроводу та надійний захист даних. Тому інформаційна система має поєднувати простий онлайн-доступ з вимогами до конфіденційності, стабільності роботи та контролю якості консультативного процесу.

Як перший аналог розглянуто BetterHelp — платформу, в якій реалізовано анкетування користувача, підбір фахівця та окремий простір для подальшої взаємодії. (рисунок 1.1).

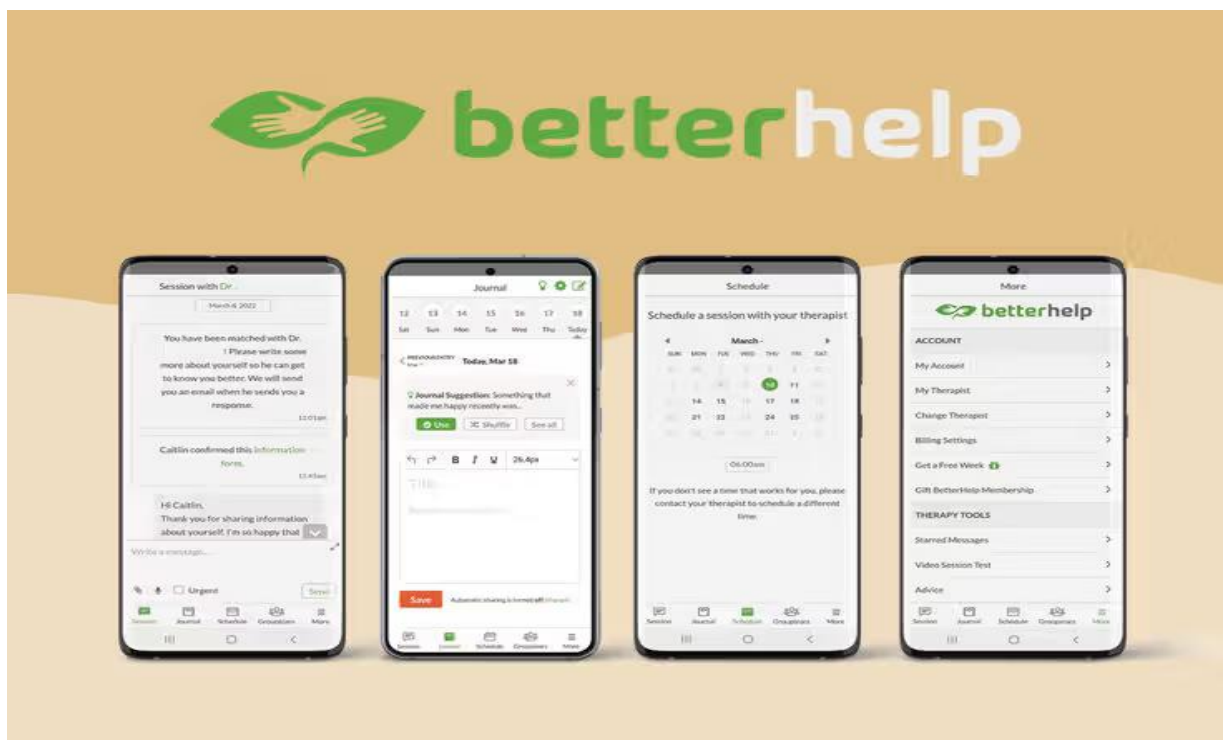


Рисунок 1.1 - Інтерфейс застосунку BetterHelp

Технічна логіка роботи сервісу базується на попередньому анкетуванні користувача, після чого система надає доступ до персональної кімнати для спілкування з терапевтом. Функціонал платформи поєднує асинхронний зв'язок

через повідомлення та синхронні живі сесії у форматі відео, аудіо або телефону. Така комбінація дозволяє підтримувати терапевтичний процес навіть за умови нестабільного інтернету, оскільки текстові та аудіофайли менш вимогливі до пропускну здатності мережі, а телефонний дзвінок слугує надійним резервним варіантом [8]. Водночас цей сервіс став важливим прикладом ризиків у сфері приватності через претензії щодо передачі чутливих даних користувачів рекламним платформам [9].

Іншим підходом до побудови системи є модель сервісу Talkspace, де пріоритет надається концепції «messaging-first». На відміну від класичних сесій, тут архітектура платформи орієнтована на безперервний терапевтичний цикл через обмін текстовими, голосовими та відеоповідомленнями (рисунки 1.1 та 1.2).



Рисунок 1.2 – Інтерфейс сервісу Talkspace

Це робить систему стійкою до розривів зв'язку, оскільки асинхронний формат дозволяє клієнту залишатися в терапії навіть за відсутності стабільного швидкісного інтернету. Окрім засобів зв'язку, Talkspace інтегрує додаткові інструменти підтримки, такі як вбудовані цифрові вправи та візуальні таймлайни прогресу, що є критично важливим для користувачів у стані емоційного виснаження. Технічний захист реалізовано через стандарти HIPAA, шифрування даних та багаторівневу авторизацію, включаючи біометричний доступ до застосунку. Важливою організаційною перевагою є впровадження функціоналу швидкої заміни фахівця в один клік, що мінімізує психологічний дискомфорт для клієнта. У зоні воєнного конфлікту така автоматизація дозволяє підтримувати безперервність допомоги та швидко реагувати на запити користувачів, не втрачаючи їхньої довіри до платформи.

Водночас, поряд із багатофункціональними екосистемами для тривалого супроводу, існують рішення, орієнтовані на максимальну швидкість та спрощення технічного входу. Одним із таких прикладів є сервіс Doxy.me, який представляє модель мінімалістичного телемедичного інструменту, орієнтованого виключно на браузерну взаємодію. Технічна логіка платформи базується на концепції одного посилання, що дозволяє користувачу приєднуватися до сесії без завантаження програмного забезпечення або створення облікового запису. Такий підхід робить сервіс доступним для власників слабких пристроїв та користувачів в зонах із нестабільною мережею. Саме усунення технічних бар'єрів на вході зменшує ризик того, що користувач відмовиться від консультації через складність підключення. (рисунок 1.3) [12].

Система реалізує функціонал віртуальної приймальні, де пацієнт проходить автоматичну перевірку зв'язку та очікує підключення фахівця. Важливою технічною особливістю є можливість модерації чату в режимі очікування та адаптація під низьку пропускну здатність каналу, що дозволяє проводити консультації в аудіоформаті при нестабільному інтернет-з'єднанні.



Рисунок 1.3 – Веб-інтерфейс Doxy.me

В підсумку, поєднання легкої веб-технології для взаємодії з користувачем через клієнтську частину з захищеною внутрішньою інфраструктурою для документації та супроводу дозволяє створити стійку систему, адаптовану до технічних реалій воєнного стану [13].

Поряд із глобальними комерційними екосистемами, які задають високі технічні стандарти галузі. Особливе значення має аналіз вітчизняних рішень, які адаптовані до умов воєнного стану. Одним з таких аналогів є український проект «ПОРУЧ», який ілюструє модель організації психологічної підтримки для конкретних цільових груп в гуманітарних умовах. На відміну від світових платформ-посередників, цей проект фокусується не на індивідуальному підборі, а на груповому форматі роботи із залученням фахівців, що мають досвід допомоги людям, травмованим війною (рисунок 1.4).

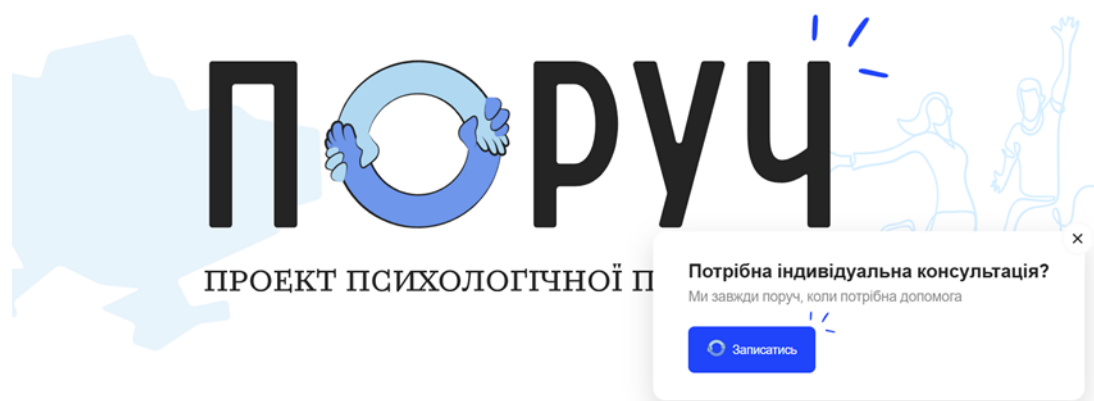


Рисунок 1.4 – Організація роботи платформи

Технічна логіка доступу базується на максимально спрощеному вході через веб-сторінку та зовнішні форми реєстрації, що дозволяє швидко залучати користувачів без створення складних персональних кабінетів. Основна взаємодія реалізована через регулярні онлайн-зустрічі в Zoom за чітким розкладом, де модерація процесу забезпечується безпосередньо ведучим-психологом та обмеженою кількістю учасників. Такий підхід ефективно знижує поріг входу для людей у стресовому стані, проте має певні технічні обмеження через розрізненість інструментів. Зокрема, керування розкладом залишається переважно ручним, а безпека та конфіденційність даних розподілені між сторонніми сервісами зв'язку та реєстрації. Попри високу вимогливість відеозв'язку до мережі, груповий формат забезпечує ефект підтримки навіть за умови вимкненої камери, а спеціальні інформаційні канали з матеріалами для самопомоги частково замінюють вбудовані модулі супроводу. Досвід проєкту «ПОРУЧ» підтверджує, що для швидкого масштабування допомоги в кризових умовах мінімалістичні рішення часто є ефективнішими за складні системи. Водночас це підкреслює необхідність створення єдиної технічної бази, яка б дозволила зберегти простоту доступу, але забезпечила кращий контроль над безпекою даних та цілісністю користувацького досвіду [14].

Проведений огляд показав, що кожна платформа вирішує лише частину завдань дистанційної підтримки. Для умов гуманітарної кризи потрібне рішення,

яке поєднує простий доступ, захист даних і керування консультаційним процесом. Основні характеристики та відмінності між розглянутими сервісами наведено в таблиці 1.1.

Таблиця 1.1 — Порівняльна характеристика систем

Характеристика	BetterHelp	Talkspace	Doxy.me	ПОРУЧ
Основний фокус	Індивідуальна терапія	Асинхронний чат	Відеозв'язок	Групові підтримка
Вхід у систему	Реєстрація + анкета	Реєстрація	Без логіну	Зовнішня форма
Вимоги до мережі	Високі (відео)	Низькі (текст)	Адаптивні	Високі (Zoom)
Безпека даних	Шифрування баз	Стандарт HIPAA	Шифрування каналу	Розподілена (низька)
Цільова аудиторія	Масовий ринок	Масовий ринок	Клініки	Постраждалі від війни

Проведений аналіз показує необхідність синтезу високої автоматизації комерційних систем із низьким порогом входу, характерним для гуманітарних ініціатив. Основною проблемою глобальних платформ залишається надлишкова складність інтерфейсів та маркетингова спрямованість, що ускладнює взаємодію для користувачів у критичних станах. Локальні та спеціалізовані проєкти, навпаки, часто позбавлені єдиної захищеної інфраструктури та інструментів автоматизованого менеджменту кейсів.

1.3 Постановка задачі розробки інформаційної системи онлайн-консультацій

В умовах воєнного часу та масових переміщень населення зростає потреба в доступній психологічній, юридичній та соціальній підтримці, яку можна отримати без прив'язки до конкретного місця. Онлайн-формат консультацій дозволяє швидко налагодити контакт із фахівцем, зменшити бар'єри звернення та забезпечити безперервність супроводу навіть під час зміни локації або нестабільного транспортного сполучення.

Актуальність задачі полягає у необхідності організувати повний цикл дистанційної допомоги як керований процес: від первинного звернення та маршрутизації до запису, проведення консультації, збереження результатів і формування сповіщень, з урахуванням конфіденційності, ролей доступу та ризиків кризових станів.

Метою даної роботи є проектування та розробка інформаційної системи надання онлайн-консультацій для осіб, які постраждали від війни, що забезпечує взаємодію між клієнтом, консультантом і адміністратором, підтримує календар і бронювання, чат/відеозв'язок, обмін матеріалами, сповіщення та журналювання подій.

Для досягнення цієї мети необхідно вирішити такі завдання:

1. Провести аналіз предметної області онлайн-консультування для постраждалих від війни та визначити основні ролі, сценарії і вимоги до сервісу.
2. Виконати огляд існуючих аналогів та сформулювати перелік функціональних вимог до інформаційної системи.
3. Спроекувати загальну архітектуру системи та визначити її ключові модулі і взаємодію між ними.
4. Розробити алгоритмічне забезпечення обробки звернень: первинний запит, маршрутизація, підбір фахівця, запис, проведення і завершення консультації.
5. Розробити інформаційне забезпечення: виділити основні сутності, побудувати інфологічну модель та ER-діаграму бази даних із підтримкою цілісності даних.
6. Передбачити механізми безпеки та конфіденційності: автентифікацію, авторизацію за ролями, контроль доступу до консультацій, аудит дій і модерацію.
7. Реалізувати програмно-технологічну частину системи (клієнтський інтерфейс і серверну логіку) та інтеграцію із сервісами сповіщень і відеозв'язку.
8. Провести тестування працездатності системи та перевірити роботу ключових сценаріїв, зокрема в умовах нестабільного інтернет-з'єднання.

2 ПРОЕКТУВАННЯ АРХІТЕКТУРИ, АЛГОРИТМІЧНОГО ТА ІНФОРМАЦІЙНОГО ЗАБЕЗПЕЧЕННЯ СИСТЕМИ

2.1 Архітектура та функціональна структура інформаційної системи онлайн-консультування

Проектування системи онлайн-консультацій потрібно почати з визначення її структури, основних модулів і зв'язків між ними. Це необхідно для того, щоб надалі коректно описати алгоритми роботи системи, структуру даних, права доступу та логіку взаємодії між користувачами. Така побудова відповідає типовому сценарію дистанційної підтримки, в якому кожен вузол відповідає за окремий етап від первинної реєстрації звернення до збереження результатів консультації.

Інформаційна система призначена для автоматизації повного циклу дистанційної взаємодії між клієнтом і консультантом, забезпечуючи не лише прямий зв'язок, а й сукупність супровідних процесів. Функціонал платформи охоплює реєстрацію користувачів, створення та обробку звернень, ведення профілів фахівців, а також механізми запису на сесії та обміну повідомленнями. Крім того, система координує роботу засобів відеозв'язку, зберігання документів, надсилання сповіщень і журналювання подій.

Загалом архітектура системи охоплює клієнтську частину, серверну логіку, базу даних, файлове сховище та модуль інтеграції з зовнішніми сервісами. Клієнтська частина виступає основним інструментом взаємодії користувача з платформою через веб- або мобільні інтерфейси, тоді як серверна частина реалізує прикладну логіку системи. Вона забезпечує обробку запитів, перевірку прав доступу, керування станами об'єктів та синхронізацію обміну даними між внутрішніми вузлами.

Основою стабільної роботи системи виступає централізована база даних, де зберігаються облікові записи, ролі, профілі та історія взаємодій. В той самий час файлове сховище забезпечує безпечну роботу з документами та вкладеннями, що використовуються в процесі консультування.

Окремим елементом архітектури є модуль інтеграції із зовнішніми сервісами, зокрема сервісами сповіщень і відеозв'язку. Така багаторівнева побудова дає змогу розмежувати технічні функції окремих шарів, спрощує масштабування системи та підвищує її загальну відмовостійкість. На рисунку 2.1 зображено загальну структуру інформаційної системи онлайн-консультування.



Рисунок 2.1 - Структура інформаційної системи онлайн-консультування

Основними учасниками системи є клієнт, консультант і адміністратор. Клієнт може зареєструватися в системі або створити звернення у спрощеному режимі. Після цього він переглядає доступних консультантів, обирає напрям допомоги, бронює вільний час, бере участь у чаті або відеоконсультації, отримує повідомлення та, за потреби, завантажує чи переглядає документи. Консультант, у свою чергу, працює зі своїм профілем, налаштовує години прийому, переглядає заплановані консультації, проводить сесії, надає відповіді та рекомендації, а також

отримує матеріали від клієнта. Адміністратор керує користувачами, виконує модерацію, працює з журналами подій, контролює стабільність роботи системи та відповідає за її безпеку.

Функціональна структура інформаційної системи онлайн-консультування будується як набір взаємопов'язаних модулів. В межах такої побудови кожен із них виконує власне завдання, але водночас взаємодіє з іншими модулями. Наприклад, запис на консультацію неможливий без перевірки користувача, а проведення консультації пов'язане з обміном повідомленнями, файлами та формуванням сповіщень. Адміністративні дії також не існують окремо, оскільки вони пов'язані з модерацією і контролем системних подій. Поділ на модулі спрощує проектування і реалізацію системи, оскільки кожна функція має окрему зону відповідальності. На рисунку 2.2 зображено взаємодію основних функціональних модулів інформаційної системи онлайн-консультування.



Рисунок 2.2 - Взаємодія основних функціональних модулів

Одним із базових модулів цієї системи є модуль керування користувачами і ролями. Його призначення полягає у забезпеченні реєстрації, автентифікації, авторизації та розподілу прав доступу. Без цього неможливо організувати безпечну роботу системи, оскільки різні категорії користувачів мають різні функції та рівні доступу. Цей модуль має підтримувати кілька типів облікових записів.

Перший тип — зареєстрований користувач з повним доступом до функцій системи. Другий тип — користувач зі спрощеним доступом, який може створити частково анонімне звернення. Такий підхід є важливим, оскільки частина осіб, які постраждали від війни, може не бажати надавати повний набір персональних даних на початковому етапі. У такому випадку система повинна підтримувати мінімально необхідну взаємодію з обмеженим набором полів, а в разі потреби ці дані можуть бути уточнені.

Другим важливим модулем є модуль профілів консультантів. Фактично він виконує роль довідника консультантів і допомагає клієнту обрати відповідного спеціаліста залежно від напряму допомоги, досвіду та доступності. На основі інформації про консультанта і його вільний час користувач може вибрати зручний момент для проведення сесії. Цей модуль тісно пов'язаний із календарем і бронюванням, оскільки саме через нього організовується запис на консультацію. Календар і бронювання взаємодіють з авторизацією, тому що створення, зміна або скасування запису можливі лише для авторизованих користувачів. Крім того, вони безпосередньо пов'язані з модулем консультацій, оскільки після підтвердження запису формується відповідна сесія.

Основним робочим модулем є модуль консультацій. Саме він забезпечує проведення онлайн-взаємодії між клієнтом і консультантом. У межах цього модуля виконуються створення сесії, контроль доступу до неї, збереження метаданих сеансу, обмін повідомленнями, передавання службових статусів і завершення консультації. Для чату це означає збереження повідомлень, часу їх відправлення та прив'язки до конкретної консультації. Якщо консультація проводиться з використанням зовнішньої платформи, система може забезпечувати створення посилання на Zoom або інший сервіс відеозв'язку, а також зберігати інформацію

про створену сесію, її учасників, час початку та завершення. Після завершення консультації система може змінити статус звернення, зберегти короткий службовий підсумок і надати можливість прикріпити рекомендації або документи.

Окреме місце в системі займає модуль сповіщень. Його призначення полягає в автоматичному інформуванні користувачів про важливі події, зокрема про реєстрацію облікового запису, підтвердження запису на консультацію, зміну часу сесії, нагадування перед її початком, надходження нового повідомлення, зміну статусу звернення або відповідь адміністратора. У найпростішому варіанті сповіщення можуть відображатися безпосередньо в самій системі. За потреби також може бути реалізовано надсилання повідомлень через email, SMS, месенджери або інші зовнішні сервіси. Такий підхід дає змогу своєчасно інформувати користувача навіть у тих випадках, коли він не перебуває в особистому кабінеті.

2.2 Алгоритми обробки звернень, запису та проведення онлайн-консультацій

Робота інформаційної системи базується на послідовному процесі обробки звернення користувача та організації онлайн-консультації. Весь процес починається зі створення звернення клієнтом, у якому зазначається тип потрібної допомоги, короткий опис проблеми та за можливості додаткові контактні дані. Після цього система виконує початкову обробку запиту, визначає напрям консультації та підбирає відповідного консультанта. На наступному етапі перевіряється доступний час у розкладі спеціаліста, після чого формується запис на консультацію та надсилається підтвердження учасникам. На рисунку 2.3 зображено алгоритм роботи системи.

Під час проведення консультації система забезпечує доступ до створеної сесії, підтримує обмін повідомленнями та, за потреби, формує посилання на зовнішню платформу для онлайн-зустрічі. Додатково може виконуватися передача матеріалів і документів, пов'язаних із конкретним зверненням. Після завершення консультації система змінює статус звернення, зберігає службову інформацію про

проведену сесію та, за потреби, фіксує зворотний зв'язок. Якщо на будь-якому етапі виникає помилка, система або пропонує альтернативний варіант дій, або реєструє проблему для подальшої обробки адміністратором.

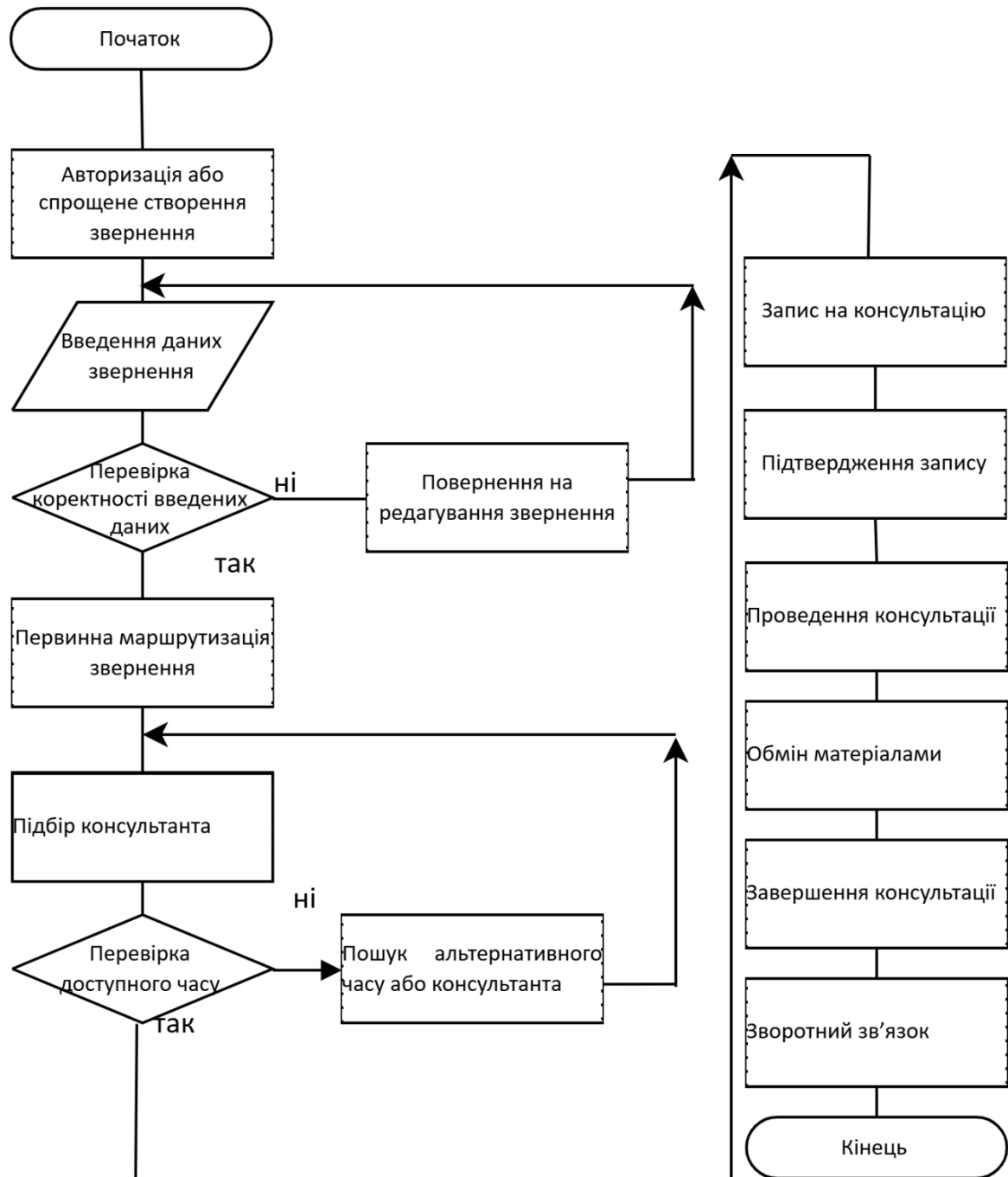


Рисунок 2.3 – Схема алгоритму роботи інформаційної системи онлайн консультацій

Алгоритм починається з взаємодії користувача з системою. На цьому етапі можливі два основні сценарії. В першому випадку користувач має обліковий запис і проходить авторизацію. У другому випадку система дозволяє створити звернення в спрощеному режимі, коли вводиться мінімальний набір даних. Після входу в систему або активації форми звернення користувач вводить такі основні дані як тему звернення, категорію проблеми, бажаний формат консультації, короткий опис ситуації та контактні дані для зворотного зв'язку. Далі система перевіряє коректність введених даних. Якщо перевірка не проходить, система не продовжує обробку звернення, а повертає користувача до форми редагування. Якщо ж дані коректні, звернення переходить до етапу первинної маршрутизації.

Первинна маршрутизація (рисунок 2.4) є одним із ключових етапів алгоритму, тому що на цьому етапі система визначає, до якого напрямку допомоги належить звернення.



Рисунок 2.4 – Схема алгоритму створення звернення та первинної маршрутизації

На основі вибраної категорії та змісту звернення система встановлює відповідний тип консультації та передає звернення до потрібної логічної гілки обробки. Якщо наданої інформації недостатньо, формується запит на уточнення даних, а звернення повертається на доопрацювання. Такий підхід дає змогу скоротити час первинного опрацювання запиту та підвищити точність його подальшої обробки.

Після завершення первинної маршрутизації система переходить до етапу підбору консультанта (рисунок 2.5).

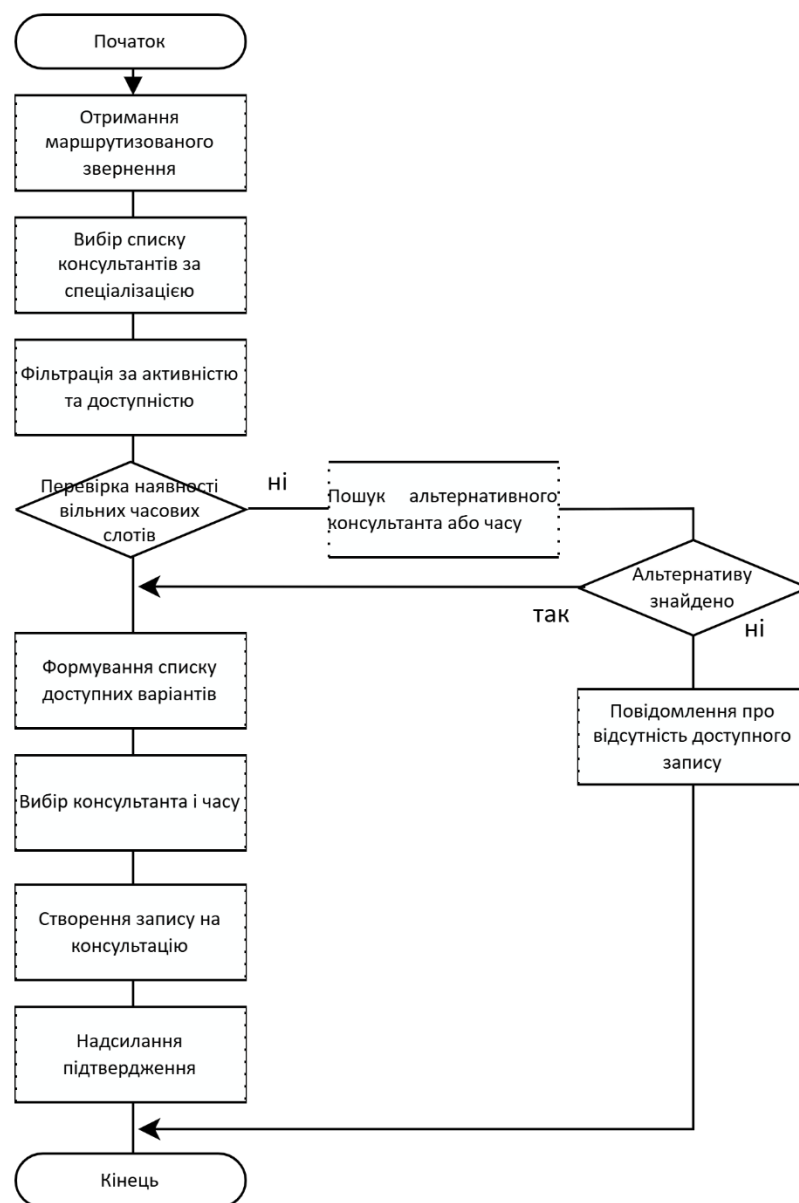


Рисунок 2.5 – Схема алгоритму підбору консультанта, запису та підтвердження консультації.

На цьому етапі консультант підбирається за кількома параметрами. До них належать напрям консультації, компетенція консультанта, активний статус профілю, доступний час, мова спілкування, а в окремих випадках пріоритетність звернення. Таким чином, підбір консультанта виконується шляхом зіставлення характеристик звернення з параметрами доступних профілів консультантів.

В процесі підбору консультанта система може виявити кілька спеціалістів, які відповідають заданим параметрам. В такому випадку користувачу може бути запропоновано вибір між доступними варіантами або автоматично визначено консультанта з найближчим вільним часом. Якщо ж відповідного консультанта або доступного запису немає, система не завершує обробку звернення одразу, а виконує пошук альтернативного консультанта або іншого часу. У разі успішного пошуку процес продовжується до етапу створення запису на консультацію. Якщо альтернативу не знайдено, користувачеві надсилається повідомлення про відсутність доступного запису. Отже, підбір консультанта пов'язаний з календарем, оскільки система має одночасно враховувати спеціалізацію фахівця та його доступний час.

Після створення запису та надсилання підтвердження обом сторонам система переходить до етапу безпосереднього проведення консультації. На цьому етапі важливо забезпечити своєчасний початок сесії, перевірку статусу запису, контроль доступу учасників, обмін повідомленнями і матеріалами, а також коректне завершення консультації із збереженням результатів. Окремо система повинна враховувати ситуації, коли консультацію неможливо розпочати або під час її проведення виникає помилка. На рисунку 2.6 зображено алгоритм проведення консультації, завершення сесії та надсилання форми зворотного зв'язку.

Після настання часу консультації система перевіряє, чи є запис активним. Якщо запис неактивний, подальше виконання алгоритму не продовжується, а користувачу надсилається повідомлення про неможливість початку консультації. Якщо запис активний, система відкриває сесію консультації та перевіряє доступ учасників. Лише після підтвердження доступу можливий перехід до основного етапу взаємодії між клієнтом і консультантом.

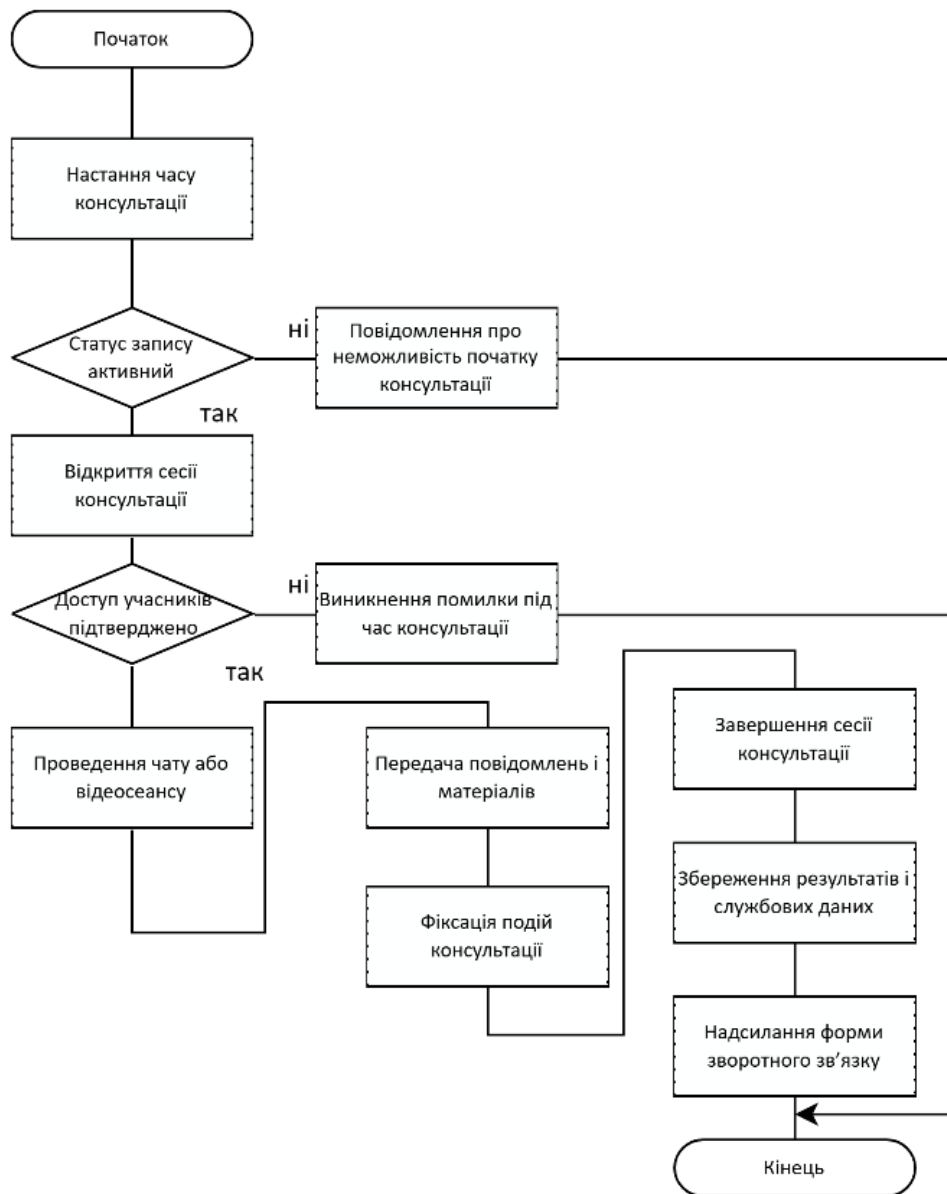


Рисунок 2.6- Алгоритм роботи сесії та зворотного зв'язку

Основна частина алгоритму охоплює проведення чату або відеосеансу, передачу повідомлень і матеріалів, а також фіксацію подій консультації. Це дає змогу забезпечити повноцінну взаємодію між сторонами в межах однієї сесії та зберігати службову інформацію про її перебіг. Якщо на етапі перевірки доступу виникає проблема, система фіксує помилкову ситуацію і завершує виконання поточного сценарію без переходу до наступних кроків.

Після завершення сесії система переходить до збереження результатів і службових даних. На цьому етапі можуть фіксуватися відомості про проведену консультацію, її тривалість, технічний стан сеансу та інша службова інформація,

яка потрібна для подальшого супроводу звернення. Після цього користувачеві надсилається форма зворотного зв'язку, що завершує даний алгоритм.

Це дає можливість поєднати в одному алгоритмі як основний сценарій проведення консультації, так і її коректне завершення. В результаті система виконує сам процес взаємодії між клієнтом і консультантом та контролює стани запису, перевіряє доступ, фіксує події, зберігає результати та підтримує зворотний зв'язок після завершення сесії.

2.3 Інформаційне забезпечення та модель бази даних системи

Для системи онлайн-консультацій інформаційне забезпечення має ключове значення. Якщо звернення, часові слоти, консультації та повідомлення зберігаються некоректно, навіть правильно спроектована архітектура не забезпечить стабільної роботи сервісу. Тому на цьому етапі потрібно окремо описати вхідну та вихідну інформацію, концептуальну інфологічну модель і ER модель даних.

Вхідною інформацією для системи є дані, що надходять від клієнта, консультанта, адміністратора та внутрішніх сервісних модулів. Основним видом таких даних є звернення клієнта. Воно містить тип потрібної консультації, категорію проблеми, короткий опис ситуації, бажаний формат взаємодії, контактні дані та, за потреби, прикріплені файли.

Звернення запускає основний цикл роботи системи, тому що після його створення виконуються маршрутизація, підбір консультанта та формування запису на консультацію. До вхідної інформації також належать категорії консультацій, які використовуються для первинного розподілу звернень і визначення напряму допомоги.

Ще одним видом вхідних даних є часові слоти, що формуються на основі графіка доступності консультантів і використовуються модулем календаря та бронювання для перевірки можливості запису на конкретну дату і час. Також під час консультації до системи надходять повідомлення та файли, які супроводжують

взаємодію між клієнтом і консультантом. Повідомлення формують історію спілкування, а файли можуть містити довідки, скановані документи, рекомендації, пам'ятки та інші матеріали, пов'язані з консультаційним процесом. Окрему групу вхідної інформації становлять події аудиту, які фіксуються під час важливих дій у системі, зокрема під час входу користувача, створення звернення, перенесення консультації, завантаження документа або адміністративного втручання.

Вихідною інформацією є дані, які система формує після обробки вхідної інформації (рисунк 2.7).

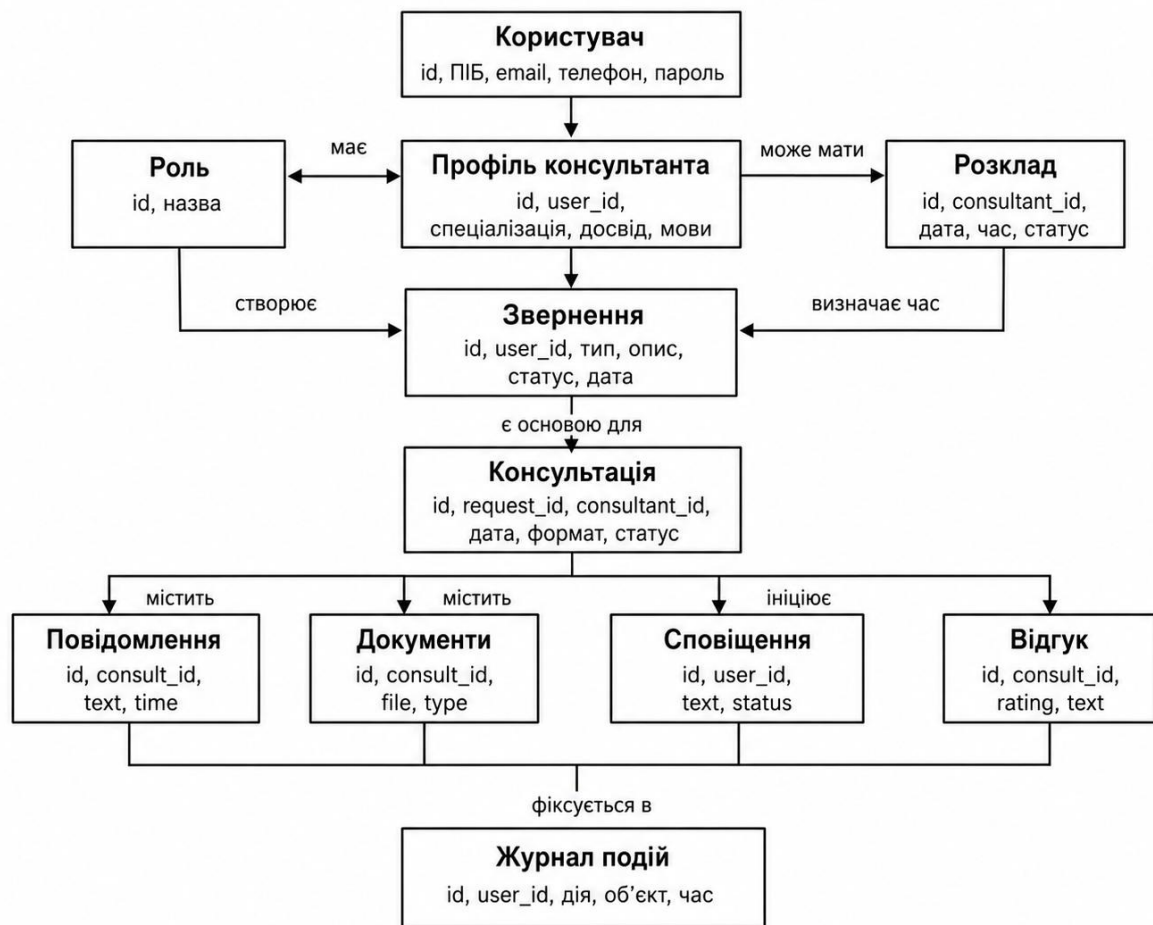


Рисунок 2.7 - Інфологічна модель даних системи онлайн консультацій

Насамперед це підтвердження запису на консультацію, яке містить відомості про консультанта, формат консультації, дату, час і поточний статус запису. Також система формує нагадування, що надсилаються клієнту і консультанту перед початком консультації. Після завершення консультації формується історія

консультацій, яка містить відомості про проведені сеанси, їх статус, дату, формат, а також пов'язані повідомлення і документи. Окрему частину вихідної інформації становлять журнали та службові звіти, що використовуються для контролю роботи системи. Вони відображають активність користувачів, стан консультацій, дані про перенесення і скасування, а також інформацію про помилки та інші події.

Наведені вхідні та вихідні дані утворюють основу інформаційної моделі системи. Для їх упорядкування на концептуальному рівні необхідно виділити основні сутності предметної області та встановити зв'язки між ними.

Після опису концептуальної моделі її можна деталізувати у вигляді ER-діаграми сутностей і зв'язків.

На цьому рівні основні сутності предметної області відображаються у вигляді таблиць бази даних, а зв'язки між ними реалізуються за допомогою первинних і зовнішніх ключів. Для проєктованої системи доцільно використовувати реляційну модель даних, оскільки вона добре підходить для зберігання структурованої інформації, забезпечення цілісності даних і виконання пошуку за статусами, датами, користувачами та консультаціями.

Основними таблицями цієї моделі є `roles`, `users`, `consultant_profiles`, `time_slots`, `requests`, `consultations`, `messages`, `files`, `notifications`, `feedback` та `audit_events` (рисунк 2.8)

Таблиця `roles` зберігає ролі користувачів у системі, а таблиця `users` містить їхні облікові дані. Для користувачів, які виконують роль консультанта, використовується таблиця `consultant_profiles`, у якій зберігаються додаткові професійні відомості. Доступний розклад консультантів фіксується в таблиці `time_slots`. Первинні звернення користувачів зберігаються в таблиці `requests`, а таблиця `consultations` описує вже конкретні консультації, сформовані на основі цих звернень. Для підтримки взаємодії під час консультації використовуються таблиці `messages` і `files`, що відповідають за обмін повідомленнями та матеріалами.

Результати оцінювання після завершення консультації фіксуються в таблиці `feedback`, а події журналювання зберігаються в таблиці `audit_events`.

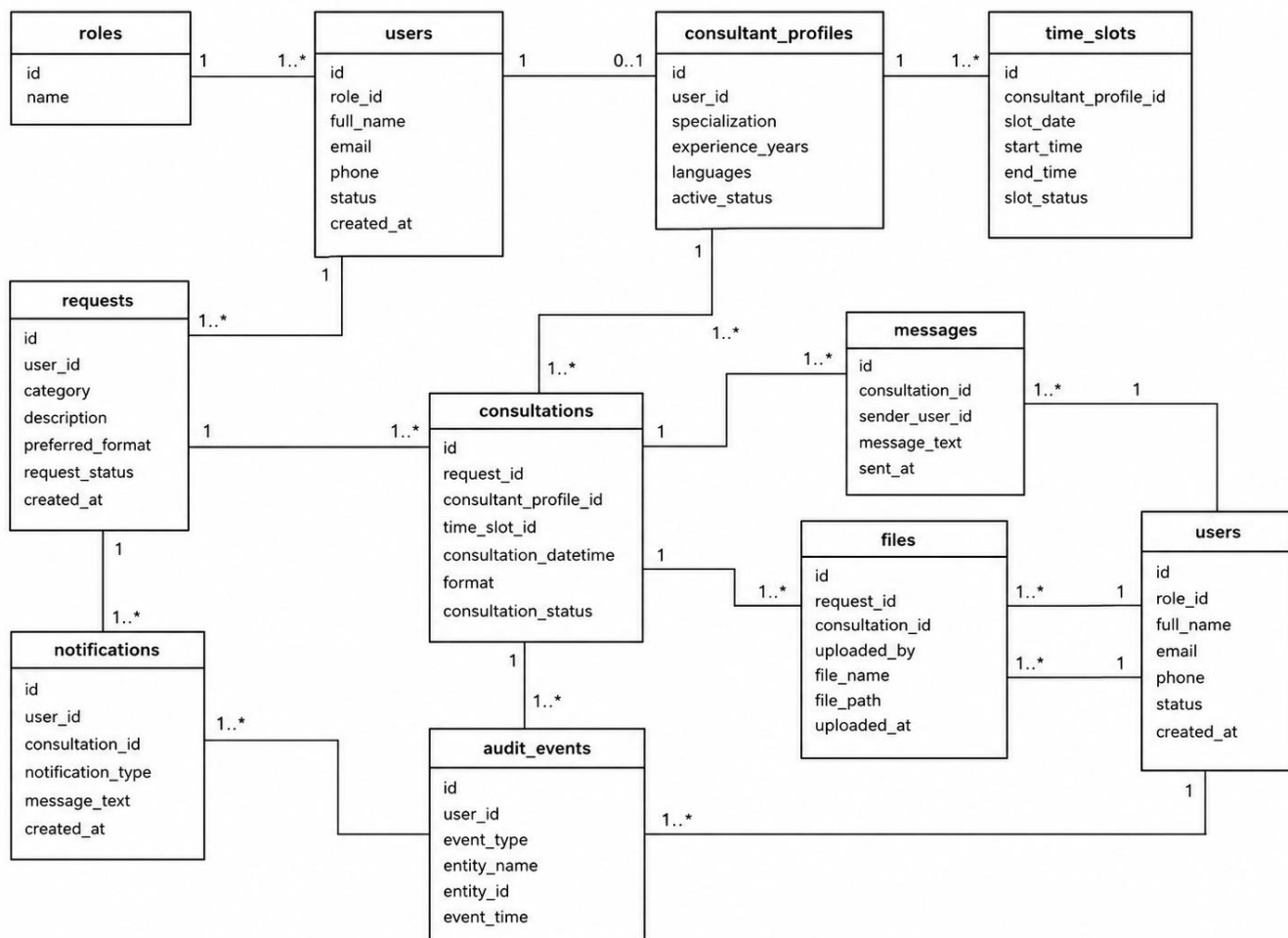


Рисунок 2.8 - ER-діаграма системи онлайн консультацій

Таблиця notifications призначена для зберігання нагадувань, підтверджень та інших службових сповіщень. Така структура дає змогу впорядковано зберігати основні дані системи та підтримувати зв'язки між її ключовими об'єктами.

3 ПРОГРАМНО-ТЕХНОЛОГІЧНА РЕАЛІЗАЦІЯ ТА ТЕСТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

3.1 Вибір та аналіз програмно-апаратних засобів для інформаційної системи

Як основний програмний засіб серверної реалізації обрано фреймворк Django. Його використання є доцільним, оскільки він містить вбудовані механізми автентифікації, авторизації, розмежування прав доступу, роботи з моделями даних, шаблонами сторінок та адміністративною панеллю. Завдяки цьому основну логіку системи можна реалізувати в одному Django-проєкті, не створюючи надмірно складної архітектури. Крім того, шаблонний механізм Django зручний для побудови сторінок кабінету користувача, консультанта та адміністратора (рисунок 3.1).

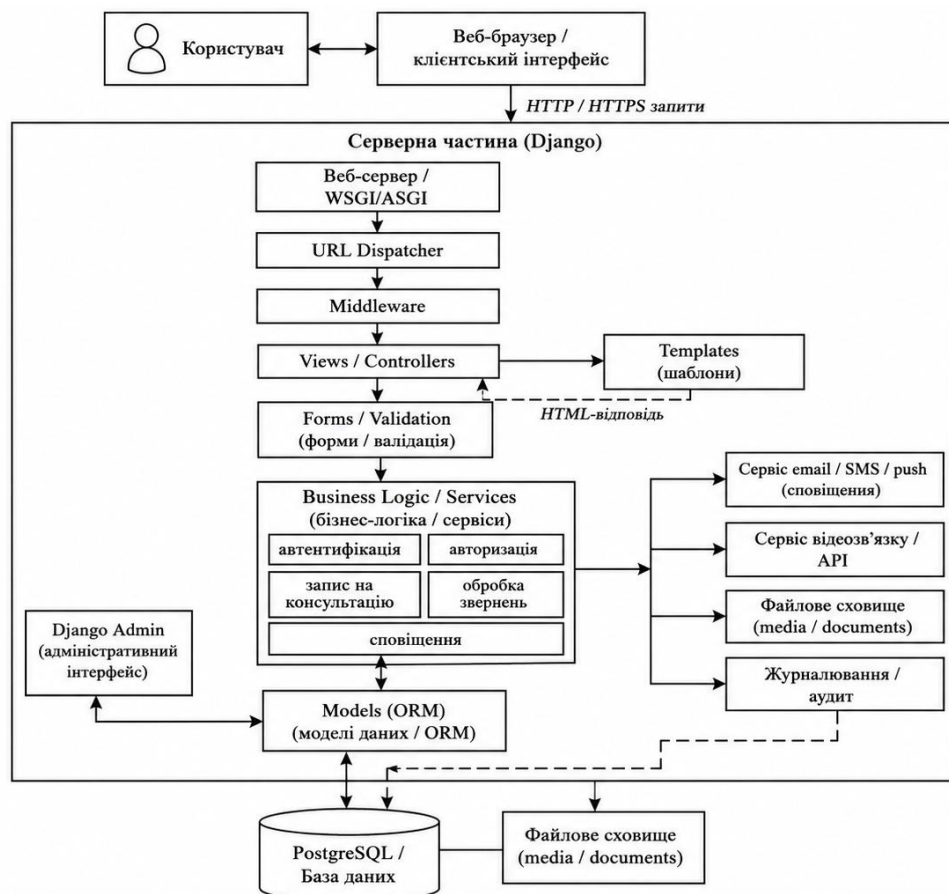
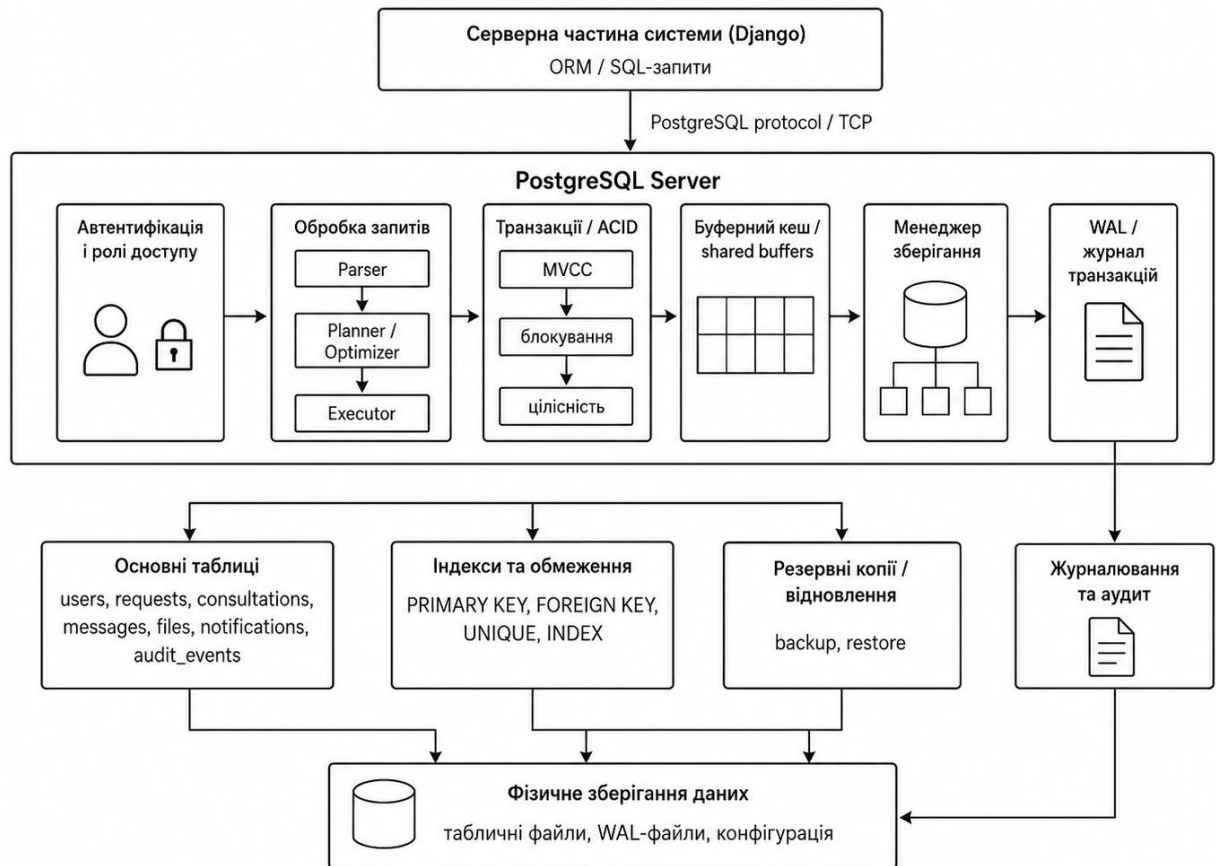


Рисунок 3.1 – Архітектура серверної частини на базі фреймворку Django

Для зберігання даних обрано PostgreSQL. Ця система керування базами даних добре підходить для реляційної моделі, що містить пов'язані сутності users, requests, consultations, messages, files, notifications та audit_events (рисуюнок 3.2).



Рисуюнок 3.2 – Система керування базами даних PostgreSQL

Інтерфейс системи побудовано на базі Bootstrap, який забезпечує адаптивність для коректної роботи на смартфонах, планшетах і ПК. Це важливо для системи онлайн-консультацій, тому що користувачі можуть працювати з нею з смартфона, планшета або комп'ютера, зокрема за нестабільного доступу до мережі. Використання Bootstrap дає змогу реалізувати єдину структуру сторінок, форми введення даних, навігаційні елементи, кнопки, таблиці та повідомлення про стан операцій. Тому інтерфейс кабінету клієнта, консультанта та адміністратора може мати однакову логіку взаємодії, що спрощує користування системою.

Для організації відеозв'язку обрано Jitsi Meet, оскільки цей сервіс підтримує вбудовування відеосесій у вебзастосунок через IFrame API. Завдяки цьому можна реалізувати консультації без розробки власного відеомодуля, зберігши при цьому

єдину точку входу в систему. У такому підході користувач переходить до відеоконсультації безпосередньо з особистого кабінету або сторінки запису, а серверна частина відповідає за формування посилання, перевірку прав доступу та прив'язку сесії до відповідної консультації. Це дозволяє поєднати відеозв'язок із журналюванням подій, сповіщеннями та збереженням інформації про проведену консультацію.

Розгортання системи можна реалізувати на платформі Render (рисунок 3.3), яка забезпечує автоматичну публікацію Django-застосунку з Git-репозиторію та інтеграцію з базою даних PostgreSQL. Після внесення змін до коду і відправлення їх у репозиторій платформа може автоматично запускати процес збирання, встановлення залежностей і перезапуску вебсервісу.

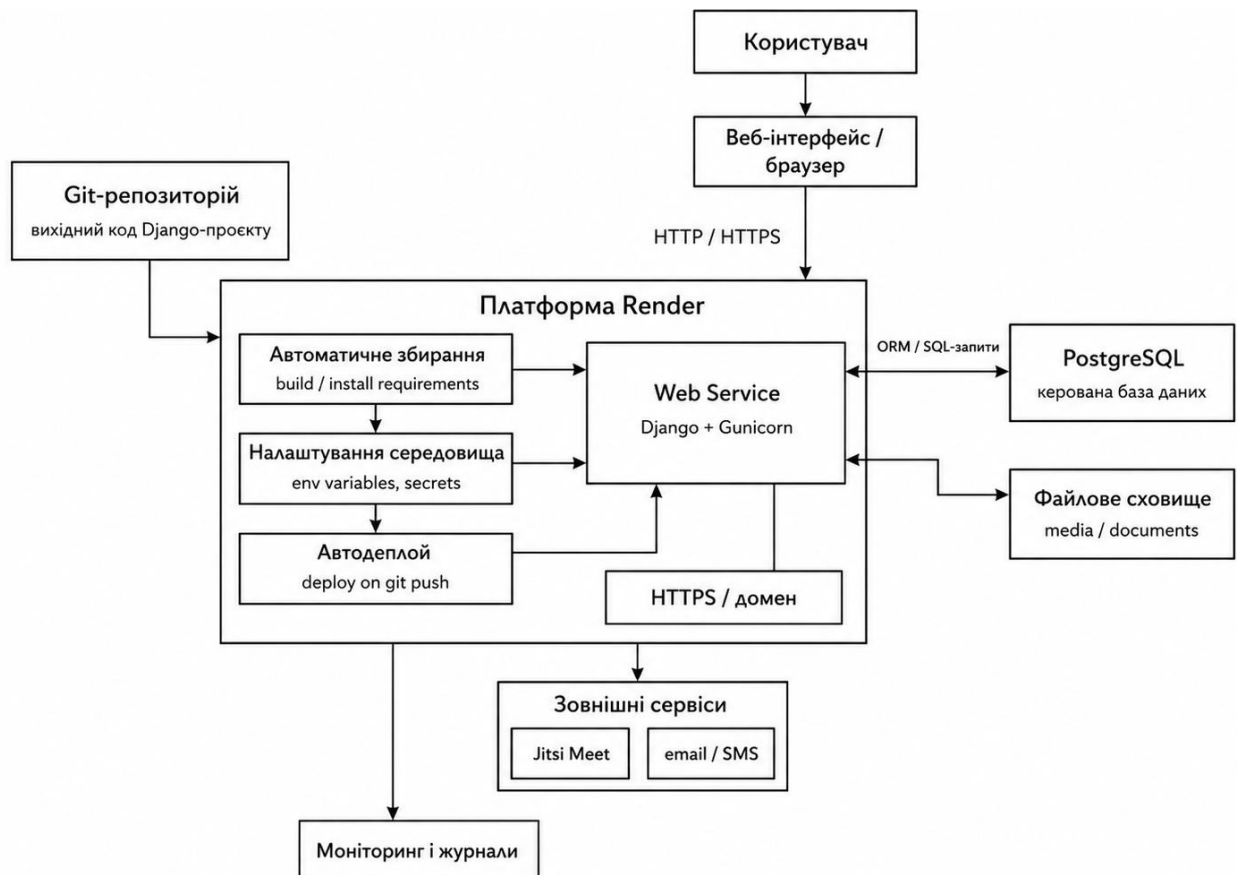


Рисунок 3.3 - Схема хмарного розгортання Django на платформі Render з підключенням PostgreSQL

Це спрощує супровід системи, зменшує потребу в ручному налаштуванні сервера та забезпечує централізоване керування розгортанням. Загальну схему хмарного розгортання Django-застосунку на платформі Render зображено на рисунку 3.3.

3.2 Реалізація модулів Django-застосунку онлайн-консультацій

Інформаційну систему реалізовано як модульний вебзастосунок. Така структура дає змогу окремо розробляти й перевіряти функції авторизації, звернень, консультацій, повідомлень, файлів і сповіщень. Його структура охоплює всі основні етапи роботи системи: автентифікацію користувачів, обробку звернень, підбір консультанта, бронювання часу, проведення консультації, обмін повідомленнями, збереження файлів і формування сповіщень. Це дозволяє послідовно реалізовувати та перевіряти окремі функціональні модулі. На початковому етапі виконується розгортання середовища розробки та базова конфігурація Django-проєкту, що є основою подальшої реалізації системи (рисунок 3.4).

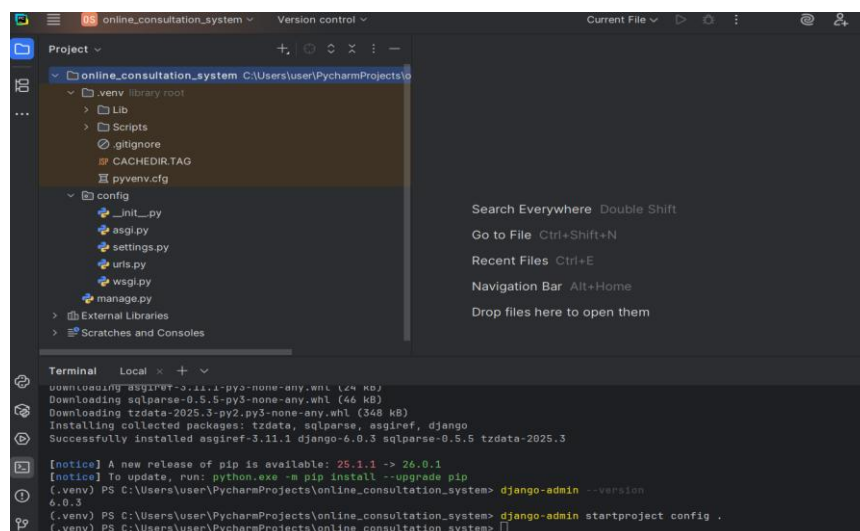


Рисунок 3.4 - Розгортання базової конфігурації системи в IDE PyCharm

Після завершення конфігурації виконується запуск локального сервера Django для перевірки працездатності системи. Це дозволяє в режимі реального часу

відстежувати коректність обробки запитів та візуалізувати інтерфейс на етапі розробки (рисунок 3.5).

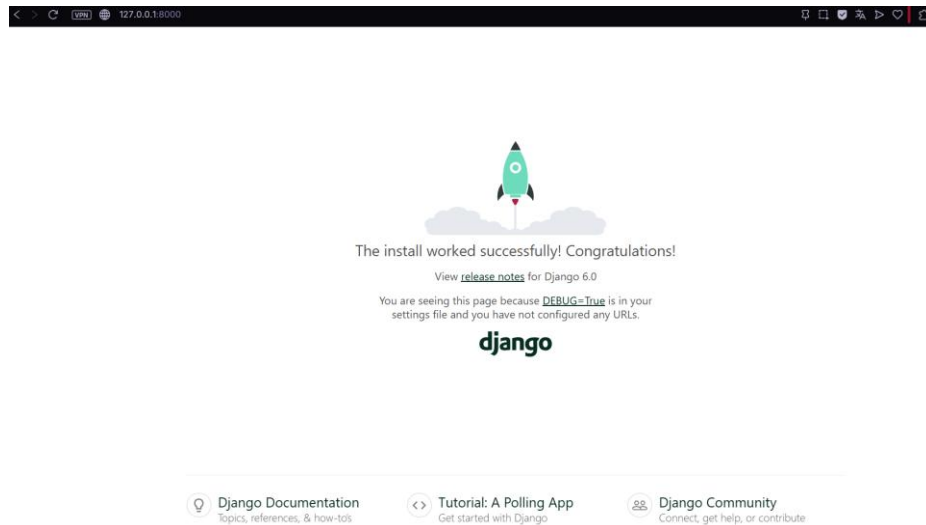


Рисунок 3.5 - Результат успішного запуску локального сервера Django

Для зручності розробки проект поділено на окремі Django-застосунки, кожен з яких відповідає за власну частину функціоналу. В застосунку Django було створено окремі функціональні модулі `users`, `requests_app`, `consultations_app`, `schedule_app`, `messages_app`, `files_app`, `notifications_app` та `audit_app`. Така побудова дозволяє відокремити логіку роботи із користувачами, зверненнями, часовими слотами, консультаціями, повідомленнями, файлами, сповіщеннями та журналом дій (рисунок 3.6).

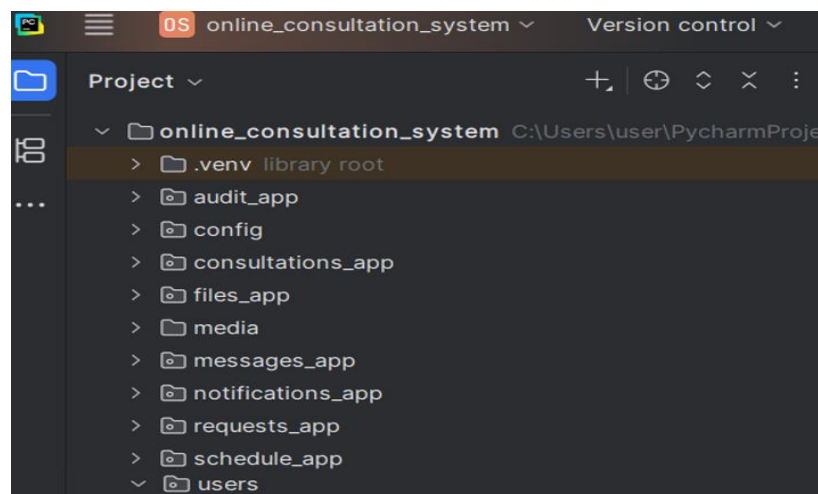
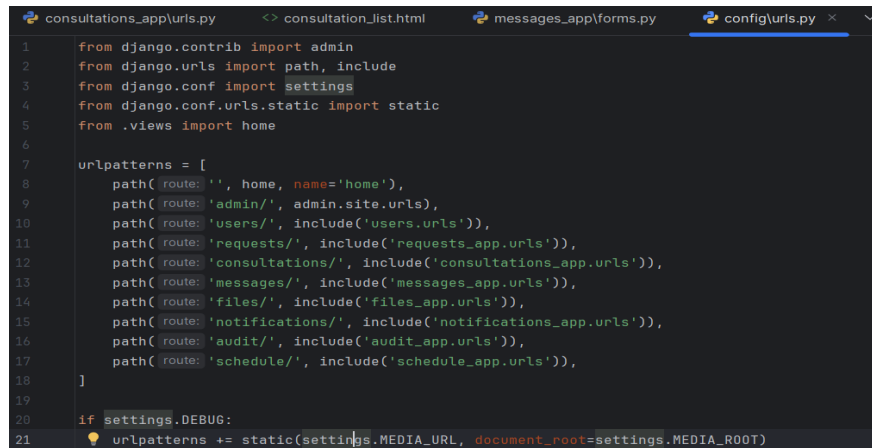


Рисунок 3.6 – Структура Django-застосунків проєкту

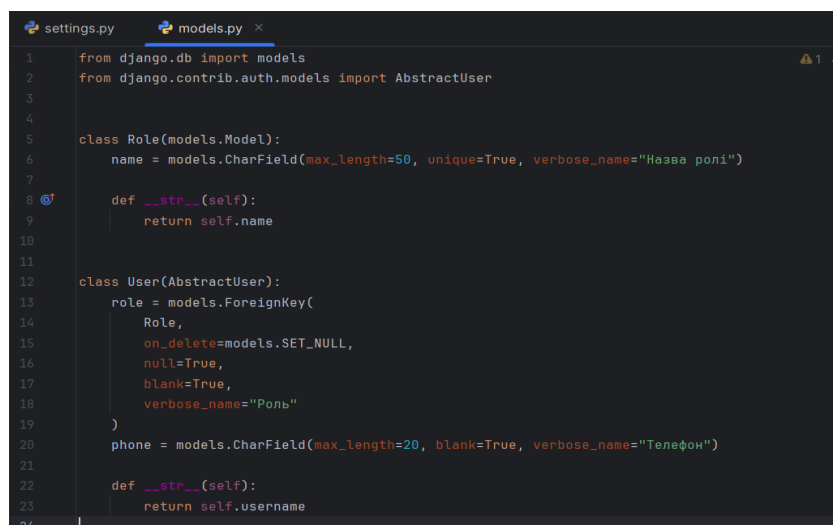
Наступним етапом налаштування є формування маршрутів у файлі `config/urls.py`. Такий підхід формує єдину точку входу в систему і забезпечує зв'язок між окремими функціональними модулями. Саме через централізований файл маршрутів система отримує узгоджену структуру переходів між сторінками, що є важливим для веб-застосунку з багатьма ролями доступу і кількома робочими сценаріями (рисунок 3.7).



```
1 from django.contrib import admin
2 from django.urls import path, include
3 from django.conf import settings
4 from django.conf.urls.static import static
5 from .views import home
6
7 urlpatterns = [
8     path(route: '', home, name='home'),
9     path(route: 'admin/', admin.site.urls),
10    path(route: 'users/', include('users.urls')),
11    path(route: 'requests/', include('requests_app.urls')),
12    path(route: 'consultations/', include('consultations_app.urls')),
13    path(route: 'messages/', include('messages_app.urls')),
14    path(route: 'files/', include('files_app.urls')),
15    path(route: 'notifications/', include('notifications_app.urls')),
16    path(route: 'audit/', include('audit_app.urls')),
17    path(route: 'schedule/', include('schedule_app.urls')),
18 ]
19
20 if settings.DEBUG:
21     urlpatterns += static(settings.MEDIA_URL, document_root=settings.MEDIA_ROOT)
```

Рисунок 3.7 – Реалізація загальної маршрутизації проекту

Після налаштування маршрутів було розроблено структуру даних для модуля користувачів у файлі `users/models.py`. Саме тут через створення моделей ролі та користувача закладається фундамент для розмежування прав доступу (рисунок 3.8).



```
1 from django.db import models
2 from django.contrib.auth.models import AbstractUser
3
4
5 class Role(models.Model):
6     name = models.CharField(max_length=50, unique=True, verbose_name="Назва ролі")
7
8     def __str__(self):
9         return self.name
10
11
12 class User(AbstractUser):
13     role = models.ForeignKey(
14         Role,
15         on_delete=models.SET_NULL,
16         null=True,
17         blank=True,
18         verbose_name="Роль"
19     )
20     phone = models.CharField(max_length=20, blank=True, verbose_name="Телефон")
21
22     def __str__(self):
23         return self.username
24
```

Рисунок 3.8 - Реалізація моделей користувачів у файлі `models.py`

Після створення базових моделей користувачів і ролей виконується формування облікового запису адміністратора системи та наповнення початкових довідкових даних.

Через адміністративну панель Django створюються перші ролі користувачів, які потім використовуються для розмежування прав доступу між клієнтами, консультантами та адміністраторами. Це дає можливість підготувати систему до подальшої реєстрації реальних користувачів і забезпечити коректну логіку авторизації залежно від ролі в системі (рисунк 3.9).

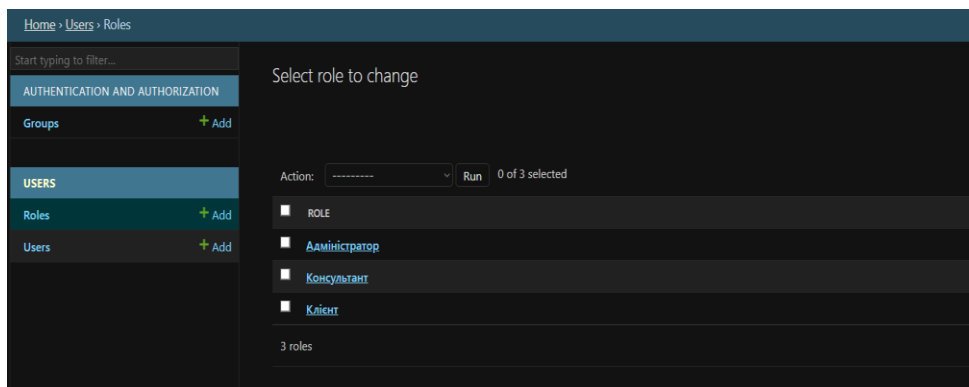


Рисунок 3.9 - Створених ролей користувачів в адміністративній панелі

За допомогою налаштувань fieldsets та add_fieldsets до форми редагування користувача додано блок додаткової інформації: роль користувача та номер телефону. (рисунк 3.10).

```
from django.contrib import admin
from django.contrib.auth.admin import UserAdmin
from .models import Role, User

@admin.register(Role)
class RoleAdmin(admin.ModelAdmin):
    list_display = ('name',)

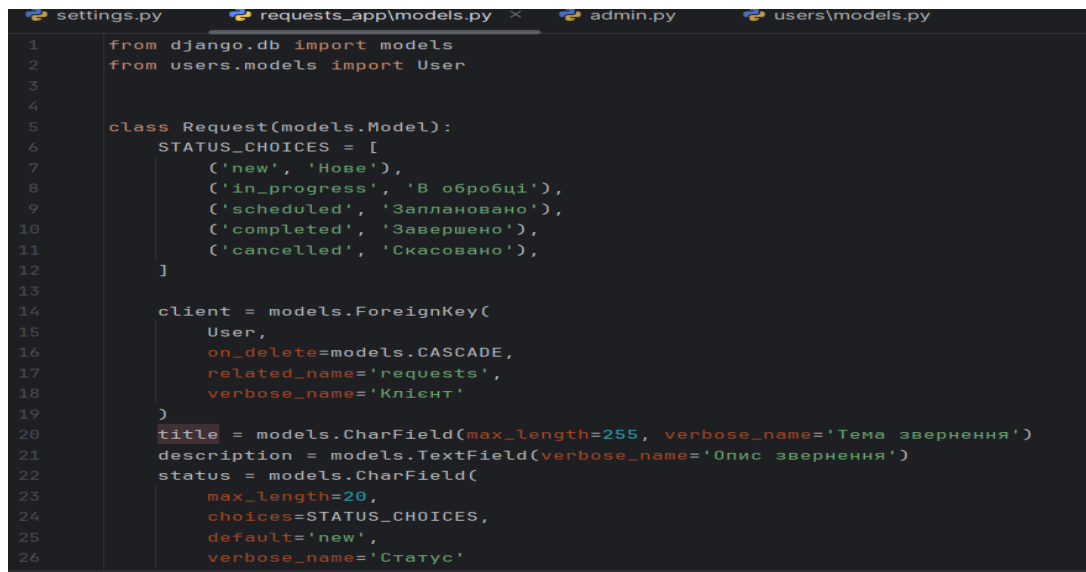
@admin.register(User)
class CustomUserAdmin(UserAdmin):
    fieldsets = UserAdmin.fieldsets + (
        ('Додаткова інформація', {
            'fields': ('role', 'phone'),
        }),
    )

    add_fieldsets = UserAdmin.add_fieldsets + (
        ('Додаткова інформація', {
            'fields': ('role', 'phone'),
        }),
    )

    list_display = ('username', 'email', 'role', 'is_staff', 'is_active')
```

Рисунок 3.10 - Кастомні класи RoleAdmin та CustomUserAdmin

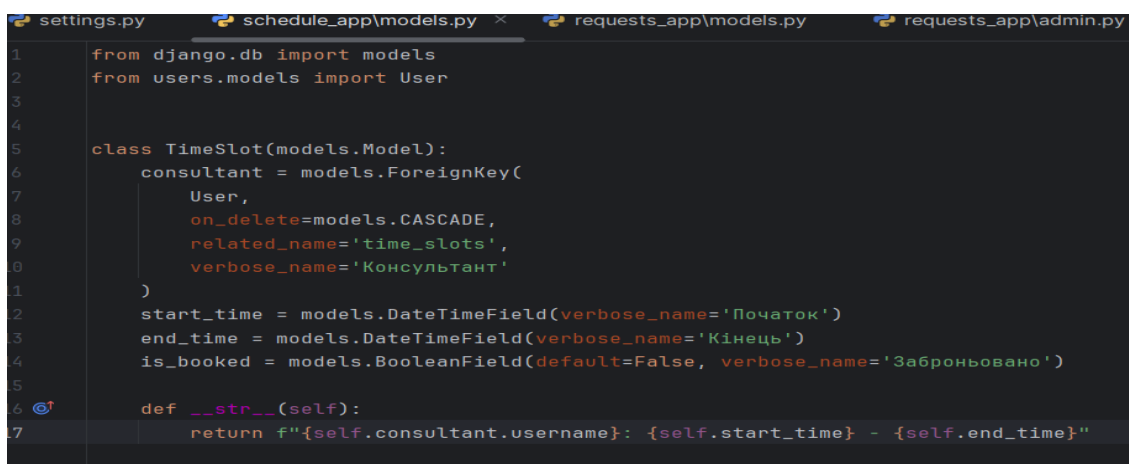
Наступним етапом програмної реалізації стало створення модуля для роботи зі зверненнями користувачів. Для цього було розроблено модель Request, яка забезпечує збереження основних параметрів первинного звернення (рисунок 3.11).



```
1 from django.db import models
2 from users.models import User
3
4
5 class Request(models.Model):
6     STATUS_CHOICES = [
7         ('new', 'Нове'),
8         ('in_progress', 'В обробці'),
9         ('scheduled', 'Заплановано'),
10        ('completed', 'Завершено'),
11        ('cancelled', 'Скасовано'),
12    ]
13
14    client = models.ForeignKey(
15        User,
16        on_delete=models.CASCADE,
17        related_name='requests',
18        verbose_name='Клієнт'
19    )
20    title = models.CharField(max_length=255, verbose_name='Тема звернення')
21    description = models.TextField(verbose_name='Опис звернення')
22    status = models.CharField(
23        max_length=20,
24        choices=STATUS_CHOICES,
25        default='new',
26        verbose_name='Статус'
```

Рисунок 3.11 - Структури моделі звернення

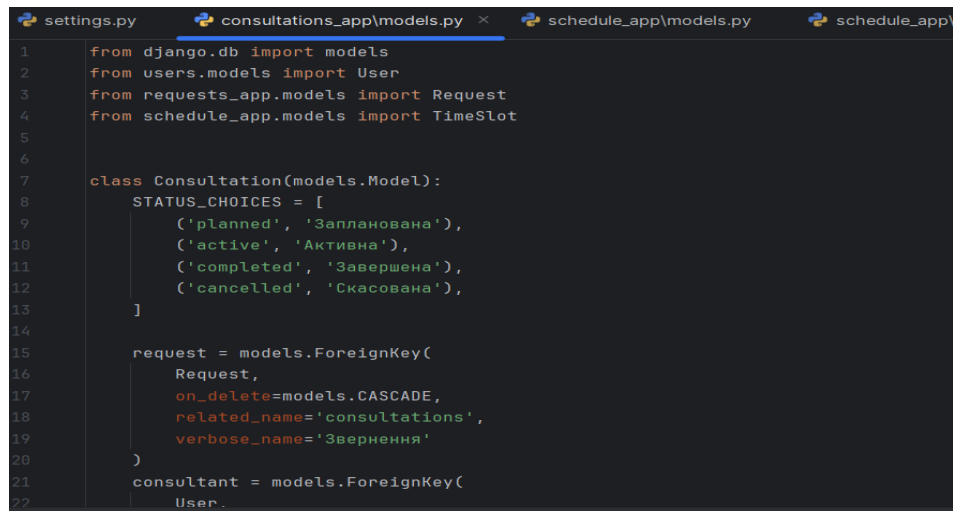
Для організації графіка роботи консультантів у структурі проекту створено модуль schedule_app. Його використання дозволило відокремити функціонал планування від основних звернень, забезпечуючи гнучке налаштування робочих годин та автоматичну перевірку доступності консультанта для проведення сесії (рисунок 3.12).



```
1 from django.db import models
2 from users.models import User
3
4
5 class TimeSlot(models.Model):
6     consultant = models.ForeignKey(
7         User,
8         on_delete=models.CASCADE,
9         related_name='time_slots',
10        verbose_name='Консультант'
11    )
12    start_time = models.DateTimeField(verbose_name='Початок')
13    end_time = models.DateTimeField(verbose_name='Кінець')
14    is_booked = models.BooleanField(default=False, verbose_name='Заброньовано')
15
16    def __str__(self):
17        return f"{self.consultant.username}: {self.start_time} - {self.end_time}"
```

Рисунок 3.12 - Модуля для керування часовими інтервалами консультацій

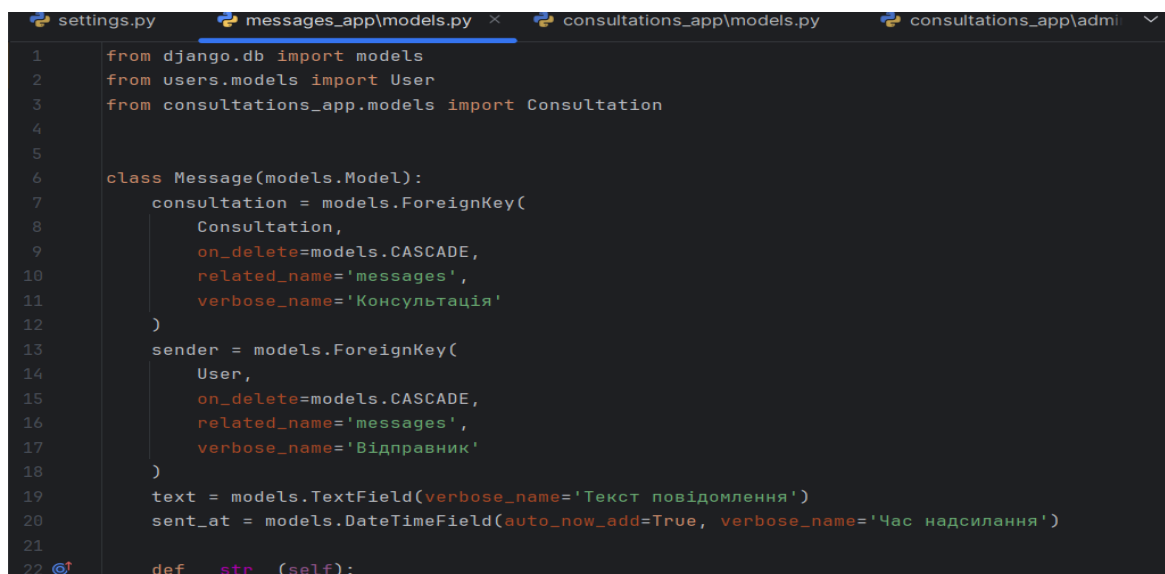
Після цього було створено окремий застосунок, який відповідає за реалізацію безпосереднього процесу консультації. За допомогою цього модуля формується зв'язок між зверненням, консультантом, клієнтом і вибраним часовим слотом. Саме тут зберігаються статус консультації, посилання на відеозустріч та інші службові параметри, необхідні для супроводу сесії (рисунок 3.13).



```
1 from django.db import models
2 from users.models import User
3 from requests_app.models import Request
4 from schedule_app.models import TimeSlot
5
6
7 class Consultation(models.Model):
8     STATUS_CHOICES = [
9         ('planned', 'Запланована'),
10        ('active', 'Активна'),
11        ('completed', 'Завершена'),
12        ('cancelled', 'Скасована'),
13    ]
14
15    request = models.ForeignKey(
16        Request,
17        on_delete=models.CASCADE,
18        related_name='consultations',
19        verbose_name='Звернення'
20    )
21    consultant = models.ForeignKey(
22        User,
```

Рисунок 3.13 - Створення модуля керування активними консультаціями

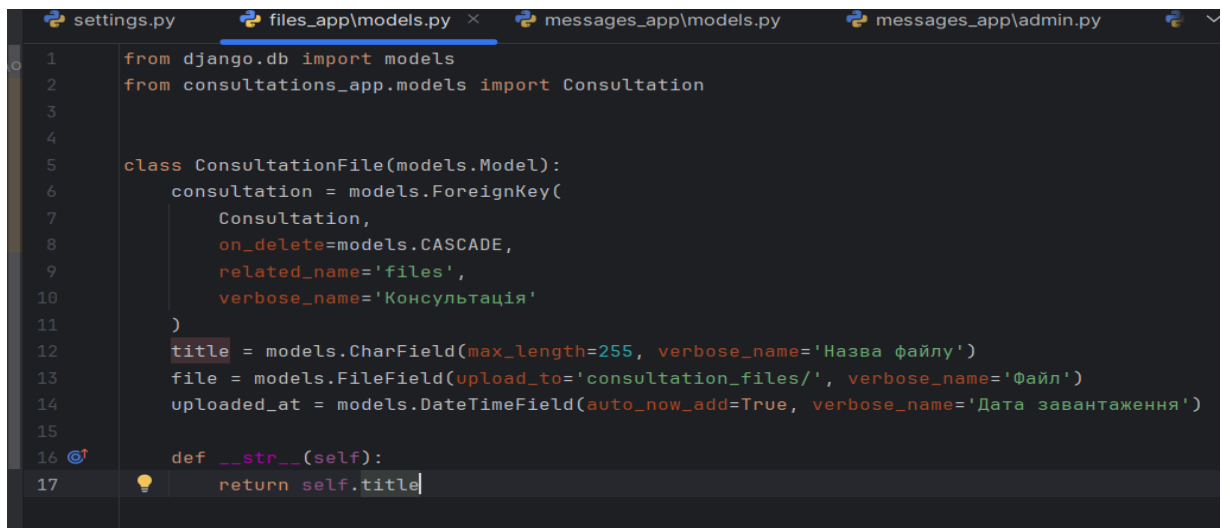
Для обміну текстовими повідомленнями реалізовано модуль messages_app. Він зберігає повідомлення, пов'язує їх із конкретною консультацією та дає змогу відображати історію діалогу між клієнтом і консультантом (рисунок 3.14).



```
1 from django.db import models
2 from users.models import User
3 from consultations_app.models import Consultation
4
5
6 class Message(models.Model):
7     consultation = models.ForeignKey(
8         Consultation,
9         on_delete=models.CASCADE,
10        related_name='messages',
11        verbose_name='Консультація'
12    )
13    sender = models.ForeignKey(
14        User,
15        on_delete=models.CASCADE,
16        related_name='messages',
17        verbose_name='Відправник'
18    )
19    text = models.TextField(verbose_name='Текст повідомлення')
20    sent_at = models.DateTimeField(auto_now_add=True, verbose_name='Час надсилення')
21
22    def __str__(self):
```

Рисунок 3.14 - Структури для системи внутрішніх повідомлень

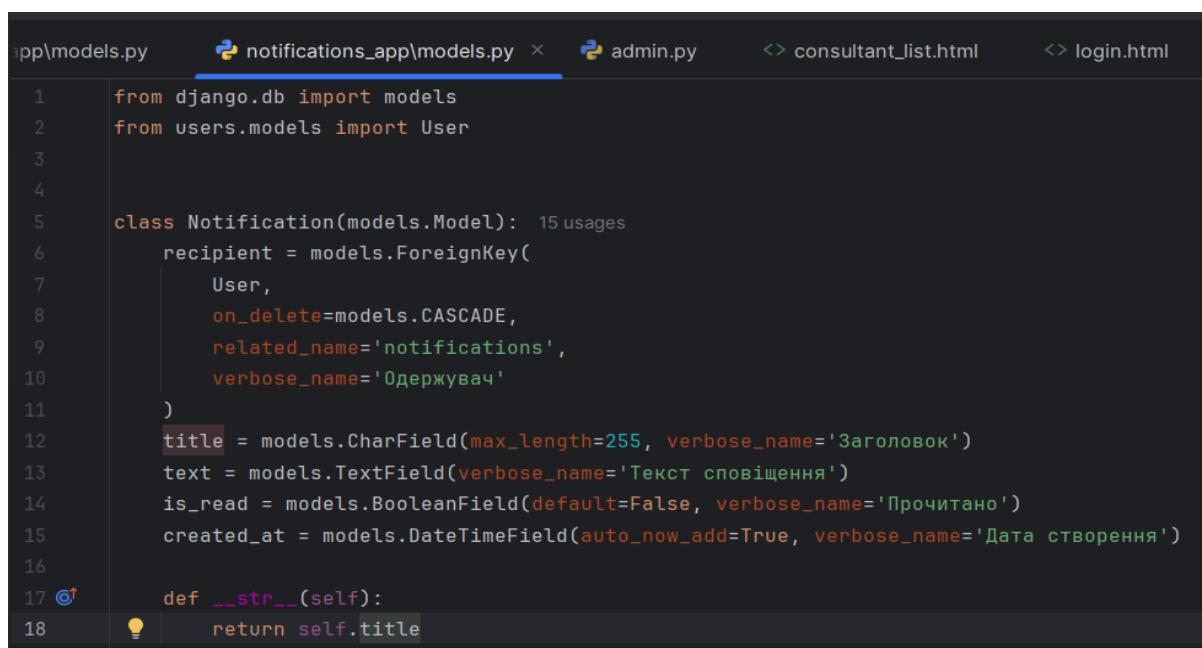
Для роботи з файлами створено окремий модуль `files_app`. У ньому реалізовано завантаження, збереження та перегляд матеріалів, пов'язаних із конкретною консультацією (рисунок 3.15).



```
1 from django.db import models
2 from consultations_app.models import Consultation
3
4
5 class ConsultationFile(models.Model):
6     consultation = models.ForeignKey(
7         Consultation,
8         on_delete=models.CASCADE,
9         related_name='files',
10        verbose_name='Консультація'
11    )
12     title = models.CharField(max_length=255, verbose_name='Назва файлу')
13     file = models.FileField(upload_to='consultation_files/', verbose_name='Файл')
14     uploaded_at = models.DateTimeField(auto_now_add=True, verbose_name='Дата завантаження')
15
16     def __str__(self):
17         return self.title
```

Рисунок 3.15 - Реалізація модуля роботи з файлами

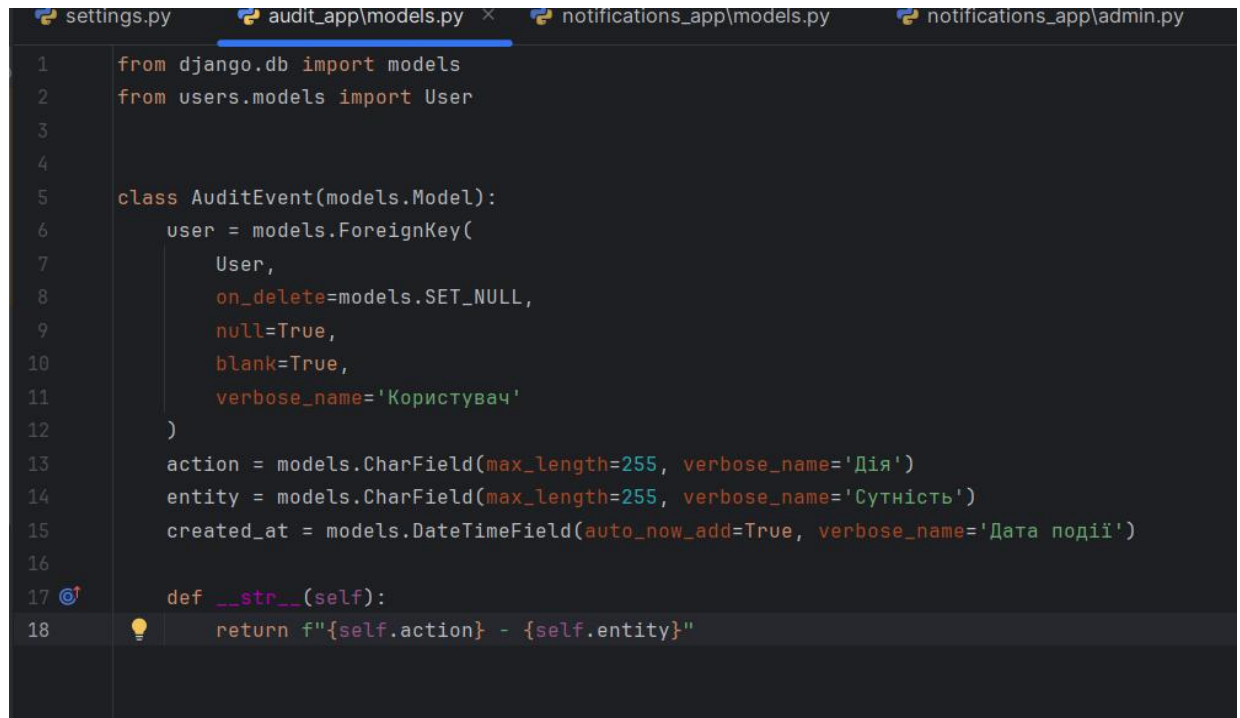
Наступним кроком було додано модуль, який забезпечує централізовану роботу внутрішніх сповіщень. У межах цього застосунку реалізовано збереження службових повідомлень про нові консультації, зміну статусу, додавання файлів або інші важливі події. (рисунок 3.16).



```
1 from django.db import models
2 from users.models import User
3
4
5 class Notification(models.Model): 15 usages
6     recipient = models.ForeignKey(
7         User,
8         on_delete=models.CASCADE,
9         related_name='notifications',
10        verbose_name='Одержувач'
11    )
12     title = models.CharField(max_length=255, verbose_name='Заголовок')
13     text = models.TextField(verbose_name='Текст сповіщення')
14     is_read = models.BooleanField(default=False, verbose_name='Прочитано')
15     created_at = models.DateTimeField(auto_now_add=True, verbose_name='Дата створення')
16
17     def __str__(self):
18         return self.title
```

Рисунок 3.16 - Структури для сервісу внутрішніх сповіщень

Для фіксації ключових дій користувачів створено модуль `audit_app`, який виконує функції журналу подій. В ньому зберігаються записи про створення звернень, запис на консультацію, зміну статусу, надсилення повідомлень, завантаження файлів та інші суттєві дії. Це дає змогу контролювати історію змін у системі, підвищує прозорість роботи користувачів і створює основу для подальшого аудиту активності (рисунк 3.17).



```
1 from django.db import models
2 from users.models import User
3
4
5 class AuditEvent(models.Model):
6     user = models.ForeignKey(
7         User,
8         on_delete=models.SET_NULL,
9         null=True,
10        blank=True,
11        verbose_name='Користувач'
12    )
13    action = models.CharField(max_length=255, verbose_name='Дія')
14    entity = models.CharField(max_length=255, verbose_name='Сутність')
15    created_at = models.DateTimeField(auto_now_add=True, verbose_name='Дата події')
16
17    def __str__(self):
18        return f"{self.action} - {self.entity}"
```

Рисунок 3.17 - Модуля журналювання та аудиту дій користувачів

Після формування основних серверних модулів виконується побудова шаблонної частини веб-застосунку. На рівні інтерфейсу сторінка `home.html` виконує роль стартової панелі інформаційної системи. На цій сторінці розміщено короткий опис призначення системи, навігаційні елементи та переходи до основних модулів: звернень, консультацій і розкладу.

Головна сторінка формує перше уявлення про систему, забезпечує зручний початок роботи та виступає центральною точкою доступу до основних сценаріїв взаємодії. Вона також візуально об'єднує основні сервіси платформи, що відповідає логіці модульної побудови системи та підтримує послідовний перехід користувача від первинного звернення до консультації (рисунк 3.18).

```

<style>
</style>

<div class="home-clean">
  <div class="home-clean-shell">

    <section class="home-hero-clean">
      <div class="home-hero-grid">
        <div class="hero-left">
          <div class="hero-badge">Онлайн-консультації</div>

          <h1 class="hero-title">
            Зручний простір для звернень, запису та онлайн-консультацій
          </h1>

          <div class="hero-actions">
            <a href="/consultations/" class="hero-btn-secondary">
              Мої консультації
            </a>
            <a href="/schedule/" class="hero-btn-link">
              Перейти до розкладу →
            </a>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>

```

Рисунок 3.18 – Частина коду головної сторінки інформаційної системи

Базовою шаблонною частиною веб-застосунку є файл `base.html`, який містить спільну структуру всіх сторінок системи. В ньому реалізовано секцію, підключення Bootstrap і власного файлу стилів `main.css`, верхнє меню навігації, а також службові блоки для підстановки індивідуального вмісту сторінок. На основі цього шаблону побудовано інші HTML-сторінки системи, зокрема `login.html`, `register.html`, `request_list.html`, `request_create.html`, `consultation_list.html`, `consultation_create.html`, `consultation_detail.html`, `slot_list.html`, `notification_list.html` та `audit_list.html` (рисунок 3.19).

```

1  {% load static %}
2  <!DOCTYPE html>
3  <html lang="uk">
4  <head>
5    <meta charset="UTF-8">
6    <meta name="viewport" content="width=device-width, initial-scale=1.0">
7    <title>{% block title %}Онлайн-консультації{% endblock %}</title>
8
9    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bootstrap.min.css" rel="stylesheet">
10   <link rel="stylesheet" href="{% static 'css/main.css' %}">
11
12   <style>
13     :root {
14       --bg-main: #f8f4ee;
15       --bg-card: #ffffff;
16       --bg-soft: #fffaf4;
17       --bg-warm: #f5e6e0;
18       --bg-lavender: #f1efff;
19       --bg-blue-soft: #eaf2ff;
20       --border-soft: #e6ddd2;
21       --text-main: #1d3557;
22       --text-muted: #6f7b8f;
23       --primary: #4a7c9f;
24       --primary-dark: #3467f2;
25       --shadow-soft: 0 12px 30px rgba(58, 42, 24, 0.06);
26       --radius-xl: 24px;

```

Рисунок 3.19 – Базовий HTML-шаблон веб-застосунку

Така організація забезпечує єдність дизайну, повторне використання однакових елементів інтерфейсу, спрощує навігацію між функціональними модулями та підвищує зручність подальшого супроводу вебзастосунку. Саме використання базового шаблону дало змогу реалізувати цілісний інтерфейс для клієнта, консультанта та адміністратора без дублювання однакових елементів на кожній окремій сторінці.

3.3 Тестування функціональних сценаріїв роботи системи

Після реалізації основних модулів було проведено тестування функцій застосунку. На цьому етапі перевірялася правильність виконання типових сценаріїв користувача.

Під час тестування було перевірено, що після успішної авторизації користувача система коректно відкриває головну сторінку та відображає активний обліковий запис у верхній панелі. На рисунку 3.20 зображено результат входу.

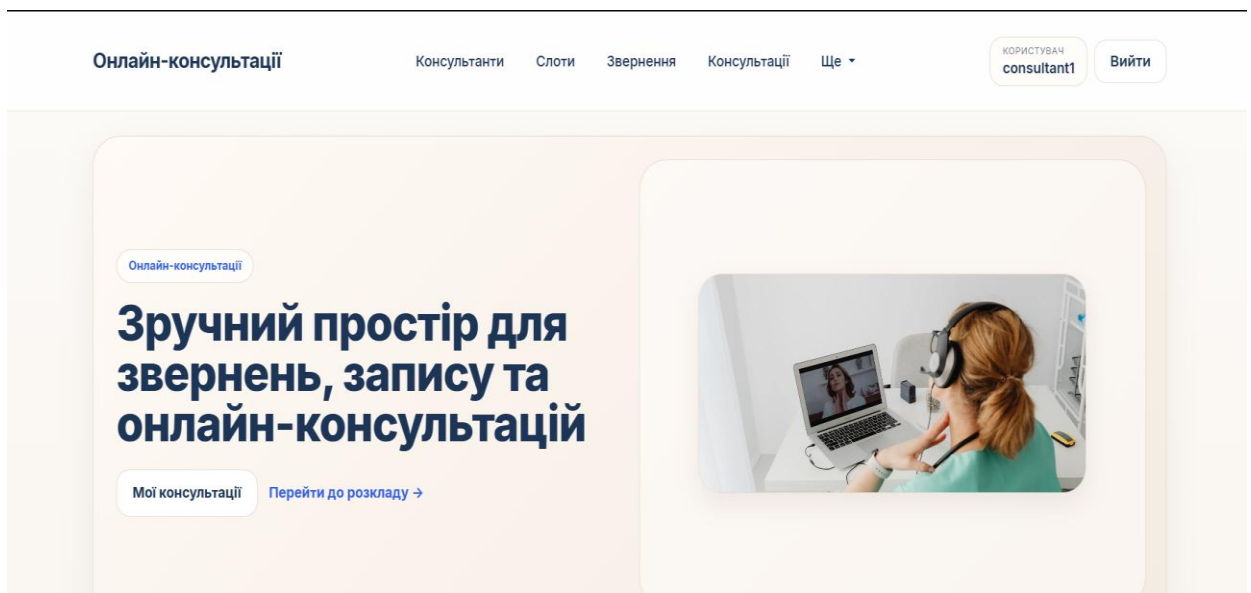


Рисунок 3.20 – Перевірка входу користувача в систему за ролями

Наступним етапом було тестування сценарію створення звернення клієнтом. На цьому кроці заповнювалася форма створення запиту. На рисунку 3.21 зображено створення запиту.

Крок 1

Вкажіть назву

Дайте зверненню коротку і зрозумілу назву.

Крок 2

Опишіть ситуацію

Коротко поясніть суть запиту або проблеми.

Крок 3

Збережіть звернення

Після цього можна буде перейти до створення консультації.

Форма

Заповніть дані звернення

Усі поля заповнюються у простій формі нижче.

Тема звернення

Безсоння

Опис звернення

Безсоння через постійні сни пережитих ситуацій

Зберегти звернення

Скасувати

Пояснення

Навіщо потрібне звернення

Звернення є початковим етапом роботи користувача в системі. На його основі далі можна створити консультацію та обрати зручний слот.

Швидкі дії

До списку звернень

До консультацій

Рисунок 3.21 – Перевірка створення нового звернення

Після збереження запиту виконувалася перевірка його відображення в загальному переліку. На рисунку 3.22 зображено результат збереження.

Звернення

Список звернень

Створити звернення

На головну

Перегляд створених звернень та швидкий перехід до створення консультації.

Усього звернень

3

Усі звернення, доступні на цій сторінці.

Швидка дія

Нове звернення

Створіть новий запит для подальшого запису на консультацію.

Створити

Далі по процесу

Консультації та розклад

Після створення звернення можна перейти до бронювання консультації та вибору вільного слота.

ID	НАЗВА	СТАТУС	ДАТА СТВОРЕННЯ	ДІЯ
#4	Безсоння Безсоння через постійні сни пережитих ситуацій	Ново	30.03.2026 19:13	Створити консультацію
#3	ПТСР Важкий стан і не контрольованість	Ново	30.03.2026 19:07	Створити консультацію
#2	тест_звер_2 Допомога ПТРС	Ново	25.03.2026 19:37	Створити консультацію

Рисунок 3.22 – Успішне збереження запиту

Після створення звернення користувач може перейти до запису на консультацію. Під час проходження цього сценарію користувач переходить зі сторінки розкладу до форми створення консультації, де слот і консультант автоматично підставлялися у форму запису. На рисунку 3.23 зображено запис користувача на консультацію.

Онлайн-консультації

Консультанти Слоти Звернення **Консультації** Ще ▾

КОРИСТУВАЧ consultant1 Вийти

Крок 1

Оберіть звернення

Спочатку виберіть заявку, для якої створюється консультація.

Крок 2

Вкажіть консультанта

Система покаже доступного спеціаліста, якого можна призначити.

Крок 3

Оберіть слот

Виберіть вільний час із розкладу та підтвердьте створення консультації.

Форма запису

Заповніть дані консультації

Після збереження консультація з'явиться у списку та буде доступна на сторінці деталей.

Звернення: **Безсоння**

Консультант: **consultant1**

Часовий слот: **consultant1: 2026-03-28 18:00:00+00:00 - 2026-03-29 00:00:00+00:00**

Посилання на консультацію: **https://meet.jit.si/test-consultation-room**

Створити консультацію Скасувати

Підказка

Як працює запис

Для створення консультації потрібно вибрати пов'язане звернення, консультанта та доступний часовий слот.

Якщо слот уже був обраний зі сторінки розкладу, поле часу може бути вже заповнене автоматично.

Швидкі дії

Відкрити розклад

Перейти до списку

Рисунок 3.23 – Запис користувача на вільний часовий слот

Після підтвердження консультації система створює відповідний запис і змінює стан слота на зайнятий. Додатково було перевірено, що система блокує створення консультації у випадку вибору некоректного часового слота, не допускаючи порушення логіки бронювання. На рисунку 3.24 зображено некоректний запис.

Онлайн-консультації

Консультанти Слоти Звернення **Консультації** Ще ▾

КОРИСТУВАЧ consultant1 Вийти

Оберіть звернення

Спочатку виберіть заявку, для якої створюється консультація.

Вкажіть консультанта

Система покаже доступного спеціаліста, якого можна призначити.

Оберіть слот

Виберіть вільний час із розкладу та підтвердьте створення консультації.

Форма запису

Заповніть дані консультації

Після збереження консультація з'явиться у списку та буде доступна на сторінці деталей.

Обраний часовий слот не належить вибраному консультанту.

Звернення: **Безсоння**

Консультант: **consultant1**

Часовий слот: **admin: 2026-03-31 12:00:00+00:00 - 2026-03-31 15:00:00+00:00**

Посилання на консультацію:

Створити консультацію Скасувати

Підказка

Як працює запис

Для створення консультації потрібно вибрати пов'язане звернення, консультанта та доступний часовий слот.

Якщо слот уже був обраний зі сторінки розкладу, поле часу може бути вже заповнене автоматично.

Швидкі дії

Відкрити розклад

Перейти до списку

Рисунок 3.24 – Блокування створення консультації

Після створення консультації було виконано перевірку сторінки детальної консультації, де реалізовано обмін повідомленнями та файлами між учасниками сесії. Було перевірено надсилання текстового повідомлення, його відображення в історії консультації, а також завантаження файлу з подальшим відображенням у списку вкладень. На рисунку 3.25 зображено надсилання повідомлень

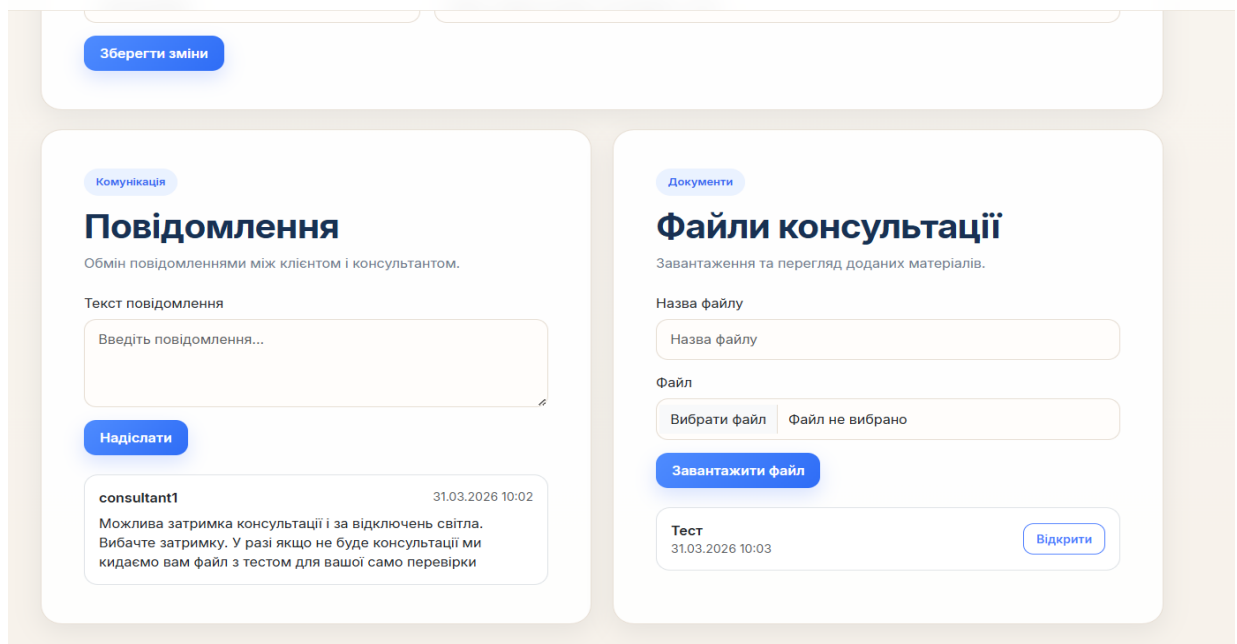


Рисунок 3.25 – Надсилання повідомлення і файлу в консультації

Під час тестування було підтверджено працездатність основних функціональних модулів інформаційної системи онлайн-консультацій. Перевірка показала, що система коректно виконує авторизацію користувачів, створення звернень, запис на консультацію та роботу з часовими слотами.

Окремо підтверджено, що система блокує некоректне бронювання та не дозволяє створювати консультації для недоступних часових слотів.

Підтверджено коректну роботу сторінки детальної консультації, обмін повідомленнями та завантаження файлів.

ВИСНОВКИ

Під час виконання кваліфікаційної роботи розроблено та протестовано інформаційну систему надання онлайн-консультацій для осіб, які постраждали від війни. Система забезпечує створення звернень, підбір консультанта, запис на консультацію, обмін повідомленнями та файлами, формування сповіщень, ведення журналу подій і підтримку основних сценаріїв дистанційної взаємодії між клієнтом, консультантом та адміністратором.

1. Проведено аналіз предметної області онлайн-консультування для осіб, які постраждали від війни. Визначено, що в умовах воєнного стану важливими є доступність допомоги, простота звернення, конфіденційність даних, наявність зрозумілого маршруту користувача та можливість отримання підтримки незалежно від місця перебування.

2. Виконано огляд і аналіз існуючих рішень дистанційної консультаційної підтримки, таких як BetterHelp, Talkspace, Doxy.me та «ПОРУЧ». Встановлено, що комерційні платформи мають розвинені функції автоматизації, але не завжди адаптовані до гуманітарних умов, а локальні ініціативи мають низький поріг входу.

3. Сформульовано постановку задачі, яка полягає в розробці інформаційної системи для організації повного циклу онлайн-консультування.

4. Розроблено загальну структуру проектованої системи. Визначено основні складові системи: клієнтську частину, серверну частину, базу даних, файлове сховище та модуль інтеграції із зовнішніми сервісами.

5. Запропоновано функціональну модель системи, яка включає модулі керування користувачами і ролями, профілями консультантів, зверненнями, календарем і бронюванням, консультаціями, повідомленнями, файлами, сповіщеннями та журналюванням подій.

6. Описано алгоритми створення звернення, первинної маршрутизації, підбору консультанта, запису на консультацію, перевірки доступу до сесії, проведення консультації, завершення взаємодії та надсилання форми зворотного зв'язку.

7. Виділено основні вхідні та вихідні дані, побудовано інфологічну модель, визначено основні сутності предметної області та сформовано ER-діаграму бази даних.

8. Обґрунтовано вибір програмно-технологічних засобів реалізації. Для серверної частини обрано Django, для зберігання даних — PostgreSQL, для побудови інтерфейсу — HTML, CSS і Bootstrap. Для організації відеозв'язку передбачено використання Jitsi Meet, а для хмарного розгортання — платформу Render.

9. Реалізовано програмно-технологічну частину системи у вигляді модульного Django-застосунку. Створено окремі застосунки `users`, `requests_app`, `consultations_app`, `schedule_app`, `messages_app`, `files_app`, `notifications_app` та `audit_app`, що забезпечують розділення логіки роботи із користувачами, зверненнями, консультаціями, розкладом, повідомленнями, файлами, сповіщеннями та журналом дій.

10. Перевірено авторизацію користувачів, створення звернення, відображення збереженого запиту, запис на вільний часовий слот, блокування некоректного бронювання, роботу сторінки детальної консультації, надсилання повідомлень і завантаження файлів.

Результати тестування підтвердили, що основні модулі системи працюють коректно та забезпечують виконання передбачених сценаріїв. Розроблена система є практичним програмним рішенням для організації дистанційної консультаційної допомоги особам, які постраждали від війни, та може бути адаптована для використання в консультаційних центрах, громадських організаціях, волонтерських ініціативах і установах соціальної підтримки.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Inter-Agency Standing Committee. *IASC Guidelines on Mental Health and Psychosocial Support in Emergency Settings*. 2007. URL: <https://interagencystandingcommittee.org/iasc-task-force-mental-health-and-psychosocial-support-emergency-settings/iasc-guidelines-mental-health-and-psychosocial-support-emergency-settings-2007> (дата звернення: 12.03.2026).
2. Inter-Agency Standing Committee. *IASC Guidance on Addressing Suicide in Humanitarian Settings*. 2022. URL: <https://interagencystandingcommittee.org/iasc-reference-group-mental-health-and-psychosocial-support-emergency-settings/iasc-guidance-addressing-suicide-humanitarian-settings> (дата звернення: 12.03.2026).
3. World Health Organization, War Trauma Foundation, World Vision International. *Psychological first aid: guide for field workers*. Geneva : *World Health Organization*, 2011. URL: <https://www.who.int/publications/i/item/9789241548205> (дата звернення: 12.03.2026).
4. American Psychological Association. *Guidelines for the Practice of Telepsychology*. 2024. URL: <https://www.apa.org/practice/guidelines/telepsychology-revision.pdf> (дата звернення: 12.03.2026).
5. Joint Task Force for the Development of Telepsychology Guidelines for Psychologists. *Guidelines for the practice of telepsychology. American Psychologist*. 2013. Vol. 68, № 9. P. 791–800. DOI: 10.1037/a0035001. URL: <https://www.apa.org/pubs/journals/features/amp-a0035001.pdf> (дата звернення: 12.03.2026).
6. World Psychiatric Association. *Telepsychiatry Global Guidelines*. 2021. URL: https://wpanet.org/wp-content/uploads/2025/07/FINAL_2021_WPA_Telepsychiatry-Guidelines-23Sep202111001.pdf (дата звернення: 12.03.2026).
7. World Health Organization. *Strengthening National Capacity for Mental Health and Psychosocial Support During the War: WHO support to Ukraine in 2022*. 2023. URL: https://cdn.who.int/media/docs/default-source/mental-health/special-initiative/who-mhpss-support-to-ukraine_feb_2023.pdf (дата звернення: 12.03.2026).

8. Tng G. Y. Q., Koh J., Soh X. C., Majeed N. M., Hartanto A. Efficacy of digital mental health interventions for PTSD symptoms: a systematic review of meta-analyses. *Journal of Affective Disorders*. 2024. Vol. 357. P. 23–36. DOI: 10.1016/j.jad.2024.04.074. URL: <https://pubmed.ncbi.nlm.nih.gov/38679205/> (дата звернення: 12.03.2026).
9. Andersson G. Internet interventions: past, present and future. *Internet Interventions*. 2018. Vol. 12. P. 181–188. DOI: 10.1016/j.invent.2018.03.008.
10. Mohr D. C., Riper H., Schueller S. M. A Solution-Focused Research Approach to Achieve an Implementable Revolution in Digital Mental Health. *JAMA Psychiatry*. 2018. Vol. 75, № 2. P. 113–114. DOI: 10.1001/jamapsychiatry.2017.3838.
11. Torous J., Jän Myrick K., Rauseo-Ricupero N., Firth J. Digital Mental Health and COVID-19: Using Technology Today to Accelerate the Curve on Access and Quality Tomorrow. *JMIR Mental Health*. 2020. Vol. 7, № 3. e18848. DOI: 10.2196/18848.
12. BetterHelp. Professional Therapy With A Licensed Therapist. URL: <https://www.betterhelp.com/> (дата звернення: 12.03.2026).
13. Federal Trade Commission. BetterHelp, Inc., In the Matter of. 2023. URL: <https://www.ftc.gov/legal-library/browse/cases-proceedings/2023169-betterhelp-inc-matter> (дата звернення: 12.03.2026).
14. Bajak F. BetterHelp shared users' sensitive health data, FTC says. *AP News*. 2023. URL: <https://apnews.com/article/betterhelp-ftc-health-data-privacy-befca40bb873661d1f8986bb75d8df07> (дата звернення: 12.03.2026).
15. Talkspace. Online Therapy and Psychiatry. URL: <https://www.talkspace.com/> (дата звернення: 12.03.2026).
16. Talkspace. Is Talkspace Safe, Secure, and Confidential? URL: <https://help.talkspace.com/hc/en-us/articles/360000286786-Is-Talkspace-Safe-Secure-and-Confidential> (дата звернення: 12.03.2026).
17. Doxy.me. The world's most loved telemedicine platform. URL: <https://doxy.me/> (дата звернення: 12.03.2026).

18. Doxy.me Help Center. Waiting room: overview. URL: <https://help.doxy.me/en/articles/95906-waiting-room-overview> (дата звернення: 12.03.2026).
19. Doxy.me. Security you can trust from doxy.me. URL: <https://doxy.me/en/security> (дата звернення: 12.03.2026).
20. ПОРУЧ — проект психологічної підтримки. URL: <https://poruch.me/> (дата звернення: 12.03.2026).
21. Міністерство охорони здоров'я України. Як під час війни отримати безоплатну психологічну та психіатричну допомогу. 2024. URL: <https://moz.gov.ua/uk/jak-pid-chas-vijni-otrimati-bezoplatnu-psihologichnu-ta-psihiatrichnu-dopomogu-> (дата звернення: 12.03.2026).
22. Всеукраїнська програма ментального здоров'я «Ти як?». URL: <https://howareu.com/> (дата звернення: 12.03.2026).
23. Mental Help. Безкоштовна професійна психологічна допомога онлайн. URL: <https://mentalhelp.com.ua/> (дата звернення: 12.03.2026).
24. Національна психологічна асоціація України. FAQ. Проведення онлайн-консультацій. URL: <https://npa-ua.org/pub/files/afa42a5247b282ca.pdf> (дата звернення: 12.03.2026).
25. Django Software Foundation. Django documentation. URL: <https://docs.djangoproject.com/> (дата звернення: 12.03.2026).
26. Django Software Foundation. Using the Django authentication system. URL: <https://docs.djangoproject.com/en/6.0/topics/auth/default/> (дата звернення: 12.03.2026).
27. Django Software Foundation. Models. URL: <https://docs.djangoproject.com/en/6.0/topics/db/models/> (дата звернення: 12.03.2026).
28. Django Software Foundation. Templates. URL: <https://docs.djangoproject.com/en/6.0/topics/templates/> (дата звернення: 12.03.2026).
29. Django Software Foundation. Testing tools. URL: <https://docs.djangoproject.com/en/6.0/topics/testing/tools/> (дата звернення: 12.03.2026).

30. PostgreSQL Global Development Group. PostgreSQL Documentation: Constraints. URL: <https://www.postgresql.org/docs/current/ddl-constraints.html> (дата звернення: 12.03.2026).

31. PostgreSQL Global Development Group. PostgreSQL Documentation: Foreign Keys. URL: <https://www.postgresql.org/docs/current/tutorial-fk.html> (дата звернення: 12.03.2026).

32. Bootstrap. Grid system. URL: <https://getbootstrap.com/docs/5.0/layout/grid/> (дата звернення: 12.03.2026).

33. Bootstrap. Forms. URL: <https://getbootstrap.com/docs/5.0/forms/overview/> (дата звернення: 12.03.2026).

34. Jitsi Meet Handbook. IFrame API. URL: <https://jitsi.github.io/handbook/docs/dev-guide/dev-guide-iframe/> (дата звернення: 12.03.2026).

35. Render. Deploy a Django App on Render. URL: <https://render.com/docs/deploy-django> (дата звернення: 12.03.2026).

36. OWASP Foundation. OWASP Top Ten Web Application Security Risks. URL: <https://owasp.org/www-project-top-ten/> (дата звернення: 12.03.2026).

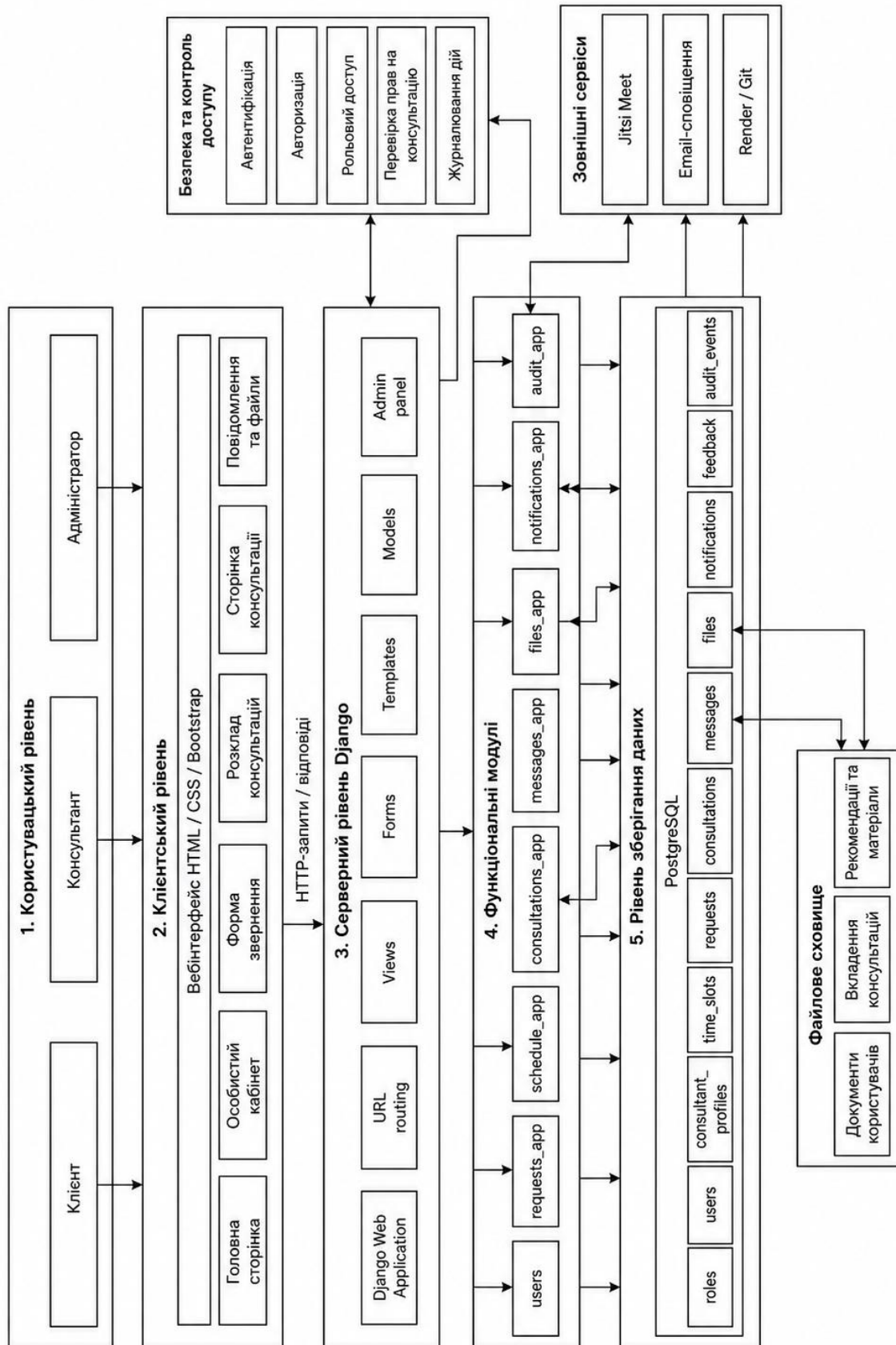
37. OWASP Foundation. File Upload Cheat Sheet. URL: https://cheatsheetseries.owasp.org/cheatsheets/File_Upload_Cheat_Sheet.html (дата звернення: 12.03.2026).

38. Комар М.П., Саченко А.О., Васильків Н.М., Гладій Г.М., Коваль В.С., Ліп'яніна-Гончаренко Х.В. Методичні рекомендації до виконання кваліфікаційної роботи з освітньо-професійної програми «Комп'ютерні науки» спеціальності 122 «Комп'ютерні науки» за першим (бакалаврським) рівнем вищої освіти. Тернопіль: ЗУНУ, 2024. 52 с.

39. Прончук Д.С. Інформаційна система надання онлайн-консультацій для осіб, які постраждали від війни // Традиційні та інноваційні підходи до наукових досліджень : матеріали XI Міжнародної наукової конференції, м. Луцьк, Україна. Луцьк, 2026. (прийнята до друку).

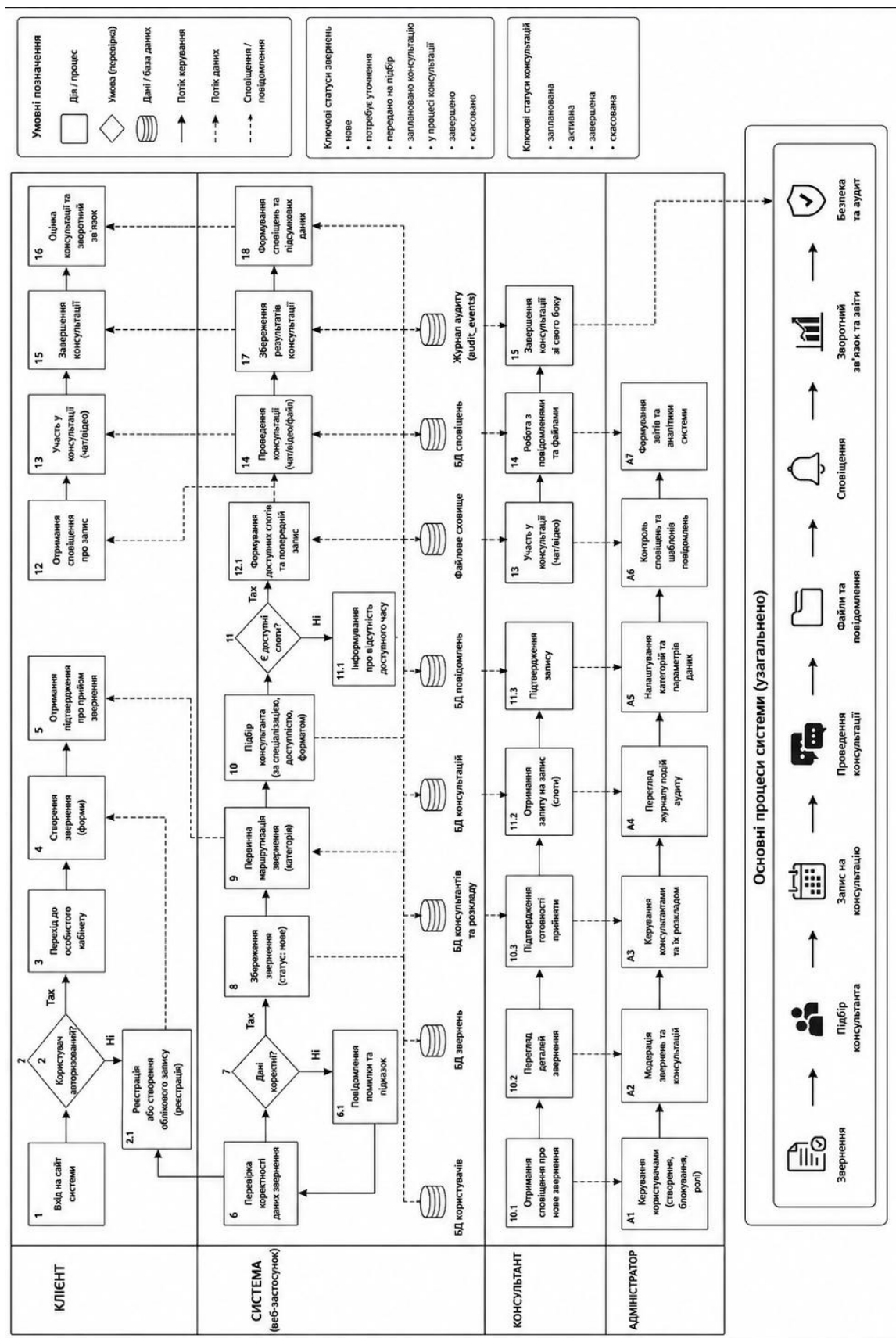
Додаток А

Деталізована архітектура інформаційної системи онлайн-консультування



Додаток Б

Узагальнена схема основних процесів інформаційної системи



Додаток В

Код програмної реалізації інформаційної системи онлайн-консультацій

Файл users/models.py - полі та розширена модель користувача.

```
from django.db import models
from django.contrib.auth.models import AbstractUser
```

```
class Role(models.Model):
    name = models.CharField(
        max_length=50,
        unique=True,
        verbose_name="Назва полі"
    )
    def __str__(self):
        return self.name
```

```
class User(AbstractUser):
    role = models.ForeignKey(
        Role,
        on_delete=models.SET_NULL,
        null=True,
        blank=True,
        verbose_name="Роль"
    )
    phone = models.CharField(
        max_length=20,
        blank=True,
        verbose_name="Телефон"
    )
    def __str__(self):
        return self.username
```

Файл users/admin.py - налаштування адміністративної панелі для моделей Role та User.

```
from django.contrib import admin
from django.contrib.auth.admin import UserAdmin
from .models import Role, User
@admin.register(Role)
class RoleAdmin(admin.ModelAdmin):
    list_display = ('name',)
@admin.register(User)
class CustomUserAdmin(UserAdmin):
    fieldsets = UserAdmin.fieldsets
    + (
        ('Додаткова інформація', {
            'fields': ('role',
            'phone'),
        }),
    )
    add_fieldsets =
    UserAdmin.add_fieldsets + (
        ('Додаткова інформація', {
            'fields': ('role',
            'phone'),
        }),
    )
    list_display = (
```

```
'username',
'email',
'role',
'is_staff',
'is_active'
```

```
)
Файл requests_app/models.py
створення модель Request, яка
використовується для збереження
первинних звернень користувачів.
from django.db import models
from users.models import User
class Request(models.Model):
    STATUS_CHOICES = [
        ('new', 'Нове'),
        ('in_progress', 'В
обробці'),
        ('scheduled',
        'Заплановано'),
        ('completed', 'Завершено'),
        ('cancelled', 'Скасовано'),
    ]
```

```
client = models.ForeignKey(
    User,
    on_delete=models.CASCADE,
    related_name='requests',
    verbose_name='Клієнт'
)
title = models.CharField(
    max_length=255,
    verbose_name='Тема
звернення'
)
```

```
description = models.TextField(
    verbose_name='Опис
звернення'
)
```

```
status = models.CharField(
    max_length=20,
    choices=STATUS_CHOICES,
    default='new',
    verbose_name='Статус'
)
```

```
created_at =
models.DateTimeField(
    auto_now_add=True,
    verbose_name='Дата
створення'
)
```

```
def __str__(self):
    return self.title
```

Реєстрація звернень в адміністративній панелі

```
from django.contrib import admin
from .models import Request
@admin.register(Request)
class RequestAdmin(admin.ModelAdmin):
```

```

list_display = (
    'title',
    'client',
    'status',
    'created_at'
)
list_filter = (
    'status',
    'created_at'
)
search_fields = (
    'title',
    'description'
)

```

Модель часових слотів консультанта

файл models.py - модель TimeSlot, яка описує доступний час консультанта для проведення консультації.

```

from django.db import models
from users.models import User
class TimeSlot(models.Model):
    consultant = models.ForeignKey(
        User,
        on_delete=models.CASCADE,
        related_name='time_slots',
        verbose_name='Консультант'
    )
    start_time = models.DateTimeField(
        verbose_name='Початок'
    )
    end_time = models.DateTimeField(
        verbose_name='Кінець'
    )
    is_booked = models.BooleanField(
        default=False,
        verbose_name='Заброньовано'
    )
    def __str__(self):
        return f"{self.consultant.username}: {self.start_time} - {self.end_time}"

```

Реєстрація часових слотів в адміністративній панелі

```

from django.contrib import admin
from .models import TimeSlot
@admin.register(TimeSlot)
class TimeSlotAdmin(admin.ModelAdmin):
    list_display = (
        'consultant',
        'start_time',
        'end_time',
        'is_booked'
    )
    list_filter = (
        'is_booked',
        'start_time'

```

```

)
search_fields = (
    'consultant__username',
)

```

Файл consultations_app/models.py - модель Consultation, яка є центральною сутністю системи. Вона пов'язує звернення, консультанта, часовий слот і посилання на онлайн-зустріч.

```

from django.db import models
from users.models import User
from requests_app.models import Request
from schedule_app.models import TimeSlot
class Consultation(models.Model):
    STATUS_CHOICES = [
        ('planned', 'Запланована'),
        ('active', 'Активна'),
        ('completed', 'Завершена'),
        ('cancelled', 'Скасована'),
    ]
    request = models.ForeignKey(
        Request,
        on_delete=models.CASCADE,
        related_name='consultations',
        verbose_name='Звернення'
    )
    consultant = models.ForeignKey(
        User,
        on_delete=models.CASCADE,
        related_name='consultations',
        verbose_name='Консультант'
    )
    time_slot = models.ForeignKey(
        TimeSlot,
        on_delete=models.SET_NULL,
        null=True,
        blank=True,
        verbose_name='Часовий слот'
    )
    meeting_link = models.URLField(
        blank=True,
        verbose_name='Посилання на консультацію'
    )
    status = models.CharField(
        max_length=20,
        choices=STATUS_CHOICES,
        default='planned',
        verbose_name='Статус'
    )
    created_at = models.DateTimeField(
        auto_now_add=True,
        verbose_name='Дата створення'
    )
    def __str__(self):
        return f"Консультація #{self.id}"

```

```

Реєстрація консультацій в
адміністративній панелі
from django.contrib import admin
from .models import Consultation
@admin.register(Consultation)
class
ConsultationAdmin(admin.ModelAdmin)
):

```

```

    list_display = (
        'id',
        'request',
        'consultant',
        'time_slot',
        'status',
        'created_at'
    )
    list_filter = (
        'status',
        'created_at'
    )
    search_fields = (
        'request__title',
        'consultant__username',
        'meeting_link'
    )

```

Файл messages_app/models.py - модель Message, яка використовується для збереження текстових повідомлень в консультації.

```

from django.db import models
from users.models import User
from consultations_app.models
import Consultation
class Message(models.Model):
    consultation =
models.ForeignKey(
    Consultation,
    on_delete=models.CASCADE,
    related_name='messages',

```

```

verbose_name='Консультація'
)
sender = models.ForeignKey(
    User,
    on_delete=models.CASCADE,
    related_name='messages',
    verbose_name='Відправник'
)
text = models.TextField(
    verbose_name='Текст
повідомлення'
)
sent_at = models.DateTimeField(
    auto_now_add=True,
    verbose_name='Час
надсилання'
)
def __str__(self):
    return f"Повідомлення від
{self.sender.username}"

```

Реєстрація повідомлень в адміністративній панелі

```

from django.contrib import admin

```

```

from .models import Message
@admin.register(Message)
class
MessageAdmin(admin.ModelAdmin):
    list_display = (
        'consultation',
        'sender',
        'sent_at'
    )
    list_filter = (
        'sent_at',
    )
    search_fields = (
        'text',
        'sender__username'
    )

```

Файл files_app/models.py - модель ConsultationFile, яка відповідає за збереження файлів, пов'язаних з консультацією.

```

from django.db import models
from consultations_app.models
import Consultation
class
ConsultationFile(models.Model):
    consultation =
models.ForeignKey(
    Consultation,
    on_delete=models.CASCADE,
    related_name='files',

```

```

verbose_name='Консультація'
)
title = models.CharField(
    max_length=255,
    verbose_name='Назва файлу'
)
file = models.FileField(
upload_to='consultation_files/',
    verbose_name='Файл'
)
uploaded_at =
models.DateTimeField(
    auto_now_add=True,
    verbose_name='Дата
завантаження'
)
def __str__(self):
    return self.title

```

Реєстрація файлів в адміністративній панелі

```

from django.contrib import admin
from .models import
ConsultationFile
@admin.register(ConsultationFile)
class
ConsultationFileAdmin(admin.ModelAdmin):
    list_display = (
        'title',
        'consultation',
        'uploaded_at'
    )

```

```

    )
    search_fields = (
        'title',
    )
)

Файл notifications_app/models.py -
модель Notification, яка
використовується для формування
системних повідомлень користувачам.
from django.db import models
from users.models import User
class Notification(models.Model):
    recipient = models.ForeignKey(
        User,
        on_delete=models.CASCADE,
related_name='notifications',
        verbose_name='Одержувач'
    )
    title = models.CharField(
        max_length=255,
        verbose_name='Заголовок'
    )
    text = models.TextField(
        verbose_name='Текст
сповіщення'
    )
    is_read = models.BooleanField(
        default=False,
        verbose_name='Прочитано'
    )
    created_at =
models.DateTimeField(
        auto_now_add=True,
        verbose_name='Дата
створення'
    )

    def __str__(self):
        return self.title

```

```

Реєстрація сповіщень в
адміністративній панелі
from django.contrib import admin
from .models import Notification
@admin.register(Notification)
class
NotificationAdmin(admin.ModelAdmin
):
    list_display = (
        'title',
        'recipient',
        'is_read',
        'created_at'
    )
    list_filter = (
        'is_read',
        'created_at'
    )
    search_fields = (
        'title',
        'text',
        'recipient__username'
    )

```

```

Файл audit_app/models.py - модель
AuditEvent, яка відповідає за
фіксацію важливих дій користувачів.
from django.db import models
from users.models import User
class AuditEvent(models.Model):
    user = models.ForeignKey(
        User,
        on_delete=models.SET_NULL,
        null=True,
        blank=True,
        verbose_name='Користувач'
    )
    action = models.CharField(
        max_length=255,
        verbose_name='Дія'
    )
    entity = models.CharField(
        max_length=255,
        verbose_name='Сутність'
    )
    created_at =
models.DateTimeField(
        auto_now_add=True,
        verbose_name='Дата події'
    )
    def __str__(self):
        return f"{self.action} -
{self.entity}"

```

```

Реєстрація журналу подій в
адміністративній панелі
from django.contrib import admin
from .models import AuditEvent
@admin.register(AuditEvent)
class
AuditEventAdmin(admin.ModelAdmin):
    list_display = (
        'action',
        'entity',
        'user',
        'created_at'
    )
    list_filter = (
        'created_at',
    )
    search_fields = (
        'action',
        'entity',
        'user__username'
    )

```

```

Головна сторінка системи
реалізована як HTML-шаблон Django.
Вона наслідує базовий шаблон
base.html,
містить інформаційний блок,
переходи до консультацій і
розкладу,
відображає основні показники роботи
системи.
{% extends 'base.html' %}
{% block title %}Головна{% endblock
%}

```

```

{% block content %}
<div class="home-clean">
  <div class="home-clean-shell">

    <section class="home-hero-clean">
      <div class="home-hero-grid">
        <div class="hero-left">
          <div class="hero-badge">
            Онлайн-консультації
          </div>
          <h1 class="hero-title">
            Зручний простір для звернень, запису та онлайн-консультацій
          </h1>
          <div class="hero-actions">
            <a href="/consultations/" class="hero-btn-secondary">
              Мої консультації
            </a>
            <a href="/schedule/" class="hero-btn-link">
              Перейти до розкладу →
            </a>
          </div>
        </div>
      </div>
    </section>

  </div>
</div>
{% endblock %}

Фрагмент блоку статистики головної сторінки
<section class="home-stats-row">
  <div class="home-stat-card">
    <div class="home-stat-top">
      <div class="home-stat-label">Звернення</div>
      <span class="home-stat-tag tag-peach">01</span>
    </div>
    <div class="home-stat-value">
      {{
request_count|default:"0" }}
    </div>
    <p class="home-stat-text">
      Створені користувачем запити в системі.
    </p>
  </div>

```

```

</div>
<div class="home-stat-card">
  <div class="home-stat-top">
    <div class="home-stat-label">Консультації</div>
    <span class="home-stat-tag tag-blue">02</span>
  </div>
  <div class="home-stat-value">
    {{
consultation_count|default:"0" }}
  </div>
  <p class="home-stat-text">
    Активні та заплановані консультації.
  </p>
</div>
<div class="home-stat-card">
  <div class="home-stat-top">
    <div class="home-stat-label">Сповіщення</div>
    <span class="home-stat-tag tag-lavender">03</span>
  </div>
  <div class="home-stat-value">
    {{
notification_count|default:"0" }}
  </div>
  <p class="home-stat-text">
    Нові системні повідомлення.
  </p>
</div>
<div class="home-stat-card">
  <div class="home-stat-top">
    <div class="home-stat-label">Вільні слоти</div>
    <span class="home-stat-tag tag-mint">04</span>
  </div>
  <div class="home-stat-value">
    {{
free_slots_count|default:"0" }}
  </div>
  <p class="home-stat-text">
    Доступний час для запису на консультацію.
  </p>
</div>
</section>

Фрагмент блоку основних дій користувача
<section class="home-main-actions">
  <div class="actions-head">
    <h2 class="home-section-title">Основні дії</h2>
  </div>

  <div class="action-list">
    <div class="action-item">
      <div>

```

```

        <div class="action-
title">
        Знайти
консультанта
    </div>
    <p class="action-
text">
        Переглянути
доступних фахівців і обрати того,
        хто підходить
для вашого запиту.
    </p>
    </div>
    <a
href="/users/consultants/"
class="action-btn">
        Переглянути
консультантів
    </a>
</div>

    <div class="action-item">
        <div>
            <div class="action-
title">
                Записатися на
                вільний час
            </div>
            <p class="action-
text">
                Оберіть часовий
                слот у розкладі для проведення
                консультації.
            </p>
        </div>
    </div>

```

```

    </div>
    <a href="/schedule/"
class="action-btn">
        Відкрити розклад
    </a>
</div>

    <div class="action-item">
        <div>
            <div class="action-
title">
                Перейти до
                консультацій
            </div>
            <p class="action-
text">
                Переглянути
                свої записи, повідомлення та
                пов'язані файли.
            </p>
        </div>
        <a
href="/consultations/"
class="action-btn">
            Відкрити
            консультації
        </a>
    </div>
</div>
</section>

```

Додаток Г
Апробація отриманих результатів

ДОВІДКА

ПРО ПРИЙНЯТТЯ НАУКОВОЇ РОБОТИ ДО ПУБЛІКАЦІЇ

10.06.2026

Шановний(і) авторе(и):

Прончук Дарина Сергіївна ,

Організаційний комітет з радістю повідомляє, що наукову роботу «ІНФОРМАЦІЙНА СИСТЕМА НАДАННЯ ОНЛАЙН-КОНСУЛЬТАЦІЙ ДЛЯ ОСІБ, ЯКІ ПОСТРАЖДАЛИ ВІД ВІЙНИ» прийнято до публікації в збірнику за матеріалами XI Міжнародної наукової конференції «Традиційні та інноваційні підходи до наукових досліджень» (19.06.2026; м. Луцьк, Україна).

Опублікована робота буде доступна з 19.06.2026 за посиланням:

<https://archives.mcnd.org.ua/index.php/conference-proceeding/issue/view/19.06.2026>

.....

Електронні сертифікати учасників конференції будуть доступні з 19 червня.

Конференцію зареєстровано в базі даних науково-технічних заходів УкрІНТЕІ (Посвідчення № 176 від 26.01.2026). Матеріали конференції знаходитимуться в відкритому доступі (Open Access) на умовах ліцензії CC BY-SA 4.0 International.

З повагою,

Віце-президент МЦНД
Голова оргкомітету конференції
НАСТАСІЯ РАБЕЙ



ІНФОРМАЦІЙНА СИСТЕМА НАДАННЯ ОНЛАЙН-КОНСУЛЬТАЦІЙ ДЛЯ ОСІБ, ЯКІ ПОСТРАЖДАЛИ ВІД ВІЙНИ

Прончук Дарина Сергіївна

Бакалавр спеціальності 122 – Комп'ютерні науки

Західноукраїнський національний університет, Україна

Науковий керівник: Осолінський Олександр Романович

ORCID ID: 0000-0002-0136-395X

к.т.н., доцент кафедри інформаційно-обчислювальних систем і управління

Західноукраїнський національний університет

Україна

Вступ

В умовах війни значна кількість людей потребує швидкої психологічної, соціальної або інформаційної підтримки, але очне звернення до фахівця не завжди є можливим. Причинами цього є вимушене переміщення, небезпека в окремих регіонах, обмежений транспортний доступ, нестабільний зв'язок та небажання людини повторно розкривати чутливі особисті дані. Тому онлайн-консультування є доцільною формою організації допомоги [1].

Разом з тим звичайного відеодзвінка для такої роботи недостатньо. Сервіс має забезпечувати не лише саму консультацію, але і повний цикл взаємодії: створення звернення, первинну маршрутизацію, обмін повідомленнями і файлами, формування сповіщень та збереження історії подій [2].

Метою роботи є розроблення інформаційної системи надання онлайн-консультацій для осіб, які постраждали від війни. Така система повинна об'єднати простий користувацький інтерфейс, серверну логіку обробки запитів, структуроване зберігання даних та засоби дистанційної взаємодії.

Архітектура інформаційної системи

Запропонована інформаційна система побудована, як модульний вебзастосунок, в якому основними учасниками є клієнт, консультант і адміністратор. Клієнт створює звернення, обирає напрям допомоги, переглядає доступний час і бере участь у консультації. Консультант налаштовує профіль та графік прийому, опрацьовує звернення, проводить сесію та працює з повідомленнями та матеріалами. Адміністратор контролює користувачів, модерує процеси, стежить за журналом подій і забезпечує організаційну цілісність системи.

Загальна структура системи (рисунок 1) включає клієнтську частину, серверну логіку, базу даних, файлове сховище та модулі інтеграції із зовнішніми сервісами. Клієнтська частина відповідає за доступ користувачів через веб-інтерфейс, а серверна частина реалізує обробку запитів, перевірку прав доступу, зміну статусів і передавання даних між внутрішніми модулями. База даних зберігає користувачів, ролі, звернення, записи, консультації та історію взаємодій, а файлове сховище використовується для документів і вкладень.



Рис. 1. Структура інформаційної системи онлайн-консультавання

Функціонально система поділяється на модулі керування користувачами і ролями, профілями консультантів, зверненнями, календарем і бронюванням,

консультаціями, повідомленнями, файлами, сповіщеннями та журналом подій. Такий поділ спрощує розроблення і подальший супровід, тому що кожен модуль має окрему зону відповідальності.

Алгоритм роботи системи починається з створення звернення. Користувач вводить тему, категорію проблеми, бажаний формат консультації, опис ситуації та контактні дані. Після перевірки коректності даних система виконує первинну маршрутизацію, визначає напрям допомоги, підбирає консультанта за спеціалізацією і доступним часом та формує запис на консультацію. Якщо відповідний час або консультант недоступні, користувачу пропонується альтернативний варіант.

Програмна реалізація та тестування

Серверну частину системи реалізовано на базі Django [3]. Для зберігання структурованих даних використано PostgreSQL [4]. Інтерфейс побудовано з використанням HTML, CSS і Bootstrap, що забезпечує адаптивність сторінок для роботи зі смартфонів, планшетів і персональних комп'ютерів [5].

У Django-проекті створено окремі застосунки `users`, `requests_app`, `consultations_app`, `schedule_app`, `messages_app`, `files_app`, `notifications_app` та `audit_app`. Вони відповідають за роботу з користувачами, первинними зверненнями, консультаціями, часовими слотами, внутрішніми повідомленнями, файлами, сповіщеннями та журналюванням дій. Для онлайн-зустрічей передбачено інтеграцію з сервісом відеозв'язку через посилання на сесію, а службові повідомлення використовуються для інформування користувачів про зміну статусу звернення або наближення консультації.

Під час тестування перевірено основні функціональні сценарії: вхід користувача в систему, створення нового звернення, відображення збереженого запиту, запис на вільний часовий слот, блокування некоректного бронювання, відкриття сторінки детальної консультації, надсилання повідомлень і завантаження файлів.

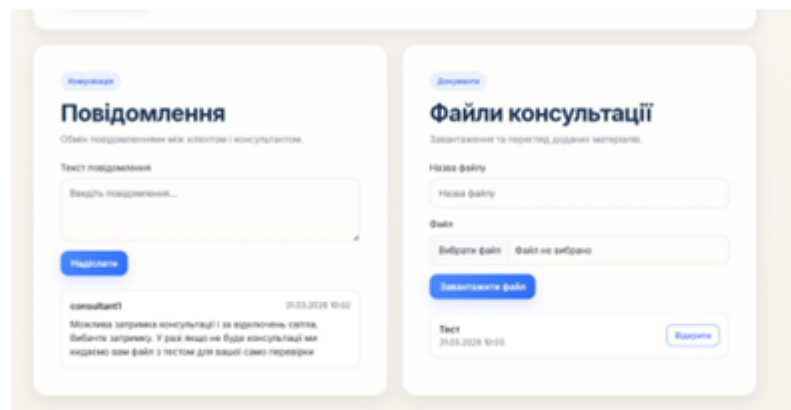


Рис. 2. Перевірка обміну повідомленнями та файлами під час консультації

Висновки. Було запропоновано інформаційну систему для організації онлайн-консультацій осіб, які постраждали від війни. Він охоплює повний цикл від первинного звернення і маршрутизації до запису, проведення консультації, обміну матеріалами, формування сповіщень та фіксації дій користувачів.

Список використаних джерел:

1. Inter-Agency Standing Committee. IASC Guidelines on Mental Health and Psychosocial Support in Emergency Settings. 2007. URL: <https://interagencystandingcommittee.org/>
2. World Health Organization, War Trauma Foundation, World Vision International. Psychological first aid: guide for field workers. Geneva : World Health Organization, 2011. URL: <https://www.who.int/publications/i/item/9789241548205>
3. Django Software Foundation. Django documentation. URL: <https://docs.djangoproject.com/>
4. PostgreSQL Global Development Group. PostgreSQL Documentation. URL: <https://www.postgresql.org/docs/>
5. Bootstrap. Grid system. URL: <https://getbootstrap.com/docs/5.0/layout/grid/>