

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВАДИМА ГЕТЬМАНА**

**Навчально-науковий інститут
«Інститут інформаційних технологій в економіці»**

Кафедра інформаційних систем в економіці

**ОСВІТНЬО-ПРОФЕСІЙНА ПРОГРАМА
«Інформаційні управляючі системи і технології»**

галузь знань	12 Інформаційні технології
спеціальність	122 Комп'ютерні науки

Форма навчання: очна (денна)

КВАЛІФІКАЦІЙНА МАГІСТЕРСЬКА РОБОТА

на тему: «Розроблення інформаційної системи для реабілітаційного центру»

здобувача Фурманюк Ірини Сергіївни

(ПІБ, підпис)

Науковий керівник: д.е.н., професор Ріппа С.П.

(науковий ступінь, учене звання, ПІБ)

(підпис)

**Робота допущена до захисту перед екзаменаційною
комісією з атестації здобувачів вищої освіти (ЕК)**

Завідувач кафедри: к.е.н., доцент. Тішков Б.О.

(науковий ступінь, учене звання, ПІБ)

(підпис)

Київ 2025

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВАДИМА ГЕТЬМАНА
Навчально-науковий інститут «Інститут інформаційних технологій в економіці»
Кафедра інформаційних систем в економіці

ОСВІТНЬО-ПРОФЕСІЙНА ПРОГРАМА
«Інформаційні управляючі системи і технології»
галузь знань 12 «Інформаційні технології»
спеціальність 122 «Комп'ютерні науки»

ПОГОДЖЕНО:

Керівник проєктної групи (гарант)
освітньо-професійної програми

_____ Устенко С.В.
« ____ » _____ 2025 р.

ЗАТВЕРДЖУЮ:

Завідувач кафедри інформаційних
систем в економіці

_____ Тішков Б.О.
« ____ » _____ 2025 р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

здобувача вищої освіти **Фурманюк Ірини Сергіївни**
очної (денної) форми навчання

на підготовку кваліфікаційної магістерської роботи
на тему: «Розроблення інформаційної системи для реабілітаційного центру»

Тему затверджено наказом ректора Університету від «20» лютого 2025 р. № **332-ст**
Кваліфікаційна магістерська робота виконується на матеріалах
умовного реабілітаційного підприємства та Інтернет-джерелах

План кваліфікаційної магістерської роботи

Розділ I Дослідження та аналіз підходів до створення інформаційних управляючих систем реабілітаційних центрів

Розділ II Характеристика інформаційних управляючих систем та методи і моделі

Розділ III Розроблення проєктних рішень та їх реалізація

Об'єкт дослідження: автоматизація процесу планування засідань МДРК для реабілітаційного центру.

Предмет дослідження: розробка інформаційної системи планування різних подій та засідань мультидисциплінарних реабілітаційних команд з функцією автоматичного пошуку вільного часу.

Мета кваліфікаційної магістерської роботи: розробка інформаційної системи для реабілітаційного центру, яка б забезпечувала зручний доступ пацієнтів до інформації про центр та його послуги, а також надавала інструменти для ефективного планування та координації роботи мультидисциплінарних реабілітаційних команд, що є ключовим елементом для розробки індивідуальних реабілітаційних планів.

Конкретні завдання, які здобувач повинен виконати для досягнення поставленої мети:

У розділі I Дослідити роботу реабілітаційного центру та процес планування засідань МДРК, проаналізувати вже існуючі інформаційні системи та скласти їх порівняльну таблицю

У розділі II Надати характеристику інформаційної системи для реабілітаційного центру. Визначити функціональні і нефункціональні вимоги. Змодельовати поведінку системи

У розділі III Розробити проектні рішення та реалізацію. обрати БД та спроектувати її структуру, створити інфологічну модель БД та на її основі розробити даталогічну. Обрати, обґрунтувати та описати архітектуру системи, визначити засоби та технології для її розробки. Реалізувати інформаційну систему та провести її тестування.

**Завдання підготував
науковий керівник**

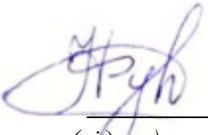
(підпис)

Ріппа Сергій Петрович

(ініціали, прізвище)

« ____ » _____ 202_ р.

**Завдання одержав
здобувач**



(підпис)

Фурманюк Ірина Сергіївна

(ініціали, прізвище)

« ____ » _____ 202_ р.

Реферат

Кваліфікаційна магістерська робота містить 96 сторінок, 1 таблицю, 32 рисунки, список використаних джерел з 52 найменувань, додатки.

«Розроблення інформаційної системи для реабілітаційного центру»

Об'єктом дослідження кваліфікаційної магістерської роботи є автоматизація процесу планування засідань МДРК для реабілітаційного центру.

Предметом дослідження є розробка інформаційної системи планування різних подій та засідань мультидисциплінарних реабілітаційних команд з функцією автоматичного пошуку вільного часу

Мета і завдання дослідження. Основною метою кваліфікаційної магістерської роботи є створення інформаційної системи для реабілітаційного центру, яка б забезпечувала зручний доступ пацієнтів до інформації про центр та його послуги, а також надавала інструменти для ефективного планування та координації роботи мультидисциплінарних реабілітаційних команд, що є ключовим елементом для розробки індивідуальних реабілітаційних планів

Відповідно до поставленої мети визначені такі *завдання*:

- Проаналізувати існуючі інформаційні системи та веб-ресурси реабілітаційних центрів.
- Визначити основні інформаційні потреби пацієнтів та працівників реабілітаційного центру у вигляді функціональних та нефункціональних вимог.
- Спроектувати функціональну структуру вебсайту, включаючи розділи для пацієнтів (інформація про центр, послуги) та працівників (календар для планування засідань).
- Розробити базу даних для зберігання інформації про центр, працівників та заплановані засідання.
- Реалізувати вебсайт згідно з розробленим проектом, забезпечивши зручний та інтуїтивно зрозумілий інтерфейс.
- Провести тестування розробленої інформаційної системи та оцінити її ефективність.
- Надати рекомендації щодо впровадження та подальшого розвитку розробленої інформаційної системи.

Теоретична, методична та практична значущість отриманих результатів. У рамках дослідження було проаналізовано сучасні підходи до

автоматизації процесів планування мультидисциплінарних засідань у реабілітаційних центрах. Практичне значення одержаних результатів полягає в готовому програмному продукті, що дозволяє значно підвищити точність та оперативність організації засідань, зменшити ризики помилок і оптимізувати робочий графік працівників реабілітаційного центру.

Рік виконання кваліфікаційної магістерської роботи – 2025.

Рік захисту роботи – 2025.

Ключові слова: реабілітаційний центр, МДРК, календар подій, інформаційна система.

Відгук
на кваліфікаційну магістерську роботу
Фурманюк Ірини Сергіївни на тему
«РОЗРОБЛЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ
РЕАБІЛІТАЦІЙНОГО ЦЕНТРУ»
за спеціальністю «Комп'ютерні науки»

Актуальність теми. Тема створення інформаційної управляючої системи для реабілітаційного центру є надзвичайно актуальною в умовах зростаючого навантаження на медичні установи та впровадження практик дистанційного планування. Розробка такого програмного продукту дозволяє упорядкувати процес координації мультидисциплінарних засідань, звести до мінімуму ризику подвійного бронювання часу та підвищити прозорість обслуговування пацієнтів. З огляду на сучасні тенденції цифровізації сфери охорони здоров'я, створена система відповідає потребам реабілітаційних центрів.

Позитивні риси кваліфікаційного бакалаврського проєкту. У роботі відзначаються чітка структура та послідовність викладу, інтеграція сучасних інструментів (Spring Boot, Drools, Thymeleaf), а також продумана архітектура з перспективою подальшого масштабування. Особливо варто відзначити реалізацію інтелектуального пошуку вільних часових слотів.

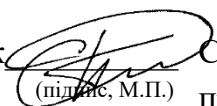
Наявність самостійних розробок автора. Підготовлено працездатний ІУС для підтримки діяльності реабілітаційних центрів у вигляді тестового застосунку з монолітною архітектурою з використанням платформ Java і Spring, UI частина ІС створена за допомогою Thymeleaf, HTML, CSS та JavaScript, що можна вважати авторським надбанням здобувача.

Цінність теоретичних висновків та практичних рекомендацій. Практична значимість роботи полягає у наданні готового рішення для автоматизації планування засідань мультидисциплінарних команд, яке легко інтегрувати в існуючу ІТ-інфраструктуру реабілітаційних центрів. Рекомендації щодо подальшого розвитку функціоналу та впровадження хмарних технологій дозволяють забезпечити гнучку масштабованість і ефективне обслуговування зростаючої кількості користувачів.

Наявність недоліків. Зауважень немає.

Загальна оцінка кваліфікаційної магістерської роботи. У цілому здобувач вищої освіти Фурманюк Ірина Сергіївна виконала дипломний проєкт на тему «РОЗРОБЛЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ РЕАБІЛІТАЦІЙНОГО ЦЕНТРУ» відповідно вимогам вищої школи України, що пред'являються до магістерського ступеню за спеціальністю «Комп'ютерні науки» і може бути рекомендована до захисту з високою оцінкою.

Науковий керівник


(підпис, М.П.)

С. П. Ріппа, д.е.н., професор,
професор кафедри інформаційних систем в економіці
Навчально-наукового інституту "Інститут
інформаційних технологій в економіці" КНЕУ

«17» травня 2025 р.

РЕЦЕНЗІЯ

на кваліфікаційну магістерську роботу
здобувачки ступеня магістра
Фурманюк Ірини Сергіївни
на тему «РОЗРОБЛЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ
РЕАБІЛІТАЦІЙНОГО ЦЕНТРУ»
за спеціальністю 122 Комп'ютерні науки

Актуальність теми: Представлена на рецензування кваліфікаційна магістерська робота є достатньою мірою актуальним дослідженням, оскільки пов'язана з розвитком, вдосконаленням, цифровізацією соціальної та гуманітарної сфер в Україні, в умовах воєнного стану.

Характеристика роботи. Рецензувана робота складається із 3-х розділів, містить два додатки. Перший розділ присвячений дослідженню і аналізу предметної області інформаційних управляючих (ІУС) систем реабілітаційних центрів, де здобувачка дає обґрунтування для побудови реабілітаційної ІУС. У другому розділі розглядаються аналіз і специфікація ІУС для реабілітаційних центрів, декларуються основні проектні рішення – інформаційна модель, діаграми прецедентів і послідовностей. Третій розділ охоплює конструктивні аспекти реалізованого проєкту – інформаційне, організаційне, програмне та технічне забезпечення для ІУС в реабілітаційній сфері.

Наявність самостійних розробок автора і пропозиції для впровадження. До авторських розробок, представлених у роботі можна віднести пропозиції здобувача щодо концепції інформаційної моделі реабілітаційних центрів як комплексного проектного рішення, що інтегрує сучасні платформи та інструменти (це Java, Spring, JavaScript, Drools, СКБД MySQL, MapStruct і IntelliJ IDEA) для отримання працездатних прототипів.

До окремих зауважень і недоліків, які не впливають на позитивне враження від рецензованої роботи, можна віднести, зокрема, незначні відхилення в оформленні списку використаних джерел та додатків.

Загальний висновок. У цілому, кваліфікаційна магістерська робота відповідає вимогам МОН щодо обраної тематики, виконана на належному рівні, за змістом та практичною цінністю може заслуговувати на високу оцінку, а її авторка, Фурманюк Ірина Сергіївна заслуговує на присвоєння їй освітньо-кваліфікаційного рівня «магістр» за спеціальністю 122 Комп'ютерні науки.

Рецензент:



Погореловська І.Д., к.е.н., доцент,
доцент кафедри комп'ютерних та інформаційних
технологій і систем факультету фінансів та
цифрових технологій Державного
податкового університету

«18» травня 2025 р.

ЗМІСТ

ЗМІСТ	8
ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ	10
ВСТУП.....	11
РОЗДІЛ 1. ДОСЛІДЖЕННЯ ТА АНАЛІЗ ПІДХОДІВ ДО СТВОРЕННЯ ІНФОРМАЦІЙНИХ УПРАВЛЯЮЧИХ СИСТЕМ РЕАБІЛІТАЦІЙНИХ ЦЕНТРІВ	15
1.1 Аналіз існуючих інформаційних управляючих систем реабілітаційних центрів	15
1.1.1 Дослідження предметної області.....	15
1.1.2 Дослідження інформаційних управляючих систем	16
1.2 Обґрунтування вибору підходів і технологій для створення інформаційної управляючої системи	18
РОЗДІЛ 2. ХАРАКТЕРИСТИКА ІНФОРМАЦІЙНИХ УПРАВЛЯЮЧИХ СИСТЕМ ТА МЕТОДИ І МОДЕЛІ.....	22
2.1 Структура і характеристика системи	22
2.2 Методи та моделі в інформаційних управляючих системах і технологіях.....	30
РОЗДІЛ 3. РОЗРОБЛЕННЯ ПРОЄКТНИХ РІШЕНЬ ТА ЇХ РЕАЛІЗАЦІЯ.....	32
3.1. Проєктування бази даних для інформаційної управляючої системи	32
3.1.1 Проєктування інфологічної (логічної) моделі БД.....	32
3.1.2 Обґрунтування вибору СКБД та проєктування фізичної моделі БД в середовищі вибраної СКБД.....	33
3.1.3 Контрольний приклад та обрахунок прогнозних обсягів БД	35
3.2. Проєктування бази знань інформаційної управляючої системи	37
3.2.1 Реалізація інтелектуального аналізу великих даних з метою наповнення бази знань	37
3.2.2 Проєктування та реалізація бази знань інформаційної управляючої системи	38
3.3. Програмне забезпечення.....	39
3.3.1 Проєктування програмного забезпечення	39
3.3.2 Вибір архітектури програмного забезпечення	41
3.3.3 Інструментальні засоби розробки для створення прикладного програмного забезпечення інформаційних управляючих систем.....	42
3.3.4 Тестування програмного забезпечення.....	44

3.3.5 Керівництво (інструкція) користувача.....	44
3.4. Технічне забезпечення.....	45
3.4.1 Обґрунтування вибору технічного забезпечення	45
3.4.2 Ресурсні вимоги до технічного забезпечення	46
3.5. Реалізація інформаційної управляючої системи.....	47
ДОДАТКИ.....	65
Додаток А. Заповнена контрольним прикладом база даних.....	65
Додаток Б. Приклад реалізованого тексту програми	68

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

БД	База даних
ЕСОЗ	Електронна система охорони здоров'я
ІРП	Індивідуальний реабілітаційний план
ІС	Інформаційна система
СКБД	Система керування базами даних
МДРК	Мультидисциплінарна реабілітаційна команда
МІС	Медична інформаційна система
НСЗУ	Національна служба здоров'я України
НТТР	Hypertext Transfer Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
SSD	Solid State Drive

ВСТУП

Актуальність теми. У сучасному світі значно зросла кількість людей, які потребують якісної реабілітації внаслідок різних травм та захворювань. Нещасні випадки на виробництві, дорожньо-транспортні пригоди, наслідки захворювань, а ще в умовах складної ситуації в Україні – травми, пов'язані з воєнними діями, призводять до збільшення потреби фізичного і психологічного лікування та в послугах реабілітаційних центрів. Ефективність роботи таких центрів напряду залежить від організації внутрішніх процесів та комунікації між пацієнтами та працівниками центру. Впровадження нових ідей та сучасних інформаційних технологій, а саме розробка спеціалізованих вебсайтів, є важливим етапом у підвищенні якості та доступності реабілітаційних послуг, адже такі системи здатні оптимізувати інформаційний обмін, вдосконалити координацію роботи мультидисциплінарних команд. Отже, розроблення інформаційної системи для реабілітаційного центру є актуальною та своєчасною темою, що відповідає сучасним вимогам у сфері охорони здоров'я та соціальної підтримки.

Аналіз останніх досліджень і публікацій у сфері інформаційних технологій в охороні здоров'я свідчить про зростаючий інтерес до роботи медичних закладів, включаючи реабілітаційні центри. Так В.О.Качмар у своїй статті «Медичні інформаційні системи – стан розвитку в Україні» вказує, що використання інформаційних технологій в сфері охорони здоров'я розвивається швидкими темпами та акцентує увагу на важливості кооперації та обміну інформацією між різними установами охорони здоров'я. На її думку, взаємосумісність медичних інформаційних систем є однією з головних умов успішної й безпечної передачі інформації про пацієнта між ними. [32]

Бісмак О.В. у своїй науковій праці «Особливості організації діяльності реабілітаційних закладів в Україні» проаналізувала проблеми організації та роботи реабілітаційних центрів в Україні та зазначила, що повноцінний розвиток сучасної системи саме для реабілітаційних центрів в Україні трохи гальмується

недооцінкою ролі реабілітаційних заходів [35]. О. А. Панченко також вважає, що «інформатизація реабілітаційної медицини в Україні в даний час далека від досконалості і потребує інтенсифікації науково-практичних досліджень в цьому напрямі». У своїй праці «Інформаційні технології в реабілітології» він розглянув результати аналізу стану та тенденцій розвитку інформатизаційних процесів у реабілітології та зробив висновки про необхідність інтенсифікації науково-практичних досліджень у цьому напрямі. [36]

Проблеми та питання пов'язані з діяльністю реабілітаційних центрів та їх автоматизацією досліджували Н.В. Заварова [28], О.В. Погонцева [37], Є.Н. Приступа [42], І.Г. Саранча [45]. Організація діяльності закладів соціальної реабілітації та інші. Проте незважаючи на значний інтерес до роботи медичних закладів [41;43;44] та реабілітаційних центрів в тому числі, все ж недостатньо досліджень зосереджено саме на специфічних потребах реабілітаційних центрів, особливо в контексті організації засідань мультидисциплінарних команд. У зв'язку з цим наукова проблема, яка розглядається в даному дослідженні, полягає у необхідності підвищення ефективності функціонування реабілітаційного центру шляхом оптимізації інформаційних процесів, в даному випадку організації роботи мультидисциплінарної реабілітаційної команди.

Об'єктом дослідження кваліфікаційної магістерської роботи є автоматизація процесу планування засідань МДРК для реабілітаційного центру.

Предметом дослідження є розробка інформаційної системи планування різних подій та засідань мультидисциплінарних реабілітаційних команд з функцією автоматичного пошуку вільного часу

Мета і завдання дослідження. Основною метою кваліфікаційної магістерської роботи є створення інформаційної системи для реабілітаційного центру, яка б забезпечувала зручний доступ пацієнтів до інформації про центр та його послуги, а також надавала інструменти для ефективного планування та координації роботи мультидисциплінарних реабілітаційних команд, що є ключовим елементом для розробки індивідуальних реабілітаційних планів

Задачі дослідження:

1. Проаналізувати існуючі інформаційні системи та веб-ресурси реабілітаційних центрів.
2. Визначити основні інформаційні потреби пацієнтів та працівників реабілітаційного центру у вигляді функціональних та нефункціональних вимог.
3. Спроекувати функціональну структуру вебсайту, включаючи розділи для пацієнтів (інформація про центр, послуги) та працівників (календар для планування засідань).
4. Розробити базу даних для зберігання інформації про центр, працівників та заплановані засідання.
5. Реалізувати вебсайт згідно з розробленим проектом, забезпечивши зручний та інтуїтивно зрозумілий інтерфейс.
6. Провести тестування розробленої інформаційної системи та оцінити її ефективність.
7. Надати рекомендації щодо впровадження та подальшого розвитку розробленої інформаційної системи.

Для досягнення поставленої мети та вирішення визначених завдань у роботі будуть використані наступні методи дослідження: аналіз наукової літератури та інтернет-ресурсів для дослідження існуючих підходів до розробки інформаційних систем в охороні здоров'я та аналізу ІС реабілітаційних центрів, системний аналіз для визначення функціональних та нефункціональних вимог до інформаційної системи, проектування для розробки структури бази даних та інтерфейсу користувача вебсайту, моделювання для створення прототипів окремих компонентів системи, статистичні методи для аналізу результатів тестування та оцінки ефективності впровадження системи.

Результати дослідження будуть використані адміністрацією реабілітаційних центрів для впровадження розробленої інформаційної системи з

метою покращення внутрішніх процесів та їх автоматизації. Працівники центру зможуть використовувати систему для кращого планування та проведення засідань мультидисциплінарних команд. Пацієнти отримають зручний доступ до інформації про реабілітаційний центр та його послуги.

Особистий внесок автора проєкту полягає у розробці концепції інформаційної системи, проектуванні її функціональних можливостей, розробці бази даних та інтерфейсу користувача вебсайту, а також у проведенні аналізу ефективності запропонованого рішення.

Кваліфікаційний магістерський проєкт складається з вступу, трьох розділів та висновків, списку використаних джерел та додатків.

РОЗДІЛ 1. ДОСЛІДЖЕННЯ ТА АНАЛІЗ ПІДХОДІВ ДО СТВОРЕННЯ ІНФОРМАЦІЙНИХ УПРАВЛЯЮЧИХ СИСТЕМ РЕАБІЛІТАЦІЙНИХ ЦЕНТРІВ

1.1 Аналіз існуючих інформаційних управляючих систем реабілітаційних центрів

1.1.1 Дослідження предметної області

Реабілітаційний центр є спеціалізованим закладом охорони здоров'я, діяльністю якого є спрямування на відновлення, підтримку або покращення фізичного, психічного, соціального стану осіб із обмеженнями життєдіяльності внаслідок захворювань, отриманих травм, вроджених вад чи інвалідності. Відповідно до законодавства України, зокрема Закону України «Про реабілітацію у сфері охорони здоров'я» [29], основною метою функціонування таких закладів є забезпечення пацієнтів комплексною, командною допомогою для досягнення максимально можливої автономії та якості життя. [25]

Сучасні реабілітаційні центри знаходяться у складному середовищі, яке вимагає чіткої організації бізнес-процесів, ефективного управління ресурсами та своєчасного прийняття управлінських рішень. До ключових процесів належать: реєстрація та облік пацієнтів, планування та проведення засідань МДРК, створення індивідуальних реабілітаційних планів (ІРП), контроль за їх виконанням, взаємодія з іншими закладами та системами охорони здоров'я, моніторинг ефективності реабілітації. [24;51]

Особливу увагу слід приділити діяльності мультидисциплінарної реабілітаційної команди. МДРК – це група фахівців різного профілю (лікарі, реабілітологи, психологи, логопеди, соціальні працівники тощо), які спільно обговорюють та приймають рішення щодо плану реабілітації конкретного пацієнта. [50] Засідання МДРК є ключовим етапом прийняття управлінських рішень у реабілітаційному процесі, адже саме на цьому етапі формується ІРП – документ, що визначає персоналізований план реабілітації. План включає чіткі цілі, методи, строки і відповідальних виконавців, що дозволяє системно

впроваджувати відновлювальні заходи, узгоджені між усіма учасниками процесу. [30]

Процес створення ІРП наразі автоматизований у багатьох закладах за допомогою сучасних медичних інформаційних систем (МІС). Такі системи відповідають технічним і функціональним вимогам, встановленим Національною службою здоров'я України (НСЗУ) та Електронною системою охорони здоров'я (ЕСОЗ), з якою забезпечується інтеграція. Це дозволяє зменшити паперове навантаження, підвищити точність документування та забезпечити прозорість і контроль за наданням медичних послуг. Водночас, незважаючи на наявність таких МІС, питання планування та організації самих засідань МДРК, як елементів управління, залишається недостатньо автоматизованим, що створює труднощі у координації учасників команди, контролі за їх участю та дотриманні регламентів. [29;31;38;39]

Серед рішень, що приймаються у реабілітаційних центрах, можна виділити такі: призначення дати та часу засідань МДРК, вибір оптимального реабілітаційного плану пацієнта та контроль за його виконанням. Проблемні ситуації виникають при перевантаженні фахівців, дублюванні функцій, нестачі часу для координації або при збої в інформаційному обміні. Рішення можуть прийматись як на рівні адміністраторів, керівництва установи, так і безпосередньо членами МДРК. Їх періодичність коливається від щоденного (наприклад, оновлення ІРП) до щотижневого (планування нових засідань), а тривалість – від кількох хвилин до кількох годин.

1.1.2 Дослідження інформаційних управляючих систем

Для реабілітаційних центрів інформаційні системи відіграють важливу роль у забезпеченні ефективної координації роботи мультидисциплінарних реабілітаційних команд (МДРК), плануванні засідань та управлінні ресурсами.[32] Для прикладу, розглянемо чотири такі системи, а саме: Smart Rehab Manager, Priamo/Q-Rehab, ReDoc та TigerConnect.

Smart Rehab Manager система пропонує комплексний підхід до управління реабілітаційними процесами. Вона забезпечує планування засідань МДРК, координацію графіків роботи фахівців, управління ресурсами клініки та надає пацієнтам доступ до розкладу процедур через спеціальний додаток. [16]

Priamo/Q-Rehab розроблені в Італії, ці дві взаємопов'язані системи підтримують весь цикл реабілітації. Priamo дозволяє створювати індивідуальні реабілітаційні плани на основі біопсихосоціальної моделі, а Q-Rehab використовує штучний інтелект для автоматичного планування терапевтичних сесій, враховуючи обмеження та переваги пацієнтів і персоналу. [17]

ReDoc система відома своєю інтуїтивною системою планування, яка дозволяє ефективно керувати розкладами терапевтів та пацієнтів. ReDoc підтримує автоматичне планування, управління авторизаціями та надає звіти про продуктивність персоналу. [21]

TigerConnect фокусується на покращенні комунікації між членами команди та пацієнтами. Система забезпечує безпечний обмін повідомленнями, відеоконсультації та автоматичні нагадування про зустрічі, сприяючи ефективній координації та своєчасному втручанню в процес лікування. [1;23]

За результатами огляду існуючих аналогів інформаційної системи було створено таблицю з порівняльними характеристиками(таблиця 1.1).

Порівняння існуючих аналогів інформаційної системи

Система	Планування засідань МДРК	Автоматичне планування	Комунікація з пацієнтами	Штучний інтелект	Мобільний доступ	Зручність інтерфейсу	Підбір вільних слотів	Групове планування	Враховання відпусток
Smart Rehab Manager	Так	Так	Так	Ні	Так	Висока	Так	Так	Так
Priamo/Q-Rehab	Так	Так	Обмежено	Так	Ні	Середня	Так	Так	Обмежено
ReDoc	Так	Так	Обмежено	Ні	Так	Висока	Так	Так	Так
TigerConnect	Ні	Ні	Так	Ні	Так	Висока	Ні	Ні	Ні

Аналіз існуючих інформаційних управляючих систем для реабілітаційних центрів виявив їхню значущість у оптимізації різноманітних аспектів діяльності. Однак, питання підтримки специфічного процесу організації засідань мультидисциплінарних реабілітаційних команд, який є ключовим для розробки індивідуальних планів лікування, часто не є пріоритетним або достатньо відображеним у функціоналі розглянутих систем. Це підкреслює актуальність розробки інформаційної системи, яка б враховувала дану специфічну потребу реабілітаційних центрів.

1.2 Обґрунтування вибору підходів і технологій для створення інформаційної управляючої системи

У межах магістерської роботи було розроблено концептуальні рішення щодо створення інформаційної системи підтримки процесу планування та проведення засідань мультидисциплінарної реабілітаційної команди (МДРК) у

реабілітаційному центрі. Основна ідея проєкту полягає в автоматизації найбільш трудомісткого та критично важливого етапу в реабілітаційному процесі — координації командної роботи фахівців, узгодження графіків для формування індивідуальних реабілітаційних планів пацієнтів. Запропонована концептуальна модель проєктного рішення базується на сучасних технологічних підходах, що відповідають як реальним потребам медичних закладів, так і вимогам державної інформаційної інфраструктури в галузі охорони здоров'я.

Одним із основоположних рішень стало визначення технологічного стеку, який забезпечує надійність, масштабованість і можливість подальшої інтеграції з іншими медичними системами. ІС реалізована за допомогою Java із використанням фреймворку Spring, що дозволило побудувати гнучку модульну архітектуру з чітким поділом логіки, інтерфейсів і даних. Зокрема, Spring Boot забезпечив спрощене конфігурування й запуск програми, Spring Data — зручну роботу з базою даних, а Spring Security — реалізацію автентифікації та авторизації користувачів. У якості системи управління базами даних було використано реляційну БД MySQL [52], що забезпечує високу швидкодію при обробці запитів.

Організація бази даних реалізована за принципом нормалізованої структури, яка відображає основні об'єкти предметної області. Це забезпечує логічну цілісність даних і мінімізує дублювання інформації.

Користувацький інтерфейс системи був спроектований із урахуванням принципів доступності та простоти використання, щоб навіть нефахівці в ІТ могли швидко адаптуватися до роботи з нею. Головний акцент зроблено на інтуїтивну навігацію, мінімізацію кількості кроків до виконання основних дій (створення засідання, додавання учасників тощо. На рисунку 1.1 представлений прототип системи створений у Figma.

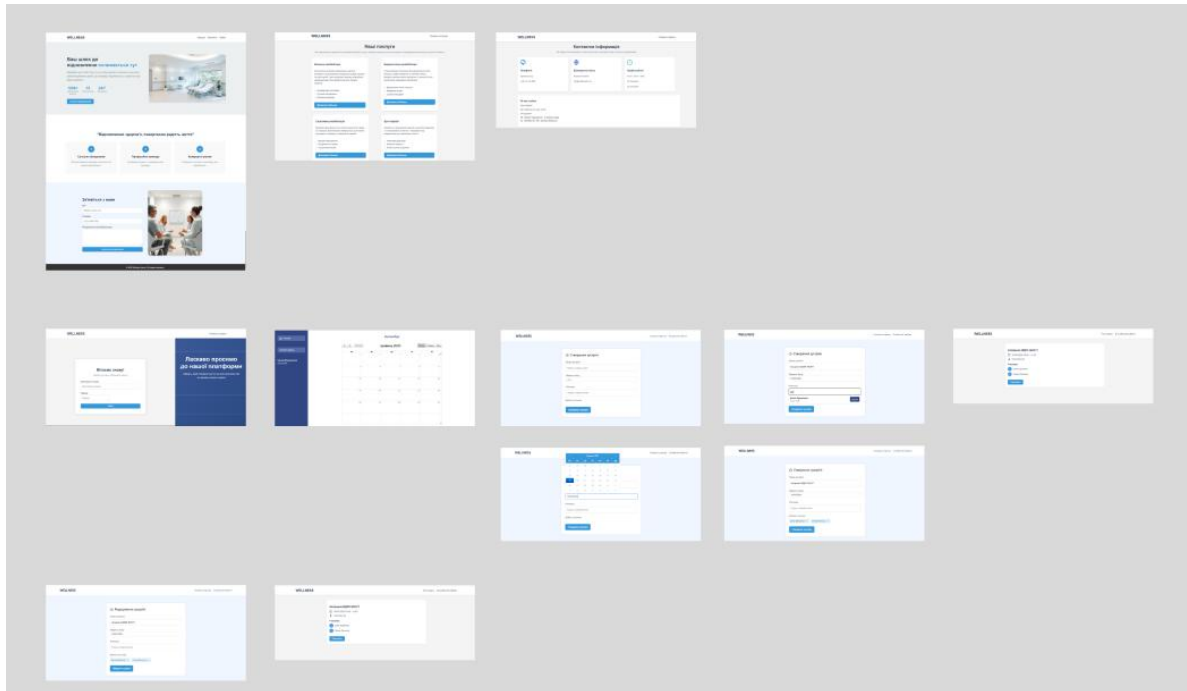


Рис. 1.1 Прототип системи
Джерело: розроблено автором

Для обґрунтування доцільності прийнятих концептуальних рішень було проведено оцінювання за ключовими критеріями: масштабованість, безпека, простота підтримки, інтеграційна здатність, швидкість розробки та виконання функціональних вимог. Наприклад, використання Spring забезпечує високу масштабованість (оцінка 9/10), гнучку безпеку на рівні контролю доступу (9/10) та значну спільноту розробників, що спрощує підтримку (8/10). У порівнянні з іншими фреймворками або платформами рішення на Java хоча й трохи поступається за швидкістю початкової розробки, однак виграє в надійності та відповідності корпоративним стандартам розробки в медичній галузі.

Серед інноваційних рішень, запропонованих у проєкті, варто відзначити механізм підбору вільних часових слотів для створення засідання відповідно до складу МДРК, який реалізовано як окремий сервіс логіки підтримки прийняття рішень.

Процес розробки ІУС включав етапи збору вимог, проектування прототипу системи, її реалізацію, тестування та впровадження. Збір вимог здійснювався на

основі вивчення нормативно-правової бази, а також аналізу процесів у реабілітаційних закладах. Вибір Java та Spring обґрунтований тим, що ця екосистема підтримує створення безпечних, розширюваних та продуктивних застосунків, що є критично важливим у сфері охорони здоров'я.

Таким чином, запропоновані концептуальні рішення для створення ІС у сфері управління засіданнями МДРК базуються на глибокому аналізі потреб предметної області, а саме реабілітаційного центру, сучасних технологічних підходах та оцінці альтернатив за ключовими критеріями ефективності. Їх реалізація дозволяє не лише автоматизувати рутинні адміністративні процеси, а й підвищити якість прийняття рішень у реабілітаційній практиці, що у підсумку позитивно впливає на результати реабілітації пацієнтів.

РОЗДІЛ 2. ХАРАКТЕРИСТИКА ІНФОРМАЦІЙНИХ УПРАВЛЯЮЧИХ СИСТЕМ ТА МЕТОДИ І МОДЕЛІ

2.1 Структура і характеристика системи

При проектуванні інформаційної системи одним із найважливіших кроків є чітке визначення основних зацікавлених сторін — стейкхолдерів, які взаємодіятимуть із системою, формуватимуть до неї вимоги та, зрештою, користуватимуться її функціональністю в процесі повсякденної роботи. У контексті інформаційної системи для реабілітаційного центру такими стейкхолдерами виступають такі основні категорії користувачів: працівники центру, до яких входить і адміністратор та звичайні користувачі (пацієнти або відвідувачі веб-сайту). Кожна з цих груп має свої потреби, які прямо впливають на формування як функціональних, так і нефункціональних вимог до системи.

Адміністратор виконує системно-технічні функції: керує користувачами, контролює наповнення та оновлення інформації в системі. Працівники реабілітаційного центру зосереджені на своїй медичній та організаційній роботі, тому система для них повинна забезпечувати швидкий доступ до особистого календаря, можливість створення та редагування зустрічей, зокрема засідань мультидисциплінарної реабілітаційної команди (МДРК). Останні мають ключове значення, адже саме на таких засіданнях ухвалюються рішення щодо формування індивідуального реабілітаційного плану пацієнта. Звичайні користувачі ж потребують інтерфейсу для ознайомлення з інформацією про центр та послуги, а також можливості подати онлайн-запит на консультацію, для подальшої обробки його адміністратором.

На основі аналізу очікувань і потреб стейкхолдерів були сформульовані функціональні вимоги до інформаційної системи. [49] Серед них: можливість перегляду інформації про реабілітаційний центр, створення запиту на консультацію, вхід в особистий кабінет, створення, редагування і перегляд

зустрічей у календарі для подій, в тому числі і засідань МДРК. Всі ці функціональні можливості забезпечують ефективне виконання ключових процесів, таких як планування засідань МДРК та комунікація між пацієнтами й медичними працівниками.

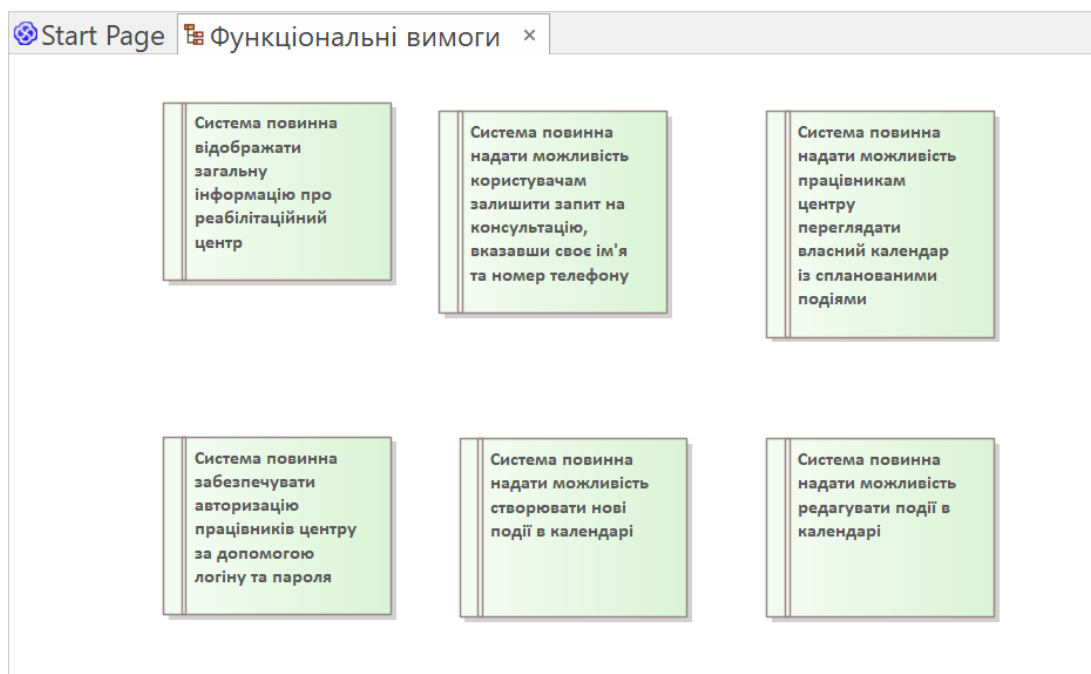


Рис. 2.1 Функціональні вимоги
Джерело: розроблено автором

Item	Stereotype	Status	Difficulty	Priority
☒ Система повинна відображати загальну інформацію про реабілітаційний центр	Functional...	Proposed	Low	High
☒ Система повинна забезпечувати авторизацію працівників центру за допомогою логіну та пароля	Functional...	Proposed	High	High
☒ Система повинна надати можливість користувачам залишити запит на консультацію, вказавши своє ім'я та номер телефону	Functional...	Proposed	Medium	Medium
☒ Система повинна надати можливість працівникам центру переглядати власний календар із спланованими подіями	Functional...	Proposed	High	High
☒ Система повинна надати можливість редагувати події в календарі	Functional...	Proposed	Medium	Medium
☒ Система повинна надати можливість створювати нові події в календарі	Functional...	Proposed	High	High

Рис. 2.2 Специфікація функціональних вимог, створена за допомогою менеджера специфікацій (Specification Manager) Enterprise Architect

Джерело: розроблено автором

Поряд із функціональними вимогами, надзвичайно важливим є врахування і нефункціональних вимог, саме характеристик системи, які не стосуються безпосередньо її функціональності, але суттєво впливають на зручність використання, надійність та ефективність роботи.[34] У нашому випадку до таких нефункціональних вимог належать, зокрема, зручність і доступність інтерфейсу; швидкодія системи при одночасному доступі великої кількості користувачів; безпека персональних даних, яка передбачає авторизацію. Також важливим нефункціональним аспектом є підтримуваність і можливість адаптації системи до змін, оскільки нормативне поле охорони здоров'я може змінюватися, як і самі потреби реабілітаційних центрів. Це означає, що архітектура системи повинна бути достатньо гнучкою, щоб забезпечити внесення змін без істотної перебудови коду чи бази даних.

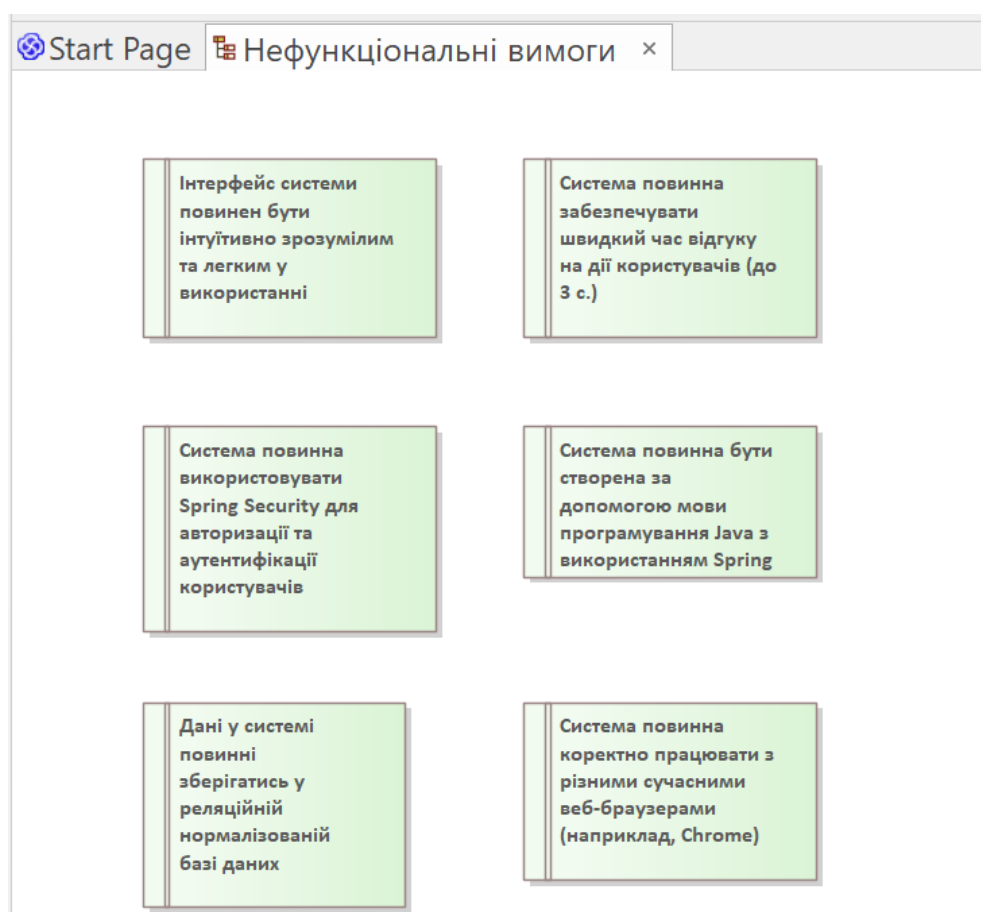


Рис. 2.3 Нефункціональні вимоги
Джерело: розроблено автором

Start Page Нефункціональні вимоги Specification Manager x				
Model One Level Requirement Hierarchy Find Package				
Item	Stereotype	Status	Difficulty	Priority
☒ Дані у системі повинні зберігатись у реляційній нормалізованій базі даних	Nonfunctio...	Proposed	Medium	High
☒ Інтерфейс системи повинен бути інтуїтивно зрозумілим та легким у використанні	Nonfunctio...	Proposed	Medium	Medium
☒ Система повинна бути створена за допомогою мови програмування Java з використанням Spring	Nonfunctio...	Proposed	Medium	High
☒ Система повинна використовувати Spring Security для авторизації та аутентифікації користувачів	Nonfunctio...	Proposed	Medium	High
☒ Система повинна забезпечувати швидкий час відгуку на дії користувачів (до 3 с.)	Nonfunctio...	Proposed	Medium	High
☒ Система повинна коректно працювати з різними сучасними веб-браузерами (наприклад, Chrome)	Nonfunctio...	Proposed	Medium	Medium

Рис. 2.4 Специфікація нефункціональних вимог, створена за допомогою менеджера специфікацій (Specification Manager) Enterprise Architect

Джерело: розроблено автором

Загалом, визначення стейкхолдерів та формулювання функціональних і нефункціональних вимог є базисом для проєктування інформаційної системи. Ці вимоги утворюють підґрунтя для створення структурної та поведінкової моделі системи, яка здатна задовольнити потреби як звичайних користувачів, так і медичного персоналу, при цьому забезпечуючи відповідність державним вимогам і стандартам безпеки.

На основі функціональних вимог було створено діаграму прецедентів (див. рис. 2.5)

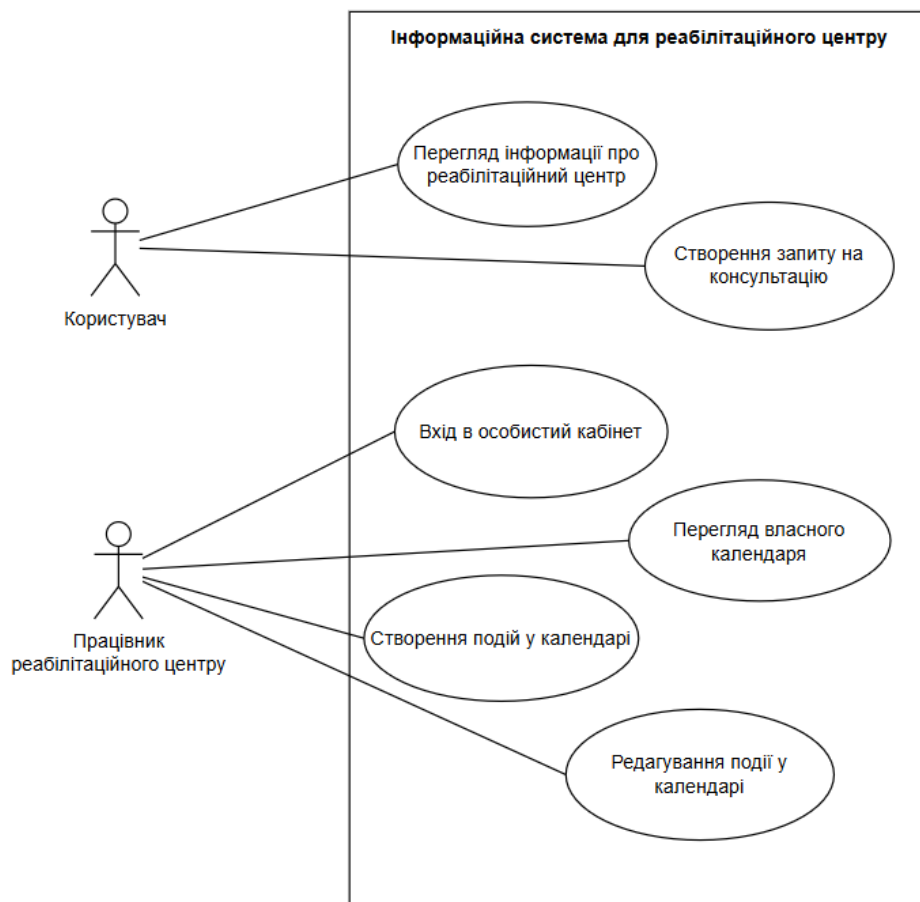


Рис. 2.5 Діаграма прецедентів
Джерело: розроблено автором

Діаграма прецедентів демонструє взаємодію між користувачами та працівниками реабілітаційного центру з створеною інформаційною системою. Користувач, представлений як пацієнт, має можливість переглядати інформацію про центр, створювати запит на консультацію. З іншого боку, працівник реабілітаційного центру може входити до особистого кабінету, а крім того, йому доступні функції перегляду власного календаря, створення нових подій у ньому та редагування вже існуючих записів. Отже, діаграма наочно представляє ключові функціональні можливості інформаційної системи з урахуванням ролей різних зацікавлених сторін. [27]

Відповідно до кожного варіанта використання, наведено діаграму послідовності [26], яка відображає, які саме класи будуть використані та послідовність їх використання. На рисунку 2.6 зображено такий прецедент, як

перегляд інформації про центр, на рисунку 2.7 можна побачити як відбувається вхід до особистого кабінету користувача, на рисунку 2.8 зображено створення запиту на консультацію, на рисунку 2.9 наведено послідовність дій в системі при відкритті особистого кабінету та показу журналу подій, на рисунку 2.10 представлено кроки для створення нової події в календарі, на рисунку 2.11 можна побачити, як відбувається редагування уже наявної події.

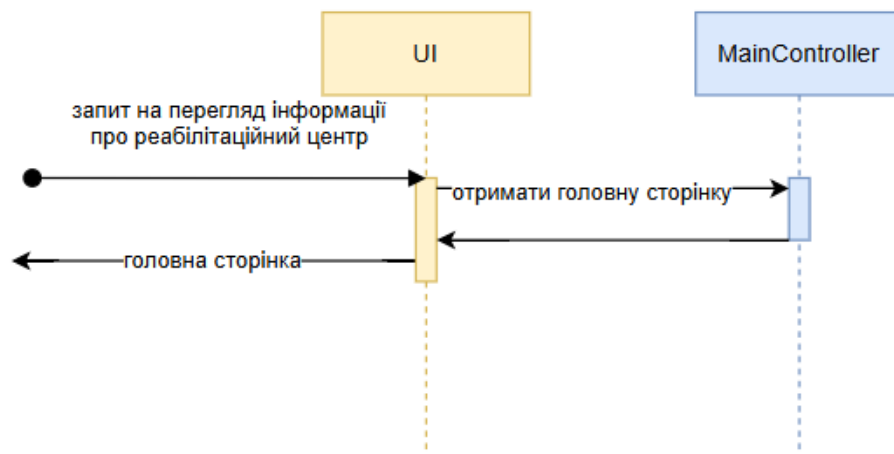


Рис. 2.6 Діаграма послідовності. Перегляд інформації про реабілітаційний центр

Джерело: розроблено автором

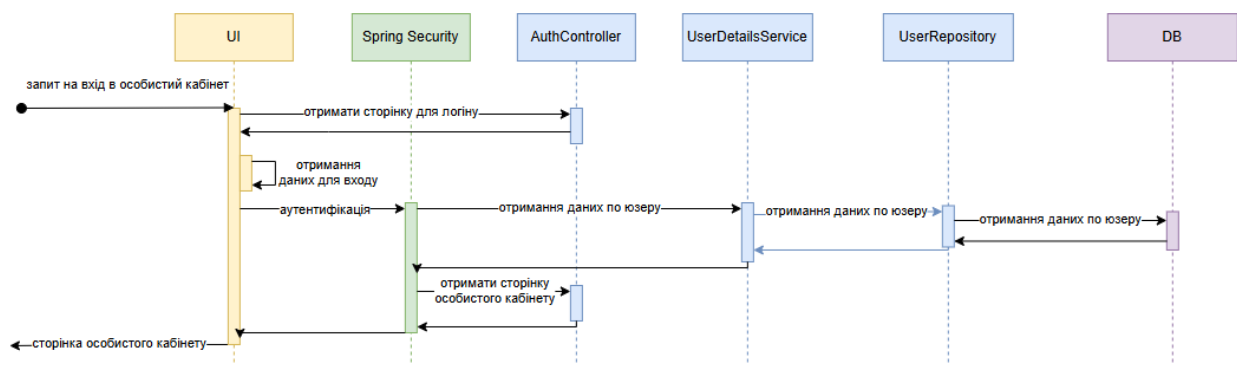


Рис. 2.7 Діаграма послідовності. Вхід в особистий кабінет

Джерело: розроблено автором

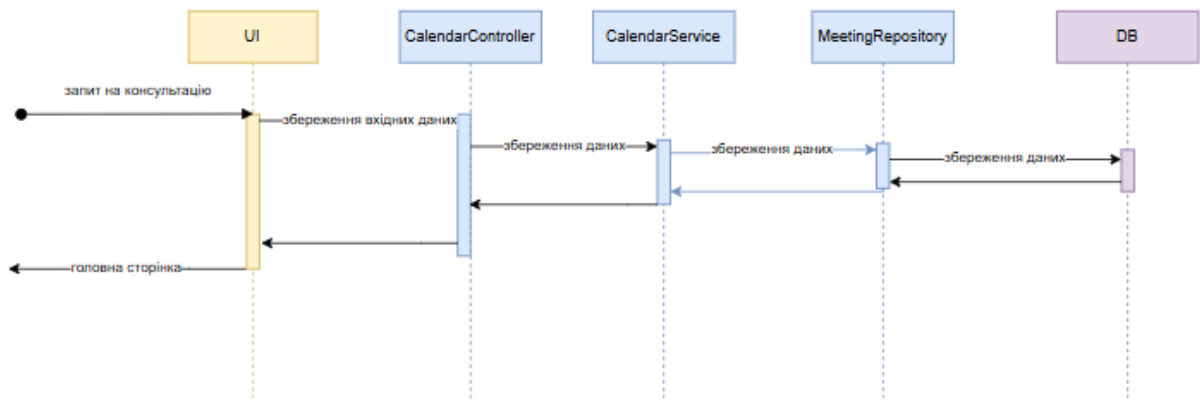


Рис. 2.8 Діаграма послідовності. Створення запиту на консультацію
Джерело: розроблено автором

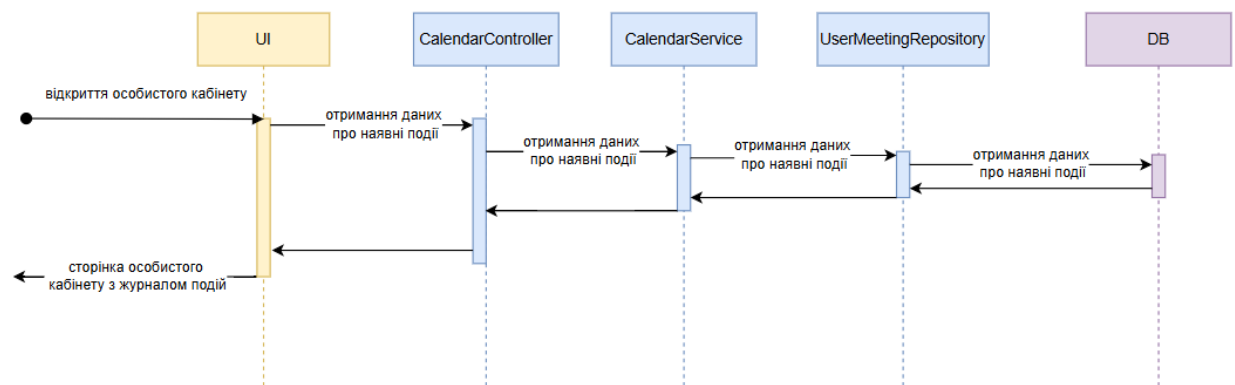


Рис. 2.9 Діаграма послідовності. Відкриття особистого кабінету та перегляд журналу подій
Джерело: розроблено автором

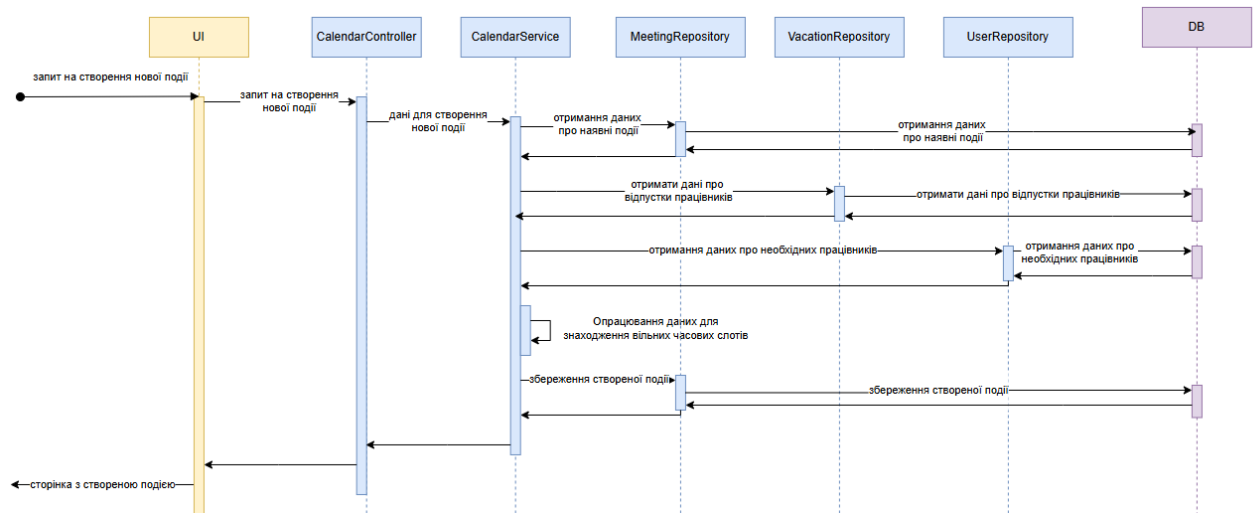


Рис. 2.10 Діаграма послідовності. Створення нової події в календарі

Джерело: розроблено автором

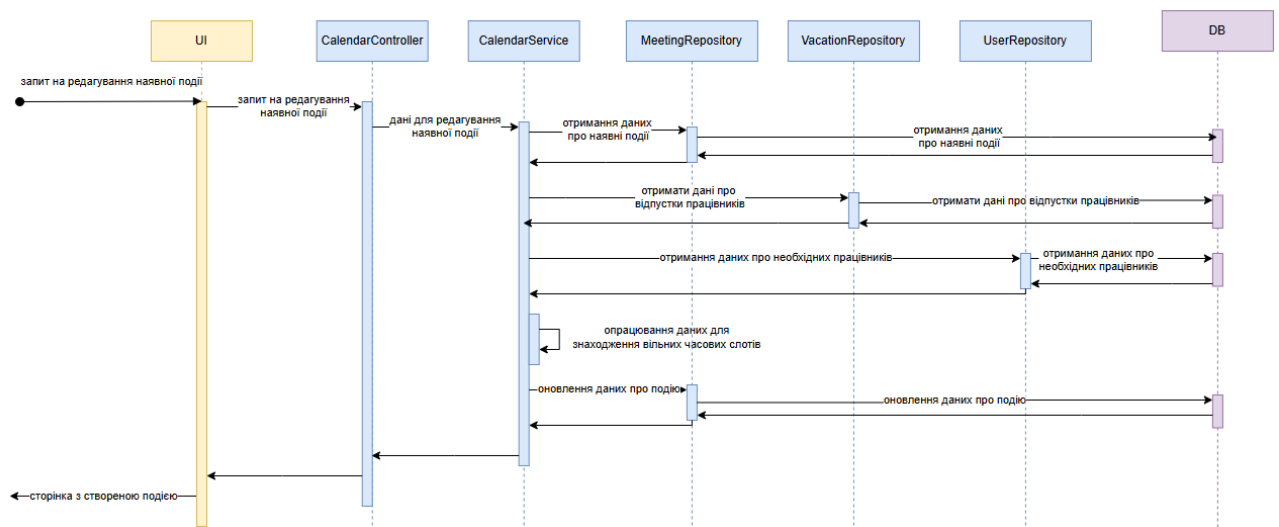


Рис. 2.11 Діаграма послідовності. Редагування уже наявної події

Джерело: розроблено автором

2.2 Методи та моделі в інформаційних управляючих системах і технологіях

У сучасних інформаційних управляючих системах, особливо в соціально-орієнтованих сферах, таких як охорона здоров'я, значну складність становить організація процесу прийняття управлінських рішень. У межах даної роботи розглядалося питання автоматизації планування засідань мультидисциплінарної реабілітаційної команди (МДРК) у реабілітаційному центрі. Це завдання належить до категорії слабоструктурованих, адже воно поєднує різнотипні дані, враховує графіки роботи, людський фактор (наприклад, відпустки), часові обмеження та інші змінні параметри. В умовах багатокритеріальності, коли необхідно узгодити наявність кількох працівників, вільний простір у календарі закладу та індивідуальні розклади кожного з учасників команди, доцільним є застосування інструментів штучного інтелекту, зокрема методів продукційної логіки.

Для реалізації даного функціоналу в межах інформаційної системи було обрано Drools — систему керування бізнес-правилами, що реалізує механізм продукційних правил. Це дозволяє організувати логіку підбору вільного часу як набір умовно-наслідкових операторів, які можна динамічно змінювати та адаптувати без суттєвих змін у програмному коді. Такий підхід особливо цінний у сфері, що швидко змінюється й вимагає гнучкої адаптації логіки планування до нових умов. [4]

Основна ідея побудови моделі підбору полягала в тому, щоб, маючи інформацію про загальний графік роботи реабілітаційного центру (тобто час, коли можна проводити зустрічі), наявні події в календарях необхідних працівників, які входять до складу МДРК (індивідуальні консультації, вже заплановані засідання, службові наради), відпустки та періоди відсутності працівників, автоматично сформувати список вільних часових слотів, придатних для створення нової зустрічі, включно із засіданням МДРК.

Кожне правило в системі Drools формалізує одну із умов, яка впливає на рішення: наприклад, «якщо співробітник перебуває у відпустці в зазначений період, то цей слот недоступний» або «якщо для заданої дати й часу всі необхідні працівники не мають подій у календарі, то слот позначається як потенційно вільний». Система перебирає всі можливі комбінації часу в межах допустимого періоду й відсіює ті, які суперечать хоча б одному з правил.

З математичної точки зору, задача пошуку вільних слотів для зустрічей у подібних умовах є комбінаційною та багатокритеріальною. З урахуванням того, що дані надходять із різних джерел, а сам процес прийняття рішення має характер нечіткого й евристичного, застосування продукційної системи дозволяє ефективно структурувати знання у вигляді набору експертних правил, що легко масштабуються. Це особливо важливо в умовах, коли кількість працівників збільшується або додаються нові умови, наприклад, обов'язковість присутності певних фахівців на кожному засіданні.

Слід підкреслити, що така модель не тільки автоматизує процес планування, а й підвищує якість управлінських рішень, дозволяючи враховувати більше факторів, ніж це було б можливим при ручному плануванні. Крім того, використання Drools як інструменту підтримки прийняття рішень відповідає сучасним підходам до створення гнучких, адаптивних інформаційних управляючих систем із елементами штучного інтелекту. Цей підхід належить до класу систем, які вирішують слабоструктуровані задачі за допомогою формалізованого опису знань та логічного виводу, зберігаючи прозорість і пояснюваність прийнятих рішень.

Отже, у контексті задачі автоматизації процесу підбору часу для зустрічей, зокрема МДРК, було використано сучасний метод штучного інтелекту — продукційні правила, реалізовані через Drools, що забезпечило ефективність, адаптивність і масштабованість системи при збереженні високої якості підтримки прийняття управлінських рішень.

РОЗДІЛ 3. РОЗРОБЛЕННЯ ПРОЄКТНИХ РІШЕНЬ ТА ЇХ РЕАЛІЗАЦІЯ

3.1. Проєктування бази даних для інформаційної управляючої системи

3.1.1 Проєктування інфологічної (логічної) моделі БД

Для бази даних інформаційної системи реабілітаційного центру було створено такі таблицьки як: «user», яка б містила усю необхідну інформацію про користувачів, «role» - для зберігання ролей користувачів та «user_role», яка б поєднувала користувачів та їх роль, а також «meeting» - для зберігання інформації про події, «vacation» - для зберігання даних про відпустку та вихідні дні працівників, «user_meeting» - для зберігання даних про те, які працівники для якої саме події мають бути залучені, а також , «consultation_request» для тих реквестів на консультацію , які створили користувачі системи і які в подальшому будуть опрацьовані працівниками реабілітаційного центру. Нижче на рисунку 3.1 представлена інфологічна модель розробленої бази даних.

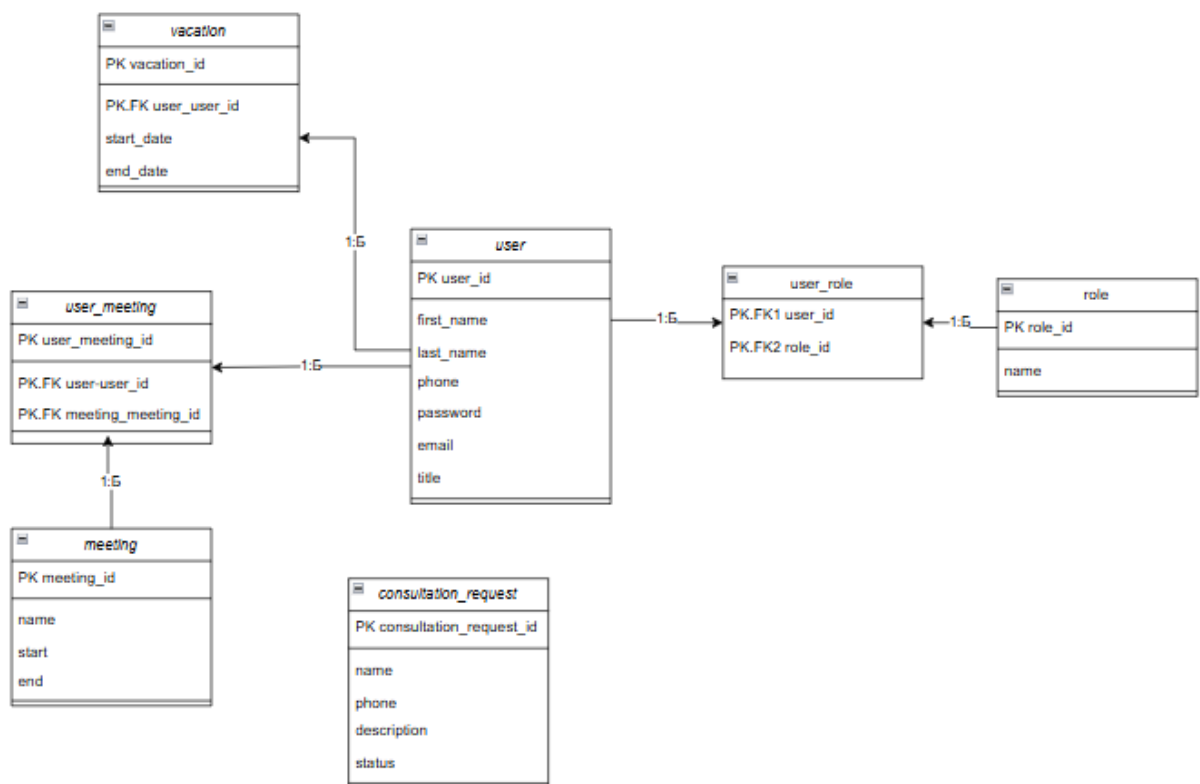


Рис. 3.1 Інфологічна модель бази даних
Джерело: розроблено автором

3.1.2 Обґрунтування вибору СКБД та проєктування фізичної моделі БД в середовищі вибраної СКБД

Для інформаційної системи реабілітаційного центру було обрано MySQL як систему управління базами даних (СУБД), що є обґрунтованим рішенням, яке базується на її функціональних можливостях, а також надійності та відповідності вимогам до зберігання й обробки даних про користувачів і їх ролі, працівників і їх відпустки, заплановані зустрічі і консультації. [13;46;48]

MySQL має ряд переваг, а саме:

- MySQL є реляційною СУБД, яка дозволяє ефективно моделювати та підтримувати зв'язки між сутностями, такими як користувачі, ролі, працівники, відпустки та ін. Використання зовнішніх ключів забезпечує цілісність даних та унеможливорює появу неконсистентних записів[14;33]
- гарантує атомарність, узгодженість, ізолюваність і довговічність операцій, що критично важливо для обробки даних, де недопустимі часткові або некоректні зміни.
- надає розширені можливості управління доступом, включаючи ролі, політики безпеки, шифрування даних та аудит дій користувачів.
- підтримує реплікацію та кластеризацію, що забезпечує масштабованість системи та безперебійну роботу навіть при зростанні кількості користувачів або обсягу даних. [47]
- оптимізована для швидкої обробки запитів і підтримує індексацію, що дозволяє ефективно працювати з великими обсягами даних без значних затримок.
- MySQL - це СУБД з відкритим кодом, що дозволяє знизити витрати на ліцензування та забезпечує гнучкість у налаштуванні та розширенні функціональності системи.

- сумісний з основними операційними системами (Windows, Linux, macOS) та інтегрується з популярними мовами програмування, в тому числі Java, що спрощує розробку та підтримку інформаційної системи.
- MySQL є надійною, безпечною та масштабованою СУБД, яка ідеально підходить для побудови інформаційної системи реабілітаційного центру, адже її функціональні можливості забезпечують ефективне управління даними користувачів, працівників та планування засідань МДПК, що сприяє підвищенню якості обслуговування та оптимізації внутрішніх процесів центру. [6]

Для розробки і адміністрування такої системи було використано офіційний графічний інструмент — MySQL Workbench, універсальне середовище, яке дозволяє проєктувати базу даних за допомогою ER-діаграм, створювати таблиці та зв'язки між ними у візуальному режимі, а також автоматично генерувати SQL-код. Крім того, MySQL Workbench містить редактор запитів із підсвічуванням синтаксису, що дозволяє легко писати, перевіряти та оптимізувати SQL-запити, збережені процедури й тригери. Він також надає зручні засоби для імпорту та експорту даних, що особливо корисно при первинному заповненні таблиць чи створенні резервних копій. За допомогою вбудованих інструментів моніторингу продуктивності адміністратор може відстежувати навантаження, виконання запитів та можливі помилки в реальному часі. [15]

На рисунку 3.2 представлена даталогічна модель спроектованої бази даних у MySQL Workbench, яка відповідає усім умовам нормалізації, а саме: усі атрибути неподільні та мають унікальні імена – 1НФ, інформаційні об'єкти не містять неповних функціональних залежностей – 2НФ, немає транзитивних залежностей між неключовими атрибутами – 3НФ та немає багатозначних залежностей – 4 НФ.

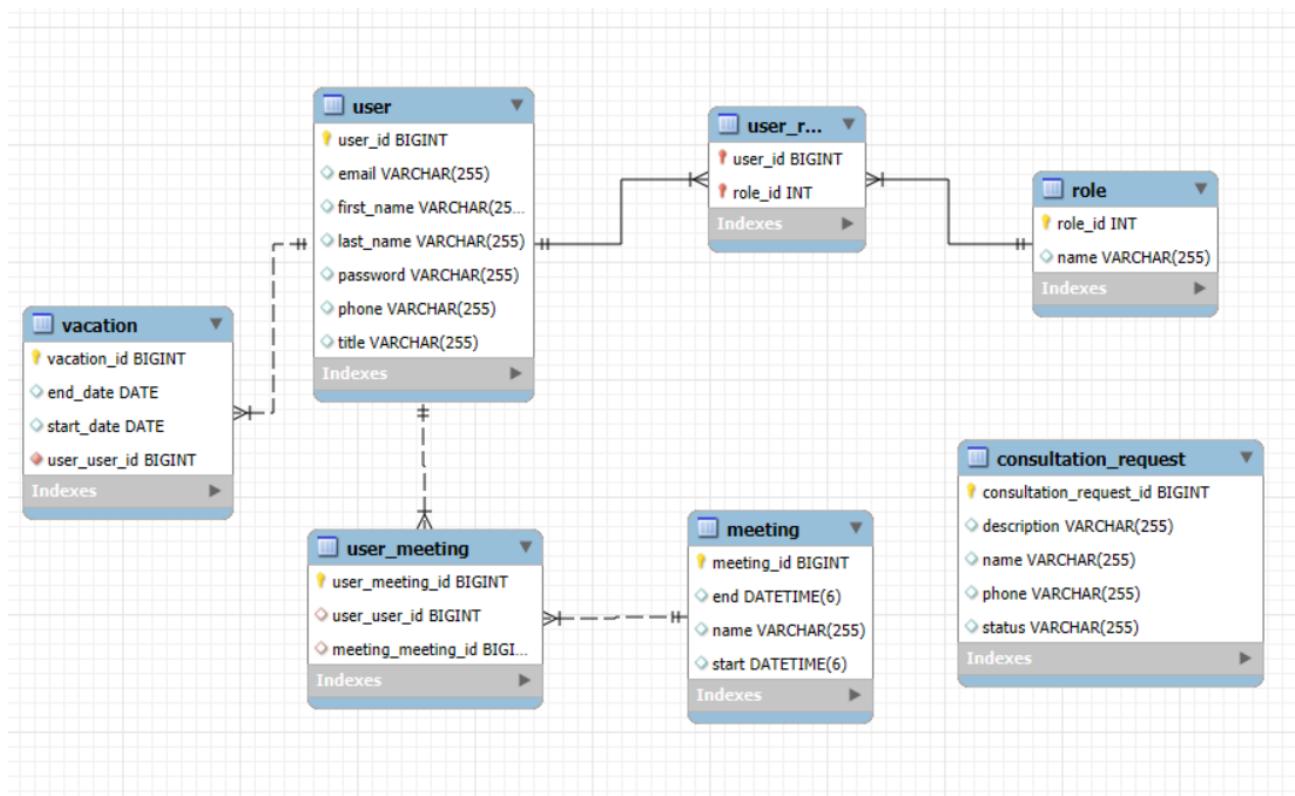


Рис. 3.2 Даталогічна модель бази даних
Джерело: розроблено автором

3.1.3 Контрольний приклад та обрахунок прогнозних обсягів БД

Для контрольного прикладу було завантажено 10 записів користувачів (таблиця user), 10 записів юзер ролей у таблицю user_role, 2 записи у таблицю role, 7 засідань МДРК у таблицю meeting, 21 зв'язків “user_meeting” для реєстрації участі працівників у цих засіданнях, 5 заявок на консультацію у таблицю consultation_request та 5 періодів відпусток у vacation таблицю. Усі таблицьки заповнені тестовими даними наведені у Додатку А.

Для прогнозного обчислення обсягу бази даних у MySQL було прийнято часовий горизонт у шість місяців та визначено вихідні характеристики для чотирьох ключових таблиць. Так, для таблиці meeting середня довжина одного запису складає 2 340 байт, а початковий обсяг даних становить 16 КБ (16 384 байт). Наприклад, щомісяця до цієї таблиці додається 10 нових засідань, що дає

приріст 23 400 байт на місяць, або 140 400 байт за півроку. Таким чином, через шість місяців загальний обсяг даних у таблиці meeting зросте до 156 784 байт, що приблизно дорівнює 153,1 КБ.

У таблиці user_meeting, яка зберігає зв'язки між користувачами та засіданнями, середній розмір одного рядка становить 780 байт. Початковий обсяг даних—16 КБ (16 384 байт)—доповнюється щомісяця приблизно 40 новими зв'язками, що відповідає приросту 31 200 байт на місяць або 187 200 байт за шість місяців. За підсумками прогнозу обсяг даних у таблиці user_meeting сягне 203 584 байт, або приблизно 198,8 КБ. Додатково для цієї таблиці передбачено початковий обсяг індексів 32 КБ, який слід врахувати при розрахунку повного розміру таблиці.

Таблиця consultation_request, що містить заявки на консультацію, має середню довжину одного запису 3 276 байт і початковий обсяг даних 16 КБ (16 384 байт). При надходженні щомісяця п'яти нових заявок (16 380 байт приросту) її розмір за шість місяців збільшиться на 98 280 байт, досягнувши 114 664 байт (приблизно 112,0 КБ).

У таблиці vacation, де фіксуються періоди відпусток, середня довжина запису також становить 3 276 байт, а початковий обсяг як даних, так і індексів—по 16 КБ (16 384 байт). Щомісяця в цю таблицю додається три записи (9 828 байт приросту), тобто за півроку загальний приріст становить 58 968 байт, і обсяг даних досягає 75 352 байт (близько 73,6 КБ).

Таким чином, контрольний приклад підтвердив коректність проєктування структури БД і дозволяє планувати її подальше зростання з урахуванням реальних навантажень у реабілітаційному центрі.

3.2. Проектування бази знань інформаційної управляючої системи

3.2.1 Реалізація інтелектуального аналізу великих даних з метою наповнення бази знань

У розробленій інформаційній системі для планування засідань МДРК реабілітаційного центру база знань відіграє центральну роль, адже саме вона забезпечує інтелектуальний відбір вільних часових слотів на основі графіка роботи центру, даних про відпустки працівників і вже запланованих подій. На рисунку 3.3 представлено дерево рішень [40] для виявлення вільних часових слотів для подальшого створення подій, в тому числі засідань МДРК реабілітаційного центру.

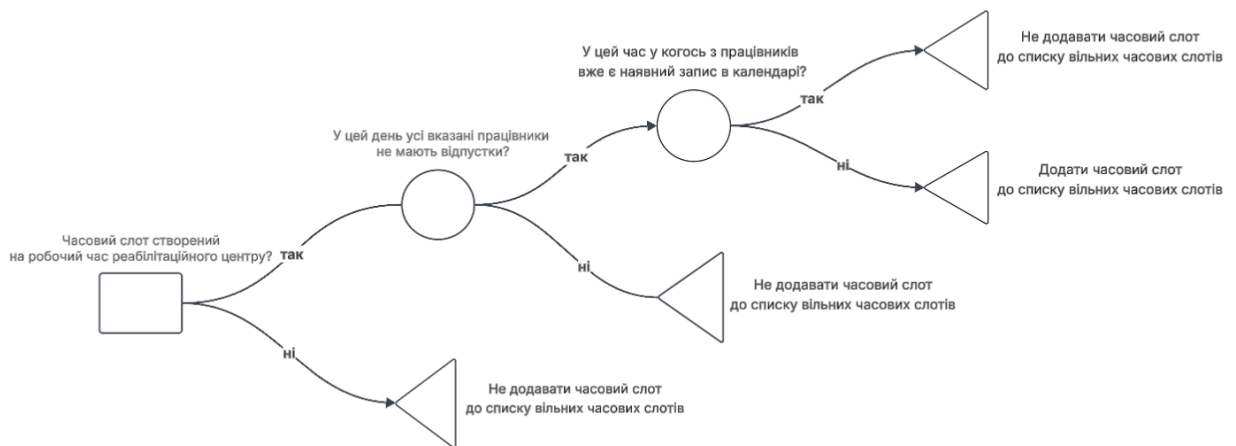


Рис. 3.3 Дерево рішень
Джерело: розроблено автором

Дерево рішень для підбору вільних часових слотів для створення засідання МДРК реабілітаційного центру починається з перевірки, чи створений часовий слот на робочий час реабілітаційного центру. Якщо ні, то цей часовий слот одразу відхиляється і не додається до списку вільних. Якщо ж часовий слот створений на робочий час, то наступним кроком є перевірка, чи всі вказані працівники не мають відпустки в цей день. Якщо хоча б один працівник має відпустку, цей часовий слот також не додається до списку вільних. У випадку, коли всі працівники не мають відпустки в цей день, система перевіряє, чи є у когось з них вже наявний запис у календарі на цей конкретний час. Якщо хоча б

один працівник має заплановану подію, цей часовий слот не буде додано до списку вільних. Лише за умови, що часовий слот створений на робочий час реабілітаційного центру, усі вказані працівники не мають відпустки і ні в кого з них немає запису в календарі на цей час, цей часовий слот буде додано до списку вільних для планування зустрічі МДРК.

3.2.2 Проектування та реалізація бази знань інформаційної управляючої системи

Для інформаційної системи реабілітаційного центру у якості механізму реалізації бази знань було обрано продукційні правила, описані у DRL-файлах та оброблювані за допомогою Drools.

На концептуальному рівні база знань містить три типи об'єктів: потенційні часові слоти (TimeSlotDto) з параметрами початку та кінця, об'єкти Meeting, які фіксують вже заплановані зустрічі з атрибутами часу початку і завершення, а також сутності Vacation із зазначенням дат початку і кінця відпустки кожного працівника. Усі ці факти завантажуються у KieSession, що є ядром Drools, де й відбувається їх обробка.

Продукційні правила реалізовані так, щоб послідовно усувати з набору потенційних слотів ті, що конфліктують зі вже зайнятими інтервалами чи випадають на час відпусток. Наприклад, правило «Remove slot that overlaps with meeting» видаляє з сесії всі слоти, для яких початок чи кінець перетинає інтервал будь-якої зустрічі, тоді як «Remove slot that overlaps with vacation» виявляє та усуває ті інтервали, які потрапляють на дні відсутності співробітника, порівнюючи дати початку слота з діапазоном відпустки.

Архітектурно робота системи будується на кількох етапах. Спочатку на основі робочого графіка формуються всі потенційні слоти фіксованої тривалості, потім разом із інформацією про заплановані зустрічі та відпустки вони

завантажуються до Drools, після чого викликається `fireAllRules`, що запускає процес виведення та відсіювання невідповідних слотів. Після застосування правил отримані вільні часові слоти можна використовувати для створення нових подій у календарі.

Таким чином, запропонований підхід із використанням продукційних правил на основі Drools забезпечує гнучкість та масштабованість системи — зміни в доменній логіці вносяться швидко й безпечно, а інтелектуальні підходи до аналізу даних гарантують, що пропонувані часові слоти максимально відповідають реальним потребам реабілітаційного центру.

3.3. Програмне забезпечення

3.3.1 Проектування програмного забезпечення

Для розробки інформаційної управляючої системи реабілітаційного центру, що реалізує функціонал планування подій, зокрема знаходження вільних часових слотів для організації засідань мультидисциплінарної реабілітаційної команди (МДРК), було використано низку сучасних технологій та інструментів. На основі попередньо сформульованого технічного завдання для побудови архітектури програмного продукту було обрано модель MVC (Model-View-Controller).

Цей шаблон передбачає поділ системи на три взаємопов'язані частини: модель даних, вигляд, тобто інтерфейс користувача та модуль керування. Застосовується для відокремлення даних (моделі) від інтерфейсу користувача (вигляду) так, щоб зміни інтерфейсу користувача мінімально впливали на роботу з даними, а зміни в моделі даних могли здійснюватися без змін інтерфейсу користувача. [12]

Він дозволяє чітко розмежувати логіку представлення, бізнес-логіку та доступ до даних, що значно полегшує супровід, тестування та масштабування системи в майбутньому.

У якості основної мови програмування було обрано Java, об'єктно-орієнтовану мову програмування, що забезпечує високу надійність, переносимість та безпеку. Також було використано фреймворк Spring, зокрема модулі Spring Boot [18] для створення контролерів, Spring Data JPA [19] для доступу до бази даних, а також Spring Security — для реалізації ролей та автентифікації користувачів.[20] Для підвищення читабельності програмного коду, шляхом зменшення шаблонного коду було застосовано бібліотеку Lombok, яка автоматично генерує геттери, сеттери, конструктори тощо. Для мапінгу об'єктів між DTO і сутностями використовувався MapStruct — високопродуктивний фреймворк для об'єктного мапінгу, який мінімізує кількість ручного коду та помилок.

Для реалізації інтелектуальної частини функціоналу з пошуку вільних часових слотів для подій було інтегровано систему правил Drools. Це дозволило винести логіку перевірки складних умов, таких як наявність часу у всіх учасників МДРК, у вигляді окремих правил, які легко змінюються без перезбирання програмного продукту.

Візуальна частина ІС реалізована за допомогою HTML, CSS, JavaScript, Bootstrap[2] та шаблонізатора Thymeleaf [22], що інтегрується з Spring і дозволяє формувати динамічні веб-сторінки, зберігаючи зручність і адаптивність інтерфейсу.

Для зберігання даних обрано СУБД MySQL, що відповідає вимогам до масштабованості та надійності, а також підтримується широким набором інструментів, зокрема MySQL Workbench, який застосовувався для візуального моделювання структури бази даних та виконання SQL-запитів.

Розробка програмного забезпечення здійснювалась у середовищі IntelliJ IDEA, яке забезпечує потужну підтримку для Java і Spring, а також має інтегровані засоби для тестування, налагодження та управління проєктами.

Під час розробки архітектури системи використовувались інструменти моделювання, зокрема Enterprise Architect[5], Draw.io [3] та Lucidchart, що дозволило створити діаграми прецедентів та послідовностей, які забезпечили системність та комплексність проєктування. Розроблене ПЗ забезпечує простоту у використанні завдяки інтуїтивно зрозумілому інтерфейсу, що тестувався у веб-браузері Google Chrome.

Таким чином, вибір технологій і підходів до розробки інформаційної системи був цілком обґрунтований цілями проєкту, саме: побудова надійного, безпечного, гнучкого і масштабованого програмного продукту, що дозволяє ефективно координувати роботу фахівців реабілітаційного центру та автоматизувати ключові аспекти планування, пов'язані з організацією засідань МДРК.

3.3.2 Вибір архітектури програмного забезпечення

Для розробки інформаційної системи реабілітаційного центру, що забезпечує функціонал планування засідань мультидисциплінарної реабілітаційної команди (МДРК), було обрано монолітну архітектуру. Це рішення було обґрунтоване поточними вимогами до системи, обсягом функціональності та ресурсами, доступними для розробки та підтримки.

Загалом, монолітна архітектура передбачає об'єднання всіх компонентів системи в єдину програму, що спрощує розгортання, тестування та обслуговування на початкових етапах розробки. Саме це дозволило швидко реалізувати основний функціонал створеної інформаційної системи реабілітаційного центру, забезпечити цілісність бізнес-логіки та зменшити

складність інтеграції між модулями. Крім того, монолітна структура полегшила відстеження та усунення помилок, оскільки всі компоненти знаходяться в одному кодовому базисі. [10]

У контексті безпеки, монолітна архітектура дозволяє ефективно реалізувати механізми аутентифікації та авторизації користувачів. Для цього використовується Spring Security з шифруванням паролів за допомогою алгоритму BCrypt, що забезпечує захист від атак типу brute-force та rainbow tables.

Незважаючи на переваги монолітної архітектури на початкових етапах, з часом, у разі розширення функціональності та зростання кількості користувачів, може виникнути потреба в переході до мікросервісної архітектури. Це дозволить масштабувати окремі компоненти системи незалежно один від одного, підвищуючи стійкість до збоїв та спрощуючи впровадження нових функцій.

Таким чином, обраний підхід з монолітною архітектурою дозволяє ефективно реалізувати поточні вимоги до ІС реабілітаційного центру, забезпечуючи при цьому можливість подальшого масштабування та переходу до мікросервісної архітектури в майбутньому.

3.3.3 Інструментальні засоби розробки для створення прикладного програмного забезпечення інформаційних управляючих систем

У процесі проєктування інформаційної управляючої системи реабілітаційного центру було використано сучасні інструментальні засоби, які забезпечують ефективну реалізацію функціональних вимог, зокрема планування подій та управління даними.

Основною мовою програмування для серверної частини обрано Java, яка завдяки своїй об'єктно-орієнтованій природі та широкому спектру бібліотек є надійною основою для створення масштабованих веб-додатків. Для спрощення

конфігурації та прискорення розробки було використано фреймворк Spring Boot, який забезпечує автоматичне налаштування компонентів та інтеграцію з іншими модулями екосистеми Spring.

Для взаємодії з реляційною базою даних MySQL було застосовано Spring Data JPA, що реалізує об'єктно-реляційне відображення (ORM) та дозволяє розробникам працювати з базою даних через об'єкти Java без необхідності написання SQL-запитів. Це забезпечує зручне управління з'єднаннями, перетворення запитів у об'єкти та інтеграцію з бізнес-логікою програми. Конфігурація підключення до бази даних здійснюється через файл `application.properties`, де вказуються параметри з'єднання, такі як URL, ім'я користувача та пароль.

Для представлення даних на стороні клієнта використовується шаблонізатор Thymeleaf, який інтегрується з Spring MVC та дозволяє створювати динамічні HTML-сторінки. Thymeleaf підтримує концепцію "природних шаблонів", що означає, що шаблони можуть бути відкриті в браузері без попередньої обробки сервером, що спрощує процес розробки та тестування інтерфейсу користувача.

Для забезпечення безпеки аутентифікації користувачів у системі використовується Spring Security з шифруванням паролів за допомогою алгоритму BCrypt. Аутентифікація здійснюється через логін та пароль, що є стандартним підходом для монолітних додатків.

У процесі розробки також використовувалися інструменти для автоматичного генерування коду та мапінгу об'єктів. Бібліотека Lombok дозволила зменшити кількість шаблонного коду, автоматично генеруючи геттери, сеттери та конструктори [8], а MapStruct забезпечив ефективне перетворення між об'єктами DTO та сутностями, що мінімізувало кількість ручного коду та помилок. [9]

Проектування дизайну системи здійснювалось у Figma, а розробка самої ІС здійснювалася у середовищі IntelliJ IDEA, яке забезпечує потужну підтримку для Java і Spring, а також має інтегровані засоби для тестування, налагодження та управління проєктами.

Таким чином, обраний інструментарій забезпечує ефективну розробку, підтримку та масштабування для інформаційних систем, відповідаючи сучасним вимогам до якості, безпеки та зручності використання.

3.3.4 Тестування програмного забезпечення

У ході розробки інформаційної системи для реабілітаційного центру було проведено комплексне тестування, що включало як автоматизовані, так і ручні перевірки. Для створення юніт-тестів було використано JUnit 5 [7] у поєднанні з Mockito [11], завдяки чому вдалося протестувати роботу окремих компонентів і гарантувати коректність їх роботи під різними вхідними параметрами. Після цього ще здійснювалося мануальне тестування всіх функціональних можливостей системи – від входу в особистий кабінет до управління календарем подій, включаючи створення засідань для МДРК. Результати обох підходів підтвердили відсутність критичних помилок та стабільну роботу всіх модулів у різних сценаріях використання.

3.3.5 Керівництво (інструкція) користувача

Для працівників реабілітаційного центру було розроблено детальну інструкцію користувача, в якій описано всі функціональні можливості інформаційної управляючої системи: від автентифікації та входу в особистий кабінет до перегляду усіх створених подій в календарі. Окремий розділ присвячений покроковому створенню та редагуванню нових подій у календарі, що є актуальним для працівників у складі МДРК, де докладно пояснюється послідовність дій у інтерфейсі, а також механізм роботи інтелектуального

модуля пошуку вільних часових слотів на основі правил Drools та перевірок доступності учасників.

3.4. Технічне забезпечення

3.4.1 Обґрунтування вибору технічного забезпечення

Для інформаційної системи реабілітаційного центру було обрано архітектуру, що передбачає використання віддалених серверів для розміщення веб-додатку та бази даних. Користувачі — адміністратор, працівники та клієнти — отримують доступ до системи через веб-браузер на комп'ютері або ноутбуці з підключенням до Інтернету.

Взаємодія між клієнтськими пристроями та серверами здійснюється через мережу Інтернет з використанням протоколу HTTP чи HTTPS, яке забезпечує захищене шифрування даних під час передачі. Для взаємодії самого додатку та бази даних використовується мережевий протокол TCP/IP. На рисунку 3.4 представлена загальна схема автоматизації.

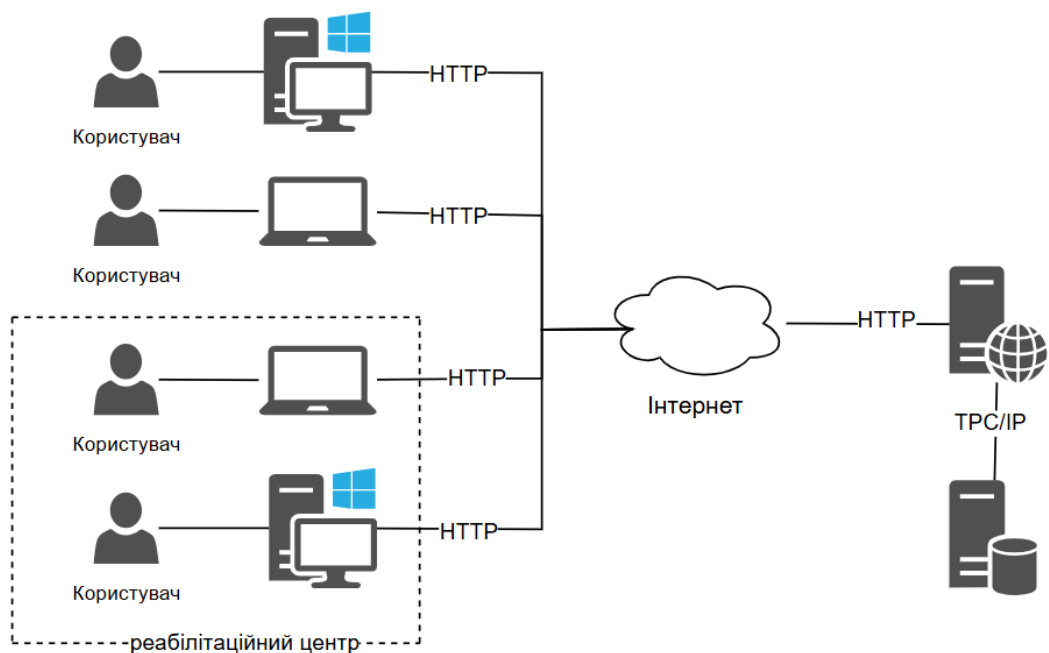


Рис. 3.4 Загальна схема автоматизації
Джерело: розроблено автором

Загальна схема автоматизації складається з наступних компонент:

1. Клієнтський пристрій: комп'ютер або ноутбук користувача з встановленим веб-браузером (наприклад, Google Chrome). Користувач вводить адресу сайту, після чого браузер надсилає HTTP/HTTPS-запит до сервера додатку.
2. Інтернет: глобальна мережа, через яку здійснюється передача даних між клієнтським пристроєм та серверами.
3. Сервер додатку: віддалений сервер, на якому розміщено веб-додаток, розроблений з використанням Java та фреймворку Spring Boot. Сервер обробляє вхідні запити, виконує бізнес-логіку та формує відповіді для клієнта. Для безпеки аутентифікації користувачів застосовується Spring Security з шифруванням паролів за допомогою алгоритму BCrypt.
4. Сервер бази даних: окремий віддалений сервер, на якому розміщено реляційну базу даних MySQL. Сервер додатку взаємодіє з базою даних через JDBC, використовуючи об'єктно-реляційне відображення (ORM) за допомогою Spring Data JPA. Це забезпечує ефективне зберігання та отримання даних, необхідних для функціонування системи реабілітаційного центру.

Така архітектура дозволяє забезпечити масштабованість, надійність та безпеку системи, а також спрощує її обслуговування та подальший розвиток.

3.4.2 Ресурсні вимоги до технічного забезпечення

Інформаційна система для реабілітаційного центру передбачає наявність двох віддалених серверів, а саме сервер на якому буде запущено додаток для реабілітаційного центру та сервер для бази даних. Доступ користувачів до ІС буде здійснюватись через мережу Інтернет з браузера ноутбука чи комп'ютера користувача.

Користувачі системи — працівники реабілітаційного центру та клієнти — отримують доступ до ІС через веб-браузер на власному персональному комп'ютері або ноутбукі. Для забезпечення комфортної роботи з веб-інтерфейсом рекомендується використовувати пристрої з процесором Intel Core i5 або AMD Ryzen 5 та оперативною пам'яттю обсягом 8 ГБ. Також для кращої сумісності та безпеки рекомендується використовувати останні версії операційних систем та актуальної версії веб браузер, наприклад, Google Chrome, Mozilla Firefox або Microsoft Edge. Не менш важливе значення має швидке та стабільне підключення до Інтернету, що забезпечує швидке завантаження веб-сторінок.

Сервер додатку, реалізований на платформі Java з використанням фреймворку Spring Boot, потребує апаратного забезпечення з процесором не нижче Intel Xeon або AMD Ryzen 7, оперативною пам'яттю обсягом 8 ГБ та твердотільним накопичувачем (SSD) ємністю не менше 256 ГБ. Сервер бази даних MySQL має аналогічні вимоги до процесора та оперативної пам'яті, а також потребує SSD ємністю від 256 ГБ для забезпечення швидкого доступу до даних та їх надійного зберігання.

Таким чином, визначені вимоги до технічного забезпечення ІУС забезпечують її ефективну та безпечну роботу, враховуючи потреби користувачів та можливості інфраструктури реабілітаційного центру.

3.5. Реалізація інформаційної управляючої системи

В результаті виконання даного кваліфікаційного магістерського проєкту була розроблена інформаційна система для реабілітаційного центру. На головній сторінці (див. рис. 3.5-3.7) якої представлена загальна інформація про реабілітаційний центр.

Ваш шлях до відновлення починається тут

Відкрийте для себе силу та життєву енергію в нашому сучасному реабілітаційному центрі, де інновації поєднуються з турботою про ваше здоров'я.

1000+Задоволених
клієнтів**15**

Років досвіду

24/7

Підтримка

[Почати відновлення](#)

Рис. 3.5 Головна сторінка

Джерело: розроблено автором

"Відновлюємо здоров'я, повертаємо радість життя"

1

Сучасне обладнання

Використовуємо передові технології для
вашого відновлення.

2

Професійна команда

Досвідчені фахівці з індивідуальним
підходом.

3

Комфортні умови

Створюємо затишну атмосферу для
відновлення.

Рис. 3.6 Головна сторінка. Продовження

Джерело: розроблено автором

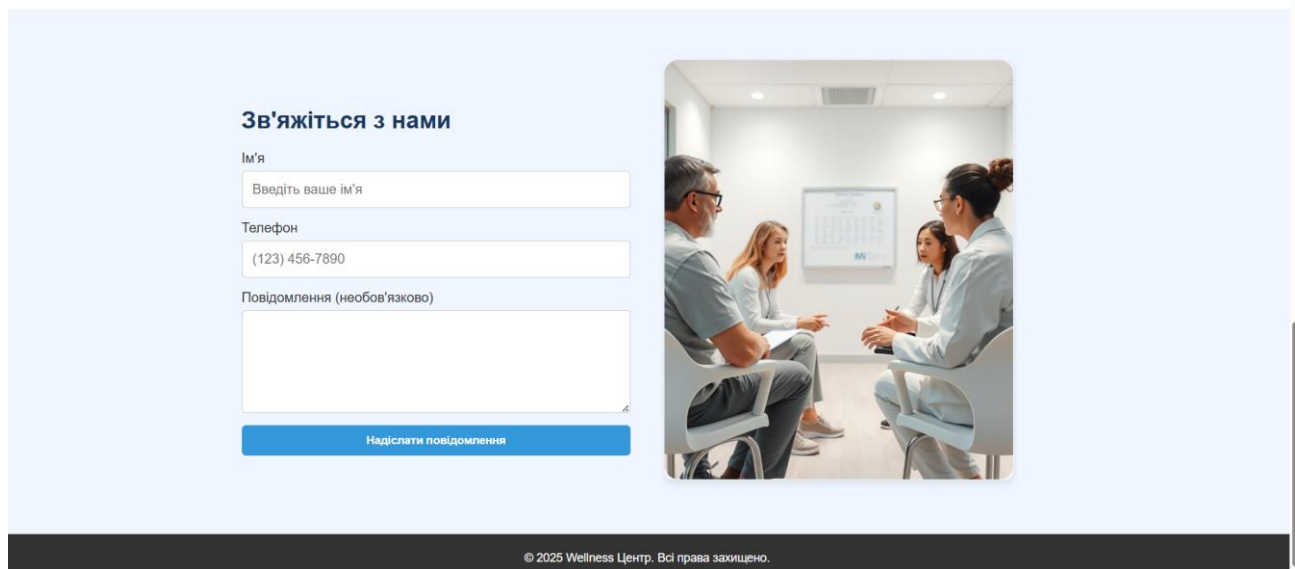


Рис. 3.7 Головна сторінка. Продовження

Джерело: розроблено автором

На головній сторінці є форма в якій можна вказати ім'я, телефон та якщо потрібно написати додатково текст для того, щоб адміністратор зателефонував користувачам, наприклад у разі виникнення питань. Зверху головної сторінки є меню, де розміщені посилання, одне з яких — на сторінку з послугами (див. рис. 3.8 – 3.9).

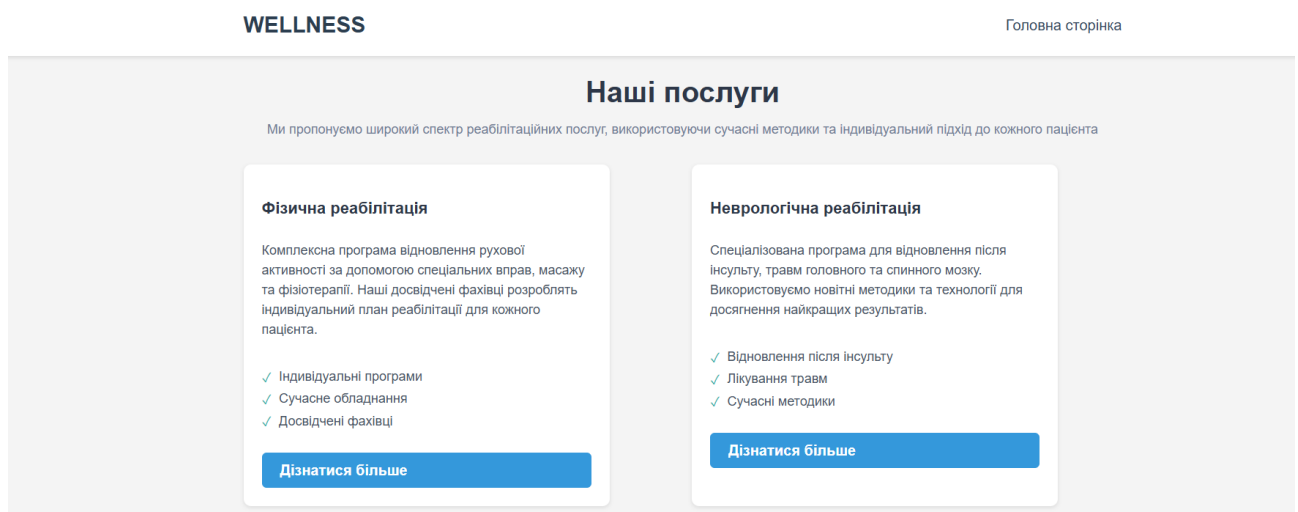


Рис. 3.8 Сторінка з послугами

Джерело: розроблено автором

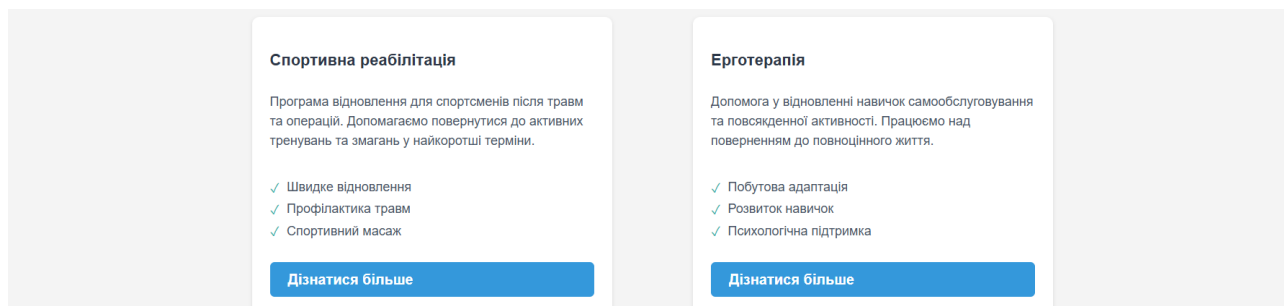


Рис. 3.9 Сторінка з послугами. Продовження

Джерело: розроблено автором

Також зверху головної сторінки розміщено посилання на сторінку з контактною інформацією даного реабілітаційного центру (рис. 3.10).

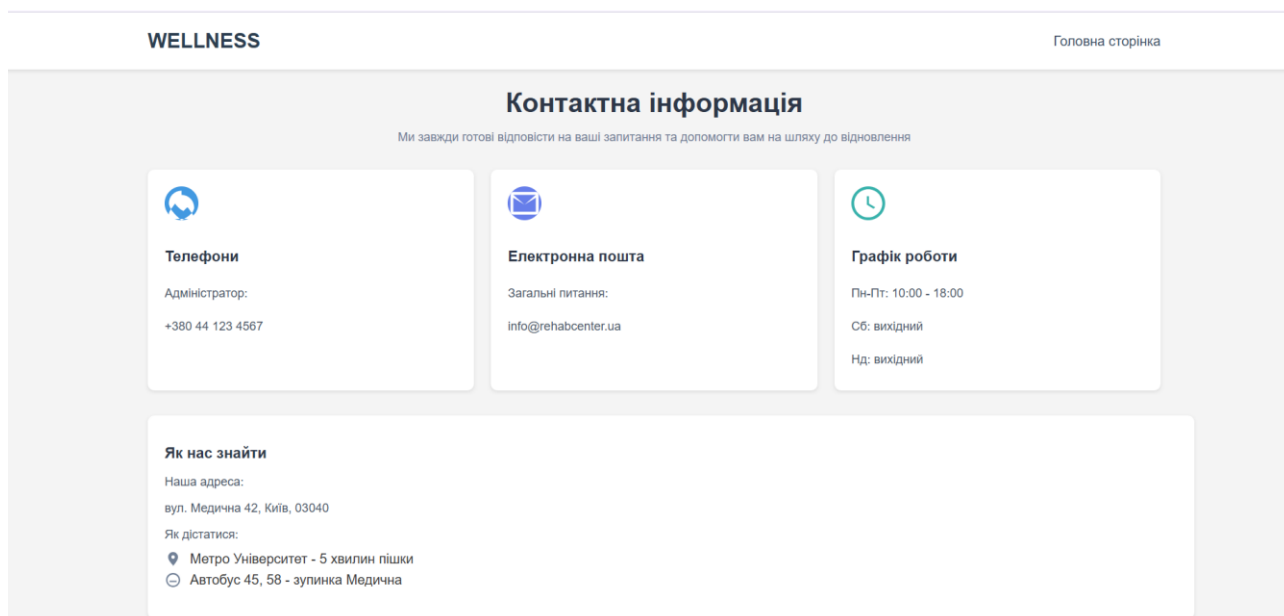


Рис. 3.10 Сторінка з контактною інформацією

Джерело: розроблено автором

Окрім цього на головній сторінці є посилання для входу в особистий кабінет, але для цього потрібно пройти аутентифікацію (рис. 3.11 – 3.12).

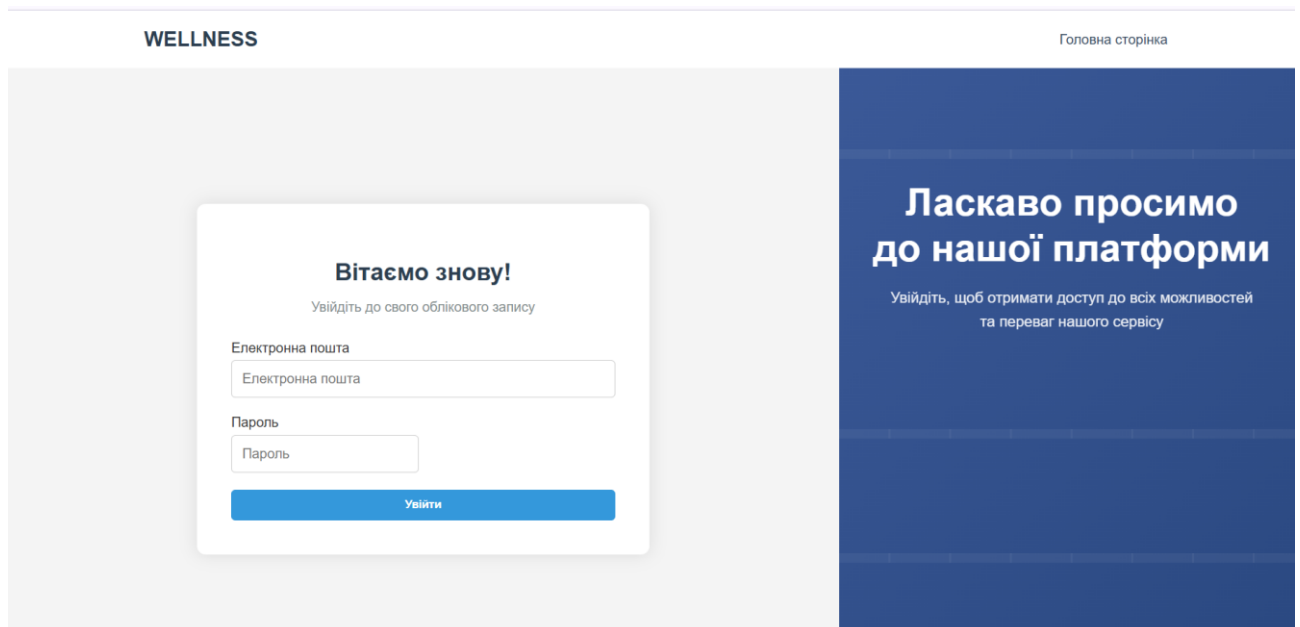


Рис. 3.11 Сторінка аутентифікації

Джерело: розроблено автором

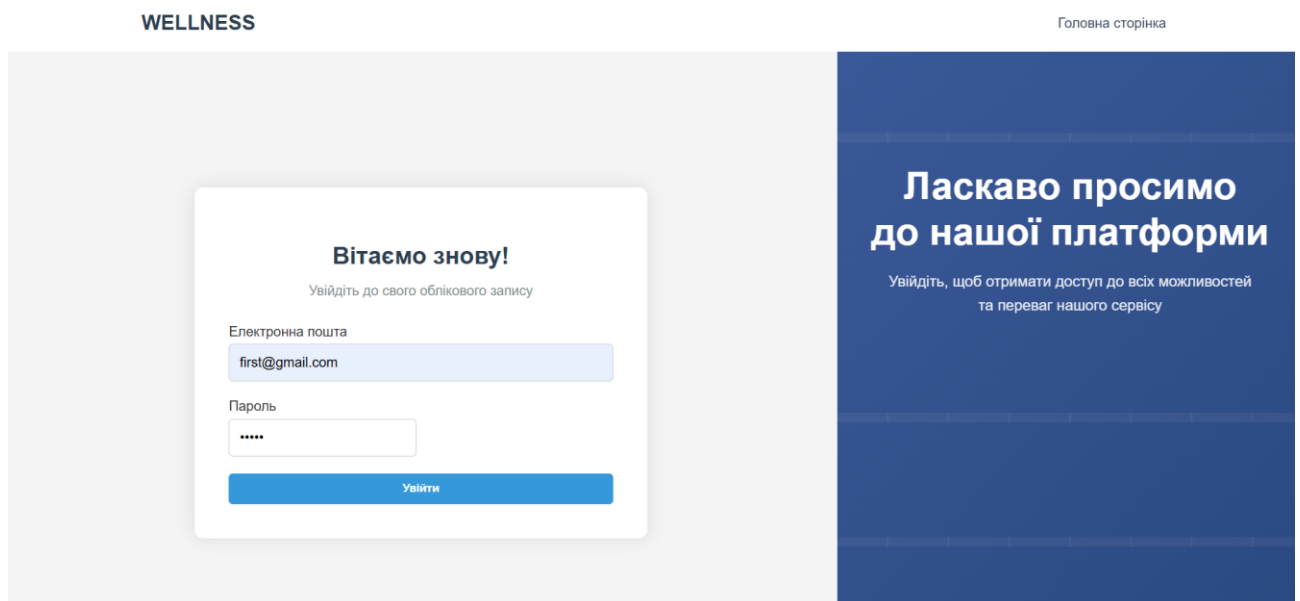


Рис. 3.12 Заповнена сторінка аутентифікації

Джерело: розроблено автором

Після успішної аутентифікації користувача перенаправляє в особистий кабінет, де представлений календар подій створених зустрічей, засідань, консультацій чи нарад (рис. 3.13).

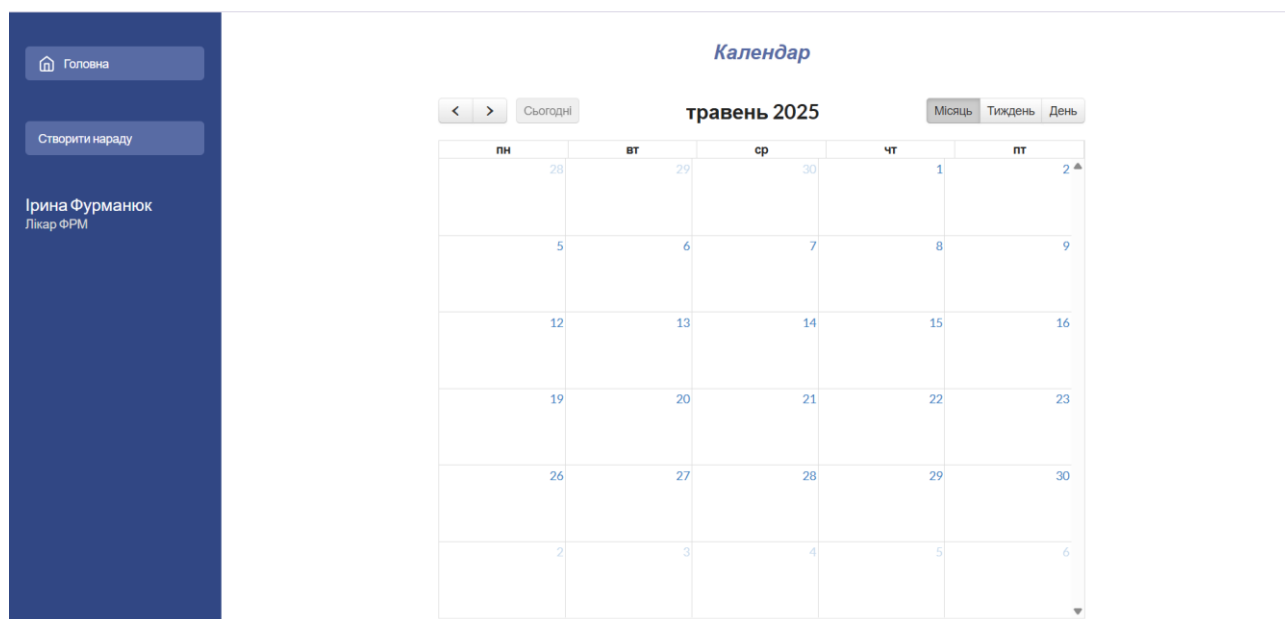
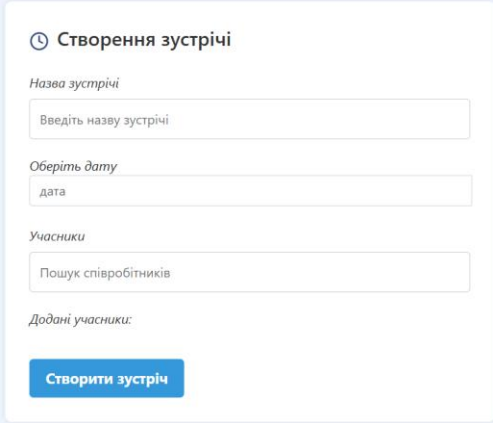


Рис. 3.13 Сторінка особистого кабінету

Джерело: розроблено автором

З особистого кабінету ми можемо перейти на сторінку створення нової зустрічі/наради. Де ввівши всі необхідні дані та обравши працівників для цієї події можна буде її створити (рис. 3.14 -3.17).



🕒 Створення зустрічі

Назва зустрічі

Введіть назву зустрічі

Оберіть дату

дата

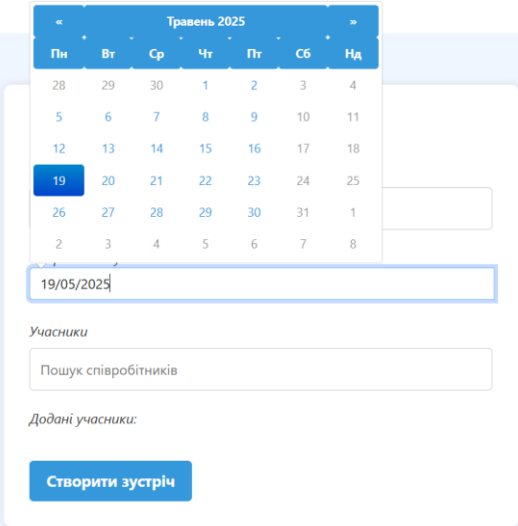
Учасники

Пошук співробітників

Додані учасники:

Створити зустріч

Рис. 3.14 Сторінка для створення події

Джерело: розроблено автором

« Травень 2025 »

Пн	Вт	Ср	Чт	Пт	Сб	Нд
28	29	30	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	1
2	3	4	5	6	7	8

19/05/2025

Учасники

Пошук співробітників

Додані учасники:

Створити зустріч

Рис. 3.15 Вигляд календаря для обрання дати

Джерело: розроблено автором

🕒 Створення зустрічі

Назва зустрічі

Засідання МДРК №2371

Оберіть дату

19/05/2025

Учасники

Ір

Ірина Фурманюк
Лікар ФРМ

Додати

Створити зустріч

Рис. 3.16 Пошук працівників для створення події

Джерело: розроблено автором

🕒 Створення зустрічі

Назва зустрічі

Засідання МДРК №2371

Оберіть дату

19/05/2025

Учасники

Пошук співробітників

Додані учасники:

Ірина Фурманюк x Олена Ясинська x

Створити зустріч

Рис. 3.17 Заповнена форма для створення події

Джерело: розроблено автором

Після створення події у календарі дані про створену подію відображаються на наступній сторінці (рис. 3.18), де в разі необхідності є можливість відредагувати запис (рис. 3.19).

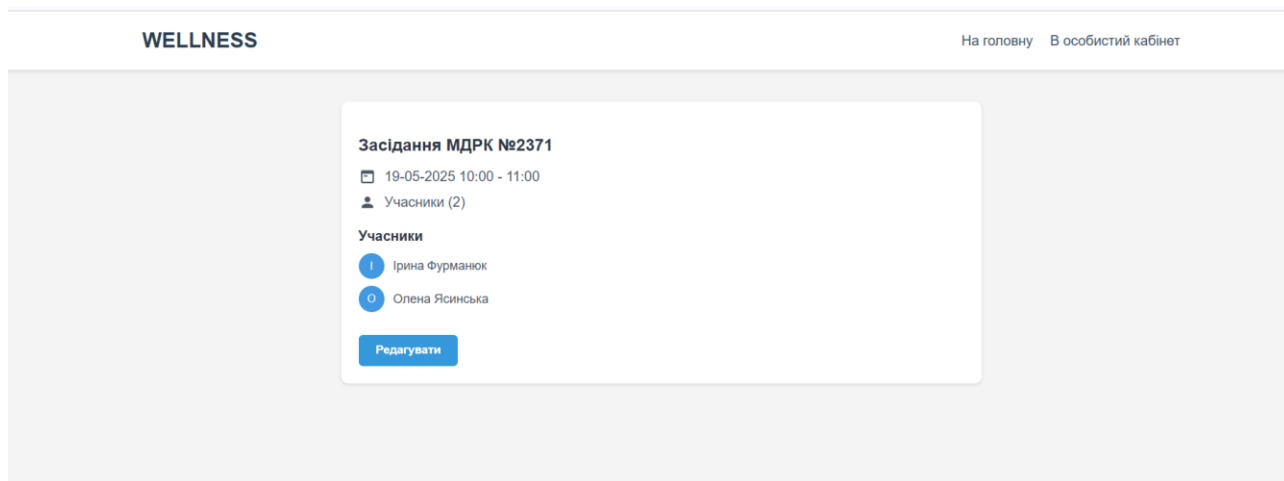


Рис. 3.18 Сторінка з даними створеної події

Джерело: розроблено автором

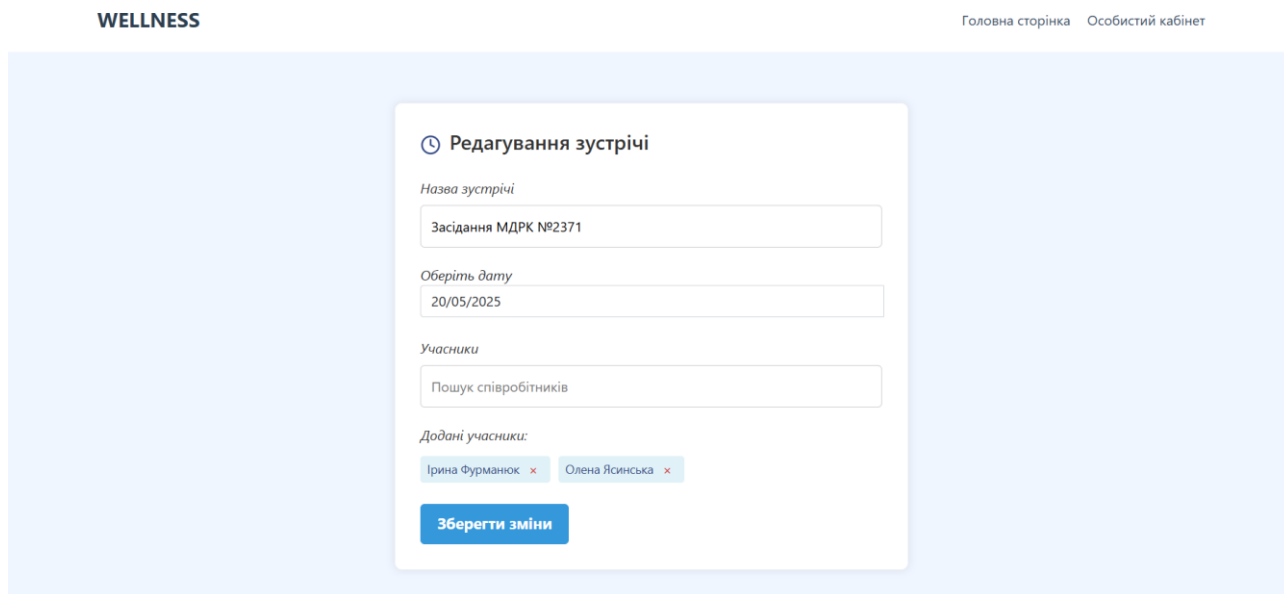


Рис. 3.19 Сторінка для редагування наявної події

Джерело: розроблено автором

Усі зміни щодо подій одразу відображаються в календарі на сторінці особистого кабінету користувача (рис. 3.20).

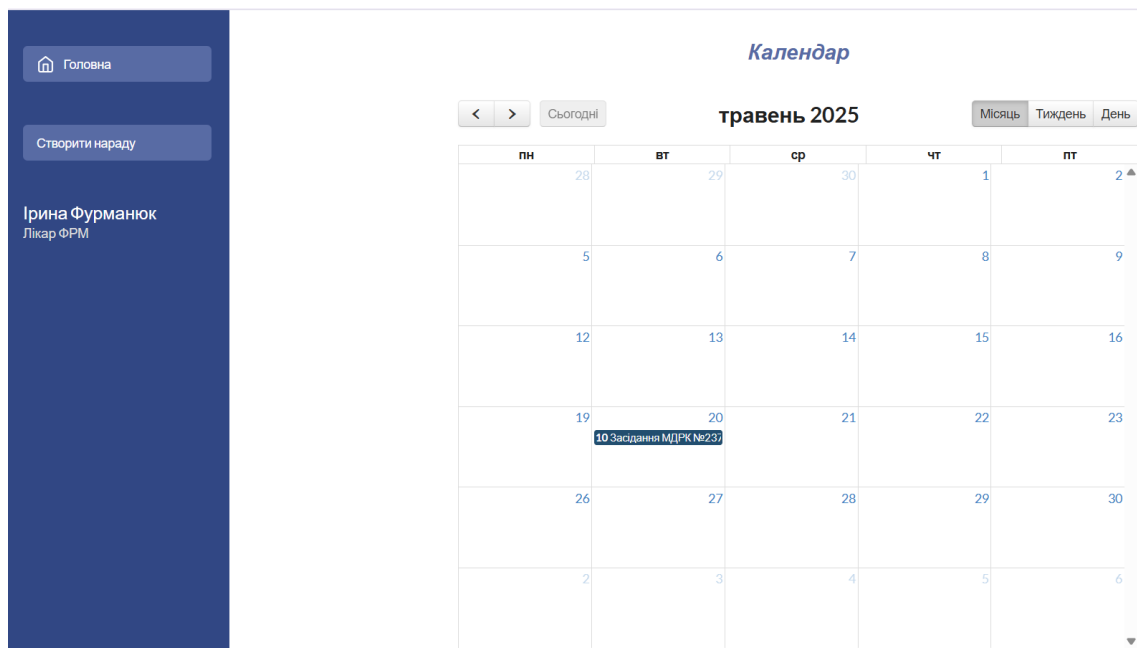


Рис. 3.20 Сторінка з оновленим журналом подій

Джерело: розроблено автором

ВИСНОВКИ

В результаті виконання даного магістерського проєкту було досліджено роботу реабілітаційного центру, та особливу увагу приділено роботі МДРК та процесу планування її засідань, також було проаналізовано уже існуючі інформаційні системи та наведено їх порівняльну таблицю.

На основі отриманих даних було визначено функціональні та нефункціональні вимоги до системи, на основі чого було змодельовано поведінку системи, що представлено у вигляді діаграм прецедентів та послідовностей, створено прототип інформаційної системи та звичайно ж реалізовано її.

Відповідно до усіх визначених вимог для реалізації системи були обрані сучасні інструменти та технології, а саме, бізнес логіка реалізована на базі Java і Spring, UI частина IC створена за допомогою Thymeleaf, HTML, CSS та JavaScript. Вибір монолітної архітектури забезпечив швидке й надійне розгортання рішення на віддалених серверах, а застосування Spring Security із шифруванням паролів за допомогою BCrypt гарантує захищеність персональних даних. Інтелектуальні можливості системи, зокрема пошук вільних часових слотів для засідань МДРК, реалізовані за допомогою Drools і гнучко налаштованих бізнес-правил, що дозволяє оперативно адаптувати механізм планування до змін у робочих графіках працівників.

Під час проєктування було проведено контрольний приклад із завантаженням даних у MySQL та обраховано прогнозне зростання обсягів бази даних. Такі засоби та інструменти розробки, як Lombok, MapStruct, IntelliJ IDEA та MySQL Workbench забезпечили високу продуктивність роботи і зменшили час на написання шаблонного коду. Моделювання поведінки системи виконувалося в Enterprise Architect, Draw.io і Lucidchart, що дозволило візуалізувати роботу інформаційної системи.

Під час тестування створено набір юніт-тестів із JUnit 5 і Mockito, що гарантують коректність внутрішньої логіки, а також проведено повноцінне мануальне тестування веб-інтерфейсу та функціональних сценаріїв. Загалом створена ІУС відповідає вимогам надійності, безпеки, адаптивності та простоти в підтримці, що підтверджує високу готовність рішення до впровадження в експлуатацію та подальшого розвитку.

Для забезпечення високої доступності та масштабованості рекомендується розгорнути систему в хмарному середовищі з використанням контейнеризації та оркестрації (Docker, Kubernetes), що дозволить гнучко керувати ресурсами й легко інтегрувати додаткові сервіси. Такий підхід гарантує адаптивність системи під зростаючі потреби реабілітаційного центру та забезпечить кращий досвід як для працівників, так і для клієнтів.

Розроблена інформаційна управляюча система є надзвичайно актуальною в сучасних умовах цифровізації медичної галузі та має істотне практичне значення, адже дозволяє оптимізувати координацію мультидисциплінарних команд, підвищити точність планування та зменшити ризики оперативних помилок, а її впровадження сприяє підвищенню якості реабілітаційних послуг та ефективності роботи закладу.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. About TigerConnect <https://sourceforge.net/software/product/TigerConnect/>
2. Bootstrap [Електронний ресурс] — Режим доступу до ресурсу: <https://getbootstrap.com/>
3. Draw.io [Електронний ресурс] — Режим доступу до ресурсу: <https://app.diagrams.net/>
4. Drools [Електронний ресурс] — Режим доступу до ресурсу: <https://www.drools.org/>
5. Enterprise Architect [Електронний ресурс] — Режим доступу до ресурсу: <https://sparxsystems.com/>
6. How to Set BCrypt Password Encoding in Spring Boot Security? [Електронний ресурс] — Режим доступу до ресурсу: <https://www.aegissofttech.com/articles/bcrypt-password-encoding-spring-security.html>
7. JUnit 5 [Електронний ресурс] — Режим доступу до ресурсу: <https://junit.org/junit5/>
8. Lombok [Електронний ресурс] — Режим доступу до ресурсу: <https://projectlombok.org/>
9. MapStruct [Електронний ресурс] — Режим доступу до ресурсу: <https://mapstruct.org/>
10. Microservices vs. monolithic architecture [Електронний ресурс] — Режим доступу до ресурсу: <https://www.atlassian.com/microservices/microservices-architecture/microservices-vs-monolith>
11. Mockito [Електронний ресурс] — Режим доступу до ресурсу: <https://site.mockito.org/>
12. MVC [Електронний ресурс] — Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/%D0%9C%D0%BE%D0%B4%D0%B5%D0%BB%D1%8C-%D0%B2%D0%B8%D0%B4-%D0%BA%D0%BE%D0%BD%D1%82%D1%80%D0%BE%D0%BB%D0%B>

- 13.MySQL [Електронний ресурс] — Режим доступу до ресурсу:
<https://www.mysql.com/>
- 14.MySQL Enterprise Edition: ключ до безпеки, продуктивності та інновацій у керуванні базами даних [Електронний ресурс] — Режим доступу до ресурсу:
<https://erc.ua/erc-reviews/24443/mysql-enterprise-edition-kliuch-do-bezpeki-produktivnosti-ta-innovatsii-u-keruvanni-bazami-danikh>
- 15.MySQL Workbench [Електронний ресурс] — Режим доступу до ресурсу:
<https://www.mysql.com/products/workbench/>
- 16.Rehab Center Management Software: Entrepreneur's Guide: Choosing the Right Rehab Management Software <https://fastercapital.com/content/Rehab-Center-Management-Software--Entrepreneur-s-Guide--Choosing-the-Right-Rehab-Management-Software.html>
- 17.Software Roundtable: Scheduling and Appointment Features [Електронний ресурс] — Режим доступу до ресурсу: <https://rehabpub.com/clinic-management/software/software-roundtable/>
- 18.Spring Boot [Електронний ресурс] — Режим доступу до ресурсу:
<https://spring.io/projects/spring-boot>
- 19.Spring Data [Електронний ресурс] — Режим доступу до ресурсу:
<https://spring.io/projects/spring-data>
- 20.Spring Security [Електронний ресурс] — Режим доступу до ресурсу:
<https://spring.io/projects/spring-security>
- 21.The Challenge of Reorganizing Rehabilitation Services at the Time of COVID-19 Pandemic: A New Digital and Artificial Intelligence Platform to Support Team Work in Planning and Delivering Safe and High Quality Care [Електронний ресурс] — Режим доступу до ресурсу:
<https://www.frontiersin.org/journals/neurology/articles/10.3389/fneur.2021.643251/full>
- 22.Thymeleaf [Електронний ресурс] — Режим доступу до ресурсу:
<https://www.thymeleaf.org/>

23. TigerConnect [Електронний ресурс] — Режим доступу до ресурсу:
<https://tigerconnect.com/>
24. Актуальна інформація з реабілітації. [Електронний ресурс] — Режим доступу до ресурсу: <https://moz.gov.ua/uk/aktualna-informacija-z-reabilitacii>
25. Брич В.В. «Можливості реалізації медичної реабілітації в умовах реформування та нової системи фінансування закладів охорони здоров'я України».
26. Діаграма послідовності [Електронний ресурс] — Режим доступу до ресурсу:
https://uk.wikipedia.org/wiki/%D0%94%D1%96%D0%B0%D0%B3%D1%80%D0%B0%D0%BC%D0%B0_%D0%BF%D0%BE%D1%81%D0%BB%D1%96%D0%B4%D0%BE%D0%B2%D0%BD%D0%BE%D1%81%D1%82%D1%96
27. Діаграма прецедентів [Електронний ресурс] — Режим доступу до ресурсу:
https://uk.wikipedia.org/wiki/%D0%94%D1%96%D0%B0%D0%B3%D1%80%D0%B0%D0%BC%D0%B0_%D0%BF%D1%80%D0%B5%D1%86%D0%B5%D0%B4%D0%B5%D0%BD%D1%82%D1%96%D0%B2
28. Заварова Н.В. Організація діяльності закладів соціальної реабілітації системи освіти України / Н.В. Заварова // Вісник Луганського національного університету ім. Т. Шевченка. Педагогічні науки. - 2012. - № 19, ч. 2. - С. 75-81.
29. Закон України «Про реабілітацію у сфері охорони здоров'я».
30. Крупа В.В. Формування професійної компетентності майбутніх фахівців з фізичної реабілітації: автореф. дис. на здобуття наук. ступеня канд. пед. наук: спец. 13.00.04 «Теорія і методика професійної освіти»/ В.В. Крупа. - Тернопіль, 2014. - 20 с.
31. М. Пустовіт, К. Сергієнко, Д. Штанагей Використання сучасних інновацій та інформаційних технологій в реабілітації / / Інноваційні та інформаційні технології у фізичній культурі, спорті, фізичній терапії та ерготерапії: матеріали І Всеукр. електрон. конф-ії, 19 квітня 2018 р. - Київ, 2018. - С. 72-74.
32. Медичні інформаційні системи – стан розвитку в Україні / В. О. Качмар //

- Український журнал телемедицини та медичної телематики. - 2010. - Т. 8, № 1. - С. 12-17. - Режим доступу: http://nbuv.gov.ua/UJRN/Ujtm_2010_8_1_4
33. Моделі даних. Реляційна модель даних [Електронний ресурс] — Режим доступу до ресурсу: https://elib.lntu.edu.ua/sites/default/files/elib_upload/BD_2016_3/page5.html
34. Нефункціональні вимоги [Електронний ресурс] — Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/%D0%9D%D0%B5%D1%84%D1%83%D0%BD%D0%BA%D1%86%D1%96%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D1%96_%D0%B2%D0%B8%D0%BC%D0%BE%D0%B3%D0%B8
35. Особливості організації діяльності реабілітаційних закладів в Україні / О. В. Бісмак // Освітологічний дискурс. - 2015. - № 4. - С. 1-12. - Режим доступу: http://nbuv.gov.ua/UJRN/osdys_2015_4_3
36. Панченко О. А. «Інформаційні технології в реабілітології». ДЗ «Науково-практичний медичний реабілітаційно-діагностичний центр МОЗ України» , DOI: <https://doi.org/10.11603/mie.1996-1960.2013.1.411>
37. Погонцева О.В. Формування готовності майбутніх фахівців з фізичної реабілітації до професійної діяльності в оздоровчих центрах: автореф. дис. на здобуття наук. ступеня канд. пед. наук: спец. 13.00.04 «Теорія і методика професійної освіти» / О.В. Погонцева. - К., 2011. - 22 с.
38. Постанова Кабінету Міністрів України від 25 квітня 2018 р. № 411 «Деякі питання електронної системи охорони здоров'я».
39. Постанова Кабінету Міністрів України від 3 листопада 2021 р. № 1268 «Питання організації реабілітації у сфері охорони здоров'я».
40. Прийняття рішень умовах ризику [Електронний ресурс] — Режим доступу до ресурсу: https://elib.lntu.edu.ua/sites/default/files/elib_upload/%D0%95%D0%9D%D0%9F_%D0%AF%D0%BA%D0%B8%D0%BC%D1%87%D1%83%D0%BA_%D0%A1%D0%B5%D0%BB%D0%B5%D0%BF%D0%B8%D0%BD%D0%B0/pa

ge29.html

41. Принципи формування єдиної медичної інформаційної системи великого міста / А. І. Хвищун, В. О. Качмар // Медична інформатика та інженерія. - 2009. - № 3. - С. 39-47. - Режим доступу: http://nbuv.gov.ua/UJRN/Mii_2009_3_7
42. Приступа Є.Н. Роль і місце фахівця з фізичної реабілітації в системі охорони здоров'я / Є.Н. Приступа, А.С. Вовканич // Педагогіка, психологія та медико-біологічні проблеми фізичного виховання і спорту. – 2011. - №11. – С. 92-96.
43. Реабілітаційний центр як бізнес-проект. [Електронний ресурс] — Режим доступу до ресурсу: <https://drukarnia.com.ua/articles/reabilitaciiiii-centr-yak-biznes-proekt-TofcX>
44. Реабілітація: права та гарантії, шляхи реалізації – 2025. [Електронний ресурс] — Режим доступу до ресурсу: <https://naiu.org.ua/archives/edition/reabilitacziya-prava-ta-garantiyi-shlyahy-realizacziyi-2025>
45. Саранча І. Г. Роль центрів соціальної реабілітації у процесі соціалізації осіб з вадами опорно-рухового апарату / І. Г. Саранча // Вісн. Кам'янець-Поділ. нац. ун-ту ім. І. Огієнка. Корекц. педагогіка і психологія. - 2011. - Вип. 3. - С. 167-172
46. Система управління базами даних MySQL [Електронний ресурс] — Режим доступу до ресурсу: <https://lemon.school/blog/systema-upravlinnya-bazamy-danyh-mysql>
47. Створення масштабованих хмарних програм за допомогою MySQL [Електронний ресурс] — Режим доступу до ресурсу: <https://proit.ua/stvoriennia-masshtabovanikh-khmarnikh-proghram-za-dopomoghoiu-mysql/>
48. СУБД [Електронний ресурс] — Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D0%BA%D0%B5%D1%80%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F_%D0%B1%D0%B0%D0%B7%D0%B0%D0%

BC%D0%B8_%D0%B4%D0%B0%D0%BD%D0%B8%D1%85

49. Функціональні вимоги [Електронний ресурс] — Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/%D0%A4%D1%83%D0%BD%D0%BA%D1%86%D1%96%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%BD%D1%96_%D0%B2%D0%B8%D0%BC%D0%BE%D0%B3%D0%B8

50. Що таке мультидисциплінарна реабілітаційна команда. [Електронний ресурс] — Режим доступу до ресурсу: www.umj.com.ua/uk/novyna-245090-shho-take-multidistsiplinarna-reabilitatsijna-komanda

51. Що таке реабілітаційні послуги (у форматі легкого читання). [Електронний ресурс] — Режим доступу до ресурсу: <https://www.google.com/search?q=https://dopomoha-info.org.ua/u-formati-lehkoho-chytannia/shho-take-reabilitaczijni-poslugy-u-formati-legkogo-chitannya/>

52. Що таке Реляційні бази даних: Основи, переваги та використання [Електронний ресурс] — Режим доступу до ресурсу: <https://optimize-il.com/shho-take-relyatsijni-bazi-danih/>

ДОДАТКИ

Додаток А. Заповнена контрольним прикладом база даних

	role_id	name
▶	1	USER
	2	ADMIN
•	NULL	NULL

Рис. А.1. Зміст таблицки «role»

Джерело: розроблено автором

	user_id	role_id
▶	2	1
	3	1
	4	1
	5	1
	6	1
	7	1
	8	1
	9	1
	10	1
	1	2
•	NULL	NULL

Рис. А.2. Зміст таблицки «user_role»

Джерело: розроблено автором

	user_id	email	first_name	last_name	password	phone	title
▶	1	first@gmail.com	Ірина	Фурманюк	\$2a\$10\$8PIDWLM3LW...	+380123456789	Лікар ФРМ
	2	second@gmail.com	Олена	Ясинська	\$10\$8PIDWLM3LW...	+380987456183	Реабілітолог
	3	test_email3@gmail.com	Олександр	Яворський	\$9\$8PIDWLM3LWb...	+380674952850	Лікар ФРМ
	4	test_email4@gmail.com	Віктор	Ратинський	\$4\$8PIDWLM3LWb...	+380986952765	Фізичний терапевт
	5	test_email5@gmail.com	Олег	Кучин	\$10\$8PIDWLM3LW...	+380674952850	Протезист-ортезист
	6	test_email6@gmail.com	Катерина	Яремчук	\$5\$8PIDWLM3LWb...	+380676542850	Ерготерапевт
	7	test_email7@gmail.com	Любомир	Костюк	\$10\$8PIDWLM3LW...	+380674936455	Реабілітолог
	8	test_email8@gmail.com	Дмитро	Середюк	\$8\$8PIDWLM3LWb...	+380674952850	Реабілітолог
	9	test_email9@gmail.com	Ігор	Грозовий	\$10\$8PIDWLM3LW...	+380674756543	Ерготерапевт
	10	test_email10@gmail.com	Вікторія	Безгін	\$7\$8PIDWLM3LWb...	+380674952850	Психотерапевт
•	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Рис. А.3. Зміст таблицки «user»

Джерело: розроблено автором

	meeting_id	name	start	end
	1	Засідання МДРК №2371	2025-05-20 10:00:00.000000	2025-05-20 11:00:00.000000
	2	Засідання МДРК №2372	2025-05-21 10:00:00.000000	2025-05-21 11:00:00.000000
	3	Засідання МДРК №2373	2025-05-23 10:00:00.000000	2025-05-23 11:00:00.000000
	4	Засідання МДРК №2374	2025-05-27 10:00:00.000000	2025-05-27 11:00:00.000000
	5	Засідання МДРК №2375	2025-05-28 10:00:00.000000	2025-05-28 11:00:00.000000
	6	Засідання МДРК №2376	2025-05-29 10:00:00.000000	2025-05-29 11:00:00.000000
	7	Засідання МДРК №2377	2025-05-20 11:00:00.000000	2025-05-20 12:00:00.000000
▶*	NULL	NULL	NULL	NULL

Рис. А.4. Зміст таблицьки «meeting»

Джерело: розроблено автором

	user_meeting_id	user_user_id	meeting_meeting_id
▶	1	1	1
	2	4	1
	3	7	1
	4	2	2
	5	7	2
	6	9	2
	7	3	3
	8	9	3
	9	5	3
	10	1	4
	11	6	4
	12	2	4
	13	3	5
	14	4	5
	15	5	5
	16	8	6
	17	9	6
	18	6	6
	19	4	7
	20	1	7
	21	9	7
*	NULL	NULL	NULL

Рис. А.5. Зміст таблицьки «user_meeting»

Джерело: розроблено автором

	consultation_request_id	description	name	phone	status
	1	Питання 1 ...	Оксана	+380674539674	TODO
	2	NULL	Роман	+380675836196	TODO
	3	NULL	Юрій	+380964537188	DONE
▶	4	Питання 2 ...	Любомир	+380983465187	TODO
	5	NULL	Ростислав	+380984528199	TODO
•	NULL	NULL	NULL	NULL	NULL

Рис. А.6. Зміст таблицки «consultation_request»

Джерело: розроблено автором

	vacation_id	end_date	start_date	user_user_id
▶	1	2025-05-20	2025-05-19	1
	2	2025-05-20	2025-05-20	3
	3	2025-05-22	2025-05-22	5
	4	2025-05-20	2025-05-19	6
	5	2025-05-24	2025-05-23	8
•	NULL	NULL	NULL	NULL

Рис. А.7. Зміст таблицки «vacation»

Джерело: розроблено автором

Додаток Б. Приклад реалізованого тексту програми

MainController.java

```
package com.rehabilitation.controller.v1;

import lombok.RequiredArgsConstructor;
import lombok.extern.slf4j.Slf4j;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.GetMapping;

@Slf4j
@Controller
@RequiredArgsConstructor
public class MainController {

    @GetMapping("/")
    public String openMainPage() {
        return "main";
    }

    @GetMapping("/error")
    public String errorPage() {
        return "error";
    }

    @GetMapping("/contacts")
    public String contacts() {
        return "contacts";
    }

    @GetMapping("/services")
```

```

    public String services() {
        return "services";
    }

}

```

AuthController.java

```

package com.rehabilitation.controller.v1;

import com.rehabilitation.dto.AuthDto;
import com.rehabilitation.entity.User;
import com.rehabilitation.service.UserDetailsServiceImpl;
import lombok.RequiredArgsConstructor;
import lombok.extern.slf4j.Slf4j;
import org.springframework.security.core.annotation.AuthenticationPrincipal;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.GetMapping;

@Slf4j
@Controller
@RequiredArgsConstructor
public class AuthController {

    private final UserDetailsServiceImpl userDetailsService;

    @GetMapping("/login")
    public String openLoginPage(Model model) {
        model.addAttribute("authorization", new AuthDto());
    }
}

```

```

        return "login";
    }

    @GetMapping("/account")
    public String account(Model model, @AuthenticationPrincipal User user) {
        model.addAttribute("user", user);
        return "user-account";
    }
}

```

UserController.java

```

package com.rehabilitation.controller.v1;

import com.rehabilitation.entity.User;
import com.rehabilitation.service.UserDetailsServiceImpl;
import lombok.RequiredArgsConstructor;
import lombok.extern.slf4j.Slf4j;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.bind.annotation.ResponseBody;

import java.util.List;

@Slf4j
@Controller
@RequiredArgsConstructor
public class UserController {

```

```

private final UserDetailsServiceImpl userDetailsService;

@GetMapping("/users")
@ResponseBody
public List<User> getUsers(@RequestParam String search) {
    return userDetailsService.getUsers(search);
}
}

```

CalendarController.java

```

package com.rehabilitation.controller.v1;

import com.rehabilitation.dto.MeetingDto;
import com.rehabilitation.dto.ModifyMeetingDto;
import com.rehabilitation.entity.User;
import com.rehabilitation.service.CalendarService;
import lombok.RequiredArgsConstructor;
import lombok.extern.slf4j.Slf4j;
import org.springframework.security.core.annotation.AuthenticationPrincipal;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.validation.annotation.Validated;
import org.springframework.web.bind.annotation.*;

@Slf4j
@Validated
@Controller
@RequiredArgsConstructor
public class CalendarController {

```

```

private final CalendarService calendarService;

@ResponseBody
@GetMapping("/meetings")
public MeetingDto[] getMeetings(@AuthenticationPrincipal User user) {
    log.info("Getting meetings for {}", user.getUserId());
    return
calendarService.getMeetings(user.getUserId()).toArray(MeetingDto[]::new);
}

@GetMapping("/calendar/new-meeting")
public String newMeeting(Model model, @AuthenticationPrincipal User user) {

    model.addAttribute("user", user);
    model.addAttribute("meeting", new ModifyMeetingDto());

    return "create-meeting";
}

@PostMapping("/calendar/new-meeting")
public String createNewMeeting(Model model, @AuthenticationPrincipal User
user, @ModelAttribute ModifyMeetingDto meeting) {

    model.addAttribute("user", user);
    var meetingId = calendarService.createMeeting(meeting);

    model.addAttribute("meeting", calendarService.getMeetingById(meetingId));
    return "created-request-page";
}

```

```

    @GetMapping("/calendar/meeting/{meetingId}")
    public String newMeeting(Model model, @AuthenticationPrincipal User user,
    @PathVariable long meetingId) {

        model.addAttribute("user", user);
        model.addAttribute("meeting", new ModifyMeetingDto());

        return "edit-meeting";
    }

    @PutMapping("/calendar/meeting/{meetingId}")
    public String createNewMeeting(Model model, @AuthenticationPrincipal User
    user, @PathVariable long meetingId,
        @ModelAttribute ModifyMeetingDto meeting) {

        model.addAttribute("user", user);

        calendarService.updateMeeting(meetingId, meeting);

        model.addAttribute("meeting", calendarService.getMeetingById(meetingId));
        return "created-request-page";
    }
}

```

UserDetailsServiceImpl.java

```

package com.rehabilitation.service;

import com.rehabilitation.entity.User;
import com.rehabilitation.repository.UserRepository;

```

```

import lombok.RequiredArgsConstructor;
import lombok.extern.slf4j.Slf4j;
import org.springframework.security.core.userdetails.UserDetails;
import org.springframework.security.core.userdetails.UserDetailsService;
import org.springframework.security.core.userdetails.UsernameNotFoundException;
import org.springframework.stereotype.Service;

import java.util.List;

import static java.util.Optional.ofNullable;

@Slf4j
@Service
@RequiredArgsConstructor
public class UserDetailsServiceImpl implements UserDetailsService {
    private final UserRepository userRepository;

    @Override
    public UserDetails loadUserByUsername(String username) {

        return ofNullable(userRepository.findByEmail(username))
            .orElseThrow(() -> {
                log.warn("User with username {} not found!", username);
                return new UsernameNotFoundException("User with username " +
username + " not found!");
            });
    }

    public List<User> getUsers(String search) {
        return
userRepository.findByFirstNameStartingWithOrLastNameStartingWith(search);
    }
}

```

```
}  
}
```

CalendarService.java

```
package com.rehabilitation.service;  
  
import com.rehabilitation.dto.MeetingDto;  
import com.rehabilitation.dto.ModifyMeetingDto;  
  
import java.util.List;  
  
public interface CalendarService {  
  
    List<MeetingDto> getMeetings(long userId);  
  
    long createMeeting(ModifyMeetingDto meeting);  
  
    MeetingDto getMeetingById(long meetingId);  
  
    void updateMeeting(long meetingId, ModifyMeetingDto meeting);  
}
```

MeetingMapper.java

```
package com.rehabilitation.mapper;  
  
import com.rehabilitation.dto.MeetingDto;  
import com.rehabilitation.dto.ModifyMeetingDto;  
import com.rehabilitation.entity.Meeting;
```

```

import com.rehabilitation.entity.User;
import com.rehabilitation.entity.UserMeeting;
import org.mapstruct.Mapper;
import org.mapstruct.Mapping;
import org.mapstruct.MappingTarget;
import org.mapstruct.Named;

import java.util.List;

import static org.mapstruct.MappingConstants.ComponentModel.SPRING;

@Mapper(componentModel = SPRING)
public interface MeetingMapper {

    @Mapping(source = "name", target = "title")
    @Mapping(source = ".", target = "users", qualifiedByName = "toUsers")
    MeetingDto toMeetingDto(Meeting userMeeting);

    @Mapping(source = "userMeetingId", target = "id")
    @Mapping(source = "meeting.name", target = "title")
    @Mapping(source = "meeting.start", target = "start")
    @Mapping(source = "meeting.end", target = "end")
    MeetingDto toMeetingDto(UserMeeting userMeeting);

    List<MeetingDto> toMeetingDtos(List<UserMeeting> userMeetings);

    @Named("toUsers")
    default List<User> toUsers(Meeting meeting) {
        return meeting.getUserMeetings()
            .stream()

```

```

        .map(UserMeeting::getUser)
        .toList();
    }

    @Mapping(target = "id", ignore = true)
    @Mapping(target = "userMeetings", ignore = true)
    void updateMeeting(@MappingTarget Meeting existingMeeting,
        ModifyMeetingDto meeting);
}

```

UserRepository.java

```

package com.rehabilitation.repository;

import com.rehabilitation.entity.User;
import org.springframework.data.jpa.repository.EntityGraph;
import org.springframework.data.jpa.repository.JpaRepository;
import org.springframework.data.jpa.repository.Query;
import org.springframework.stereotype.Repository;

import java.util.List;

@Repository
public interface UserRepository extends JpaRepository<User, Long> {

    @EntityGraph(attributePaths = "roles")
    User findByEmail(String email);

    @Query(value = "SELECT * FROM User u WHERE u.first_name LIKE ?1% OR u.last_name LIKE ?1%", nativeQuery = true)

```

```
List<User> findByFirstNameStartingWithOrLastNameStartingWith(String search);  
}
```

UserMeetingRepository.java

```
package com.rehabilitation.repository;
```

```
import com.rehabilitation.entity.UserMeeting;  
import org.springframework.data.jpa.repository.JpaRepository;  
import org.springframework.stereotype.Repository;
```

```
import java.util.List;
```

```
@Repository
```

```
public interface UserMeetingRepository extends JpaRepository<UserMeeting, Long>  
{
```

```
    List<UserMeeting> findByUser_UserId(long userId);  
}
```

AvailableTimeSlotsDecisionMaker.java

```
package com.rehabilitation.service;
```

```
import java.util.List;
```

```
import com.rehabilitation.dto.TimeSlotDto;  
import com.rehabilitation.entity.Meeting;  
import com.rehabilitation.entity.Vacation;
```

```

public interface AvailableTimeSlotsDecisionMaker {

    List<TimeSlotDto> findAvailableSlots(List<TimeSlotDto> allSlots,
                                         List<Meeting> meetings,
                                         List<Vacation> vacations);

}

```

AvailableTimeSlotsDecisionMakerImpl.java

```

package com.rehabilitation.service;

import java.util.ArrayList;
import java.util.List;

import org.kie.api.runtime.KieContainer;
import org.kie.api.runtime.KieSession;
import org.kie.api.runtime.rule.FactHandle;
import org.springframework.stereotype.Component;

import com.rehabilitation.dto.TimeSlotDto;
import com.rehabilitation.entity.Meeting;
import com.rehabilitation.entity.Vacation;

import lombok.RequiredArgsConstructor;
import lombok.extern.slf4j.Slf4j;

@Slf4j
@Component
@RequiredArgsConstructor
public class AvailableTimeSlotsDecisionMakerImpl implements
AvailableTimeSlotsDecisionMaker {

```

```

private final KieContainer kieContainer;

@Override
public List<TimeSlotDto> findAvailableSlots(List<TimeSlotDto> allSlots,
                                           List<Meeting> meetings,
                                           List<Vacation> vacations) {
    log.info("Finding available slots...");

    KieSession kSession = kieContainer.newKieSession();

    allSlots.forEach(kSession::insert);
    meetings.forEach(kSession::insert);
    vacations.forEach(kSession::insert);

    kSession.fireAllRules();

    List<TimeSlotDto> availableSlots = new ArrayList<>();
    for (FactHandle handle : kSession.getFactHandles()) {
        Object fact = kSession.getObject(handle);
        if (fact instanceof TimeSlotDto) {
            availableSlots.add((TimeSlotDto) fact);
        }
    }

    kSession.dispose();

    return availableSlots;
}
}

```

DroolsConfig.java

```
package com.rehabilitation.config;

import java.util.List;

import org.kie.api.KieServices;
import org.kie.api.builder.KieBuilder;
import org.kie.api.builder.KieFileSystem;
import org.kie.api.builder.KieRepository;
import org.kie.api.runtime.KieContainer;
import org.kie.api.runtime.KieSession;
import org.kie.internal.io.ResourceFactory;
import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.Configuration;
import org.springframework.context.annotation.RequiredArgsConstructor;

import lombok.RequiredArgsConstructor;

@Configuration
@RequiredArgsConstructor
public class DroolsConfig {

    @Bean
    public KieServices kieServices() {
        return KieServices.Factory.get();
    }

    @Bean
    public KieFileSystem getKieFileSystem() {
```

```

        var kieFileSystem = kieServices().newKieFileSystem();
        var rules = List.of("available_slots.drl");
        rules.forEach(rule ->
kieFileSystem.write(ResourceFactory.newClassPathResource(rule)));
        return kieFileSystem;
    }

```

@Bean

```

public KieBuilder kieBuilder() {
    var kieBuilder = kieServices().newKieBuilder(getKieFileSystem());
    kieBuilder.buildAll();
    return kieBuilder;
}

```

@Bean

```

public KieContainer kieContainer() {
    var kb = kieBuilder().getKieModule();
    return kieServices().newKieContainer(kb.getReleaseId());
}

```

@Bean

```

public KieSession kieSession() {
    return kieContainer().newKieSession();
}

```

@Bean

```

public KieRepository kieRepository() {
    return kieServices().getRepository();
}
}

```

available_slots.drl

```
package rules;
```

```
import com.rehabilitation.dto.TimeSlotDto;
```

```
import com.rehabilitation.entity.Meeting;
```

```
import com.rehabilitation.entity.Vacation;
```

```
import java.time.LocalDate;
```

```
rule "Remove slot that overlaps with meeting"
```

```
when
```

```
    $slot : TimeSlotDto($sStart : start, $sEnd : end)
```

```
    Meeting(start < $sEnd, end > $sStart)
```

```
then
```

```
    retract($slot);
```

```
end
```

```
rule "Remove slot that overlaps with vacation"
```

```
when
```

```
    $slot : TimeSlotDto($sStart : start, $sEnd : end)
```

```
    Vacation($vStart : startDate, $vEnd : endDate)
```

```
    eval($sStart.toLocalDate().compareTo($vStart) >= 0 &&  
$sStart.toLocalDate().compareTo($vEnd) <= 0)
```

```
then
```

```
    retract($slot);
```

```
end
```

application.yaml

```
server:
```

```
    port: 8282
```

spring:

datasource:

username: root

password: rootroot

driver-class-name: com.mysql.cj.jdbc.Driver

url: jdbc:mysql://localhost:3306/calendar_db

hikari:

schema: calendar

jpa:

database-platform: org.hibernate.dialect.MySQLDialect

show-sql: true

hibernate:

ddl-auto: update

logging:

pattern:

console: "%d{yyyy-MM-dd HH:mm:ss.SSS} [%thread] %-5level %logger{36} -
%msg%n"

create-meeting.html

```
!DOCTYPE html>

<html lang="uk" xmlns:th="http://www.thymeleaf.org">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Створення зустрічі</title>

    <link rel="stylesheet" th:href="@{/css/create-meeting.css}">

    <link rel="stylesheet" th:href="@{/css/style.css}">

    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
rel="stylesheet"

        integrity="sha384-
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLA
SjC" crossorigin="anonymous">

    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/bootstrap-
datepicker/1.9.0/css/bootstrap-datepicker.min.css">

    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">

    <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>

    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-
alpha3/dist/js/bootstrap.bundle.min.js"></script>

    <script src="https://cdnjs.cloudflare.com/ajax/libs/bootstrap-
datepicker/1.9.0/js/bootstrap-datepicker.min.js"></script>

    <script src="https://cdnjs.cloudflare.com/ajax/libs/bootstrap-
datepicker/1.10.0/locales/bootstrap-datepicker.uk.min.js"></script>

    <script src="/js/datepicker.js" type="text/javascript"></script>

    <script src="/js/create-meeting.js" type="text/javascript"></script>

</head>

<body>

<header class="header">
```

```

<div class="container">
  <div class="header__inner">
    <a href="/" class="logo">WELLNESS</a>
    <nav class="nav">
      <a class="nav__link" th:href="@{/}">Головна сторінка</a>
      <a class="nav__link" th:href="@{/account}">Особистий кабінет</a>
    </nav>
  </div>
</div>
</header>
<div class="app-container">
  <main class="main-content-blue">
    <div class="create-meeting-container">
      <div class="create-meeting-header">
        <svg viewBox="0 0 24 24" fill="none" stroke="currentColor" stroke-
width="2" stroke-linecap="round" stroke-linejoin="round" class="header-
icon"><circle cx="12" cy="12" r="10"></circle><polyline points="12 6 12 16
14"></polyline></svg>
        <h2 class="header-title">Створення зустрічі</h2>
      </div>
      <form id="createMeetingForm" th:method="POST"
th:action="@{/calendar/new-meeting}" th:object="{meeting}" >
        <div class="form-group">
          <i><label for="meetingTitle" class="form-label">Назва
зустрічі</label></i>
          <input type="text" id="meetingTitle" class="form-input"
th:field="*{title}" placeholder="Введіть назву зустрічі" required>
        </div>
        <i><label class="booking_data_value">Оберіть дату</label></i>
        <div class="input-group date" data-provide="datepicker" id="datepicker">
          <input class="input_field form-control" type="text" placeholder="дата"
th:field="*{start}">
          <div class="input-group-addon">

```

```

        <span class="glyphicon glyphicon-th"></span>
    </div>

    <input class="input_field form-control" type="hidden"
placeholder="дата" th:value="\${meeting.start}" th:field="*{end}">

    <div class="input-group-addon">

        <span class="glyphicon glyphicon-th"></span>

    </div>

</div>

<input id="participantIds" class="input_field form-control" type="hidden"
placeholder="юзер ід" th:field="*{userIds}">

<br>

<div class="form-group">

    <i><label for="participants" class="form-label">Учасники</label></i>

    <div class="participants-input-wrapper">

        <input type="text" id="participants" class="form-input"
placeholder="Пошук співробітників">

        <div id="participantsList" class="participants-list"
th:value="*{userIds}">

            </div>

        </div>

    </div>

</div>

<div class="form-group">

    <i><label class="form-label">Додані учасники: </label></i>

    <div id="addedParticipants" class="added-participants">

        </div>

    </div>

    <button type="submit" class="button button--primary">

        <a class="main_buttons_text">Створити зустріч</a>

    </button>

</form>

</div>

</main>

```

```
</div>
</body>
</html>
```

calendar.js

```
$(document).ready(function() {

    $('#calendar').fullCalendar({
        header: {
            left: 'prev,next today',
            center: 'title',
            right: 'month,basicWeek,basicDay'
        },
        locale: "uk",
        weekends: false,
        defaultDate: Date.now(),
        navLinks: true, // can click day/week names to navigate views
        editable: true,
        eventLimit: true, // allow "more" link when too many events
        events: {
            url: 'http://localhost:8282/meetings',
            method: 'GET',
            failure: function() {
                alert('there was an error while fetching events!');
            },
            color: '#224E6F', // a non-ajax option
            textColor: 'white' // a non-ajax option
        }
    });
});
```

```
});
```

create-meeting.css

```
.create-meeting-container {  
  background-color: #fff;  
  padding: 30px;  
  border-radius: 8px;  
  box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);  
  width: 600px;  
  margin: 30px auto;  
}
```

```
.create-meeting-header {  
  display: flex;  
  align-items: center;  
  margin-bottom: 20px;  
}
```

```
.header-icon {  
  width: 24px;  
  height: 24px;  
  margin-right: 10px;  
  color: #314784;  
}
```

```
.header-title {  
  font-size: 24px;  
  color: #333;  
}
```

```
.form-group {  
  margin-bottom: 20px;  
}
```

```
.form-label {  
  display: block;  
  font-size: 16px;  
  color: #555;  
  margin-bottom: 5px;  
}
```

```
.form-input {  
  width: 100%;  
  padding: 10px;  
  border: 1px solid #ccc;  
  border-radius: 4px;  
  font-size: 16px;  
  box-sizing: border-box;  
}
```

```
.form-row {  
  display: flex;  
  gap: 20px;  
}
```

```
.form-row .form-group {  
  flex: 1;  
}
```

```
.input-with-icon {  
  position: relative;  
}
```

```
.input-icon {  
  position: absolute;  
  right: 10px;  
  top: 50%;  
  transform: translateY(-50%);  
  width: 20px;  
  height: 20px;  
  color: #777;  
}
```

```
.participants-input-wrapper {  
  position: relative;  
}
```

```
#participantsList {  
  position: absolute;  
  top: 100%;  
  left: 0;  
  right: 0;  
  background-color: #fff;  
  border: 1px solid #ccc;  
  border-top: none;  
  border-radius: 0 0 4px 4px;  
  z-index: 10;  
  max-height: 200px;  
  overflow-y: auto;
```

```
display: none; /* Приховати список за замовчуванням */
}

.participant-item {
padding: 10px;
display: flex;
justify-content: space-between;
align-items: center;
border-bottom: 1px solid #eee;
}

.participant-item:last-child {
border-bottom: none;
}

.participant-info {
flex-grow: 1;
}

.participant-name {
font-weight: bold;
}

.participant-role {
color: #777;
font-size: 14px;
}

.add-participant-button {
background-color: #314784;
```

```
color: #fff;
border: none;
padding: 8px 12px;
border-radius: 4px;
cursor: pointer;
font-size: 14px;
transition: background-color 0.3s ease;
}
```

```
.add-participant-button:hover {
  background-color: #4a69bd;
}
```

```
#addedParticipants {
  display: flex;
  gap: 10px;
  padding: 5px 0;
  align-items: center;
}
```

```
.added-participant-tag {
  background-color: #e0f2f7;
  color: #314784;
  padding: 5px 8px;
  border-radius: 4px;
  font-size: 14px;
  display: flex;
  align-items: center;
}
```

```
.remove-participant-button {  
  background: none;  
  border: none;  
  color: #d32f2f;  
  cursor: pointer;  
  margin-left: 5px;  
  font-size: 16px;  
  line-height: 1;  
}
```

```
.remove-participant-button:hover {  
  color: #f44336;  
}
```

```
.button--primary {  
  padding: 12px 20px;  
  font-size: 18px;  
  border: none;  
  text-decoration: none;  
  text-color: white;  
}
```

```
.main-content-blue {  
  flex-grow: 1;  
  padding: 30px;  
  background-color: #F0F6FF;  
}
```

```
.datepicker td{  
  text-align: center;
```

```

padding: 8px 20px;
font-size: 14px;
color: #3498db;
font-weight: 400;
}
.datepicker th{
    text-align: center;
padding: 8px 20px;
font-size: 14px;
background-color: #3498db;
color: #FFFFFF;
}

```

RehabilitationApplication.java

```

package com.rehabilitation;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class RehabilitationApplication {

    public static void main(String[] args) {
        SpringApplication.run(RehabilitationApplication.class, args);
    }
}

```



Звіт подібності

метадані

Назва організації

Kyiv National Economic University named after Vadym Hetman KNEU

Заголовок

Розроблення інформаційної системи для реабілітаційного центру

Автор

Науковий керівник / Експерт

Фурманюк Ірина Сергіївна Pinna С.П.

підрозділ

кафедра інформаційних систем в економіці

Обсяг знайдених подібностей

Коефіцієнт подібності визначає, який відсоток тексту по відношенню до загального обсягу тексту було знайдено в різних джерелах. Зверніть увагу, що високі значення коефіцієнта не автоматично означають плагіат. Звіт має аналізувати компетентна / уповноважена особа.



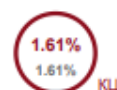
25

Довжина фрази для коефіцієнта подібності 2



7379

Кількість слів



58772

Кількість символів

Тривога

У цьому розділі ви знайдете інформацію щодо текстових спотворень. Ці спотворення в тексті можуть говорити про МОЖЛИВІ маніпуляції в тексті. Спотворення в тексті можуть мати навмисний характер, але частіше характер технічних помилок при конвертації документа та його збереженні, тому ми рекомендуємо вам підходити до аналізу цього модуля відповідально. У разі виникнення запитань, просимо звертатися до нашої служби підтримки.

Заміна букв		1
Інтервали		0
Мікропробіли		34
Білі знаки		0
Парафрази (SmartMarks)		6

Подібності за списком джерел

Нижче наведений список джерел. В цьому списку є джерела із різних баз даних. Колір тексту означає в якому джерелі він був знайдений. Ці джерела і значення Коефіцієнту Подібності не відображають прямого плагіату. Необхідно відкрити кожне джерело і проаналізувати зміст і правильність оформлення джерела.

10 найдовших фраз		Колір тексту
ПОРЯДКОВИЙ НОМЕР	НАЗВА ТА АДРЕСА ДЖЕРЕЛА URL (НАЗВА БАЗИ)	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
1	https://www.wikiwand.com/uk/Model-view-controller	49 0.66 %
2	Облік розрахунків із заробітної плати у ДВНЗ «КНЕУ ім. В. Гетьмана».docx 5/27/2020 Kyiv National Economic University named after Vadym Hetman KNEU (кафедра обліку в кредитних і бюджетних установах та економічного аналізу)	21 0.28 %

3	http://reposit.univ-sport.edu.ua/bitstream/handle/787878787/1378/MatKonf2018_NUPESU2%20%281%29.pdf?sequence=1&isAllowed=y	20 0.27 %
4	https://docplayer.net/54674713-Issn-richchyu-mntk-vottp-i-20-richchyu-mntzh-vottp-prisvyachuietsya.html	14 0.19 %
5	Система управління ресторану (програмне рішення) 6/12/2020 Kyiv National Economic University named after Vadym Hetman KNEU (кафедра інформаційних систем в економіці)	12 0.16 %
6	Some Aspects of Using Node.js and MongoDB in Creating Educational Web Applications Ткаченко Олександр, Бондар Віталій;	11 0.15 %
7	KPM_Мацьков О.Г..docx 12/8/2023 Bogomolets National Medical University (BNMU) (Departments of NMU)	10 0.14 %
8	Інформаційна система аналізу CEO показників сайту 6/22/2020 Kyiv National Economic University named after Vadym Hetman KNEU (кафедра інформаційних систем в економіці)	9 0.12 %
9	Система управління ресторану (програмне рішення) 6/12/2020 Kyiv National Economic University named after Vadym Hetman KNEU (кафедра інформаційних систем в економіці)	9 0.12 %
10	https://od.kubg.edu.ua/index.php/journal/article/view/301	8 0.11 %
з бази даних RefBooks (0.15 %)		
ПОРЯДКОВИЙ НОМЕР	ЗАГОЛОВЕ	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
джерело: Paperity		
1	Some Aspects of Using Node.js and MongoDB in Creating Educational Web Applications Ткаченко Олександр, Бондар Віталій;	11 (1) 0.15 %
з домашньої бази даних (0.89 %)		
ПОРЯДКОВИЙ НОМЕР	ЗАГОЛОВЕ	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
1	Система управління ресторану (програмне рішення) 6/12/2020 Kyiv National Economic University named after Vadym Hetman KNEU (кафедра інформаційних систем в економіці)	28 (3) 0.38 %
2	Облік розрахунків із заробітної плати у ДВНЗ «КНЕУ ім. В. Гетьмана».docx 5/27/2020 Kyiv National Economic University named after Vadym Hetman KNEU (кафедра обліку в кредитних і бюджетних установах та економічного аналізу)	21 (1) 0.28 %
3	Інформаційна система аналізу CEO показників сайту 6/22/2020 Kyiv National Economic University named after Vadym Hetman KNEU (кафедра інформаційних систем в економіці)	9 (1) 0.12 %
4	Інформаційна система з управління договорами страхової компанії 6/19/2018 Kyiv National Economic University named after Vadym Hetman KNEU (кафедра інформаційних систем в економіці)	8 (1) 0.11 %
з програми обміну базами даних (0.53 %)		

ПОРЯДКОВИЙ НОМЕР	ЗАГОЛОВОК	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
1	Розроблення вебзастосунку «Облік послуг салону краси» 5/24/2023 Kharkiv National University of Economics named after S.Kuznets (KNUE) (KNUE)	18 (3) 0.24 %
2	KPM_Мацьков О.Г..docx 12/8/2023 Bogomolets National Medical University (BNMU) (Departments of NMU)	16 (2) 0.22 %
3	Розробка web-моделі розрахунку енергозатрат 12/13/2018 National University Chernihiv Politechnika (NUCP) 2 (Науковий вісник Полісся)	5 (1) 0.07 %

з Інтернету (1.80 %)

ПОРЯДКОВИЙ НОМЕР	ДЖЕРЕЛО URL	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)
1	https://www.wikiwand.com/uk/Model-view-controller	49 (1) 0.66 %
2	https://od.kubg.edu.ua/index.php/journal/article/view/301	27 (4) 0.37 %
3	http://deposit.uni-sport.edu.ua/bitstream/handle/78787878/1378/MatKonf2018_NUPESU2%20%281%29.pdf?sequence=1&isAllowed=y	26 (2) 0.35 %
4	https://docplayer.net/54674713-Issn-richchyu-mntk-vottp-i-20-richchyu-mntzh-vottp-prisvyachuietsya.html	14 (1) 0.19 %
5	http://fisiit.kneu.edu.ua/ua/depts9/k_informacijnyh_system_v_ekonomici/	11 (2) 0.15 %
6	https://fisiit.kneu.edu.ua/ua/depts9/k_informacijnyh_system_v_ekonomici/vikladi21/Krasnyuk.M.T/	6 (1) 0.08 %

Список прийнятих фрагментів (немає прийнятих фрагментів)

ПОРЯДКОВИЙ НОМЕР	ЗМІСТ	КІЛЬКІСТЬ ОДНАКОВИХ СЛІВ (ФРАГМЕНТІВ)
---------------------	-------	---------------------------------------