

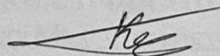
Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра автоматизації та інтелектуальних інформаційних технологій

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

Розробка автоматизованої системи управління електронною чергою
медичного закладу

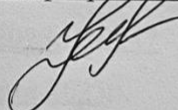
Виконав: студент 1 курсу, групи ЗАКІТ-22м
спеціальності 151 – Автоматизація та
комп'ютерно-інтегровані технології



Максим КОНЕЦУЛ

Керівник: д.т.н., доцент,

т.в.о.зав. кафедри КСУ



Марія ЮХИМЧУК

« 4 » зрудня 2023 р.

Опонент: доцент. каф. КН



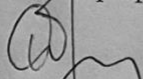
Володимир ОЗЕРАНСЬКИЙ

« 8 » зрудня 2023

Допущено до захисту

Зав. кафедри АІТ,

д.т.н. проф.



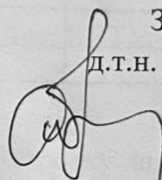
Олег БІСКАЛО

« 11 » зрудня 2023 р.

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра автоматизації та інтелектуальних інформаційних технологій
Рівень вищої освіти другий (магістерський)
Галузь знань – 15 – Автоматизація та приладобудування
Спеціальність – 151 – Автоматизація та комп'ютерно-інтегровані технології
Освітня програма – Інформаційні системи і Інтернет речей

ЗАТВЕРДЖУЮ

Завідувач кафедри АІТ
д.т.н. проф. Олег БІСКАЛО



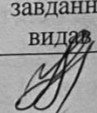
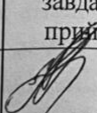
“20” вересня 2023 року

**ЗАВДАННЯ
НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ**

студенту Конецулу Максиму Вікторовичу
(прізвище, ім'я, по батькові)

1. Тема магістерської кваліфікаційної роботи «Розробка автоматизованої системи управління електронною чергою медичного закладу»
керівник роботи Юхимчук Марія Сергіївна
затверджені наказом ВНТУ від “18” вересня 2023р. №247
2. Термін подання студентом роботи “12” грудня 2023р.
3. Вихідні дані до магістерської кваліфікаційної роботи максимальний рівень шуму у приміщенні – 70 дБ; підтримка ОС – Windows: максимальний час завантаження – 5 с; авторизація користувачів – так; максимальна кількість запитів до системи – до 1000 шт/с; мови графічного та голосового інтерфейсів – українська, англійська, російська; достовірність розпізнавання мови – 95%; використання кодування під час передачі даних – так.
4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) вступ, дослідження об'єкта автоматизації, дослідження основних аспектів автоматизації системи управління електронною чергою, дослідження взаємодії модуля управління електронною чергою з іншими модулями в автоматизованій системі управління медичними закладами, дослідження роботи та порівняльний аналіз існуючих систем управління електронною чергою медичного закладу, проектування та розробка автоматизованої системи управління електронною чергою медичного закладу.
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) ER-модель бази даних системи, UML-діаграма класів системи, UML-діаграма варіантів використання автоматизованої системи, вигляд екрану розробленого web-додатку – 13 шт.

6. Консультанти розділів роботи

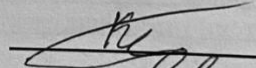
Розділ змістової частини роботи	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1-5	д.т.н., доцент, т.в.о.зав. кафедри КСУ Марія ЮХИМЧУК	 20.09.2023	 12.12.2023

КАЛЕНДАРНИЙ ПЛАН

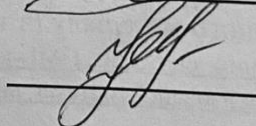
№ з/п	Назва етапів роботи	Строк виконання етапів роботи	Примітка
1	Дослідження актуальності поставленої задачі	20.09.2023р.	викона
2	Дослідження основних аспектів автоматизації системи управління електронною чергою	27.09.2023р.	викона
3	Дослідження роботи та порівняльний аналіз існуючих систем управління електронною чергою медичного закладу	01.10.2023р.	викона
4	Проектування та розробка автоматизованої системи управління електронною чергою медичного закладу	05.10.2023р.	викона
5	Оформлення пояснювальної записки, графічного матеріалу і презентації	29.10.2023р.	викона
6	Графічні матеріали: Use-case UML-діаграма UML-діаграма класів ER-діаграма структури бази даних вигляд екранів розробленого web-додатку	05.11.2023р. 14.11.2023р.	викона
7	Захист МКР	20.11.2023р. 14.12.2023р.	

Дата видачі завдання "20" вересня 2023р.

Студент

 Максим КОНЕЦЬ

Керівник роботи

 Марія ЮХИМЧУК

УДК 614.2

Кон

чергою м

151 – Авт

Інформац

Бібліогр.

У

управлін

автомати

черги, до

модулям

Пр

медични

включас

ролях У

роботи с

К

оптиміза

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра автоматизації та інтелектуальних інформаційних технологій

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

Розробка автоматизованої системи управління електронною чергою
медичного закладу

Виконав: студент 1 курсу, групи ЗАКІТ-22м
спеціальності 151 – Автоматизація та
комп'ютерно-інтегровані технології

Максим КОНЕЦУЛ

Керівник: д.т.н., доцент,

т.в.о.зав. кафедри КСУ

Марія ЮХИМЧУК

«_____» _____ 2023 р.

Опонент: доцент, каф. КН

Володимир ОЗЕРАНСЬКИЙ

«_____» _____ 2023

Допущено до захисту

Зав. кафедри АІТ,

д.т.н. проф.

Олег БІСІКАЛО

«_____» _____ 2023 р.

Вінниця ВНТУ – 2023 рік

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра автоматизації та інтелектуальних інформаційних технологій
Рівень вищої освіти другий (магістерський)
Галузь знань – 15 – Автоматизація та приладобудування
Спеціальність – 151 – Автоматизація та комп'ютерно-інтегровані технології
Освітня програма – Інформаційні системи і Інтернет речей

ЗАТВЕРДЖУЮ

Завідувач кафедри АІТ
д.т.н. проф. Олег БІСІКАЛО

“20” вересня 2023 року

ЗАВДАННЯ НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ

студенту Конечулу Максиму Вікторовичу
(прізвище, ім'я, по батькові)

1. Тема магістерської кваліфікаційної роботи «Розробка автоматизованої системи управління електронною чергою медичного закладу»

керівник роботи Юхимчук Марія Сергіївна

затверджені наказом ВНТУ від “18” вересня 2023р. №247

2. Термін подання студентом роботи “12” грудня 2023р.

3. Вихідні дані до магістерської кваліфікаційної роботи максимальний рівень шуму у приміщенні – 70 дБ; підтримка ОС – Windows; максимальний час завантаження – 5 с; авторизація користувачів – так; максимальна кількість запитів до системи – до 1000 шт/с; мови графічного та голосового інтерфейсів – українська, англійська, російська; достовірність розпізнавання мови – 95%; використання кодування під час передачі даних – так.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) вступ, дослідження об'єкта автоматизації, дослідження основних аспектів автоматизації системи управління електронною чергою, дослідження взаємодії модуля управління електронною чергою з іншими модулями в автоматизованій системі управління медичними закладами, дослідження роботи та порівняльний аналіз існуючих систем управління електронною чергою медичного закладу, проектування та розробка автоматизованої системи управління електронною чергою медичного закладу.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) ER-модель бази даних системи, UML-діаграма класів системи, UML-діаграма варіантів використання автоматизованої системи, вигляд екрану розробленого web-додатку – 13 шт.

6. Консультанти розділів роботи

Розділ змістової частини роботи	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1-5	д.т.н., доцент, т.в.о.зав. кафедри КСУ Марія ЮХИМЧУК	20.09.2023	12.12.2023

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Строк виконання етапів роботи	Примітка
1	Дослідження актуальності поставленої задачі	20.09.2023р.	виконав
2	Дослідження основних аспектів автоматизації системи управління електронною чергою	27.09.2023р.	виконав
3	Дослідження роботи та порівняльний аналіз існуючих систем управління електронною чергою медичного закладу	01.10.2023р.	виконав
4	Проектування та розробка автоматизованої системи управління електронною чергою медичного закладу	05.10.2023р.	виконав
5	Оформлення пояснювальної записки, графічного матеріалу і презентації	29.10.2023р.	виконав
6	Графічні матеріали: Use-case UML-діаграма UML-діаграма класів ER-діаграма структури бази даних вигляд екранів розробленого web-додатку	05.11.2023р. 14.11.2023р. 20.11.2023р.	виконав
7	Захист МКР	14.12.2023р.	

Дата видачі завдання “20” вересня 2023р.

Студент _____ Максим КОНЕЦУЛ

Керівник роботи _____ Марія ЮХИМЧУК

АНОТАЦІЯ

УДК 614.2:004.9

Конецул М.В. Розробка автоматизованої системи управління електронною чергою медичного закладу. Магістерська кваліфікаційна робота зі спеціальності 151 – Автоматизація та комп'ютерно-інтегровані технології, освітня програма – Інформаційні системи і Інтернет речей. Вінниця: ВНТУ, 2023. 116 с. На укр. мові. Бібліогр.: 2 назв; рис.: 16; табл. 4.

У магістерській роботі розглянуто розробку автоматизованої системи управління електронною чергою медичного закладу. В ході дослідження об'єкта автоматизації вивчені основні аспекти функціонування систем електронної черги, досліджено взаємодію модуля управління електронною чергою з іншими модулями в автоматизованій системі управління медичними закладами.

Проведено порівняльний аналіз існуючих систем управління чергою в медичних закладах. Розроблено технічні та програмні засоби системи, включаючи UML-діаграми та функціональні можливості для користувачів у ролях User та Administrator. Результати тестування свідчать про правильність роботи системи та зручність її використання.

Ключові слова: управління чергою, безпека даних, Agile методика, оптимізація процесів, тестування системи, медичні заклади.

ANNOTATION

Konetsul M.V. Development of an Automated System for Managing Electronic Queues in a Medical Facility. Master's thesis in the field of 151 – Automation and Computer-Integrated Technologies, Educational Program – Information Systems and Internet of Things. Vinnytsia: Vinnytsia National Technical University, 2023. 116 pages. In Ukrainian. Bibliography: 2 titles; figures: 16; tables: 4.

The master's thesis explores the development of an automated queue management system for a medical facility. During the investigation of the automation object, fundamental aspects of the electronic queue systems' operation were studied, and the interaction of the electronic queue management module with other modules in the automated system for managing medical facilities was examined.

A comparative analysis of existing queue management systems in medical facilities was conducted. The technical and software components of the system were developed, including UML diagrams and functional capabilities for users in the roles of User and Administrator. The testing results affirm the correctness of the system's functioning and its user-friendly nature.

Keywords: queue management, data security, Agile methodology, process optimization, system testing, medical institutions.

ЗМІСТ

ВСТУП.....	8
1 ДОСЛІДЖЕННЯ ОБ'ЄКТА АВТОМАТИЗАЦІЇ.....	11
2 ДОСЛІДЖЕННЯ ОСНОВНИХ АСПЕКТІВ АВТОМАТИЗАЦІЇ СИСТЕМИ УПРАВЛІННЯ ЕЛЕКТРОННОЮ ЧЕРГОЮ	14
2.1 Аналіз особливостей роботи систем електронної черги.....	14
2.2 Особливості підключення та налаштування бази даних.....	17
2.3 Дослідження переваг інтеграції з іншими системами.....	21
2.4 Дослідження автоматизованого оповіщення та розсилка інформації.....	23
2.5 Відслідковування та аналіз даних.....	25
2.6 Аналіз необхідності технічної підтримки та безпеки системи	26
3 ДОСЛІДЖЕННЯ ВЗАЄМОДІЇ МОДУЛЯ УПРАВЛІННЯ ЕЛЕКТРОННОЮ ЧЕРГОЮ З ІНШИМИ МОДУЛЯМИ В АВТОМАТИЗОВАНІЙ СИСТЕМІ УПРАВЛІННЯ МЕДИЧНИМИ ЗАКЛАДАМИ	29
3.1 Інтеграція з іншими системами управління медичними закладами...29	
3.2 Оптимізація обміну даними між модулями.....	30
3.3 Синхронізація роботи модуля зі стандартами безпеки та конфіденційності.....	32
4 ДОСЛІДЖЕННЯ РОБОТИ ТА ПОРІВНЯЛЬНИЙ АНАЛІЗ ІСНУЮЧИХ СИСТЕМ УПРАВЛІННЯ ЕЛЕКТРОННОЮ ЧЕРГОЮ МЕДИЧНОГО ЗАКЛАДУ	39
4.1 Огляд сервісу Health24	41
4.2 Огляд сервісу helsi.me.....	44
4.3 Огляд сервісу MedCard24	47
4.4 Огляд сервісу Medics	49
5 ПРОЕКТУВАННЯ ТА РОЗРОБКА АВТОМАТИЗОВАНОЇ СИСТЕМИ	

УПРАВЛІННЯ ЕЛЕКТРОННОЮ ЧЕРГОЮ МЕДИЧНОГО ЗАКЛАДУ ...	54
5.1 Проектування автоматизованої системи управління.....	54
5.2 Опис технічних засобів та мов програмування	63
5.3 Розробка та реалізація автоматизованої системи управління електронною чергою медичного закладу.....	64
ВИСНОВКИ	74
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	76
Додатки	81
Додаток А.....	82
Додаток Б	86
Додаток В.....	103
Додаток Г	110

ВСТУП

Актуальність. У нашому епохальному інформаційному часі, автоматизація стала невід'ємною частиною сучасного життя. Незалежно від галузі діяльності, автоматизовані системи змінюють парадигму управління і взаємодії. Серед них особливе місце посідають системи управління чергою, які стали необхідністю для різноманітних сфер, але особливо важливі для медичних закладів.

Завдяки своїй унікальній природі, автоматизовані системи управління чергою стали надзвичайно популярними в сучасному суспільстві. Однією з основних причин цієї популярності є доступність, швидкість та зручність, яку вони принесли у повсякденне життя.

Медичні заклади – це одна з областей, де автоматизовані системи управління чергою виявилися особливо ефективними. Такі системи дозволяють медичним закладам оптимізувати процес надання медичних послуг і забезпечують ряд переваг для всіх сторін, які беруть участь у цьому процесі [1].

По-перше, автоматизовані системи зменшують фізичний контакт між пацієнтами, лікарями та іншим персоналом. У ситуаціях, коли безпека є пріоритетом, це особливо важливо. Пацієнти можуть реєструватися в черзі та очікувати свого чергового візиту без необхідності бути фізично присутніми в коридорах лікарні, що знижує ризик поширення інфекційних захворювань.

По-друге, ці системи полегшують взаємодію між пацієнтами та адміністраторами медичних закладів. Вони можуть записуватися на прийоми, отримувати сповіщення про свій стан у черзі і навіть вибирати оптимальний час для відвідин лікаря без довгих телефонних розмов і чекання на лінії.

По-третє, важливо відзначити, що автоматизовані системи управління чергою сприяють ефективній організації роботи медичного персоналу. Лікарі та медичні сестри можуть більш точно розпланувати свій робочий день, завдяки передбаченню обсягу роботи та кількості пацієнтів на прийомі. Це дозволяє

оптимізувати час, знижувати очікування пацієнтів і враховувати нагальність медичних випадків.

Більше того, автоматизовані системи забезпечують точний облік пацієнтів, допомагаючи медичному закладу уникнути втрати даних і забезпечуючи їхню безпеку. Вони також здатні надавати аналітичну інформацію, яка допомагає адміністраторам і лікарям у прийнятті більш обґрунтованих рішень для покращення роботи медичного закладу [2].

Завдяки цим перевагам, автоматизовані системи управління чергою в медичних закладах стали важливим інструментом для поліпшення доступу до медичних послуг, зниження витрат часу та ресурсів, а також підвищення якості медичного обслуговування.

Крім цього, важливо відзначити, що автоматизовані системи управління чергою виконують важливу роль у забезпеченні відповідності медичних закладів стандартам і регуляціям у галузі охорони здоров'я, а також у підвищенні задоволеності пацієнтів та оптимізації роботи персоналу [3].

Отже, дослідження даного напрямку дозволить підкреслити важливість цієї інновації для забезпечення зручності і безпеки медичних послуг для всіх її учасників.

Мета і завдання дослідження. Метою магістерської кваліфікаційної роботи є розробка та впровадження автоматизованої системи управління електронною чергою для медичного закладу з метою поліпшення організації надання медичних послуг та оптимізації робочих процесів. Робота спрямована на створення інноваційного інструменту, який сприятиме покращенню доступу до медичних послуг, зменшенню адміністративних навантажень та підвищенню задоволення як пацієнтів, так і медичного персоналу медичного закладу.

Для досягнення наведеної мети були поставлені та вирішені наступні задачі:

1. проведено аналіз сучасних аспектів автоматизації систем управління електронною чергою;
2. досліджено взаємодію модуля управління електронною чергою з іншими модулями в автоматизованій системі управління медичними закладами;

3. проведено дослідження роботи та порівняльний аналіз існуючих систем управління електронною чергою медичного закладу;
4. досліджено практичні аспекти функціонування автоматизованої системи управління електронною чергою медичного закладу;
5. спроектовано та розроблено автоматизовану систему управління електронною чергою медичного закладу;
6. проведено дослідження та тестування отриманих результатів.

Об'єктом дослідження є процес розробки та впровадження автоматизованої системи управління електронною чергою для медичного закладу.

Предметом дослідження є методи та засоби програмування, за допомогою яких можна виконати розробку та впровадження автоматизованої системи управління електронною чергою для медичного закладу.

Методи дослідження. У процесі дослідження застосовувалися: теорія алгоритмів, Agile підхід до розробки програмного забезпечення, методика Scrum, методи управління базами даних, методи розробки веб-сайту, засоби безпеки та захисту даних.

Новизна одержаних результатів.

1. Проведено дослідження роботи існуючих систем управління електронними чергами в медичних закладах та сформульовано список вимог до системи, що розроблялася.
2. Спроектовано та розроблено автоматизовану систему управління електронною чергою медичного закладу.
3. Впроваджено методи управління транзакціями бази даних, що гарантують ізольованість та атомарність операцій.

Практичне значення одержаних результатів полягає в тому, що на основі отриманих теоретичних положень виконано розробку та впровадження автоматизованої системи управління електронною чергою для медичного закладу.

1 ДОСЛІДЖЕННЯ ОБ'ЄКТА АВТОМАТИЗАЦІЇ

Темою магістерської кваліфікаційної роботи є розробка автоматизованої системи управління електронною чергою медичного закладу.

В рамках дослідження актуальності теми та планування етапів виконання роботи необхідно дослідити особливості систем електронної черги, варіанти підключення та налаштування бази даних, можливості інтеграції з іншими системами, особливості розробки інтерфейсу для систем керування чергою, типи автоматизованого оповіщення та розсилки інформації, відслідковування та аналіз даних, реалізацію технічної підтримки та безпеки автоматизованої системи.

Аналіз кожного етапу допоможе збудувати повне бачення готового програмного продукту та його функціональності, розробити на основі цього необхідні UML та ER діаграми, а також визначитися з необхідним програмним забезпеченням.

Місце задачі, що розв'язується в роботі, у загальному контексті автоматизації систем управління чергою в медичних закладах надзвичайно важливе та відіграє значущу роль у поліпшенні організації медичного обслуговування та підвищенні комфорту як для пацієнтів, так і для медичного персоналу та адміністраторів.

Автоматизована система управління електронною чергою, що буде спроектована та розроблена в даній роботі, вирішує ряд ключових задач. Однією з основних є зниження фізичного контакту між пацієнтами та медичним персоналом, що є особливо важливим у сучасних умовах підвищеної уваги до гігієни та здоров'я. Це забезпечує надійний та безпечний спосіб реєстрації та очікування прийому для пацієнтів, які можуть це робити з власного приміщення.

Крім того, автоматизована система дозволяє пацієнтам реєструватися на прийоми онлайн, вибирати доступні дати та часи, а також отримувати сповіщення про свій стан у черзі та можливість прийому. Це робить процес отримання медичних послуг більш комфортним та ефективним.

Додатково, система допомагає медичному персоналу точно планувати графік прийому, враховуючи нагальність випадків та об'єм роботи. Це дозволяє оптимізувати робочий день лікарів та медичних сестер, зменшуючи час очікування пацієнтів та підвищуючи загальну продуктивність.

Усі ці функції та можливості роблять автоматизовану систему управління чергою невід'ємною складовою сучасного медичного закладу, забезпечуючи зручність та надійність при наданні медичних послуг, а також ефективне використання ресурсів та зменшення негативного впливу на здоров'я пацієнтів та медичного персоналу.

Область застосування розробки

Розроблена автоматизована система управління електронною чергою в медичному закладі має широкий спектр можливих застосувань. Перш за все, вона стане невід'ємною складовою для будь-якого медичного закладу, незалежно від його масштабу та спеціалізації. Така система може бути успішно впроваджена в лікарнях, поліклініках, стоматологічних клініках, діагностичних центрах, інших медичних установах та підприємствах охорони здоров'я.

Крім того, автоматизована система може знайти застосування в інших галузях, де важливо ефективно керувати чергою та надавати послуги за попереднім записом. Наприклад, в громадських установах, таких як паспортні служби, установи соціального обслуговування, банки, торгові центри та будь-які інші організації, де важливо організовувати прийом клієнтів з дотриманням правил безпеки та мінімізації очікування.

Ця універсальність та гнучкість системи дозволяють застосовувати її в різних галузях і сприяють вдосконаленню організації робочих процесів та покращенню обслуговування клієнтів у всіх сферах діяльності, де важлива ефективна управління чергою та зручність для всіх учасників.

Вимоги до програмно-технічного забезпечення, що розробляється в магістерській кваліфікаційній роботі, можна описати наступним чином:

1. локальна база даних: для зберігання та управління інформацією про черги, пацієнтів та робочі процеси в медичному закладі

використовується база даних, що ґрунтується на системі MySQL. Ця система забезпечує ефективний доступ до даних та забезпечує їх цілісність.

2. веб-розробка: створення та розгортання веб-сайту з автоматизованою системою управління електронною чергою використовує сучасні інструменти та технології веб-розробки. Мови розмітки HTML та CSS використовуються для створення інтерфейсу користувача, JavaScript

3. для реалізації взаємодії на стороні клієнта. Python є основною мовою програмування для розробки веб-додатків з використанням фреймворку Django.

4. серверні рішення: для розгортання та управління веб-сервером, базою даних та контейнеризації використовується Docker. Це дозволяє створювати та використовувати контейнери для зручного розгортання та управління середовищем веб-додатку.

5. засоби безпеки: забезпечення конфіденційності та цілісності даних пацієнтів є пріоритетом у розробці. Використовуються засоби шифрування та захисту даних на рівні веб-застосунку та бази даних. Додатково, використання Docker дозволяє ізолювати та захищати компоненти системи в контейнерах для підвищення безпеки.

Функції розробки:

- забезпечення доступу користувачів до системи та персональних акаунтів;
- управління пошуком лікарів;
- бронювання та управління прийомами;
- перегляд історії візитів.

Для виконання роботи необхідні наступні вхідні дані:

- локальна база даних, яку описує ER-діаграма;
- відповідно налаштовані програмні та технічні засоби, що необхідні для розробки та розгортання автоматизованої системи управління електронною чергою медичного закладу.

2. ДОСЛІДЖЕННЯ ОСНОВНИХ АСПЕКТІВ АВТОМАТИЗАЦІЇ СИСТЕМИ УПРАВЛІННЯ ЕЛЕКТРОННОЮ ЧЕРГОЮ

2.1. Аналіз особливостей роботи систем електронної черги

В сучасному світі, де технології відіграють важливу роль у всіх аспектах нашого життя, медичні заклади не можуть залишатися осторонь. Однією з ключових інновацій, яка революціонує медичну сферу, є впровадження електронних черг. Основні концепції та цілі цієї технології завжди були і залишаються орієнтованими на поліпшення різних аспектів медичного обслуговування [5].

1. Покращення доступності та зручності:

Однією з головних мет цієї ініціативи є забезпечення швидкого та легкого доступу пацієнтів до медичного обслуговування. Електронні черги дозволяють пацієнтам реєструватися та записуватися на прийоми в онлайн-режимі, уникнувши довгих черг та зайвих очікувань.

2. Оптимізація робочих процесів:

Електронні черги допомагають оптимізувати робочі процеси в медичних закладах. Лікарі та медичний персонал можуть ефективно планувати графік прийому та розподіляти ресурси, щоб відповідати потребам пацієнтів. Це дозволяє зменшити час очікування пацієнтів, забезпечуючи більше точний та оперативний прийом.

3. Збереження даних та підвищення безпеки:

Електронні черги забезпечують збереження важливої медичної інформації в цифровому форматі, що робить її доступною та легкою в управлінні. Також, ця технологія дозволяє застосовувати високі стандарти безпеки, забезпечуючи захист конфіденційної інформації та персональних даних пацієнтів.

4. Зменшення людського фактору:

Електронні черги допомагають знизити вплив людського фактору на організацію черг та призначення прийому. Це робить процес більш об'єктивним

та усуває можливі помилки, що можуть виникнути в результаті людського недбалості або неправильного розподілу ресурсів.

Усі ці концепції та цілі електронних черг спрямовані на покращення медичної допомоги, роблячи її більш доступною, ефективною та безпечною для всіх учасників процесу - від пацієнтів до медичного персоналу та адміністрації медичних закладів [6.7].

Впровадження електронних черг в медичних закладах принесло численні переваги порівняно з традиційними системами чергування, які базуються на паперових списках та телефонних дзвінках. Однак разом з перевагами існують і недоліки, які варто враховувати.

Переваги:

1. зручність для пацієнтів;
2. оптимізація часу;
3. збереження даних;
4. безпека даних.

Недоліки:

1. необхідність доступу до Інтернету: для користування електронними чергами пацієнтам необхідний доступ до Інтернету, що може створювати несприятливі умови для деяких груп населення;
2. витрати на впровадження: впровадження електронних систем може вимагати значних фінансових витрат на розробку, обладнання та навчання персоналу;
3. технічні проблеми: технічні збої, такі як відключення Інтернету чи системні помилки, можуть призвести до перебоїв у роботі системи чергування;
4. опір з боку персоналу: деякі члени медичного персоналу можуть виявити опір перед впровадженням нової технології та відчувати незручність у переході зі звичних методів роботи.

В розробці автоматизованої системи управління електронною чергою для медичного закладу існують також певні вимоги, які мають бути враховані для

забезпечення її ефективності, надійності та зручності для користувачів, наприклад [8]:

Зручний інтерфейс користувача: Система повинна мати інтуїтивно зрозумілий та зручний інтерфейс для користувачів, що включає в себе пацієнтів, медичний персонал та адміністраторів. Це дозволить легко реєструватися, записуватися та отримувати інформацію про стан черги.

Онлайн-реєстрація та запис на прийом: Система повинна надавати можливість пацієнтам реєструватися та записуватися на прийоми через Інтернет, вказуючи зручний для них час та лікаря.

Моніторинг та оновлення черги в реальному часі: Система повинна надавати можливість адміністраторам та медичному персоналу моніторити та оновлювати чергу в режимі реального часу, враховуючи зміни та нагальні випадки.

Сповіщення та нагадування: Система має забезпечувати можливість автоматичного сповіщення пацієнтів про їхній стан у черзі, дату та час прийому, а також нагадування про надходження їхнього призначеного часу.

Безпека даних: Забезпечення безпеки та конфіденційності даних пацієнтів є критичним аспектом. Система повинна використовувати високі стандарти шифрування та забезпечення захисту даних.

Масштабованість: Система повинна бути готовою до масштабування, оскільки обсяги пацієнтів та робочі навантаження можуть змінюватися з часом.

Інтеграція з іншими системами: Система має бути здатною інтегруватися з іншими медичними системами, такими як системи управління пацієнтською інформацією або системи управління лікарськими записами.

Підтримка та оновлення: Важливо мати механізми підтримки користувачів та можливість регулярних оновлень системи для виправлення помилок та додавання нових функцій.

Враховуючи ці вимоги, компанії розробники можуть створити автоматизовану систему управління електронною чергою, яка буде відповідати потребам медичного закладу та його користувачів [9-11].

2.2. Особливості підключення та налаштування бази даних

Один з найважливіших аспектів при розробці автоматизованої системи управління електронною чергою в медичному закладі – це вибір та налаштування бази даних для зберігання інформації про черги. Вибір правильної бази даних є критичним, оскільки це визначає продуктивність, безпеку та доступність даних в системі. Існують певні аспекти, які слід враховувати при виборі типу бази даних:

Необхідність чіткого визначення вимоги до продуктивності – обсягу даних та завдань, які система має обробляти. Для великих медичних закладів, де потрібно обробляти велику кількість запитів, можливо, варто розглядати використання реляційних баз даних, таких як MySQL, PostgreSQL або Microsoft SQL Server, з високою продуктивністю [12].

Враховання можливої складності структури даних, адже деякі реляційні бази даних можуть бути кращими для даних з складною структурою, тоді як NoSQL бази даних, такі як MongoDB або Cassandra, можуть бути кращими для неструктурованих даних.

Забезпечення даних. Обрана база даних повинна надавати механізми шифрування даних та контролю доступу.

Можливість масштабування бази даних у разі зростання обсягу даних або навантаження на систему, а також можливість планування завантаження та резервного копіювання. Деякі NoSQL бази даних мають добру масштабованість для горизонтального розширення.

Комплексність запитів визначає знову ж таки структуру зберігання даних та керування продуктивністю.

І ще один не важливий момент – вартість власності, тобто вартість ліцензування та підтримки для бази даних, що можуть впливати на загальні витрати на проект.

Вибір типу бази даних є важливим рішенням, і він повинен враховувати конкретні потреби та вимоги медичного закладу. Правильно налаштована та оптимізована база даних гарантує ефективну роботу системи управління

електронною чергою та забезпечує зручний доступ до інформації для всіх зацікавлених сторін [13].

Ефективне проектування схеми бази даних є ключовим кроком у розробці автоматизованої системи управління електронною чергою в медичному закладі. Оптимальна структура бази даних дозволяє забезпечити швидкий доступ до інформації, зменшити обсяги зберігання та зробити систему більш масштабованою. При проектуванні бази даних слід керуватись наступними принципами:

1. Таблиці та відносини: заздалегідь визначте таблиці та відносини між ними. Наприклад, можливі таблиці для зберігання інформації про пацієнтів, лікарів, прийоми та інші релевантні дані.

2. Ідентифікатори: Визначте унікальні ідентифікатори для записів у кожній таблиці, які дозволяють однозначно ідентифікувати дані.

3. Нормалізація: Використовуйте принципи нормалізації, щоб уникнути дублювання даних та забезпечити цілісність інформації.

4. Індеси: Визначте поля для створення індексів, які полегшують пошук та сортування даних.

5. Зовнішні ключі: Використовуйте зовнішні ключі для зв'язків між таблицями та забезпечення цілісності даних.

6. Типи даних: Вибирайте відповідні типи даних для кожного поля, щоб економити місце та забезпечувати правильну обробку даних.

7. Денормалізація (при потребі): У деяких випадках, де швидкий доступ до даних є пріоритетом, може бути використана денормалізація для зберігання даних у вигляді готових до використання наборів.

8. Безпека даних: Враховуйте механізми шифрування та аутентифікації для захисту конфіденційності та цілісності даних.

9. Резервне копіювання: Плануйте регулярне резервне копіювання та відновлення даних для запобігання втраті інформації.

10. Масштабованість: Проектуйте базу даних так, щоб вона була готовою до масштабування з плином часу та зростанням обсягів даних [14].

У контексті розробки автоматизованої системи управління електронною чергою в медичному закладі, однією з ключових задач є забезпечення цілісності та надійності даних. Для досягнення цих цілей, використання транзакцій є необхідним принципом бази даних.

Транзакція – це логічно об'єднана послідовність одного або декількох операцій бази даних, які вважаються атомарними, тобто вони виконуються повністю або не виконуються взагалі. Важливо, щоб транзакції були атомарними, консистентними, ізольованими та додали недостовірність (ACID), що гарантує цілісність даних та запобігає конфліктам та перезапису даних.

Приклад ситуації, коли використання транзакцій важливе, це коли два користувачі спробують зареєструватися на один і той же прийом або провести операцію одночасно. Без використання транзакцій це може призвести до конфлікту, перезапису або втрати даних. Однак завдяки транзакціям, система здатна виявити конфліктні операції і вирішити їх, забезпечуючи цілісність даних.

Крім того, використання транзакцій може забезпечити відмінний рівень ізоляції, що означає, що операції одного користувача не впливають на операції інших користувачів, і кожен користувач бачить систему в стабільному стані.

В результаті використання транзакцій, система управління електронною чергою в медичному закладі може забезпечити надійну та безпечну роботу з даними, уникнути конфліктів та перезапису інформації, а також забезпечити відмінний рівень обслуговування для пацієнтів та медичного персоналу [15].

У контексті розробки автоматизованої системи управління електронною чергою в медичному закладі важливо розглянути рівні ізоляції даних, які визначають, як операції з даними ізолюються одна від одної в багатокористувацькому середовищі. Оптимальний вибір рівня ізоляції допоможе забезпечити правильну роботу системи та уникнути конфліктів даних.

В таблиці 2.1 представлено інформацію про рівні ізоляції в базах даних.

Таблиця 2.1 – Рівні ізоляції в базах даних

Рівень ізоляції	Брудне читання	Неповторне читання	Фантомне читання
Read Uncommitted	+	+	+
Read Committed	-	+	+
Read Committed з використанням версіонування рядків	-	+	+
Repeatable Read	-	-	+
Snapshot	-	-	-
Serializable	-	-	-

Read Uncommitted (Незакомічені читання): На цьому рівні ізоляції дозволяється читати незакомічені дані, тобто дані, які можуть бути змінені іншим користувачем і не фіксовані в базі даних. Цей рівень надає високу продуктивність, але може виникнути проблема "брудного" читання, коли користувач бачить незакомічені дані.

Read Committed (Читання з комітом): На цьому рівні дозволяється читати тільки закомічені дані. Це усуває проблему "брудного" читання, але може призвести до інших конфліктів, таких як проблема читання "фантомів", коли дані можуть змінюватися між запитамі.

Repeatable Read (Повторне читання): На цьому рівні ізоляції забороняється зміна даних, які читаються в поточній транзакції. Це усуває проблему читання "фантомів", але може виникнути проблема блокування ресурсів іншими користувачами.

Serializable (Серіалізована): На цьому рівні ізоляції забороняється будь-яка зміна даних, які читаються в поточній транзакції, і уникнути будь-яких конфліктів. Це надає найвищий рівень ізоляції, але може призвести до блокування транзакцій та зниження продуктивності.

Для системи управління чергою в медичному закладі важливо враховувати баланс між продуктивністю та цілісністю даних. Зазвичай, рівень ізоляції "Read Committed" вважається оптимальним для більшості сценаріїв, оскільки він усуває проблему "брудного" читання та дозволяє ефективно працювати з даними без надмірного блокування. Однак, в конкретних випадках, якщо цілісність даних має вирішальне значення, розгляд рівня "Serializable" може бути виправданим.

Вибір рівня ізоляції повинен враховувати конкретні вимоги та потреби конкретного медичного закладу. Оскільки, правильно спроектована схема бази даних становить фундамент для ефективної роботи автоматизованої системи управління електронною чергою в медичному закладі. Вона дозволяє забезпечити точний та швидкий доступ до необхідної інформації, а також полегшує розширення та підтримку системи [16-18].

2.3. Дослідження переваг інтеграції з іншими системами

Можливості інтеграції включають в себе обмін даними, автоматизацію процесів, а також оптимізацію роботи медичного персоналу.

Важливість інтеграції полягає в тому, що вона забезпечує цілісну картину стану пацієнта, оскільки медичний персонал може отримати доступ до повної інформації про пацієнта, що дозволяє приймати інформовані рішення та покращує якість медичної допомоги.

Здатна також мінімізувати помилки та збільшити ефективність роботи медичного закладу, а також впливає на задоволеність пацієнтів.

Інтеграція з системами інформаційних технологій медичних закладів стає стратегічним кроком для покращення медичного обслуговування та забезпечення більшої ефективності у роботі медичного закладу. На рисунку 2.1 представлено можливі варіанти інтеграції між собою медичних інформаційних та систем автоматизованого управління.



Рисунок 2.1 - Варіанти інтеграції між собою медичних інформаційних та систем автоматизованого управління

Процеси обміну даними включають в себе наступні аспекти:

1. Збір та відправка даних: Автоматизована система управління чергою повинна здійснювати збір необхідних даних, таких як інформація про пацієнтів та їх призначені прийоми, і відправляти ці дані іншим системам.
2. Стандартизація даних: Для успішної інтеграції важливо використовувати стандартизовані формати даних, щоб забезпечити зрозумілість та спільну мову між системами.
3. Синхронізація та поновлення: Дані мають бути синхронізовані між системами в режимі реального часу або відповідно до заданих інтервалів для забезпечення актуальності інформації.
4. Моніторинг обміну даними: Слід встановити механізми моніторингу для виявлення та вирішення можливих проблем обміну даними.

5. **Захист даних:** Для забезпечення безпеки даних слід використовувати шифрування та інші методи захисту при передачі і зберіганні.

6. **Автоматичне сповіщення про помилки:** Система повинна мати можливість автоматично сповіщати про помилки обміну даними та надавати засоби для їх вирішення [19-21].

Процеси обміну даними між системами є важливою частиною інтеграції та дозволяють автоматизованій системі управління чергою та іншим системам медичного закладу працювати як єдина ефективна одиниця. Ця інтеграція покращує продуктивність та якість медичного обслуговування, спрощує процеси та підвищує задоволеність пацієнтів та медичного персоналу.

2.4. Дослідження автоматизованого оповіщення та розсилка інформації

Розробка системи автоматичних сповіщень для пацієнтів про стан черги спрямована на полегшення їхнього досвіду та забезпечення інформаційного комфорту. Ця система дозволяє автоматично надсилати пацієнтам повідомлення про їхнє місце в черзі, очікуваний час прийому та іншу важливу інформацію. Основні переваги системи включають:

1. зменшення незручностей та тривалого очікування;
2. підвищення зручності та швидка відповідь;
3. збільшення точності;
4. оптимізація робочих процесів.

Розробка системи автоматичних сповіщень підвищує якість обслуговування пацієнтів та допомагає медичним закладам створити ефективний та зручний досвід для всіх користувачів системи управління чергою.

Однією з ключових складових системи автоматичних сповіщень для пацієнтів про стан черги є використання різних каналів зв'язку. Це включає в себе відправку SMS-повідомлень, email-листів та можливість взаємодії через інші засоби комунікації.

SMS є дуже ефективним каналом сповіщення через свою швидкість доставки та високий рівень доступності. Пацієнти отримують короткі та зрозумілі повідомлення про свій стан у черзі та призначений час прийому.

Email надає більше можливостей для передачі деталізованої інформації. Пацієнти можуть отримувати розгорнуті повідомлення, які містять не лише інформацію про чергу, але і додаткові вказівки, наприклад, щодо підготовки до прийому.

А вже залежно від специфіки медичного закладу та вподобань пацієнтів, можуть використовуватися інші канали зв'язку, такі як месенджери чи спеціалізовані додатки. Це дозволяє забезпечити мультиплатформений доступ до інформації та враховувати індивідуальні потреби користувачів [22].

Використання різних каналів зв'язку забезпечує гнучкість та комфорт для пацієнтів, дозволяючи їм обирати той, який найкраще відповідає їхнім потребам. Це сприяє підвищенню ефективності сповіщень та забезпеченню швидкого та надійного інформування.

Запити на захист особистих даних є критичним аспектом при розробці системи автоматичних сповіщень для пацієнтів про стан черги. Забезпечення конфіденційності та безпеки особистої інформації пацієнтів є важливим завданням, і це можна досягти за допомогою різних технологічних заходів та практик.

1. Шифрування даних: Всі особисті дані, що передаються через різні канали зв'язку, повинні бути шифровані, щоб захистити їх від несанкціонованого доступу. Використання сучасних шифрувальних алгоритмів (наприклад, SSL/TLS для email або SMS) дозволяє забезпечити надійний захист даних пацієнтів.

2. Аутентифікація та авторизація: Перед надсиланням особистої інформації пацієнтам, система повинна забезпечити їх аутентифікацію та авторизацію. Тільки уповноважені користувачі повинні мати доступ до особистих даних. Використання механізмів аутентифікації, таких як двофакторна аутентифікація, забезпечує вищий рівень безпеки.

3. Моніторинг та аудит: Система повинна мати механізми моніторингу та аудиту дій користувачів. Це дозволяє виявити та реагувати на незвичайну активність, а також вести журнали подій для подальшого аналізу.

4. Дотримання законодавства: Для забезпечення захисту особистих даних, система повинна дотримуватися відповідних законодавчих вимог, таких як Загальний регламент про захист персональних даних (GDPR) чи HIPAA (Health Insurance Portability and Accountability Act, що стосується медичної інформації).

Загальна мета полягає в тому, щоб забезпечити, щоб особисті дані пацієнтів залишаються в безпеці та конфіденційності під час всього процесу розсилки інформації. Спільна робота з технологічними засобами, такими як шифрування, аутентифікація та дотримання законодавства, гарантує надійний захист особистих даних та забезпечує довіру користувачів системи [23].

2.5. Відслідковування та аналіз даних

Моніторинг продуктивності є необхідною складовою системи управління чергою, оскільки він допомагає забезпечити стабільну роботу та вчасну реакцію на проблеми. Регулярний аналіз даних про продуктивність дозволяє покращувати систему та надавати найкращий досвід для користувачів.

Технічні деталі моніторингу продуктивності включають:

1. Системи журналювання: Реєстрація подій та дій, що відбуваються в системі, допомагає виявити аномалії та проблеми. Сучасні системи моніторингу використовують журнали подій, які можуть бути проаналізовані для виявлення несправностей.

2. Метрики продуктивності: Вимірювання часу відгуку системи, завантаження серверів, кількість запитів і транзакцій дозволяють оцінити продуктивність системи. Інструменти моніторингу автоматично збирають ці метрики та надають аналіз.

3. Сповіщення про проблеми: Якщо система виявляє аномалії або проблеми продуктивності, вона може автоматично надсилати сповіщення адміністраторам. Це дозволяє реагувати на проблеми швидко та уникнути відключень.

4. Автоматизоване виявлення проблем: Використання алгоритмів машинного навчання та штучного інтелекту дозволяє передбачити проблеми та запропонувати оптимальні рішення для їх вирішення [24].

З технічної точки зору збір та аналіз статистичних даних включають:

1. Безпосередньо збір даних, коли система автоматично збирає дані про черги, час очікування, час обслуговування, інформацію про пацієнтів та лікарів. Ці дані фіксуються у реальному часі та зберігаються у спеціальній базі даних.

2. Етап коли, зібрані дані аналізуються та візуалізуються за допомогою спеціалізованих інструментів. Графіки, діаграми та звіти допомагають адміністраторам та керівництву медичного закладу бачити актуальну статистику та тренди.

3. Етап прогнозування та оптимізації, коли на основі аналізу даних можна розробити алгоритми прогнозування і оптимізації роботи черг. Це дозволяє максимально використовувати ресурси, зменшити час очікування та покращити обслуговування пацієнтів.

4. Етап реагування на несправності, що допомагає вчасно виявляти проблеми та несправності в роботі системи чергування. Це дозволяє швидко реагувати та виправляти недоліки.

5. Постійне покращення системи на основі аналітики є ключовим для забезпечення найвищого рівня обслуговування пацієнтів у медичному закладі. Аналіз даних допомагає ідентифікувати можливості для оптимізації та покращення, що в свою чергу сприяє підвищенню ефективності та задоволення користувачів системи [25].

2.6. Аналіз необхідності технічної підтримки та безпеки системи

Забезпечення стабільної роботи та безпеки системи черги в медичному закладі є важливою задачею для забезпечення якості обслуговування пацієнтів та захисту їхніх особистих даних. Технічна підтримка та заходи з кібербезпеки

допомагають уникнути потенційних проблем і забезпечують безперебійну роботу системи. Тому вкрай необхідним стає постійний моніторинг та механізми оперативного реагування на несправності в роботі системи.

До того ж, регулярні оновлення та покращення програмного забезпечення дозволяють виправляти вразливості та забезпечує сумісність з останніми технологічними стандартами.

Захист від втрати даних здійснюється завдяки системі резервного копіювання, що дозволяє відновлювати дані в разі аварії або видалення. Використання сучасних методів кібербезпеки, таких як шифрування, автентифікація та багатошарова захист, захищає систему від кіберзагроз.

Наявність служби технічної підтримки допомагає вирішувати технічні проблеми та відповідати на запити користувачів. Наявність цілодобової служби технічної підтримки не є обов'язковою, а скоріше вирішується в залежності від масштабів медичного закладу та добового навантаження на автоматизовану систему.

Існують також засоби для забезпечення стабільності серверу та транзакційності бази даних, що включають в себе:

1. Моніторинг серверу: Постійний моніторинг стану серверу дозволяє вчасно виявляти незвичайні навантаження, витоки пам'яті, аномальне використання ресурсів, інші проблеми, які можуть впливати на його стабільність.
2. Автоматизовану систему реагування: У випадку виявлення проблем або збоїв, система має автоматичні механізми реагування, які намагаються відновити нормальну роботу серверу без втрати даних та обслуговування пацієнтів.
3. Резервне копіювання бази даних: Регулярні резервні копії бази даних дозволяють відновлювати інформацію в разі її втрати або пошкодження.
4. Заходи безпеки бази даних: Використання методів шифрування та автентифікації допомагає захистити дані в базі від несанкціонованого доступу.

5. Відновлення транзакцій: У випадку виникнення проблем з транзакціями, система має механізми для їх відновлення та уникнення втрати даних.

6. Створення різних рівнів доступу та деякі інші.

Оскільки забезпечення стабільності серверу та транзакційності бази даних є важливим завданням для забезпечення безперебійної роботи системи управління чергою. Технічна підтримка та резервне планування можуть забезпечити ефективність та надійність цієї системи в медичному закладі [26, 27].

В цей же час, заходи забезпечення безпеки даних та обмеження доступу є критичними для захисту особистих даних пацієнтів та забезпечення надійності системи управління чергою в медичному закладі. Вони допомагають уникнути можливих порушень безпеки та зберегти довіру користувачів до системи.

Дослідивши основні аспекти автоматизації системи управління електронною чергою, слід перейти для аналізу існуючих систем аналогів, щоб порівняти їх функціональні можливості, схожість та відмінність між собою, а також відповідність дослідженим пунктам надійної та ефективної автоматизованої системи управління електронною чергою в медичному закладі.

3. ДОСЛІДЖЕННЯ ВЗАЄМОДІЇ МОДУЛЯ УПРАВЛІННЯ ЕЛЕКТРОННОЮ ЧЕРГОЮ З ІНШИМИ МОДУЛЯМИ В АВТОМАТИЗОВАНІЙ СИСТЕМІ УПРАВЛІННЯ МЕДИЧНИМИ ЗАКЛАДАМИ

3.1 . Інтеграція з іншими системами управління медичними закладами

В контексті розвитку медичної сфери важливо визнати велике значення інтеграції систем управління, яка сприяє покращенню надання медичних послуг та оптимізації робочих процесів. Інтеграція модуля управління електронною чергою з іншими системами управління медичними закладами є ключовим аспектом, спрямованим на вирішення сучасних викликів у галузі охорони здоров'я, а саме:

1. Синергія роботи систем

Інтеграція модуля управління електронною чергою з інформаційними системами медичних закладів дозволяє створити силове поле, в якому кожна система доповнює та підтримує іншу. Це сприяє ефективній обробці та передачі даних, зменшує час очікування пацієнтів та поліпшує координацію роботи персоналу.

2. Оптимізація робочих процесів

Інтеграція дозволяє автоматизувати багато аспектів управління медичним закладом, таких як запис на прийом, облік медичної інформації та взаємодія з лабораторіями. Це сприяє оптимізації робочих процесів, що в свою чергу призводить до підвищення продуктивності та зменшення витрат часу.

3. Покращення якості обслуговування пацієнтів

Інтеграція модуля управління електронною чергою з іншими системами сприяє вищому рівню сервісу для пацієнтів. Швидка обробка інформації та автоматизована координація роботи персоналу дозволяють забезпечити ефективне та індивідуальне обслуговування, що важливо для задоволення потреб пацієнтів.

4. Забезпечення стабільності та безпеки

Інтеграція систем управління підвищує стабільність та безпеку медичних закладів. Здатність модуля управління електронною чергою взаємодіяти з іншими системами забезпечує надійність обробки даних та виключає ризик інформаційних збоїв.

Інтеграція з іншими системами управління медичними закладами визначається як крок вперед до високоефективного та сучасного управління медичним процесом. Цей підхід не лише відповідає сучасним стандартам, але й сприяє наданню високоякісної та доступної медичної допомоги.

До того ж, успішна взаємодія цих модулів суттєво впливає на загальну продуктивність систем. В цьому контексті виникає потреба вдосконалення внутрішніх процесів, зокрема оптимізації обміну даними між різними модулями. Розглядаючи вже встановлені зв'язки з іншими системами, важливо розглянути та удосконалити способи, якими ці системи обмінюються інформацією. Такий поглиблений аналіз дозволить ефективно вбудовувати оптимізаційні стратегії вже на етапі проектування інтегрованих систем управління медичними закладами [28].

3.2. Оптимізація обміну даними між модулями

Оптимізація обміну даними між модулями в системах управління медичними закладами є ключовим елементом для забезпечення ефективного та безперебійного функціонування інтегрованих систем. Дана проблематика обумовлена необхідністю узгодженого обміну інформацією для оптимального використання ресурсів та підвищення якості медичного обслуговування.

На рисунку 3.1 зображено послідовність кроків по забезпеченню оптимізованого обміну даними між модулями автоматизованої системи.



Рисунок 3.1 – Послідовність кроків по забезпеченню оптимізованого обміну даними між модулями автоматизованої системи

Першим етапом оптимізації є детальний аналіз обсягу та типів даних, що обмінюються між різними модулями систем управління. Визначення основних категорій даних, таких як медичні записи, графіки прийому пацієнтів та інші параметри, дозволяє ефективно організувати обмін інформацією.

Далі, на основі результатів аналізу визначаються оптимальні протоколи обміну даними. Розробка стандартів та нормативів, які визначають формати даних, методи передачі та узгоджують схему обміну, є необхідним етапом для уніфікації процесу обміну даними та забезпечення сумісності систем.

На наступному етапі оптимізація обміну даними передбачає вдосконалення алгоритмів обробки інформації для швидкого та точного обчислення результатів. Використання ефективних методів кешування, стиснення даних та паралельних обчислень сприяє покращенню продуктивності системи.

Особлива увага приділяється аспектам безпеки обміну даними. Використання шифрування, механізмів аутентифікації та контролю доступу дозволяє попереджати несанкціонований доступ до медичної інформації та зберігати конфіденційність пацієнтів.

Останнім кроком визначено моніторинг та постійне вдосконалення, хоча останнім він є умовно, оскільки оптимізація обміну даними – це постійний процес, який вимагає систематичного моніторингу та аналізу результатів. Регулярне вдосконалення алгоритмів, адаптація до змінних умов та впровадження нових технологій є важливими етапами для забезпечення високої ефективності обміну даними в системах управління медичними закладами.

3.3 Синхронізація роботи модуля зі стандартами безпеки та конфіденційності

В умовах надзвичайної важливості конфіденційності та безпеки медичних даних, питання синхронізації роботи модуля управління електронною чергою з визначеними стандартами безпеки виступає як визначальний аспект впровадження інтегрованих систем управління медичними закладами. На фоні стрімкого розвитку технологій та поглиблення цифрової трансформації в медичній галузі, забезпечення високого рівня конфіденційності та безпеки є необхідністю першочергового значення.

Перший крок у синхронізації роботи модуля полягає в ретельному вивченні та визначенні стандартів безпеки, які визначають вимоги та протоколи для обробки та зберігання медичних даних. Врахування національних та міжнародних нормативів дозволяє розробити адаптовані стратегії безпеки.

Визначення стандартів безпеки передбачає врахування особливостей обробки медичних даних, таких як розпізнавання особливо чутливих даних, які вимагають особливих заходів захисту. При цьому важливо не лише дотримуватися загальноприйнятих нормативів, але й адаптувати їх до специфічних вимог та викликів медичної галузі.

Україна визначає ряд стандартів та законів, що стосуються безпеки і захисту персональних даних. Основні з них включають:

- Закон України "Про захист персональних даних"

Закон визначає правові засади захисту персональних даних та встановлює права та обов'язки суб'єктів даних та операторів, які обробляють ці дані.

- Відомості про стан захисту персональних даних в основних системах обробки інформації

Згідно з Законом України "Про захист персональних даних", оператори повинні регулярно подавати відомості щодо стану захисту персональних даних відповідним органам державної влади.

- Нормативно-правові акти Національної комісії, що здійснює державне регулювання у сферах зв'язку та інформатизації (НКРЗІ)

НКРЗІ визначає нормативно-правові акти, які регулюють захист персональних даних у сфері зв'язку та інформатизації.

– Міжнародні стандарти: GDPR

Хоча GDPR (Загальний регламент з охорони даних) не є безпосередньо обов'язковим для України, враховуючи її стратегічні плани наближення до європейських стандартів, багато компаній дотримуються цих стандартів для забезпечення високого рівня захисту персональних даних.

Ці стандарти та закони складають основну рамку для забезпечення безпеки та конфіденційності персональних даних в Україні.

Крім того, визначення стандартів безпеки передбачає вивчення та врахування технічних аспектів, таких як шифрування та механізми аутентифікації. Важливим етапом є вибір оптимальних криптографічних методів, які забезпечать не тільки високий рівень захисту, але й збережуть високу продуктивність системи обробки даних.

У процесі визначення стандартів безпеки враховуються та аналізуються вимоги до архітектури системи та методів забезпечення розділення прав доступу. Це включає визначення рівнів доступу до медичних даних для різних категорій користувачів, що допомагає уникнути несанкціонованого доступу та забезпечити принцип "потрібен-знав".

Остаточо, визначення стандартів безпеки – це взаємодія вимог юридичного, технічного та етичного плану, що спрямована на створення надійного та ефективного механізму захисту медичної інформації. Цей підхід вирішує найважливіші виклики, пов'язані із забезпеченням приватності та безпеки даних, і дозволяє вдосконалено впроваджувати інтегровані системи управління в сфері охорони здоров'я [30].

У контексті забезпечення високого рівня безпеки та конфіденційності в системах управління медичними закладами, імплементація шифрування та механізмів аутентифікації стає особливо невід'ємною частиною стратегії захисту інформації та повинна здійснюватись комплексно, включаючи заходи

для забезпечення фізичної безпеки, визначення способів ідентифікації та аутентифікації, розмежування доступів та використання методів для шифрування та протоколювання, тощо. Структурність згаданих вище етапів захисту зображено на рисунку 3.2.

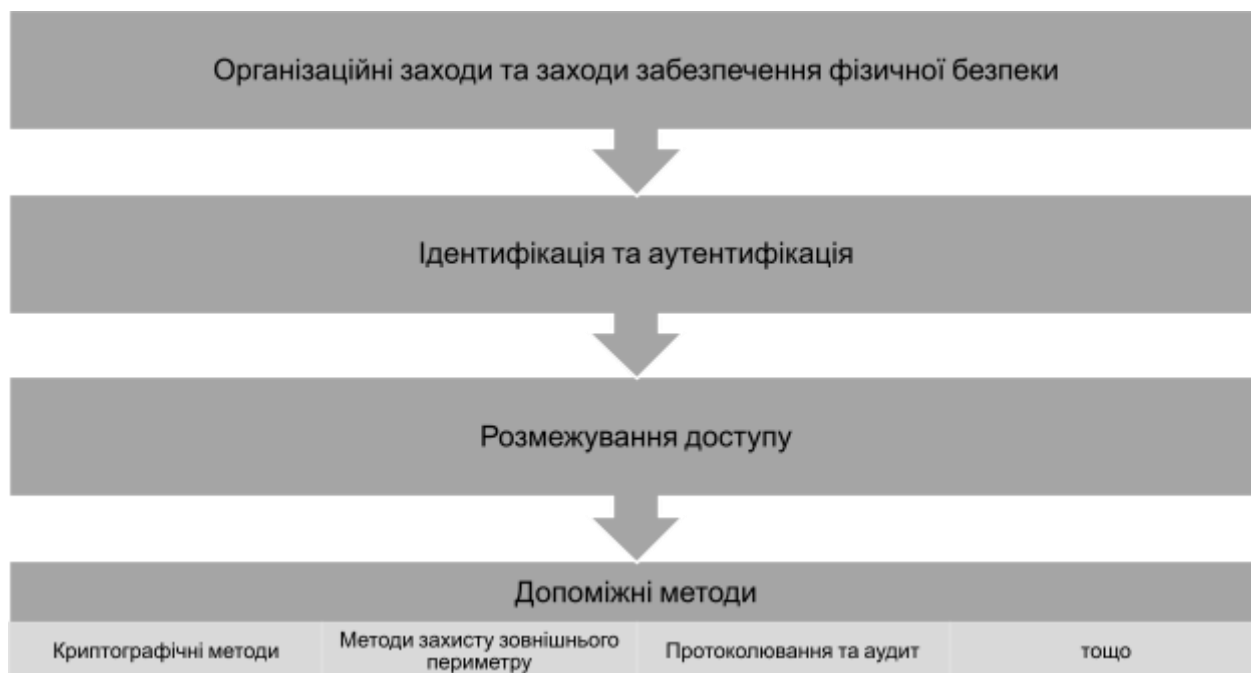


Рисунок 3.2 – Структура заходів забезпечення високого рівня безпеки та конфіденційності в автоматизованих системах управління

Імплементація шифрування відіграє ключову роль у збереженні конфіденційності персональних медичних даних. Застосування сучасних алгоритмів шифрування, таких як AES (Advanced Encryption Standard) або RSA (Rivest–Shamir–Adleman), дозволяє ефективно захищати дані від несанкціонованого доступу під час їхньої передачі та зберігання. Технології шифрування можуть бути інтегровані в різні рівні системи, забезпечуючи безпеку на різних етапах обробки інформації.

Окрім шифрування, механізми аутентифікації є також важливою складовою в системах управління медичними закладами, оскільки вони визначають, хто має доступ до конфіденційної інформації. Двофакторна аутентифікація, біометричні методи (відбиток пальця, розпізнавання обличчя),

або використання електронних карт можуть бути ефективними механізмами для перевірки особи та підтвердження її права доступу до системи.

У контексті розробки веб-додатків для автоматизованих систем управління медичними закладами можна використовувати різноманітні технології та програмні засоби для забезпечення безпеки та конфіденційності. Наприклад, використання HTTPS (SSL/TLS) для зашифрованого обміну даними між клієнтом та сервером, використання фреймворків з вбудованими механізмами безпеки (наприклад, Spring Security для Java або Django для Python), а також регулярне оновлення програмних компонентів для запобігання використанню застарілих, потенційно вразливих версій.

Додатково, важливо розглядати використання систем контролю версій, таких як Git, для ефективного відстеження та керування змінами в програмному коді. Вони дозволяють розробникам спільно працювати над проектом, відокремлюючи різні функціональності та забезпечуючи легкість виявлення та виправлення можливих помилок безпеки.

Доцільно також впроваджувати механізми моніторингу заходів безпеки, такі як системи реєстрації подій та аналізатори журналів, які дозволяють оперативно виявляти потенційні загрози та вчасно реагувати на них. Використання систем інтегрованого моніторингу, таких як Prometheus чи ELK Stack (Elasticsearch, Logstash, Kibana), сприяє збору, аналізу та візуалізації даних щодо безпеки в реальному часі.

Крім того, рекомендується вивчення та впровадження принципів DevSecOps, де аспекти безпеки вбудовуються в усі етапи розробки та експлуатації веб-додатків. Це сприяє виробленню культури безпеки в команді розробників та забезпечує попередження вразливостей на ранніх стадіях проекту [31].

Усі ці заходи взаємодіють, створюючи надійний бар'єр захисту веб-додатків у сфері управління медичними закладами, що забезпечує не лише конфіденційність та цілісність даних, але й стійкість до сучасних кіберзагроз.

Розробка ефективних політик доступу та їхній подальший моніторинг в системах управління медичними закладами грають критичну роль у забезпеченні безпеки та конфіденційності оброблюваних даних. Оптимально спроектовані політики доступу визначають, які користувачі мають доступ до конкретних ресурсів та як цей доступ може бути здійснений.

Розпочинаючи з розробки політик доступу, важливо визначити різні рівні доступу для різних користувачів та груп. Це може включати в себе визначення рівнів доступу для медичного персоналу, адміністраторів та інших сторін, залежно від їхніх функцій та обов'язків. Чітко визначені політики також враховують принцип "потрібен-знав", щоб уникнути надмірного надання прав доступу.

Ефективні політики доступу мають супроводжуватися системами моніторингу, що дозволяють в реальному часі відстежувати та аналізувати використання ресурсів та прав доступу. Це може включати аудитування входжень та виходжень користувачів, виявлення можливих аномалій у поведінці, та автоматизовані системи сповіщення для оперативного реагування на події, що вимагають уваги.

Впровадження систем IAM дозволяє централізовано управляти правами доступу користувачів та контролювати їхню ідентичність. Такі системи надають можливість автоматизованого керування доступом, змінювати права користувачів в реальному часі та ефективно впроваджувати принципи найменших привілеїв.

Політики доступу повинні бути динамічно адаптовані до змін у структурі організації та її вимог. Регулярне оновлення політик, їхнє переглядання та адаптація до нових викликів є важливою складовою ефективного управління доступом.

Враховуючи ці аспекти, розробка та моніторинг політик доступу стають надійною основою для забезпечення не тільки безпеки та конфіденційності, але й ефективного управління інформаційними ресурсами у медичних закладах [32].

Швидкі та постійні зміни в кіберзагрозах та технологічних стандартах вимагають систематичного підходу до удосконалення заходів безпеки для забезпечення високого рівня захисту вразливих даних.

Починаючи з контингентної оцінки ризиків, важливо систематично аналізувати нові загрози та слабкі місця в існуючих заходах безпеки. Це дозволяє визначити потенційні точки вразливості та вживати превентивних заходів перед можливими атаками.

В той час як регулярні аудити безпеки є ефективним засобом виявлення вразливостей та перевірки відповідності існуючих заходів безпеки з вимогами законодавства та стандартів. Вони дозволяють виявляти аномалії та проблеми безпеки, що виникають з часом, та приймати необхідні заходи для їх виправлення.

Удосконалення заходів безпеки включає ефективну кіберполітику, що передбачає навчання персоналу щодо найсучасніших методів та стратегій безпеки. Це може бути вирішено через проведення тренінгів та навчань з безпеки, які підвищують обізнаність персоналу із сучасними загрозами та методами їх попередження.

Використання штучного інтелекту та машинного навчання дозволяє автоматизувати процес виявлення та відповіді на потенційні загрози. Системи, що використовують ці технології, можуть надавати швидке реагування на виявлені аномалії та пристосовуватись до нових викликів у реальному часі.

Сучасний підхід до безпеки полягає в адаптивній стратегії, яка враховує нові технологічні та кіберзагрози. Застосування технологій, які забезпечують автоматичне виявлення, відгук та адаптацію до загроз, грає важливу роль у забезпеченні стійкості та відповідності систем управління медичними закладами.

Усі ці заходи разом створюють динамічний інфраструктурний підхід до безпеки, що дозволяє системам управління медичними закладами постійно

адаптуватись до зростаючих загроз та залишатись стійкими в умовах швидкозмінюючого кіберсередовища [33].

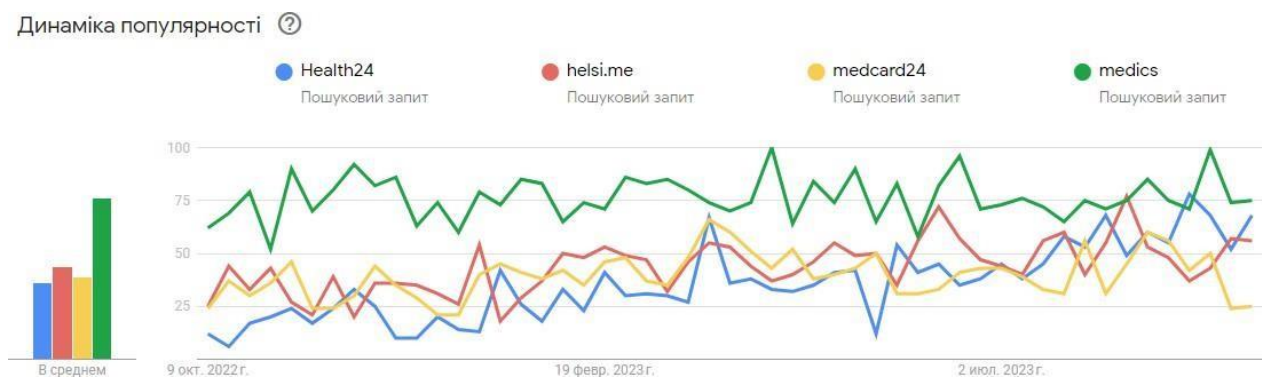
У даному розділі розглянуті ключові аспекти, що визначають ефективність та безпеку взаємодії між різними модулями. Інтеграція з іншими системами управління виявилася критичною для забезпечення синергії та оптимізації процесів в медичних закладах. Оптимізація обміну даними та синхронізація роботи зі стандартами безпеки та конфіденційності забезпечують високий рівень захисту інформації.

Вивчення визначення стандартів безпеки, імплементація шифрування та механізмів аутентифікації, а також розробка політик доступу та моніторингу визначають стратегії захисту конфіденційної інформації та контролю доступу користувачів. Постійне вдосконалення заходів безпеки є необхідною умовою для адаптації до швидко змінюючогося кіберсередовища та забезпечення стабільного та безпечного функціонування автоматизованих систем управління медичними закладами. Усі ці аспекти об'єднуються для створення комплексної та ефективної інфраструктури, що відповідає високим стандартам безпеки в сфері охорони здоров'я.

4. ДОСЛІДЖЕННЯ РОБОТИ ТА ПОРІВНЯЛЬНИЙ АНАЛІЗ ІСНУЮЧИХ СИСТЕМ УПРАВЛІННЯ ЕЛЕКТРОННОЮ ЧЕРГОЮ МЕДИЧНОГО ЗАКЛАДУ

Щоб порівняти функціональні можливості існуючих систем управління електронною чергою, їх схожість та відмінність між собою, а також відповідність дослідженим пунктам надійної та ефективної автоматизованої системи управління електронною чергою в медичному закладі, було обрано 4 найбільш популярні сервіси, що спеціалізуються на співпраці з медичними закладами та наданню інформації про лікарів та години їх прийому – Health24, Helsi.me, Medcard24, а також Medics.

Якщо подивитись на статистику по запитах в Google Trends, що представлена на рисунках 4.1 та 4.2, то очевидним стає, що є лідер серед цих сервісів, проте інші три не відстають в кількості пошуків в мережі Інтернет. До того ж прослідковується деяка географія використання сервісів.



Рисунко 4.1 – Динаміка популярності сервісів, що порівнюються, за останній рік (жовтень 2022 – вересень 2023)

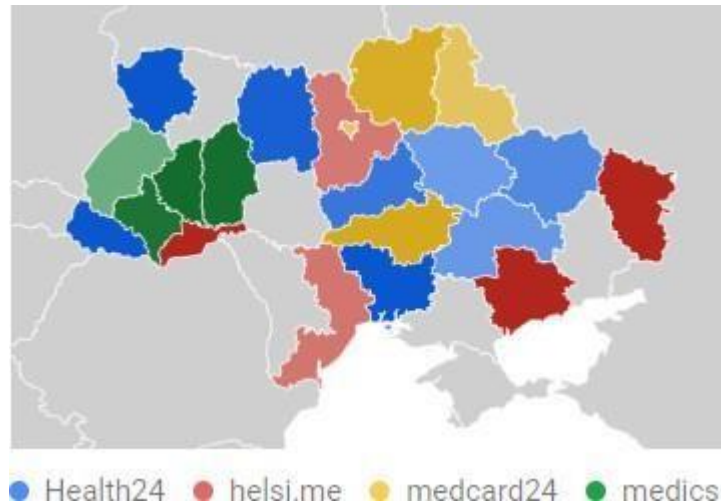


Рисунок 4.2 – Порівняння популярності сервісів у відповідних регіонах країни (жовтень 2022 – вересень 2023)

Всі 4 сервіси працюють схожим чином. Хоча можна виокремити два основних варіанти роботи з сервісом.

Перший варіант взаємодії можна описати наступними кроками. Для користування електронною реєстратурою необхідно спершу зареєструватися на веб-сайті та активувати свій особистий профіль шляхом підтвердження номера телефону. Після успішного входу до особистого кабінету пацієнта, пацієнт отримує можливість автоматичного заповнення декларації з вибраним сімейним лікарем. Після чого залишається тільки записатися на прийом до лікаря для офіційного підпису декларації. Після підписання декларації ваш лікар активує ваш особистий профіль на платформі, відкриваючи доступ до вашої медичної інформації, такої як направлення, вакцинаційний графік, номер декларації, електронні рецепти та інформацію про прийоми у різних спеціалістів.

Другий варіант дещо спрощений – у пацієнта є змога записатись на прийом до лікаря не прив'язуючись до наявності чи відсутності оформленої декларації та/або направлення до якогось спеціаліста. Таким чином пацієнт швидко може зарезервувати вільний час лікаря, витрачаючи мінімум часу на взаємодію з сервісом [34].

4.1. Огляд сервісу Health24

Health 24 - це інтернет-портал, спеціалізований на пошуку клінік і лікарів, який надає широкий спектр медичних послуг та інформації для пацієнтів. Портал дозволяє знайти лікаря будь-якої спеціалізації та зареєструватися на прийом до нього, зменшуючи тимчасові витрати та виключаючи потребу в чергах лікарень чи клінік.

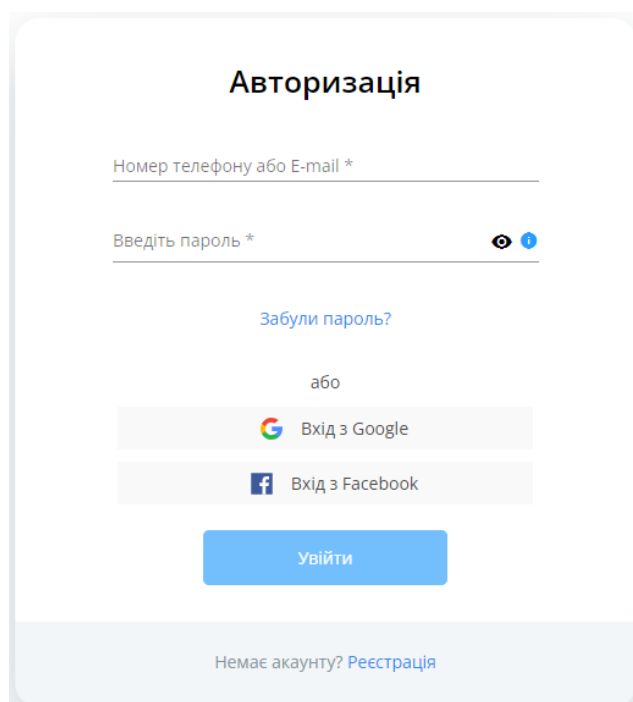
Основні можливості та особливості Health 24:

1. Зручний пошук: Пацієнти можуть швидко знайти лікаря за допомогою вибору конкретної медичної спеціалізації. Сайт надає інформацію про різні клініки та лікарів по всій Україні, що полегшує пошук.

2. Інформація про клініки: Health 24 надає інформацію про найбільш надійні та перевірені клініки, їхні контактні дані та розташування.

3. Вартість послуг: Пацієнти можуть порівнювати вартість прийому лікаря в різних медичних установах та знаходити доступне та якісне лікування.

На рисунку 4.3-4.6 представлено деякі функціональні можливості роботи з сервісом Health24.



The image shows a login form titled "Авторизація" (Authorization). It contains the following elements:

- A text input field labeled "Номер телефону або E-mail *".
- A password input field labeled "Введіть пароль *", which includes an eye icon for toggling visibility and a small blue circle with an 'i' icon.
- A blue link labeled "Забули пароль?" (Forgot password?).
- The word "або" (or) centered below the link.
- A button with the Google logo and the text "Вхід з Google" (Sign in with Google).
- A button with the Facebook logo and the text "Вхід з Facebook" (Sign in with Facebook).
- A prominent blue button labeled "Увійти" (Log in).
- At the bottom, a link labeled "Немає акаунту? Реєстрація" (Don't have an account? Register).

Рисунок 4.3 – Можливі варіанти для реєстрації та логування на сервісі Health24

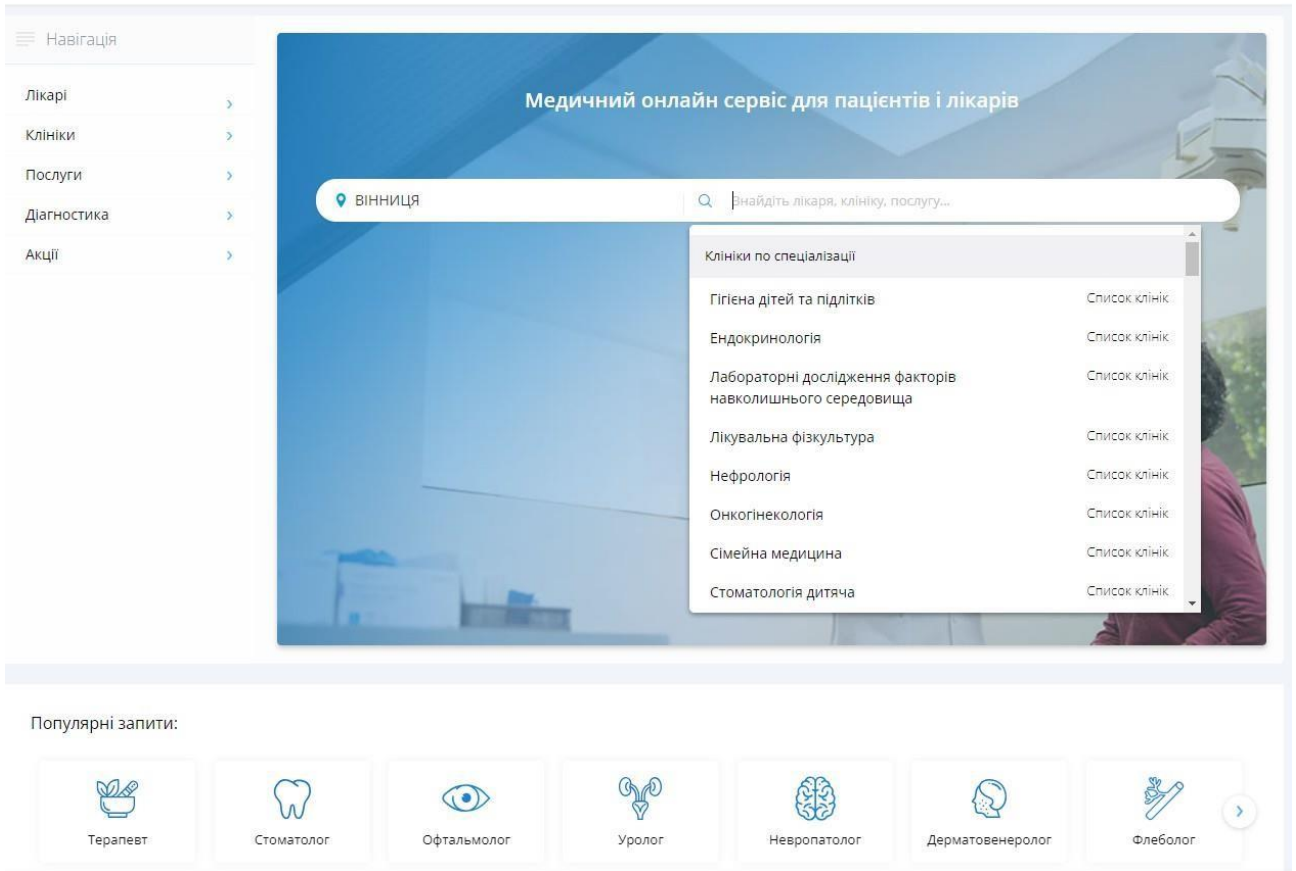


Рисунок 4.4 – Пошук клінік та лікарів на сервісі Health24

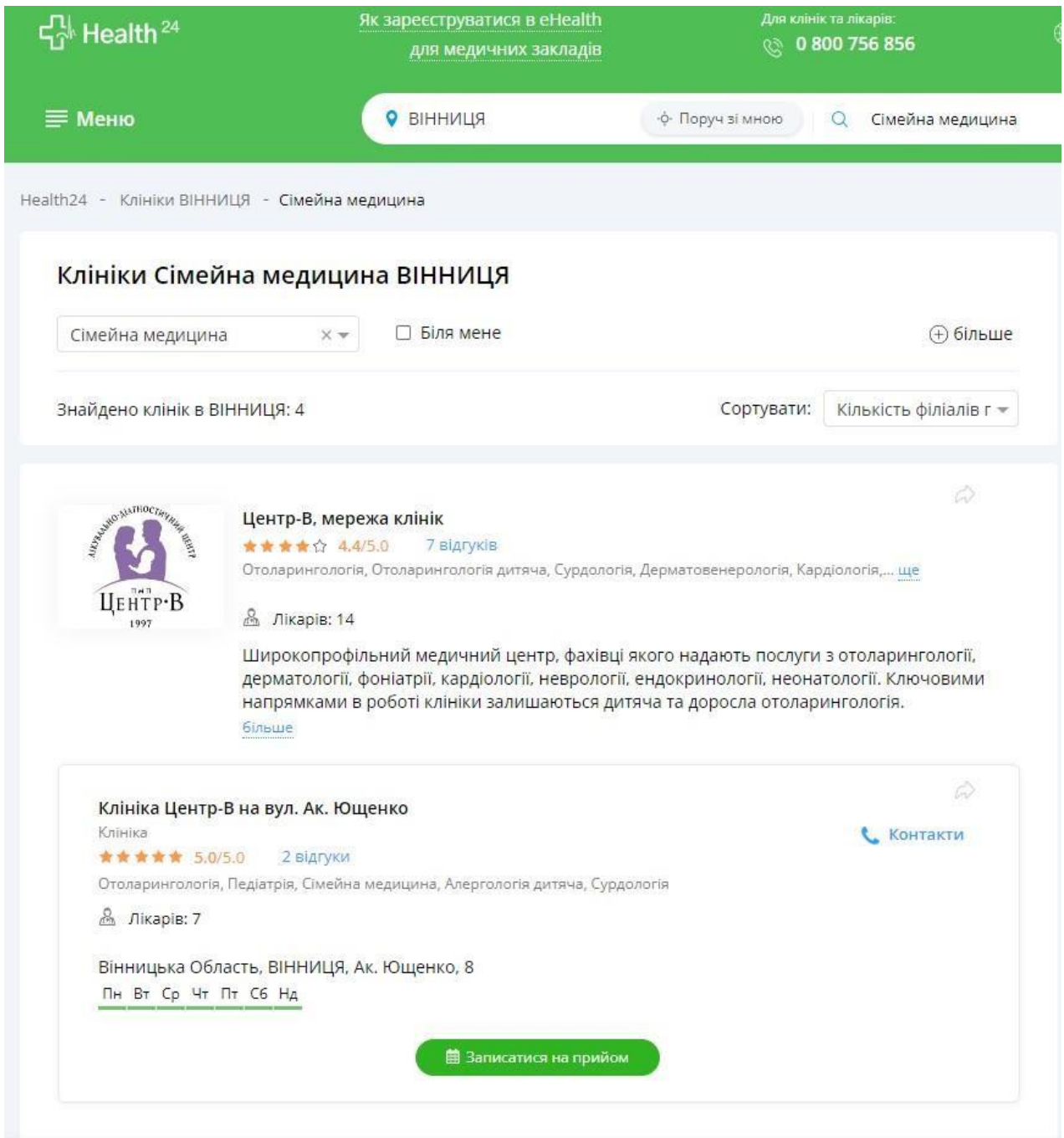


Рисунок 4.5 – Вибір клініки на сервісі Health24

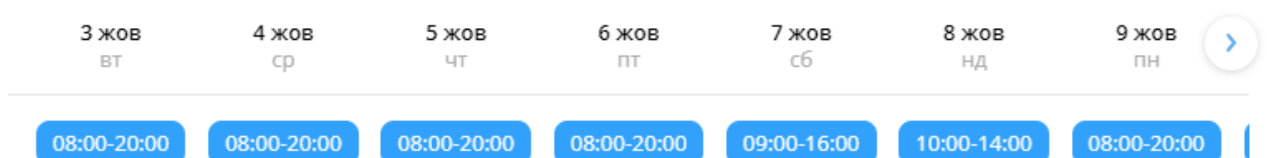


Рисунок 4.6 – Можливість вибору вільного до резервації часу

4.2. Огляд сервісу helsi.me

Helsi.me - це передова, надійна та зручна електронна медична система, розроблена для всіх гілок медичного сектору, включаючи пацієнтів, лікарів і медичні заклади, як державні, так і приватні.

Для пацієнтів надається:

1. зручний вибір лікаря;
2. швидкий онлайн запис;
3. доступ до електронної медичної картки (ЕМК);
4. миттєві результати аналізів та діагностики в режимі реального часу;
5. керування призначеннями лікаря і плану лікування.

Для лікарів:

1. зручне ведення медичних записів: лікарі можуть легко вести історію хвороби пацієнтів та ЕМК;
2. швидкий доступ до результатів діагностики і аналізів;
3. використання клінічних протоколів: зручні інструменти для надання якісної медичної допомоги;
4. зручний кабінет для прийому пацієнтів.

Для медичних закладів:

1. комплексна автоматизація роботи;
2. настроювана система: програмне забезпечення може бути адаптоване під конкретні потреби вашої установи;
3. підтримка реформи: допомагає учасникам медичної реформи в Україні, зокрема в закріпленні пацієнтів за лікарями і реєстрації декларацій;
4. моніторинг і управлінська статистика: дані для керівництва і аналізу;
5. звітність і статистика: інструменти для аналізу та ведення обліку.

На рисунку 4.7-4.10 представлено деякі функціональні можливості роботи з сервісом helsi.me.

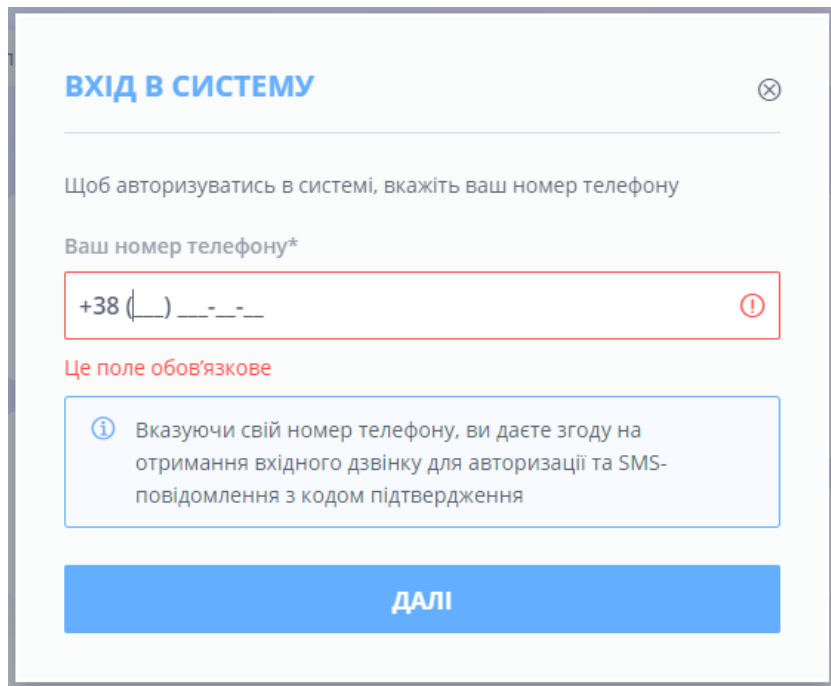


Рисунок 4.7 - Можливі варіанти для реєстрації та логування на сервісі hel.si.me

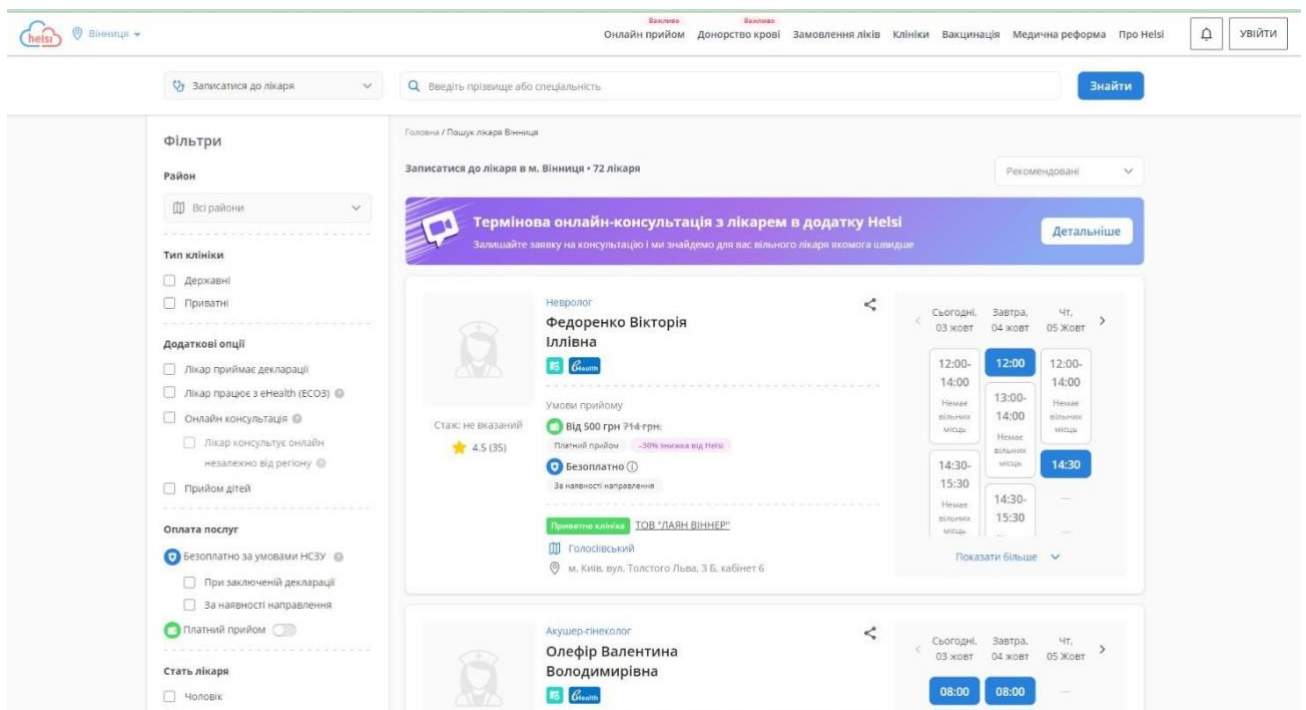


Рисунок 4.8 – Вибір лікарів на сервісі hel.si.me

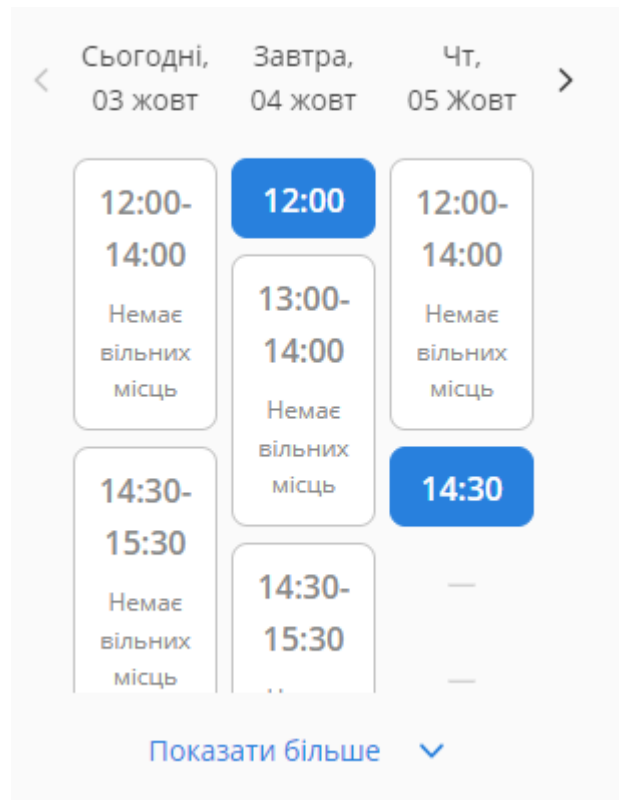


Рисунок 4.9 – Відображення вільних термінів на сервісі helsi.me

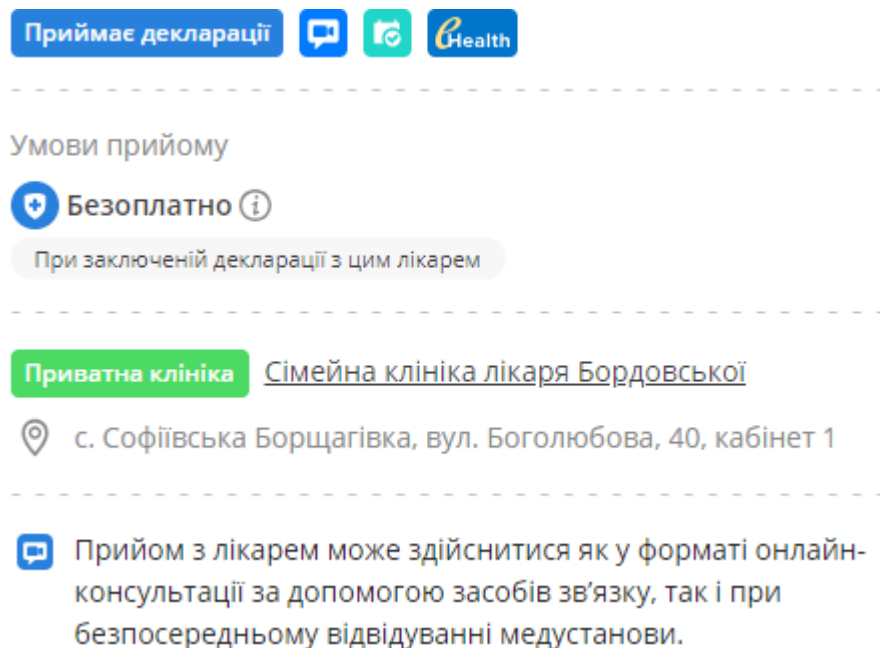


Рисунок 4.10 – Зручне відображення деталей про умови роботи лікаря на сервісі helsi.me

4.3. Огляд сервісу MedCard24

MedCard24 - це інноваційна медична система, яка спрощує доступ до медичних послуг та забезпечує комфортне обслуговування пацієнтів з будь-якого куточку.

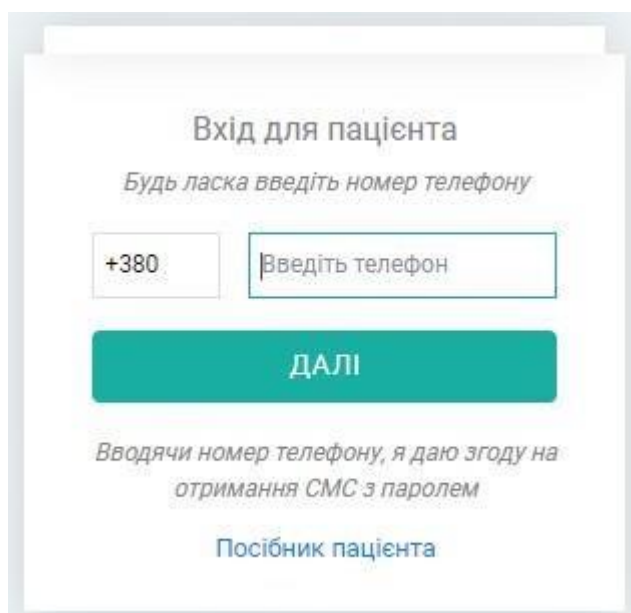
Основні можливості MedCard24:

1. виклик в один клік;
2. швидке скасування;
3. реєстрація 24/7;
4. рецепти, які не можна загубити;
5. захищена особиста інформація.

Переваги MedCard24:

1. зручність: інтуїтивно зрозумілий інтерфейс для всіх користувачів;
2. безкоштовність: використання MedCard24 абсолютно безкоштовне;
3. захист даних;
4. надійність.

На рисунку 4.11-4.14 представлено деякі функціональні можливості роботи з сервісом MedCard24.



Вхід для пацієнта

Будь ласка введіть номер телефону

+380 Введіть телефон

ДАЛІ

Вводячи номер телефону, я даю згоду на отримання СМС з паролем

Посібник пацієнта

Рисунок 4.11 – Можливі варіанти для реєстрації та логування на сервісі MedCard24

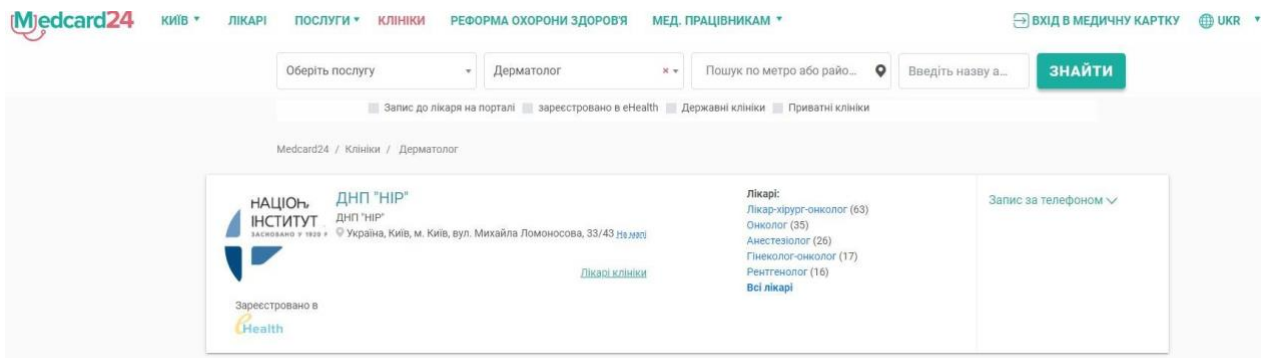


Рисунок 4.12 – Пошук клініки та лікарів на сервісі MedCard24

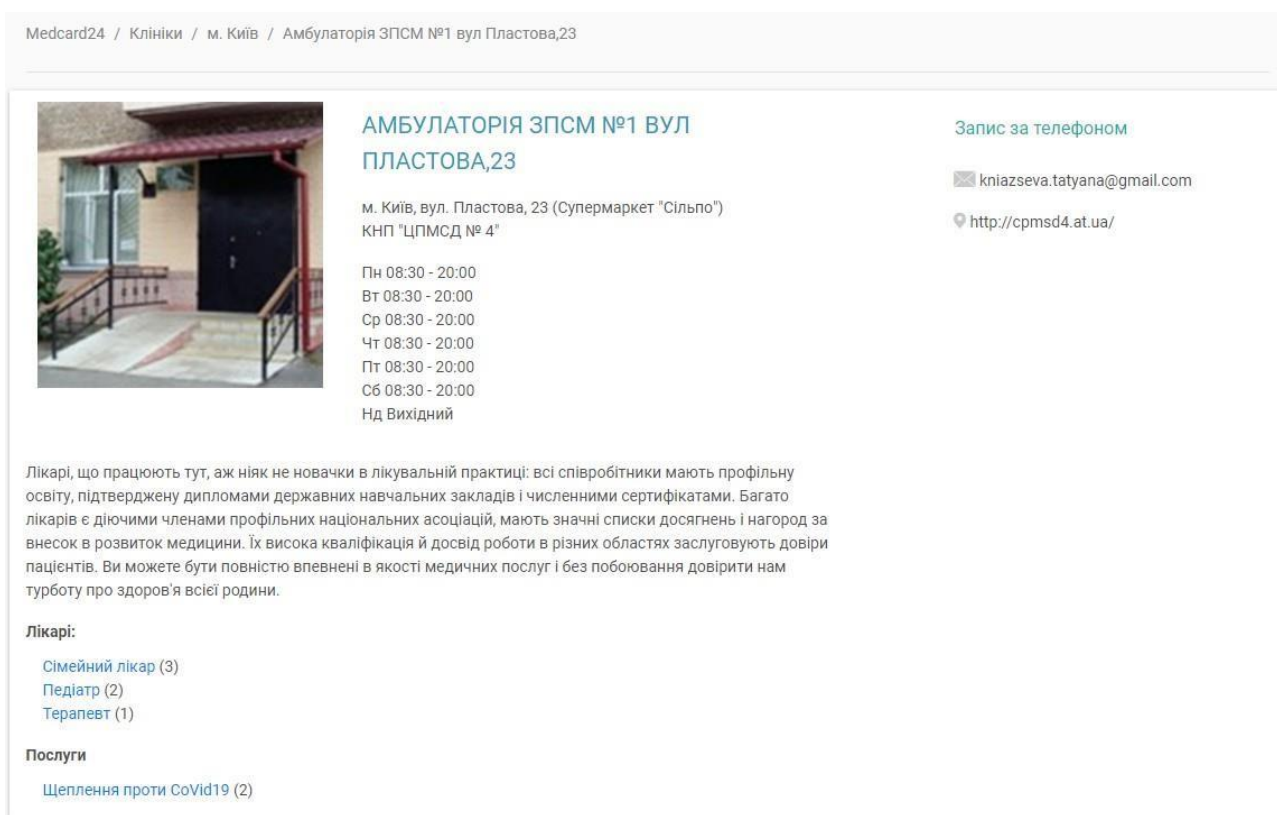


Рисунок 4.13 – Зручне відображення контактної інформації про медичний заклад на сервісі MedCard24

The screenshot displays the 'Запис на прийом' (Appointment) screen in the MedCard24 app. At the top, a teal header contains the title. Below it, the user's name 'КНП "ЦПМСД № 4"' is shown. The clinic name 'Амбулаторія ЗПСМ №1 вул Пластова,23' is displayed in blue. A location pin icon is followed by the address 'м. Київ, вул. Пластова , 23 (Супермаркет "Сільпо")'. A teal button labeled 'Запис за телефоном' with a downward arrow is positioned below the address. The main form area has a light teal background and includes a date picker set to '09.10.2023', a time picker set to '09:45', and a text input field for the phone number. A large teal button at the bottom is labeled 'Записатись на прийом'.

Рисунок 4.14 – Форма запису на прийом на сервісі MedCard24

4.4. Огляд сервісу Medics

Medics - це інноваційна система, яка спрощує доступ до медичних послуг та забезпечує зручність взаємодії між пацієнтами, лікарями і медичними закладами.

Основні Можливості Medics:

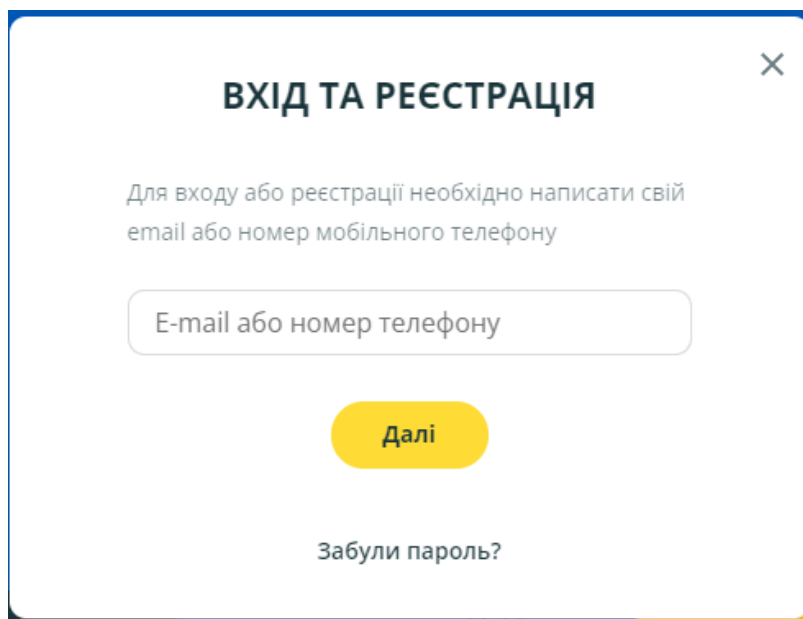
1. медична картка: вся необхідна інформація про прийоми, обстеження та лікарські засоби завжди під рукою;
2. пошук лікарів і медичних закладів: зручний пошук допомагає швидко знайти лікаря або медичний заклад, які потрібні, навіть з урахуванням місцезнаходження;

3. рейтинг лікарів, що базується на відгуках пацієнтів;
4. онлайн запис;
5. сімейний доступ: можливість слідкувати за рекомендаціями лікаря для дітей та підопічних, маючи доступ до їхньої медичної інформації;
6. сповіщення.

Medics співпрацює з реформою eHealth, щоб покращити медичну систему і зробити її більш доступною та зручною для всіх.

Для медичних закладів та лікарів Medics пропонує комплексну автоматизацію роботи медичних закладів та надає лікарям інструменти для зручного ведення пацієнтської інформації. А для пацієнтів Medics робить медичну картку доступною, зручною та оновленою, надаючи вам контроль над вашим здоров'ям.

На рисунку 4.15-4.18 представлено деякі функціональні можливості роботи з сервісом Medics.



ВХІД ТА РЕЄСТРАЦІЯ

Для входу або реєстрації необхідно написати свій email або номер мобільного телефону

E-mail або номер телефону

Далі

[Забули пароль?](#)

Рисунок 4.15 – Можливі варіанти для реєстрації та логування на сервісі Medics


ПОШУК ЛІКАРІВ

Всі спеціальності Розширений пошук

Тип: День:

Місто: Стать:

Клініка:



Островська Ангеліна Олександрівна
 Акушер-гінеколог
 ★★★★★ 116

Акушер-гінеколог Акушер-гінеколог	Приватна поліклініка "Малютко Плюс" Приватна	Тривалість 10 хв.	<input type="button" value="Запису немає"/>
Акушер-гінеколог лікар акушер-гінеколог	місце надання послуг 1 Державна	Тривалість 30 хв.	<input type="button" value="Запису немає"/>

Рисунок 4.16 – Пошук лікарів та клінік на сервісі Medics

Педіатр
Лікар

Амбулаторія загальної практики сімейної медицини №6
Державна


Тривалість
15 хв.

15:00 - 15:15 Завтра	15:15 - 15:30 Завтра	15:45 - 16:00 Завтра	16:00 - 16:15 Завтра	16:15 - 16:30 Завтра	16:45 - 17:00 Завтра	<input type="button" value="Запис на інший час >"/>
-------------------------	-------------------------	-------------------------	-------------------------	-------------------------	-------------------------	--


Рисунок 4.17 – Форма запису до лікаря на сервісі Medics

Інформація Відгуки


Відгуки пацієнтів



Олександра К.
Дуже приємний лікар!
13:26, 02.10.2023



Світлана В.
Лікар , дуже сподобалася. Компетентна і чуйна, прийомом задоволена . Хотілось би більше таких лікарів;)
14:35, 26.09.2023



Аліна .
Чудовий лікар , завжди на зв'язку , вічлива , турбується про пацієнтів ☺
17:12, 23.09.2023

[Всі відгуки](#)

Місце роботи

- місце надання послуг 3**
Акушер-гінеколог
- проспект Миру, 61, Хмельницький
- (0382) 67-16-65
- ТОВАРИСТВО З ОБМЕЖЕНОЮ ВІДПОВІДАЛЬНІСТЮ "БОМЕД"**
Акушер-гінеколог
- Зарічанська, 11Е, Хмельницький

Рисунок 4.18 – Відображення коментарів пацієнтів та місця роботи лікаря на сервісі Medics

Розглянуті чотири сервіси - Health 24, HELSI.ME, MedCard24 та Medics - мають багато спільних функціональних можливостей, що роблять їх дуже подібними в забезпеченні медичних послуг та комунікації між пацієнтами, лікарями і медичними закладами.

Вони всі надають можливість пацієнтам знайти та записатися до лікаря онлайн, швидко отримати доступ до своєї медичної інформації, включаючи результати обстежень та прийоми. Також у них є функціональність для пошуку лікарів та медичних закладів, а також можливість переглядати рейтинг лікарів на основі відгуків пацієнтів.

Кожен з цих сервісів прагне полегшити доступ до медичних послуг та підвищити комфорт пацієнтів, а також спростити роботу лікарів та медичних закладів. Вони також активно впроваджують інноваційні технології для покращення якості медичної допомоги та забезпечення безпеки даних пацієнтів.

Порівняльна характеристика даних сервісів представлена в зведеній таблиці 4.1.

Таблиця 4.1 – Порівняльна характеристика розглянутих сервісів Health 24, Helsi.me, MedCard24 та Medics

Критерій	Назва сервісу			
	Health 24	Helsi.me	MedCard24	medics
Зрозумілий інтерфейс	+	+	+	+
Можливість швидкої реєстрації на прийом	+	+	+	+
Медична картка	+	+	+	+
Інтеграція з іншими сервісами (бронювання ліків, запис на донорство крові і тп)	-	+	+	-
Коментарі від користувачів / пацієнтів	-	-	+	+
Коментарі від лікарів / замітки / оновлення	-	-	-	+
Зручне та вичерпне відображення контактної інформації про медичні заклади	-	+	+	+
Мобільний застосунок	-	+	+	+
Безпечне зберігання персональних даних	+	+	+	+
Плата за користування	-	-	-	-

Дослідивши та проаналізувавши роботу існуючих систем управління електронною чергою медичного закладу, слід перейти до проектування власної розробки в межах магістерської кваліфікаційної роботи, розглянути варіанти використання, розробити структуру бази даних, спроектувати та розгорнути локально власний сервіс для керування електронною чергою.

5. ПРОЕКТУВАННЯ ТА РОЗРОБКА АВТОМАТИЗОВАНОЇ СИСТЕМИ УПРАВЛІННЯ ЕЛЕКТРОННОЮ ЧЕРГОЮ МЕДИЧНОГО ЗАКЛАДУ

5.1 Проектування автоматизованої системи управління

Проектування автоматизованої системи є одним із найважливіших етапів у розробці будь-якого програмного продукту. Цей процес передбачає створення деталізованого плану системи та визначення її архітектури, функціональності та взаємодії між компонентами. Правильно спроектована система стає основою для успішної розробки та подальшої ефективної роботи [35-37].

Етапи проектування автоматизованої системи:

1. Аналіз вимог: Спочатку визначаються всі вимоги до системи. Це включає в себе розуміння потреб користувачів, функціональні та нефункціональні вимоги, а також обмеження.

2. Створення концепції: На цьому етапі розробляється загальна ідея системи, її мета та завдання. Визначається, як система вирішує вимоги та сприяє досягненню поставлених цілей.

3. Проектування архітектури: На цьому етапі визначаються компоненти системи, їх структура та взаємозв'язки. У разі автоматизованої системи управління чергою в медичному закладі, це може бути розробка архітектури сервера, клієнтського веб-інтерфейсу та бази даних.

4. Розробка UML-діаграм: UML (Unified Modeling Language) діаграми є потужним інструментом для візуалізації структури та функціональності системи. Наприклад, діаграми варіантів використання можуть бути використані для моделювання взаємодії користувачів із системою, а діаграми класів - для проектування структури коду.

5. Проектування структури бази даних: У контексті системи управління чергою в медичному закладі, структура бази даних грає важливу роль. Вона включає в себе таблиці для зберігання інформації про пацієнтів, лікарів, черги, назначень, тощо.

Правильно спроектована база даних допомагає забезпечити ефективну роботу системи та забезпечує надійне зберігання даних.

6. Вибір технологій: На цьому етапі вибираються мови програмування, фреймворки та інші технології, які будуть використовуватися для розробки системи. У контексті веб-системи це можуть бути мови програмування, такі як Python або JavaScript, та веб-фреймворки, такі як Django або React.

7. Створення плану розробки: На основі всіх попередніх етапів розробляється план реалізації системи. Цей план включає в себе розклад роботи, розподіл завдань між розробниками, терміни та контрольні точки.

Проектування системи є ключовим кроком у створенні автоматизованої системи управління чергою в медичному закладі. Вірно виконані UML-діаграми та структура бази даних гарантують ефективну реалізацію та надійну роботу системи [38].

Створення Use Case UML-діаграми

На рисунку 4.1 представлено Use Case UML-діаграму для автоматизованої системи управління електронною чергою медичного закладу, на якій зображено два актора: пацієнта та адміністратора, які на високому рівні деталізації мають майже однакові варіанти взаємодії з системою. Наприклад:

1. Вхід в систему: Перший варіант використання - це процес, коли зареєстрований користувач входить в систему, вводячи свої облікові дані. У цьому випадку, система перевіряє ідентифікаційні дані та надає доступ до особистого кабінету.

2. Управління персональною інформацією акаунта.

3. Управління записами на прийом: Пацієнт може обрати лікаря, дату та час прийому через веб-інтерфейс системи. Після цього, система перевіряє доступність обраного лікаря та часу та реєструє користувача на прийом. Адміністратор у свою чергу має доступ до

4. Інформації і бронювань пацієнтів, має можливість відмінити або перенести бронювання, створити його самостійно.

5. Перегляд історії відвідувань: Пацієнт може переглядати свою історію відвідувань лікарів, результати обстежень та призначення через особистий кабінет. Схожий функціонал присутній і для користувача в ролі адміністратор.

6. Пошук лікаря та клініки: Пацієнт та адміністратор можуть шукати лікарів та медичні заклади за різними параметрами, такими як спеціалізація, рейтинг, розташування тощо.

7. Управління повідомленнями: Система також здатна надсилати повідомлення користувачам, наприклад, нагадування про надходження часу прийому. Пацієнт в даному випадку отримує повідомлення, а адміністратор має можливість налаштувати канал зв'язку, частоту та вміст повідомлень.

8. Управління статистикою: Варіант використання доступний користувачам в ролі адміністратор, адже відображає дані усіх бронювань пацієнтів та лікарів.

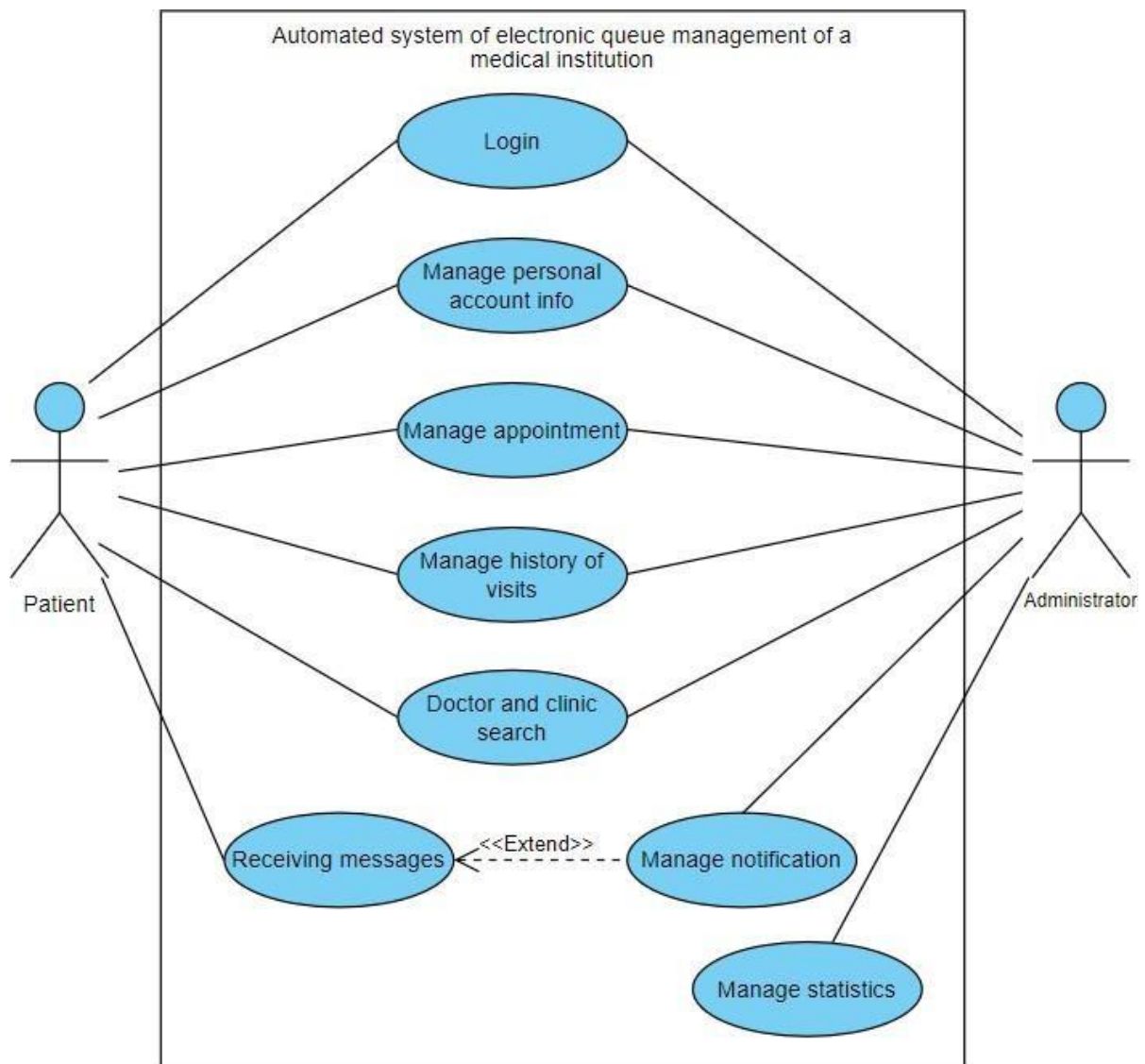


Рисунок 5.1 – Use Case UML-діаграма для автоматизованої системи управління електронною чергою медичного закладу

Використання UML-діаграми варіантів використання допомагає не лише концептуалізувати та деталізувати всі можливі шляхи взаємодії користувачів із системою, але й надає візуальне представлення цих варіантів, що робить інформацію більш доступною та зрозумілою для всіх учасників проекту, включаючи розробників, замовників, менеджмент та інших зацікавлених сторін. Графічне зображення варіантів використання спрощує аналіз, виявлення потреб у функціональності та визначення оптимальних рішень для покращення функціональності системи [39-41].

Створення UML-діаграми класів

На рисунку 5.2 представлено UML-діаграму класів для автоматизованої системи управління електронною чергою медичного закладу. Діаграма класів визначає структуру програмного забезпечення та встановлює основні взаємодії між його класами, що робить її невід'ємною частиною процесу розробки автоматизованих систем управління. Цей інструмент UML надає статичний погляд на систему та забезпечує деталізацію компонентів, які входять в її склад.

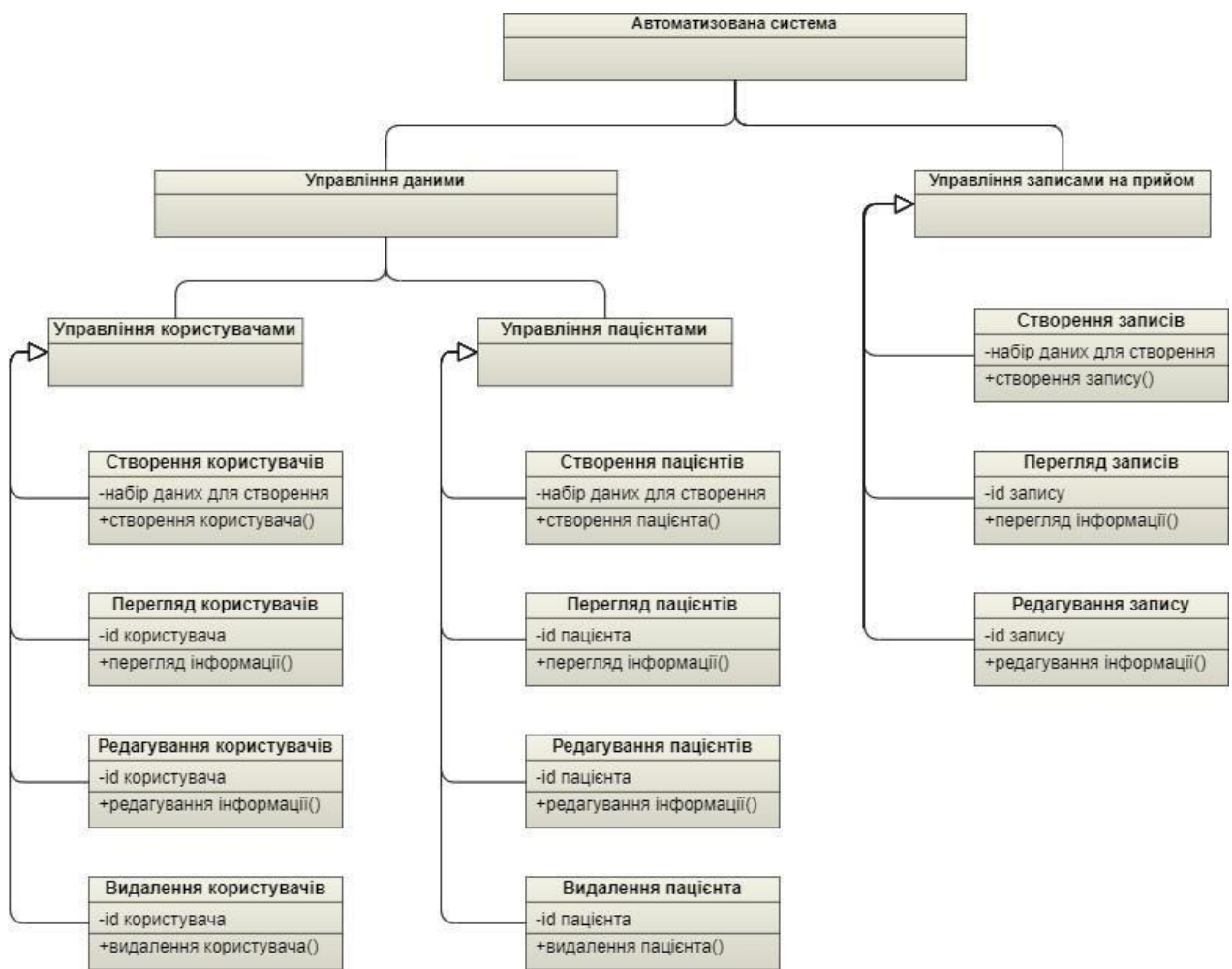


Рисунок 5.2 – UML-діаграма класів для автоматизованої системи управління електронною чергою медичного закладу

Використання діаграм класів має значні переваги при прямому та зворотному проектуванні. Під час прямого проектування, діаграма дозволяє чітко визначити структуру нової системи, зокрема класи, їх атрибути та методи.

При зворотному проектуванні, вона служить для аналізу та опису існуючих систем [42-44].

Елементи діаграми класів включають в себе класи, представлені прямокутниками, які містять ім'я класу, його атрибути та методи. Ця структура надає інформацію про статичну складову системи, визначаючи основні аспекти її функціонування. Взаємозв'язки між класами відображаються різними типами та визначають логічні відносини між об'єктами.

Автоматизована система містить два головних класи: управління даними, а також управління записами на прийом. Кожен з них містить необхідні класи нащадки для управління персональною інформацією користувачів та спеціалістів, а також інформацією про візити.

Створення ER-діаграми

ER-діаграма бази даних стає критичним етапом при проектуванні автоматизованої системи управління електронною чергою медичного закладу. Цей інструмент визначає структуру та взаємозв'язки між сутностями системи, надаючи чітке розуміння потреб у зберіганні та обробці даних.

Необхідність ER-діаграми полягає в тому, щоб визначити ключові сутності, їх атрибути та зв'язки, які визначають логіку взаємодії між різними аспектами медичного закладу. Це дозволяє ефективно моделювати інформаційні потреби системи та гарантує оптимальну організацію даних.

У контексті поточної розробки, використання нормалізованої бази даних для системи управління електронною чергою медичного закладу має визначені переваги для OLTP (Online Transaction Processing) систем. Нормалізація дозволяє уникнути аномалій даних, забезпечуючи їх консистентність, і зменшує дублювання інформації [45-47].

Однією з ключових переваг нормалізованої бази даних є збереження інтегритету даних та відсутність аномалій при вставці, оновленні та видаленні даних. Це стає критичним для системи управління чергою, де точність та актуальність інформації є важливими аспектами. Крім того, нормалізація

дозволяє оптимізувати швидкість доступу до даних, що є ключовим фактором у реальному часі, особливо в умовах медичного закладу.

ER-діаграма, що визначає структуру бази даних для системи управління електронною чергою медичного закладу, включає ключові сутності, такі як Clinics, Specialists, Users та Appointments. В таблицях 5.1-5.4 зображено структури цих сутностей та їх атрибути.

Сутність "Clinics" представляє медичний заклад та визначає його основні характеристики. До складу цієї сутності входять атрибути, такі як назва клініки, адреса, телефон, електронна пошта та інші параметри, які дозволяють ідентифікувати та описати конкретний заклад.

Таблиця 5.1 – Структура сутності "Clinics"

Clinics		
Назва атрибуту	Тип даних	Обмеження
Id	INT	AUTO_INCREMENT PRIMARY KEY
Name	VARCHAR(255)	NOT NULL
Address	VARCHAR(255)	NOT NULL
PhoneNumber	VARCHAR(20)	NOT NULL
DetailedInfo	TEXT	NOT NULL
Email	VARCHAR(255)	NOT NULL

Сутність "Users" визначає користувачів системи, що можуть бути в ролі User або Administrator. До атрибутів цієї сутності входять особисті дані, логін, пароль, а також роль у системі.

Таблиця 5.2 – Структура сутності "Users "

Users		
Назва атрибуту	Тип даних	Обмеження
Id	INT	AUTO_INCREMENT PRIMARY KEY
FullName	VARCHAR(255)	NOT NULL
Email	VARCHAR(255)	UNIQUE NOT NULL
RoleId	INT	NOT NULL
PhoneNumber	VARCHAR(20)	NOT NULL
DateOfBirth	DATE	NOT NULL
City	VARCHAR(100)	
Address	VARCHAR(255)	
Password	VARCHAR(255)	NOT NULL
Comment	TEXT	

Сутність "Specialists" визначає лікарів системи. До атрибутів цієї сутності входять особисті дані та спеціалізація, а також стан лікаря – Active або Not Active.

Таблиця 5.3 – Структура сутності " Specialists "

Specialists		
Назва атрибуту	Тип даних	Обмеження
Id	INT	AUTO_INCREMENT PRIMARY KEY
FullName	VARCHAR(255)	NOT NULL
Specialty	VARCHAR(255)	NOT NULL
Email	VARCHAR(255)	UNIQUE NOT NULL
PhoneNumber	VARCHAR(20)	NOT NULL
Photo	BLOB	
StatusId	INT	NOT NULL

Сутність "Appointments" визначає записи на прийом до лікаря в медичному закладі. Атрибути цієї сутності включають дату та час прийому, статус (заплановано, відмінено, завершено), а також ідентифікатор лікаря та пацієнта.

Таблиця 5.4 – Структура сутності " Appointments "

Appointments		
Назва атрибуту	Тип даних	Обмеження
Id	INT	AUTO_INCREMENT PRIMARY KEY
UserId	INT	NOT NULL
SpecialistId	INT	NOT NULL
DateTimeFrom	DATETIME	NOT NULL
DateTimeTo	DATETIME	NOT NULL
State	INT	NOT NULL
Comment	TEXT	

Взаємозв'язки між цими сутностями визначають логіку взаємодії в системі. Наприклад, взаємозв'язок між "Users" та "Appointments" вказує, що кожен користувач може мати багато записів на прийом, а кожен запис на прийом пов'язаний з конкретним користувачем. Такі взаємозв'язки визначають основні моменти обробки даних та їх взаємозв'язок в рамках системи управління чергою медичного закладу.

Детальна ER-діаграма для автоматизованої системи управління електронною чергою медичного закладу представлена на рисунку 5.3.

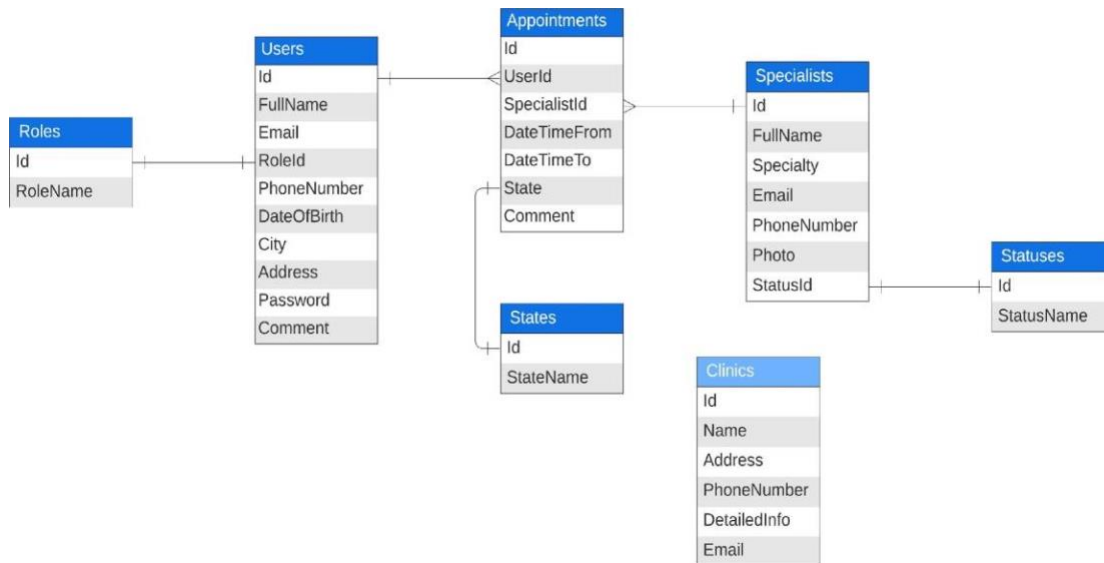


Рисунок 5.3 – ER-діаграма для автоматизованої системи управління електронною чергою медичного закладу

5.2 Опис технічних засобів та мов програмування

Автоматизована система управління електронною чергою медичного закладу є інноваційним веб-додатком, розробленим з використанням передових технічних засобів та сучасних мов програмування, що гарантує ефективну та надійну роботу системи.

У розробці системи використовуються мови програмування високого рівня, такі як Python та JavaScript. Python використовується для серверної частини, забезпечуючи стійку та ефективну обробку запитів, а JavaScript використовується для клієнтської частини, щоб забезпечити динамічні та інтерактивні можливості користувачам.

У якості фреймворків для веб-розробки обрані Django (на основі Python) та React.js (на основі JavaScript). Django дозволяє швидко розробляти стійкі та безпечні серверні компоненти, тоді як React.js забезпечує гнучкість та швидкість в розробці інтерфейсу користувача.

База даних системи управління чергою реалізована з використанням MySQL, що забезпечує ефективне зберігання та оптимальний доступ до даних.

Для оптимізації та забезпечення високої продуктивності використовується система керування базою даних MySQL InnoDB.

Забезпечуючи відмінну масштабованість та надійність, система побудована з використанням контейнеризації за допомогою Docker, що дозволяє легко розгортати та керувати додатком в будь-якому середовищі [48-51].

5.3 Розробка та реалізація автоматизованої системи управління електронною чергою медичного закладу

Для авторизації в системі необхідно ввести електронну пошту та пароль, як зображено на рисунку 5.4.



Рисунок 5.4 – Вікно авторизації в системі

В залежності від ролі користувача, що зареєструвався в системі, набір функціональних можливостей буде відрізнятися. Всього передбачено дві ролі: Administrator та User (Patient).

Функціональні можливості користувачів в ролі User

На рисунку 5.5 зображено вигляд головного екрану автоматизованої системи управління електронною чергою медичного закладу для користувача в ролі User. Даному користувачеві доступні такі вкладки як «Головна», «Наші спеціалісти», «Мій календар», «Історія візитів», а також через іконку в правому верхньому куті знаходиться доступ до персонального акаунту.

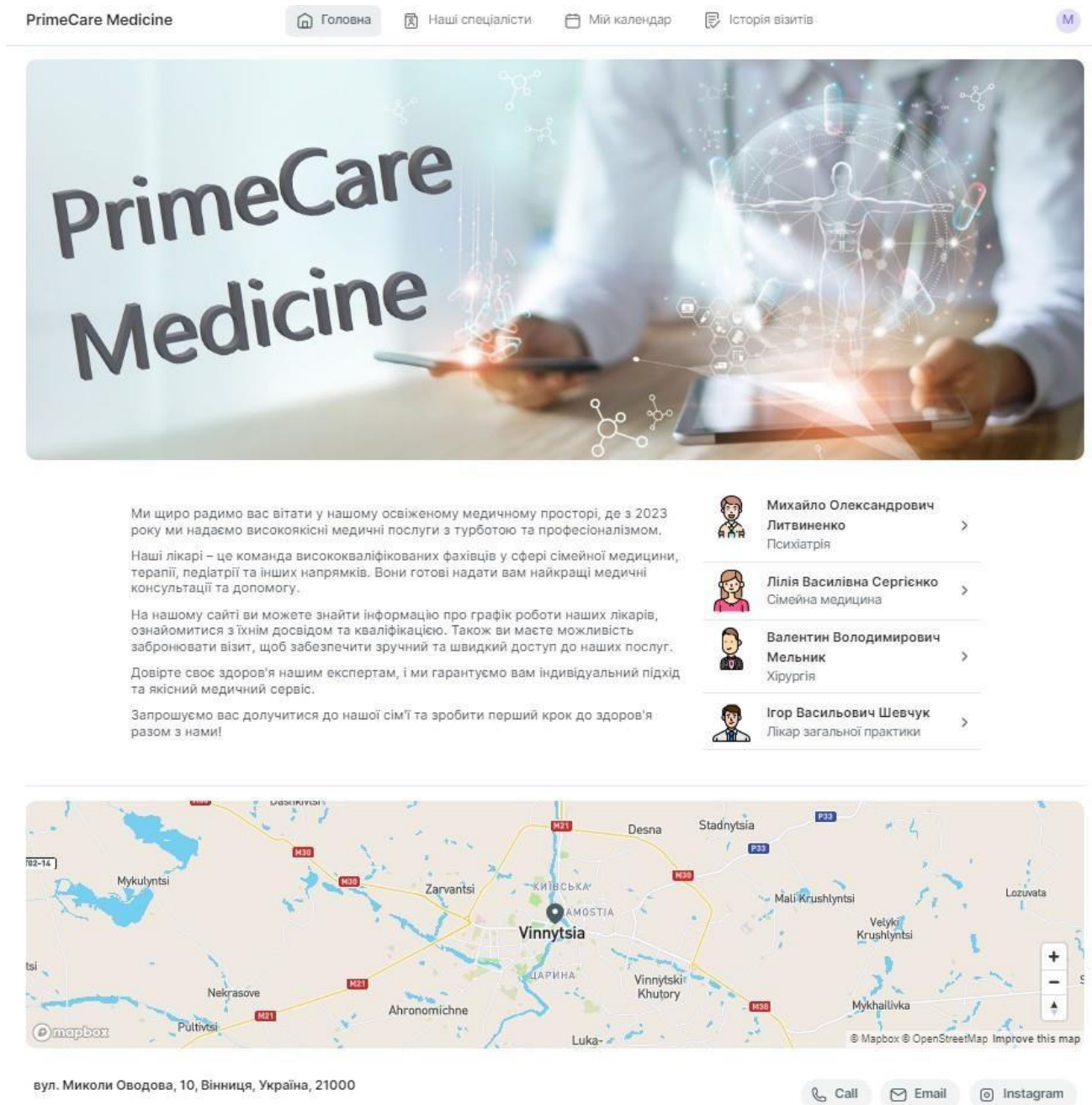


Рисунок 5.5 – Вигляд головного екрану автоматизованої системи управління електронною чергою медичного закладу для користувача в ролі User

У вкладці «Наші спеціалісти», що зображена на рисунку 5.6, відображаються лікарі клініки, що в даний момент мають статус «Active».

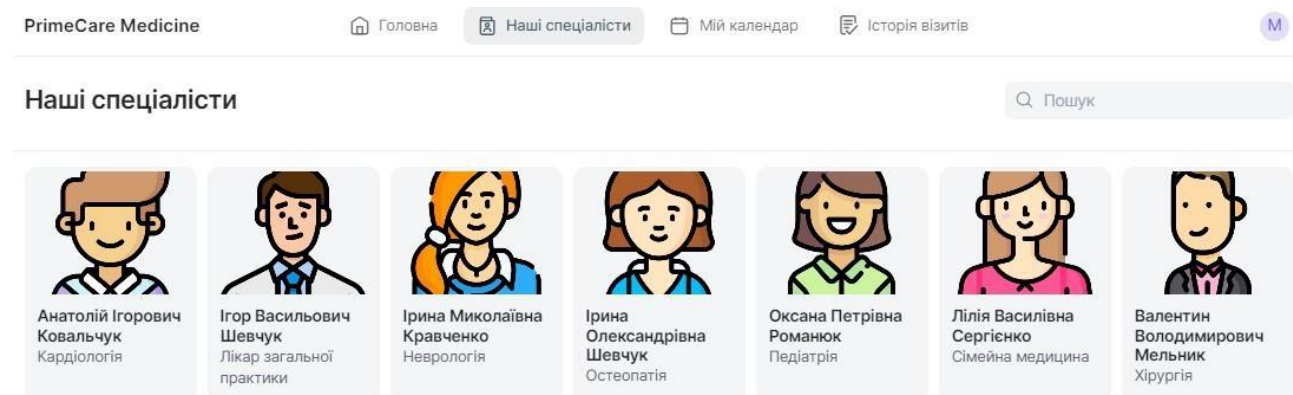


Рисунок 5.6 – Відображення списку активних лікарів у вкладці «Наші спеціалісти»

Натиснувши на будь-якого лікаря, користувач переходить в профіль обраного лікаря, де відображається актуальний календар з прийомом. На даній сторінці можна зробити бронювання візиту, якщо часовий слот має статус «Вільно», або зробити скасування заброньованого візиту, якщо статус слоту «Заброньовано» та ід поточного користувача співпадає з ід користувача візиту, який необхідно скасувати. Тобто користувачу доступні до перегляду усі часові слоти та їх статус, але керувати можна лише власними візитами.

На рисунку 5.7 зображено профіль спеціаліста з одним заброньованим візитом на дату 21 листопада 2023 року. Однаковим кольором позначаються часові слоти з однаковим статусом. В даному випадку «Вільно» та «Заброньовано».

Наші спеціалісти > Оксана Петрівна Романюк



Оксана Петрівна Романюк

Педіатрія

Задзвонити на рецепцію

листоп. 20-е - листоп. 26-е 2023

Тижд... ▾

Сьогодні

<

>

	20 пон	21 вів	22 сер	23 чтв	24 птн	25 суб	26 нед
12 дп							
1 дп							
2 дп							
3 дп							
4 дп							
5 дп							
6 дп							
7 дп							
8 дп							
9 дп							
10 дп			10:00 – 10:30 Вільно				
11 дп		10:30 – 11:00 Заброньовано	11:00 – 11:30 Вільно				
12 дп		11:30 – 12:00 Вільно	12:00 – 12:30 Вільно				
1 пп		12:30 – 13:00 Вільно					
2 пп		13:30 – 14:00 Вільно					

Рисунок 5.7 – Профіль спеціаліста з одним заброньованим візитом на дату 21 листопада 2023 року

В свою чергу, на рисунку 5.8, представлено, як змінився часовий слот на 21 листопада 2023 року на 11:30, а також невелике повідомлення в нижній частині екрану про успішне бронювання візиту. Запис цієї активності в базу даних підхоплює ід користувача та лікаря, передаючи їх в таблицю з бронюваннями.

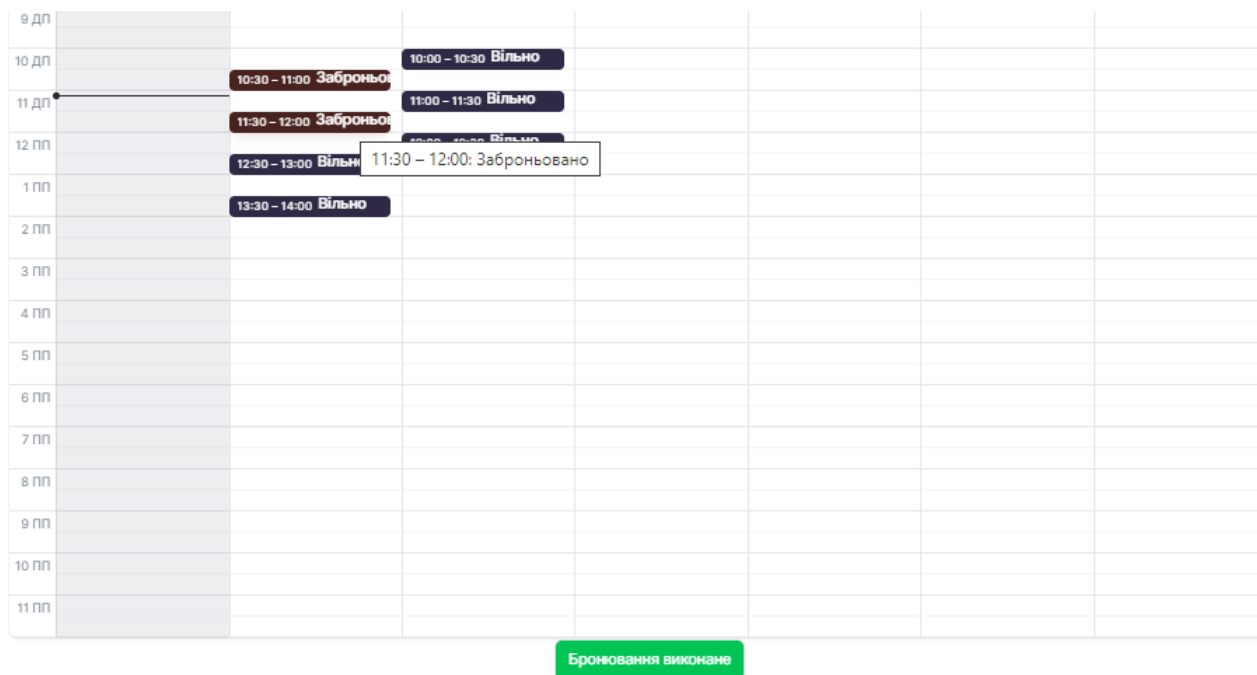


Рисунок 5.8 – Профіль того ж спеціаліста після бронювання вільного часового слоту

Користувач має можливість переглядати свої візити у календарному вигляді через вкладку «Мій календар», що зображено на рисунках 5.9. Тут однаковим кольором позначаються часові слоти з до одного й того ж спеціаліста. В даному випадку в користувача є записи до чотирьох спеціалістів.

В календарному режимі можна робити скасування бронювання. Для цього потрібно натиснути на будь-який слот в статусі «Заброньовано». Ця активність прибере ід користувача в записі часового слоту і поверне його статус в статус «Вільно». Повідомлення про скасування бронювання зображене на рисунку 5.10.

листопада 2023

Місяць ▾

Сьогодні

<

>

пон	вів	сер	чтв	птн	суб	нед
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
				Завершено Завершено	Завершено	
20	21	22	23	24	25	26
	Заброньовано Заброньовано	Заброньовано				

Рисунок 5.9 – Відображення візитів користувача у календарному вигляді у вкладці «Мій календар»

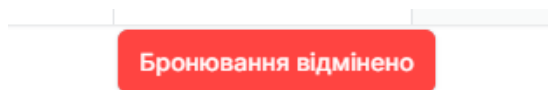


Рисунок 5.10 – Скасування бронювання у вкладці «Мій календар»

Також доступ до історії візитів доступний через вкладку «Історія візитів», що зображено на рисунку 5.11. Тут зміна будь-якої інформації не передбачена.

Дата	Час	Статус	Спеціаліст	Спеціалізація
1 22 листопада 2023 р.	17:00 - 18:00	Заброньовано	Валентин Володимирович Мельник	Хірургія
2 21 листопада 2023 р.	11:30 - 12:00	Заброньовано	Оксана Петрівна Романюк	Педіатрія
3 21 листопада 1023 р.	10:30 - 11:00	Заброньовано	Оксана Петрівна Романюк	Педіатрія
4 18 листопада 2023 р.	15:00 - 15:30	Завершено	Анатолій Ігорович Ковальчук	Кардіологія
5 17 листопада 2023 р.	15:30 - 16:00	Завершено	Лілія Василівна Сергієнко	Сімейна медицина
6 17 листопада 2023 р.	14:30 - 15:30	Завершено	Лілія Василівна Сергієнко	Сімейна медицина

Рисунок 5.11 – Відображення візитів користувачі через вкладку «Історія візитів»

Відображення персональної інформації користувача можливе при переході в його профіль в правому верхньому куті, представлено на рисунку 5.12. Вміст профілю відповідає спроектованій раніше базі даних, всі поля тут можуть бути редаговані.

PrimeCare Medicine

Головна | Наші спеціалісти | Мій календар | Історія візитів

М

Повне ім'я
Максим

Контактний номер телефону
+38063-33-33-333

Email
email-mail@gmail.com

Місто
Вінниця

Адреса
вул. Пирогова 1

Дата народження
1 січня 1980 р.

Додаткова інформація

Пароль

✓ Зберегти дані

Рисунок 5.12 – Відображення персональної інформації користувача

Функціональні можливості користувачів в ролі Administrator

Користувач в ролі Administrator має доступ до сторінок «Головна», «Наші спеціалісти», «Управління візитами», «Історія візитів», «Користувачі».

Зміст сторінок «Головна» та «Наші спеціалісти» не змінюється та відповідає рисункам 5.5-5.8. Відмінність тільки в тому, що в календарному перегляді вільних часових слотів в профілі спеціаліста, користувач в ролі Administrator може виконати будь-які дії з слотом, вказавши при цьому дані іншого користувача. Ця функція необхідна, наприклад, для бронювання часового слоту в телефонному режимі.

Додатково на сторінці «Наші спеціалісти» з'являється можливість додавати до бази даних нових спеціалістів та редагувати інформацію про існуючих. Графік прийому формується автоматично на основі вказаних початкових та кінцевих часових проміжків, а також може бути змінений користувачем в ролі Administrator.

На сторінці «Управління візитами» користувач в ролі Administrator має можливість управляти візитами за допомогою канбан дошки, переміщуючи відповідний запис до наступного стовпця, що зображено на рисунку 5.13. При цьому відбувається автоматична зміна статусу візиту з «Заброньовано» в «В процесі» та «Завершено».

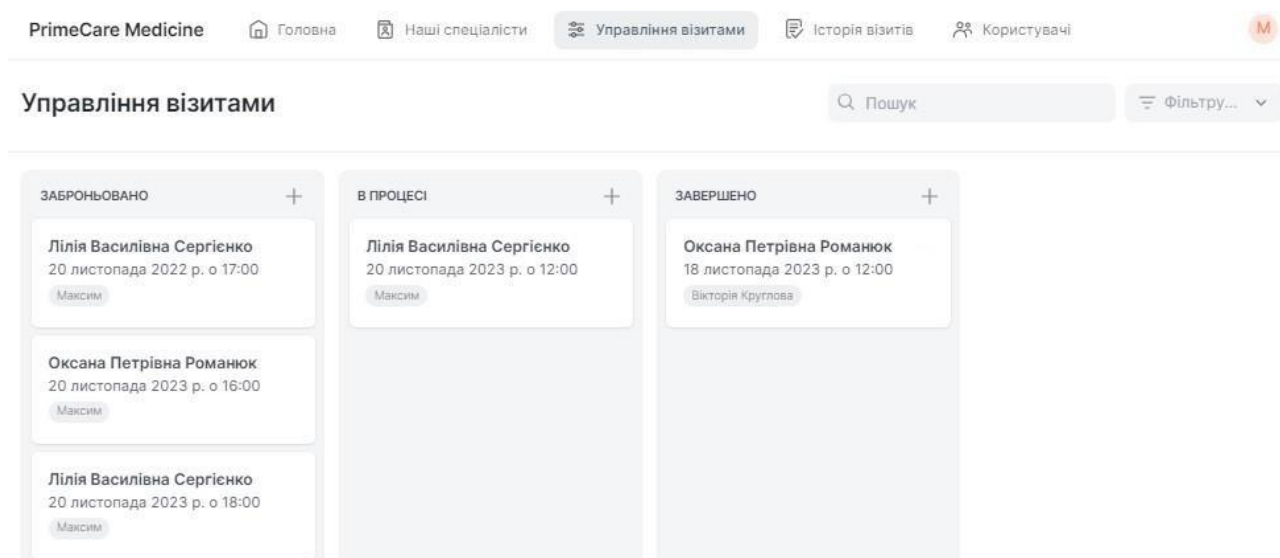


Рисунок 5.13 – Управління візитами за допомогою канбан дошки

На сторінці «Історія візитів» користувач в ролі Administrator має доступ до списку всіх візитів, зображено на рисунку 5.14, окрім тих, що мають статус «Вільно», оскільки їх відображення тут не є інформативним.

Дата	Час	Статус	Спеціаліст	Спеціалізація	Пацієнт
22 листопада 2023 р.	17:00 - 18:00	Заброньовано	Валентин Володимирович Мельник	Хірургія	Максим
22 листопада 2023 р.	11:00 - 11:30	Заброньовано	Оксана Петрівна Романюк	Педіатрія	Вікторія Круглова
22 листопада 2023 р.	10:00 - 10:30	Заброньовано	Оксана Петрівна Романюк	Педіатрія	Вікторія Круглова
21 листопада 2023 р.	11:30 - 12:00	Заброньовано	Оксана Петрівна Романюк	Педіатрія	Максим
21 листопада 2023 р.	10:30 - 11:00	Заброньовано	Оксана Петрівна Романюк	Педіатрія	Максим
18 листопада 2023 р.	15:00 - 15:30	Завершено	Анатолій Ігорович Ковальчук	Кардіологія	Максим

Рисунок 5.14 – Відображення історії візитів для користувача в ролі Administrator

Додатковою можливістю адміністратора є можливість перегляду та редагування інформації в профілях користувачів, що представлено на рисунках 5.15 та 5.16. Присутня також можливість пошуку користувачів та створення **НОВИХ**.

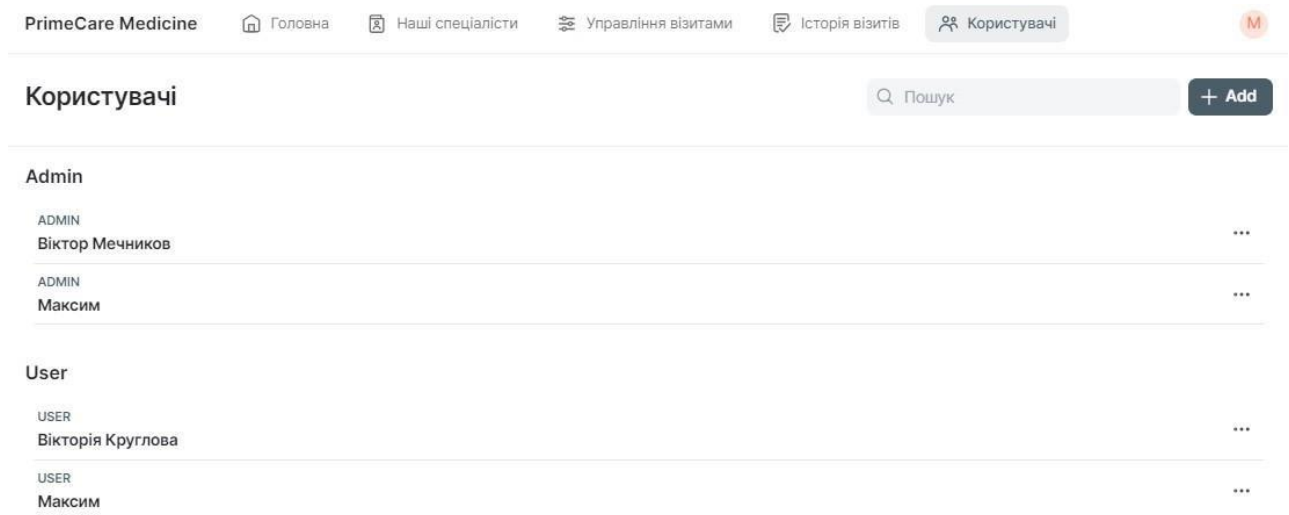


Рисунок 5.15 – Відображення списку користувачів для користувача в ролі Administrator

PrimeCare Medicine [Головна](#) [Наші спеціалісти](#) [Управління візитами](#) [Історія візитів](#) [Користувачі](#) M

Користувачі > Максим

Повне ім'я	Максим
Email	email-mail@gmail.com
Контактний номер телефону	+38063-33-33-333
Місто	Вінниця
Адреса	вул. Пирогова 1
Дата народження	1 січня 1980 р.
Додаткова інформація	
Роль	User
Пароль	*****

✓ Зберегти дані

Рисунок 5.16 – Сторінка перегляду та редагування інформації в профілях користувачів для користувача в ролі Administrator

Автоматизована система управління електронною чергою медичного закладу розроблена у відповідності до поставлених функціональних та технічних вимог, протестована на правильність роботи усіх блоків системи та взаємодії з базою даних. Правильність роботи підтверджується співпадінням очікуваних та фактичних результатів.

Отже, розроблена система може сприяти покращенню доступу до медичних послуг, зменшенню адміністративних навантажень та підвищенню задоволення як пацієнтів, так і медичного персоналу медичного закладу, а вдосконалення та оснащення системи додатковими функціями у відповідності до запитів ринку дозволяє її використання для бізнесу будь-якого масштабу.

ВИСНОВКИ

У ході виконання магістерської кваліфікаційної роботи було проведено глибоке дослідження та аналіз системи управління електронною чергою для медичного закладу з метою поліпшення організації надання медичних послуг та оптимізації робочих процесів. Метою роботи було створення інноваційного інструменту, спрямованого на покращення доступу до медичних послуг, зменшення адміністративних навантажень та підвищення задоволення як пацієнтів, так і медичного персоналу.

В першому розділі було проведено дослідження та розроблено концепцію автоматизованої системи управління чергою у медичному закладі. Описано актуальність теми та етапи планування роботи, включаючи аналіз особливостей систем електронної черги, можливості реалізації функцій для пацієнтів та медичного персоналу, і встановлено важливість цієї системи для поліпшення медичного обслуговування. Також узагальнено область застосування розробленої системи в медичних та інших галузях, а також сформульовано вимоги до програмно-технічного забезпечення, що розробляється.

В другому розділі було проведено аналіз сучасних аспектів автоматизації систем управління електронною чергою, досліджено роботу існуючих систем, визначено їхні переваги та недоліки. Детально вивчено особливості підключення та налаштування бази даних, інтеграції з іншими системами, автоматизованого оповіщення та розсилки інформації, відслідковування та аналіз даних, а також необхідність технічної підтримки та безпеки системи.

В третьому розділі досліджено взаємодію модуля управління електронною чергою з іншими модулями в автоматизованій системі управління медичними закладами, а саме інтеграцію з іншими системами управління медичними закладами, оптимізацію обміну даними між модулями та синхронізацію роботи модуля зі стандартами безпеки та конфіденційності.

В четвертому розділі проведено огляд та аналіз існуючих систем управління електронною чергою медичних закладів, враховано питання роботи таких сервісів, як Health24, Helsi.me, MedCard24 та Medics. Результати

дослідження були представлені у вигляді зведеної таблиці.

В п'ятому розділі виконано проектування системи, включаючи створення Use Case UML-діаграми, UML-діаграми класів, ER-діаграми. Описано технічні засоби, мови програмування та реалізовано функціональні можливості для користувачів та адміністраторів.

Здійснено тестування розробленої системи, результати якого свідчать про її ефективність та можливість використання в медичних закладах для поліпшення організації робочих процесів. Робота виконана в повному обсязі, відповідає поставленим завданням та меті, і може служити важливим внеском у сферу автоматизації управління електронною чергою в медичних закладах.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. David Loshin. ETL (Extract, Transform, Load) // Business Intelligence. — 2nd. — Morgan Kaufmann, 2012. — 400 p. — ISBN 978-0-12-385890-0.
2. David Haertzen. ETL Tools // The Analytical Puzzle: Profitable Data Warehousing, Business Intelligence and Analytics. — Technics Publications, 2012. — 346 p. — ISBN 978-1-935504-20-7.
3. Ralph Kimball, The Data Warehouse ETL Toolkit: Practical Techniques for Extracting, Cleaning, Conforming, and Delivering Data. — John Wiley & Sons, 2004. — 528 p. — ISBN 978-0-764-56757-5.
4. Codd E.F., "Providing OLAP (On-line Analytical Processing) to User-Analysts: An IT Mandate". Codd & Date, Inc. 1993.
5. Hackathorn R. "Data Warehousing Energizes Your Enterprise," Datamation, Feb.1, 1995, p.39.
6. Паклин Н. Б. Бизнес-аналитика: от данных к знаниям (+ CD). — СПб.: Изд. Питер, 2009. — 624 с.
7. Чубукова И. А. Data Mining: учебное пособие. — М.: Интернет-университет информационных технологий: БИНОМ: Лаборатория знаний, 2006. — 382 с. — ISBN 5-9556-0064-7.
8. Гарсиа-Молина Г., Системы баз данных. Полный курс = Database Systems: The Complete Book. — Вильямс, 2003. — 1088 с. — ISBN 5-8459-0384-X.
9. Date, C. J. Database in Depth. — O'Reilly, 2005. — 240 с. — ISBN 0-596-10012-4.
10. Дейт К. Дж. Введение в системы баз данных = Introduction to Database Systems. — 8-е изд. — М.: Вильямс, 2005. — 1328 с. — ISBN 5-8459-0788-8 (рус.) 0-321-19784-4 (англ.).
11. Ситник В. Ф. Інтелектуальний аналіз даних (дейтамайнінг): Навч. посібник. — К.: КНЕУ, 2007. — 376 с.
12. Журкин, И.Г. Информационные системы / И.Г. Журкин, С.В. Шайтура. — М.: Кудиц-Пресс, 2014. — 272 с.

13. H. Witten, *Data Mining: Practical Machine Learning Tools and Techniques*. — 3rd Edition. — Morgan Kaufmann, 2011. — P. 664. — ISBN 9780123748560.
14. Дюк В. *Data Mining: учебный курс (+CD)*. — СПб.: Изд. Питер, 2001. — 368 с.
15. Талгатова З.Т. Анализ и сравнение существующих моделей процессов etl для хранилищ данных // *Технические науки - от теории к практике: сб. ст. по матер. LIV междунар. науч.-практ. конф. № 1(49)*. — Новосибирск: СибАК, 2016.
16. Журавлёв Ю.И. *Распознавание. Математические методы. Программная система. Практические применения*. — М.: Изд. «Фазис», 2006. — 176 с. — ISBN 5-7036-0108-8.
17. Зиновьев А. Ю. *Визуализация многомерных данных*. — Красноярск: Изд. Красноярского государственного технического университета, 2000. - 180с.
18. Таланов В. М. *Проектирование информационных систем и баз данных. Учеб. пособие*. Саранск: Изд-во СВМО, 2013. 72 с.
19. Александров Э.Э. *Введение в программирование на языке С*. Саранск: Изд-во Мордов. ун-та, 2009. 316 с.
20. Александров Э.Э. *Программирование на языке С в Microsoft Visual Studio 2010*. Саранск: Изд-во Мордов. ун-та, 2010. 424 с.
21. Devlin, B., "Data warehouse: from architecture to implementation". Addison Wesley Longman, Inc. (1997). ISBN 0201964252.
22. Дубовой В.М. *Моделювання процесів і систем керування / В.М.Дубовой, С.М.Москвіна, О.Д.Никитенко*. — Універсум. — Вінниця, 2009. — 103 с.
23. Бритов П.А. *Практика построения хранилищ данных: SAS System*, [Электронный ресурс]. — Режим доступа: <http://www.sas.com/offices/europe/russia/articles/1999/dbms04.html>.
24. Dobre A., *Operators and classification for data mapping in semantic integration*. In: *Proceedings of the 22nd International Conference on Conceptual Modeling (ER'03)*, LNCS, vol. 2813, Chicago, USA, P. 534–547.

25. Trujillo J., A UML based approach for modeling ETL processes in data warehouses. In: Proceedings of the 22nd International Conference on Conceptual Modeling. LNCS, Chicago, USA.
26. Kimball R., The Data Warehouse ETL Toolkit. Practical Techniques for Extracting, Cleaning, Conforming and Delivering Data Wiley.
27. Vassiliadis P., Conceptual modeling for ETL processes. In: Proceedings of the Fifth ACM International Workshop on Data Warehousing and OLAP.
28. M. Jakobsson. Love and authentication. In CHI '08: Proceeding of the Twenty-Sixth Annual SIGCHI Conference on Human Factors in Computing Systems, New York, NY, USA, 2008. ACM. -C 197–200.
29. K. D. Mitnick. The Art of Deception: Controlling the Human Element of Security. Wiley, 2002.
30. R. W. Proctor Improving password security and memorability to protect persona and organizational information. Int. J. Hum.-Comput. Stud., 65(8): 2007. -C 744–757.
31. M. Just. Designing authentication systems with challenge questions. In L. F. Cranor and S. Garfinkel, editors, Security and Usability: Designing Secure Systems that People Can Use, , Sebastopol, CA, 2005. O'Reilly Media, Inc. - C 143–155.
32. S. Brostoff. Ten strikes and you're out: Increasing the number of login attempts can improve password usability. In Proceedings of CHI 2003 Workshop on HCI and Security Systems, 2003.
33. Кузнецов С. Д. Основы баз данных. — 2-е изд. — М.: Интернет-университет информационных технологий; БИНОМ. Лаборатория знаний, 2007. — 484 с. — ISBN 978-5-94774-736-2.
34. Kimball, R., "The Data Warehouse Toolkit. Practical Techniques for Building Dimensional Data Warehouses". John Wiley & Sons, Inc (1996). ISBN 0471153370.
35. Kimball, R. et al., "The Data Warehouse Lifecycle Toolkit: Expert Methods for Designing, Developing and Deploying Data Warehouses". John Wiley & Sons, Inc (1998). ISBN 0471255475.

36. Silverston, L., Inmon, W., Graziano, K., "The Data Model Resource Book. A Library of Logical Data Models and Data Warehouse Designs". John Wiley & Sons, Inc (1997). ISBN 0471153672.
37. Konovalov, V. P. Learning word embeddings for low resource languages: the case of buryat / V. P. Konovalov, Z. B. Tumunbayarova // *Komp'yuternaja Lingvistika i Intellektual'nye Tehnologii*. – 2018. – С. 331–341.
38. Wang, R., "Data Quality Requirements Analysis and Modeling", Ninth International Conference of Data Engineering, 1993, Vienna, Austria.
39. Marco D., *Universal Meta Data Models*, Wiley, 2004.
40. Дейт, К. Дж. Введение в системы баз данных. 8-е издание/ К. Дж. Дейт: – Москва: Издательский дом «Вильямс», 2005. – 1328 с.
41. Хоббс Л. Разработка и эксплуатация хранилищ баз данных/ Лилиан Хоббс, Сьюзан Хилсон, Шилпа Лоуенд: – Москва: Издательство «КУДИЦ-Образ», 2004. – 596 с.
42. Dehdouh, K. Building OLAP Cubes from Columnar NoSQL Data Warehouses. In 6th International Conference Model and Data Engineering. Springer International Publishing 2016, 166–179.
43. Асадуллаев С. "Фирменные архитектуры хранилищ данных", *PC Week / RE*, 1998, № 32-33, стр. 156-157.
44. Jukic, N., Sharma, A., Nestorov, S. and Jukic, B. 2015. Augmenting Data Warehouses with Big Data. *Information Systems Management* 2015, 200–209.
45. Leong-Hong B.W., *Data Dictionary / Directory Systems*. John Wiley & Sons. 1982. (Леонг-Хонг Б., Плагман Б. "Системы словарей-справочников данных", М.: Финансы и статистика, 1986).
46. Ларсон Б. Разработка бизнес-аналитики в Microsoft SQL Server 2005. – Спб.: Питер, 2008. – 684 с.
47. *Data Governance & Data Modeling Solutions* [Электронный ресурс] – Режим доступа до ресурсу: <https://www.erwin.com/>
48. ERwin [Электронный ресурс] – Режим доступа до ресурсу: <https://old.dataone.org/software-tools/erwin>

- ERwin [Электронный ресурс] – Режим доступа до ресурсу:
<https://old.dataone.org/software-tools/erwin>
49. What you'll love about SQL Server 2019 [Электронный ресурс] – Режим доступа до ресурсу: <https://www.microsoft.com/en-us/sql-server/sql-server-2019>
50. SQL Server Tutorial [Электронный ресурс] – Режим доступа до ресурсу:
<https://www.sqlservertutorial.net/>
51. Connect Microsoft SQL Server to Metabase [Электронный ресурс] – Режим доступа до ресурсу: https://www.metabase.com/data_sources/microsoft-sql-server
52. The world's leading analytics platform [Электронный ресурс] – Режим доступа до ресурсу: <https://www.tableau.com/>
53. Визуализация и аналитика данных Tableau [Электронный ресурс] – Режим доступа до ресурсу: <https://corewin.ua/ru/analytics-and-data/tableau/>
54. Описание системы Tableau Desktop [Электронный ресурс] – Режим доступа до ресурсу: <https://soware.ru/products/tableau-desktop>
55. Аббакумов А.А. Базы данных (MS SQL Server). Учеб. пособие. Саранск: Изд-во СВМО, 2015. 66 с.
56. Берегер А. Microsoft SQL Server 2005 Analysis Services. OLAP и многомерный анализ данных. – Спб.: ВХБ – Петербург, 2007. – 928с.

Додатки

**Додаток А
(обов'язковий)**

ЗАТВЕРДЖУЮ

Зав. кафедри АІТ

Олег БІСІКАЛО

«_____» _____ 2023 р.

ТЕХНІЧНЕ ЗАВДАННЯ

на виконання магістерської кваліфікаційної роботи

**Розробка автоматизованої системи управління електронною чергою
медичного закладу**

08-31.МКР.008.02.000 ТЗ

Студент групи ЗАКІТ-22м

Максим КОНЕЦУЛ

Керівник: д.т.н., доцент,

т.в.о.зав. кафедри КСУ

Марія ЮХИМЧУК

1. Назва та галузь застосування

1.1. Назва – Розробка автоматизованої системи управління електронною чергою медичного закладу.

1.2. Галузь застосування – Комп'ютеризовані системи управління бізнес-процесів підприємств та установ.

2. Підстава для проведення розробки.

Тема магістерської кваліфікаційної роботи затверджена наказом по ВНТУ №247 від “18” вересня 2023р.

3. Мета та призначення розробки.

Метою магістерської кваліфікаційної роботи є розробка та впровадження автоматизованої системи управління електронною чергою для медичного закладу з метою поліпшення організації надання медичних послуг та оптимізації робочих процесів. Робота спрямована на створення інноваційного інструменту, який сприятиме покращенню доступу до медичних послуг, зменшенню адміністративних навантажень та підвищенню задоволення як пацієнтів, так і медичного персоналу медичного закладу.

4. Джерела розробки.

Магістерська кваліфікаційна робота виконується вперше. В ході проведення розробки повинні використовуватись такі документи:

1. Ситник В. Ф. Інтелектуальний аналіз даних (дейтамайнінг): Навч. посібник. — К.: КНЕУ, 2007. — 376 с.
2. Чубукова И. А. Data Mining: учебное пособие. — М.: Интернет университет информационных технологий: БИНОМ: Лаборатория знаний, 2006. — 382 с. — ISBN 5-9556-0064-7.
3. Ralph Kimball, Joe Caserta. The Data Warehouse ETL Toolkit: Practical Techniques for Extracting, Cleaning, Conforming, and Delivering Data. — John Wiley & Sons, 2004. — 528 p. — ISBN 978-0-764-56757-5.

4. David Loshin. ETL (Extract, Transform, Load) // Business Intelligence. — 2nd. — Morgan Kaufmann, 2012. — 400 p. — ISBN 978-0-12-385890-0

5. Вимоги до розробки.

5.1. Перелік головних етапів виконання розробки:

- визначити аспекти функціонування систем управління електронною чергою;
- дослідити функціональні можливості систем аналогів;
- скласти вимоги до автоматизованої системи, що розробляється;
- спроектувати та розробити систему у відповідності до поставлених вимог та функцій:
- забезпечення доступу користувачів до системи та персональних аккаунтів;
- управління пошуком лікарів;
- бронювання та управління прийомами;
- перегляд історії візитів.

5.2. Основні технічні вимоги до розробки.

5.2.1. Вимоги до програмної платформи:

- WINDOWS 7\8\10.

5.2.2. Умови експлуатації системи:

- робота на стандартних ПЕОМ в приміщеннях зі стандартними умовами;
- можливість цілодобового функціонування системи;
- текст програмного забезпечення системи є цілком закритим.

6. Стадії та етапи розробки.

6.1 Пояснювальна записка:

- Дослідження актуальності поставленої задачі 20.09.2023р.
- Дослідження основних аспектів автоматизації

- | | |
|---|--------------|
| системи управління електронною чергою | 27.09.2023р. |
| – Дослідження роботи та порівняльний аналіз існуючих систем управління електронною чергою медичного закладу | 01.10.2023р. |
| – Проектування та розробка автоматизованої системи управління електронною чергою медичного закладу | 05.10.2023р. |
| – Оформлення пояснювальної записки, графічного матеріалу і презентації | 29.10.2023р. |

6.2 Графічні матеріали:

- | | |
|---|--------------|
| – Use-case UML-діаграма | 05.11.2023р. |
| – ER-діаграма структури бази даних | 14.11.2023р. |
| – вигляд екранів розробленого web-додатку | 20.11.2023р. |

7. Порядок контролю і приймання.

- 7.1. Хід виконання роботи контролюється керівником роботи. Рубіжний контроль провести до «01» листопада 2023 р.
- 7.2. Атестація проекту здійснюється на попередньому захисті. Попередній захист магістерської кваліфікаційної роботи провести до «21» листопада 2023р.
- 7.3. Підсумкове рішення щодо оцінки якості виконання роботи приймається на засіданні ЕК.
- 7.4. Захист магістерської кваліфікаційної роботи провести до «18» грудня 2023р.

**Додаток Б
(обов'язковий)**

ІЛЮСТРАТИВНА ЧАСТИНА

**Розробка автоматизованої системи управління електронною чергою
медичного закладу**

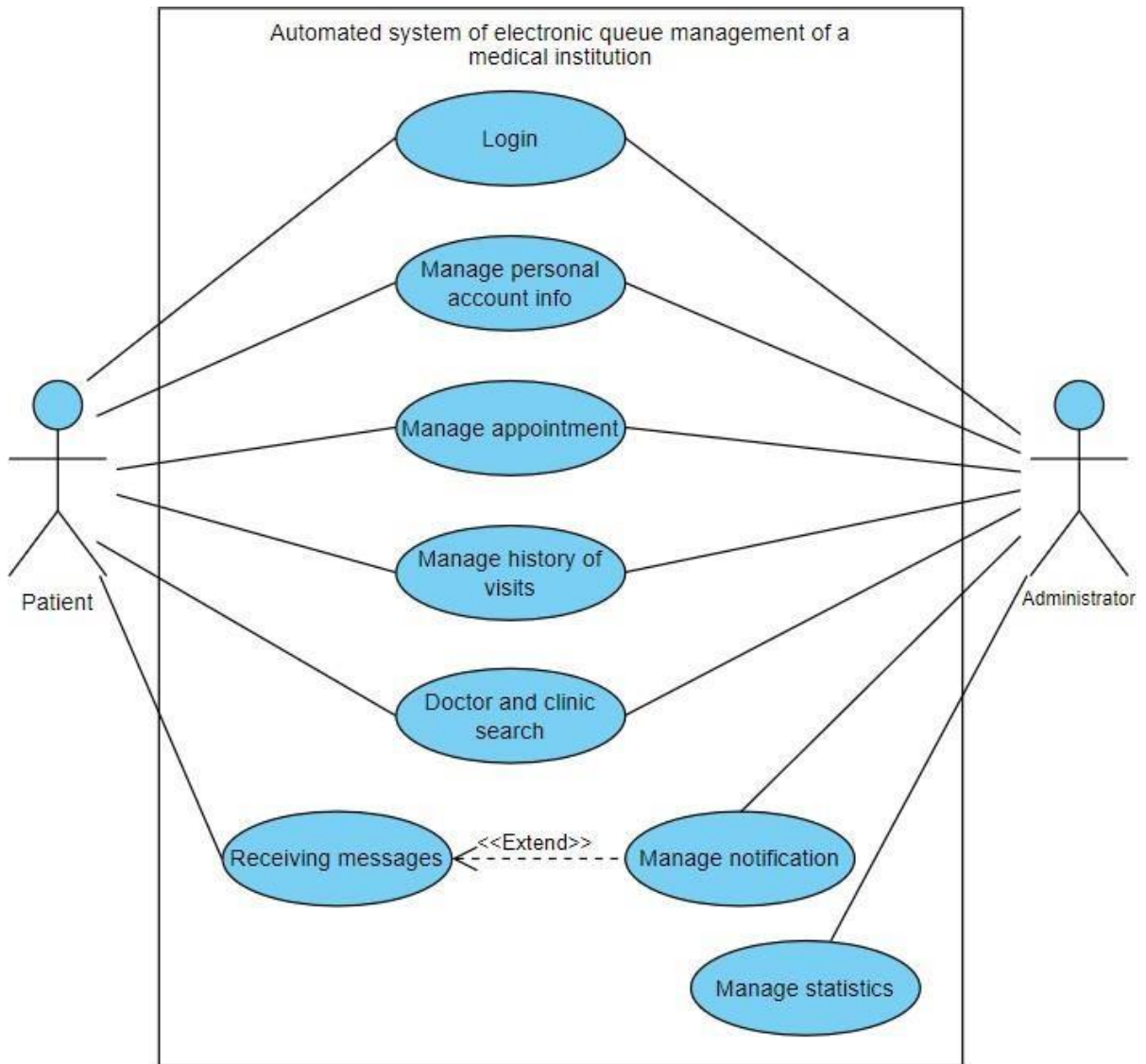


Рисунок Б.1 Use Case UML-діаграма для автоматизованої системи управління електронною чергою медичного закладу

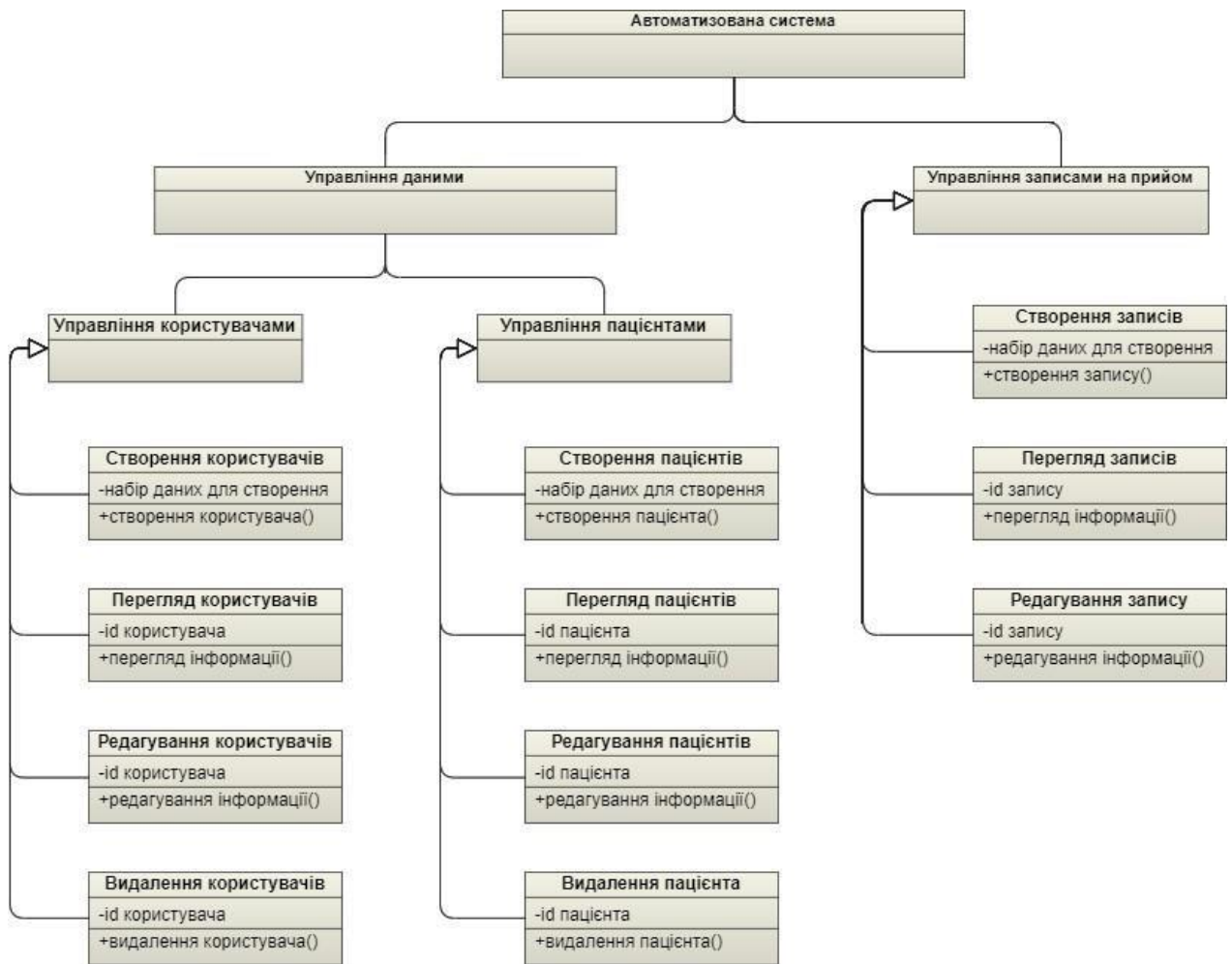


Рисунок Б.2. UML-діаграма класів діаграму для автоматизованої системи управління електронною чергою медичного закладу

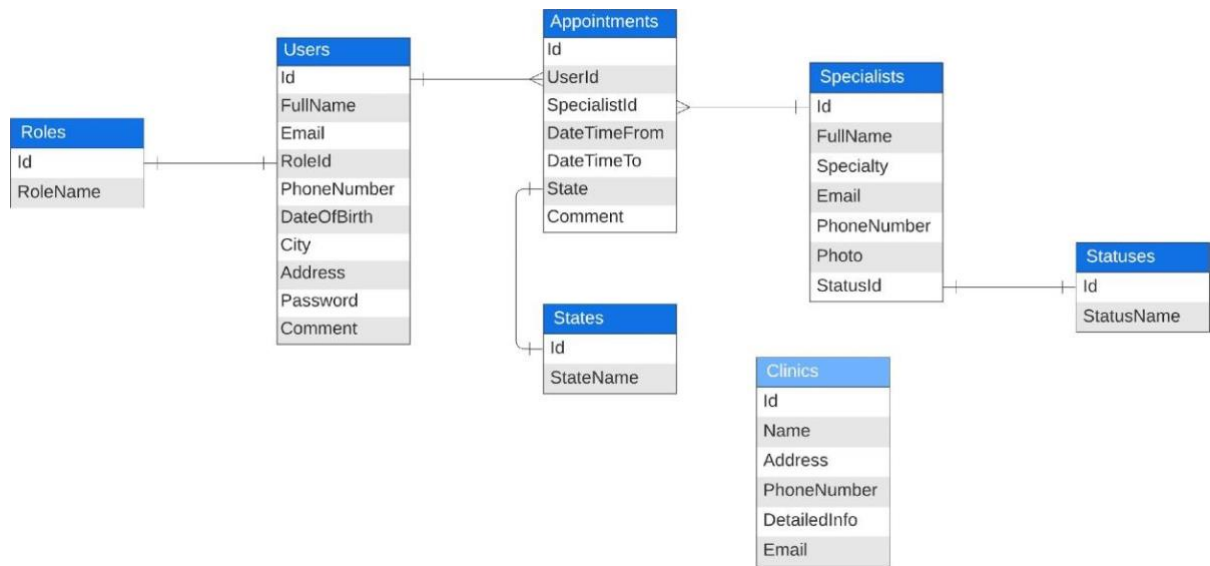


Рисунок Б.3. ER-діаграма для автоматизованої системи управління електронною чергою медичного закладу



Рисунок Б.4. Вікно авторизації в системі



Ми щиро радімо вас вітати у нашому освіженому медичному просторі, де з 2023 року ми надаємо високоякісні медичні послуги з турботою та професіоналізмом.

Наші лікарі – це команда висококваліфікованих фахівців у сфері сімейної медицини, терапії, педіатрії та інших напрямків. Вони готові надати вам найкращі медичні консультації та допомогу.

На нашому сайті ви можете знайти інформацію про графік роботи наших лікарів, ознайомитися з їхнім досвідом та кваліфікацією. Також ви маєте можливість забронювати візит, щоб забезпечити зручний та швидкий доступ до наших послуг.

Довірте своє здоров'я нашим експертам, і ми гарантуємо вам індивідуальний підхід та якісний медичний сервіс.

Запрошуємо вас долучитися до нашої сім'ї та зробити перший крок до здоров'я разом з нами!



**Михайло Олександрович
Литвиненко**
Психіатрія



Лілія Василівна Сергієнко
Сімейна медицина



**Валентин Володимирович
Мельник**
Хірургія



Ігор Васильович Шевчук
Лікар загальної практики



вул. Миколи Оводова, 10, Вінниця, Україна, 21000

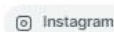


Рисунок Б.5. Вигляд головного екрану автоматизованої системи управління електронною чергою медичного закладу для користувача в ролі User

Наші спеціалісти








 <p>Анатолій Ігорович Ковальчук Кардіологія</p>	 <p>Ігор Васильович Шевчук Лікар загальної практики</p>	 <p>Ірина Миколаївна Кравченко Неврологія</p>	 <p>Ірина Олександрівна Шевчук Остеопатія</p>	 <p>Оксана Петрівна Романюк Педіатрія</p>	 <p>Лілія Василівна Сергієнко Сімейна медицина</p>	 <p>Валентин Володимирович Мельник Хірургія</p>
--	--	--	--	---	---	--

Рисунок Б.6. Відображення списку активних лікарів у вкладці «Наші спеціалісти»

Наші спеціалісти > Оксана Петрівна Романюк



Оксана Петрівна Романюк
Педіатрія

[Задзвонити на реєстрацію](#)

листоп. 20-е - листоп. 26-е 2023

Тижд... ▾

Сьогодні

<

>

	20 пон	21 вів	22 сер	23 чтв	24 птн	25 суб	26 нед
12 дп							
1 дп							
2 дп							
3 дп							
4 дп							
5 дп							
6 дп							
7 дп							
8 дп							
9 дп							
10 дп							
11 дп		10:30 – 11:00 Заброньовано	10:00 – 10:30 Вільно				
12 дп		11:30 – 12:00 Вільно	11:00 – 11:30 Вільно				
1 пп		12:30 – 13:00 Вільно	12:00 – 12:30 Вільно				
2 пп		13:30 – 14:00 Вільно					

Рисунок Б.7. Профіль спеціаліста з одним заброньованим візитом на дату 21 листопада 2023 року

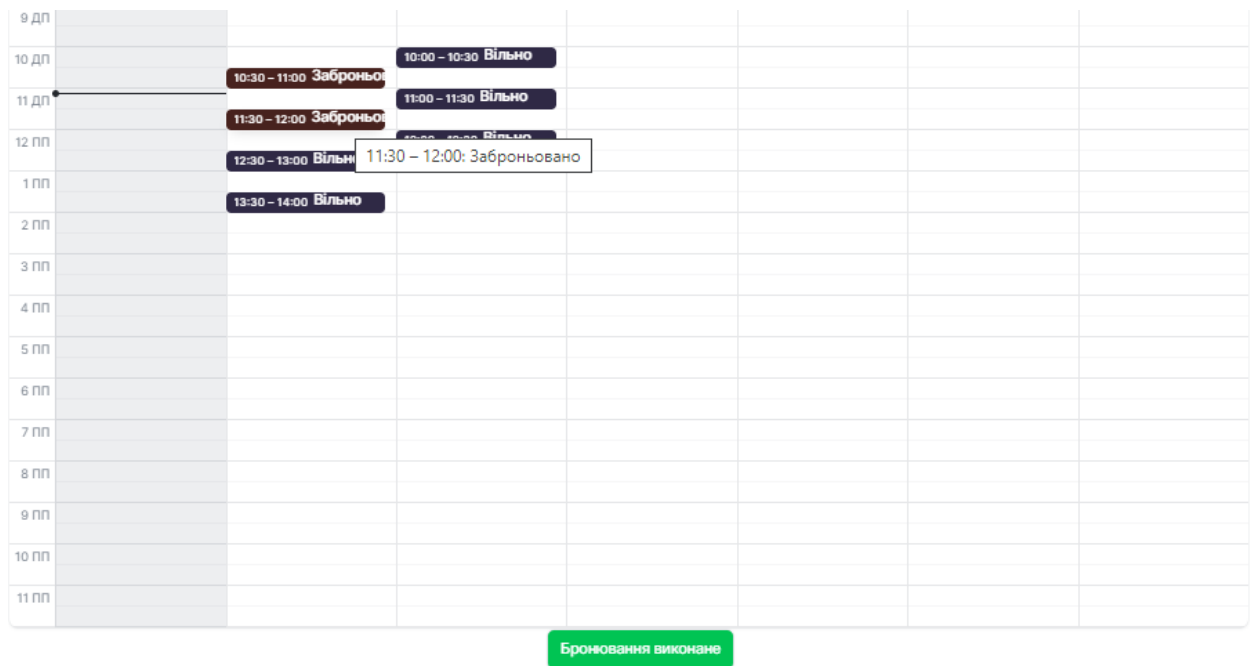


Рисунок Б.8. Профіль того ж спеціаліста після бронювання вільного часового слоту

листопада 2023

Місяць ▾

Сьогодні

<

>

пон	вів	сер	чтв	птн	суб	нед
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17 Завершено Завершено	18 Завершено	19
20	21 Заброньовано Заброньовано	22 Заброньовано	23	24	25	26
27	28	29	30	1	2	3

Рисунок Б.9. Відображення візитів користувача у календарному вигляді у вкладці «Мій календар»



Бронювання відмінено

Рисунок Б.10. Скасування бронювання у вкладці «Мій календар»

	Дата	Час	Статус	Спеціаліст	Спеціалізація
1	22 листопада 2023 р.	17:00 - 18:00	Заброньовано	Валентин Володимирович Мельник	Хірургія
2	21 листопада 2023 р.	11:30 - 12:00	Заброньовано	Оксана Петрівна Романюк	Педіатрія
3	21 листопада 1023 р.	10:30 - 11:00	Заброньовано	Оксана Петрівна Романюк	Педіатрія
4	18 листопада 2023 р.	15:00 - 15:30	Завершено	Анатолій Ігорович Ковальчук	Кардіологія
5	17 листопада 2023 р.	15:30 - 16:00	Завершено	Лілія Василівна Сергієнко	Сімейна медицина
6	17 листопада 2023 р.	14:30 - 15:30	Завершено	Лілія Василівна Сергієнко	Сімейна медицина

Рисунок Б.11. Відображення візитів користувачі через вкладку «Історія візитів»

Повне ім'я

Максим

Контактний номер телефону

+38063-33-33-333

Email

email-mail@gmail.com

Місто

Вінниця

Адреса

вул. Пирогова 1

Дата народження

1 січня 1980 р.

Додаткова інформація

Пароль

✓ Зберегти дані

Рисунок Б.12. Відображення персональної інформації користувача

Управління візитами

Пошук

Фільтру...

The image shows a Kanban board for appointment management. It consists of three columns: 'ЗАБРОНЬОВАНО' (Booked), 'В ПРОЦЕСІ' (In Progress), and 'ЗАВЕРШЕНО' (Completed). Each column has a plus sign in the top right corner. The 'ЗАБРОНЬОВАНО' column contains three cards for 'Лілія Василівна Сергієнко' and 'Оксана Петрівна Романюк'. The 'В ПРОЦЕСІ' column contains one card for 'Лілія Василівна Сергієнко'. The 'ЗАВЕРШЕНО' column contains one card for 'Оксана Петрівна Романюк'. Each card displays the patient's name, the appointment date and time, and the name of the specialist.

Статус	Пациєнт	Дата та час	Спеціаліст
ЗАБРОНЬОВАНО	Лілія Василівна Сергієнко	20 листопада 2022 р. о 17:00	Максим
ЗАБРОНЬОВАНО	Оксана Петрівна Романюк	20 листопада 2023 р. о 16:00	Максим
ЗАБРОНЬОВАНО	Лілія Василівна Сергієнко	20 листопада 2023 р. о 18:00	Максим
В ПРОЦЕСІ	Лілія Василівна Сергієнко	20 листопада 2023 р. о 12:00	Максим
ЗАВЕРШЕНО	Оксана Петрівна Романюк	18 листопада 2023 р. о 12:00	Вікторія Круглова

Рисунок Б.13. Управління візитами за допомогою канбан дошки

	Дата	Час	Статус	Спеціаліст	Спеціалізація	Пацієнт
1	22 листопада 2023 р.	17:00 - 18:00	Заброньовано	Валентин Володимирович Мельник	Хірургія	Максим
2	22 листопада 2023 р.	11:00 - 11:30	Заброньовано	Оксана Петрівна Романюк	Педіатрія	Вікторія Круглова
3	22 листопада 2023 р.	10:00 - 10:30	Заброньовано	Оксана Петрівна Романюк	Педіатрія	Вікторія Круглова
4	21 листопада 2023 р.	11:30 - 12:00	Заброньовано	Оксана Петрівна Романюк	Педіатрія	Максим
5	21 листопада 1023 р.	10:30 - 11:00	Заброньовано	Оксана Петрівна Романюк	Педіатрія	Максим
6	18 листопада 2023 р.	15:00 - 15:30	Завершено	Анатолій Ігорович Ковальчук	Кардіологія	Максим
7	17 листопада 2023 р.	15:30 - 16:00	Завершено	Лілія Василівна Сергієнко	Сімейна медицина	Максим
8	17 листопада 2023 р.	14:30 - 15:30	Завершено	Лілія Василівна Сергієнко	Сімейна медицина	Максим

Рисунок Б.14. Відображення історії візитів для користувача в ролі Administrator

Користувачі

Пошук

+ Add

Admin

ADMIN

Віктор Мечников

...

ADMIN

Максим

...

User

USER

Вікторія Круглова

...

USER

Максим

...

Рисунок Б.15. Відображення списку користувачів для користувача в ролі Administrator

Користувачі > Максим

Повне ім'я	Максим
Email	email-mail@gmail.com
Контактний номер телефону	+38063-33-33-333
Місто	Вінниця
Адреса	вул. Пирогова 1
Дата народження	1 січня 1980 р.
Додаткова інформація	
Роль	User
Пароль	*****

✓ Зберегти дані

Рисунок Б.16. Сторінка перегляду та редагування інформації в профілях користувачів для користувача в ролі Administrator

Додаток В

Лістинг програмного забезпечення

```
Додавання можливості створення нового візиту:
# views.py
from django.shortcuts import render, redirect
from .models import Visit
from .forms import VisitForm # Додамо це пізніше

def add_appointment(request):
    if request.method == 'POST':
        form = VisitForm(request.POST)
        if form.is_valid():
            form.save()
            return redirect('appointment_list')
        else:
            form = VisitForm()

    return render(request, 'add_appointment.html', {'form': form})

Додатковий фрагмент коду для відображення посилань у меню
на додавання нового візиту:
<!-- navigation.html (шаблон меню) -->
<a href="{% url 'appointment_list' %}">Appointments</a>
<a href="{% url 'add_appointment' %}">Add
Appointment</a>

Django (Python, серверна частина):
# models.py
from django.db import models
from django.contrib.auth.models import User

class Specialist(models.Model):
    user = models.OneToOneField(User,
on_delete=models.CASCADE)

    status = models.CharField(max_length=20)

class Visit(models.Model):
    datetime = models.DateTimeField()

    status = models.CharField(max_length=20, default='Available')

    specialist = models.ForeignKey(Specialist,
on_delete=models.CASCADE)

    patient = models.ForeignKey(User, on_delete=models.CASCADE,
blank=True, null=True)

# Додайте інші поля за необхідності

class UserProfile(models.Model):
    user = models.OneToOneField(User,
on_delete=models.CASCADE)

    role = models.CharField(max_length=20)

Створення форми для введення даних нового візиту:
# forms.py
from django import forms
from .models import Visit

class VisitForm(forms.ModelForm):
    class Meta:
        model = Visit
        fields = ['datetime', 'status', 'specialist', 'patient']

Оновлення шляхів у файлі urls.py:
# urls.py
from django.urls import path
from .views import get_appointments, add_appointment

urlpatterns = [
    path('appointments/', get_appointments,
name='appointment_list'),
    path('add_appointment/', add_appointment,
name='add_appointment'),
]

Створення HTML-шаблону для форми додавання нового візиту
(add_appointment.html):
<!-- add_appointment.html -->
{% extends 'base.html' %}

{% block content %}
<h2>Add New Appointment</h2>
<form method="post" action="{% url 'add_appointment' %}">

React.js (клієнтська частина):
// AppointmentList.js
import React, { useState, useEffect } from 'react';
```

```

const AppointmentList = ({ userRole }) => {
  const [appointments, setAppointments] = useState([]);

  useEffect(() => {
    fetch('/api/appointments/')
      .then(response => response.json())
      .then(data => setAppointments(data));
  }, []);

  const renderAppointments = () => {
    return appointments.map(appointment => (
      <li key={appointment.id}>
        {appointment.datetime} - {appointment.status}
        {userRole === 'Administrator' && (
          <button onClick={() =>
            handleEditAppointment(appointment.id)}>Edit</button>
        )}
      </li>
    ));
  };

  const handleEditAppointment = (appointmentId) => {};

  return (
    <div>
      <h2>Appointment List</h2>
      <ul>{renderAppointments()}</ul>
    </div>
  );
};

```

export default AppointmentList;

Підключення до бази даних та створення моделей:

settings.py

```

DATABASES = {
  'default': {
    'ENGINE': 'django.db.backends.mysql',
    'NAME': 'database',
    'USER': 'Admin',
    'PASSWORD': '1111',
    'HOST': 'localhost',
    'PORT': '3306',

```

```

}
}

# models.py
from django.db import models
from django.contrib.auth.models import User

class Specialist(models.Model):
    user = models.OneToOneField(User,
    on_delete=models.CASCADE)

    status = models.CharField(max_length=20)

class Visit(models.Model):
    datetime = models.DateTimeField()
    status = models.CharField(max_length=20, default='Available')
    specialist = models.ForeignKey(Specialist,
    on_delete=models.CASCADE)

    patient = models.ForeignKey(User, on_delete=models.CASCADE,
    blank=True, null=True)

class UserProfile(models.Model):
    user = models.OneToOneField(User,
    on_delete=models.CASCADE)

    role = models.CharField(max_length=20)

```

Лістинг файлу index.html

```

<!DOCTYPE html>
<html lang="uk-UA">
<head>
  <!-- Global site tag (gtag.js) - Google Analytics -->
  <script async
src="https://www.googletagmanager.com/gtag/js?id=UA-39381181-1" ></script>
  <script>
    window.dataLayer = window.dataLayer || [];

    function gtag() {
      dataLayer.push(arguments);
    }

    gtag('js', new Date());

    gtag('config', 'UA-39381181-1');
  </script>
  <meta charset="UTF-8"><script
type="text/javascript">(window.NREUM||(NREUM={})).init={ajax:
{deny_list:["bam.nr-data.net"]};(window.NREUM||(NREUM={})).l
oader_config={licenseKey:"f84b877fa4",applicationID:"307784036"
};window.NREUM||(NREUM={}).__nr_require=function(t,e,n){func
tion r(n){if(!e[n]){var
i=e[n]={exports:{}};t[n][0].call(i.exports,function(e){var
i=t[n][1][e];return r(i||e)},i,i.exports)}return
e[n].exports}if("function"===typeof __nr_require)return
__nr_require;for(var i=0;i<n.length;i++){r(n[i]);return
r}({1:[function(t,e,n){function r(o){function i(t,e,n){return
o(t,[u.now()].concat(f(arguments)),e?null:this,n),e?void 0:this}var
o=t("handle"),a=t(8),f=t(9),c=t("ee").get("tracer"),u=t("loader"),s=N
REUM;"undefined"===typeof window.newrelic&&(newrelic=s);var

```

```
d=["setPageViewName","setCustomAttribute","setErrorHandler","finished","addToTrace","inlineHit","addRelease"],p="api-",l=p+"ixn-";a(d,function(t,e){s[e]=i(p+e,!0,"api")},s.addPageAction=i(p+"addPageAction",!0),s.setCurrentRouteName=i(p+"routeName",!0),e.exports=newRelic,s.interaction=function(){return(new r).get(0);var m=r.prototype={createTracer:function(t,e){var n={,r=this,i="function"==typeof e;return o(1+"tracer",[u.now(),t,n],function(){if(c.emit(i?"":"no-")+"fn-start",[u.now(),r,i],n),i)try{return e.apply(this,arguments)}catch(t){throw c.emit("fn-err",[arguments,this,t],n)}finally{c.emit("fn-end",[u.now(),n])}}};a("actionText,setName,setAttribute,save,ignore,onEnd,getContext,end,get".split(","),function(t,e){m[e]=i(1+e)}),newRelic.noticeError=function(t,e){"string"==typeof t&&(t=new Error(t),o("err",[t,u.now(),!1,e])),{,},2:[function(t,e,n){function r(t){if(NREUM.init){for(var e=NREUM.init,n=t.split("."),r=0;r<n.length-1;r++)if(e=e[n[r]],"object"!==typeof e)return e;return e=e[n[n.length-1]]}e.exports={getConfig:r},{,},3:[function(t,e,n){function r(t){return f.exists&&performance.now?Math.round(performance.now()):o=Math.max((new Date).getTime(),o)-a}function i(o){return o}var o=(new Date).getTime(),a=o,f=(10);e.exports=r,e.exports.offset=a,e.exports.getLastTimestamp=i},{,},4:[function(t,e,n){function r(t){return!(t||t.protocol||"file:"===t.protocol)}e.exports=r},{,},5:[function(t,e,n){function r(t,e){var n=t.getEntries();n.forEach(function(t){"first-paint"===t.name?d("timing",["fp",Math.floor(t.startTime)]:"first-contentful-paint"===t.name&&d("timing",["fcp",Math.floor(t.startTime)]))}function i(t,e){var n=t.getEntries();n.length>0&&d("lcp",[n.length-1])}function o(t){t.getEntries().forEach(function(t){t.hadRecentInput||d("cls",[t])})}function a(t){if(t instanceof m&&!g){var e=Math.round(t.timeStamp),n={type:t.type};e<=p.now()?n.fid=p.now()-e:e=p.offset&&e<=Date.now()?e=p.offset,n.fid=p.now()-e:e=p.now(),g=!0,d("timing",["fi","e,n"])}function f(t){"hidden"===t&&d("pageHide",[p.now()])}if(!("init" in NREUM&&"page_view_timing" in NREUM.init&&"enabled" in NREUM.init.page_view_timing&&NREUM.init.page_view_timing.enabled===!1)){var c,u,s,d=["handle"],p=(("loader"),l=(7),m=NREUM.o.EV;if("PerformanceObserver" in window&&"function"==typeof window.PerformanceObserver){c=new PerformanceObserver(r);try{c.observe({entryTypes:["paint"]})}catch(v){}u=new PerformanceObserver(i);try{u.observe({entryTypes:["largest-contentful-paint"]})}catch(v){}s=new PerformanceObserver(o);try{s.observe({type:"layout-shift",buffered:!0})}catch(v){}if("addEventListener" in document){var g=!1,h=["click","keydown","mousedown","pointerdown","touchstart"];h.forEach(function(t){document.addEventListener(t,a,!1)})(f)},{,},6:[function(t,e,n){function r(t,e){if(!i)return!1;if(!i)return!1;if(!e)return!0;if(!o)return!1;for(var ar=n.o.split("."),r=e.split("."),a=0;a<r.length;a++)if(r[a]!==n[a])return!1;return!0}var i=null,o=null,a=/Version\/(\S+)/s+Safari;if(navigator.userAgent){var f=navigator.userAgent,c=f.match(a);c&&f.indexOf("Chrome")===-1&&f.indexOf("Chromium")===-1&&(i="Safari",o=c[1])}e.exports={agent:i,version:o,match:r},{,},7:[function(t,e,n){function r(t){function e(o){t(a&&document[a]?document[a]:document[i]?"hidden":"visible")}o.addEventListener("in document&&o&&document.addEventListener(o,e,!1)}e.exports=r;var ar=i,o,a;"undefined"!==typeof document.hidden?(i="hidden",o="visibilitychange",a="visibilityState"):"undefined"!==typeof document.msHidden?(i="msHidden",o="msvisibilitychange"):"undefined"!==typeof document.webkitHidden&&(i="webkitHidden",o="webkitvisibilitychange",a="webkitVisibilityState"),{,},8:[function(t,e,n){function r(t,e){var n=[],r="",o=0;for(r=t).call(t,r)&&(n[o]=e(r,t,r),o+=1);return n}var i=Object.prototype.hasOwnProperty;e.exports=r},{,},9:[function(t,e,n){function r(t,e,n){e[e]=0,"undefined"==typeof n&&(n=t.length);for(var r=-1,i=n-e|0,o=Array(i<0?0:i);++r<i;)o[r]=t[e+r];return o}e.exports=r},{,},10:[function(t,e,n){e.exports={exists:"undefined"!==typeof window.performance&&window.performance.timing&&"undefined"!==typeof window.performance.timing.navigationStart}},{,},ee:[function(t,e,n){function r(o){function i(t){function e(t){return t&&t instanceof r?t:t.u(t,c,a):a}function
```

```
n(n,r,i,o,a){if(!a===!1&&(a=!0),!l.aborted|o){t&&a&&(n,r,i);for(var f=e(i),c=v(n),u=c.length,s=0;s<u;s++)c[s].apply(f,r);var p=d[w[n]];return p&&p.push([b,n,r,f],f)}function o(t,e){y[t]=v(t).concat(e)}function m(t,e){var n=y[t];if(n)for(var r=0;r<n.length;r++)n[r]===e&&n.splice(r,1)}function v(t){return y[t]||[]}function g(t){return p[t]=p[t]|i(n)}function h(t,e){l.aborted||s(t,function(t,n){e=e||"feature",w[n]=e,e in d||d[e]=[]})}var y={,w={,b={on:o,addEventListener:o,removeEventListener:m,emit:t,n:get,g.listeners:v,context:e,buffer:h,abort:f,aborted:!1};return b}function o(t){return u(t,c,a)}function a(){return new r}function f(o){d.api||d.feature)&&(l.aborted=!0,d=l.backlog=)}var c="nr@context",u=t("gos"),s=t(8),d={},p={},l=e.exports=i;e.exports.getOrSetContext=o,l.backlog=d,{,},gos:[function(t,e,n){function r(t,e,n){if(i.call(t,e))return t[e];var r=n;if(Object.defineProperty&&Object.keys)try{return Object.defineProperty(t,e,{value:r,writable:!0,enumerable:!1}),r}catch(h){}return t[e]=r}var i=Object.prototype.hasOwnProperty;e.exports=r},{,},handle:[function(n,t,e,n){function r(t,e,n,r){i.buffer([t,r],i.emit(t,e,n))}var i=t("ee").get("handle");e.exports=r,r.ee=i},{,},id:[function(t,e,n){function r(t){var e=typeof t;return!t||"object"!==e&&"function"!==e?-1:t===window?0:t(o,f,unction){return i++}}}var i=0,o="nr@id",a=t("gos");e.exports=r},{,},loader:[function(t,e,n){function r(o){if(R++){var t=M.info=NREUM.info,e=v.getElementsByTagName("script")[0];if(setTimeout(u.abort,3e4),!(t&&t.licenseKey&&t.applicationID&&e))return u.abort();c(E,function(e,n){t[e]||(t[e]=n)});var n=a(o);f("mark",["onload",n+M.offset],null,"api"),f("timing",["load",n]);var r=v.createElement("script");0===t.agent.indexOf("http://")||0===t.agent.indexOf("https://")?r.src=t.agent:r.src=l+":"+t.agent,e.parentNode.insertBefore(r,e)}function i(o){"complete"===v.readyState&&o()function o(){f("mark",["domContent"],a()+M.offset,null,"api")}var a=t(3),f=t("handle"),c=t(8),u=t("ee"),s=t(6),d=t(4),p=t(2),l=p.getConfig(igation("ssl")===!1?"http":"https",m=window,v=m.document,g="addEventListener",h="attachEvent",y=m.XMLHttpRequest,nrWrapper).prototype,b=!d(m.location);NREUM.o={ST:setTimeout,S1:m.setImmediate,CT:clearTimeout,XHR:y,REQ:m.Request,EV:m.Event,PR:m.Promise,MO:m.MutationObserver};var x=""+location,E={beacon:"bam.nr-data.net",errorBeacon:"bam.nr-data.net",agent:"js-agent.newrelic.com/nr-1211.min.js"},O=y&&w&w&w[g]&&!/CriOS/.test(navigator.userAgent),M=e.exports={offset:a.getLastTimestamp(),now:a.origin,x,features:{},xhrWrappable:O,userAgent:s,disabled:b};if(!b){t(1),t(5),v[g]?v[g]({DOMContentLoaded":o,!1}),m[g]({load:r,!1}):v[h]({"onreadystatechange":i}),m[h]({"onload":o}),f("mark",["firstbyte",a.getLastTimestamp()],null,"api");var R=0},{,},"wrap-function":[function(t,e,n){function r(t,e){function o(e,n,r,c,u){function nrWrapper(o){var a,s,p;try{a=this,o=d(arguments),s="function"==typeof r?r(o,a,r)|{}:{}catch(l){i(l,[],[o,a,c],s,t)}f(n+"start",[o,a,c],s,u);try{return p=e.apply(a,o)}catch(m){throw f(n+"err",[o,a,m],s,u)}finally{f(n+"end",[o,a,p],s,u)}return a(e):e(n)}nrWrapper[p]=e,o(e,nrWrapper,t,nrWrapper)}function r(t,e,r,i,o){r||("r"===t);var f,c,u,s="";for(u=0;u<e.length;u++)c=e[u],f=t[c],a(f)(f(c)=n(f,s?c+r:i,c,o))}function f(n,r,o,a){if(!m[e])var t=(t=s).n.inPlace-r.n.flag=p,n}function i(t,e){e||e=s;try{e.emit("internal-error",t)}catch(n){}}function o(t,e,n){if(Object.defineProperty&&Object.keys)try{var r=Object.keys(t);return r.forEach(function(n){Object.defineProperty(e,n,{get:function(){return t[n]},set:function(e){return t[n]=e,e}})}),e}catch(o){i([o],n)}for(var a in t).call(t,a)&&(e[a]=t[a]);return e}function a(t){return!(t&&t instanceof Function&&t.apply&&t.call)}function f(t,e){var n=e(t);return n[p]=t,o(t,n,s)}function c(t,e,n){var r=t[e];t[e]=f(r,n)}function u(o){for(var t=arguments.length,e=new Array(t),n=0;n<t;++n)e[n]=arguments[n];return e}var s=t("ee"),d=t(9),p="nr@original",l=Object.prototype.hasOwnProperty,m=!1;e.exports=r,e.exports.wrapFunction=f,e.exports.wrapInPlace=c,e.exports.argsToArray=u},{,},{,},l["loader"]];</script><meta name="viewport" content="width=device-width,initial-scale=1"><meta name="csrf-param" content="_csrf-cloud"><meta name="csrf-token" content="gRRXwuGfbAsGnJQBuxulfNZbAnASiaGPhhQtINk2tIb1ex-jqNcVQ0jR1meITdE4kswGn_HiPE0bV934suc7Q=="><title></title>
```

```
<link href="/assets/8d0ea680/fullcalendar.print.css"
rel="stylesheet" media="print">
<link href="/assets/8d0ea680/fullcalendar.min.css"
rel="stylesheet">
<link href="/assets/89f520a0/scheduler.css"
rel="stylesheet">
<link href="/assets/720ff176/css/font-awesome.min.css"
rel="stylesheet">
<link href="/assets/3c97b4c/css/bootstrap.css"
rel="stylesheet">
<link href="/assets/7ad1e5dc/css/AdminLTE.min.css"
rel="stylesheet">
<link href="/assets/7ad1e5dc/css/skins/_all-skins.min.css"
rel="stylesheet">
<link
href="/assets/2cd327/media/css/jquery.dataTables.css"
rel="stylesheet">
<link
href="/assets/1722ec01/css/kv-bootstrap-notify.min.css"
rel="stylesheet">
<link href="/css/site.css" rel="stylesheet"></head>
<body class="hold-transition skin-default sidebar-mini">
<div class="wrapper">

<header class="main-header hidden-xs">

  <a class="logo" href="/"><span
class="logo-mini">CRM</span><span class="logo-lg">MediSpark
<strong>CRM</strong></span></a>

  <div class="btn-premium-plan">
    <a class="btn btn-xs btn-warning"
href="/billing/default/index">pro</a> </div>
  <nav class="navbar navbar-static-top"
role="navigation">

    <a href="#" class="sidebar-toggle"
data-toggle="push-menu" role="button">
      <span class="sr-only">Toggle navigation</span>
    </a>

    <a class="mobile-header-button glyphicon
glyphicon-user" href="/"></a> <a class="mobile-header-button
glyphicon glyphicon-calendar" href="/app-calendar/index"></a>
    <a class="mobile-header-button glyphicon
glyphicon-picture" href="/storage/files/index"></a>
    <div class="navbar-custom-menu">

      <ul class="nav navbar-nav">
        <li class="dropdown user user-menu">
          <a href="#" class="dropdown-toggle" data-
toggle="dropdown">
             <span
class="hidden-xs">Іглінський Андрій</span>
          </a>
          <ul class="dropdown-menu">
            <!-- User image -->
            <li class="user-header">
               <p>
                Іглінський Андрій
              </p>
            </li>
            <!-- Menu Body -->
            <li class="user-body">
            </li>
            <!-- Menu Footer -->
            <li class="user-footer">
              <div class="pull-left">
                <div class="pull-right">
                  <a class="btn btn-default btn-flat"
href="/auth/logout">Вихід</a>
                </div>
              </li>
            </ul>
          </div>
        </li>
      </ul>
    </div>

    <aside class="main-sidebar">
      <section class="sidebar">
        <div class="sidebar-new-patient">
          <a class="btn btn-success btn-block new-
patient-fast-button" href="/client/create"><span
class="hide-on-collapsed">Новий пацієнт</span><span
class="glyphicon glyphicon-plus show-on-collapsed"></span></a>
          </div>

          <ul class="sidebar-menu tree main"
data-widget="tree"><li><a href="/"><i class="fa fa-users"></i>
<span>Пацієнти</span></li>
            <li class="active"><a href="/app-calendar/index"><i
class="fa fa-calendar"></i> <span>Розклад</span></a></li>
            <li><a href="/appointment/index"><i class="fa
fa-book"></i> <span>Записи</span></a></li>
            <li><a href="/reports/index"><i class="fa
fa-bar-chart"></i> <span>Аналітика</span></a></li>
            <li><span><hr></span></li>
            <li class="small-menu-item treeview"><a href="#"><i
class="fa fa-gears"></i> <span>Налаштування</span> <span>
```

```
</nav>
</header>

<header class="main-header mobile visible-xs-block">

  <nav class="navbar navbar-static-top"
role="navigation">

    <a href="#" class="sidebar-toggle"
data-toggle="push-menu" role="button">
      <span class="sr-only">Toggle navigation</span>
    </a>

    <a class="mobile-header-button glyphicon
glyphicon-user" href="/"></a>
    <a class="mobile-header-button glyphicon
glyphicon-calendar" href="/app-calendar/index"></a>
    <a class="mobile-header-button glyphicon
glyphicon-picture" href="/storage/files/index"></a>
    <a class="mobile-header-button glyphicon
glyphicon-book" href="/requests/default/index"></a>
    <div class="navbar-custom-menu">

      <ul class="nav navbar-nav">
        <li class="dropdown user user-menu">
          <a href="#" class="dropdown-toggle" data-
toggle="dropdown">
             <span
class="hidden-xs">Іглінський Андрій</span>
          </a>
          <ul class="dropdown-menu">
            <!-- User image -->
            <li class="user-header">
               <p>
                Іглінський Андрій
              </p>
            </li>
            <!-- Menu Body -->
            <li class="user-body">
            </li>
            <!-- Menu Footer -->
            <li class="user-footer">
              <div class="pull-left">
                <div class="pull-right">
                  <a class="btn btn-default btn-flat"
href="/auth/logout">Вихід</a>
                </div>
              </li>
            </ul>
          </div>
        </li>
      </ul>
    </div>

    <aside class="main-sidebar">
      <section class="sidebar">
        <div class="sidebar-new-patient">
          <a class="btn btn-success btn-block new-
patient-fast-button" href="/client/create"><span
class="hide-on-collapsed">Новий пацієнт</span><span
class="glyphicon glyphicon-plus show-on-collapsed"></span></a>
          </div>

          <ul class="sidebar-menu tree main"
data-widget="tree"><li><a href="/"><i class="fa fa-users"></i>
<span>Пацієнти</span></li>
            <li class="active"><a href="/app-calendar/index"><i
class="fa fa-calendar"></i> <span>Розклад</span></a></li>
            <li><a href="/appointment/index"><i class="fa
fa-book"></i> <span>Записи</span></a></li>
            <li><a href="/reports/index"><i class="fa
fa-bar-chart"></i> <span>Аналітика</span></a></li>
            <li><span><hr></span></li>
            <li class="small-menu-item treeview"><a href="#"><i
class="fa fa-gears"></i> <span>Налаштування</span> <span>
```

```

class="pull-right-container"><i class="fa fa-angle-left
pull-right"></i></span></a>
  <ul class="treeview-menu" >
    <li><a href="/service/index"><i class="fa
fa-user-md"></i> <span>Послуги</span></a></li>
    <li><a href="/users"><i class="fa fa-user-md"></i>
<span>Працівники</span></a></li>
    <li><a href="/clinic/view"><i class="fa fa-home"></i>
<span>Клініка</span></a></li>
  </ul>
</li>
<li class="small-menu-item"><a
href="/user/update?id=120"><i class="fa fa-wrench"></i>
<span>Мій профіль</span></a></li>
</div>
</section>
</aside>

<div class="modal fade" tabindex="-1" role="dialog">
  <div class="modal-dialog" role="document">
    <div class="modal-content">
      <div class="modal-header">
        <button type="button" class="close"
data-dismiss="modal" aria-label="Close"><span
aria-hidden="true">&times;</span>
        </button>
        <h4 class="modal-title">Modal title</h4>
      </div>
      <div class="modal-body">
        <p>One fine body&hellip;</p>
      </div>
      <div class="modal-footer">
        <button type="button" class="btn btn-default"
data-dismiss="modal">Close</button>
        <button type="button" class="btn
btn-primary">Save changes</button>
      </div>
    </div><!-- /.modal-content -->
  </div><!-- /.modal-dialog -->
</div><!-- /.modal -->

  <div class="content-wrapper">
    <section class="content">
      <div class="alert alert-info visible-xs-block">
        <p>Щоб додати прийом записніть місце.</p>
      </div>
      <div id="event_calendar" class="fullcalendar" lang="ru"
data-plugin-name="fullCalendar">
        <div class="fc-loading" style="display:none;">Loading
      </div>
    </div>

    <style>

      th[data-resource-id="u-121"] {
        background-color: #0011ff }

      th[data-resource-id="u-122"] {
        background-color: #ff0000 }

      th[data-resource-id="u-123"] {
        background-color: #1e00ff }

      th[data-resource-id="u-124"] {
        background-color: #00aaff }

      th[data-resource-id="u-125"] {
        background-color: #fbff00 }

      .fc-body tr[data-time="0:00:00"] td {
        border-top: 2px solid #0f0f0f !important;
      }

      .fc-body tr[data-time="1:00:00"] td {
        border-top: 2px solid #0f0f0f !important;
      }

      .fc-body tr[data-time="2:00:00"] td {
        border-top: 2px solid #0f0f0f !important;
      }

      .fc-body tr[data-time="3:00:00"] td {
        border-top: 2px solid #0f0f0f !important;
      }

      .fc-body tr[data-time="4:00:00"] td {
        border-top: 2px solid #0f0f0f !important;
      }

      .fc-body tr[data-time="5:00:00"] td {
        border-top: 2px solid #0f0f0f !important;
      }

      .fc-body tr[data-time="6:00:00"] td {
        border-top: 2px solid #0f0f0f !important;
      }

      .fc-body tr[data-time="7:00:00"] td {
        border-top: 2px solid #0f0f0f !important;
      }

      .fc-body tr[data-time="8:00:00"] td {
        border-top: 2px solid #0f0f0f !important;
      }

      .fc-body tr[data-time="9:00:00"] td {
        border-top: 2px solid #0f0f0f !important;
      }

      .fc-body tr[data-time="10:00:00"] td {
        border-top: 2px solid #0f0f0f !important;
      }

      .fc-body tr[data-time="11:00:00"] td {
        border-top: 2px solid #0f0f0f !important;
      }

      .fc-body tr[data-time="12:00:00"] td {
        border-top: 2px solid #0f0f0f !important;
      }

      .fc-body tr[data-time="13:00:00"] td {
        border-top: 2px solid #0f0f0f !important;
      }

      .fc-body tr[data-time="14:00:00"] td {
        border-top: 2px solid #0f0f0f !important;
      }

      .fc-body tr[data-time="15:00:00"] td {
        border-top: 2px solid #0f0f0f !important;
      }

      .fc-body tr[data-time="16:00:00"] td {
        border-top: 2px solid #0f0f0f !important;
      }

      .fc-body tr[data-time="17:00:00"] td {
        border-top: 2px solid #0f0f0f !important;
      }

      .fc-body tr[data-time="18:00:00"] td {
        border-top: 2px solid #0f0f0f !important;
      }

      .fc-body tr[data-time="19:00:00"] td {
        border-top: 2px solid #0f0f0f !important;
      }

      .fc-body tr[data-time="20:00:00"] td {
        border-top: 2px solid #0f0f0f !important;
      }

      .fc-body tr[data-time="21:00:00"] td {
        border-top: 2px solid #0f0f0f !important;
      }

```

```

.fc-body tr[data-time="22:00:00"] td {
border-top: 2px solid #0f0f0f !important;
}

.fc-body tr[data-time="23:00:00"] td {
border-top: 2px solid #0f0f0f !important;
}

</style>
</section>
</div>
<div class='control-sidebar-bg'></div> </div>

<div id="modal" class="fade modal" role="dialog">
<div class="modal-dialog">
<div class="modal-content">
<div class="modal-header">
<button type="button" class="close"
data-dismiss="modal" aria-hidden="true">&times;</button>
<h4 class="modal-title"></h4>
</div>
<div class="modal-body">

</div>
</div>
</div>
</div>

<div id="modal-file-upload" class="fade modal"
role="dialog">
<div class="modal-dialog">
<div class="modal-content">
<div class="modal-header">
<button type="button" class="close"
data-dismiss="modal" aria-hidden="true">&times;</button>
<h4 class="modal-title">Uploading...</h4>
</div>
<div class="modal-body">

</div>
</div>
</div>
</div>
</div>
<script src="/assets/c5025936/jquery.js"></script>
<script src="/assets/6095f25d/yii.js"></script>
<script src="/assets/8548e316/moment.js"></script>
<script src="/assets/8d0ea680/fullcalendar.js"></script>
<script src="/assets/8d0ea680/locale-all.js"></script>
<script src="/assets/8d0ea680/locale/ru.js"></script>
<script src="/assets/89f520a0/scheduler.js"></script>
<script src="/assets/3c97b4c/js/bootstrap.js"></script>
<script
src="/assets/7ad1e5dc/js/adminlte.min.js"></script>
<script
src="/assets/2cd327/media/js/jquery.dataTables.js"></script>
<script
src="/assets/1722ec01/js/bootstrap-notify.min.js"></script>
<script src="/js/common.js"></script>
<script>jQuery(function ($) {
var loading_container = jQuery("#event_calendar
.fc-loading");
jQuery("#event_calendar").empty().append(loading_contai
ner);
jQuery("#event_calendar").fullCalendar({ "loading": functio
n(isLoading, view) {
jQuery("#event_calendar").find('.fc-loading').tog
gle(isLoading);
}, "themeSystem": "bootstrap3", "eventDrop": function(event,
revertFunc, resource) {
var resourceId = event.resourceId;
var url =
'/app-calendar/event-time-updated' + '?id=' + event.id + '&newStart='

```

```

+ event.start.format() + '&duration=' + delta.asSeconds() +
'&newResourceId=' + resourceId;
$.ajax({
url: url,
type: 'POST',
success: function (data) {
notify(data.notify, 'success');
},
error: function(jqXHR, errMsg) {
console.error(errMsg);
notify(data.notify, 'danger');
revertFunc();
}
});

}, "eventResize": function(event, delta,
revertFunc) {
var url =
'/app-calendar/event-time-updated' + '?id=' + event.id + '&newEnd='
+ event.end.format();
$.ajax({
url: url,
type: 'POST',
success: function (data) {
notify(data.notify, 'success');
},
error: function(jqXHR, errMsg) {
console.error(errMsg);
notify(data.notify, 'danger');
revertFunc();
}
});
}, "select": function( start, end, jsEvent,
view, resource) {
var ajaxUrl =
'/app-calendar/new-event-to-user?start=' + start + '&end=' + end +
'&resource=' + resource.id ;
$.showModal(ajaxUrl, true);
}, "eventClick": function(event, jsEvent, view ) {
var ajaxUrl = '/app-calendar/detail-event' + '?id=' +
event.id;
$.showModal(ajaxUrl, true);
}, "events": "/app-calendar/json-events-by-user", "grou
pByDateAndResource": true, "resources": [{"id": "u-121", "title": "Гуцу
ляк К.В.\r\nДорослий стоматолог"}, {"id": "u-122", "title": "Пак
А.В.\r\nДитячий стоматолог"}, {"id": "u-123", "title": "Вітковська
В.В.\r\nДорослий стоматолог"}, {"id": "u-124", "title": "Гладких
С.Г.\r\nОртодонт"}, {"id": "u-125", "title": "Флагман
І.І.\r\nПародонтолог"}], "defaultView": "agendaDay", "selectable": tru
e, "weekends": true, "editable": true, "allDaySlot": false, "views": {"agend
aFiveDays": {"type": "agenda", "duration": {"days": 5}, "buttonText": "5
дней"}, {"minTime": "07:00:00", "maxTime": "21:00:00", "businessHo
urs": [{"dow": [0], "start": "10:00", "end": "18:00", "inactive": false}, {"do
w": [1], "start": "08:00", "end": "20:00", "inactive": false}, {"dow": [2], "st
art": "08:00", "end": "20:00", "inactive": false}, {"dow": [3], "start": "08:0
0", "end": "20:00", "inactive": false}, {"dow": [4], "start": "08:00", "end": "
20:00", "inactive": false}, {"dow": [5], "start": "08:00", "end": "20:00", "in
active": false}, {"dow": [6], "start": "10:00", "end": "18:00", "inactive": fal
se}], "nowIndicator": true, "slotDuration": "00:15:00", "slotLabelInterva
l": "00:30:00", "slotLabelFormat": "H:mm", "viewRender": function(
view, element ) {
$('#event_calendar').fullCalendar
(option, 'height', window.innerHeight - 50);
}, "header": {"center": "title", "left": "p
rev, next today", "right": "agendaDay, agendaFiveDays, listWeek"} });
jQuery("#modal").modal({"show": false});
jQuery("#modal-file-upload").modal({"show": false});
});</script><script
type="text/javascript">window.NREUM||(NREUM={});NREUM.inf
o={"beacon":"bam.nr-data.net","licenseKey":"f84b877fa4","applicati
onID":"307784036","transactionName":"MQNaMBADRAHAEDQ
DQhJbRYLTQoNAGZLFxIOFg==","queueTime":0,"applicationTime
":215,"atts":{"HURZRlgZHh4=","errorBeacon":"bam.nr-data.net","ag
ent":""}}</script></body>
</html>

Лістинг файлу common.js
function notify(message, type, delay) {
$.notify({
message: message || 'Добавлено!'
}, {
type: type || 'info',

```

```

        delay: delay || 2
    });
}

(function ($) {
    $.extend({
        showModal: function (url, ajaxForm) {
            $.ajax({
                url: url,
                type: 'GET',
                success: function (data) {
                    if (typeof data.result !== 'undefined') {
                        $('#modal
.modal-title').html(data.result.title);
                        $('#modal
.modal-body').html(data.result.content);
                        $('#modal').modal('show');
                        if (ajaxForm === true) {
                            $.modalAjaxFormAction($('#modal'));
                        }
                    }
                },
                error: function (jqXHR, errMsg) {
                    console.error(errMsg);
                }
            });
        },

        modalAjaxFormAction: function (modal) {
            var modalForm = modal.find('form');

            if (typeof modalForm !== 'undefined') {
                modalForm.on('beforeSubmit', function (e) {
                    e.preventDefault();
                    var form = $(this),
                        data = form.serializeArray(),
                        method = form.attr('method'),
                        action = form.attr('action');

                    $.ajax({
                        url: action,
                        type: method,
                        data: data,
                        success: function (data) {
                            if (typeof data.validation !==
'undefined') {
                                form.yiiActiveForm('updateMessages'
, data.validation, true);
                            } else {
                                if (typeof data.result !== 'undefined') {
                                    $('#modal
.modal-title').html(data.result.title);
                                    $('#modal
.modal-body').html(data.result.content);
                                    if (typeof data.result.reload !==
'undefined') {
                                        if ($.pjax) {
                                            $.pjax.reload({container:
data.result.reload});
                                        }
                                        //refetch events of calendar
                                        if ($.fullCalendar) {
                                            $('#event_calendar').fullCalend
ar("refetchEvents");
                                        }
                                    }
                                }

                                notify(data.notify, 'success');
                                $('#modal').modal('hide');
                            }
                        }
                    });
                },

                (function () {
                    $(document).ready(function () {
                        $('#modal').on('hidden.bs.modal', function (e) {
                            $('#modal .modal-title').html("");
                            $('#modal .modal-body').html("");
                        });

                        $(document).on('click', '[data-modal-form]', function
(e) {
                            e.preventDefault();
                            $.showModal($(this).attr('data-modal-form'), true);
                        });

                        $(document).on('click', '[data-modal]', function (e) {
                            e.preventDefault();
                            $.showModal($(this).attr('data-modal'), true);
                        });

                        $(document).on('change', '#calendar-widget-form',
function (e) {
                            $('#calendar-widget-form').submit();
                        });

                        $(document).on('click', '.remove-table-row', function
(e) {
                            $(this).closest('tr').remove();
                            return false;
                        });

                        $(document).on('click', '.remove-on-click', function
(e) {
                            $(this).remove();
                            return false;
                        });

                        $(document).on('click', '.remove-parent-on-click',
function (e) {
                            $(this).parent().remove();
                            return false;
                        });

                        $("form.comment-ajax-form").submit(function
(event) {
                            event.preventDefault(); // stopping submitting
                            event.stopImmediatePropagation();
                            var _this = this;
                            var data = $(this).serializeArray();
                            var url = $(this).attr('action');
                            $.ajax({
                                url: url,
                                type: 'post',
                                dataType: 'json',
                                data: data
                            })
                            .done(function (response) {
                                if (response.result.success === true) {
                                    if ($.pjax) {
                                        var pjaxContainer =
                    ($(_this).closest('.client-comment-box').find('[data-pjax-container]').at
tr('id')).pjaxContainer;
                                        $.pjax.reload({container: pjaxContainer
+
                    $('#'+
                    pjaxContainer));
                                    }
                                    $(_this).trigger("reset");
                                }
                            })
                            .fail(function () {
                                console.log("error");
                            });

                            return false;
                        });
                    })(jQuery);

                    error: function (jqXHR, errMsg) { notify(errMsg, 'danger'); console.error(errMsg);
                }
            }
        },
    });
});
return false;
});

```


Додаток Г
(обов'язковий)

ПРОТОКОЛ
ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ
НА НАЯВНІСТЬ ТЕКСТОВИХ ЗАПОЗИЧЕНЬ

Назва роботи: «Розробка автоматизованої системи управління електронною чергою медичного закладу»

Тип роботи: Магістерська кваліфікаційна робота
(БДР, МКР)

Підрозділ АПТ, ФІТА
(кафедра, факультет)

Показники звіту подібності Unicheck

Оригінальність 98.5% Схожість 1.5%

Аналіз звіту подібності (відмітити потрібне)

- Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.
- Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її автора. Роботу направити на розгляд експертної комісії кафедри.
- Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Особа, відповідальна за перевірку Роман МАСЛІЙ
(підпис) (прізвище, ініціали)

Ознайомлені з повним звітом подібності, який був згенерований системою Unicheck щодо роботи.

Автор роботи Максим КОНЕЦУЛ
(підпис) (прізвище, ініціали)

Керівник роботи Марія ЮХИМЧУК
(підпис) (прізвище, ініціали)

