

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

ОСВІТНЬО-ПРОФЕСІЙНА ПРОГРАМА «КОМП'ЮТЕРНІ НАУКИ»
галузь знань 12 «Інформаційні технології»
спеціальність 122 «Комп'ютерні науки»

Форма навчання: денна

КВАЛІФІКАЦІЙНА БАКАЛАВРСЬКА РОБОТА

на тему: «РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ СТОМАТОЛОГІЧНОЇ
КЛІНІКИ »

здобувача Садуллаєва Владислава Едуардовича

(ПБ)

_____ (Підпис)

Науковий керівник:
д.ф-м.н., професор
Миронцов М. Л.

**Робота допущена до захисту перед
екзаменаційною комісією з атестації
здобувачів вищої освіти
завідувач кафедри:**
к.е.н., доцент
_____ Тішков Б.О.

Київ 2025

Міністерство освіти і науки України
Київський національний економічний університет імені Вадима Гетьмана
Навчально-науковий інститут «Інститут інформаційних технологій в економіці»

Кафедра інформаційних систем в економіці

ОСВІТНЬО-ПРОФЕСІЙНА ПРОГРАМА «КОМП'ЮТЕРНІ НАУКИ»

галузь знань 12 «Інформаційні технології»

спеціальність 122 «Комп'ютерні науки»

ПОГОДЖЕНО:

Керівник проєктної групи(гарант)
освітньо-професійної програми

_____Помазун О.М.

“ _____ ” _____ 2025 р.

ЗАТВЕРДЖУЮ:

Завідувач кафедри

_____Тішков Б.О.

“ _____ ” _____ 2025 р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

здобувача вищої освіти Садуллаєв Владислав Едуардович

очної (денної) форми навчання

на підготовку кваліфікаційної бакалаврської роботи
на тему: «Розробка інформаційної системи стоматологічної клініки»

Тему затверджено наказом ректора Університету від « 7 » березня 2025 р.
№ 466-ст.

Кваліфікаційна бакалаврська робота виконується на матеріалах
Отриманих з мережі Інтернет, відкритих джерел інформації

План кваліфікаційної бакалаврської роботи

Розділ I Характеристика та аналіз предметної галузі

Розділ II Розробка вимог і моделювання інформаційної системи

Розділ III Проєктування та реалізація компонентів системи

Об'єкт дослідження процес розробки інформаційної системи для зубної клініки.

Предмет дослідження методи, засоби та технології проєктування інформаційної системи для зубної клініки, що забезпечують автоматизацію її ключових бізнес-процесів та підвищення ефективності роботи.

Мета кваліфікаційної бакалаврської роботи полягає в закріпленні та поглибленні знань шляхом розробки концепції та проєктних рішень для інформаційної системи, що дозволить автоматизувати ключові процеси клініки, покращити якість обслуговування пацієнтів та оптимізувати управлінську діяльність

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

У розділі I провести аналіз предметної області, дослідити існуючі програмні рішення для автоматизації стоматологічних клінік та виявити їх переваги й недоліки.

У розділі II визначити бізнес-вимоги, а також функціональні та нефункціональні вимоги до системи; розробити моделі поведінки, структури та користувацького інтерфейсу системи; створити інтерактивний прототип основних функціональних компонентів.

У розділі III обґрунтувати вибір інформаційного, технічного та програмного забезпечення; спроектувати структуру бази даних та архітектуру системи; реалізувати ключові компоненти веб-додатку.

**Завдання підготував
науковий керівник**

_____ **Миронцов Микита Леонідович**
« 10 » березня 2025 р.

**Завдання одержав
здобувач**

_____ **Садуллаєв Владислав Едуардович**
« 10 » березня 2025 р.

Відгук
 про кваліфікаційну бакалаврську роботу
 здобувача навчально-наукового інституту
 «Інститут інформаційних технологій в економіці»
 освітньо-професійної програми
 «Комп'ютерні науки»

Садуллаєва Владислава Едуардовича

на тему

«Розробка інформаційної системи стоматологічної клініки»

1. Актуальність теми:

Актуальність зумовлена необхідністю автоматизації бізнес-процесів стоматологічних клінік для підвищення ефективності роботи, якості обслуговування та конкурентоспроможності.

2. Позитивні риси кваліфікаційної бакалаврської роботи:

Проведено аналіз предметної галузі та існуючих рішень. Розроблено бізнес-, функціональні та нефункціональні вимоги. Розроблено інтерактивний прототип. Обґрунтовано вибір технологічного стеку та спроектовано структуру бази даних. Реалізовано функціональний прототип веб-додатку.

3. Наявність самостійних розробок автора:

Робота виконана автором самостійно. Самостійно розроблено концепцію та проєктні рішення ІС стоматологічної клініки. Створено системну діаграму ОРМ, діаграми бізнес-вимог та функціональних вимог, а також діаграму нефункціональних вимог. Побудовано діаграми прецедентів та послідовностей UML для ключових сценаріїв. Розроблено інфологічну та даталогічну моделі БД. Проведено практичну реалізацію ключових компонентів веб-додатку.

4. Цінність теоретичних висновків та практичних рекомендацій:

Практична цінність полягає у створенні функціонального прототипу ІС, який є міцним фундаментом для подальшої реалізації повнофункціонального веб-додатку. Впровадження такої системи сприятиме підвищенню ефективності роботи персоналу, зменшенню ймовірності помилок у веденні документації та покращенню якості обслуговування пацієнтів, що має значне практичне значення для медичних закладів.

5. Наявність недоліків:

Математичний опис задачі обмежується лише двома базовими формулами розрахунку показників.

6. Загальна оцінка кваліфікаційної бакалаврської роботи та її допущення до захисту перед ЕК:

Робота Садуллаєва В. Е. відповідає вимогам щодо написання та оформлення кваліфікаційних бакалаврських робіт і може бути допущена до захисту.

Науковий керівник

(підпис)

професор, д.ф-м.н.Миронцов М.Л.

“ ____ ” _____ 2025 р.

Рецензія
на кваліфікаційну бакалаврську роботу
здобувача вищої освіти
Садуллаєва Владислава Едуардовича
(прізвище, ім'я, по батькові)

Тема: «Розробка інформаційної системи стоматологічної клініки»

Актуальність теми кваліфікаційної бакалаврської роботи і доцільність її розроблення:

Тема кваліфікаційної роботи є актуальною, оскільки сучасні медичні установи, потребують автоматизації для ефективного управління пацієнтами, записами, послугами та обліком. Розроблення такої системи є доцільним для оптимізації робочих процесів, підвищення якості обслуговування та впорядкування внутрішньої діяльності клініки.

Якість проведеного дослідження:

Проведене дослідження виконано на високому рівні. У роботі детально проаналізовано предметну область, визначено вимоги до інформаційної системи, розроблено прототип та реалізацію системи.

Позитивні риси кваліфікаційної бакалаврської роботи:

Робота має чітку структуру та логічний виклад матеріалу. Практична реалізація системи, включаючи розробку API та моделей даних для пацієнтів, послуг та записів, підтверджує високий рівень практичних навичок здобувача.

Зауваження:

Бажано було б докладніше розглянути питання інтеграції з існуючими медичними системами та стандартами обміну даними. Також можна було б розширити аналіз аспектів кібербезпеки системи.

Практична значимість висновків і рекомендацій:

Розроблена інформаційна система здатна суттєво оптимізувати роботу стоматологічної клініки, автоматизуючи ключові процеси та покращуючи взаємодію з пацієнтами.

Місце роботи та посада рецензента

Науковий ступінь, учене звання (за наявності)

Підпис засвідчую:

Вікторія секретар
(посада, підпис)

Інститут телекомунікацій і глобального інформаційного простору
д.т.н., професор, завідувач відділу
В.М. Трищенко
(підпис, ПІБ)

Місце печатки організації, де працює рецензент



АНОТАЦІЯ

кваліфікаційної бакалаврської роботи
здобувача першого (бакалаврського) рівня вищої освіти, 4 курсу,
виконаної на тему: «Розробка інформаційної системи стоматологічної
клініки»

Київ: кафедра інформаційних систем в економіці, 2025 р.

Кваліфікаційна бакалаврська робота присвячена проєктуванню та розробці інформаційної системи для стоматологічної клініки, спрямованої на автоматизацію ключових бізнес-процесів та підвищення якості обслуговування пацієнтів. Робота складається зі вступу, трьох розділів, висновків та додатків. У першому розділі проведено аналіз предметної галузі, досліджено існуючі інформаційні системи у сфері стоматології та виявлено їхні переваги та недоліки. У другому розділі детально розроблено бізнес-, функціональні та нефункціональні вимоги, виконано моделювання системи за допомогою нотацій UML, OPM та BPMN, а також створено інтерактивний прототип користувацького інтерфейсу. Третій розділ присвячений опису проєктних рішень щодо інформаційного, технічного та програмного забезпечення, обґрунтовано вибір технологічного стеку (MongoDB, Node.js), спроектовано структуру бази даних та продемонстровано результати реалізації функціонального прототипу. Робота завершується узагальненням отриманих результатів та визначенням напрямів для подальшого розвитку системи.

РЕФЕРАТ

Кваліфікаційна бакалаврська робота містить 63 сторінок, 5 таблиць, 34 рисунків, список використаних джерел з 30 найменувань, 2 додатків.

«Тема роботи»

Об'єкт дослідження: процес розробки інформаційної системи для зубної клініки.

Предмет дослідження методи, засоби та технології проектування інформаційної системи для зубної клініки, що забезпечують автоматизацію її ключових бізнес-процесів та підвищення ефективності роботи.

Мета і завдання дослідження. Мета роботи полягає в закріпленні та поглибленні знань шляхом розробки концепції та проектних рішень для інформаційної системи, що дозволить автоматизувати ключові процеси клініки, покращити якість обслуговування пацієнтів та оптимізувати управлінську діяльність.

Завдання:

- 1) провести моделювання предметної області стоматологічної клініки;
- 2) визначити бізнес-вимоги, а також функціональні та нефункціональні вимоги до системи;
- 3) розробити моделі поведінки, структури та користувацького інтерфейсу системи;
- 4) створити інтерактивний прототип основних функціональних компонентів інформаційної системи;
- 5) розробити тест-кейси для перевірки відповідності системи вимогам та провести трасування вимог.

Теоретична, методична та практична значущість отриманих результатів. Одержані результати можуть бути використані для подальшої реалізації повнофункціонального веб-додатку ІС управління зубною клінікою. Впровадження такої системи сприятиме підвищенню ефективності роботи персоналу, зменшенню ймовірності помилок у веденні документації та покращенню якості обслуговування пацієнтів, що має значне практичне значення для медичних закладів.

Рік виконання кваліфікаційної бакалаврської роботи – 2025.

Рік захисту роботи – 2025.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ	3
ВСТУП	4
РОЗДІЛ 1. ХАРАКТЕРИСТИКА ТА АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ	7
1.1 Характеристика предметної галузі та об'єкта дослідження	7
1.2 Аналіз літературних джерел та практичного досвіду використання ІС і технологій в предметній галузі	8
РОЗДІЛ 2. РОЗРОБКА ВИМОГ І МОДЕЛЮВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ.....	12
2.1 Аналіз і специфікація вимог до інформаційної системи.....	12
2.2 Постановка та алгоритм розв'язання задачі	18
2.2.1 Характеристика вхідної та вихідної інформації	19
2.2.2 Алгоритм розв'язання задачі	22
2.3 Моделювання інформаційної системи	25
2.3.1 Моделювання поведінки системи	26
2.3.2 Моделювання структури системи	28
2.3.3 Розподіл вимог за компонентами системи.....	30
2.3.4 Моделювання та прототипування користувацького інтерфейсу	31
РОЗДІЛ 3. ПРОЄКТУВАННЯ ТА РЕАЛІЗАЦІЯ КОМПОНЕНТІВ СИСТЕМИ.....	41
3.1 Інформаційне забезпечення	41
3.1.1 Загальна характеристика інформаційного забезпечення	41
3.1.2 Організація збору і передавання первинної інформації	42

3.1.3 Побудова системи класифікації та кодування	42
3.1.4 Проєктування форм первинних документів та відеокадрів	43
3.1.5 Структура інформаційних масивів	45
3.1.6 Вибір СКБД	45
3.1.7 Інфологічна модель бази даних	46
3.2 Технічне забезпечення	49
3.2.1 Загальні положення та схема автоматизації	49
3.2.2 Структура комплексу технічних засобів та 3.2.3 Опис автоматизованого робочого місця (АРМ).....	50
3.2.3 Схема мережі передачі даних	50
3.3 Програмне забезпечення	51
3.3.1 Структура програмного забезпечення.....	51
3.3.2 Системне програмне забезпечення.....	52
3.3.3 Прикладне програмне забезпечення.....	52
3.3.4 Програмна документація.....	53
3.4 Результати реалізації інформаційної системи	53
ВИСНОВОК	57
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	59
ДОДАТКИ	62

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

ІС – інформаційна система

БД – база даних

МІС – медична інформаційна система

UML – уніфікована мова моделювання (Unified Modeling Language)

BPMN – нотація та модель бізнес-процесів (Business Process Model and Notation)

DFD – діаграма потоків даних (Data Flow Diagram)

IDEF0 – методологія функціонального моделювання (Integration Definition for Function Modeling)

IDEF3 – методологія моделювання для опису сценаріїв та послідовностей процесів (Integration Definition for Process Description Capture Method)

OPM – об'єктно-процесна методологія (Object-Process Methodology)

SysML – мова системного моделювання (Systems Modeling Language)

ПЗ – програмне забезпечення

ВСТУП

У сучасних умовах динамічного розвитку цифрових технологій, заклади охорони здоров'я, зокрема стоматологічні клініки, стикаються з необхідністю адаптації до змін середовища через впровадження інноваційних управлінських рішень. Цифрова трансформація бізнес-процесів є одним із ключових чинників підвищення конкурентоспроможності медичних установ. Організація якісного обслуговування пацієнтів, ефективне керування великими обсягами інформації та оптимізація внутрішніх процесів стають ключовими завданнями для успішної діяльності таких закладів.

Актуальність теми. Зростаючий попит на стоматологічну допомогу та потреба пацієнтів у зручному доступі до своїх медичних даних підкреслюють актуальність розробки спеціалізованих інформаційних систем (ІС). Впровадження таких систем дозволяє автоматизувати рутинні операції, мінімізувати людський фактор, підвищити точність ведення медичної документації та якість обслуговування пацієнтів. Діяльність сучасної стоматологічної клініки охоплює широкий спектр завдань: від запису на прийом та ведення електронних медичних карток до управління графіками лікарів, фінансового обліку та комунікації з пацієнтами. Усі ці процеси потребують ефективної координації та швидкої обробки даних, що робить тему проєктування власної ІС надзвичайно актуальною та такою, що має практичне значення.

Аналіз останніх досліджень і публікацій. Аналіз останніх досліджень і публікацій допоміг визначити ключові проблеми та перспективи галузі. Зокрема, стаття І. С. Петренко та О. А. Ковальчука «Впровадження електронних систем в закладах охорони здоров'я: виклики та перспективи в Україні» [12] дозволила оцінити поточний стан цифровізації медичної сфери в країні та виявити потребу в гнучких і кастомізованих ІТ-рішеннях, здатних враховувати унікальні бізнес-процеси конкретних клінік, що й стало однією з відправних точок для даної роботи. На ринку існує низка інформаційних систем, призначених для медичних закладів. Деякі стоматологічні клініки, наприклад, "Люмі-Дент", "ПРАЙМ", "Ddent", використовують власні або адаптовані рішення для онлайн-запису,

ведення електронних карток та комунікації з пацієнтами. Однак, часто такі системи є або частиною великих медичних інформаційних систем (МІС) загального призначення, або вузькоспеціалізованими рішеннями з обмеженим функціоналом. Незважаючи на наявність готових рішень, розробка кастомізованої інформаційної системи дозволяє врахувати унікальні потреби конкретного закладу, забезпечити необхідний рівень гнучкості та оптимізувати витрати на автоматизацію. Таким чином, проектування власної ІС, орієнтованої на специфічні завдання, стає ключовим фактором для підвищення ефективності та конкурентоспроможності клініки.

Об'єктом дослідження виступає процес розробки інформаційної системи для зубної клініки.

Предметом дослідження є методи, засоби та технології проектування інформаційної системи для зубної клініки, що забезпечують автоматизацію її ключових бізнес-процесів та підвищення ефективності роботи.

Мета кваліфікаційної бакалаврської роботи полягає в закріпленні та поглибленні знань шляхом розробки концепції та проектних рішень для інформаційної системи, що дозволить автоматизувати ключові процеси клініки, покращити якість обслуговування пацієнтів та оптимізувати управлінську діяльність.

Для досягнення поставленої мети необхідно вирішити такі завдання:

- провести моделювання предметної області стоматологічної клініки;
- визначити бізнес-вимоги, а також функціональні та нефункціональні вимоги до системи;
- розробити моделі поведінки, структури та користувацького інтерфейсу системи;
- створити інтерактивний прототип основних функціональних компонентів інформаційної системи;
- розробити тест-кейси для перевірки відповідності системи вимогам та провести трасування вимог.

Методи дослідження. У процесі виконання роботи були використані сучасні методології проектування інформаційних систем, зокрема об'єктно-процесне моделювання (OPM), уніфікована мова моделювання (UML), а також нотації BPMN, DFD, IDEF0 та IDEF3 для опису функціональних моделей та бізнес-процесів. Для створення прототипу користувацького інтерфейсу застосовано інструменти візуального дизайну, такі як Figma.

Теоретична, методична та практична значущість отриманих результатів. Одержані результати можуть бути використані для подальшої реалізації повнофункціонального веб-додатку ІС управління зубною клінікою. Впровадження такої системи сприятиме підвищенню ефективності роботи персоналу, зменшенню ймовірності помилок у веденні документації та покращенню якості обслуговування пацієнтів, що має значне практичне значення для медичних закладів.

Структура роботи. Кваліфікаційна бакалаврська робота складається зі вступу, трьох розділів, висновків, переліку використаних джерел та додатків. Загальний обсяг роботи становить 48 сторінок, вона містить 35 рисунків та список літератури з 6 найменувань.

РОЗДІЛ 1. ХАРАКТЕРИСТИКА ТА АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ

1.1 Характеристика предметної галузі та об'єкта дослідження

Успішне проектування та розробка інформаційної системи значною мірою залежить від глибокого розуміння та точного представлення предметної області, для якої ця система створюється. Предметна галузь даного проєкту охоплює сукупність бізнес-процесів, об'єктів та взаємозв'язків, що існують у сфері діяльності сучасної стоматологічної клініки. Ця сфера характеризується високою інтенсивністю інформаційних потоків та необхідністю ефективної координації дій між персоналом та пацієнтами.

Моделювання предметної області є ключовим етапом, що дозволяє формалізувати знання про діяльність закладу. Метою такого моделювання є створення чіткої та структурованої концептуальної моделі бізнес-процесів та даних клініки. Це включає ідентифікацію ключових сутностей, таких як пацієнти, лікарі, адміністратори, записи на прийом, медичні картки та послуги, а також визначення їхніх атрибутів та взаємозв'язків.

Діяльність стоматологічної клініки як об'єкта дослідження охоплює широкий спектр завдань, що потребують ефективної координації та швидкої обробки даних. Для наочного представлення системи як єдиного цілого в її оточенні, а також для візуалізації основного процесу та ключових об'єктів, була розроблена системна діаграма з використанням об'єктно-процесної методології (ОПМ), що зображена на рисунку 1.1.

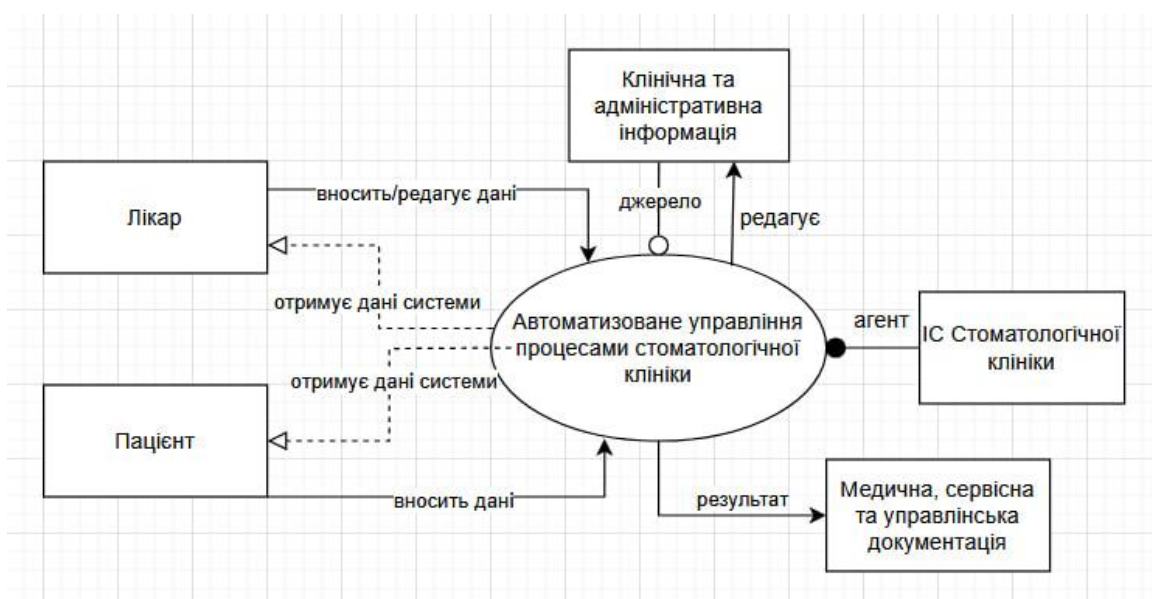


Рисунок 1.1 – Системна діаграма «Автоматизоване управління процесами стоматологічної клініки»

Джерело: розроблено самостійно.

На схемі представлено головний процес "Автоматизоване управління процесами стоматологічної клініки", який символізує комплексну роботу проєктованої інформаційної системи. З цим процесом взаємодіють ключові учасники (актори): "Пацієнт" та "Лікар", які ініціюють взаємодію або надають вхідні дані. Основним типом даних, що споживаються системою для виконання її функцій, є "Клінічна та адміністративна інформація". Об'єкт "ІС Стоматологічної клініки" визначений як агент, що відповідає за виконання головного процесу. В результаті роботи системи генерується "Медична, сервісна та управлінська документація", що є основним результатом її діяльності. Створення такої моделі слугує фундаментом для визначення меж системи, розробки структури бази даних та формування вимог до її функціоналу.

1.2 Аналіз літературних джерел та практичного досвіду використання ІС і технологій в предметній галузі

Для обґрунтування концепції нової інформаційної системи необхідно проаналізувати практичний досвід та ІТ-рішення, які вже успішно функціонують на українському ринку. Багато провідних стоматологічних клінік, розуміючи

важливість цифрової взаємодії з клієнтом, інвестують у власні або адаптовані інформаційні системи. Розглянемо функціональні можливості, що пропонують клініки "Люмі-Дент", "ПРАЙМ" та "Ddent", як приклади реалізації таких систем.

1. IT-рішення мережі клінік «Люмі-Дент»

«Люмі-Дент» є великою мережею стоматологічних центрів, що використовує комплексну інформаційну систему, орієнтовану на максимальну зручність для пацієнта. Їхній веб-портал є не просто сайтом, а повноцінним інструментом взаємодії.

Ключові функції: Система надає розширені можливості для онлайн-запису, включаючи вибір філії, лікаря, послуги та конкретного часового слоту. Пацієнтам доступний особистий кабінет, де зберігається історія візитів та фінансових операцій. Наявний детальний онлайн-прайс та реалізована інтеграція з месенджерами для нагадувань і підтверджень.

Переваги: Високий рівень автоматизації та клієнтоорієнтованості. Єдиний інформаційний простір для великої мережі.

Особливості: Очевидно, що це власна дорога розробка або глибоко кастомізоване рішення, недоступне для купівлі як готовий продукт.

2. Система клініки «ПРАЙМ»

Стоматологічна клініка «ПРАЙМ» також демонструє ефективне використання IT для залучення та обслуговування пацієнтів. Їх система фокусується на простоті та ефективності ключових процесів.

Ключові функції: Система забезпечує зручний та інтуїтивно зрозумілий процес онлайн-запису. Реалізовано ведення електронної бази пацієнтів та їхніх візитів. Активно використовуються канали комунікації для підтвердження записів та інформування про спеціальні пропозиції.

Переваги: Простота та надійність основного функціоналу, що знижує поріг входження для користувачів.

Особливості: Ймовірно, це адаптована версія однієї з комерційних МІС або менш комплексне власне рішення порівняно з «Люмі-Дент».

3. Цифрові інструменти клініки «Ddent»

Клініка «Ddent» є прикладом закладу, що впровадив базові, але ефективні цифрові інструменти для автоматизації взаємодії з пацієнтами.

Ключові функції: Основою є модуль онлайн-запису, що дозволяє пацієнтам самостійно обирати час для візиту. Система забезпечує ведення електронного розкладу, що оптимізує роботу адміністраторів.

Переваги: Фокус на найважливішій функції — записі на прийом — дозволяє швидко впровадити рішення та отримати миттєву користь.

Особливості: Функціонал може бути обмеженим, не включаючи особистий кабінет чи складну фінансову аналітику, що є типовим для вузькоспеціалізованих рішень.

Для наочного порівняння можливостей, що надаються ІТ-рішеннями цих клінік, наведемо зведену таблицю.

Таблиця 1
Порівняння інформаційних систем

№	Можливості	Ddent	ПРАЙМ	Люмі-Дент
1	Безпека даних клієнтів	Так	Так	Так
2	Інтуїтивна зручність інтерфейсу	Так	Так	Так
3	Доступ до медичних карток пацієнтів	Так	Так	Так
4	Доступність на всіх пристроях	Так	Так	Так
5	Безкоштовна версія з основними функціями	Ні	Ні	Ні
6	Можливість ведення медичної документації	Так	Так	Так
7	Інтерфейс для завантаження файлів	Так	Ні	Так
8	Можливість онлайн-запису на прийом	Так	Так	Так
9	Історія медичних відвідувань	Так	Так	Так
10	Можливість пошуку	Так	Так	Так

РОЗДІЛ 2. РОЗРОБКА ВИМОГ І МОДЕЛЮВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

Етап визначення вимог та подальшого моделювання є фундаментальним у життєвому циклі розробки будь-якої інформаційної системи. Саме вимоги закладають основу для всіх подальших проєктних рішень, процесу розробки, тестування та успішного впровадження. Якісно сформульовані вимоги дозволяють чітко визначити очікування замовника та кінцевих користувачів, мінімізувати ризики та уникнути значних переробок на пізніх стадіях проєкту. У цьому розділі детально розглядаються бізнес-вимоги, функціональні та нефункціональні вимоги, а також на основі них створюються моделі майбутньої системи.

2.1 Аналіз і специфікація вимог до інформаційної системи

Розроблення вимог до системи, що проєктується, передбачає визначення її цілей, функцій та обмежень, що на неї накладаються. Вимоги є умовами або можливостями, які повинна мати система для задоволення стандартів, специфікацій та інших формальних документів. Цей процес є частиною системного інжинірингу, від якого великою мірою залежить успішність всього проєкту. Вимоги до системи традиційно поділяються на три основні категорії: бізнес-вимоги, що визначають стратегічні цілі; функціональні вимоги, що описують конкретні можливості системи; та нефункціональні вимоги, що встановлюють критерії її якості.

Бізнес-вимоги формулюють цілі та завдання найвищого рівня, які інформаційна система повинна вирішити для досягнення стратегічних переваг та підвищення ефективності діяльності стоматологічної клініки. Вони відповідають на питання "Навіщо створюється система?" та визначають її цінність для бізнесу, обґрунтовуючи інвестиції в її розробку. Ключові бізнес-вимоги до проєктованої ІС, що відображають основні очікування від її впровадження, представлені на рисунку 2.1.

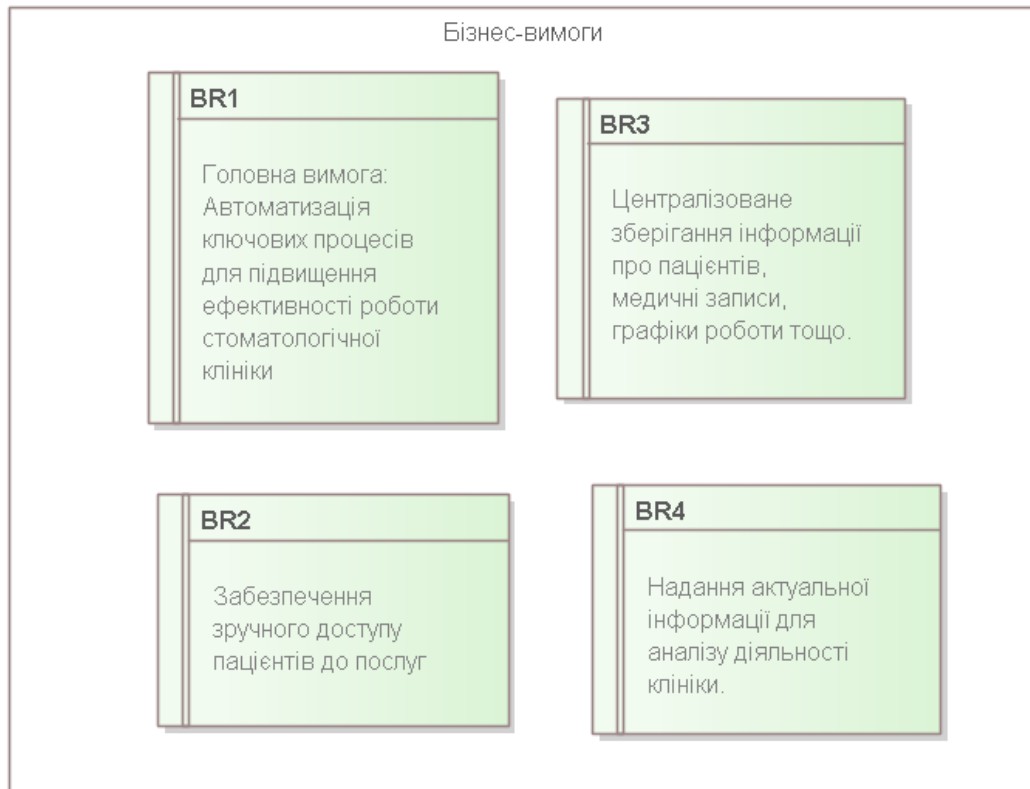


Рисунок 2.1 – Діаграма бізнес-вимог до системи

Джерело: розроблено самостійно.

На представленій діаграмі візуалізовано ієрархічну структуру бізнес-вимог, які є рушійною силою проєкту. Центральною бізнес-вимогою (BR1) є "Автоматизація ключових процесів для підвищення ефективності роботи стоматологічної клініки". Ця вимога є основоположною, оскільки головна мета впровадження ІС полягає в оптимізації щоденних операцій та раціональному використанні ресурсів клініки. Дана вимога декомпонується на кілька специфічних, що конкретизують шляхи її досягнення. Вимога BR2 "Забезпечення зручного доступу пацієнтів до послуг" підкреслює клієнтоорієнтованість майбутньої системи. Її реалізація дозволить пацієнтам легше та швидше взаємодіяти з клінікою, наприклад, при самостійному записі на прийом чи отриманні доступу до своєї медичної інформації, що, у свою чергу, підвищить їхню задоволеність та лояльність. Вимога BR3 "Централізоване зберігання інформації про пацієнтів, медичні записи, графіки роботи тощо"

зумовлена великими обсягами даних, з якими щоденно працює медичний заклад. Централізація даних у єдиному сховищі забезпечить їхню цілісність, консистентність, швидкий доступ для авторизованого персоналу та суттєво зменшить ризики втрати чи спотворення важливої інформації. Нарешті, вимога BR4 "Надання актуальної інформації для аналізу діяльності клініки" спрямована на підтримку прийняття обґрунтованих управлінських рішень. Система повинна не просто зберігати дані, а й надавати інструменти для їх аналізу, генеруючи звіти, що дозволять керівництву оцінювати ефективність роботи, фінансові показники та планувати подальший розвиток бізнесу.

Функціональні вимоги визначають конкретні дії, операції та завдання, які інформаційна система повинна виконувати для задоволення потреб користувачів. Вони детально описують поведінку системи та її взаємодію з користувачами. Модель основних функціональних вимог до проєктованої інформаційної системи, що демонструє їх ієрархічну структуру, наведена на рисунку 2.2.

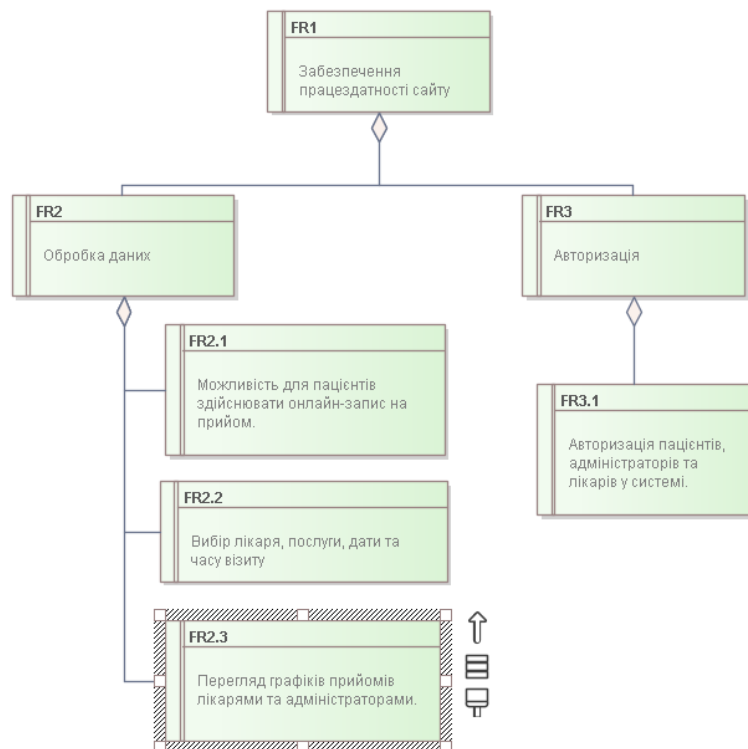


Рисунок 2.2 – Ієрархія функціональних вимог до системи

Джерело: розроблено самостійно.

На діаграмі визначено ключові функціональні можливості системи, що є основою для її подальшого проєктування. Базовою вимогою є FR1: Забезпечення працездатності сайту. Ця вимога є загальною, але критично важливою, оскільки вона охоплює всі аспекти, необхідні для стабільного та коректного функціонування веб-інтерфейсу, гарантуючи, що всі компоненти сайту будуть доступні користувачам, а навігація — логічною та безперебійною. Комплексна вимога FR2: Обробка даних стосується основних операцій з інформацією, які система повинна підтримувати, і деталізується на кілька підпорядкованих вимог. FR2.1: Можливість для пацієнтів здійснювати онлайн-запис на прийом є однією з ключових функцій, що надає пацієнтам інтуїтивно зрозумілий інтерфейс для самостійного запису до лікаря через інтернет. Ця можливість підкріплюється вимогою FR2.2: Вибір лікаря, послуги, дати та часу візиту, яка гарантує, що пацієнт матиме змогу обрати всі необхідні параметри візиту. Для внутрішніх користувачів системи передбачена вимога FR2.3: Перегляд графіків прийомів лікарями та адміністраторами, що дозволяє лікарям бачити свій розклад, а адміністраторам — координувати роботу всієї клініки. Важливою є вимога FR3: Авторизація, яка визначає необхідність контролю доступу до системи. Вона реалізується через FR3.1: Авторизація пацієнтів, адміністраторів та лікарів у системі, що передбачає наявність унікальних облікових даних для кожного типу користувача та надання доступу до функцій згідно з визначеною роллю.

Для детальної специфікації та управління цими вимогами протягом життєвого циклу проєкту використовуються спеціалізовані CASE-засоби, наприклад, Specification Manager в середовищі Enterprise Architect, приклад якого наведено на рисунку 2.3.

✓ FR1 Забезпечення працездатності сайту	Functional	Proposed	Medium	Medium
✓ FR2 Обробка даних	Functional	Approved	Medium	Medium
✓ FR2.1 Можливість для пацієнтів здійснювати онлайн-запис на прийом.	Functional	Approved	Medium	Medium
✓ FR2.2 Вибір лікаря, послуги, дати та часу візиту	Functional	Approved	Medium	Medium
✓ FR2.3 Перегляд графіків прийомів лікарями та адміністраторами.	Functional	Approved	Medium	Medium

Рисунок 2.3.1 – Приклад специфікації вимог у Specification Manager

✓ FR3 Авторизація	Functional	Validated	Medium	Medium
✓ FR3.1 Авторизація пацієнтів, адміністраторів та лікарів у системі.	Functional	Validated	Medium	Medium
✓ Requirement A.2.2 [Details of the requirement]	Functional	Validated	Medium	Medium
✓ Requirement A.2.3 [Details of the requirement]	Functional	Validated	Medium	Medium

Рисунок 2.3.2 – Приклад специфікації вимог у Specification Manager

Джерело: розроблено самостійно.

Нефункціональні вимоги визначають критерії якості, атрибути та характеристики системи, які не пов'язані безпосередньо з її функціями, але є критично важливими для її ефективної та зручної експлуатації. Вони описують, наскільки добре система виконує свої завдання, і значною мірою впливають на загальне враження користувачів. Ключові нефункціональні вимоги, згруповані за категоріями якості, представлено на рисунку 2.4.

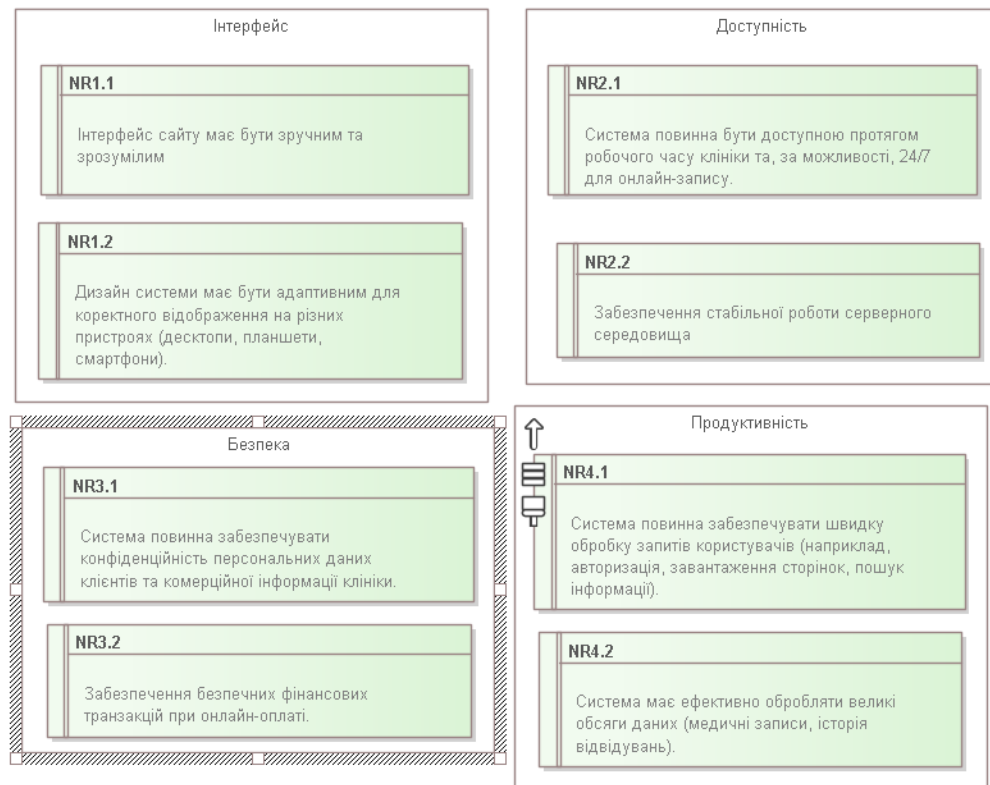


Рисунок 2.4 – Діаграма нефункціональних вимог до системи

Джерело: розроблено самостійно.

На діаграмі вимоги згруповані за чотирма категоріями для кращого розуміння аспектів якості системи. Категорія "Інтерфейс" визначає вимоги до взаємодії користувача з системою. Вона включає вимогу до зручності та інтуїтивної зрозумілості (NR1.1), що мінімізує час на навчання та ймовірність помилок, а також вимогу до адаптивності дизайну (NR1.2) для комфортної роботи на будь-якому пристрої, будь то десктоп, планшет чи смартфон. Категорія "Доступність" стосується забезпечення стабільного доступу. Вимога NR2.1 гарантує, що персонал зможе використовувати систему протягом робочого дня, а пацієнти матимуть можливість записатися на прийом у будь-який час (24/7). Це підкріплюється вимогою NR2.2 щодо стабільної роботи серверного середовища. Вимоги категорії "Безпека" є пріоритетними, особливо в медичній сфері. NR3.1 вимагає забезпечення повної конфіденційності чутливої інформації, як-от медичні записи та особисті дані пацієнтів. Якщо система передбачає онлайн-платежі, вимога NR3.2 зобов'язує використовувати захищені

протоколи для обробки фінансових транзакцій. Категорія "Продуктивність" встановлює вимоги до швидкодії. NR4.1 вимагає, щоб час відгуку системи на дії користувача був мінімальним, аби не викликати роздратування. Вимога NR4.2 гарантує, що система зберігатиме свою продуктивність навіть зі зростанням кількості пацієнтів та обсягу медичної інформації з часом.

2.2 Постановка та алгоритм розв'язання задачі

Після визначення та специфікації вимог до системи, наступним логічним кроком є перехід від загального опису "що" система повинна робити, до більш формалізованого опису "як" вона буде це робити на концептуальному рівні. Цей етап є критично важливим, оскільки він закладає фундамент для подальшого технічного проєктування та розробки. Відповідно до стандартів інженерії програмного забезпечення, цей процес включає формальну постановку задачі та опис алгоритмів її розв'язання. Постановка задачі дозволяє чітко окреслити межі функціональності, визначити всі вхідні потоки даних, які система буде обробляти, та описати структуру й призначення вихідної інформації, що є результатом її роботи. У свою чергу, розробка алгоритмів деталізує логічну послідовність дій, які система має виконати для перетворення вхідних даних на очікувані результати. Цей формалізм допомагає уникнути неоднозначності та слугує основою для проєктування архітектури та бази даних. Розроблена інформаційна модель задачі представлена на Рис. 2.5.

Даталогічна

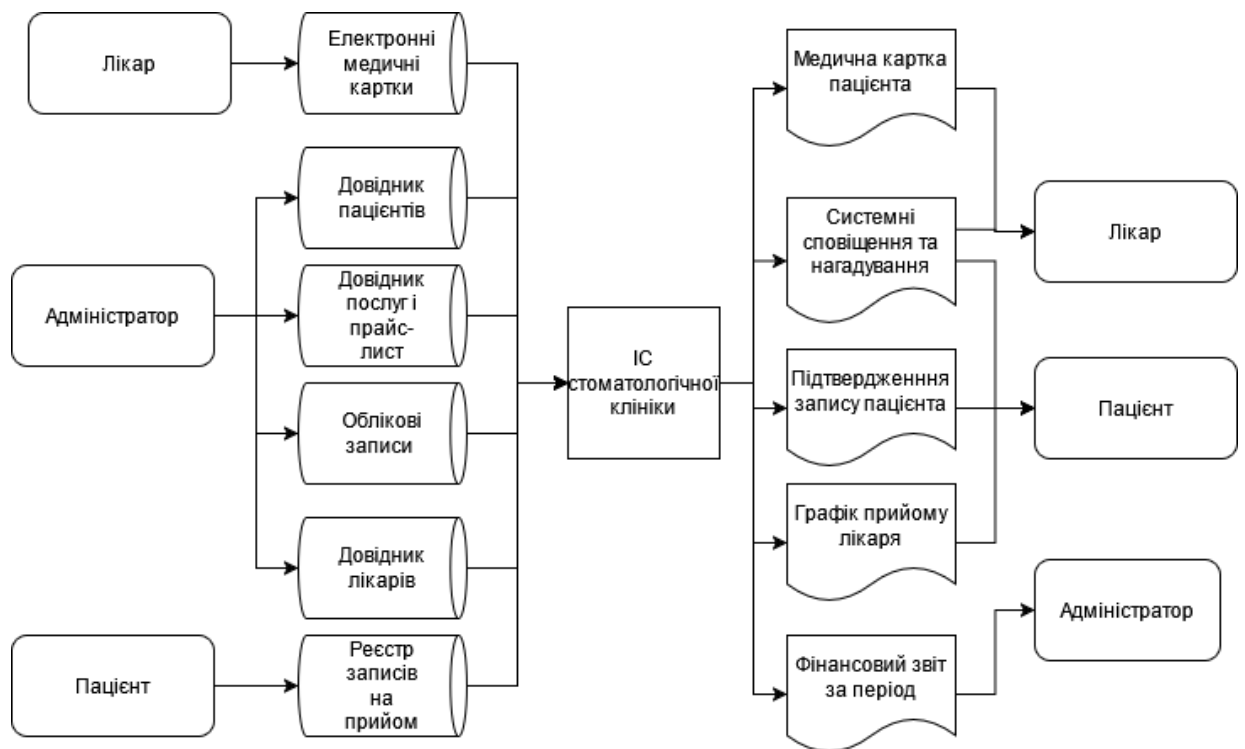


Рисунок 2.5 – Схема інформаційної моделі задачі

2.2.1 Характеристика вхідної та вихідної інформації

Призначенням задачі, що розв'язується, є комплексна автоматизація управлінських та сервісних процесів стоматологічної клініки. Техніко-економічна сутність полягає в оптимізації використання робочого часу персоналу, підвищенні точності та оперативності ведення документації, а також покращенні клієнтського досвіду за рахунок надання сучасних цифрових сервісів. Об'єктами управління, що підлягають автоматизації, є записи пацієнтів на прийом, графіки роботи лікарів, електронні медичні картки, облік наданих послуг та фінансові розрахунки. Система функціонує у постійному режимі (реального часу) для транзакційних операцій, таких як запис на прийом чи оплата, та за запитом для генерації аналітичної звітності.

Вхідна інформація системи формується з даних, що надаються її користувачами: пацієнтами, лікарями та адміністраторами. До цієї інформації належать декілька ключових потоків даних. По-перше, це облікові дані (логін та пароль), які користувач вводить у формі авторизації для отримання доступу до системи. По-друге, це дані для запису на прийом, які пацієнт або адміністратор

вводить у відповідній формі: ідентифікатор пацієнта, обрана послуга, спеціаліст, бажана дата та конкретний часовий слот. По-третє, це персональні та клінічні дані, що вносяться лікарем в електронну медичну картку пацієнта під час або після консультації. Джерелами інформації є безпосередньо користувачі, що взаємодіють із системою через веб-інтерфейс.

Перелік вхідних повідомлень вказаний в таблиці 2.1.

Таблиця 2.1
Вхідні повідомлення системи

№ з/п	Назва вхідного повідомлення	Ідентифікатор	Форма подання	Термін і частота надходження	Джерело
1	Дані про нового клієнта	IN-01	Веб-форма	У момент заповнення	Клієнт
2	Запис на прийом	IN-2	Веб-форма	У момент подання запиту	Клієнт
3	Дані про оплату послуг	IN-3	Веб-форма, API	У момент здійснення оплати	Клієнт, платіжна система
4	Результати обслуговування	IN-4	PDF	Після завершення прийому	Лікар
5	Дані для підтвердження запису	IN-5	Email, SMS	У момент створення запису	Адміністратор
6	Відомості про здійснені оплати	IN-6	PDF	У момент оплати	Система

Вихідна інформація є результатом обробки вхідних даних системою і призначена для різних категорій користувачів. Для пацієнтів ключовим результатом є отримання підтвердження про запис із деталями візиту (дата, час,

лікар, послуга), а також доступ до своєї історії візитів та медичних записів в особистому кабінеті. Для лікарів та адміністраторів система генерує актуальні графіки роботи та прийомів, що відображають поточне завантаження. Для керівництва клініки вихідною інформацією є аналітичні звіти, що дозволяють оцінювати фінансові показники та ефективність роботи персоналу. Усі ці дані представляються у вигляді відеограм (інформація на екранах) та, за потреби, можуть бути експортовані у формати документів.

Перелік вихідних повідомлень наведено в таблиці 2.2.

Таблиця 2.2
Вихідні повідомлення системи

№ з/п	Назва повідомлення	Ідентифікатор	Формат подання і вимоги	Частота створення	Час виконання	Споживачі інформації
1	Нагадування про прийом	D-01	Email, Sms	Один раз за день	Відразу	Клієнти
2	Підтвердження запису	D-02	Email, Sms	Після обробки запису	До 10 хвилин	Клієнти
3	Квитанція про оплату	D-03	PDF	Після завершення оплати	До 10 хвилин	Клієнти
4	Щоденний розклад прийомів	D-04	Excel	Щоденно	Відразу	Лікарі
5	Звіт про виконанні прийоми	D-05	PDF	Після прийому	До 10 хвилин	Адміністратор
6	Оновлення бази даних	D-06	SQL	Після кожної дії в системі	Відразу	Система

2.2.2 Алгоритм розв’язання задачі

Логіка функціонування системи базується на чітко визначених алгоритмах, які описують послідовність кроків для виконання ключових операцій. В основі цих алгоритмів лежать бізнес-правила, реалізовані через програмну логіку та взаємодію з базою даних. Для наочного представлення логіки та послідовності етапів розрахунку використовуються діаграми декомпозиції ОРМ, що деталізують ключові процеси.

Використовувана інформація

Для забезпечення функціонування системи та виконання її ключових завдань використовуються структуровані інформаційні масиви, що зберігаються в базі даних. Ці масиви містять усю необхідну інформацію про пацієнтів, персонал, послуги та операційну діяльність клініки . Перелік основних масивів, що використовуються, наведено у таблицях 2.3 та 2.4.

Таблиця 2.3
Використовувана інформація системи

Назва масиву	Ідентифікатор	Максимальна кількість записів
Довідник пацієнтів	PATIENTS_DB	50000
Довідник лікарів	DOCTORS_DB	100
Облікові записи	USERS_AUTH	50600
Реєстр записів на прийом	APPOINTMENTS_REG	200000
Електронні медичні картки	MEDICAL_RECORDS	100000
Довідник послуг та прайс-лист	SERVICES_PRICE_LIST	1000

Таблиця 2.4
Результатна інформація системи

Назва масиву	Ідентифікатор	Максимальна кількість записів
Підтвердження запису пацієнту	APPOINTMENT_CONFIRM	200000
Графік прийому лікаря	DOCTOR_SCHEDULE_VIEW	5000
Електронна медична картка пацієнта	PATIENT_RECORD_VIEW	100000
Фінансовий звіт за період	FINANCIAL_REPORT	1000
Системні сповіщення та нагадування	NOTIFICATIONS_LOG	1000000
Медична картка пацієнта	PATIENT_CARD_VIEW	N

Математичний опис задачі:

Для аналізу діяльності клініки в системі передбачено можливість розрахунку ключових показників.

Розрахунок вартості візиту пацієнта (C_{visit}): Формула дозволяє визначити загальну вартість усіх послуг, наданих пацієнту в рамках одного візиту.

$$C_{visit} = \sum_{i=1}^N (P_i \times Q_i)$$

де:

- P_i – вартість i -тої послуги з прайс-листа.
- Q_i – кількість наданих i -х послуг під час одного візиту.
- N – загальна кількість унікальних послуг, наданих пацієнту за один візит.

Розрахунок завантаженості лікаря (L_{doctor}): Формула розраховує відсоток завантаженості робочого часу лікаря, що є важливим показником ефективності планування графіків.

$$L_{doctor} = \frac{T_{booked}}{T_{total}} \times 100\%$$

де:

- T_{booked} – кількість заброньованих годин у графіку лікаря за період.
- T_{total} – загальна кількість робочих годин лікаря за той самий період.

Алгоритм розв'язання задачі на ЕОМ

Функціонування інформаційної системи відбувається за алгоритмом, що визначає логічну послідовність операцій залежно від ролі користувача. Загальна логіка роботи системи візуалізована на блок-схемі (Рис. 2.6) та включає такі кроки:

Авторизація: Будь-яка взаємодія з персоналізованими даними починається з авторизації. Користувач вводить облікові дані, які система перевіряє. У разі успіху надається доступ, у разі невдачі – виводиться повідомлення про помилку.

Розгалуження за ролями: Після успішного входу система визначає роль користувача («Пацієнт», «Лікар» чи «Адміністратор») і надає доступ до відповідного функціонала.

Пацієнт: Може переглядати графіки лікарів, записуватися на прийом, обираючи послугу, дату та час, переглядати історію своїх візитів та здійснювати оплату.

Лікар: Має доступ до свого робочого графіка, може переглядати записи пацієнтів та вести їхні електронні медичні картки.

Адміністратор: Володіє найширшими правами. Окрім функцій лікаря та пацієнта (наприклад, запис пацієнта по телефону), він керує профілями користувачів, графіками роботи лікарів та прайс-листом послуг.

Виконання операцій: Користувач обирає дію в меню. Система обробляє запит, взаємодіє з базою даних (створення, читання, оновлення, видалення

даних) і повертає результат у вигляді оновленого інтерфейсу або системного повідомлення.

Завершення роботи: Користувач може вийти із системи, що завершує поточну сесію.

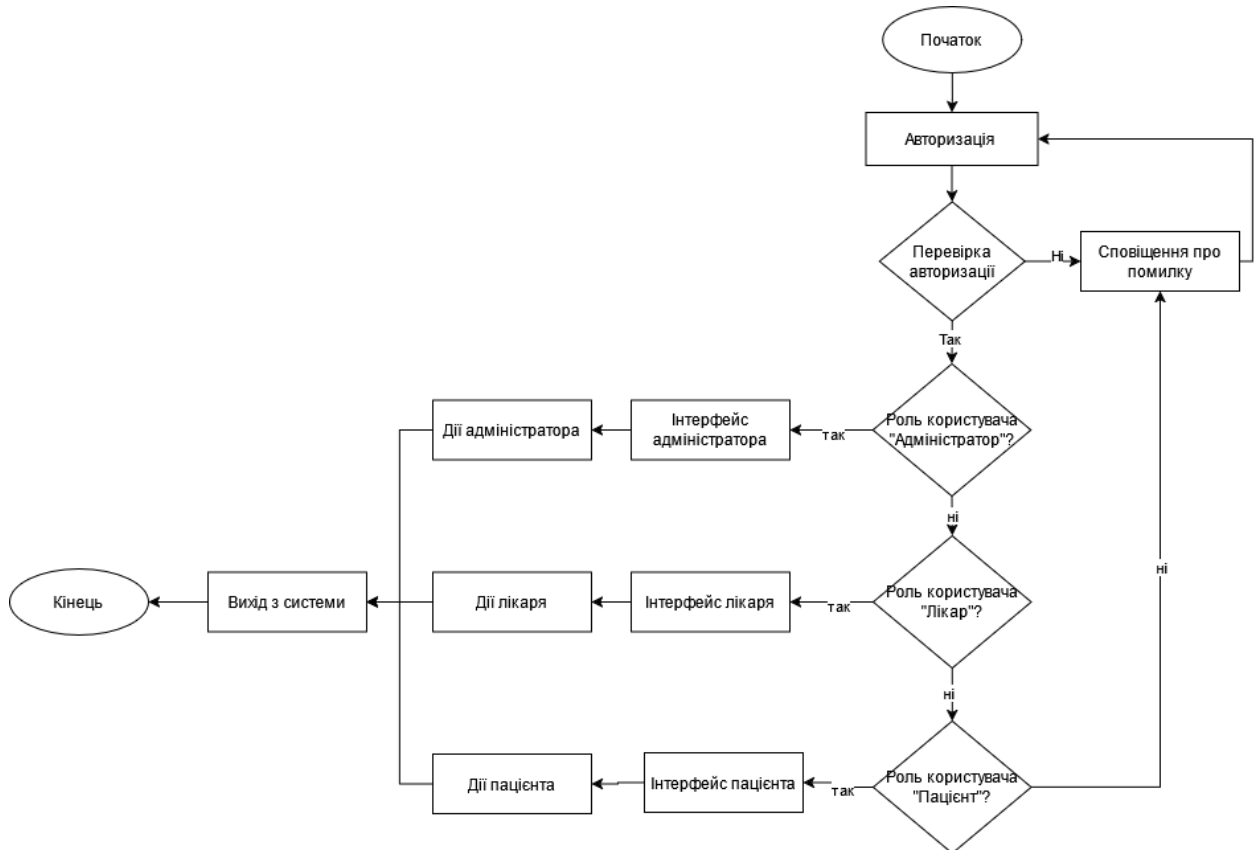


Рисунок 0.6 – Алгоритм розв’язання задачі функціонування системи

2.3 Моделювання інформаційної системи

Моделювання є універсальним та потужним засобом для вивчення і проектування складних систем, оскільки воно дозволяє спростити реальність шляхом абстрагування від несуттєвих деталей і зосередитись на ключових аспектах. Моделе-орієнтований інжиніринг, як сучасний підхід, передбачає, що основними артефактами розробки є саме моделі, з яких згодом можуть генеруватися інші системні елементи, включаючи код та документацію. Цей підхід дозволяє на ранніх етапах візуалізувати майбутню систему, проаналізувати різні варіанти архітектурних рішень, виявити потенційні проблеми та узгодити бачення продукту між усіма зацікавленими сторонами. Для забезпечення точності та однозначності інтерпретації проєктних рішень у

даній роботі використовуються загальновизнані стандарти моделювання, такі як Уніфікована мова моделювання (UML) та Мова системного моделювання (SysML).

2.3.1 Моделювання поведінки системи

Моделювання поведінки фокусується на динамічних аспектах системи, описуючи, як вона взаємодіє з користувачами та як її внутрішні компоненти реагують на зовнішні події для виконання поставлених завдань. Основними інструментами для цього є діаграми прецедентів та послідовності UML.

Діаграма прецедентів є високорівневим поданням функціональних вимог системи з точки зору користувача. Вона наочно демонструє межі системи та основні сценарії її використання. Діаграма прецедентів для ІС стоматологічної клініки наведена на рисунку 2.7.

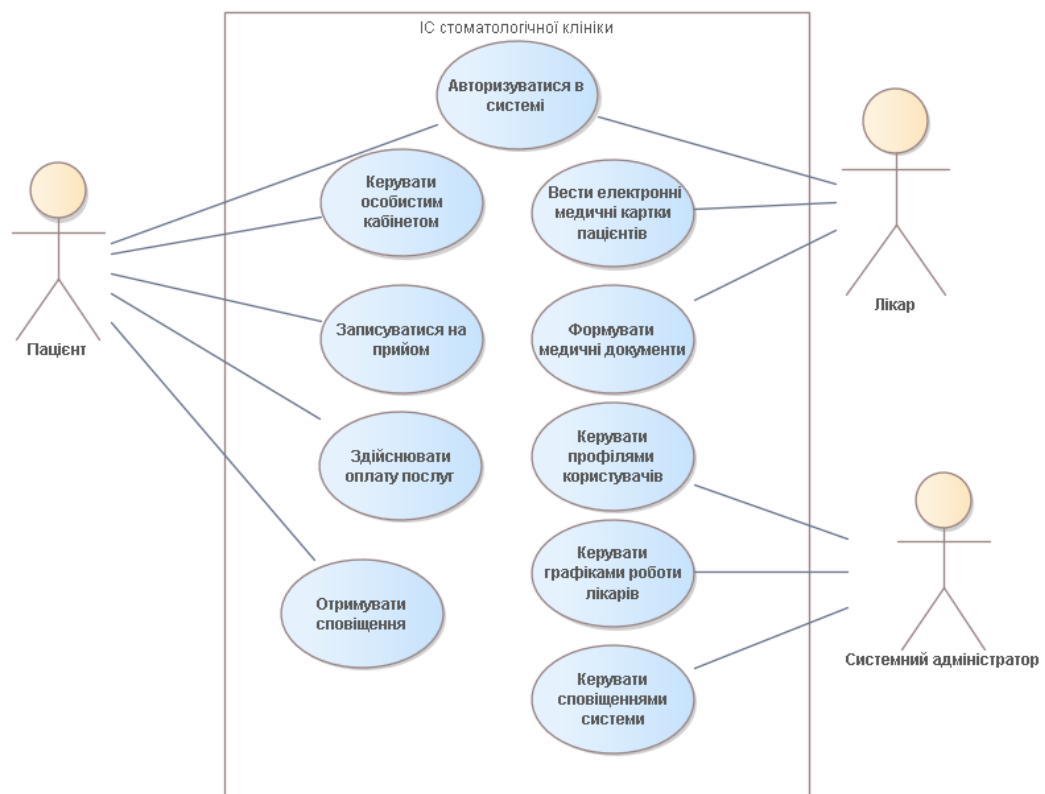


Рисунок 2.7 – Діаграма прецедентів

Джерело: розроблено самостійно.

На діаграмі визначено три ключові актори: "Пацієнт", "Лікар" та "Системний адміністратор". Прецедент "Авторизуватися в системі" є базовим і обов'язковим для доступу до персоналізованого функціоналу для всіх акторів. Для "Пацієнта" ключовими прецедентами є "Записуватися на прийом", "Керувати особистим кабінетом", "Здійснювати оплату послуг" та "Отримувати сповіщення". "Лікар" взаємодіє з системою через прецеденти "Вести електронні медичні картки пацієнтів" та "Формувати медичні документи". "Системний адміністратор" виконує управлінські функції через прецеденти "Керувати профілями користувачів" та "Керувати графіками роботи лікарів".

Діаграми послідовності деталізують внутрішню логіку виконання прецедентів, візуалізуючи взаємодію об'єктів системи в часі. Вони демонструють, як об'єкти обмінюються повідомленнями для досягнення певної мети. На рисунку 2.8 зображено діаграму послідовності для сценарію «Вхід в систему».

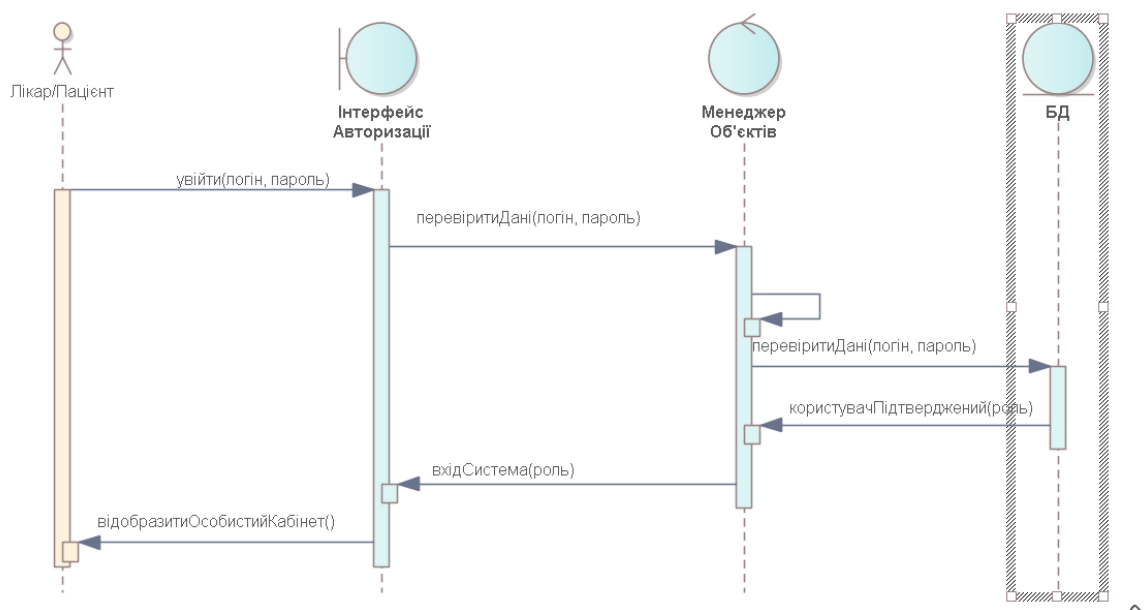


Рисунок 2.8 – Діаграма послідовності «Вхід в систему»

Джерело: розроблено самостійно.

Діаграма ілюструє процес авторизації, де користувач надсилає запит до об'єкта Інтерфейс Авторизації, який передає дані для перевірки об'єкту Менеджер Об'єктів. Той, у свою чергу, звертається до БД і після успішної

перевірки повертає результат, що призводить до відображення особистого кабінету. На рисунку 2.9 показано сценарій «Створення запису до лікаря».

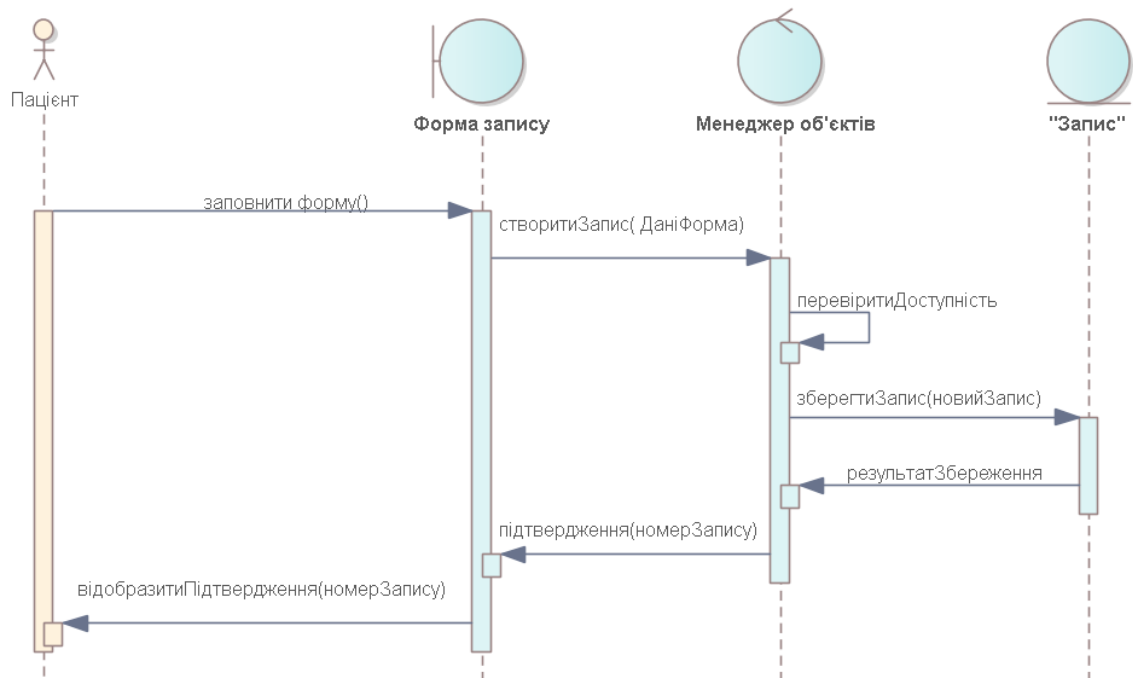


Рисунок 2.9 – Діаграма послідовності «Створення запису до лікаря»

Джерело: розроблено самостійно.

Ця діаграма описує, як пацієнт взаємодіє з Формою запису, дані з якої надходять до Менеджера об'єктів. Менеджер координує перевірку доступності та збереження нового запису в об'єкті "Запис", після чого повертає підтвердження користувачеві.

2.3.2 Моделювання структури системи

Моделювання структури визначає статичні аспекти системи: її основні "будівельні блоки", їхні властивості та взаємозв'язки. Це створює чітке уявлення про архітектуру майбутньої ІС. Для цього використовуються структурні діаграми, зокрема діаграми визначення блоків (BDD) мови SysML. На рисунку 2.10 зображено діаграму визначення блоків для системи.

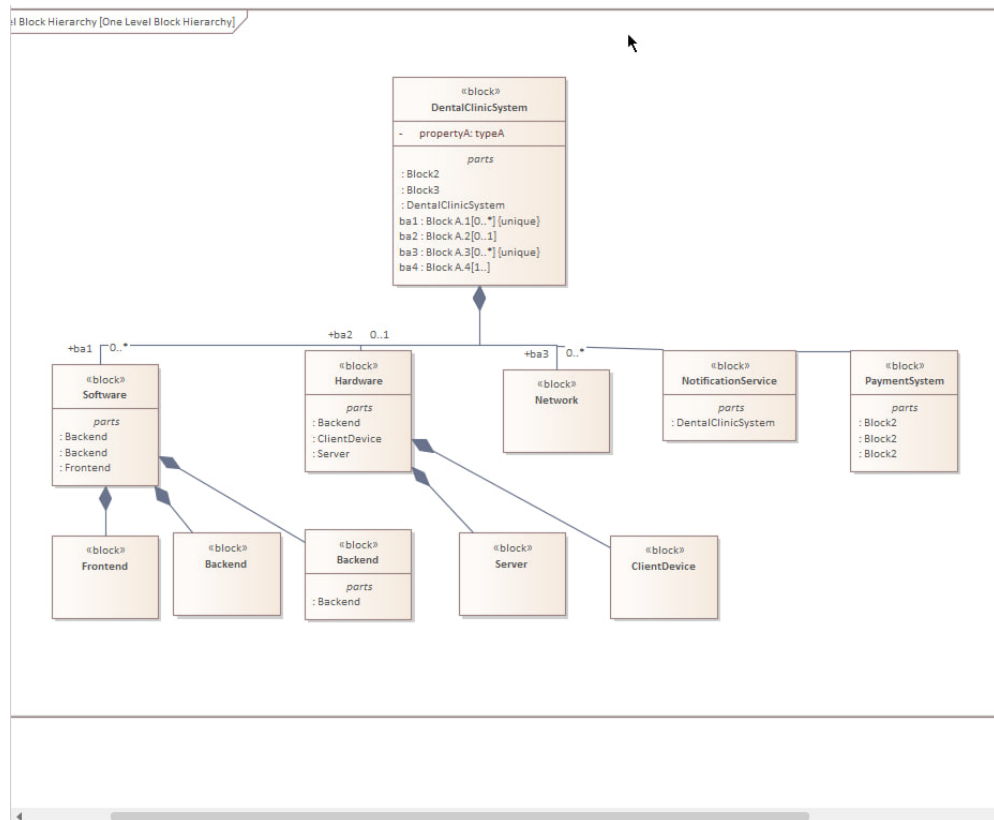


Рисунок 2.10 – Діаграма визначення блоків

Джерело: розроблено самостійно.

Центральним елементом моделі є блок `DentalClinicSystem`, що представляє всю систему як єдине ціле. Він декомпозиується на кілька ключових підсистем. Блок `<block> Software` інкапсулює все програмне забезпечення і, в свою чергу, складається з частин `Frontend` та `Backend`. Блок `<block> Hardware` описує фізичну інфраструктуру, необхідну для роботи системи, включаючи клієнтські пристрої (`ClientDevice`) та сервери (`Server`). Блок `<block> Network` відповідає за мережеву інфраструктуру. Окремо виділені сервісні блоки, такі як `<block> NotificationService` для надсилання сповіщень та `<block> PaymentSystem` для обробки платежів.

2.3.3 Розподіл вимог за компонентами системи

Верифікація проєктних рішень та забезпечення повноти покриття вимог є ключовим завданням, яке вирішується шляхом трасування. Трасування дозволяє відстежити зв'язок від початкової бізнес-потреби до конкретного елемента проєкту та відповідного тестового сценарію, що гарантує, що жодна вимога не була втрачена або неправильно реалізована. Для візуалізації цих зв'язків розробляється діаграма трасування, наведена на рисунку 2.11.

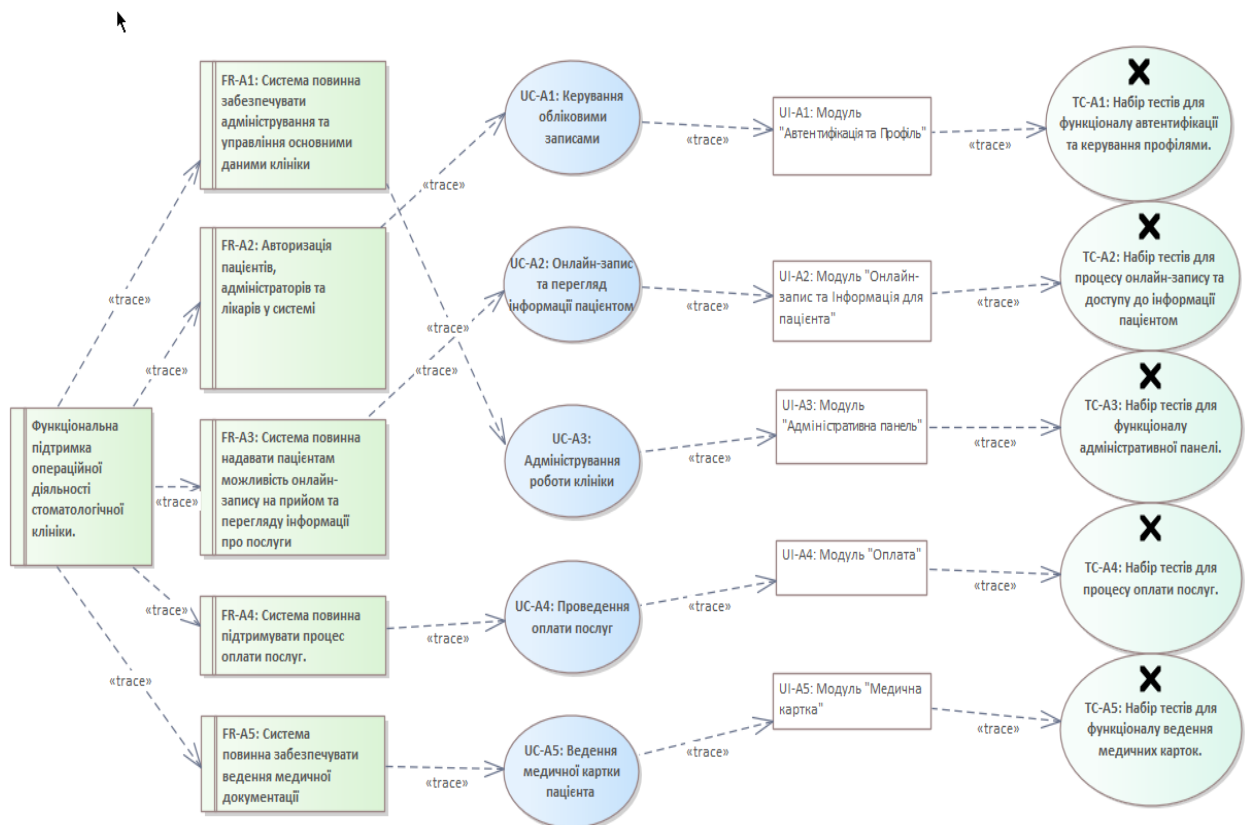


Рисунок 2.11 – Діаграма трасування

Джерело: розроблено самостійно.

Діаграма трасування демонструє логічний ланцюжок реалізації вимог. Процес починається з високорівневої функціональної вимоги, наприклад, FR-A2: "Авторизація пацієнтів, адміністраторів та лікарів у системі". Ця вимога знаходить своє втілення у конкретному варіанті використання, такому як UC-A2: "Онлайн-запис та перегляд інформації пацієнтом", який може вимагати попередньої автентифікації. Реалізація цього прецеденту покладається на конкретний програмний компонент або модуль, позначений на діаграмі як UI-

A1: "Модуль 'Аутентифікація та Профіль'". Нарешті, для перевірки коректності роботи цього модуля та відповідності початковій вимозі розробляється набір тест-кейсів, наприклад, ТС-A1: "Набір тестів для функціоналу аутентифікації та керування профілями". Цей наскрізний зв'язок "вимога → прецедент → компонент → тест" забезпечує прозорість процесу розробки та є основою для ефективного управління змінами та контролю якості.

2.3.4 Моделювання та прототипування користувацького інтерфейсу

Моделювання користувацького інтерфейсу (UI) є ключовим етапом у проєктуванні інформаційної системи, що слідує за визначенням її поведінки та структури. Цей етап спрямований на створення ефективної, зручної та інтуїтивно зрозумілої взаємодії між користувачем та системою. Якісно спроектований інтерфейс значно підвищує продуктивність роботи користувачів, знижує кількість помилок та сприяє позитивному загальному враженню від використання системи (User Experience - UX). Цей процес охоплює розробку логічної структури інтерфейсу, визначення навігаційних шляхів та створення візуальних прототипів різного ступеня деталізації.

На початковому етапі розробляється загальна структура навігації, що ілюструє основні екрани та логіку переходів між ними. Ця структура представлена на рисунку 2.12.

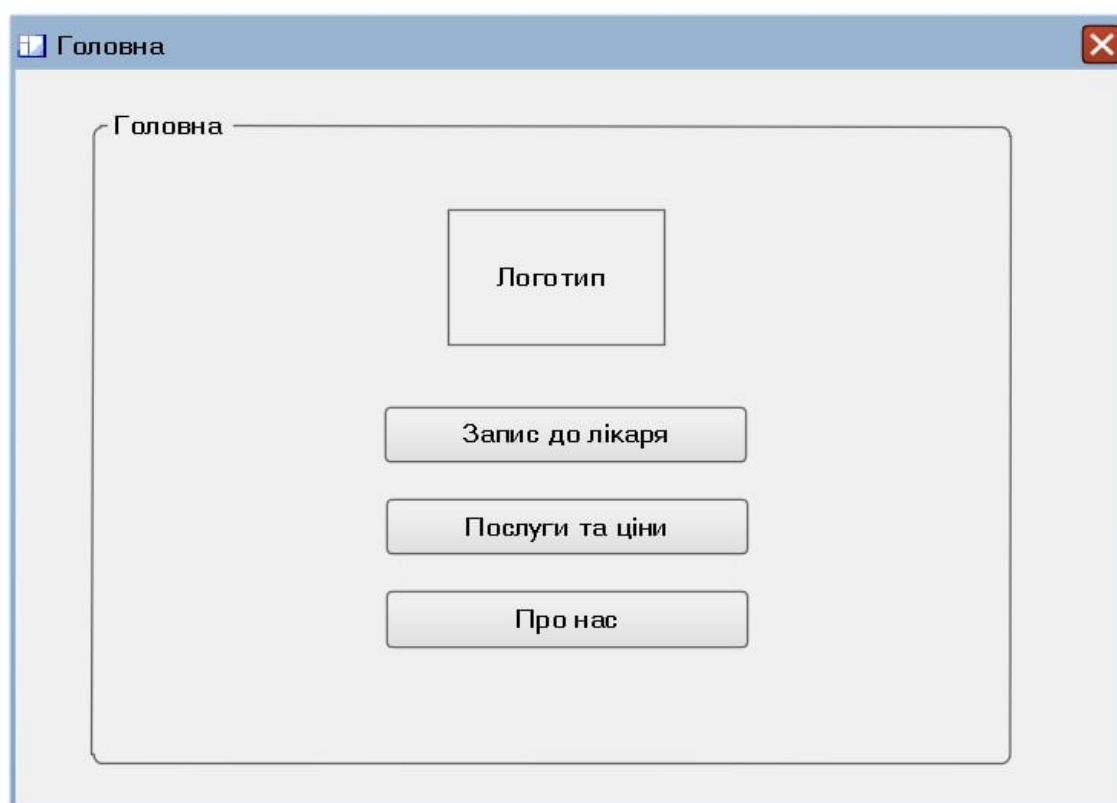


Рисунок 2.12 – Діаграма навігації користувацьким інтерфейсом
Джерело: розроблено самостійно.

Наведена діаграма відображає ієрархічну структуру навігації, де початковою точкою є "Головна сторінка". З неї користувач може перейти до "Авторизації/Реєстрації", а після успішної автентифікації — до "Особистого кабінету" з функціями запису на прийом, перегляду записів та оплати послуг. Окремою гілкою є "Адмін. панель", доступна для адміністративного персоналу, що забезпечує логічне розмежування функціоналу для різних ролей. Для візуалізації ключових екранів були розроблені спрощені макети (wireframes), що показують розташування основних елементів. На рисунку 2.13 зображено інтерфейс вибору послуги та дати.

Запис/Послуги

Логотип

Повернутися

Послуги

Оберіть послугу:

Консультація стоматолога

Оберіть дату:

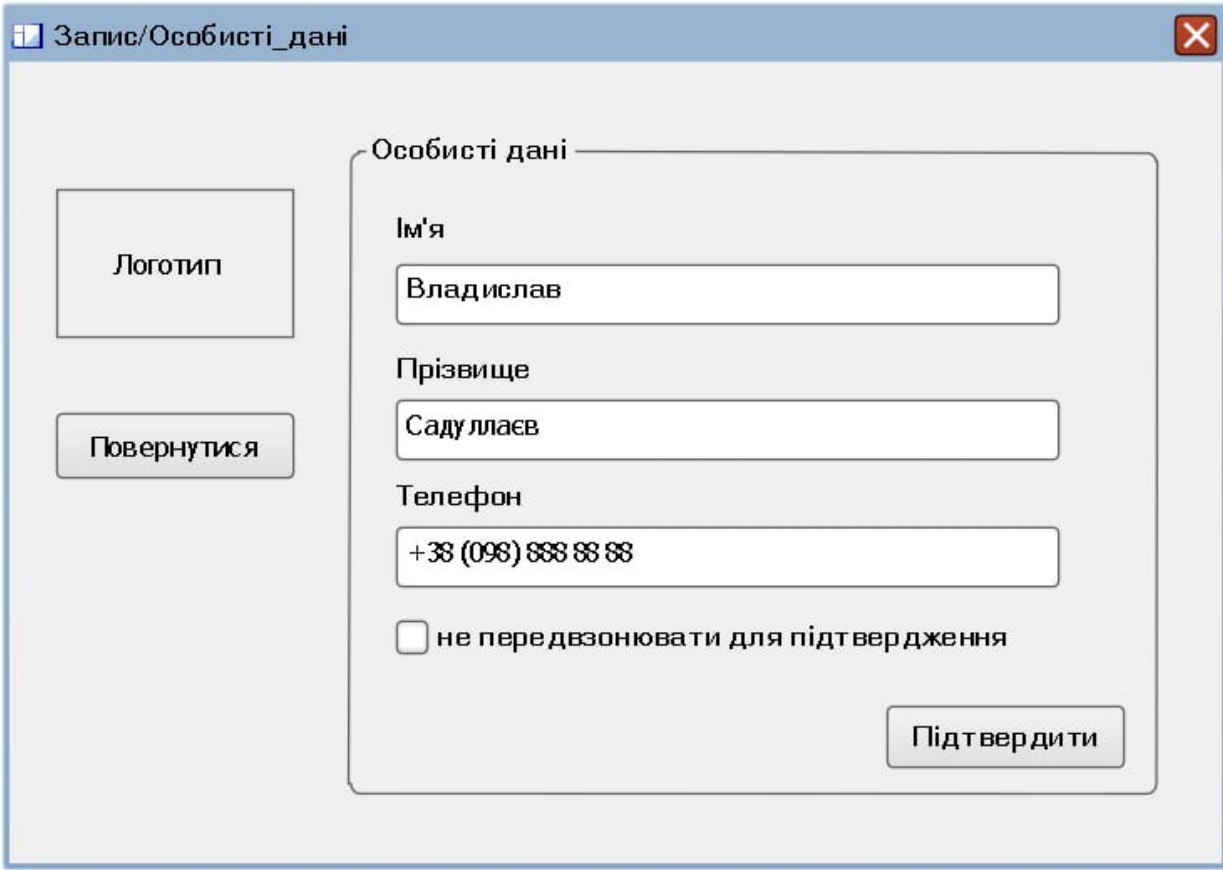
15.05.2025

Підтвердити

Рисунок 2.13 – Діаграма інтерфейсу вибору послуги і дати

Джерело: розроблено самостійно.

Для реєстрації нових пацієнтів або оновлення їхніх даних передбачено відповідну екранну форму (рисунок 2.14), яка містить стандартний набір полів для збору основної ідентифікаційної та контактної інформації про клієнта.



Запис/Особисті_дані

Логотип

Повернутися

Особисті дані

Ім'я
Владислав

Прізвище
Садуллаєв

Телефон
+38 (096) 888 88 88

☐ не передзвонювати для підтвердження

Підтвердити

Рисунок 2.14 – Діаграма інтерфейсу заповнення даних клієнта

Джерело: розроблено самостійно.

Важливим елементом взаємодії є надання користувачеві зворотного зв'язку про результати його дій. Приклад візуального сповіщення про успішне створення запису показано на рисунку 2.15.

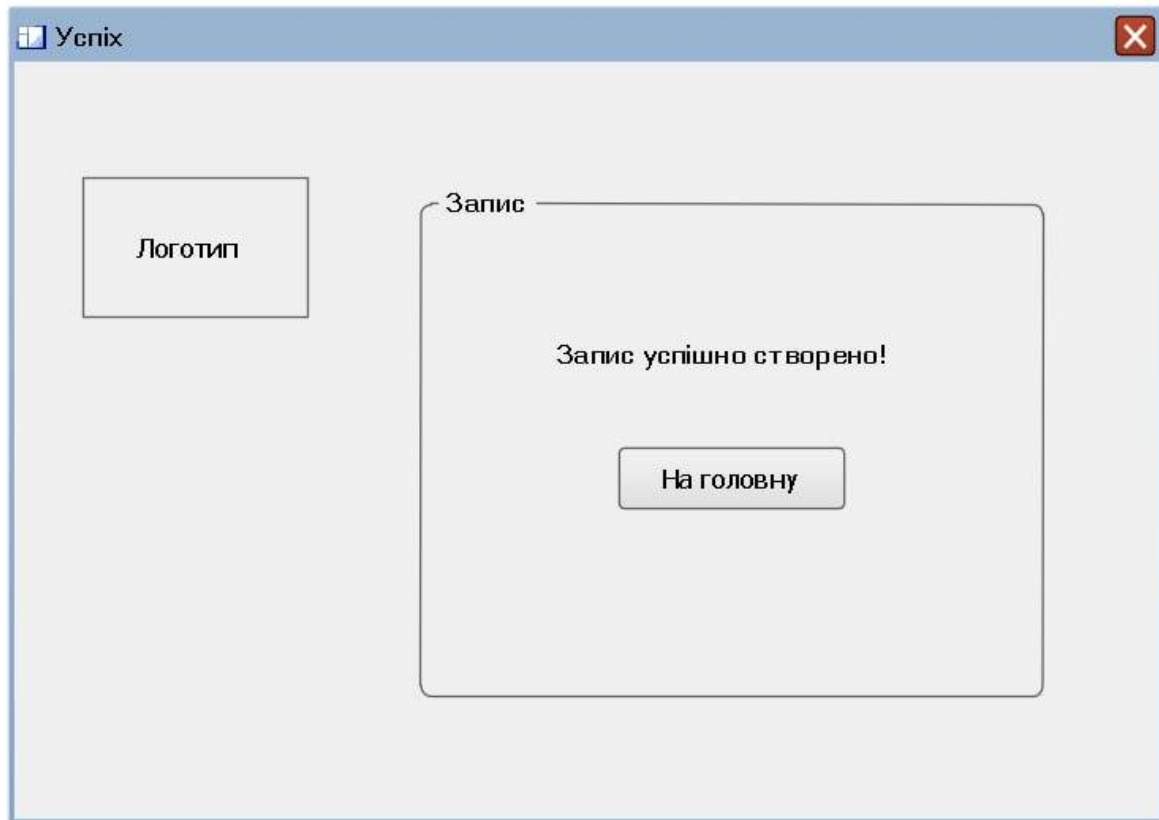


Рисунок 2.15 – Діаграма сповіщення про успішний запис

Джерело: розроблено самостійно.

Після розробки початкових моделей, наступним кроком є створення високодеталізованого інтерактивного прототипу. Цей етап дозволяє візуалізувати майбутній продукт, перевірити зручність та логіку користувацької взаємодії, а також отримати зворотний зв'язок на ранніх стадіях проєктування. Розроблений у середовищі Figma клікабельний прототип слугує наочною демонстрацією ключових функціональних можливостей та елементів інтерфейсу інформаційної системи. Нижче представлені ключові екранні форми розробленого прототипу.

Головна сторінка (рисунок 2.16) є точкою входу для всіх користувачів, містить контактну інформацію та кнопки для реєстрації та входу.

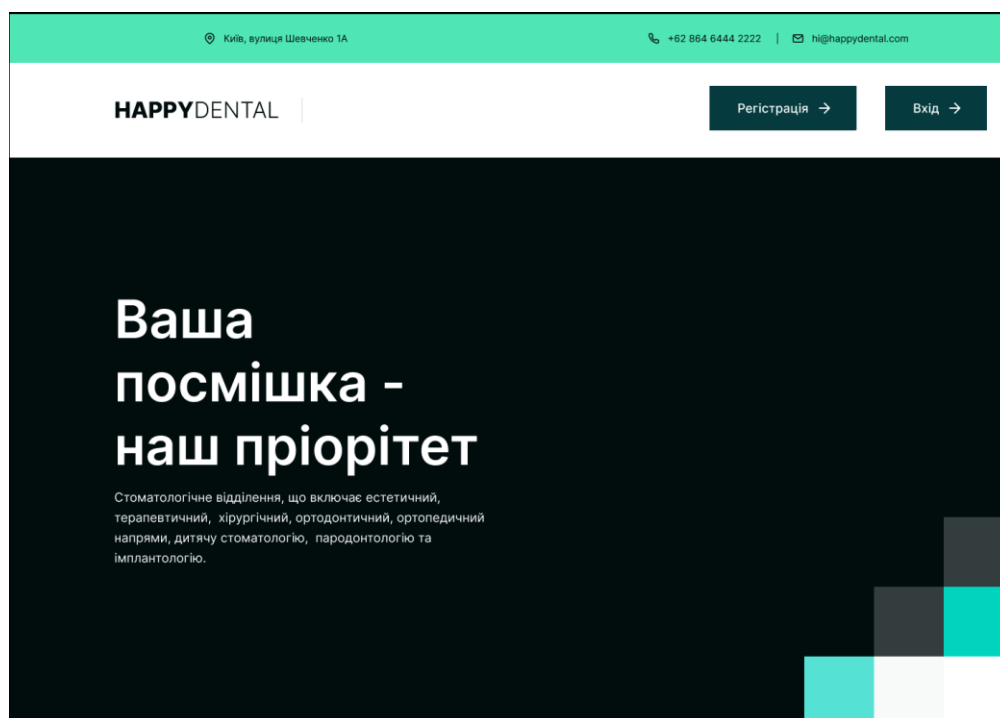


Рисунок 2.16 – Головна сторінка прототипу

Джерело: розроблено самостійно.

Форми реєстрації та авторизації (рисунки 2.17 та 2.18) реалізують функціонал створення нового облікового запису та входу в систему для існуючих користувачів, включаючи окремі сценарії для клієнта та адміністратора.

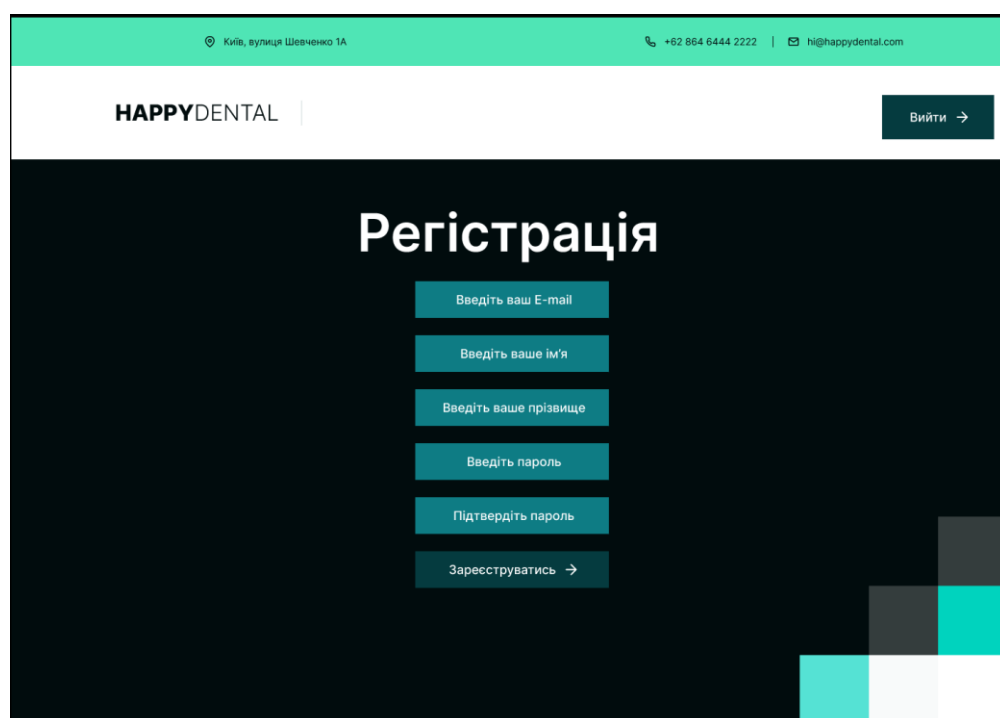


Рисунок 2.17 – Форма реєстрації

Джерело: розроблено самостійно.

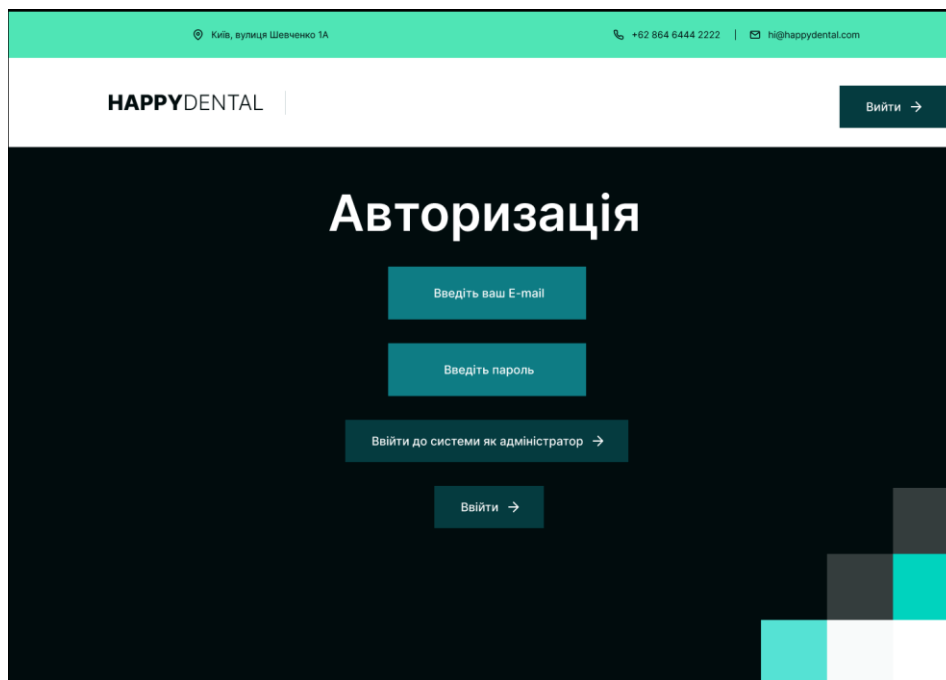


Рисунок 2.18 – Форма авторизації

Джерело: розроблено самостійно.

Після авторизації користувачі потрапляють у головне меню, вигляд якого залежить від їхньої ролі. На рисунку 2.19 показано меню для клієнта з доступом до послуг та запису до лікаря, а на рисунку 2.20 — меню для адміністратора з функціями управління клієнтами та цінами.

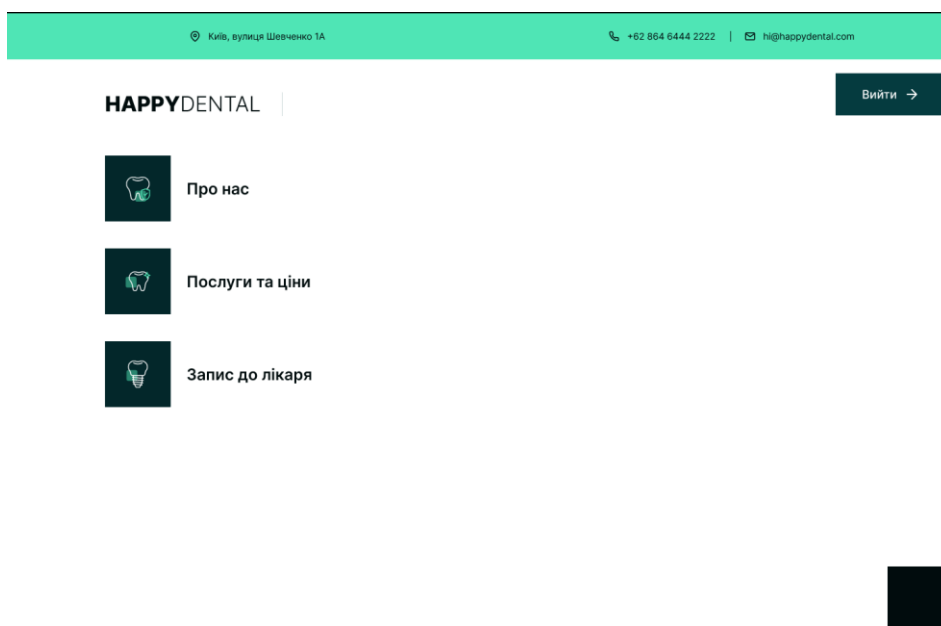


Рисунок 2.19 – Головне меню клієнта

Джерело: розроблено самостійно.

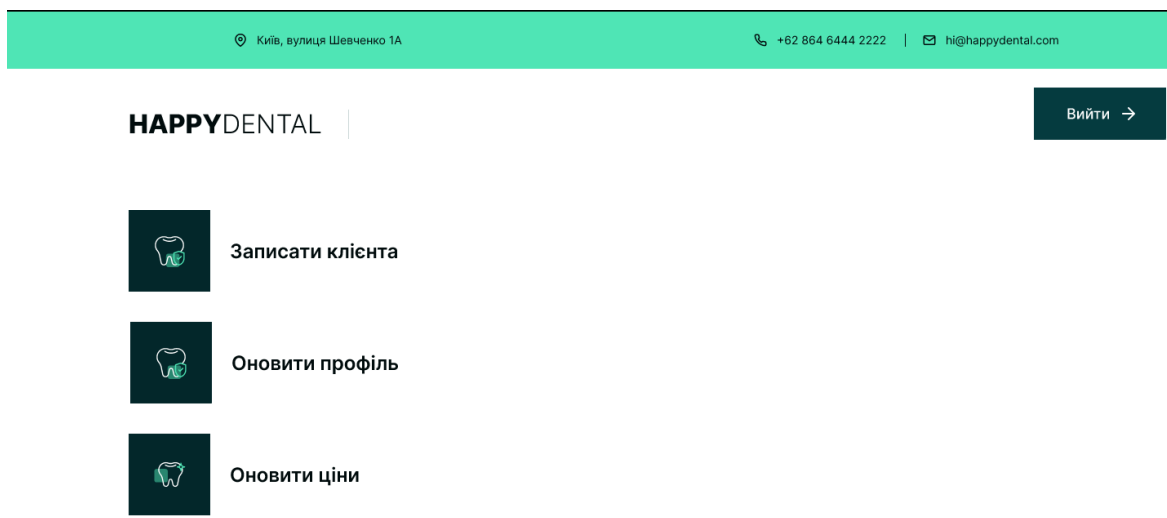


Рисунок 2.20 – Головне меню адміністратора

Джерело: розроблено самостійно.

Процес запису до лікаря включає вибір послуги з прайс-листа (рисунок 2.21) та вибір зручної дати в інтерактивному календарі (рисунок 2.22).

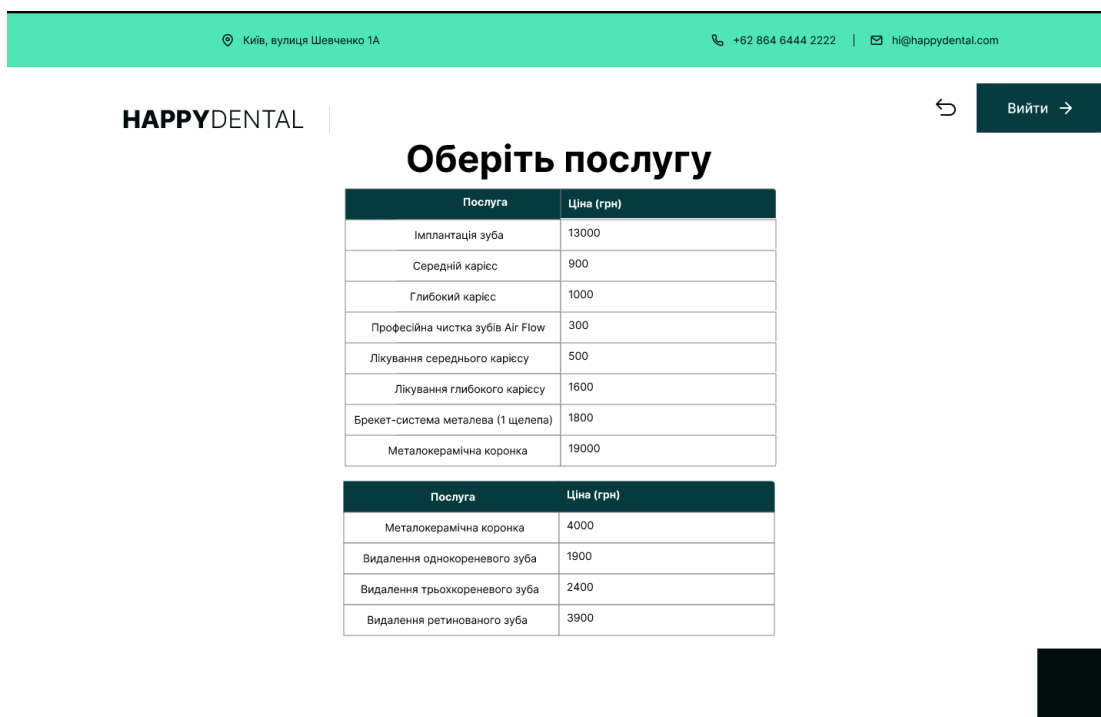


Рисунок 2.21 – Екран вибору послуги

Джерело: розроблено самостійно.

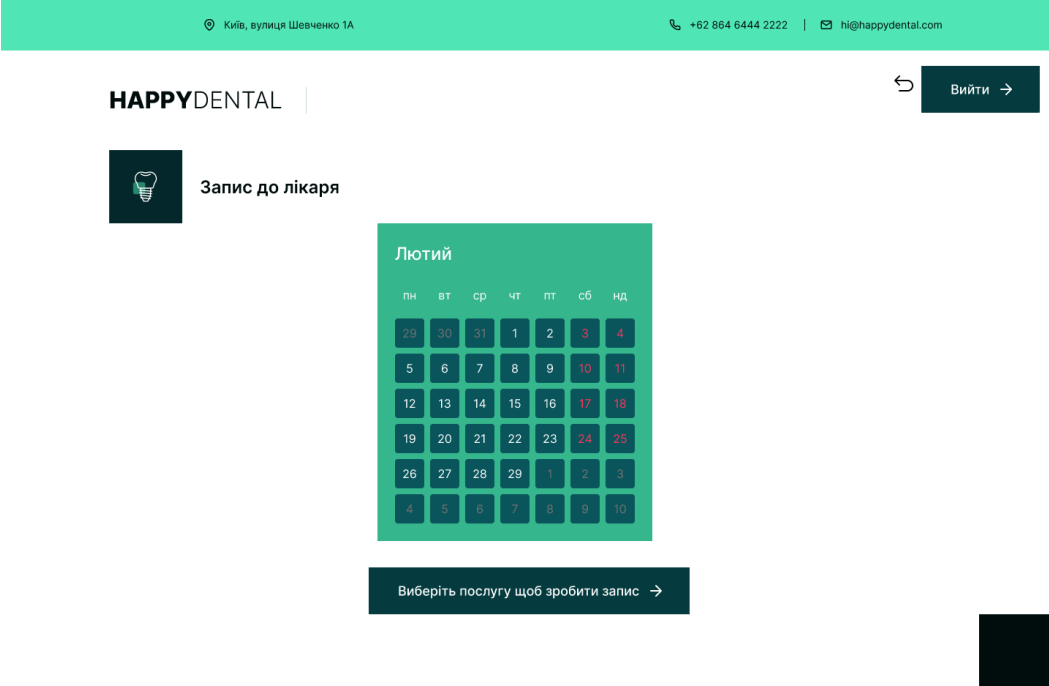


Рисунок 2.22 – Екран вибору дати

Джерело: розроблено самостійно.

Адміністративна панель надає інструменти для управління ключовими даними клініки, такі як запис клієнта (рисунок 2.23) та оновлення цін на послуги (рисунок 2.24).

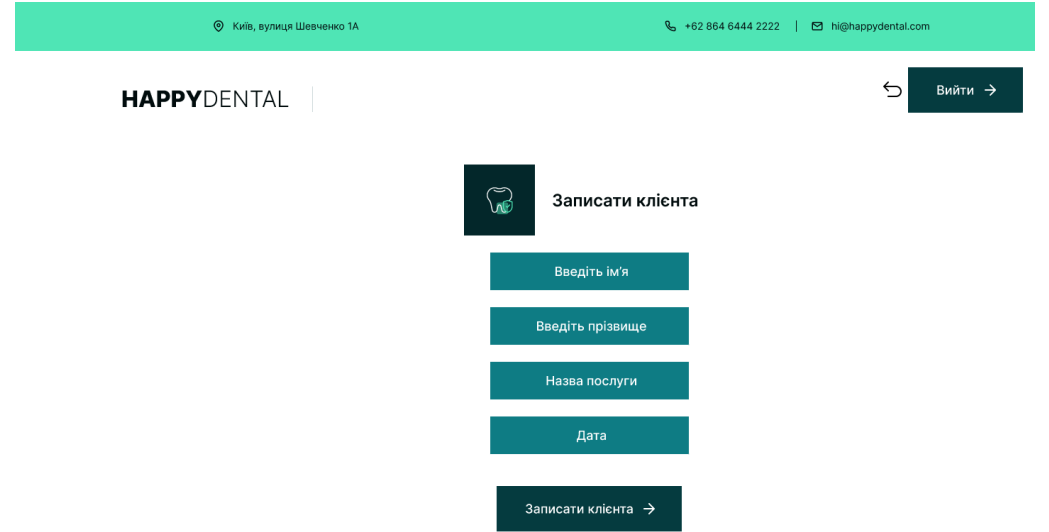


Рисунок 2.23 – Форма запису клієнта адміністратором

Джерело: розроблено самостійно.

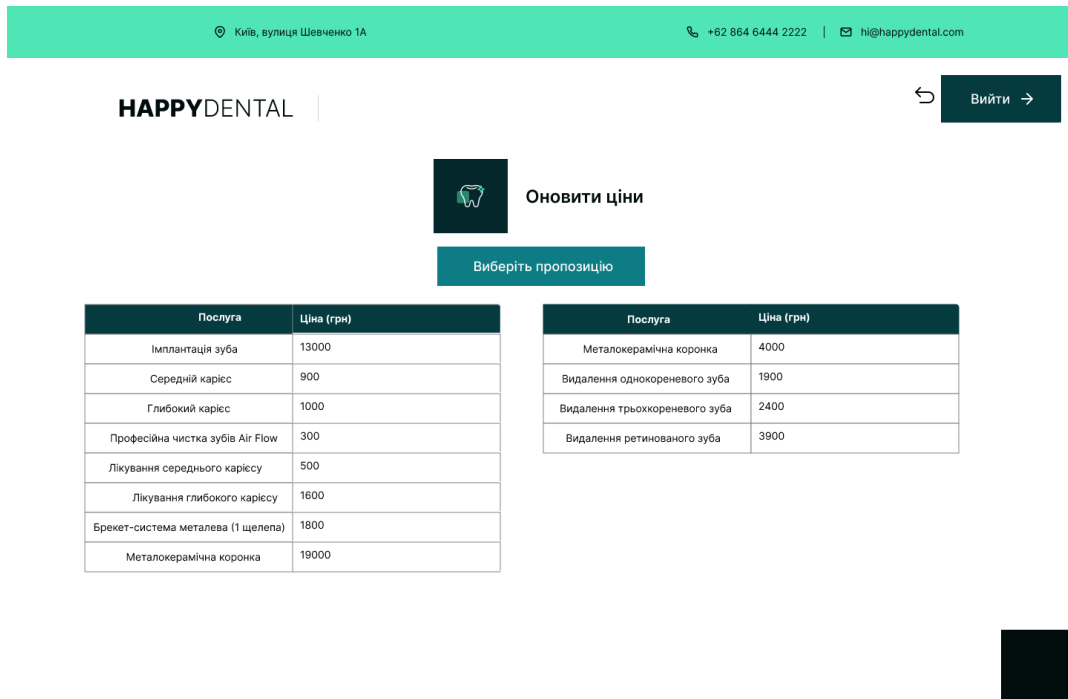


Рисунок 2.24 – Форма оновлення цін

Джерело: розроблено самостійно.

Розробка прототипів на ранніх етапах проєктування дозволяє візуалізувати майбутню систему, протестувати різні варіанти взаємодії та внести необхідні корективи до початку повноцінної розробки, що значно знижує ризики та вартість виправлення помилок на пізніх стадіях.

РОЗДІЛ 3. ПРОЄКТУВАННЯ ТА РЕАЛІЗАЦІЯ КОМПОНЕНТІВ СИСТЕМИ

Етап проєктування та реалізації є логічним продовженням аналізу та моделювання системи, що були проведені в попередніх розділах. На цьому етапі абстрактні моделі, діаграми та визначені вимоги перетворюються на конкретні технічні та програмні рішення, що формують ядро майбутньої інформаційної системи. Даний розділ присвячений детальному опису проєктних рішень для ІС стоматологічної клініки за різними видами забезпечення: інформаційного, технічного та програмного. Кожен компонент розглядається з точки зору його структури, функцій та взаємодії з іншими елементами системи. Також у цьому розділі представлено результати практичної реалізації ключових компонентів системи, що демонструють життєздатність розробленої концепції та відповідність кінцевого продукту поставленим вимогам.

3.1 Інформаційне забезпечення

Інформаційне забезпечення (ІЗ) є одним з найважливіших компонентів будь-якої інформаційної системи, оскільки воно визначає структуру, спосіб організації та методи обробки даних. ІЗ включає в себе сукупність уніфікованих систем класифікації та кодування інформації, нормативно-довідкову документацію, а також методологію побудови та ведення баз даних. Якісно спроектоване інформаційне забезпечення є запорукою цілісності, актуальності та несуперечливості даних у системі.

3.1.1 Загальна характеристика інформаційного забезпечення

Інформаційне забезпечення проєктованої системи побудовано на принципах централізованого зберігання даних та моделі документа-орієнтованої бази даних. Централізований підхід, реалізований за допомогою єдиної бази даних, дозволяє уникнути дублювання інформації, забезпечити її високу консистентність та надати уніфікований доступ до даних для всіх підсистем та користувачів. Основним носієм даних у системі є електронні сховища, що

управляються обраною системою управління базами даних (СКБД) MongoDB. Контроль достовірності та цілісності інформації реалізовано на двох рівнях: на рівні додатку, де відбувається валідація даних, що надходять від користувачів через веб-форми, та на рівні бази даних, де за допомогою визначених схем даних (Mongoose Schemas) забезпечується дотримання заданої структури та типів даних.

3.1.2 Організація збору і передання первинної інформації

Збір первинної інформації в системі організовано в інтерактивному режимі через розроблені екранні форми веб-додатку. Цей процес є повністю автоматизованим і виключає необхідність ручного введення даних з паперових носіїв. Основними джерелами інформації виступають кінцеві користувачі системи. Пацієнти надають свої персональні дані, обирають послуги та бажаний час візиту через форму запису на прийом. Адміністративний персонал, у свою чергу, відповідає за наповнення та актуалізацію довідкової інформації, зокрема, прайс-листа послуг, через спеціалізовану адміністративну панель. Технічно передання інформації від клієнтської частини (браузера) до серверної здійснюється за протоколом HTTP. Використання асинхронних запитів (fetch API) та формату JSON для передачі даних забезпечує швидку та ефективну комунікацію, що дозволяє оновлювати дані на сторінці без її повного перезавантаження, покращуючи загальний досвід користувача.

3.1.3 Побудова системи класифікації та кодування

Для впорядкування та однозначної ідентифікації об'єктів у системі використовується система класифікації та кодування. Ключовим класифікатором у проєкті є довідник стоматологічних послуг. Він містить перелік усіх послуг, що надаються клінікою, та їхню вартість. Кожна послуга має унікальне текстове найменування, яке є зрозумілим для користувача (наприклад, "Професійна чистка зубів"). Для внутрішньої ідентифікації в системі використовується механізм кодування, що надається СКБД MongoDB. Кожному

документу в колекції послуг (Prices) автоматично присвоюється унікальний 12-байтний ідентифікатор `_id` (тип `ObjectId`). Цей згенерований код виступає як первинний ключ і використовується в інших частинах системи, зокрема в колекції записів на прийом (Registers), для встановлення реляційного зв'язку. Такий підхід забезпечує цілісність посилань та дозволяє уникнути неоднозначності при посиланні на конкретну послугу.

3.1.4 Проектування форм первинних документів та відеокадрів

Оскільки система є повністю цифровою, роль первинних документів виконують екранні форми (відеокадри) веб-сайту, через які користувачі вводять дані. Проектування цих форм було зосереджено на забезпеченні максимальної зручності та інтуїтивної зрозумілості. Ключовою формою є "Запис на прийом" (рисунок 3.1), яка містить поля для введення ПІБ, номера телефону, випадаючий список для вибору послуги, елементи для вибору дати та часу, а також текстове поле для коментаря. Іншою важливою формою є "Управління прайс-листом" (рисунок 3.2), доступна в адміністративній панелі. Вона має табличне представлення списку послуг та цін, а також кнопки для виконання CRUD-операцій (додавання, редагування, видалення), що дозволяє адміністратору легко керувати контентом.


Повернутись на головну сторінку

Запис на прийом

Заповніть форму, і ми зв'яжемося з вами для підтвердження запису.

Ваше ім'я

Телефон

Оберіть послугу

Дата прийому

Час

Коментар (за бажанням)

Записатися

Або зателефонуйте нам: +38 (095) 123-45-87

Рисунок 3.1.1 – Екранна форма "Запис на прийом"

Джерело: розроблено самостійно.


Адмін-панель
Адміністратор Вийти

Управління прайс-листом

Додавайте, редагуйте та видаляйте послуги та їх ціни

+ Додати послугу

Всього послуг
15

Найдорожча
10000 грн

Найдешевша
150 грн

НАЗВА ПОСЛУГИ	ЦІНА	ДІЇ
Пломбування каналів	700 грн	Редагувати Видалити
Лікування пульпіту (3 канали)	2700 грн	Редагувати Видалити
Консультація стоматолога	520 грн	Редагувати Видалити
Лікування карієсу (2-3 поверхні)	1200 грн	Редагувати Видалити
Протезування зубів	від 4000 грн	Редагувати Видалити
Анестезія	150 грн	Редагувати Видалити
Встановлення пломби	900 грн	Редагувати Видалити
Лікування карієсу (1 поверхня)	800 грн	Редагувати Видалити
Рентген зуба	200 грн	Редагувати Видалити
Відбілювання зубів	2500 грн	Редагувати Видалити
Лікування пульпіту (2 канали)	2200 грн	Редагувати Видалити

Рисунок 3.1.2 – Екранна форма "Управління прайс-листом"

Джерело: розроблено самостійно.

3.1.5 Структура інформаційних масивів

Інформаційна база системи складається з двох основних логічних масивів, реалізованих у вигляді колекцій MongoDB: Prices (послуги) та Registers (записи на прийом). Структура кожної колекції жорстко визначена за допомогою відповідних схем, розроблених з використанням бібліотеки Mongoose.

Колекція Prices призначена для зберігання довідкової інформації про послуги та їх вартість. Кожен документ у цій колекції має наступну структуру:

1. name: тип String, обов'язкове поле. Зберігає назву послуги.
2. price: тип String, обов'язкове поле. Зберігає вартість послуги.

Колекція Registers є основною транзакційною таблицею і зберігає інформацію про записи пацієнтів на прийом. Структура документа в цій колекції є більш складною:

1. full_name: тип String, обов'язкове поле. ПІБ пацієнта.
2. phone_number: тип String, обов'язкове поле. Контактний номер телефону.
3. comment: тип String, необов'язкове поле. Додатковий коментар від пацієнта.
4. date: тип Date, обов'язкове поле. Дата запланованого візиту.
5. time: тип String, обов'язкове поле. Час запланованого візиту.
6. type_service: тип mongoose.Schema.Types.ObjectId, посилання на колекцію Price. Це поле реалізує зв'язок між записом та послугою.

3.1.6 Вибір СКБД

Для проєкту було обрано нереляційну документо-орієнтовану систему управління базами даних MongoDB. На відміну від традиційних реляційних баз даних, які вимагають жорстко визначеної схеми у вигляді таблиць, MongoDB пропонує значно більшу гнучкість, що є ключовою перевагою для сучасних веб-додатків. Таке рішення обґрунтоване кількома факторами:

Гнучкість схеми даних: Документна модель MongoDB дозволяє легко модифікувати та розширювати структуру даних без складних та

ресурсозатратних міграцій. Це особливо корисно для проєктів на ранніх стадіях, де вимоги можуть змінюватися.

Нативна підтримка JavaScript: Оскільки серверна частина розроблена на Node.js, використання MongoDB з її BSON/JSON форматом даних є природним вибором. Це усуває необхідність в ORM-перетвореннях (Object-Relational Mapping) і спрощує код, оскільки дані в базі зберігаються у форматі, близькому до об'єктів JavaScript.

Продуктивність: Для операцій, що є типовими для даного проєкту (додавання нового запису, вибірка списку послуг), MongoDB демонструє високу швидкість читання та запису.

Масштабованість: Архітектура MongoDB спроектована з розрахунком на горизонтальне масштабування (шардинг), що робить її перспективним вибором для систем, які планують зростання обсягів даних та навантаження. в

3.1.7 Інфологічна модель бази даних

Інфологічна модель, також відома як концептуальна, представляє структуру даних на високому рівні абстракції, незалежно від конкретної реалізації СКБД. Вона описує основні сутності предметної області, їхні атрибути та логічні зв'язки між ними. Для даної системи була розроблена комплексна інфологічна модель, візуалізована на рисунку 3.1.8.

Модель включає такі ключові сутності:

1. Users — центральна сутність для автентифікації та авторизації. Зберігає облікові дані (email, password) та role (наприклад, «пацієнт», «лікар», «адміністратор»), що дозволяє розмежовувати доступ до функціонала системи.
2. Doctors та Patients — спеціалізовані сутності, що розширюють Users. Вони зберігають профільну інформацію: full_name та specialization для лікаря; full_name та phone_number для пацієнта. Кожен Doctor та Patient має обов'язковий зв'язок з відповідним User (відношення «один-до-одного»).

3. Services — довідник, що містить перелік усіх послуг, які надає клініка, з їх назвою (name) та вартістю (price).
4. Appointments — основна транзакційна сутність, що фіксує факт запису на прийом. Вона пов'язує пацієнтів, лікарів та послуги, зберігаючи date і time візиту. Ця сутність має зв'язки типу «багато-до-одного» з сутностями Patients, Doctors та Services.
5. MedicalRecords — сутність для ведення електронних медичних карток. Кожна картка належить одному пацієнту (patient_id) і містить масив записів (entries), що дозволяє зберігати повну історію лікування.

Таким чином, розроблена інфологічна модель комплексно описує всі інформаційні об'єкти та взаємозв'язки, що є необхідними для автоматизації процесів стоматологічної клініки, та слугує фундаментом для фізичного проектування бази даних

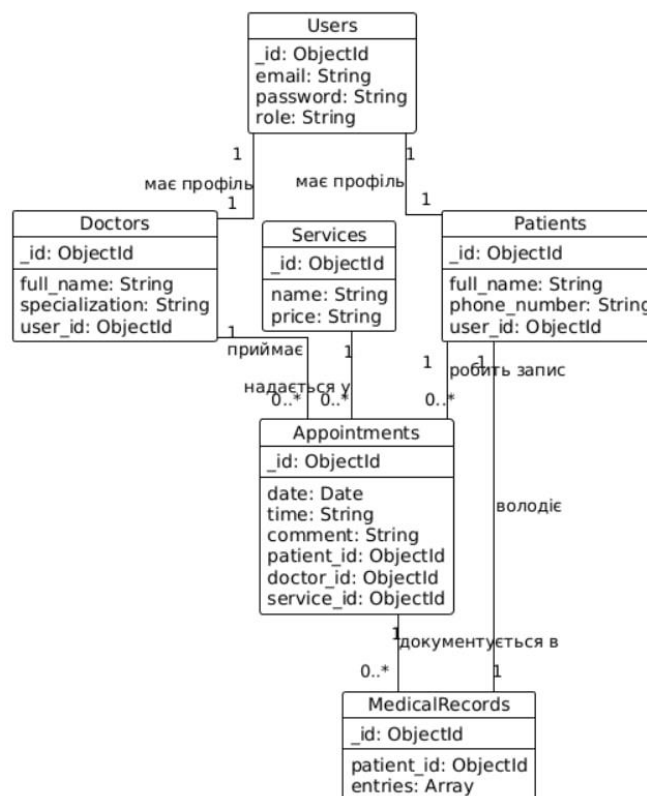


Рисунок 3.1.3 – Інфологічна модель бази даних

Джерело: розроблено самостійно.

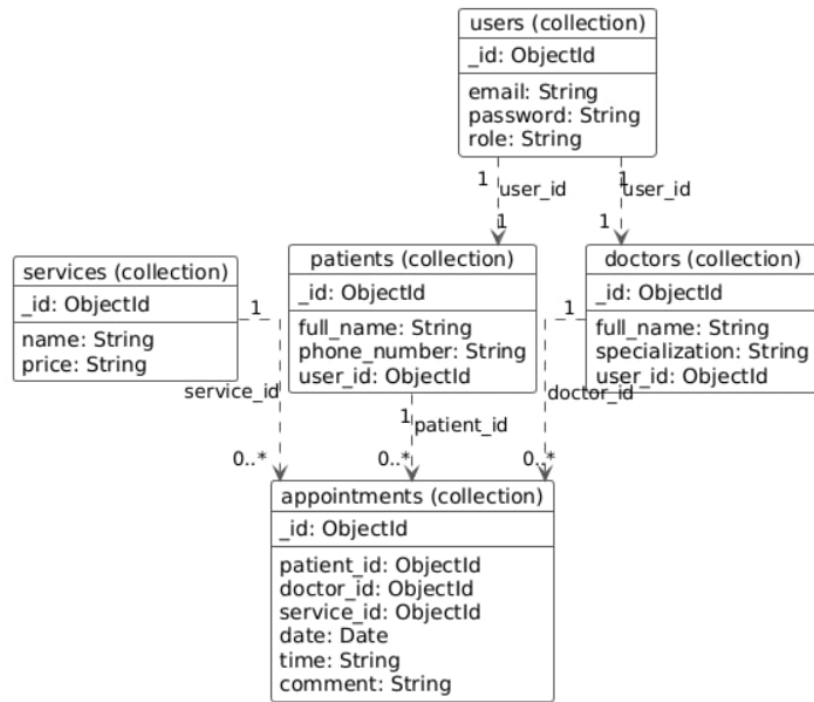
Даталогічна (або фізична) модель описує, як саме інфологічна модель реалізована в середовищі конкретної обраної СКБД — MongoDB. У випадку з документо-орієнтованою базою даних, сутності представлені як колекції, а їхні екземпляри — як документи у форматі BSON/JSON.

На відміну від реляційних СУБД, зв'язки в MongoDB реалізуються на рівні додатку. Для даного проєкту було обрано підхід посилань (referencing), який є аналогом використання зовнішніх ключів. Цей підхід візуалізовано на схемі даталогічної моделі.

Ключові аспекти реалізації:

1. Сутності як Колекції: Кожна сутність з інфологічної моделі (Users, Patients, Doctors, Services, Appointments) перетворена на відповідну колекцію в MongoDB (users, patients, doctors і т.д.).
2. Реалізація зв'язків: Зв'язки типу «один-до-багатьох» реалізовано через зберігання ідентифікаторів. Наприклад, у кожному документі колекції appointments присутні поля patient_id, doctor_id та service_id. Ці поля зберігають унікальні значення _id (ObjectId) відповідних документів з колекцій patients, doctors та services.
3. Зв'язок профілів: Аналогічно, зв'язок між загальним профілем користувача та його спеціалізованою роллю реалізовано через поле user_id у колекціях patients та doctors, яке посилається на документ у колекції users.

Такий підхід є ефективним, оскільки він дозволяє уникнути дублювання даних (наприклад, повне ім'я пацієнта не копіюється в кожен його запис на прийом) та забезпечує гнучкість структури. Фактично, код схем Mongoose, що використовується в програмному коді, є формальним описом цієї даталогічної моделі



3.1.4 Даталогічна модель бази даних

Джерело: розроблено самостійно

3.2 Технічне забезпечення

Технічне забезпечення визначає комплекс апаратних засобів, обчислювальних мереж та комунікаційного обладнання, необхідних для надійного та ефективного функціонування інформаційної системи.

3.2.1 Загальні положення та схема автоматизації

На поточному етапі життєвого циклу проєкту, що включає розробку та тестування, інформаційна система розгорнута на локальному сервері. Такий підхід є стандартною практикою в сучасній веб-розробці, оскільки дозволяє швидко вносити зміни та тестувати функціонал без необхідності розгортання на віддаленому сервері. Всі компоненти системи — клієнтська частина, серверна частина та база даних — функціонують в межах однієї фізичної машини, комп'ютера розробника.

3.2.2 Структура комплексу технічних засобів та 3.2.3 Опис автоматизованого робочого місця (АРМ)

В умовах локального розгортання поняття "комплекс технічних засобів" (КТЗ) та "автоматизоване робоче місце" (АРМ) є тотожними і представляють собою персональний комп'ютер, на якому ведеться розробка. Аналізуючи технологічний стек проєкту (Node.js, MongoDB, сучасні веб-браузери), можна сформулювати наступні мінімальні технічні вимоги до АРМ для комфортної розробки та функціонування системи:

Операційна система: Windows 10/11, macOS 10.15 (Catalina) і вище, або сучасний дистрибутив Linux (наприклад, Ubuntu 20.04+).

Процесор (CPU): Мінімум 2-ядерний процесор з тактовою частотою від 2.2 ГГц (наприклад, Intel Core i3, AMD Ryzen 3 або краще) для забезпечення стабільної роботи середовища виконання Node.js та фонових процесів СКБД.

Оперативна пам'ять (RAM): Мінімум 8 ГБ. Цей об'єм є рекомендованим для забезпечення плавної одночасної роботи операційної системи, редактора коду (напр., VS Code), терміналу, локального сервера Node.js, служби MongoDB та веб-браузера з відкритими інструментами розробника.

Вільне місце на диску: Мінімум 5 ГБ для розміщення файлів проєкту, встановлення Node.js, СКБД MongoDB, залежностей та зберігання даних у базі.

3.2.3 Схема мережі передачі даних

Для локального розгортання на localhost схема мережевої взаємодії є максимально простою, оскільки весь обмін даними відбувається всередині однієї машини через локальний мережевий інтерфейс (loopback interface, 127.0.0.1). Фізична мережева інфраструктура не задіюється. Комунікація між компонентами відбувається за протоколом TCP/IP через різні порти для розмежування служб (наприклад, веб-сервер на порту 3000, а MongoDB на стандартному порту 27017).

3.3 Програмне забезпечення

При розробці програмних компонентів системи значна увага приділялася не лише реалізації функціоналу, але й забезпеченню високої якості програмного коду. Основою для цього стали принципи, викладені у праці Роберта Мартіна «Чистий код: створення, аналіз та рефакторинг» [8] . Застосування цих підходів допомогло у створенні структурованого, читабельного та самодокументованого коду, що є критично важливим для подальшої підтримки, масштабування та розвитку інформаційної системи. Програмне забезпечення (ПЗ) є сукупністю програмних та документальних засобів, що забезпечують створення та експлуатацію системи. Воно поділяється на системне, яке створює операційне середовище, та прикладне, яке реалізує бізнес-логіку.

3.3.1 Структура програмного забезпечення

Проект реалізовано за класичною трирівневою клієнт-серверною архітектурою, яка є стандартом для сучасних веб-додатків. Ця структура включає:

1. Клієнтська частина (Frontend): Шар представлення, з яким безпосередньо взаємодіє користувач. Реалізований за допомогою стандартних веб-технологій: HTML для розмітки, CSS (з використанням фреймворку Tailwind CSS для швидкого стилізового оформлення) та нативного JavaScript для обробки подій та асинхронних запитів до сервера.
2. Серверна частина (Backend): Шар бізнес-логіки. Побудована на програмній платформі Node.js з використанням веб-фреймворку Express.js. Ця частина відповідає за обробку запитів від клієнта, реалізацію бізнес-правил, автентифікацію та взаємодію з базою даних.
3. Шар даних (Data Layer): Реалізований за допомогою СКБД MongoDB та бібліотеки Mongoose, яка виступає як шар об'єктного

моделювання даних (ODM), спрощуючи роботу з базою даних з коду додатку.

3.3.2 Системне програмне забезпечення

Основою для функціонування розробленої системи є набір системного програмного забезпечення та бібліотек, які створюють необхідне операційне середовище. Згідно з файлом `package.json`, ключовими компонентами є:

Середовище виконання Node.js: Асинхронна платформа на основі рушія V8, що дозволяє виконувати JavaScript-код на сервері.

СКБД MongoDB: Система управління базами даних.

express: Мінімалістичний та гнучкий веб-фреймворк для Node.js, що є основою для побудови API та маршрутизації.

mongoose: Бібліотека для об'єктного моделювання даних (ODM) для MongoDB, що надає зручний інтерфейс для роботи з даними, включаючи валідацію, визначення схем та виконання запитів.

hjs (Hogan.js): Шаблонізатор, що використовується для рендерингу HTML-сторінок на сервері.

morgan, cookie-parser, debug: Допоміжні бібліотеки (middleware) для Express, що використовуються для логування HTTP-запитів, розбору cookie та налагодження додатку.

3.3.3 Прикладне програмне забезпечення

Прикладне ПЗ — це унікальний код, розроблений спеціально для вирішення завдань даного проєкту. Воно складається з двох основних частин.

Frontend-додаток: Включає в себе HTML-файли з розширенням `.hjs`, які є шаблонами для відображення сторінок, а також клієнтські JavaScript-скрипти. Прикладом є скрипт на головній сторінці, який асинхронно запитує у сервера список послуг і динамічно формує прайс-лист на сторінці.

Backend-додаток (api.js): Є ядром системи. Цей файл містить реалізацію RESTful API. Наприклад, маршрут POST /register відповідає за логіку реєстрації на прийом: він приймає дані з тіла запиту, проводить їх валідацію, знаходить відповідну послугу в колекції Prices для зв'язування, створює новий документ Register та зберігає його в базі даних. Аналогічно реалізовано маршрути для отримання, додавання та оновлення послуг (GET /price_list, POST /price_list, PUT /price_list/:id).

3.3.4 Програмна документація

На даному етапі розробки, для проєкту такого масштабу, основною програмною документацією є сам вихідний код. Сучасні практики розробки, такі як дотримання чистоти коду, використання зрозумілих імен змінних та функцій, а також чітка структура проєкту, роблять код значною мірою самодокументованим. Наприклад, структура API-маршрутів у файлі api.js чітко відображає функціональні можливості системи. Крім того, діаграми UML та SysML, розроблені в другому розділі, слугують високорівневою візуальною документацією архітектури та поведінки системи. Для більш складних систем у майбутньому може бути створено окреме керівництво користувача та адміністратора.

3.4 Результати реалізації інформаційної системи

В результаті виконання проєктних та програмних робіт було створено функціональний прототип інформаційної системи у вигляді веб-додатку "Усмішка", що реалізує ключові бізнес-процеси стоматологічної клініки. Розроблений прототип повністю відповідає вимогам, визначеним у другому розділі, та демонструє життєздатність обраних архітектурних та технологічних рішень.

Реалізований веб-сайт (рисунок 3.5) має сучасний та адаптивний дизайн, що забезпечує зручну взаємодію як на настільних комп'ютерах, так і на

мобільних пристроях (вимога NR1.2). Головна сторінка надає загальну інформацію про клініку та її послуги.

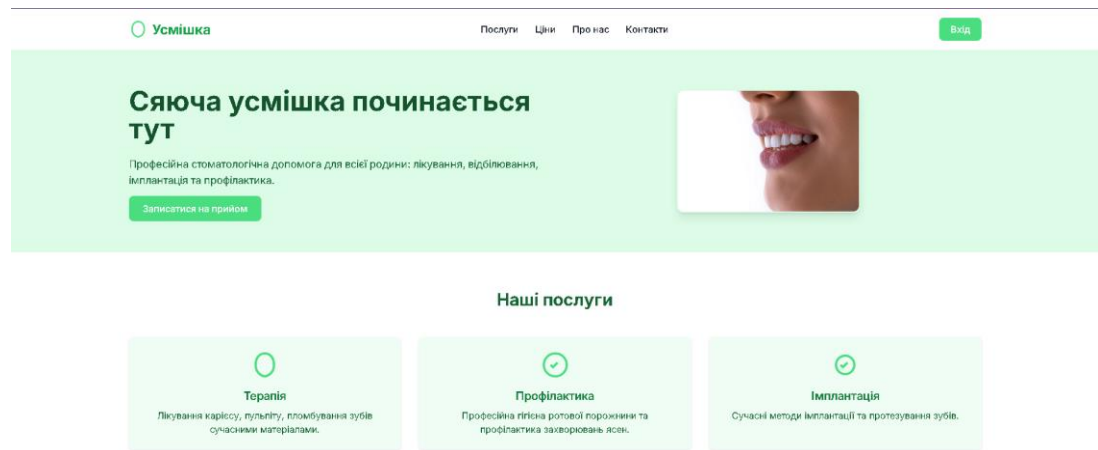


Рисунок 3.5.1 – Головна сторінка реалізованого сайту

Джерело: розроблено самостійно.

Прайс-лист	
Пломбування каналів	700 грн
Лікування пульпиту (3 канали)	2700 грн
Консультація стоматолога	520 грн
Лікування карієсу (2-3 поверхні)	1200 грн
Протезування зубів	від 4000 грн
Анастезія	150 грн
Встановлення пломби	900 грн
Лікування карієсу (1 поверхня)	800 грн
Рентген зуба	200 грн
Відбілювання зубів	2500 грн
Лікування пульпиту (2 канали)	2200 грн
Імплантація зуба	від 10000 грн
Лікування пульпиту (1 канал)	1500 грн
Гігієнічне чищення зубів	1000 грн
Зняття зубного каменю	800 грн

*Ціни актуальні станом на червень 2025 року

Детальну консультацію та індивідуальний план лікування можна отримати на прийомі!

Рисунок 3.5.2 – Головна сторінка реалізованого сайту

Джерело: розроблено самостійно.



Рисунок 3.5.3 – Головна сторінка реалізованого сайту

Джерело: розроблено самостійно.

Ключовий функціонал, що задовольняє вимогу FR2.1, — це можливість онлайн-запису на прийом. Користувач може легко перейти до форми запису (рисунок 3.1), заповнити необхідні поля та надіслати заявку. Дані обробляються бекендом і зберігаються в базі даних MongoDB, що підтверджує успішну реалізацію серверної логіки.

Для адміністративного персоналу було реалізовано панель управління (рисунок 3.2), що дозволяє керувати переліком послуг та їх вартістю. Ця функція є прямою реалізацією потреби в гнучкому управлінні довідковими даними. Зміни, внесені через адмін-панель, миттєво відображаються на публічній частині сайту завдяки динамічному завантаженню даних, що було продемонстровано в реалізації прайс-листа. Таким чином, створений прототип є не просто набором веб-сторінок, а повноцінною системою з клієнтською, серверною частинами та базою даних, що успішно реалізує базовий цикл взаємодії різних ролей користувачів та є міцною основою для подальшого розвитку.

Окрім функціоналу запису, в рамках реалізації було розроблено інтерфейс для оплати стоматологічних послуг онлайн. Це дозволяє клієнтам швидко та безпечно розраховуватися за отримані послуги, що підвищує якість сервісу. Приклад реалізованої сторінки оплати наведено на рисунку 3.6.

Оплата услуг

Швидка та безпечна оплата стоматологічних послуг онлайн

Вибрана послуга

Лікування пульпіту (3 канали) - 2700 грн

Кількість: 1

Дані пацієнта

i

1

Підсумок замовлення

Послуга: Лікування пульпіту (3 канали)

Кількість: 1

До сплати: 2700 грн

Дані для оплати

Спосіб оплати

☒ Картка

☐ Переказ

Номер картки

1234 5678 9012 3456

Термін дії

CVV

MM/YY

123

Ім'я власника картки

IVAN PETROV

Безпечна оплата

Всі дані передаються через захищене з'єднання SSL. Ваші фінансові дані не зберігаються на наших серверах.

Сплатити 2700 грн

Потрібна допомога?

Зв'яжіться з нами, якщо у вас виникли питання щодо оплати

+7 {495} 123-45-67

info@ulybka.ru

Рисунок 3.6 – Інтерфейс сторінки "Оплата услуг".

Джерело: розроблено самостійно.

ВИСНОВОК

У ході виконання кваліфікаційної бакалаврської роботи було успішно досягнуто поставленої мети, яка полягала у розробці концепції та проектних рішень для інформаційної системи стоматологічної клініки. Створений функціональний прототип доводить життєздатність обраних рішень та має практичне значення для автоматизації діяльності медичних закладів.

За результатами проведеної роботи було зроблено такі висновки:

1. Проаналізовано предметну галузь та обґрунтовано актуальність проєкту. У першому розділі було проведено аналіз бізнес-процесів сучасної стоматологічної клініки та досліджено існуючі на ринку інформаційні системи. Виявлено, що готові рішення часто є або надто узагальненими, або мають обмежений функціонал, що не задовольняє унікальні потреби клінік. Це підтвердило доцільність розробки кастомізованої ІС, орієнтованої на конкретні завдання закладу.
2. Розроблено комплексну модель вимог та спроектовано архітектуру системи. У другому розділі були детально визначені та специфіковані бізнес-вимоги, функціональні та нефункціональні вимоги до системи. На основі цих вимог було створено моделі, що описують поведінку (діаграми прецедентів та послідовності UML) та структуру системи (діаграма визначення блоків SysML). Це дозволило сформулювати чітке та узгоджене бачення майбутнього продукту.
3. Створено інтерактивний прототип користувацького інтерфейсу. Важливою частиною проєктної роботи стала розробка клікабельного прототипу в середовищі Figma. Було спроектовано ключові екранні форми, зокрема головну сторінку, форми реєстрації та авторизації, інтерфейси для запису на прийом та адміністративну панель. Це дозволило візуалізувати користувацьку взаємодію та перевірити зручність навігації ще до етапу програмування.
4. Реалізовано функціональний прототип веб-додатку. У третьому розділі було обґрунтовано вибір технологічного стеку, зокрема нереляційної

СКБД MongoDB та платформи Node.js. Було спроектовано інфологічну та даталогічну моделі бази даних. Результатом практичної роботи став функціональний прототип веб-додатку “Усмішка”, який реалізує ключовий функціонал: онлайн-запис на прийом для пацієнтів та управління переліком послуг і цін для адміністратора.

5. Підтверджено практичну значущість отриманих результатів. Розроблений прототип є міцним фундаментом для подальшої реалізації повнофункціональної інформаційної системи. Впровадження такої системи в діяльність стоматологічної клініки сприятиме підвищенню ефективності роботи персоналу, оптимізації управлінської діяльності, покращенню якості обслуговування пацієнтів та підвищенню конкурентоспроможності закладу.

Таким чином, усі завдання, поставлені в рамках кваліфікаційної роботи, були повністю виконані. Робота демонструє засвоєння теоретичних знань та практичних навичок з аналізу, проєктування та розробки інформаційних систем. Напрямами для подальшого розвитку проєкту є розширення функціоналу (ведення електронних медичних карток, управління графіками лікарів, фінансовий облік), інтеграція зі сторонніми сервісами та розгортання системи у хмарній інфраструктурі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Адаптивний дизайн: повний посібник / Web.dev by Google. URL: <https://web.dev/learn/design/> (дата звернення: 12.06.2025).
2. Бондаренко С. М. Основи розробки веб-застосунків : навч. посіб. Харків : ХНУ імені В. Н. Каразіна, 2018. 172 с.
3. Гайд по роботі з API в JavaScript за допомогою Fetch / DigitalOcean Community. URL: <https://www.digitalocean.com/community/tutorials/how-to-use-the-fetch-api-with-node-js> (дата звернення: 12.06.2025).
4. Інфологічне моделювання даних / Studfile. URL: <https://studfile.net/preview/5425624/page:2/> (дата звернення: 12.06.2025).
5. Інформаційна безпека в медичних системах: відповідність HIPAA та GDPR / Блог "Security Boulevard". URL: <https://securityboulevard.com/2023/09/hipaa-and-gdpr-compliance-in-healthcare-it/> (дата звернення: 12.06.2025).
6. Керівництво по валідації даних в Mongoose / FreeCodeCamp. URL: <https://www.freecodecamp.org/news/introduction-to-mongoose-for-mongodb-d2a7aa593c57/> (дата звернення: 12.06.2025).
7. Козачук І. А. Проектування інтерфейсів користувача для веб-орієнтованих систем // Матеріали конференції "Інформаційні технології в освіті та науці". Київ : КПІ ім. Ігоря Сікорського, 2022. С. 112–115.
8. Мартін Р. Чистий код: створення, аналіз та рефакторинг. Київ : Видавнича група BHV, 2021. 448 с.
9. Методологія розробки ПЗ: порівняння Waterfall, Agile, Scrum, Kanban / Блог Atlassian. URL: <https://www.atlassian.com/agile/project-management/project-management-methodologies> (дата звернення: 12.06.2025).
10. Основи тестування програмного забезпечення: посібник для початківців / Guru99. URL: <https://www.guru99.com/software-testing-introduction-importance.html> (дата звернення: 12.06.2025).
11. Офіційна документація Node.js / Node.js Foundation. URL: <https://nodejs.org/uk/docs/> (дата звернення: 12.06.2025).

- 12.Петренко І. С., Ковальчук О. А. Впровадження електронних систем в закладах охорони здоров'я: виклики та перспективи в Україні // Ефективна економіка. 2023. № 11. URL: <http://www.economy.nayka.com.ua/?op=1&z=9541> (дата звернення: 12.06.2025).
- 13.Проектування RESTful API: найкращі практики / Smashing Magazine. URL: <https://www.smashingmagazine.com/2018/01/designing-restful-apis/> (дата звернення: 12.06.2025).
- 14.Ситник В. Ф., Писаревська Т. А., Єрьоміна Н. В., Краєва О. С. Основи інформаційних систем : навч. посіб. 2-ге вид., перероб. і доп. Київ : КНЕУ, 2001. 420 с.
- 15.Ситник Н. В. Проектування баз і сховищ даних : навч. посіб. Київ : КНЕУ, 2004. 348 с.
- 16.Системи автоматизації в медицині: огляд рішень для клінік / Блог MedEir. URL: <https://medeir.com/blog/avtomatizacziya-medichnih-zakladiv> (дата звернення: 12.06.2025).
- 17.Соціальна інженерія та захист даних в медичних закладах / Інформаційний портал "SecurityLab". URL: <https://www.securitylab.ru/analytics/514210.php> (дата звернення: 12.06.2025).
- 18.Тішков Б. О. Системний аналіз інформаційних технологій. Київ : КНЕУ, 2019. 280 с.
- 19.Цифрова трансформація в охороні здоров'я України / Портал "Ефективна медицина". URL: <https://effective-medicine.com/digital-transformation-in-healthcare> (дата звернення: 12.06.2025).
- 20.Як створити ER-діаграму для бази даних / Блог компанії DBeaver. URL: <https://dbeaver.com/2023/05/16/how-to-create-an-er-diagram-for-a-database/> (дата звернення: 12.06.2025).
- 21.Business Process Model and Notation (BPMN) / Object Management Group. URL: <https://www.bpmn.org/> (дата звернення: 12.06.2025).

- 22.Express.js - веб-фреймворк для Node.js / Express.js. URL: <https://expressjs.com/> (дата звернення: 12.06.2025).
- 23.Figma: The Collaborative Interface Design Tool / Figma, Inc. URL: <https://www.figma.com/> (дата звернення: 12.06.2025).
- 24.Full-stack розробка з Node.js, Express та MongoDB / DOU.ua. URL: <https://dou.ua/lenta/articles/full-stack-node-express-mongodb/> (дата звернення: 12.06.2025).
- 25.MongoDB Documentation / MongoDB, Inc. URL: <https://www.mongodb.com/docs/> (дата звернення: 12.06.2025).
- 26.Mongoose ODM - моделювання об'єктів для MongoDB / Mongoose.js. URL: <https://mongoosejs.com/docs/guide.html> (дата звернення: 12.06.2025).
- 27.Object-Process Methodology ISO/PAS 19450:2015 / International Organization for Standardization. URL: <https://www.iso.org/standard/65303.html> (дата звернення: 12.06.2025).
- 28.OMG Systems Modeling Language (SysML) / Object Management Group. URL: <https://www.omg.org/spec/SysML/> (дата звернення: 12.06.2025).
- 29.OMG Unified Modeling Language (UML) / Object Management Group. URL: <https://www.omg.org/spec/UML/> (дата звернення: 12.06.2025).
- 30.Tailwind CSS - документація / Tailwind Labs. URL: <https://tailwindcss.com/docs/> (дата звернення: 12.06.2025).

ДОДАТКИ

Додаток А

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
 МІНІСТЕРСТВО З ПИТАНЬ СТРАТЕГІЧНИХ ГАЛУЗЕЙ ПРОМИСЛОВОСТІ УКРАЇНИ
 Київський національний економічний університет імені Вадима Гетьмана
 Вроцлавський університет економіки та бізнесу (Польща)
 Інститут інновацій та конкуренції Макса Планка (Німеччина, Мюнхен)
 Національний університет Ірландії в Мейноті (Ірландія, Мейнот)
 Інститут кібернетики імені В.М. Глушкова НАН України
 Інститут проблем моделювання в енергетиці ім. Г.Є. Пухова НАН України
 Інститут проблем штучного інтелекту МОН та НАН України
 Інститут телекомунікацій і глобального інформаційного простору НАН України
 Запорізький національний університет
 Івано-Франківський національний технічний університет нафти і газу
 Національний технічний університет України «КПІ ім. Ігоря Сікорського»
 Національний університет «Острозька академія»
 Одеський національний економічний університет
 Харківський національний університет радіоелектроніки
 Хмельницький національний університет

СУЧАСНІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА СИСТЕМИ В УПРАВЛІННІ

Збірник матеріалів

VI Міжнародної науково-практичної конференції
молодих вчених, аспірантів і студентів

10–11 квітня 2025 р.



Рис. А-1. Титульний лист збірника матеріалів конференції

<i>Небава О. М.</i> ЦИФРОВІЗАЦІЯ ЯК КЛЮЧОВИЙ ЧИННИК РОЗВИТКУ ІНТЕЛЕКТУАЛЬНОГО ПОТЕНЦІАЛУ ПІДПРИЄМСТВА	41
<i>Плетос Е. А., Орлик О. В.</i> СУЧАСНІ ЗАСОБИ КОМУНІКАЦІЇ З КЛІЄНТАМИ В БІЗНЕСІ.....	42
<i>Пригунов М. С.</i> ВПРОВАДЖЕННЯ АВТОМАТИЗОВАНИХ СИСТЕМ УПРАВЛІННЯ ГОТЕЛЕМ ДЛЯ ПІДВИЩЕННЯ ЯКОСТІ ОБСЛУГОВУВАННЯ ГОСТЕЙ.....	44
<i>Попович К. С., Гатак Н.М.</i> СУЧАСНІ ТРЕНДИ В КАДРОВІЙ ПОЛІТИЦІ ПІДПРИЄМСТВ В УМОВАХ ЦИФРОВІЗАЦІЇ	46
<i>Руденко К. А.</i> ІНФОРМАЦІЙНІ СИСТЕМИ І ТЕХНОЛОГІЇ УПРАВЛІННЯ В ІННОВАЦІЙНІЙ ЕКОНОМІЦІ.....	48
<i>Садуллаєв В. Е.</i> ІНФОРМАЦІЙНА СИСТЕМА СТОМАТОЛОГІЧНОЇ КЛІНІКИ	49
<i>Самборський Д. О.</i> СУЧАСНІ ІНСТРУМЕНТИ ТА ТЕХНОЛОГІЇ ДЛЯ АВТОМАТИЗОВАНОГО ВИЯВЛЕННЯ АНОМАЛІЙ У ВИРІВНИЧИХ ПРОЦЕСАХ	51
<i>Стасюцький М. О.</i> ЕВОЛЮЦІЯ ТА ІНТЕГРАЦІЯ ПІДХОДІВ ДО РОЗРОБКИ КОНКУРЕНТНИХ СТРАТЕГІЙ БІЗНЕС-ОРГАНІЗАЦІЙ.....	53
<i>Флюод Д. В.</i> ІНФОРМАЦІЙНА СИСТЕМА ДЛЯ УПРАВЛІННЯ РОБОТОЮ САПЕРНОГО ОПЕРАТОРА ...	54
<i>Харченко Е.В.</i> ТЕХНОЛОГІЧНІ ПРОГРЕСИ ЯК ПІДХІД УДОСКОНАЛЕННЯ ЕКОНОМІКИ	57
<i>Циганенко Г.В.</i> РОЛЬ ІННОВАЦІЙ ТА ТЕХНОЛОГІЧНИХ ЗМІН У ЗАБЕЗПЕЧЕННІ СТІЙКОСТІ ПІДПРИЄМСТВ В КРИЗОВИХ УМОВАХ	58
<i>Шаманський А. А.</i> ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ДЛЯ КОМЕРЦІЇ	60
<i>Яворська М.І.</i> INSTAGRAM ЯК МАРКЕТИНГОВИЙ ІНСТРУМЕНТ ДЛЯ БІЗНЕСУ	62

Рис. А-2. Зміст матеріалів конференції

*Садуллаєв В. Е., студент
Київський національний економічний
університет імені Вадима Гетьмана
vladsadullaiev04@gmail.com*

ІНФОРМАЦІЙНА СИСТЕМА СТОМАТОЛОГІЧНОЇ КЛІНІКИ

На сьогоднішній день перед сучасними зубними клініками постають важливі завдання – організація обслуговування пацієнтів, керування інформаційною базою клієнтів та оптимізація внутрішньо-клінічних процесів. Збільшується попит на якісну стоматологічну допомогу та потреба у забезпеченні пацієнтам доступу до їхніх медичних даних. Значимість теми розробки інформаційної системи (ІС) для зубних клінік обумовлена потребою в цифровому управлінні клінікою та забезпеченні зручного доступу до послуг для клієнтів.

В Україні функціонує система щорічного планового огляду мешканців України. У 2019 році лікарями-стоматологами у державних закладах охорони здоров'я стоматологічного профілю планово оглянуто всього 7 872 765 осіб, що становить 18,8 % до відповідної кількості населення України.

Для глибшого розуміння функціональних можливостей сучасних ІС, які використовуються в стоматологічних клініках, було проведено порівняльний аналіз трьох популярних медичних закладів: **Ddent**, **ПРАЙМ** та **Люмі-Дент**. Аналіз проводився за десятьма основними критеріями, що охоплюють технічні, функціональні та сервісні характеристики.

У процесі аналізу були розглянуті три клініки: **Ddent**, **ПРАЙМ** та **Люмі-Дент**, кожна з яких використовує власну інформаційну систему для обслуговування пацієнтів. Усі системи підтримують базовий набір функцій, зокрема безпеку даних, доступ до медичних карток, онлайн-запис, перегляд історії візитів та інтеграцію з мобільними пристроями. Система **Ddent** має цілодобову доступність і гнучкий інтерфейс для завантаження медичних файлів. **Люмі-Дент** використовує сучасні технології, такі як цифрова томографія та лазерна стоматологія. **ПРАЙМ** зосереджений на якості послуг і досвіді лікарів, але не підтримує завантаження файлів через інтерфейс.

Таким чином, незважаючи на схожий функціонал, існуючі рішення мають певні обмеження. Це підтверджує доцільність розробки власної інформаційної системи, що поєднуватиме ключові переваги сучасних платформ із додатковими можливостями, адаптованими до конкретних потреб стоматологічної клініки.

Моя інформаційна система для стоматологічної клініки створюється з використанням процесного підходу до проектування. Цей підхід фокусується на аналізі, моделюванні та вдосконаленні бізнес-процесів, що сприяє ефективній роботі системи. Він базується на розподілі діяльності на взаємопов'язані процеси, кожен з яких має чітку мету, необхідні ресурси, виконавців і результати. Основна ідея підходу полягає в уявленні всіх операцій організації як системи процесів, спрямованих на досягнення цілей. У стоматологічній клініці це включає всі етапи, від запису на прийом до лікування й оплати, організовані у вигляді структурованих процесів. Завдяки цьому підходу можна:

1. Організувати діяльність у структуровану систему;
2. Оптимізувати операції;
3. Автоматизувати рутинні завдання;
4. Підвищити якість обслуговування.

На рисунку 1 зображена контекстна діаграма, яка містить загальний опис інформаційної системи стоматологічної клініки і її взаємодію з зовнішнім середовищем. Вона поділяється на фрагменти: авторизації, форми запису, заповнення, відправки запису та оплати послуг. Всі ці фрагменти зображені на діаграмах декомпозиції. Кожна з них описує подробиці роботи системи.

Вхідна інформація в ІС зубної клініки призначена для організації реєстрації, записів на прийом, оплат, формування звітів і нагадувань. Вона надходить як дані від клієнтів, адміністраторів й ветеринарів через інтерфейс системи та інших джерел у вигляді файлів. До вхідної інформації відноситься: Дані про нового клієнта, Запис на прийом, Дані про оплату послуг, Результати обслуговування, Дані для підтвердження запису, Відомості про здійснені оплати, Реєстр клієнтів, Розклад прийомів, Дані про виконані операції, Шаблони повідомлень, Оновлений перелік послуг.

Вихідна інформація створюється системою для виконання таких завдань, як сповіщення клієнтів, адміністраторів і лікарів, а також фіксування результатів роботи. Вона використовується для інформування, підготовки звітів і організації управління записами та послугами клініки. До вихідної інформації відноситься: Нагадування про прийом, Підтвердження запису, Квитанція про оплату, Щоденний розклад прийомів, Звіт про виконанні прийоми, Оновлення бази даних.

Інформаційна система зубної клініки має чітко структуровану систему інформаційного забезпечення, що складається з кількох ключових елементів. Основні компоненти системи поділяються на методичні та інструментальні матеріали, систему класифікації та кодування, а також інформаційну базу, що в свою чергу розділяється на внутрішню та зовнішню частини.

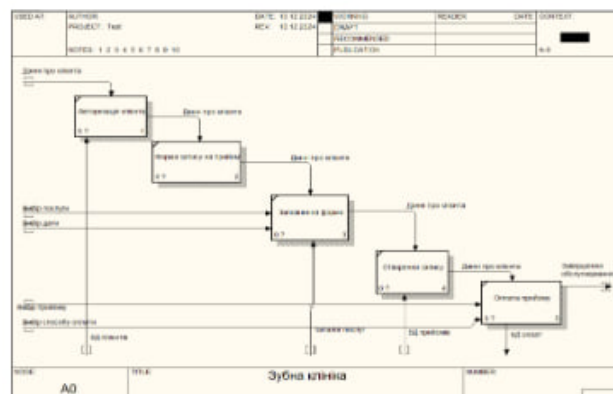


Рисунок 1. Діаграма декомпозиції – Обслуговування в зубній клініці.

50

Рис. А-4. Виклад тез (друга частина тексту)

Список використаних джерел

1. Аналіз основних показників стоматологічної допомоги в Україні за 2020 рік. librarynmu.com. URL: <http://ir.librarynmu.com/bitstream/123456789/3442/1/Аналіз%20основних%20показників%20стоматологічної.pdf>
2. Що таке BPMN-діаграма і навіщо вона потрібна. IAMPM. URL: <https://iampm.club/ua/blog/shho-take-bpmn-diagrama-i-navishho-vona-potribna-z-prikladami-2/>
3. Прогноз ринку стоматологічних послуг. biogentech. URL: <https://biogentech.com.ua/blog/post/prognoz-rinku-stomatologicheskikh-poslug-2022-2032->

Рис. А-5. Виклад тез (Список джерел)

Лістинг програмного коду реалізації RESTful API

```
const express = require('express');
const router = express.Router();
const Register = require('../schematics/register');
const Price = require('../schematics/price');

router.post('/register', async (req, res) => {
  const { full_name, phone_number, date, time, type_service, comment } =
    req.body;
  if (!full_name || !phone_number || !date || !time || !type_service) {
    return res.status(400).json({ error: 'All fields are required' });
  }
  try {
    const priceDoc = await Price.findOne({ name: type_service });
    if (!priceDoc) {
      return res.status(400).json({ error: 'Service not found' });
    }
    const newRegistration = new Register({
      full_name,
      phone_number,
      date,
      time,
      type_service: priceDoc._id,
      comment
    });
    await newRegistration.save();
    res.status(201).json({ message: 'User registered successfully' });
  } catch (error) {
    console.error('Error registering user:', error);
    res.status(500).json({ error: 'Internal server error' });
  }
});

router.get('/price_list', async (req, res) => {
  try {
    const prices = await Price.find();
    res.json(prices);
  } catch (error) {
    console.error('Error fetching price list:', error);
    res.status(500).json({ error: 'Internal server error' });
  }
});

router.post('/price_list', async (req, res) => {
  const { name, price } = req.body;
  if (!name || !price) {
    return res.status(400).json({ error: 'All fields are required' });
  }
  try {
    const newPrice = new Price({ name, price });
    await newPrice.save();
  }
});
```

```

    res.status(201).json({ message: 'Price added successfully' });
  } catch (error) {
    console.error('Error adding price:', error);
    res.status(500).json({ error: 'Internal server error' });
  }
});

router.get('/price_list/:id', async (req, res) => {
  try {
    const price = await Price.findById(req.params.id);
    if (!price) {
      return res.status(404).json({ error: 'Service not found' });
    }
    res.json(price);
  } catch (error) {
    res.status(500).json({ error: 'Internal server error' });
  }
});

router.put('/price_list/:id', async (req, res) => {
  try {
    const { name, price } = req.body;
    const updated = await Price.findByIdAndUpdate(req.params.id, { name,
price }, { new: true });
    if (!updated) {
      return res.status(404).json({ error: 'Service not found' });
    }
    res.json(updated);
  } catch (error) {
    console.error('Error updating price:', error);
    res.status(500).json({ error: 'Internal server error' });
  }
});

module.exports = router;

```

StrikePlagiarism.com

StrikePlagiarism.com

Дата звіту

6/13/2025

Звіт не був оцінений

Дата редагування

Звіт подібності

метадані

Назва організації

Kyiv National Economic University named after Vadym Hetman KNEU

Заголовок

РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ СТОМАТОЛОГІЧНОЇ КЛІНІКИ

Автор

Науковий керівник / Експерт

Садулласє Владислав ЕдуардовичМиронцов Микита Леонідович

підрозділ

кафедра інформаційних систем в економіці

Обсяг знайдених подібностей

4.02%

4.02%

КП 1

25

Довжина фрази для коефіцієнта подібності 2

0.98%

0.98%

КП 2

8075

Кількість слів

3.08%

3.08%

КЦ

63820

Кількість символів

Тривога

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

Заміна букв		0
Інтервали		0
Мікропробіли		6
Білі знаки		0
Парафрази (SmartMarks)		22

Подібності за списком джерел

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

10 найдовших фраз

Копір тексту

ПОРЯДКОВИЙ НОМЕР	НАЗВА ТА АДРЕСА ДЖЕРЕЛА URL (НАЗВА БАЗИ)	КІЛЬКІСТЬ ІДЕНТИЧНИХ СЛІВ (ФРАГМЕНТІВ)	
1	https://ir.kneu.edu.ua/bitstreams/a5c7419f-c089-4b0c-8e70-3923cf3566a2/download	48	0.59 %
2	https://ir.kneu.edu.ua/bitstreams/b20c497c-320d-46ce-8d1d-a56f3c2eec3f/download	31	0.38 %
3	https://ir.kneu.edu.ua/bitstreams/066dbca0-6afa-4cf4-a181-c3e88a2aab6f/download	23	0.28 %
4	https://ir.kneu.edu.ua/server/api/core/bitstreams/8886d65e-0996-456f-8b04-70b8d9683e0e/content	21	0.26 %
5	https://ir.kneu.edu.ua/bitstreams/066dbca0-6afa-4cf4-a181-c3e88a2aab6f/download	20	0.25 %