

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту
(повна назва)

Кафедра Інформатики
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)

ДОСЛІДЖЕННЯ ТА РОЗРОБКА МЕДИЧНОЇ ІНФОРМАЦІЙНОЇ
СИСТЕМИ ДЛЯ ПІДТРИМКИ ПАЛІАТИВНИХ ПАЦІЄНТІВ
(тема)

Виконав:
студент 2 курсу, групи ІНФМ-20-1

Висоцький Д.О.
(прізвище, ініціали)

Спеціальності 122 Комп'ютерні науки
(код і повна назва спеціальності)

Тип програми освітньо-професійна

Освітня програма Інформатика
(повна назва освітньої програми)

Керівник доц. Тітова О.В.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____
(підпис)

Кобилін О.А.
(прізвище, ініціали)

2021 р.

Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту
(повна назва)Кафедра Інформатики
(повна назва)Рівень вищої освіти другий (магістерський)Спеціальність 122 Комп'ютерні науки
(код і повна назва)Тип програми освітньо-професійнаОсвітня програма Інформатика
(повна назва освітньої програми)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

«___» _____ 2021 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУстудентові Висоцькому Данилу Олеговичу
(прізвище, ім'я, по батькові)1. Тема роботи: Дослідження та розробка медичної інформаційної системи для підтримки паліативних пацієнтівзатверджена наказом по університету від «22» жовтня 2021 року №1574Ст.2. Термін подання студентом роботи до екзаменаційної комісії 26 листопада 2021 р.3. Вихідні дані до роботи реалізація компоненту інформаційної підсистеми підтримки паліативних пацієнтів, мова програмування Java.

4. Перелік питань, що потрібно опрацювати в роботі _____

1. Аналіз предметної області (Інформаційні системи: характеристика, основні поняття. Медичні інформаційні системи. Класифікація медичних інформаційних систем. Паліативна допомога).2. Моделювання підсистеми підтримки паліативних пацієнтів.3. Програмна реалізація компоненту медичної інформаційної підсистеми підтримки паліативних пацієнтів.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) Дослідження невіршених проблем паліативних пацієнтів у медичній інформаційній системі. постановка задачі, діаграми UML, опис моделі, опис застосунку.

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Консультант з дотримання діючих стандартів та норм	Доцент Белова Н.В.		

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	22.10.2021	
2	Аналіз завдання, підбір літератури	23.10.21-25.10.21	
3	Аналіз літератури з досліджуваної проблеми	25.10.21-31.10.21	
4	Аналіз технічних засобів	01.11.21-02.11.21	
5	Розробка методу	03.11.21-10.11.21	
6	Програмна реалізація	10.11.21-23.11.21	
7	Оформлення пояснювальної записки	23.11.21-02.12.21	
8	Перевірка на плагіат	03.12.2021	
9	Рецензування	04.12.2021	
10	Підготовка презентації та доповіді	05.12.2021	
11	Занесення роботи в електронний архів	06.12.2021	
12	Попередній захист кваліфікаційної роботи	06.12.2021	

Дата видачі завдання 22 жовтня 2021 р.

Студент _____
(підпис)

Керівник роботи _____ доц. Тітова О.В.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ/ABSTRACT

Пояснювальна записка до атестаційної роботи: 62 с., 1 табл., 21 рис., 33 джерела.

ІНФОРМАЦІЙНА СИСТЕМА, АВТОМАТИЗАЦІЯ, ПРОЦЕС, МОДЕЛЮВАННЯ, ПРОГРАМУВАННЯ, JAVA, СЕРЕДОВИЩЕ, ПРОГРАМНА РЕАЛІЗАЦІЯ, ФУНКЦІОНАЛ, ПАЛІАТИВНІ ПАЦІЄНТИ, БАЗА ДАНИХ, АВТОМАТИЗОВАНЕ ТЕСТУВАННЯ.

Спроектовано концепцію медичної інформаційної підсистеми для вирішення питань підтримки паліативних пацієнтів. Для розкриття суті інформаційної моделі використано UML-діаграми. Розроблено діаграму IDEF0, діаграму варіантів використання, діаграму класів, діаграму діяльності та діаграму послідовності, які дозволяють виконати програмну реалізацію інформаційної підсистеми. Розроблено інтерфейс користування додатком для паліативних пацієнтів. Програмна модель додатку реалізована на мові програмування. Для написання серверної частини компоненту медичної інформаційної системи було обрано монолітну архітектуру. Клієнтна частина програми реалізована за допомогою фреймворку React Native.

У результаті роботи розроблений компонент медичної інформаційної підсистеми підтримки паліативних пацієнтів.

INFORMATION SYSTEM, AUTOMATION, PROCESS, MODELING, PROGRAMMING, JAVA, ENVIRONMENT, SOFTWARE IMPLEMENTATION, FUNCTIONAL, PALLIATIVE PATIENTS, DATA BASES, AUTOMATION TESTING.

The concept of the medical information subsystem for the decision of support of palliative patients is designed. UML diagrams were used to reveal the essence of the information model. IDEF0 diagram, usage diagram, class diagram, activity diagram and sequence diagram have been developed, which allow to perform software implementation of the information subsystem. The application use interface for palliative patients has been developed. The software model of the application is implemented in the programming language. A monolithic architecture was chosen to write the server part of the medical information system component. The client part of the program is implemented using the React Native framework.

As a result of work the component of a medical information subsystem of support of palliative patients is developed.

ЗМІСТ

Перелік умовних позначень, символів, одиниць, скорочень і термінів	7
Вступ	8
1 Аналіз предметної області	9
1.1 Основні поняття ІС	9
1.2 Медичні інформаційні системи	13
1.3 Приклади медичних інформаційних систем	13
1.3.1 Electronic Medical Record та Electronic Health Record	13
1.3.2 Medical Practice Management Software	15
1.3.3 Master Patient Index	17
1.3.4 Patient Portals	19
1.3.5 Remote Patient Monitoring	20
1.4 Паліативна допомога	22
1.4.1 Важливість надання паліативної медичної допомоги	22
1.4.2 Проблеми надання паліативної медичної допомоги	24
1.4.3 Кроки для покращень у паліативній допомозі	25
1.5 Постановка задачі дослідження	27
2 Моделювання підсистеми підтримки паліативних пацієнтів	28
2.1 Існуючі методи підтримки паліативних пацієнтів	28
2.2 Розробка моделі інформаційної системи підтримки паліативних пацієнтів	31
2.2.1 Концепція інформаційної моделі підтримки паліативних пацієнтів	31
2.2.2 Діаграми діяльності та послідовності моделі інформаційної системи	42
3 Програмна реалізація компоненту медичної інформаційної підсистеми підтримки паліативних пацієнтів	46
3.1 Використані технології	46

	6
3.2 Архітектура серверної частини	48
3.3 Реалізація клієнтської частини	50
3.4 Опис бази даних	51
3.5 Інтеграція зі сторонніми МІС	52
3.6 Інструкція користувача	54
3.7 Автоматизоване тестування компоненту	55
Висновки.....	58
Перелік джерел посилання	59

**ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ,
СКОРОЧЕНЬ І ТЕРМІНІВ**

ІС – інформаційна система

МІС – медичних інформаційних систем

EHR – Electronic Health Record

EMR – Electronic Medical Record

MPMS – Medical Practice Management Software

MPI – Master Patient Index

RPM – Remote Patient Monitoring

ВСТУП

Стратегія розвитку паліативної допомоги в Україні на період до 2027 року, винесена на обговорення громадськості, визначає, що забезпечення реалізації прав і свобод людини і громадянина, передбачених Конституцією та Законами України, впровадження в Україні європейських стандартів життя вимагає визнати права важкохворих пацієнтів та створити умови для досягнення максимальної якості життя для паліативних пацієнтів усіх вікових категорій [1].

На сьогодні в Україні створено 2 центри, 7 хоспісів та понад 60 самостійних відділень паліативної допомоги, що забезпечує доступ до 1500 стаціонарних паліативних ліжок, при рекомендованій Всесвітньою організацією охорони здоров'я потребі у 3500 ліжок.

Автоматизація медичного інформаційного процесу у сучасному світі передбачає створення інформаційних систем, які спрямовані на інформаційну підтримку або покращення робочого процесу лікарів. Однак, на сьогодні існує дуже мало медичних інформаційних систем для підтримки медичних питань хворих.

Метою даної кваліфікаційної роботи є дослідження та розробка компонентів МІС для підвищення ефективності надання медичних послуг паліативним хворим .

Об'єктом дослідження є автоматизація процесу надання медичних послуг.

Предмет розробки – підсистема інформаційної підтримки паліативних пацієнтів.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Основні поняття ІС

Інформаційна система являє собою сукупність багатьох наборів даних, які забезпечують успішне виконання бізнес-цілі. Інформаційні системи не є окремою моделлю ІТ-індустрії. Натомість важливим аспектом їх успішної реалізації є їх інтеграція з даними та бізнес-процесами. Тому інформаційну систему краще розглядати як трикутник. Три частини цього трикутника представляють процеси, людей і комп'ютери. Інформаційна система вимагає, щоб кожен з цих компонентів працював належним чином, щоб досягти успіху. Серед основних понять під час розробки інформаційних систем можна виділити:

- дані;
- інформація;
- модель;
- об'єкт;
- інформаційний процес;
- інформаційна система.

Даними називають формалізовану інформацію, придатну для подальшої обробки, зберігання та передачі засобами автоматизації професійної діяльності.

Дані – це зареєстровані сигнали, тобто складова частина інформації.

Під інформацією розуміють будь-які відомості про процеси та явища, які в тій чи іншій формі передаються між об'єктами матеріального світу. Інформація – це продукт взаємодії даних та методів, розглянутий у контексті цієї взаємодії.

Під моделлю предметної області, явища чи процесу зазвичай розуміють спрощене, формалізоване уявлення, зручне для комп'ютерної обробки. У сучасному підході до визначення цього поняття розрізняють функціональну

модель та модель даних. Функціональна модель описує процес перетворення інформації та зводиться, по-перше, до традиційних понять моделювання та програмування систем та алгоритмів (екранна форма введення даних, екранна форма результатної інформації, математичний або алгоритмічний зв'язок між входом та виходом), по-друге, до концепту граф-схеми, що в сучасному сенсі реалізується за допомогою тих чи інших засобів графічного опису бізнес-процесів (BPWin, UML і т.п.). Модель даних відображає процес типізації даних та способи розміщення їх у комп'ютерній системі. Уявлення про дані як складової частини програмного модуля («інкапсуляція») притаманно об'єктно-орієнтованого програмування. Опис та реалізація концепту бази даних також може бути здійснена графічно, наприклад, за допомогою засобу ERWin.

З погляду моделі даних, на основі якої будується інформаційна система, прийнято виділяти три рівня моделювання: концептуальна модель, логічна модель, фізична модель.

Концептуальна модель даних складається із трьох позицій:

- тип (структурний еквівалент) даних; наприклад, текстовий тип, числовий, гіперпосилальний і т.п.;
- операції, які можна застосовувати до даних; наприклад, операції додавання не ідентичні для текстових та числових даних;
- обмеження цілісності, під якими слід розуміти обмеження на набір чи безліч допустимих значень даних.

Логічна модель даних – це спосіб представлення бази даних інформаційної системі. До подібних моделей належать:

- ієрархічна модель, опис її являє собою дерево або оргграф, у кожному вершину якого, крім першої, входить тільки одна дуга, а з будь-якої вершини (крім останньої, кінцевої) може довільною виходити число дуг. В ієрархічній структурі підлеглий елемент завжди пов'язаний лише з одним вихідним елементом;

– мережева модель заснована на ідеї впливу різних об'єктів на той самий об'єкт; всередині такої моделі в кожен вершину може входити кілька дуг, що походять від інших вершин. Ці моделі ближчі до реальності, проте дуже складні для опису та аналізу;

– реляційна модель, у її основі засноване уявлення об'єктів як двомірних таблиць, пов'язаних друг з одним посередництвом ключів чи відносин. Для реляційних моделей застосовний точний математичний опис, тобто реляційна алгебра, саме тому вони лягли в основу сучасних засобів проектування баз даних.

Фізична модель даних – це спосіб розміщення даних у пам'яті комп'ютера, що залежить від апаратної частини комп'ютера та від його операційної системи.

Об'єкт лежить в основі об'єктно-орієнтованого програмування та побудови сучасних реляційних баз даних (БД). Під об'єктом також розуміють фрагмент комп'ютерної програми. Під предметною областю зазвичай розуміють первинну інформацію про об'єкти, процеси та явища навколишнього світу, яка може бути основою для побудови інформаційної системи. У цьому контексті реальні об'єкти у складі предметної області вимагають уточнення характеристик (типізація «властивостей» об'єкта) та інформаційних зв'язків (виявлення типових «зв'язків» між об'єктами). Лише після типізації («моделювання») як самих об'єктів, так і їх інформаційних зв'язків у складі предметної галузі можна виділити абстрактні сутності та перейти до проектування БД або функціональної бізнес-системи.

У процесі типізації реальних об'єктів з'являється нове поняття – клас. Кажуть, що об'єкти належать до одного класу, якщо їх властивості обумовлені одним і тим же набором типових характеристик, а методи поведінки і події, що відбуваються з ними, типізовані. У такому разі кожен конкретний об'єкт, виділений з одного класу об'єктів, буде екземпляром класу. Наприклад, під час проектування реляційних БД виділяють класи

об'єктів типу Таблиця, Запит, Форма, Звіт тощо. Таблиця «Клієнти», заповнена конкретною інформацією, буде екземпляром класу таблиць.

Сукупність усіх інформаційних процедур утворює інформаційний процес – технологічну основу управлінської діяльності [2]. Розрізняють інформаційний процес як такий, що описує технологічні етапи роботи з інформацією (збір, зберігання, переробка, відображення та передача інформації), та інформаційний потік, що описує зв'язки між об'єктами предметної області. Наприклад, інформація, що надходить із бухгалтерії до аналітичного відділу, буде у складі граф-схеми підприємства визначена як інформаційний потік, а самі етапи збору, зберігання, обробки та передачі облікової інформації – як інформаційний процес.

Інформаційна система є людино-машинною системою, що забезпечує використання комп'ютерних технологій, збір, передачу, обробку та зберігання інформації для управління виробництвом.

Звичайно, якщо інформаційні процеси та потоки повністю формалізуються, така система може працювати і без участі людини. Наприклад, складання автомобіля цілком за допомогою роботів-автоматів цілком досяжна. Проте в економічній галузі це важко реалізувати. Не тільки через складність управлінських процесів, але ще й тому, що економіка в цілому являє собою динамічний процес, що постійно змінюється і розвивається.

Об'єкти («сутності») використовують формалізовані процедури («зв'язку») для досягнення певного стану («події»). Подія відбувається всередині інформаційної системи і сприймається як позитивний результат її діяльності («прагматична цінність» діяльності), безпосередньо пов'язаний з оголошеною наперед метою діяльності, з ефективністю процесу діяльності, з мінімізацією витрат, необхідного досягнення оголошеного результату [3]. Таким чином, основою створення ІС є інформаційна модель (або діаграма «Сутність-Зв'язок» у трактуванні, пов'язаному з CASE-технологією), покладена в основу як опису самої предметної області.

1.2 Медичні інформаційні системи

Медичні інформаційні системи відносяться до систем, які призначені для управління даними охорони здоров'я. Сюди входять системи, які збирають, зберігають, керують та передають медичну карту пацієнта в електронному форматі, оперативне управління лікарні або систему підтримки рішень щодо політики охорони здоров'я.

Медичні інформаційні системи також включають ті системи, які обробляють дані, пов'язані з діяльністю постачальників та організацій охорони здоров'я. Як комплексні зусилля, вони можуть бути використані для покращення результатів пацієнтів, інформування про дослідження та впливу на формування політики та прийняття рішень. Оскільки інформаційні системи охорони здоров'я зазвичай отримують доступ, обробляють або підтримують великі обсяги конфіденційних даних, безпека є першочерговою проблемою.

1.3 Приклади медичних інформаційних систем

Медичні інформаційні системи можуть використовувати кожен у сфері охорони здоров'я, від пацієнтів до клініцистів і службовців охорони здоров'я. Вони збирають дані та збирають їх у спосіб, який можна використовувати для прийняття рішень щодо охорони здоров'я [4].

1.3.1 Electronic Medical Record та Electronic Health Record

Electronic Health Record (EHR) зосереджуються на документуванні та зберіганні медичної інформації пацієнта. У минулому клініцистам доводилося документувати все на папері вручну, а тепер вони можуть просто заповнити цю інформацію на своєму комп'ютері або мобільному пристрої.

Electronic Medical Record (EMR) починалися як спосіб усунути час і помилки, які виникають при ручному складанні діаграм даних пацієнтів. Історично проблема з EMR полягала в тому, що інформацію про пацієнта можна було переглянути лише в одному кабінеті – тому, якщо пацієнта переведуть в іншу клініку, його медична інформація не буде відповідати цьому прикладу. У результаті цих обмежень були створені електронні медичні картки, щоб дозволити обмін даними пацієнтів у різних медичних закладах. За допомогою програмного забезпечення EMR та EHR пацієнта, якого переміщують, скажімо, до відділення невідкладної допомоги, можна належним чином лікувати, оскільки різні лікарі мають доступ до їхньої інформації [5]. Основні принципи на яких основана МІС EMR зображено на рисунку 1.1.

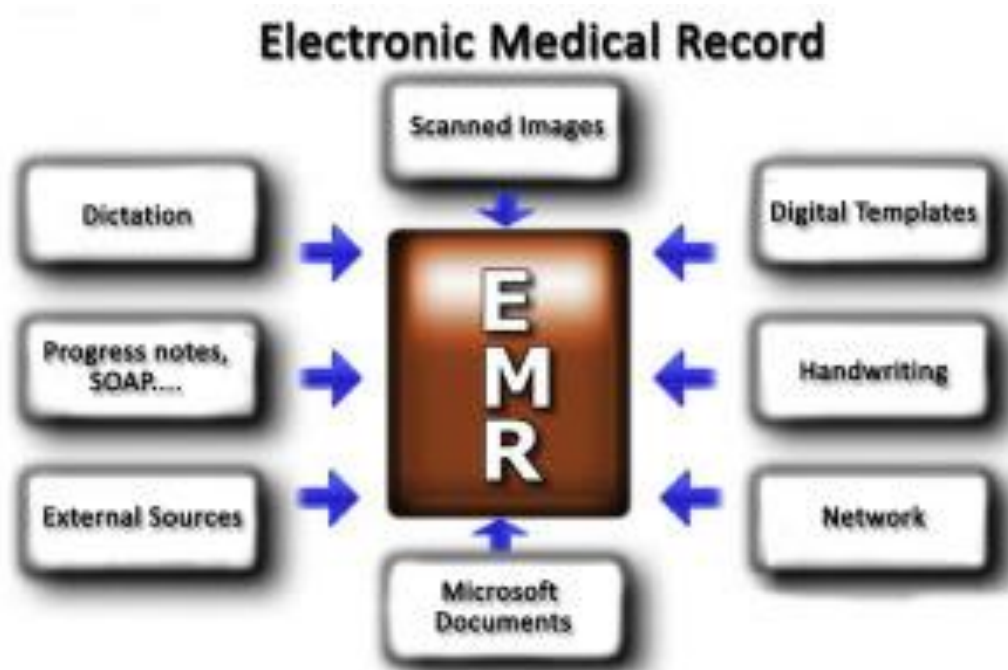


Рисунок 1.1 – Принципи медичної інформаційної системи EMR

Відрізнити EHR від EMR може бути важко, останніми роками інсайдери галузі почали використовувати ці два терміни як взаємозамінні.

Ці системи також можуть попередити, коли пацієнти мають пройти профілактичні процедури та огляди. Крім того, EHR допомагають лікарям

лікувати пацієнтів, дивлячись на їх історію та порівнюючи дані про стан здоров'я з попередніми записами.

EHR також економлять гроші організацій охорони здоров'я, економлячи місце. Оскільки інформація про пацієнта зберігається в електронному вигляді, приміщення можуть змінити місце, яке спочатку використовувалося для зберігання документів. Вони також усувають ризик пошкодження або загублених файлів.

Важливо пам'ятати, що програмне забезпечення для керування практикою та електронні медичні записи зосереджені на абсолютно різних аспектах вашої практики.

1.3.2 Medical Practice Management Software

Medical Practice Management Software (MPMS) розроблено, щоб допомогти медичним закладам працювати більш ефективно. Це технологічне рішення, яке можна використовувати для планування зустрічей, відстеження часу прийому, керування циклами виставлення рахунків та організації інформації про пацієнтів [6].

MPMS є важливою складовою будь-якого сучасного медичного закладу, оскільки воно дозволяє організаціям охорони здоров'я залишатися організованими та забезпечувати високий рівень обслуговування пацієнтів. Об'єднавши всю інформацію, пов'язану з пацієнтами та лікарями, в єдину платформу, медичні працівники можуть легко орієнтуватися на важливі деталі про пацієнтів та іншу інформацію. Без цього програмного забезпечення організація будь-якої медичної практики була б трудомісткою і підлягала б значним людським помилкам. На рисунку 1.2 наведено приклад інтерфейсу MICS MPMS.

Інтуїтивно зрозуміле та ефективне MPMS має важливе значення для робочого процесу практики. До особливостей MPMS відносять: відстеження інформації про пацієнтів, централізоване виставлення рахунків та звіти.

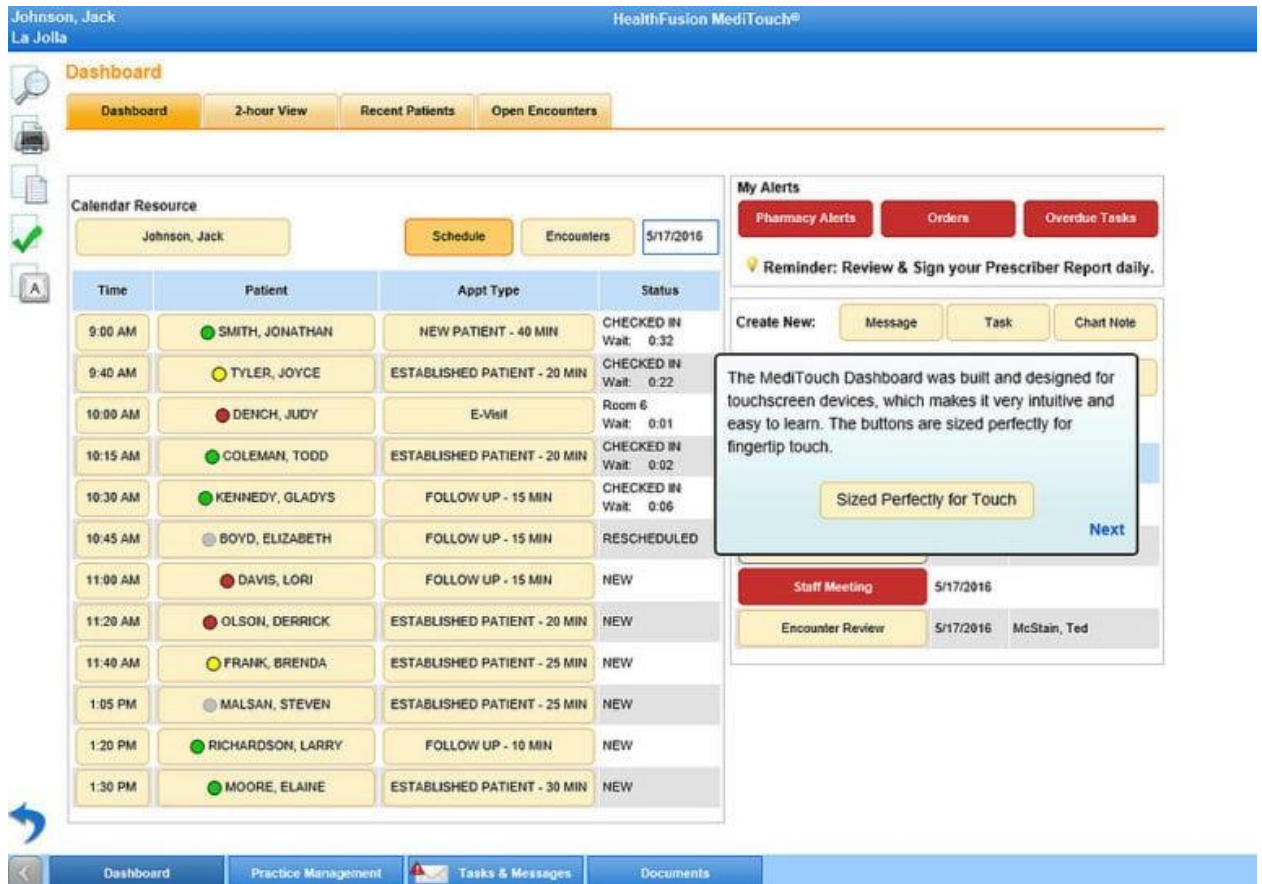


Рисунок 1.2 – Інтерфейс медичної інформаційної системи MPMS

Відстеження інформації про пацієнтів є найосновнішою і, можливо, найважливішою особливістю MPMS є можливість підтримувати актуальну інформацію про пацієнтів, включаючи історію хвороби, сімейний анамнез, ліки, алергії та останні лабораторні результати.

Можна використовувати MPMS, щоб перевірити статус страхування пацієнта, підтвердити, чи він все ще активний, і оцінити доплати, власні витрати та попередні авторизації.

Більшість MPMS включає інструмент календаря, який дозволяє персоналу переднього офісу планувати пацієнтів за місцем розташування та

постачальником, що дозволяє швидко та легко переглядати інформацію про пацієнтів перед відвідуванням.

Централізоване виставлення рахунків, належне використання MPMS дозволить обробляти платежі та записувати історію платежів на одній платформі з вбудованими можливостями обробки кредитних карток. Співробітники фронт-офісу можуть фіксувати власні витрати наперед, коли пацієнт приходить на прийом, а співробітники бек-офісу можуть кодувати претензії про відшкодування та подавати їх платникам (наприклад, страховим компаніям), керуючи тими, яким було відмовлено або відхилено.

Звіти, клініцистам та іншим медичним працівникам потрібні дані пацієнтів, щоб бути ефективними. MPMS дозволяє створювати важливі звіти про пацієнтів для інших лікарів. Це може покращити догляд за пацієнтами та гарантувати, що ваша практика надає пацієнтам і лікарям точну корисну інформацію [7].

Аналогічно, MPMS повинно мати можливість генерувати фінансові звіти, які допоможуть відстежувати прибутковість вашого бізнесу. Ці звіти містять таку інформацію, як дебіторська заборгованість, дні претензій, проведені в дебіторській заборгованості, і загальна сума відшкодувань порівняно з очікуваними відшкодуваннями.

1.3.3 Master Patient Index

Master Patient Index (MPI) – також відомий як головний індекс пацієнтів, реєстр пацієнтів або реєстр клієнтів є електронною базою даних, яка містить демографічну інформацію про кожного пацієнта, який отримує медичні послуги.

MPI має на меті унікальну ідентифікацію осіб шляхом зберігання такої інформації, як ім'я, дата народження, стать тощо, і присвоєння кожній людині унікального ідентифікатора.

МРІ важливий для надання медичним працівникам усієї інформації, що стосується конкретного пацієнта. Якщо клієнт отримав допомогу в кількох різних установах – кожен із власною системою реєстрації та ведення пацієнтів – МРІ дозволяє різним системам обмінюватися інформацією та пов'язувати всі записи, пов'язані з одним пацієнтом [8].

Схему роботи МІС МРІ, коли клієнт отримав допомогу в різних лікарнях зображено на рисунку 1.3.



Рисунок 1.3 – Схему роботи МІС МРІ

Таким чином, медичний працівник має доступ до всієї історії хвороби та лікування пацієнта, що полегшує забезпечення безперервного лікування. МРІ:

- веде центральний реєстр усіх пацієнтів та їх демографічні дані, призначаючи унікальний ідентифікатор кожному пацієнту, який потім можна використовувати для зв'язку з електронною картою здоров'я на рівні підприємства;

- виключає повторювані записи реєстрації пацієнтів, які виникають через зміни в демографічних показниках пацієнта (наприклад, пацієнт перемістився в інше місце), помилки введення даних під час реєстрації пацієнта або відсутня демографічна інформація;

– дозволяє медичним працівникам визначити, які медичні заклади відвідав пацієнт, щоб отримати допомогу.

Якщо МРІ відсутній, існує ризик того, що медичні працівники будуть лікувати пацієнта, не маючи всієї інформації, необхідної для встановлення ефективного діагнозу та надання пацієнту належного догляду.

1.3.4 Patient Portals

Patient Portal або портал для пацієнтів – це безпечний онлайн-сайт, який надає пацієнтам зручний цілодобовий доступ до особистої медичної інформації з будь-якого місця, де є підключення до Інтернету [9]. Використовуючи безпечне ім'я користувача та пароль, пацієнти можуть переглядати таку інформацію про здоров'я, як:

- недавні відвідування лікаря;
- виписні підсумки;
- ліки;
- щеплення;
- алергії;
- результати лабораторних досліджень.

Деякі портали пацієнтів також дозволяють:

- надсилати безпечне повідомлення своєму лікарю;
- робити запит на оновлення рецепта;
- призначати нетермінові зустрічі;
- оновлювати контактну інформацію;
- здійснювати платежі;
- завантажувати та заповнювати форми;
- переглядувати навчальні матеріали.

Завдяки порталу для пацієнтів ви можете контролювати своє здоров'я та догляд. Портали для пацієнтів також можуть заощадити ваш час,

допомогти спілкуватися зі своїм лікарем та підтримати лікування між відвідуваннями.

1.3.5 Remote Patient Monitoring

Remote Patient Monitoring (RPM) – це підкатегорія телемедицини в домашніх умовах, яка дозволяє пацієнтам використовувати мобільні медичні пристрої та технології для збору даних про стан здоров'я пацієнтів і надсилання їх медичним працівникам. Загальні фізіологічні дані, які можна зібрати за допомогою програм RPM, включають життєво важливі показники, вагу, кров'яний тиск і частоту серцевих скорочень. Після збору дані пацієнта надсилаються до кабінету лікаря за допомогою спеціальної комп'ютерної системи телемедицини або програмного застосунку, який можна встановити на комп'ютер, смартфон або планшет [10]. Основні компоненти MIC RPM зображено на рисунку 1.4.

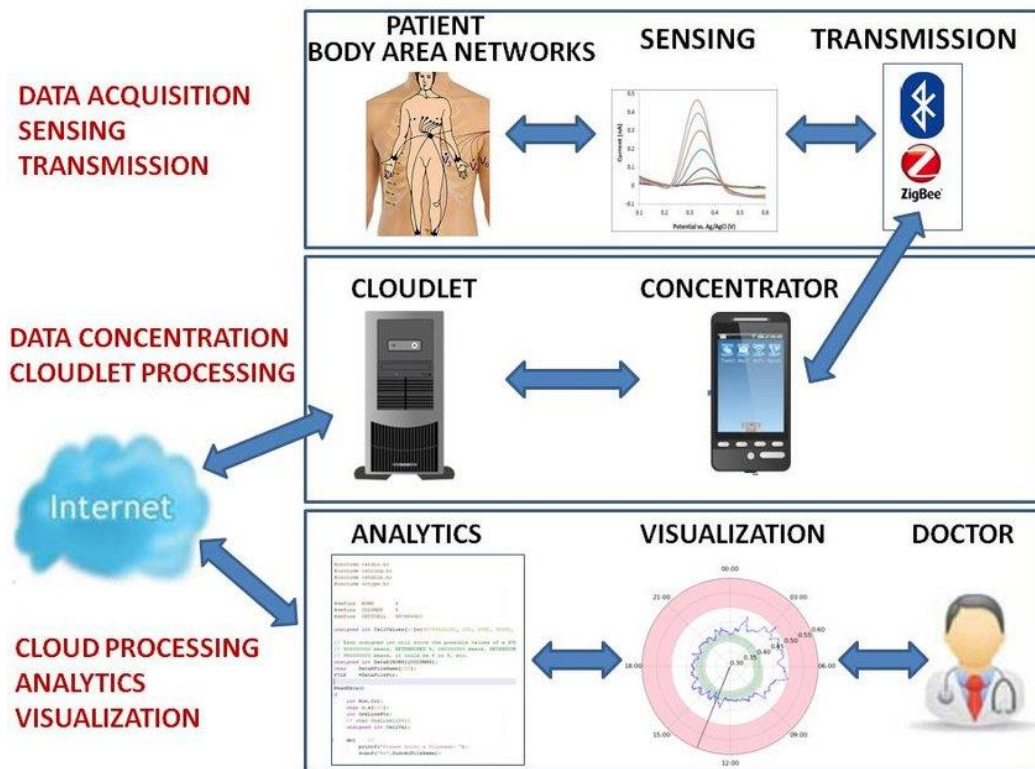


Рисунок 1.4 – Основні компоненти MIC RPM

RPM часто використовується для допомоги пацієнтам, які потребують хронічного догляду, догляду після виписки або догляду похилого віку. Підключаючи пацієнтів із високим ризиком до віддаленого моніторингу, він може сповіщати організації охорони здоров'я про потенційні проблеми зі здоров'ям або відстежувати дані пацієнтів між відвідуваннями. Крім того, RPM може використовуватися підприємствами, які хочуть реєструвати випадки компенсації працівникам, щоб переконатися, що працівники на правильному шляху, щоб повернутися до роботи.

Методи RPM можуть відрізнятися залежно від пристрою, який використовується, або стану, що контролюється, більшість технологій включає подібні компоненти. Перший – це бездротовий датчик, який може вимірювати конкретні фізіологічні параметри та зберігати дані, які він збирає. Це сховище також має включати спосіб підключення до додаткових датчиків, баз даних медичних працівників та пов'язаних програм. Зазвичай програми надають користувачам інтерфейс для відстеження або аналізу даних і відображення рекомендацій щодо лікування.

Дані, зібрані пристроями RPM, надсилаються в потрібне місце та зберігаються в реляційній базі даних. Це дозволяє організаціям охорони здоров'я, які мають бездротові телекомунікаційні дані, розглядатися як окремі випадки або в контексті всієї історії здоров'я. Часто пристрій може сповіщати пацієнтів, коли медичний працівник переглянув дані або виявив проблему, яка вимагає від пацієнта з'явитися.

Переваги Remote Patient Monitoring:

- підвищена залучення пацієнтів, пристрої RPM дозволяють пацієнтам відігравати вирішальну роль у керуванні та розумінні стану свого здоров'я;
- покращена якість медичної допомоги, RPM надає пацієнтам та медичним працівникам доступ до більш релевантних даних про пацієнтів, покращуючи таким чином загальну якість та ціннісне обслуговування;

– кращий доступ до медичної допомоги. Оскільки RPM дозволяє пацієнтам самостійно проходити базове обстеження здоров'я, медичним працівникам дозволено лікувати більше пацієнтів;

– вищий рівень освіти та підтримки, RPM щодня надає пацієнтам інформацію та відгуки про їхні особисті умови, навчаючи їх та надаючи підтримку;

– впевненість пацієнтів. Постійний моніторинг може дати пацієнтам спокій, що будь-які потенційні проблеми будуть виявлені вчасно.

1.4 Паліативна допомога

Паліативна допомога – це спеціалізована медична допомога людям, які живуть з тяжким захворюванням. Цей вид допомоги спрямований на полегшення симптомів і стресу хвороби. Метою паліативної допомоги є покращення якості життя як пацієнта, так і його родини [11].

Паліативну допомогу надає спеціально навчена команда лікарів, медсестр та інших спеціалістів, які працюють разом з іншими лікарями пацієнта, щоб забезпечити додатковий рівень підтримки. Паліативна допомога ґрунтується на потребах пацієнта, а не на прогнозі пацієнта. Це доречно в будь-якому віці і на будь-якій стадії серйозного захворювання, і його можна забезпечити разом із куративним лікуванням [12].

1.4.1 Важливість надання паліативної медичної допомоги

Надання паліативної медичної допомоги – це підхід, спрямований на покращення якості життя пацієнтів (дорослих та дітей) та їхніх сімей, які стикаються з проблемами через небезпечні для життя захворювання. Завдяки ранній діагностиці, правильній оцінці та лікуванню болю, а також вирішенню

інших проблем, чи то фізичних, психосоціальних чи духовних, паліативна допомога дозволяє запобігти та полегшити страждання.

Полегшення страждань вимагає вирішення питань, що виходять за межі усунення фізичних симптомів. При наданні паліативної допомоги використовується комплексний підхід до надання підтримки пацієнтам та особам, що доглядають за ними. Мається на увазі задоволення практичних потреб та консультування у разі втрати близьких. Крім того, пропонується система, орієнтована на сприяння пацієнтам у підтримці якомога вищої активності протягом усього життя [13].

Паліативна медична допомога здобула безумовне визнання в рамках прав людини на здоров'я. Її надання має відбуватися в рамках орієнтованого на людину та комплексного медичного обслуговування з особливою увагою до конкретних потреб та переваг індивідуумів.

Паліативна медична допомога потрібна при багатьох захворюваннях. Більшість дорослих, які її потребують, страждають на такі хронічні захворювання, як серцево-судинні (38,5%), рак (34%), хронічні респіраторні захворювання (10,3%), СНІД (5,7%) та діабет (4,6%). Паліативна медична допомога може знадобитися при багатьох інших станах, у тому числі нирковій недостатності, хронічних захворюваннях печінки, розсіяному склерозі, хворобі Паркінсона, ревматоїдному артриті, неврологічних захворюваннях, деменції, вроджених вадах розвитку та туберкульозі з лікарською стійкістю.

Біль є одним із найчастіших і серйозних симптомів, який відчувають пацієнти, які потребують паліативної медичної допомоги. Наприклад, наприкінці життя біль від помірної до гострої будуть відчувати 80% хворих на СНІД або рак і 67% пацієнтів із серцево-судинним захворюванням або хронічною обструктивною хворобою легень.

Опіоїдні препарати також можуть полегшити інші поширені болючі фізичні симптоми, у тому числі задишку. Боротьба з такими симптомами на

ранньому етапі є частиною етичного обов'язку полегшити страждання та дотриматись людської гідності.

1.4.2 Проблеми надання паліативної медичної допомоги

За оцінками, щорічно паліативної медичної допомоги потребують 40 мільйонів осіб, 78% з яких проживають у країнах з низьким і середнім рівнем доходу. Щодо дітей, 98% тих, хто потребує паліативної медичної допомоги, проживають у країнах з низьким і середнім рівнем доходу, а майже половина з них – в Африці.

Для вирішення проблеми незадоволеної потреби паліативної медичної допомоги в усьому світі необхідно усунути ряд серйозних перешкод:

- найчастіше паліативна медична допомога не реалізується у жодному вигляді в рамках національних стратегій та систем охорони здоров'я;
- навчання працівників охорони здоров'я у галузі паліативної медичної допомоги нерідко проводиться в обмеженому обсязі або не передбачено;
- доступ населення до опіоїдних аналгетиків є недостатнім і не відповідає міжнародним конвенціям про доступ до основних лікарських засобів.

За даними обстеження ВООЗ щодо неінфекційних захворювань, проведеного серед 194 держав-членів у 2019 р., фінансування паліативної допомоги було доступне в 68% країн, а охоплення принаймні половини пацієнтів, які її потребують, згідно інформації наданої державами, вдалося забезпечити лише в 40% країн [14].

Міжнародний комітет з контролю над наркотиками дійшов висновку, що у 2018 р. на частку 79 відсотків світового населення, головним чином людей, які проживають у країнах з низьким та середнім рівнем доходу, припадало лише 13% від загального обсягу морфіну, який застосовується для

знеболювання та полегшення страждань, або 1 відсоток від 388 тонн морфіну, що виробляється у всьому світі. Ці значення перевищують показники, зареєстровані в 2014 р., коли на частку 80 відсотків світового населення припадало всього 9,5 відсотка морфіну, що застосовується для знеболювання та полегшення страждань, проте нерівне становище країн з низьким та середнім рівнем доходу та країн з високим рівнем доходу в частині використання наркотичних засобів для паліативної допомоги, як і раніше, викликає стурбованість.

До інших факторів, що перешкоджають наданню паліативної медичної допомоги, належать:

- відсутність поінформованості серед осіб, які формулюють політику, працівників охорони здоров'я та громадськості про те, що являє собою паліативна медична допомога і яку користь вона може принести пацієнтам та системам охорони здоров'я;

- культурні та соціальні бар'єри, наприклад уявлення про смерть та процес вмирання;

- помилки щодо паліативної медичної допомоги, наприклад, думка про те, що вона призначена тільки для онкологічних хворих або що вона належить в останні тижні життя;

- помилкові уявлення про те, що доступ до опіоїдних анальгетиків сприятиме зростанню випадків зловживання психоактивними речовинами.

1.4.3 Кроки для покращень у паліативній допомозі

На національних системах охорони здоров'я лежить відповідальність за включення паліативної медичної допомоги до континууму допомоги, що надається особам з хронічними та небезпечними для життя розладами здоров'я, а також за інтеграцію її надання з програмами профілактики,

виявлення на ранньому етапі та лікування захворювань. Маються на увазі, як мінімум, такі компоненти:

- заходи політики у галузі охорони здоров'я, спрямовані на облік та фінансування послуг паліативної допомоги в рамках національних систем охорони здоров'я на всіх рівнях медичного обслуговування;

- заходи політики, спрямовані на посилення та збільшення людських ресурсів, у тому числі навчання наявних фахівців охорони здоров'я, включення паліативної медичної допомоги до основної навчальної програми всіх нових працівників охорони здоров'я, а також навчання волонтерів та представників громадськості;

- політика щодо лікарських засобів, що забезпечує доступність основних препаратів для симптоматичного лікування, зокрема опіоїдних анальгетиків, для застосування з метою полегшення болю та проявів дихальної недостатності.

Найбільша ефективність паліативної допомоги досягається за умови своєчасного звернення до неї на відповідних етапах хвороби. Своєчасне надання паліативної допомоги не лише покращує якість життя пацієнтів, а й дозволяє скоротити кількість випадків невинного звернення за госпітальною допомогою та послугами охорони здоров'я.

Паліативну медичну допомогу слід надавати відповідно до принципів загального охоплення медичним обслуговуванням. Все населення, незалежно від рівня доходів, характеру захворювання чи віку, повинне мати доступ до встановленого в національних масштабах набору основних медичних послуг, включаючи паліативну медичну допомогу. Фінансові системи та системи соціального захисту повинні враховувати, що бідні та маргіналізовані групи населення мають право людини на паліативну медичну допомогу [15].

Робота у складі багатопрофільної бригади фахівців вимагає від середнього медичного персоналу, особливо осіб, які працюють з пацієнтами, які страждають на тяжкі захворювання, наявності навичок паліативної допомоги.

Спеціалізована паліативна допомога є компонентом паліативного медичного обслуговування. Проте стійка, якісна та доступна система паліативної допомоги має бути інтегрована в первинне медико-санітарне обслуговування, надання допомоги за місцем проживання та вдома та забезпечувати сприяння особам, які здійснюють догляд, наприклад, членам сімей пацієнтів та громадським волонтерам. Забезпечення паліативної допомоги має розглядатися як етичний обов'язок працівників охорони здоров'я.

1.5 Постановка задачі дослідження

Метою даної кваліфікаційної роботи є дослідження та розробка компонентів МІС для підвищення ефективності надання медичних послуг паліативним хворим .

Об'єктом дослідження є автоматизація процесу надання медичних послуг.

Предмет розробки – підсистема інформаційної підтримки паліативних пацієнтів.

Для досягнення мети необхідно вирішити такі завдання:

- проаналізувати існуючі методи підтримки паліативних пацієнтів;
- розробити модель інформаційної системи підтримки паліативних пацієнтів;
- розробити програмну реалізацію компоненту медичної інформаційної підсистеми.

2 МОДЕЛЮВАННЯ ПІДСИСТЕМИ ПІДТРИМКИ ПАЛІАТИВНИХ ПАЦІЄНТІВ

2.1 Існуючі методи підтримки паліативних пацієнтів

Паліативна допомога – це підхід, що має на меті поліпшення якості життя пацієнта і членів його сім'ї, які опинилися перед захворюванням, що загрожує життю.

Паліативна медична допомога – комплекс заходів, що включають медичні втручання, заходи психологічного характеру та догляд, що здійснюються з метою покращення якості життя невиліковно хворих громадян та спрямовані на полегшення болю, інших тяжких проявів захворювання.

Паліативна допомога складається з двох компонентів:

- полегшення страждань хворого протягом усього періоду хвороби (поряд з радикальним лікуванням);
- медична допомога в останні місяці, дні та години життя.

Завдання паліативної допомоги – забезпечення кращої якості життя пацієнта. Вона не обмежується переліком певних діагнозів. До того ж важливою частиною допомоги є робота з близькими пацієнта під час його хвороби та їхня психологічна підтримка.

Існує три види паліативної допомоги: хоспісна, термінальна та допомога вихідного дня.

Хоспісна допомога, головна мета якої – це всебічна турбота про пацієнта. У хоспісі вирішуються всі проблеми невиліковного хворого, включаючи надання місця для проживання в останні дні життя та усунення нападів болю. Співробітники хоспісу забезпечують всі потреби пацієнта: фізичні, духовні, емоційні та соціальні. Потрапити до хоспісу можна за направленням лікаря. Підставами для цього є невиліковне захворювання у

тяжкій стадії, больовий синдром, який неможливо усунути в домашніх умовах, неможливість догляду з боку членів сім'ї та інше.

Термінальна допомога – це всебічна паліативна допомога пацієнтові в останні дні його життя.

Під допомогою наприкінці життя розуміється як паліативний догляд останні дві чи три доби перед смертю, так і допомогу пацієнтам, які вмирають у неспеціалізованих клініках. У деяких країнах до цього поняття також входить остання медична допомога хворим, які не потребують паліативного догляду.

Допомога вихідного дня надається у виділених для цього днях, щоб родичі, які постійно зайняті доглядом за хворим, могли відпочити.

Цим займаються фахівці у денних стаціонарах, відділеннях при хоспісах чи виїзні служби вдома у хворого.

Паліативна медична допомога поділяється на:

- паліативну первинну медичну допомогу;
- долікарську;
- лікарську;
- паліативну спеціалізовану медичну допомогу.

Паліативна допомога хворим може бути організована трьома способами:

- в домашніх умовах. Паліативна допомога вдома може бути організована тільки якщо в кабінеті протибольової терапії, до якого належить хворий, є власний транспорт. Догляд вдома включає знеболювання, догляд, соціальну, психологічну підтримку. Оскільки хоспісів та відділень для паліативних пацієнтів мало, багато хворих змушені продовжити лікування вдома, де їх доглядають тільки родичі. Тому крім державних організацій додому іноді виїжджають волонтери та приватні хоспіси. Такий спосіб не передбачає перебування поряд з хворим фахівця цілодобово. Більшість турбот все одно лягає на родичів, які іноді змушені залишати роботу, щоб доглядати хворого;

– в умовах денного стаціонару, медичний персонал веде прийом хворих, лікує в денному стаціонарі, дає необхідні консультації. Якщо денний стаціонар відсутній, то можливе закріплення за протибольовим кабінетом кількох ліжок у звичайній лікарні. Така форма є прийнятною тільки для тих пацієнтів, хто за станом здоров'я може сам відвідувати кабінет протибольової терапії, в інших випадках це неможливо;

– стаціонарно, надання паліативної допомоги передбачено в умовах стаціонару, тобто у спеціалізованих при лікарнях, хоспісах та будинках сестринського догляду. Стаціонарна допомога надається цілодобово медичним персоналом, який пройшов спеціальну підготовку. До неї входять медичні втручання для усунення болю, перебування у стаціонарі, харчування, психологічна підтримка хворого та його сім'ї тощо. Стаціонарна паліативна допомога рекомендується в більшості випадків, але багато хворих вважають за краще провести останні дні будинку, у звичній обстановці.

Паліативна допомога поєднує три підходи.

Купірування болю та больових відчуттів. Для цього проводиться симптоматична терапія, яка знімає болючі напади, викликані хворобою. Вона має на меті забезпечити максимально задовільну якість життя за мінімального сприятливого прогнозу. Для ефективного усунення больових відчуттів потрібно точно оцінити їх характер, створити тактику боротьби та забезпечити належний догляд за хворим. Наприклад, при щоденних тяжких головних болях, спричинених мігренню, самостійний прийом знеболювальних може лише спровокувати нові напади. Фахівець паліативної медицини, зокрема невролог, призначить пацієнтові правильне лікування, порадить комплекс заходів щодо фізичної реабілітації, складе правильний режим дня. Найбільш доступним і простим у усуненні болю способом є фармакотерапія.

Психологічна підтримка. Тяжка хвороба, госпіталізація, операції, зміна способу життя, можлива інвалідність та загроза смерті впливають на психологічний стан хворого негативно. Хворий не здатний адаптуватися до

нових умов існування, він постійно відчуває страх, приреченість, що несприятливо впливає на загальний стан. Родичі зазвичай також не можуть підтримати хворого психологічно, оскільки самі відчувають стрес. Паліативна допомога передбачає роботу психологів як із хворим, так і з його близькими. Іноді у цій роботі беруть участь волонтери. За бажанням хворого можуть бути проведені релігійні обряди.

Соціальна підтримка. Паліативна допомога включає соціальну підтримку сім'ї хворого та його самого. Адже психологічні проблеми посилюються ще й від усвідомлення соціальних труднощів, спричинених витратами на догляд та лікування. У багатьох хворих виникають матеріальні проблеми, комусь необхідне поліпшення житлових умов, але мало хто поінформований про доступні їм соціальні пільги. Фахівці соцслужби зобов'язані провести вивчення соціальних проблем пацієнта, розробити разом із лікарями план соціальної реабілітації, поінформувати хворого про його права та можливі пільги та допомогти їх отримати.

У сучасному світі більшість підходів паліативної медицини спрямовані на очний контакт з спеціалістом в області медицини. Таким чином, теперішній час для покращення життя для паліативних хворих є необхідність впровадження медичних інформаційно-довідкових систем, а саме систем, які би працювали онлайн [16].

2.2 Розробка моделі інформаційної системи підтримки паліативних пацієнтів

2.2.1 Концепція інформаційної моделі підтримки паліативних пацієнтів

На теперішній час у сучасному світі більшість медичних інформаційних систем (МІС) направлені на інформаційну підтримку лікарів та в меншій мірі на підтримку пацієнтів.

Розроблюючи концептуальну модель враховуються основні фактори, за допомогою яких система зможе працювати ефективно.

Одним із найкращих варіантів для розкриття суті інформаційної моделі є UML-діаграми та методологія IDEF0.

Методологію IDEF0 можна вважати наступним етапом розвитку відомої графічної мови опису функціональних систем SADT (Structured Analysis and Design Technique). Історично, IDEF0, як стандарт був розроблений в 1981 році в рамках великої програми автоматизації промислових підприємств, яка мала позначення ICAM (Integrated Computer Aided Manufacturing) і була запропонована департаментом Військово-Повітряних Сил США. Власне сімейство стандартів IDEF успадкувало своє позначення від назви цієї програми (IDEF = ICAM DEFinition). У процесі практичної реалізації учасники програми ICAM зіткнулися з необхідністю розробки нових методів аналізу процесів взаємодії у промислових системах. При цьому, крім удосконаленого набору функцій для опису бізнес-процесів, однією з вимог до нового стандарту була наявність ефективної методології взаємодії в рамках «аналітик-фахівець». Іншими словами, новий метод повинен був забезпечити групову роботу над створенням моделі, за участю всіх аналітиків і фахівців, зайнятих у рамках проекту [17].

Через війну пошуку відповідних рішень народилася методологія функціонального моделювання IDEF0. З 1981 року стандарт IDEF0 зазнав кілька незначних змін, в основному обмежуючого характеру, і остання його редакція була випущена в грудні 1993 року Національним Інститутом Стандарів і Технологій США.

Графічна мова IDEF0 напрочуд проста і гармонійна. В основі методології лежать чотири основні поняття:

Першим є поняття функціонального блоку (Activity Box) (рис. 2.1). Функціональний блок графічно зображується у вигляді прямокутника і уособлює деяку конкретну функцію в рамках аналізованої системи. За

вимогами стандарту назва кожного функціонального блоку має бути сформульована у дієслівному способі.

Кожна із чотирьох сторін функціонального блоку має своє певне значення (роль), при цьому:

- верхня сторона має значення «Керування» (Control);
- ліва сторона має значення «Вхід» (Input);
- права сторона має значення «Вихід» (Output);
- нижня сторона має значення «Механізм» (Mechanism).

Кожен функціональний блок у рамках єдиної системи, що розглядається, повинен мати свій унікальний ідентифікаційний номер.

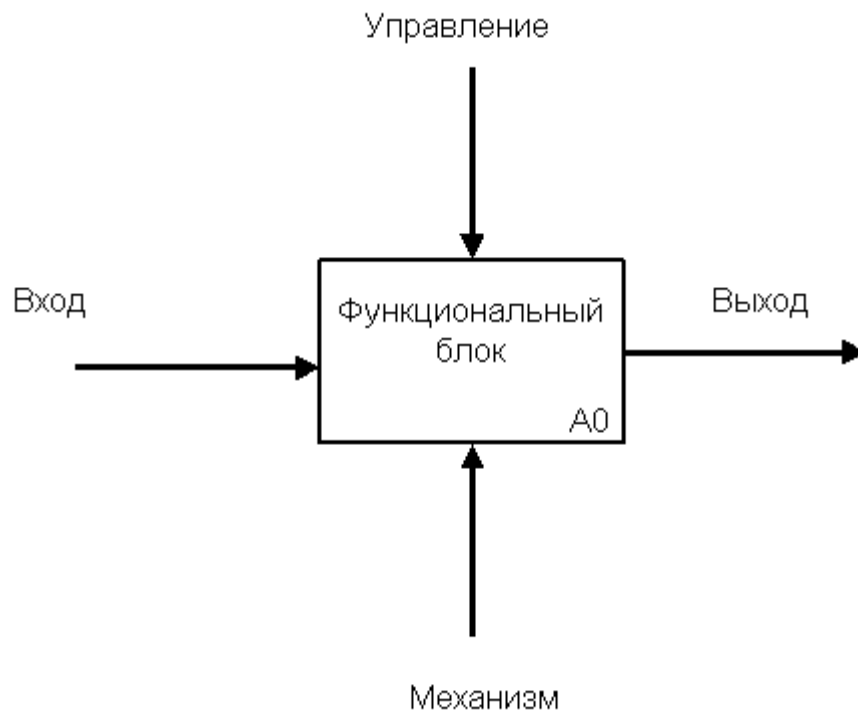


Рисунок 2.1 – Функціональний блок

Другим «китом» методології IDEF0 є поняття інтерфейсної дуги (Arrow). Також інтерфейсні дуги часто називають потоками чи стрілками. Інтерфейсна дуга відображає елемент системи, який обробляється функціональним блоком або впливає на функцію, відображену цим функціональним блоком.

Графічним відображенням інтерфейсної дуги є спрямована однонаправлена стрілка. Кожна інтерфейсна дуга повинна мати своє унікальне найменування (Arrow Label). На вимогу стандарту, найменування має бути оборотом іменника.

За допомогою інтерфейсних дуг відображають різні об'єкти, що тією чи іншою мірою визначають процеси, що відбуваються в системі. Такими об'єктами можуть бути елементи реального світу або потоки даних та інформації.

Залежно від того, до якої зі сторін підходить дана інтерфейсна дуга, вона називається «вхідною», «вихідною» або «керуючою». Крім того, «джерелом» (початком) і «приймачем» (кінцем) кожної функціональної дуги можуть бути тільки функціональні блоки, при цьому «джерелом» може бути тільки вихідна сторона блоку, а «приймачем» будь-яка з трьох, що залишилися.

Необхідно відзначити, що будь-який функціональний блок за вимогами стандарту повинен мати принаймні одну інтерфейсну дугу, що управляє, і одну вихідну. Кожен процес має відбуватися за якимись правилами (відображається керуючою дугою) і повинен видавати певний результат (дуга, що виходить), інакше його розгляд не має жодного сенсу.

При побудові IDEF0 – діаграм важливо правильно відокремлювати вхідні інтерфейсні дуги від керівників.

У реальному процесі робочому, що робить обробку, видають заготовлю і технологічні вказівки з обробки (або правила техніки безпеки при роботі зі верстатом). Помилково може здатися, що й заготівля та документ із технологічними вказівками є вхідними об'єктами, проте це не так.

В ході роботи діаграма IDEF0 інформаційної моделі підтримки паліативних пацієнтів, яка зображена на рисунку 2.2.

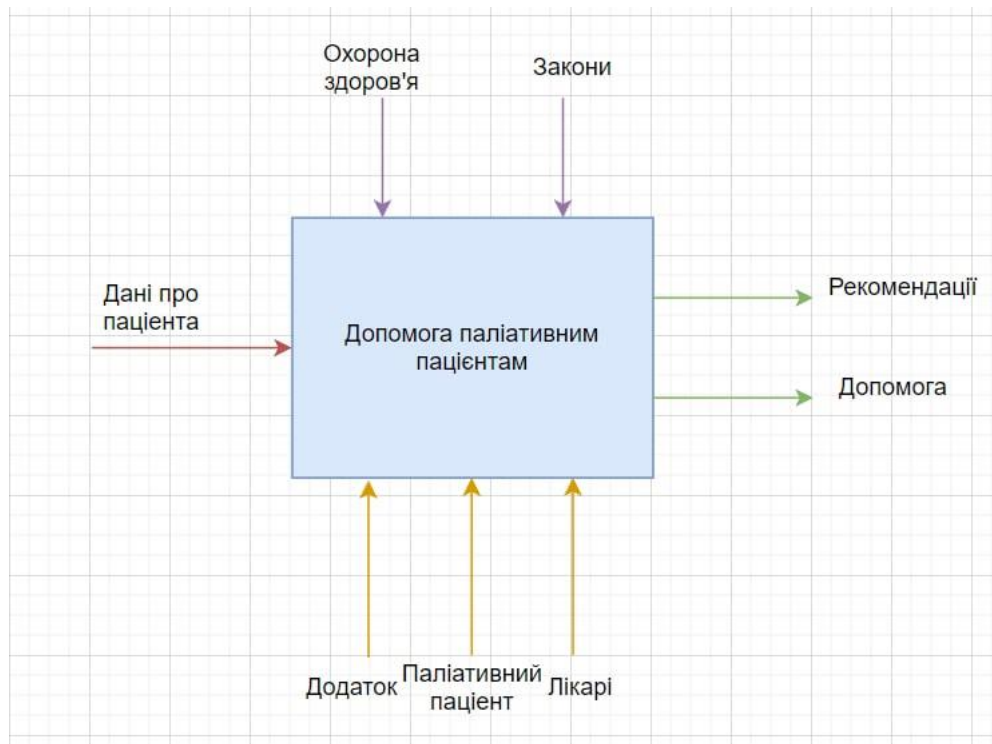


Рисунок 2.2 – Діаграма IDEF0 інформаційної моделі підтримки паліативних пацієнтів

UML (Unified Modeling Language) стандарт призначений для спрощення спілкування та взаємодії учасників проекту, скорочення часу на пояснення та засвоєння інформації та для полегшення документування.

UML – графічна нотація, призначена для опису і моделювання процесів, що протікають в ході розробки. Діаграми даної нотації розрізняються за типами і описують різні аспекти розробки. Розрізняють два основні типи UML-діаграм: структурні та поведінкові.

Структурні діаграми відображають елементи, з яких складається система. Поведінкові моделі описують процеси, що протікають в системі.

У кожній мові моделювання існує свій словник. Словник UML складається з наступних елементів:

- сутності – це абстракції, які є основними елементами моделей;
- відносини – це зв'язки між сутностями;
- діаграми – це відображення взаємодії сутностей і відносин.

Сутності в UML може бути 4 типів:

- структурні сутності є іменами існуючими моделі, її статичні частини. Це класи, компоненти, інтерфейси і т.д., які відповідають фізичним елементам системи;
- поведінкові сутності є дієсловами моделі, описують її поведінку в часі і просторі;
- анотаційні сутності є пояснювальними частинами моделі. Це примітки та коментарі до елементів системи;
- групуючі сутності – організуючі частини моделі. Вони з'єднують елементи системи групи.

Існує 4 типи відносин в UML:

- залежність, відношення між двома сутностями, при якому зміна однієї (незалежної) сутності призводить до зміни другої сутності. Графічно це зображається пунктирною стрілкою;
- узагальнення, відношення «спеціалізації-узагальнення», де спеціалізований об'єкт може бути підставлений замість узагальненого;
- асоціація, ставлення, що описує семантичний зв'язок між об'єктами. Графічно зображується у вигляді суцільної стрілки, яка може містити кратність або імена ролей. Агрегація – різновид асоціації і відображає відношення частини до цілого;
- композиція, різновид агрегації, де взаємозв'язок частини з цілим ще більш великий;
- реалізація, відношення між класифікаторами, при якому один визначає зобов'язання, а другий здійснює його виконання.

Приклад зображення відносин у UML зображено на рисунку 2.3.



Рисунок 2.3 – Типи відносин в UML

В UML існує 10 типів діаграм:

- об'єктів (object);
- класів (class);
- взаємодії (interaction);
- варіантів використання (use-case);
- послідовності (sequence);
- станів (statechart);
- кооперацій (Colaboration);
- компонентів (component);
- розгортання (розміщення) (deployment);
- діяльності (activity).

Під час аналізу було виявлено, що МІС Health Help провела моделювання своєї системи з використання мови UML. Під час моделювання найбільша увага була сконцентрована на діаграмі класів, діаграмі варіантів та діаграмі діяльності. Діаграма класів була використана для опису бекенда і дозволила прискорити розуміння бекенд-розробником створюваної системи, а також спростила та уніфікувала процес створення та найменування класів та їх атрибутів. Діаграма варіантів використання допомогла при визначенні сценаріїв користувача. Діаграма дозволила розробити кожен сценарій, позначити взаємодію різних користувачів із системою та виявити сценарії, які є включенням та розширенням існуючих. Таким чином була отримана найбільш повна картина сценаріїв системи.

Діаграми діяльності була використана для відображення всього функціонального потоку та набору дій, які може здійснювати користувач у системі для досягнення певного результату з усіма паралельними, альтернативними діями, а також взаємодія користувачів, які мають різні ролі в системі: наприклад, звичайний користувач та адміністратор тощо.

В ході роботи була розроблена діаграма класів (рис. 2.4.) Діаграма класів пояснює структуру системи підтримки паліативних пацієнтів та сутність моделі. Всього у системі представлено 8 сутностей: прийом, лікар,

спеціалізація, лікарня, пацієнт, нотатки, магазин, ліки. Кожна з них з'єднана різними типами зав'язків.

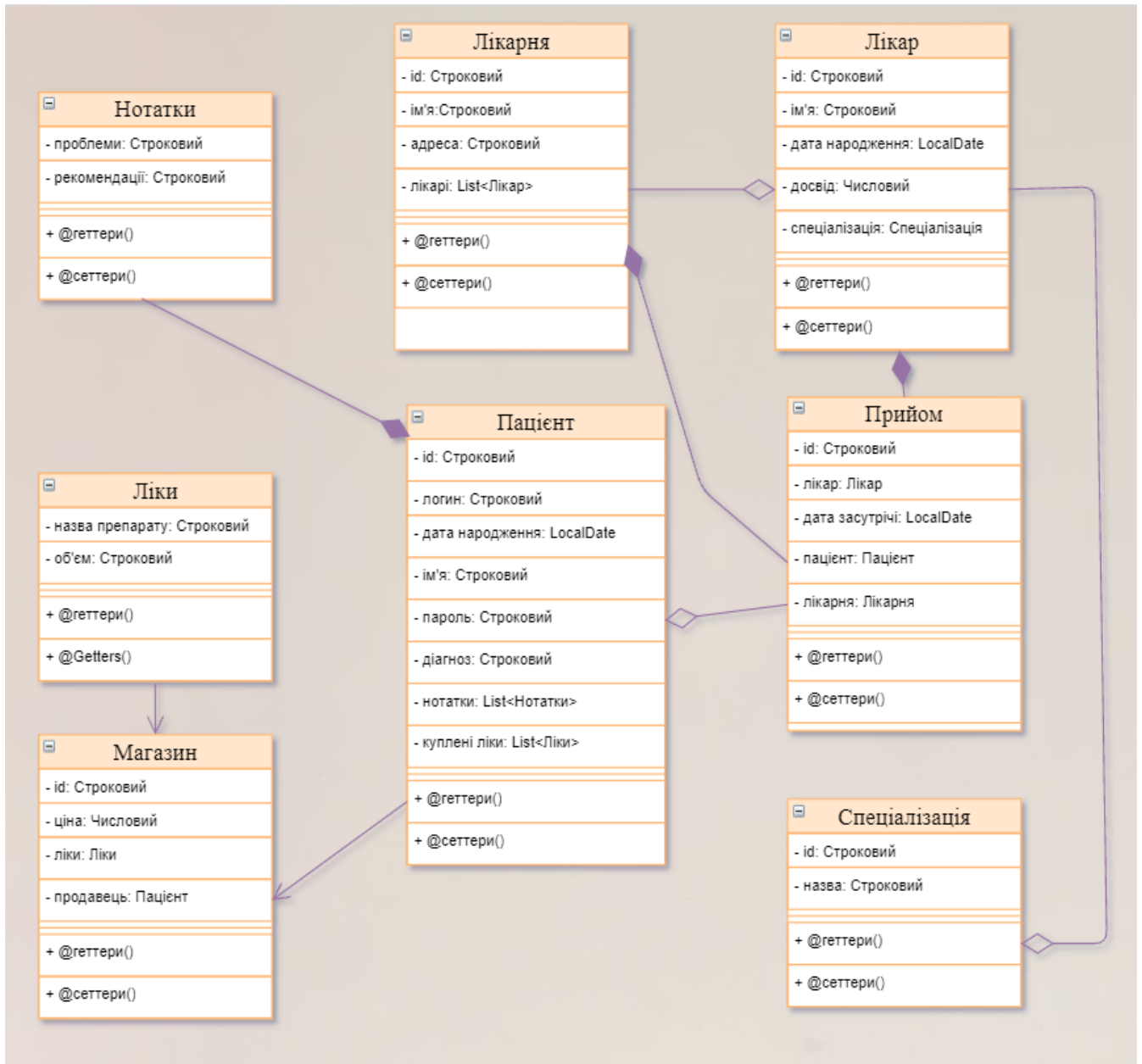


Рисунок 2.4 – Діаграма класів інформаційної моделі підтримки паліативних пацієнтів

Опис класів показано у таблиці 2.1.

Таблиця 2.1 – Опис класів інформаційної моделі підтримки паліативних пацієнтів

Клас	Опис класу	Склад полів
Спеціалізація	Спеціалізація лікаря	Id спеціалізації лікаря
		Назва спеціалізації
Лікар	Клас лікар, відображення зареєстрованого у МІС лікаря який надає онлайн допомогу паліативному пацієнту	Id лікаря
		Ім'я лікаря
		Дата народження лікаря
		Досвід лікаря
		Спеціалізація лікаря
Лікарня	Інформація про лікарню	Id лікарні
		Назва лікарні
		Адреса лікарні
		Лікарі лікарні
Прийом	Клас який відповідає за зустріч між лікарем та пацієнтом у конкретній лікарні	Id прийому
		Лікар
		Дата зустрічі
		Пацієнт
		Лікарня
Пацієнт	Клас, який зберігає всю інформацію про паліативного пацієнта	Id пацієнта
		Логин пацієнта
		Дата народження пацієнта
		Ім'я пацієнта
		Пароль пацієнта
		Діагноз пацієнта
		Нотатки пацієнта
		Куплені ліки пацієнтом

Продовження таблиці 2.1

Клас	Опис класу	Склад полів
Ліки	Клас, який зберігає інформацію про ліки	Назва препарату
		Об'єм препарату
Магазин	Клас, який зберігає інформацію про ліки на продажі та їх продавця	Id
		Ціна
		Продавець
		Ліки
Нотатки	Клас, який знає про проблему, що турбує пацієнта та зберігає рекомендацію у вирішенні цієї проблеми.	Проблеми пацієнта
		Рекомендації до проблеми

Діаграма варіантів використання інформаційної моделі підтримки паліативних пацієнтів наведена на рисунку 2.5.

Діаграма варіантів використання на якій зображено відношення між акторами та варіантами використання в системі підтримки паліативних пацієнтів.

Суть даної діаграми полягає в наступному: проєктована система представляється у вигляді безлічі сутностей чи акторів, що взаємодіють із системою за допомогою так званих варіантів використання. Варіант використання використовують для описання послуг, які система надає актору. Іншими словами, кожен варіант використання визначає деякий набір дій, який виконує система при діалозі з актором. При цьому нічого не говориться про те, яким чином буде реалізована взаємодія акторів із системою.

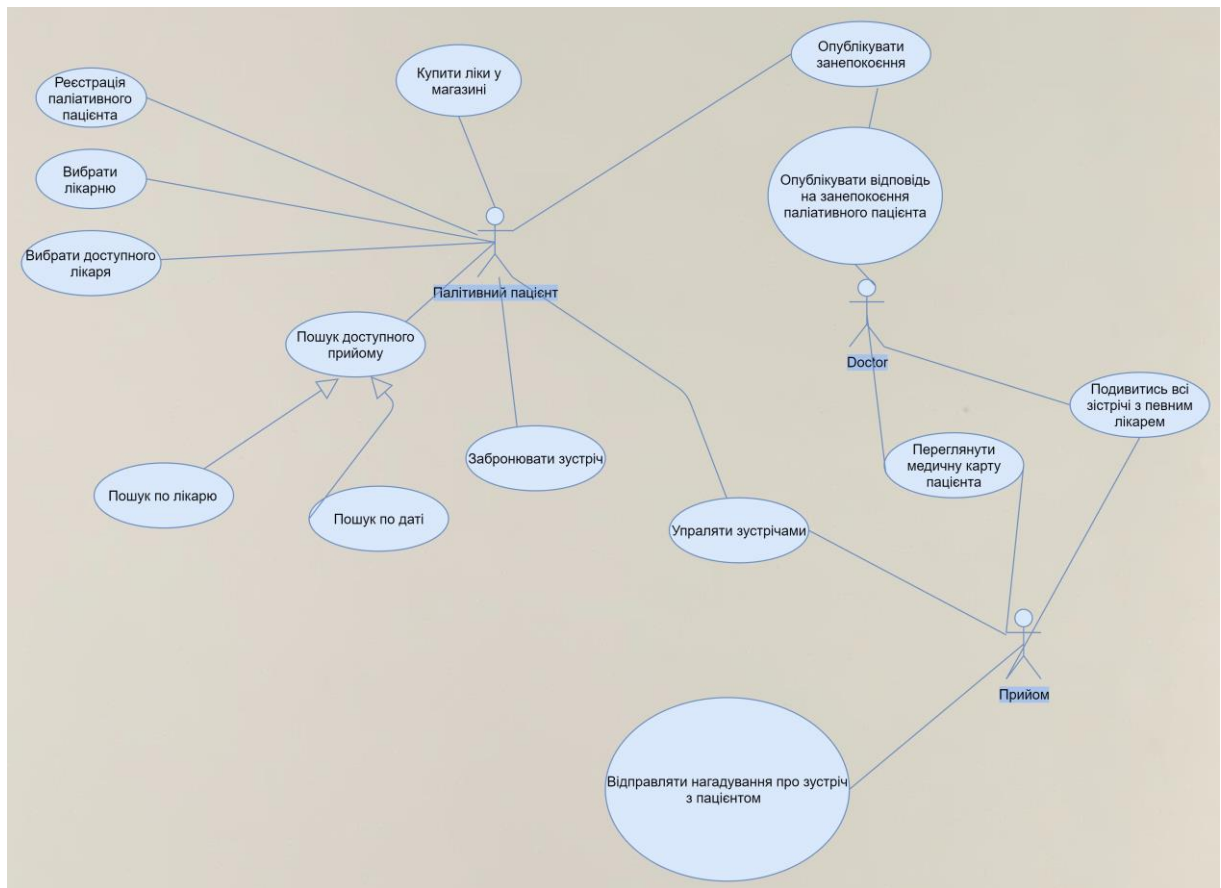


Рисунок 2.5 – Діаграма варіантів використання інформаційної моделі підтримки паліативних пацієнтів

Основні можливості системи підтримки паліативних пацієнтів наступні. Пацієнт має можливість зареєструватися у системі, переглянути свій профіль, переглянути список лікарень та їх лікарів, обрати лікаря та призначити зустріч з ним. У даній системі передбачена можливість скористуватись магазином, у якому користувачі системи (пацієнти) зможуть придбати ліки, які були виставлені на продаж іншими пацієнтами (у сучасному світі дуже складно швидко знайти ліки, які є життєво необхідними паліативним хворим, тому вони зможуть їх придбати у інших людей за зручними цінами та за наявності чека аптеки, або здійснити благодійно-безкоштовну передачу-обмін медичних препаратів). Крім того, в даній системі кожен пацієнт зможе отримати швидку онлайн консультацію найдосвідченіших лікарів та переглянути інформаційний довідник системи.

Для розробки спеціалізованої (вузькоспрямованої) автоматизованої інформаційної системи потрібно залучити достатню кількість експертів в певній галузі (зокрема фахівців в паліативній медицині) та інвестувати великі кошти, тому в даній роботі розробляється мобільний додаток з двома основними функціями: магазин перепродажу/обміну ліків між пацієнтами та онлайн підтримка для людей, які хворі діабетом та на СНІД. У подальшому планується розширення застосунку даної системи, в якому буде присутній мобільний кабінет для лікарів.

Таким чином, для програмної реалізації моделі підтримки паліативних пацієнтів необхідно скласти діаграми діяльності та діаграми послідовності.

2.2.2 Діаграми діяльності та послідовності моделі інформаційної системи

Діаграма діяльності функції онлайн консультації («проблема до лікаря») розроблена з використанням уніфікованої мови моделювання, яка використовується у парадигмі об'єктно-орієнтованого програмування та наведена на рисунку 2.6.

Дана діаграма діяльності описує принцип роботи функції «проблема до лікаря».

На початку пацієнт проходить авторизацію. У разі відсутності акаунта йому необхідно пройти реєстрацію. Після реєстрації користувач авторизується в системі, в меню навігації обирає функцію «проблема до лікаря». На екрані з'являється довідник з існуючими питаннями (в бета процесі функція підтримує тільки людей хворих на діабет та СНІД). Пацієнт аналізує інформацію довідника та обирає варіант онлайн консультації відповідно до стану здоров'я. У разі відсутності відповіді на конкретне питання пацієнт очікує на консультацію від досвідчених лікарів [18].

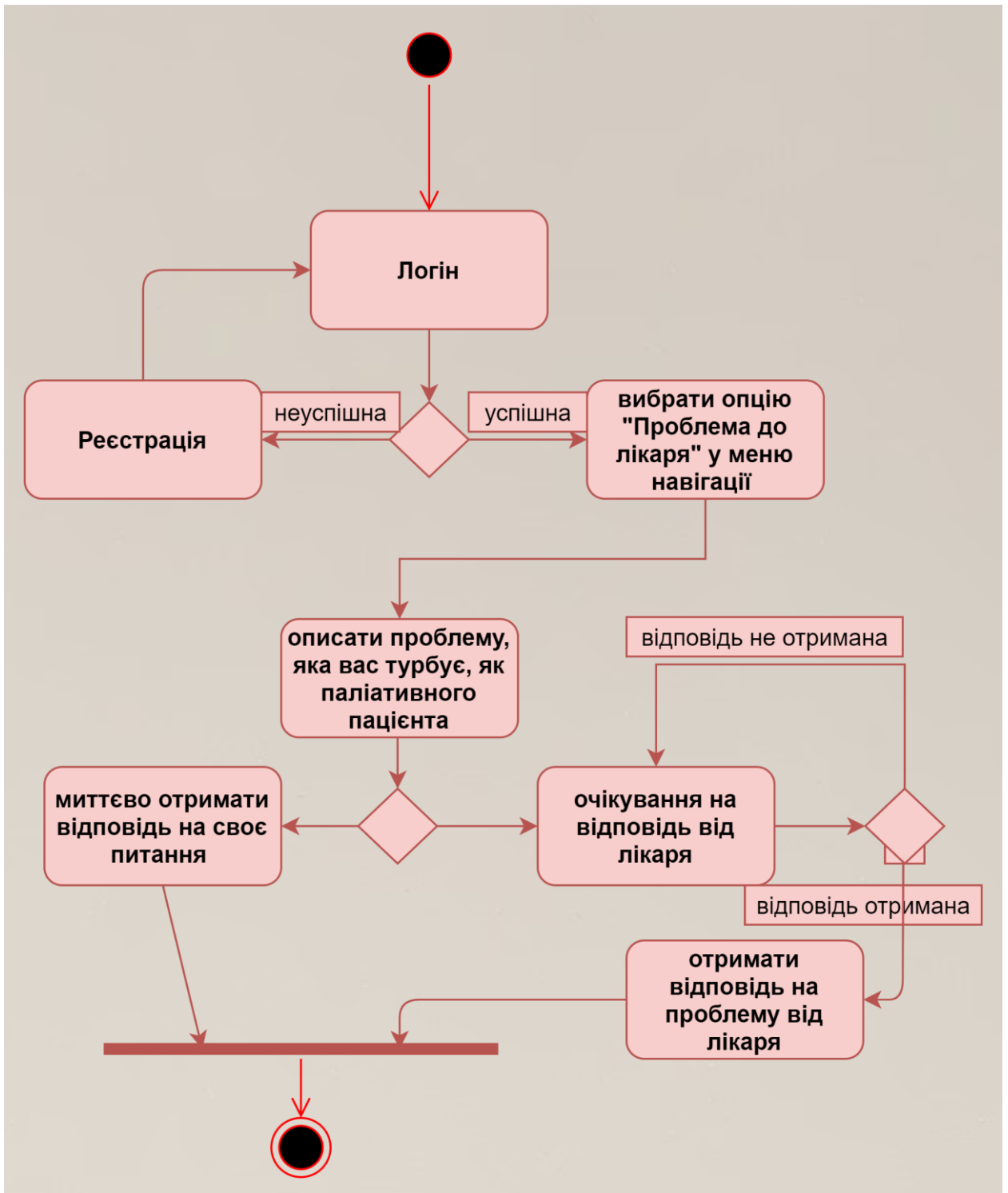


Рисунок 2.6 – Діаграма діяльності функції «проблема до лікаря»

Діаграма послідовності принципу роботи магазину перепродажу/обміну ліків між пацієнтами та онлайн підтримка для людей, які хворі на діабет. наведена на рисунку 2.7

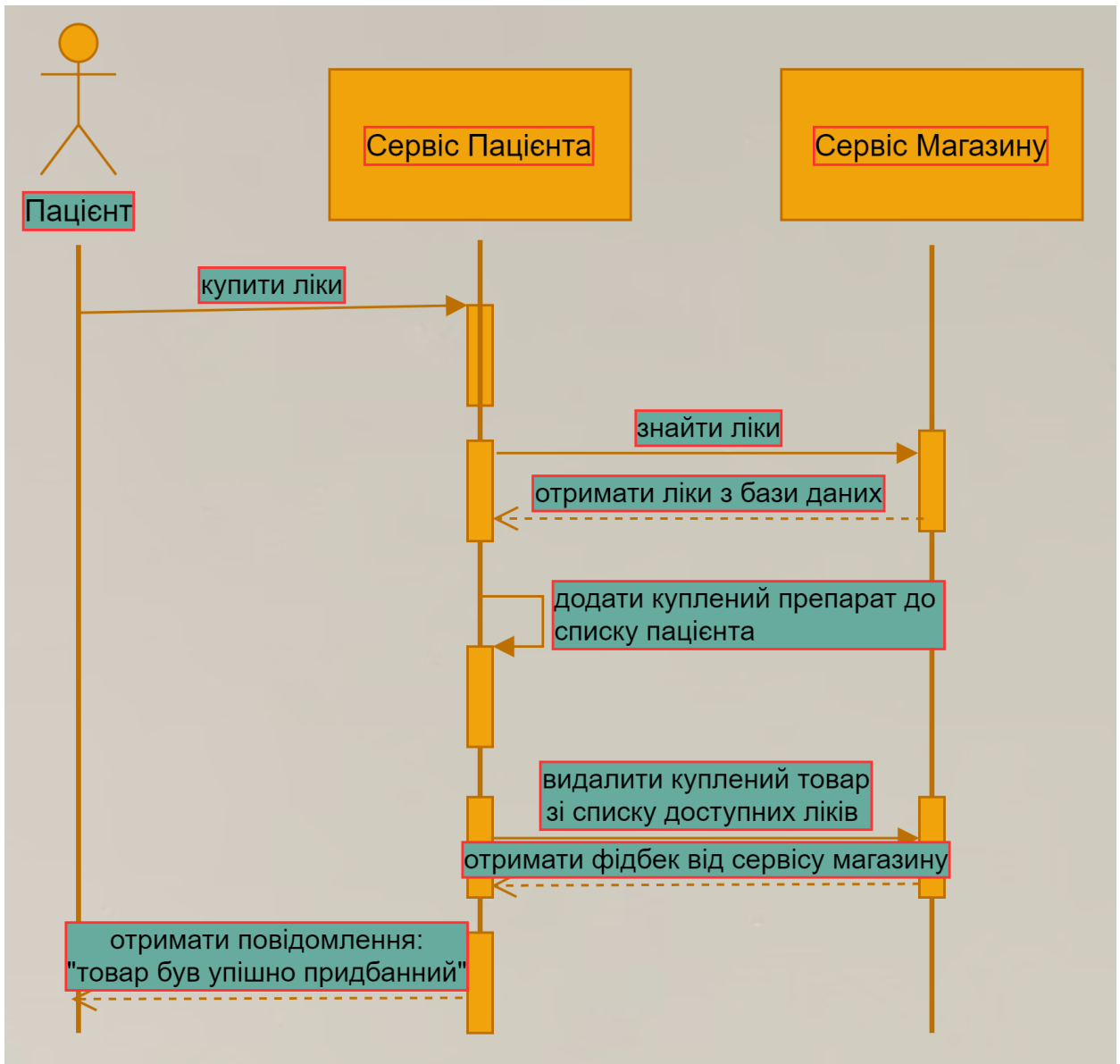


Рисунок 2.7 – Діаграма послідовності принципу роботи магазину перепродажу/обміну ліків між пацієнтами

Дана діаграма послідовності принципу роботи магазину перепродажу/обміну ліків між пацієнтами.

На початку клієнт виконує запит на купівлю конкретного товару. Сервіс пацієнта, обробляє його запит та відсилає запит для пошуку мед. препарату до сервісу магазину. Сервіс пацієнта дочекавшись відповіді, добавляє товар обраний пацієнтом до його списку куплених товарів пацієнта. Далі сервіс магазину видаляє товар з магазину, а сервіс пацієнта повідомляє користувача (пацієнта) про успішну покупку медичних препаратів або їх обмін [19].

Висновки за розділом:

- аналіз існуючих методів підтримки паліативних пацієнтів показав, що на сьогодні актуальним є питання інформацізації підсистеми підтримки паліативних пацієнтів;
- одним із найкращих варіантів для розкриття суті інформаційної моделі є UML-діаграми та методологія IDEF0;
- у ході розробки моделі інформаційної підсистеми підтримки паліативних пацієнтів побудовано такі діаграми: діаграма варіантів використання, діаграма класів, діаграма діяльності та діаграма послідовності, які дозволяють виконати програмну реалізацію інформаційної підсистеми підтримки паліативних пацієнтів [20].

3 ПРОГРАМНА РЕАЛІЗАЦІЯ КОМПОНЕНТУ МЕДИЧНОЇ ІНФОРМАЦІЙНОЇ ПІДСИСТЕМИ ПІДТРИМКИ ПАЛІАТИВНИХ ПАЦІЄНТІВ

3.1 Використані технології

Для обробки даних з бази у компоненті медичної інформаційної підсистеми підтримки паліативних пацієнтів було використано популярний фреймворк Hibernate.

Hibernate – це фреймворк Java, який спрощує розробку програми Java для взаємодії з базою даних. Це легкий інструмент з відкритим вихідним кодом, ORM (об’єктно-реляційне відображення). Hibernate реалізує специфікації JPA (Java Persistence API) для збереження даних [21].

Інструмент ORM спрощує створення даних, маніпулювання даними та доступ до них. Це техніка програмування, яка зіставляє об’єкт з даними, що зберігаються в базі даних. На рисунку 3.1 зображено схему дії ORM.

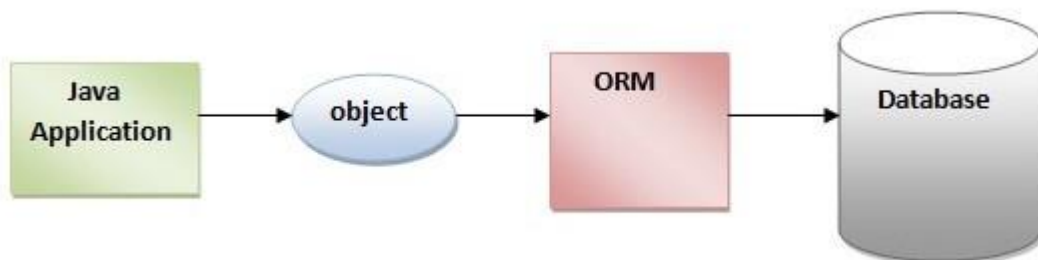


Рисунок 3.1 – Основна мікросервісна архітектура

Переваги Hibernate Framework:

- з відкритим вихідним кодом і легким Фреймворк Hibernate є відкритим вихідним кодом за ліцензією LGPL і легким;
- швидка продуктивність. Продуктивність фреймворку сплячого режиму висока, оскільки кеш використовується внутрішньо в фреймворку сплячого режиму. Існують два типи кешу в кеші першого рівня і кешу

другого рівня фреймворку глибокого сну. Кеш першого рівня увімкнено за замовчуванням;

– незалежний запит від бази даних HQL (Hibernate Query Language) – це об'єктно-орієнтована версія SQL. Він генерує незалежні запити від бази даних. Таким чином, не потрібно писати специфічні запити до бази даних. Перед Hibernate, якщо базу даних змінено для проекту, нам також потрібно змінити запит SQL, що призведе до проблеми обслуговування;

– автоматичне створення таблиці Фреймворк Hibernate надає можливість автоматичного створення таблиць бази даних. Тому немає необхідності створювати таблиці в базі даних вручну;

– спрощує складне об'єднання Отримати дані з кількох таблиць легко в режимі глибокого сну;

– надає статистику запитів і статус бази даних Hibernate підтримує кеш запитів і надає статистику про стан запитів і бази даних.

Мовою програмування було обрано Java версії 14. Java 14 зберігла стандарти Oracle, а саме пришвидшувати інновації, надаючи підприємствам і спільноті розробників нові вдосконалення, випускаючи нові функції кожні шість місяців. Останній комплект розробки Java (JDK) надає нові функції, включаючи дві нові довгоочікувані функції попереднього перегляду – відповідність шаблону, наприклад, `instanceof` (JEP 305) і записи (JEP 359), а також другий попередній перегляд текстових блоків (JEP 368). Крім того, остання версія Java додає підтримку мови Java для виразів перемикачів, відкриває нові API для безперервного моніторингу даних JDK Flight Recorder, розширює доступність Z Garbage Collector з низькою затримкою для macOS і Windows, а також додає в модулі інкубатор, упаковку автономних програм Java та новий API доступ до зовнішньої пам'яті для безпечного та ефективного користування за межами купи Java.

3.2 Архітектура серверної частини

Для написання серверної частини компоненту МІС було обрано монолітну архітектуру.

Монолітна архітектура вважається традиційним способом побудови застосунків. Монолітний додаток будується як єдиний і неподільний блок. Зазвичай таке рішення містить клієнтський інтерфейс користувача, серверну програму та базу даних. Він уніфікований, і всі функції керуються та обслуговуються в одному місці. Зазвичай монолітні програми мають одну велику кодову базу і не мають модульності. Якщо необхідно щось оновити або змінити, вони мають доступ до тієї самої бази коду. Таким чином, вони вносять зміни у весь стек відразу [22]. На рисунку 3.2 зображено основну ідею монолітної архітектури.

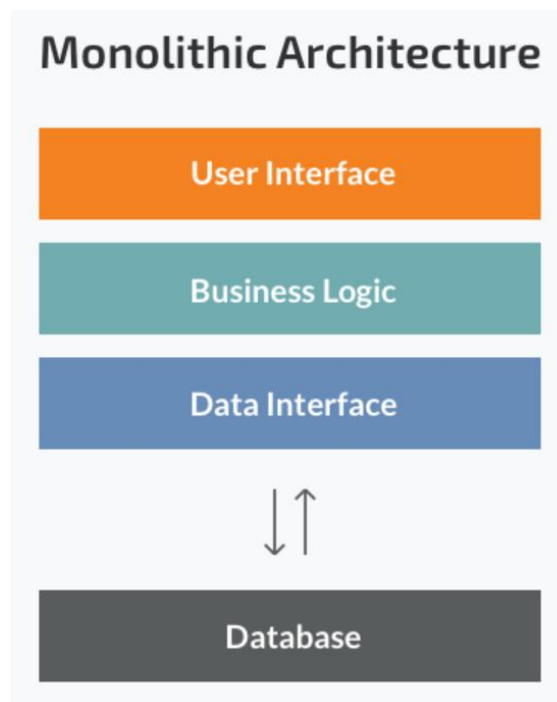


Рисунок 3.2 – Ідея монолітної архітектури

Сильні сторони монолітної архітектури:

– менше наскрізних проблем. Наскрізні проблеми – це проблеми, які впливають на всю програму, як-от ведення журналів, обробка, кешування та

моніторинг продуктивності. У монолітному застосунку ця область функціональності стосується лише однієї програми, тому з нею простіше працювати;

– простіше налагодження та тестування. На відміну від архітектури мікросервісів, монолітні програми набагато легше налагоджувати та тестувати. Оскільки монолітний додаток є єдиним неподільним блоком, ви можете виконувати наскрізне тестування набагато швидше;

– простий у розгортанні. Ще одна перевага, пов'язана із простотою монолітних застосунків – це легше розгортання. Коли справа доходить до монолітних програм, не потрібно обробляти багато розгортань – лише один файл або каталог;

– простий у розробці. Поки монолітний підхід є стандартним способом створення застосунків, будь-яка команда інженерів має необхідні знання та можливості для розробки монолітного застосунку.

На рисунку 3.3 зображена схема монолітної архітектури розроблюваного компоненту МІС.

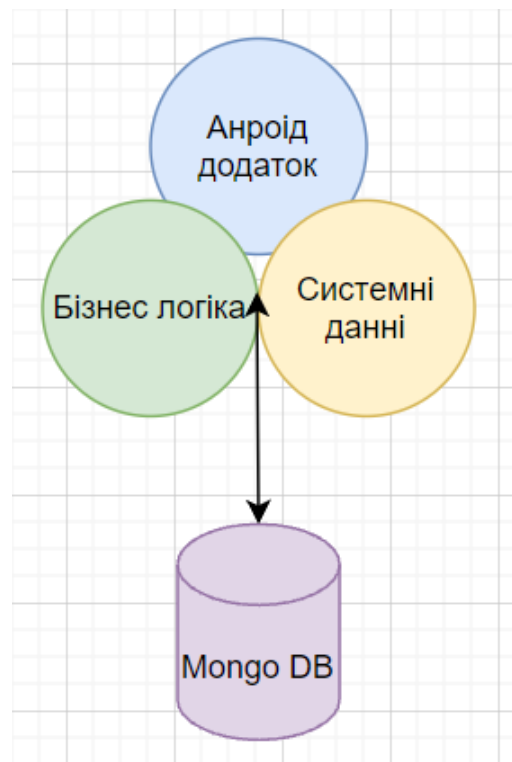


Рисунок 3.3 – Монолітна архітектура МІС

Монолітна архітектура має три компоненти:

- інтерфейс користувача на стороні клієнта;
- додаток на стороні сервера;
- інтерфейс даних.

Всі три частини взаємодіють з єдиною базою даних. Програмне забезпечення, побудоване на цій моделі, працює з однією базою коду. У результаті, щоразу, коли зацікавлені сторони хочуть внести оновлення або зміни, вони мають доступ до того самого набору коду.

3.3 Реалізація клієнтської частини

Для написання інтерфейсу клієнта було обрано Android додаток. В якості фреймворку обрано React Native.

React Native – це фреймворк JavaScript для написання реальних мобільних застосунків для iOS та Android. Він заснований на React, бібліотеці JavaScript від Facebook для створення користувацьких інтерфейсів, але замість того, щоб орієнтуватися на браузер, він націлений на мобільні платформи. Іншими словами: веб-розробники тепер можуть писати мобільні застосунки, які виглядають і відчуються справді «рідними», не вдаючись до бібліотеки JavaScript. Крім того, оскільки більшість коду, можна спільно використовувати між платформами, React Native полегшує одночасну розробку як для Android, так і для iOS. Подібно до React для Інтернету, програми React Native написані з використанням суміші JavaScript і XML-розмітки, відомої як JSX. Потім під капотом React Native «міст» викликає нативні API візуалізації в Objective-C (для iOS) або Java (для Android) [23]. Таким чином, програма відтворюватиметься за допомогою реальних компонентів мобільного інтерфейсу, а не веб-переглядів, і буде виглядати і відчувати себе як будь-який інший мобільний додаток. React Native також надає інтерфейси JavaScript для API платформ, тому програми React Native

можуть отримати доступ до функцій платформи, таких як камера телефону або місцезнаходження користувача. React Native наразі підтримує як iOS, так і Android, а також має потенціал для розширення на майбутні платформи.

React Native фактично відтворює за допомогою стандартних API візуалізації своєї хост-платформи, дозволяє йому виділитися з більшості існуючих методів розробки кросплатформних застосунків, таких як Cordova або Ionic. Існуючі методи написання мобільних застосунків із використанням комбінацій JavaScript, HTML і CSS зазвичай відображаються за допомогою веб-переглядів. Хоча цей підхід може працювати, він також має недоліки, особливо щодо продуктивності. Крім того, вони зазвичай не мають доступу до набору власних елементів інтерфейсу користувача хост-платформи. Коли ці фреймворки намагаються імітувати власні елементи інтерфейсу користувача, результати зазвичай «відчуваються» лише трохи [24]. Зворотна інженерія всіх дрібних деталей таких речей, як анімація, вимагає величезних зусиль, і вони можуть швидко застаріти.

3.4 Опис бази даних

Сховищем даних було обрано нереляційну базу даних MongoDB.

Технологія баз даних NoSQL зберігає інформацію в документах JSON замість стовпців і рядків, які використовуються реляційними базами даних. Щоб було зрозуміло, NoSQL означає «не тільки SQL», а не «без SQL» взагалі. Це означає, що база даних NoSQL JSON може зберігати та отримувати дані буквально «без SQL». Або ви можете поєднати гнучкість JSON з потужністю SQL для найкращого з обох світів. Отже, бази даних NoSQL створені так, щоб бути гнучкими, масштабованими та здатними швидко реагувати на потреби сучасного бізнесу в управлінні даними.

На рисунку 3.4 зображено приклад зберігання сутності у базі даних компоненту МІС.

```
{
  "_id" : ObjectId("6182f47158183fa09414b733"),
  "ФІО" : "Баварский Давид Лазович",
  "Дата народження" : "01.01.1984",
  "Досвід" : "7 років",
  "Спеціалізація" : "Терапевт"
}
```

Рисунок 3.4 – Сутність лікар у системі

3.5 Інтеграція зі сторонніми МІС

Реальна інтеграція зі сторонніми МІС була тимчасово замінена за допомогою макетів об'єктів (Mock Object). Макети у об'єктно-орієнтованому програмуванні – це об'єкти, що імітують поведінку справжніх об'єктів контрольованими способами, тобто, реалізують інтерфейси справжніх об'єктів, але не мають власної реальної функціональності. Для реалізації цієї ідеї було обрано WireMock HTTP mock сервер. За своєю суттю це веб-сервер, який може бути підготовлений для обслуговування стандартних відповідей на певні запити (заглушки) і який фіксує вхідні запити, щоб їх можна було перевірити пізніше (перевірка).

Він також має набір інших корисних функцій, включаючи запис/відтворення взаємодій з іншими API, ін'єкцію помилок і затримок, моделювання поведінки з визначенням стану.

Його можна використовувати як бібліотеку будь-яким застосунком JVM або запускати як окремий процес на тому ж хості, що й тестована система, або на віддаленому сервері.

Усі функції WireMock доступні через інтерфейс REST (JSON) та SOAP (XML) і Java API. Крім того, заглушки можна налаштувати за допомогою файлів JSON та XML [25].

Так для отримання інформації про пацієнта зі сторонніх МІС було використано mapping JSON-об'єкт (рис. 3.5) та XML файл-відповідь (рис. 3.6) з інформацією про певного паліативного пацієнта.

```
{
  "priority": 1000,
  "request": {
    "url": "/MIS/int:",
    "headers": {
      "SOAPAction": {
        "matches": "\"/MIS/HealthSystem//patient/*.\""
      }
    }
  },
  "response": {
    "status": 200,
    "headers": {
      "Content-Type": "text/xml"
    },
    "bodyFileName": "patient_response.xml",
    "transformers": [
      "response-template"
    ]
  }
}
```

Рисунок 3.5 – Mapping JSON-об'єкт

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <ns0:patientResponse xmlns:ns0="http://health.com/MIS">
      <ns0:patientName>Borodenko Ivan Ivanovich</ns0:patientName>
      <ns0:patientDiagnosis>AIDS</ns0:patientDiagnosis>
      <ns0:patientAge>50</ns0:patientAge>
      <ns0:patientDateOfBirth>03.11.1971</ns0:patientDateOfBirth>
      <ns0:patientNotes>swollen lymph nodes, weight loss, fever, diarrhea and cough</ns0:patientNotes>
    </ns0:patientResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Рисунок 3.6 – XML файл-відповідь

Так згідно XML файлу-відповіді було отримано інформацію про пацієнта «Borodenko Ivan Ivanovich».

3.6 Інструкція користувача

Для використання мобільного застосунку інформаційної підсистеми підтримки паліативних пацієнтів користувачу потрібно мати телефон/планшет на базі Android або використовувати емулятор, який підтримує ОС Android. Перед використанням додаток потрібно встановити на відповідний пристрій [26].

Для використання мобільного застосунку потрібно авторизуватися в системі або авторизувати хворого, якщо ви його опікун. Форма авторизації паліативно-хворого пацієнта наведена на рисунку 3.7.

Після авторизації пацієнт потрапляє на головне меню застосунку. Натиснувши на кнопку «проблема до лікаря» пацієнту відкривається список з інформацією про хворобу (у роботі реалізована функцію інформаційної підтримки хворих на СНІД) [27].

Так, наприклад, на обране питання хворого: «Здравствуйтесь. У мене вопрос. Может ли тест на ВИЧ быть отрицательным и в то же время на СПИД положительным?» (рис. 3.8) користувач функції «проблема до лікаря» отримує відповідь: «Здравствуйтесь, Руслан! Это один и тот же тест. СПИД – это последняя клиническая стадия ВИЧ-инфекции, которая развивается у пациентов, не принимающих антиретровирусные препараты.»

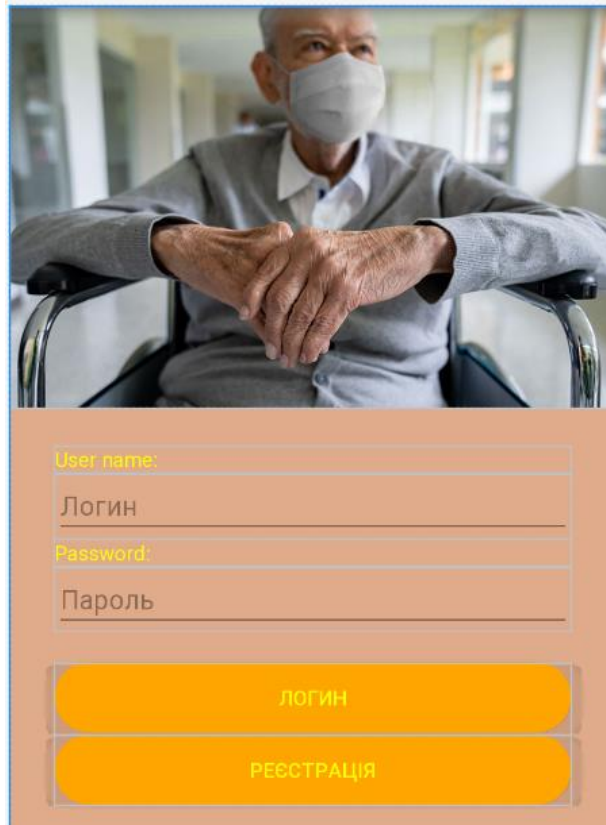


Рисунок 3.7 – Форма авторизації паліативно-хворого пацієнта

Руслан

29 октября 2021 20:39

Здравствуйте. У меня вопрос. Может ли тест на ВИЧ быть отрицательным и в то же время на СПИД положительным?

Здравствуйте, Руслан! Это один и тот же тест. СПИД – это последняя клиническая стадия ВИЧ-инфекции, которая развивается у пациентов, не принимающих антиретровирусные препараты.

[вопрос](#)

Рисунок 3.8 – Відповідь на питання паліативно-хворого пацієнта на СНІД

3.7 Автоматизоване тестування компоненту

Для тестування компоненту медичної інформаційної підсистеми підтримки паліативних пацієнтів було використано фреймворк Karate.

Karate – це єдиний інструмент з відкритим вихідним кодом, який поєднує в єдиній уніфікованій структурі автоматизацію тестування API,

макетів, тестування продуктивності та навіть автоматизацію інтерфейсу користувача. Синтаксис BDD, популяризований Cucumber, є мовним нейтральним і простим навіть для тих, хто не програмує. Твердження та звіти HTML вбудовані, і є можливість паралельного запуску тестів для підвищення швидкості розробки [28].

Існує також кросплатформний автономний виконуваний файл для команд, які не знають Java. Тести розробляються в простому, читабельному синтаксисі – ретельно розробленому для HTTP, JSON, GraphQL і XML. Присутня можливість змішувати API та автоматизацію тестування інтерфейсу користувача в рамках одного тестового сценарію [29].

Переваги Karate:

- скрипти є звичайним текстом, не вимагають кроку компіляції, а команди можуть співпрацювати за допомогою Git;
- інтегрований на основі Cucumber-JVM;
- присутня можливість тестувати та отримувати звіти результатів тестування [30];
- підтримка паралельного виконання програмного коду;
- підтримка створення тестів без знання мови програмування Java [31].

На рисунку 3.9 зображено сценарій отримання всіх запитів на допомогу паліативно хворого пацієнта Іванова.

```
@name=retrieveAdvisesForIvanovPatient
Scenario: Retrieve all advises for Ivanov patient
  Given url adviseUrl
  * param patient.id = Ivanov
  When method get
  Then status 200
```

Рисунок 3.9 – Сценарій отримання всіх запитів на допомогу паліативно хворого пацієнта Іванова

Результат звіту про успішне тестування зображено на рисунку 3.10.

The screenshot displays a test runner interface. On the left, a sidebar shows a 'Scenarios' list with one scenario: 'Retrieve all advises for Ivanov patient'. Above the list, a score is shown as '1' in a green box and '0' in a red box. The main area shows the test details for the selected scenario. At the top, a blue header contains 'Summary | Tags | Feature: src/test/java/apiTests/advisesV21/advises.feature *This feature file is used to test the advises handling of the API - V2.1.*'. Below this, a yellow bar indicates 'parallel=false' and 'ms: 4631'. The test steps are listed in a table:

Step	Description	Duration (ms)
>>	Background:	
16	Given url adviseUrl	3582
>>	* param patient.id = Ivanov	3552
17	When method get	1
18	Then status 200	2

Рисунок 3.10 – Звіт про успішне тестування

ВИСНОВКИ

У рамках кваліфікаційної роботи був розроблений і реалізований компонент інформаційної підсистеми підтримки паліативних пацієнтів.

Досліджено частину невирішених питань медичної інформаційної системи для підтримки паліативних пацієнтів.

Для розв'язання проблем була спроектована модель компоненту медичної інформаційної системи для підтримки паліативних пацієнтів.

Було розглянуто та проаналізовано існуючі рішення медичних інформаційних систем, досліджено існуючі методи підтримки паліативних пацієнтів.

Для реалізації системи була створена поетапна модель розробки та проектування кожного компонента розроблюваної системи, опис необхідних бізнес та функціональних вимог. Спроектовано та описано модель даних у системі, а також модель функціональних можливостей компоненту медичної інформаційної системи підтримки паліативних пацієнтів [32].

В якості тестування компоненту було розроблено модель автоматизованого тестування системи до її функціональних та бізнес вимог.

У ході проектування компоненту була розроблена модель основної концепції медичної інформаційної підсистеми підтримки паліативних пацієнтів.

У рамках розробки системи, було створено та адаптовано:

- компонент онлайн підтримки паліативних пацієнтів;
- компонент довідника для паліативно хворих на СНІД.

Результати роботи було апробовано на конференції «Trends in science and practice of today», 19-22 жовтня 2021р., Анкара, Туреччина [33].

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Паліативна допомога в Україні: куди рухатися далі? URL: <https://www.vz.kiev.ua/paliativna-dopomoga-v-ukrayini-kudy-ruhatysya-dali> (дата звернення: 19.10.2021).
2. Костащук, Я. (2018). КЛАСИФІКАЦІЯ ІНФОРМАЦІЙНИХ СИСТЕМ. In Інформаційне суспільство: проблеми формування і розвитку в сучасних умовах.
3. Балусьва, О. В. (2007). Удосконалення механізмів державного управління розвитком інформаційної системи галузі охорони здоров'я (Doctoral dissertation, ступеня канд. наук з державного управління: спец. 25.00. 02 „Механізми державного управління”/ОВ Балусьва.–Донецьк, 2007.–22 с).
4. What is a Health Information System? URL: <https://digitalguardian.com/blog/what-health-information-system> (дата звернення: 19.10.2021).
5. What is Health Information Technology? Benefits of Healthcare Software Systems <https://www.selecthub.com/medical-software/7-categories-healthcare-information-technology> (дата звернення: 19.10.2021).
6. Ситников, Д. Э., Титов, С. В., & Титова, Е. В. (2011). Семантические свойства информативности ассоциативных зависимостей между признаками объектов в базах данных.
7. How to Choose Medical Software URL: <https://www.businessnewsdaily.com/8787-choosing-medical-software.html> (дата звернення: 19.10.2021).
8. Implementing a Master Patient Index URL: <https://www.colleaga.org/article/implementing-master-patient-index> (дата звернення: 19.10.2021).

9. What is a patient portal? URL: <https://www.healthit.gov/faq/what-patient-portal> (дата звернення: 19.10.2021)

10. Remote patient monitoring (RPM) URL: <https://searchhealthit.techtarget.com/definition/remote-patient-monitoring-RPM> (дата звернення: 19.10.2021).

11. Паліативна допомога у новій Програмі медичних гарантій: перелік безоплатних послуг URL: <https://nszu.gov.ua/novini/paliativna-dopomoga-u-novij-programi-medichnih-garantij-pere-521> (дата звернення: 19.10.2021).

12. Паліативна допомога: в Україні будуть створені Центри медичної реабілітації та паліативної допомоги дітям URL: <https://moz.gov.ua/article/health/paliativna-dopomoga-v-ukraini-budut-stvoreni-centri-medichnoi-reabilitacii-ta-paliativnoi-dopomogi-ditjam> (дата звернення: 19.10.2021).

13. Паліативна допомога у новій Програмі медичних гарантій, - НСЗУ надає перелік безоплатних послуг URL: <https://www.kmu.gov.ua/news/paliativna-dopomoga-u-novij-programi-medichnih-garantij-nszu-nadaye-perelik-bezoplatnih-poslug> (дата звернення: 19.10.2021).

14. Паліативна допомога: де і як пацієнту отримати безоплатно URL: <https://www.rv.gov.ua/news/paliativna-dopomoga-de-i-yak-paciyentu-otrimati-bezoplatno> (дата звернення: 20.10.2021).

15. ВІДДІЛЕННЯ ПАЛІАТИВНОГО ЛІКУВАННЯ URL: <http://spital.org.ua/hospital/viddilennia-paliativnoho-likuvannia> (дата звернення: 19.10.2021).

16. Титова, Е. В. Сравнительная характеристика простых и расширенных ассоциативных правил для признаков объектов в базах данных. Системы обробки інформації–Харьков: НАНУ, ПАНМ, ХВУ.–2003.– №, 31-37.

17. Sitnikov, D., Titova, O., Minukhin, S., Kovalenko, A., & Titov, S. (2018, October). Informativity of association rules from the viewpoint of information

theory. In 2018 International Scientific-Practical Conference Problems of Infocommunications. Science and Technology (PIC S&T) (pp. 595-598). IEEE.

18. Семенець, А. В., & Марценюк, В. П. (2015). Розробка платформи системи підтримки прийняття рішення для медичної інформаційної системи з відкритим кодом OpenEMR. Медична інформатика та інженерія, (3), 22-40.

19. Ivashchuk, O., & Ivashchuk, D. (2021). ВИКОРИСТАННЯ МЕДИЧНИХ ІНФОРМАЦІЙНИХ СИСТЕМ У ФАХОВІЙ ПІДГОТОВЦІ МАЙБУТНІХ ЛІКАРІВ. Науковий вісник Ужгородського університету. Серія: «Педагогіка. Соціальна робота», (1 (48)), 166-169.

20. Паліативна та хоспісна допомога URL: <https://phc.org.ua/zakhvoryuvannya-ta-informaciya/vilsnid/likuvannya-ta-profilaktika/paliativna-ta-khospisna-dopomoga> (дата звернення: 19.10.2021).

21. All You Need to Know About UML Diagrams URL: <https://tallyfy.com/uml-diagram/> (дата звернення: 19.10.2021).

22. Гадецька, З. М., & Холопова, М. О. (2016). Моделювання бізнес-процесів діяльності підприємства. Ефективна економіка, (5).

23. What Is Java URL: <https://www.edureka.co/blog/what-is-java/> (дата звернення: 19.10.2021).

24. Why Is the Spring Framework so Popular? URL: <https://dzone.com/articles/why-spring-framework-is-popular> (дата звернення: 20.10.2021).

25. REST API: What is it, and what are its advantages in project development? URL: <https://bbvaopen4u.com/en/actualidad/rest-api-what-it-and-what-are-its-advantages-project-development> (дата звернення: 21.10.2021).

26. Introduction to Android Development URL: <https://code.tutsplus.com/articles/introduction-to-android-development--mobile-7> (дата звернення: 22.10.2021).

27. A type-safe HTTP client for Android and Java URL: <https://square.github.io/retrofit/> (дата звернення: 23.10.2021).

28. Overview of MongoDB URL: <https://www.studytonight.com/mongodb/overview-of-mongodb> (дата звернення: 23.10.2021).

29. Ситников, Д. Э., Ситникова, П. Э., Титов, С. В., & Титова, Е. В. (2019). Определение параметров обобщенных ассоциативных правил методом декомпозиции. Системи обробки інформації, (1), 58-63.

30. Тітова, О. В. (2018). Основи інформаційного забезпечення управління.

31. Основні поняття баз даних. Відомості про інформаційні системи. URL: <https://sites.google.com/view/ddkbmta-nfo/лекції/системи-керування-базами-даних-microsoft-access/основні-поняття-баз-даних> (дата звернення: 25.10.2021).

32. Medical information system (2016). Що таке eHealth. URL: <https://www.mcmed.ua/ua/ehealth> (дата звернення: 25.10.2021).

33. Titova, O., & Vysotskyi, D. (2021). The development of a subsystem for palliative patients information support.