

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 перший (бакалаврський)

РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ ПОШУКУ ТА
БРОНЮВАННЯ ЛІКАРСЬКИХ ЗАСОБІВ В АПТЕЧНІЙ МЕРЕЖІ
(тема)

Виконала:
студентка 4 курсу, групи ІТІНФ-18-2

Доренська 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 проф. Машталір С.В.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____
(підпис)

Кобилін О.А.
(прізвище, ініціали)

2022 р.

Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту
(повна назва)

Кафедра Інформатики
(повна назва)

Рівень вищої освіти перший (бакалаврський)

Спеціальність 122 Комп'ютерні науки
(код і повна назва)

Тип програми освітньо-професійна

Освітня програма Інформатика
(повна назва освітньої програми)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

«_____» _____ 2022 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Доренській Дар'ї Олегівні
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка інформаційної системи пошуку та бронювання лікарських засобів в аптечній мережі

затверджена наказом університету від 16 травня 2022 року № 541Ст

2. Термін подання студентом роботи до екзаменаційної комісії 30 травня 2022 р.

3. Вихідні дані до роботи: науково-методична та науково-технічна література, дані інтернет-мережі, CASE-засоби візуального проектування даних ERwin, середовище розробки програмних систем Intellij Idea, технологія доступу до даних ADO.NET, СУБД Microsoft SQL Server, інтегроване середовище SQL Server Management Studio.

4. Перелік питань, що потрібно опрацювати в роботі _____

1. Аналіз існуючих принципів моделювання та механізмів реалізації баз даних.

2. Моделювання структури бази даних інформаційної системи вибраної предметної області.

3. Моделювання та реалізація інформаційної системи пошуку та бронювання лікарських засобів в аптечній мережі.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) схема моделей даних за синтаксисом Чена, модель даних, згенерована в SQL Server, діаграма реалізації можливостей користувачів застосунку, скріншоти наповнених таблиць бази даних, скріншоти форм 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	18.04.2022	
2	Аналіз завдання, підбір літератури	18.04.22-21.04.22	
3	Аналіз літератури з досліджуваної проблеми	22.04.22-25.04.22	
4	Аналіз технічних засобів	26.04.22-30.04.22	
5	Розробка інформаційної системи вибраної предметної області	01.05.22-14.05.22	
6	Програмна реалізація	15.05.22-23.05.22	
7	Оформлення пояснювальної записки	24.05.22-28.05.22	
8	Перевірка на плагіат	30.05.22	
9	Рецензування	29.05.22	
10	Підготовка презентації та доповіді	29.05.22-01.06.22	
11	Занесення роботи в електронний архів	02.06.22	
12	Попередній захист кваліфікаційної роботи	08.06.22	

Дата видачі завдання 18 квітня 2022 р.

Студент _____
(підпис)

Керівник роботи _____ проф. Машталір С.В.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ/ABSTRACT

Пояснювальна записка до кваліфікаційної роботи: 42 с., 8 табл., 11 рис., 1 дод., 30 джерел.

ІНФОРМАЦІЙНА СИСТЕМА, СУБД, РЕЛЯЦІЙНА БАЗА ДАНИХ, НОРМАЛІЗАЦІЯ ДАНИХ, ER-МОДЕЛЬ, МОДЕЛЬ «СУТНІСТЬ – ЗВ’ЯЗОК», МОВА ЗАПИТІВ SQL, CASE-ЗАСІБ ERWIN, MS ACCESS, MS SQL SERVER, MS SQL SERVER MANAGEMENT STUDIO.

Об’єктом роботи є система пошуку лікарських засобів.

Метою роботи є моделювання та реалізація інформаційної системи пошуку та бронювання лікарських засобів в аптечній мережі.

Робота присвячена розробці інформаційної системи пошуку та бронювання лікарських засобів в аптечній мережі. Дана система зберігає інформацію про медичні засоби, які є в асортименті та наявності у різних аптеках, що знаходяться в певному радіусі від заданого геокоду користувача. Система взаємодіє з Google API для отримання актуальних геоданих.

Під час розробки інформаційної системи були використані: теорія проектування реляційних баз даних на базі будування ER-моделі; основні принципи нормалізації реляційних баз даних; мова запитів SQL; CASE-засіб візуального проектування даних ERWIN, середовище розробки програмних систем IntelliJ Idea, мова програмування Java 16, фреймворк для Java-платформи Spring.

INFORMATION SYSTEM, DBMS, RELATIVE DATABASE, NORMALIZATION OF DATA, ER-MODEL, MODEL “ENTITY – CONNECTION”, MS ACCESS, MS SQL SERVER, MS SQL SERVER MANAGEMENT STUDIO.

The object of the work is the search for medical supplies.

The goal of work is the modeling and implementation of an information system for the searching and booking of medicines in the pharmacy.

The work is dedicated to the development of the information system "Home Pharmacy". An information system provides information about medical conditions, which is in the range and availability of various pharmacies, which is within the radius of the given geocode of the customer. System interacts with Google API for obtaining up-to-date geo-data.

During development of the information system were used: basic principles of normalization of relational databases; SQL query language; CASE-tool for visual data design ERwin, IntelliJ Idea development environment for software systems, Java 16 language programming, framework for Spring Java-platform.

ЗМІСТ

Перелік умовних позначень, символів, одиниць, скорочень і термінів	7
Вступ.....	8
1 Аналіз принципів проєктування та реалізації інформаційних систем	9
1.1 Основні етапи життєвого циклу інформаційних систем та сучасні підходи до керування ІТ проєктами	9
1.2 Існуючі властивості та класифікація інформаційних систем.....	11
1.3 Існуючі властивості реляційних баз даних	11
1.4 Case-засоби моделювання структури БД.....	12
1.5 Принципи моделювання структури баз даних.....	14
1.6 Постановка задачі	15
2 Моделювання структури інформатичної системи пошуку та бронювання лікарських засобів в аптечній мережі	17
2.1 Специфікація вимог до інформаційної системи	17
2.2 Розробка бізнес-правил бази даних.....	17
2.3 Розробка концептуальної моделі.....	18
2.4 Функціональна структура	21
2.5 Побудова логічної моделі даних	22
2.6 Побудова моделі даних за допомогою CASE-засобу візуального проєктування FRWIN.....	28
2.7 Функціоне проєктування.....	28
3 Моделювання та реалізація інформатичної системи для пошуку та бронювання лікарських засобів в аптечній мережі	32
3.1 Вибір інструментальних засобів для реалізації	32
3.2 Етапи реалізації інформаційної системи для пошуку та бронювання лікарських засобів в аптечній мережі	33
3.3 Етапи реалізації інформатичної системи для пошуку та бронювання лікарських засобів в аптечній мережі	33

3.4 Перспективи подальшої роботи.....	34
Висновки	35
Перелік джерел посилання	36
Додаток А Ілюстрація роботи розробленої інформаційної системи	40

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

- API – Application Programming Interface (програмний інтерфейс програми)
- CASE – Computer-Aided Software Engineering (комп'ютерна інженерія програмного забезпечення)
- GUI – Graphical User Interface (графічний інтерфейс користувача)
- IDE – Integrated Development Environment (інтегроване середовище розробки)
- RDBMS – Relational Database Management System (система управління реляційними базами даних)
- SQL – Structured Query Language (структурована мова запитів)
- UML – Unified Modeling Language (уніфікована мова моделювання)
- ЖЦ – життєвий цикл
- ІС – інформаційна система
- СУБД – система управління базами даних

ВСТУП

Сучасний світ заснован в еру цифрових технологій, коли неможливо уявити існування без інформаційних застосунків, які займають одну з передових ролей у розвитку сучасної цивілізації. Одним з ключових аспектів цифрових технологій є зберігання та обробка інформації. Для цього надзвичайно зручно використовувати бази даних.

Бази даних значно полегшують роботу з інформацією: спрощують пошук та фільтрацію, виконують за лічені секунди завдання великого обсягу значно ефективніше ніж це зробила би людина, або напівавтоматична система. Дана структура використовується майже в усіх сферах діяльності людини. Бази даних є зручним та надійним способом зберігати та використовувати інформацію.

В рамках кваліфікаційної роботи було обрано реляційну модель бази даних, тому що такий спосіб організації інформації є найбільш наочним і доступним для розуміння користувачем, модель може бути зрозуміло представлена у інформаційній системі, що дозволяє легко маніпулювати даними: додавати, редагувати, видаляти, тощо.

Актуальність роботи полягає у тому, що згідно зі звітами фармацевтичних компаній, переважна більшість мешканців європейських країн та країн північною Америки користуються аптечними мережами. Великий попит на лікарські засоби формує потребу в якісній системі для пошуку та бронюванню лікарських засобів з метою заощадження коштів та часу.

1 АНАЛІЗ ПРИНЦИПІВ ПРОЄКТУВАННЯ ТА РЕАЛІЗАЦІЇ ІНФОРМАЦІЙНИХ СИСТЕМ

1.1 Основні етапи життєвого циклу інформаційних систем та сучасні підходи до керування ІТ проектами

Упродовж багатьох років життєвий цикл розробки програмного забезпечення, або скорочено ЖЦ, закріпився як процес, який допомагає створювати інформаційні системи, технічні проєкти та розробляти програмне забезпечення з нуля, охоплюючи ключові фази, до яких можна віднести планування, реалізацію та обслуговування системних рішень.

Життєвий цикл став критично важливим процесом завдяки стандартизованим етапам, які створюють баланс між витратами, якістю та часом, щоб задовольнити сучасні бізнес-потреби певної складності в певні терміни, з обмеженим бюджетом.

За галузевими стандартами інформаційна система – це поєднання апаратних засобів, програмного забезпечення та людських ресурсів, які виконують призначені завдання зі збору, обробки та відображення інформації. У цьому контексті ЖЦ допомагає системі реалізуватися. Важливо циклічно розвивати й експлуатувати інформаційні системи, оскільки вони постійно зазнають зміни умов. До розповсюджених оновлень відносяться: оновлення апаратного забезпечення, оновлення інструментів розробки програмного забезпечення, підвищення вимог користувачів або масштабування обсягу інформації в бізнес-середовищі та в області доменних моделей [1-3].

Життєвий цикл системи завжди складається з планування, реалізації та обслуговування. Але зазвичай, через ітераційні процеси та зміну бізнес-потреб долучають більше етапів, включаючи аналіз, проєктування, тестування та виведення з експлуатації або перехід до нової інформаційної системи. Усі

зазначені етапи стали вирішальними в ЖЦ та можуть використовуватися відповідно до природи кожного проєкту.

За даними опитувань вибірки зі ста ІТ-компаній, розташованих в США, для розробки програмного забезпечення найчастіше обирають методологію Agile, оскільки вона допомагає створювати продукти ітераційно та гнучко, щоб легко й ефективно орієнтуватися та керувати вимогами до інформаційних систем. Методологія Agile може працювати в гармонії з процесом ЖЦ, об'єднуючи етапи з відрізками ітерацій. Плануючи, команда складає «дорожню карту», яка визначає основні етапи проєкту з кроками, які будуть втілені для їх досягнення. Крім того, команда буде мати чітке розуміння строків проєкту, та термінів яких повинен дотримуватися кожен розробник. Найголовніше – команда отримає змогу збирати та розподіляти ресурси туди, де вони потрібні.

Спрощення розробки кінцевого продукту є найважливішою серед ключових практик розробки програмного забезпечення. З цією метою використання процесу ЖЦ значною мірою допомагає розділити та розбити об'ємні завдання на менші, більш керовані кроки, які легше оцінити та виконати. Завдяки чіткій структурі етапів, ті, хто бере участь у розробці, можуть допомогти сформуванню проєкту і керувати ним спрощено.

Життєвий цикл допомагає завчасно планувати та аналізувати структуровані етапи та цілі конкретного проєкту, для полегшення делегування та вирішування певних задач, орієнтованих на досягнення цілі. процеси адаптивні до мінливих потреб користувачів і вимог, саме тому ретельний план має велике значення у визначенні витрат і кадрових рішень. Це забезпечують чіткі цілі та результати, планові виміри ефективності та перевірку кожної фази життєвого циклу [4].

Етапи ЖЦ розроблені таким чином, щоб поступово вдосконалювати або змінювати систему протягом її розвитку. Якщо послідовно дотримуватися принципів проєктування інформаційних систем, вдасться реалізувати повністю оперативну, високоякісну систему, яка відповідає та/або перевищує вимоги клієнта в межах визначеного часу та бюджетних обмежень.

1.2 Існуючі властивості та класифікація інформаційних систем

Інформаційна система складається з п'яти важливих компонентів: апаратного забезпечення, програмного забезпечення, баз даних, інформаційної мережі та людей. Ці компоненти об'єднуються для виконання введення, обробки, виведення даних, зворотного зв'язку та керування системою.

Апаратне забезпечення складається з пристрою для зчитування даних та виведення інформації на екран, процесора, операційної системи та медіа-пристроїв. Програмне забезпечення реалізується завдяки іншому програмному забезпеченню і процедурам, які виконує розробник. База даних складається з даних, організованих у необхідній структурі. Мережа – поєднує в собі концентратори засобів зв'язку та мережеві пристрої. Люди являють собою операторів пристроїв, адміністраторів мережі та спеціалістів з системи.

Обробка інформації складається з введення; обробки даних, збереження інформації, виводу та контролю. На етапі введення даних інструкції надходять до системи, далі вони проходять декілька етапів обробки на етапі виконання програмами та інших запитів. На етапі виведення дані подаються у структурованому форматі та звітах [5-6].

1.3 Існуючі властивості реляційних баз даних

Для перевірки надійності бази даних потрібно правильно її спроектувати та створити. Системи баз даних розробляються таким чином, щоб відповідати набору властивостей, відомих як ACID. Атомарність, узгодженість, ізоляція та довговічність – це властивості, які становлять основу роботи з даними.

Атомарність гарантує, що жодна транзакція не буде зафіксована у системі частково: будуть або виконані всі її операції, або виконано жодної. Оскільки практично неможливо одночасно і атомарно виконати всю послідовність операцій усередині транзакції, вводиться поняття «відкату»: якщо транзакцію не

вдається повністю завершити, результати всіх її виконаних дій будуть скасовані і система повернеться до попереднього стану.

Транзакція, що досягає свого нормального завершення і тим самим фіксує свої результати, зберігає узгодженість бази даних. Іншими словами, кожна успішна транзакція за визначенням фіксує лише допустимі результати. Узгодженість є обширним поняттям. Наприклад, у банківській системі може бути вимога рівності суми, що списується з одного рахунку, сумі, що зараховується на інший. Це бізнес-правило, яке не може бути гарантоване лише перевірками цілісності, його повинні дотриматися програмісти при написанні коду транзакцій. Якщо яка-небудь транзакція здійснить списання, але не здійснить зарахування, то система залишиться в некоректному стані і якість узгодженості буде порушена.

Під час виконання транзакції паралельні операції не повинні впливати на результат. Ця вимога формує третій принцип ACID – ізолюваність, який на практиці вимагає надзвичайно багато ресурсів, тому в реальних БД існують режими, які тільки частково ізолюють транзакцію – вони називаються рівнями ізолюваності [7].

Четвертий принцип становить надійність. Незалежно від проблем на рівнях апаратного забезпечення, наприклад, знеструмлення системи або збоїв в обладнанні, зміни, зроблені успішно завершеною транзакцією, повинні залишитися збереженими після повернення системи в роботу. Іншими словами, якщо користувач отримав підтвердження від системи, що транзакція виконана, він може бути впевнений, що зроблені ним зміни не будуть скасовані через будь-який збій.

1.4 Case-засоби моделювання структури БД

Інструменти CASE для дизайну логічних баз даних – це частина етапу аналізу та створення ефективного програмного забезпечення для бізнес-

проектів, яка, в багатьох випадках, є найбільш складною та найважливішою в розробці інформаційної системи.

CASE-технологія являє собою методологію проектування ІС, а також набір інструментальних засобів, що дозволяють у наочній формі моделювати предметну область, аналізувати цю модель на всіх етапах розробки і супроводу ІС і розробляти вебзастосунки відповідно до інформаційних потреб користувачів. Більшість існуючих CASE-засобів засновано на методологіях, як правило, структурного або об'єктно-орієнтованого аналізу і проектування, що використовують специфікації у вигляді діаграм або текстів для опису зовнішніх вимог, зв'язків між моделями системи, динаміки поведінки системи та архітектури програмних засобів. Усі сучасні CASE-засоби можуть бути класифіковані по типах і категоріях. Класифікація за типами відображає функціональну орієнтацію CASE-засобів на ті або інші процеси ЖЦ. Класифікація за категоріями визначає міру інтегрованості виконуваних функцій і включає окремі локальні засоби, вирішення невеликих автономних завдань, набір частково інтегрованих засобів, що охоплюють більшість етапів життєвого циклу ІС і повністю інтегровані засоби, що підтримують увесь ЖЦ ІС і пов'язані загальним репозиторієм [8-9].

Окрім цього, CASE-засоби можна класифікувати за наступними ознаками:

- застосованим методологіям і моделям систем БД;
- мірою інтегрованості з СУБД;
- доступними платформам.

Класифікація за методологіями в основному співпадає з компонентним складом CASE-засобів і включає наступні основні типи:

- засоби аналізу Upper-CASE, призначені для побудови і аналізу моделей предметної області Design/IDEF Meta Software, BPwin Logic Works;
- засоби аналізу і проектування Middle-CASE, що підтримують найбільш поширені методології проектування, використовуються для створення проектних специфікацій Vantage Team Builder Cayenne, Designer/2000 ORACLE, Silverrun CSA, PRO-IV McDonnell Douglas, CASE. Аналітик Макро Проект.

Виходом таких засобів є специфікації компонентів і інтерфейсів системи, архітектура системи, алгоритми і структури даних;

- засоби проектування баз даних, що забезпечують моделювання даних і генерацію схем баз даних, як правило, на мові SQL для найбільш поширених СУБД. До них відносяться ERwin Logic Works, S-Designor SDP і DataBase Designer ORACLE. Засоби проектування баз даних є також у складі CASE-засобів: Vantage Team Builder, Designer/2000, Silverrun і PRO-IV;

- засоби розробки застосунків. До них відносяться засоби 4GL Uniface Compuware, JAM JYACC, PowerBuilder Sybase, Developer/2000 ORACLE, New Era Informix, SQL Windows Gupta, Delphi Borland і генератори кодів, що входять до складу Vantage Team Builder, PRO-IV і частково – в Silverrun;

- засоби реінжинірингу, що забезпечують аналіз програмного коду і схем баз даних, а також формування на їх основі різних моделей і проектних специфікацій. Засоби аналізу схем БД і формування ERD входять до складу Vantage Team Builder, PRO-IV, Silverrun, Designer/2000, ERwin і S-Designor. У області аналізу програмного коду найбільшого поширення набувають об'єктно-орієнтовані CASE-засоби, що забезпечують реінжиніринг програм на мові C++ , Rational Rose (Rational Software), Object Team.

1.5 Можливості мови SQL

Більшість сучасних реляційних СУБД використовують саме мову SQL. Американський інститут національних стандартів (American National Standards Institute – ANSI) та Міжнародна організація стандартів (International Standards Organization – ISO) займаються описом і підтримкою стандартів цієї мови. Усі сучасні СУБД підтримують певний стандарт, проте є й відхилення, які в кожному конкретному випадку специфікуються в документації програмного продукту. Окрім того, у багатьох системах розроблено розширення SQL, що надає змогу використовувати мову запитів у середовищі програмування [10].

SQL надає такі можливості:

- створювати та видаляти таблиці в базі даних, а також змінювати заголовки таблиць;
- вставляти, змінювати й видаляти рядки в таблицях;
- виконувати пошук даних у багатьох таблицях та впорядковувати результати цього пошуку;
- описувати процедури підтримки цілісності;
- визначати та оновлювати інформацію про захист даних.

1.6 Постановка задачі

Об'єктом роботи є пошук медичних засобів.

Метою роботи є моделювання та реалізація інформаційної системи пошуку та бронювання медичних засобів в аптечній мережі.

Нижче наведені опис предметної області, вимоги користувача до даних і функціоналу інформаційної системи:

- а) ІС повинна відображати меню для пошуку лікарських засобів;
- б) користувач програми повинен мати можливість:
 - 1) знайти медичний засіб в найближчих аптеках згідно до його геолокації;
 - 2) забронювати медичний засіб в аптечній мережі;
 - 3) порівняти ціни на медичний засіб в різних аптечних мережах;
 - 4) подивитися детальну інформацію про медичний засіб.

Редагувати данні має можливість адміністратор та модератор баз даних. Змінювати дані усіх таблиць, редагувати їх структуру, змінюючи типи, атрибути, тощо може тільки адміністратор .

Для досягнення мети необхідно вирішити наступні задачі:

- дослідити теорію проєктування реляційних БД на основі побудови ER-моделі, основ принципів нормалізації БД, принципів побудови запитів до БД (мови DQL SQL, DML SQL, DDL);
- дослідити інструментами проєктування і програмування ІС: CASE-засобів візуального проєктування даних ERwin, та середовище розробки IntelliJ Idea;
- дослідити вимоги до програмного забезпечення і вимоги до функціональності ІС;
- розробити бізнес-правила і глосарій;
- на основі бізнес-правил розробити ER-модель і відобразити її за допомогою ER-діаграми в синтаксисі Чена;
- розробити модель даних за допомогою CASE-засобу візуального проєктування даних ERwin;
- сформулювати структуру БД в СУБД;
- розробити дизайн інформаційної системи;
- розробити програмний код для забезпечення необхідної функціональності ІС;
- провести тестування ІС.

Для розробки ІС необхідно використовувати такі інструменти:

- ERwin – CASE-засіб проєктування БД;
- MS SQL Server – СУБД;
- IntelliJ Idea, Java 16, Spring – середа, мова та фреймворк для розробки програмного забезпечення.

2 МОДЕЛЮВАННЯ СТРУКТУРИ ІНФОРМАЦІЙНОЇ СИСТЕМИ ПОШУКУ ТА БРОНЮВАННЮ ЛІКАРСЬКИХ ЗАСОБІВ В АПТЕЧНІЙ МЕРЕЖІ

2.1 Специфікація вимог до інформаційної системи

Інформаційна система повинна бути розроблена на мові програмування JAVA з використанням реляційної моделі даних Hibernate в межах багатовимірної архітектури фреймворку Spring

Вебзастосунок має виконувати такі функції як: пошук, бронювання, оновлення та перегляд актуальної інформації стосовно ліків, їх складу, ціни, терміну придатності та медичних рецептів,

Цільова аудиторія застосування – люди віком від 14 років, які мають намір проводити моніторинг актуального становища ліків в аптеці, замовляти та бронювати медичні засоби з метою покращення стану свого здоров'я

Для повноцінної роботи з програмою користувачу потрібно мати персональний комп'ютер чи смартфон на якому буде доступ до мережі Інтернет, мишу та клавіатуру для вводу даних та взаємодії з додатком. Рекомендована операційна система – Windows 7 або новіша, Android 5 або новіша, IOS [11].

2.2 Розробка бізнес-правил бази даних

Інформаційна система пошуку та бронювання лікарських засобів в аптечній мережі займається урахуванням медичних засобів для застосування їх у лікуванні певних захворювань та допомагає відстежити ліки, в яких завершився чи наближається до завершення термін придатності. Вебзастосування зберігає інформацію щодо адреси та рейтингу обраної аптеки, містить дані стосовно актуальних цін.

Проаналізувавши предметну область інформаційної системи пошуку та бронювання лікарських засобів в аптечній мережі можна виділити такі бізнес-правила:

- назва аптеки унікальна в рамках інформаційної системи;
- виробник виготовляє багато ліків. Одні ліки виготовляє один виробник;
- ID ліків унікально в рамках інформаційної системи;
- дата завершення терміну придатності вказується працівником аптеки;
- ліки можуть зберігатися у ml (мілілітрах), mg (міліграмах) або amount (штуках);
- ліки можуть бути призначені тільки для дітей;
- користувач має надати право на обробку та використання його особистих даних;
- користувач має запланувати дату отримання ліків і зазначити її при бронюванні товару;
- рейтинг аптеки може бути: good (гарним), normal (нормальним), bad (поганим).

Проаналізувавши ці бізнес правила можна перейти до побудови функціональної структури даної предметної області.

2.3 Розробка концептуальної моделі

Концептуальна модель – це уявлення системи, яка використовує концепції та ідеї для формування зазначеного уявлення. Концептуальне моделювання використовується в багатьох областях, починаючи від природних наук і соціальної економіки і закінчуючи розробкою програмного забезпечення – де воно посіло суттєву роль.

Використовуючи концептуальну модель для представлення абстрактних ідей, важливо відрізнити модель концепції від моделі, яка є концептуальною. Тобто модель, по своїй суті, є річчю сама по собі, але вона також містить концепцію того, що представляє, тому концептуальні моделі часто використовуються як абстрактне уявлення об'єктів реального світу [12-14].

Протягом періоду появи і оновлення різноманітних реалізацій концептуальних моделей для програмного забезпечення з'явилося кілька найкращих практик. Концептуальна модель повинна:

- визначити перелік типів сутностей, інформація про які зберігатиметься у базі даних (табл. 2.1);
- бути доступною для всіх членів команди, щоб полегшити співпрацю;
- бути адаптивною, щоб безперервно відображати актуальну інформацію;
- містити як візуальні, так і письмові форми діаграм, щоб краще пояснити абстрактні поняття, які можуть бути представлені;
- встановити відповідні терміни та поняття, які будуть використовуватися протягом усього проекту;
- забезпечити базову структуру для сутностей проекту.

Таблиця 2.1 – Відомості про типи сутностей

Назва сутності	Опис	Особливості використання
1	2	3
Medecine	Медичний засіб	Містить інформацію про медичні засоби та ліки, які містяться в аптеці
Pharmacy	Аптека	Містить в собі інформацію про аптеки

Продовження таблиці 2.1

1	2	3
Search history	Історія пошуку	Містить в собі інформацію про користувача та ліки, які він запитував
User	Користувач	Містить інформацію про користувача
Form	Форма	Містить перелік форм та видів медичних засобів
Storage	Склад	Містить перелік наявних медичних засобів в аптеці

Наступним кроком встановлено типи зв'язків між обраними сутностями в моделі даних (табл. 2.2).

Таблиця 2.2 – Відомості про типи зв'язків між сутностями

Назва сутності	Назва сутності	Зв'язок
Medicine	Storage	1:M
Pharmacy	Storage	1:1
User	Search history	1:M
Medicine	Search history	1:M
Form	Medicine	1:M

На основі цих таблиць було побудовано концептуально модель даних у синтаксисі Чена (рис. 2.1).

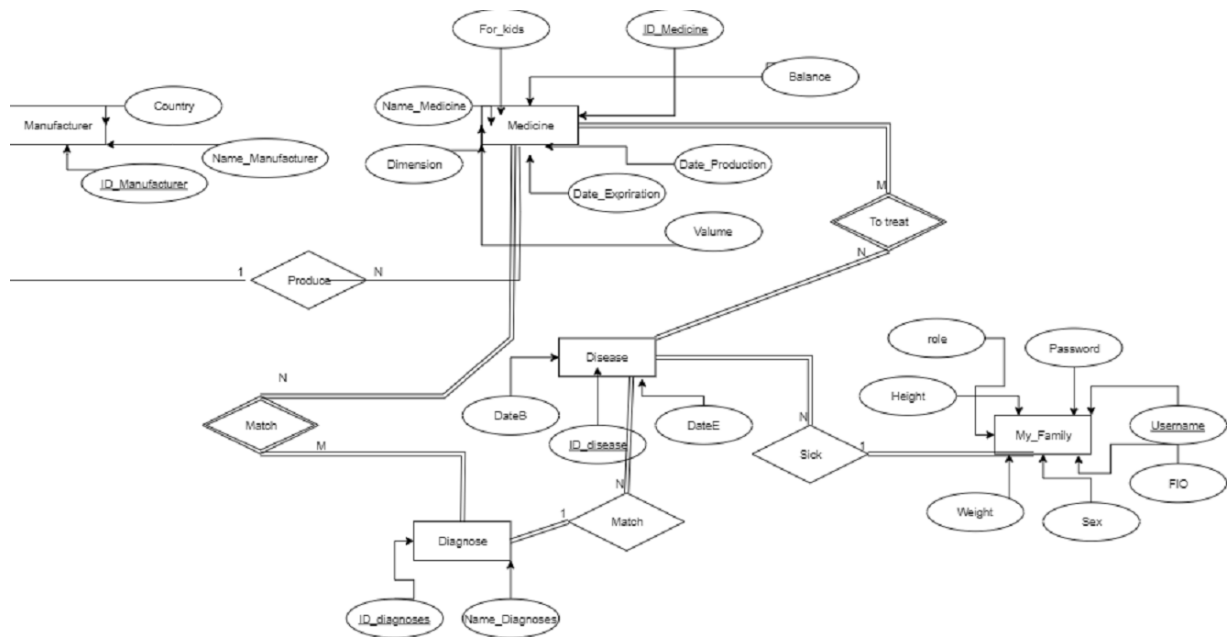


Рисунок 2.1 – Концептуальна модель даних

2.4 Функціональна структура

Для належного відображення функціональної структури доречно використати UseCase діаграму. UseCase діаграма – це модель, на якій зображено відношення між акторами та прецедентами (варіантами використання) в системі. Суть даної діаграми полягає в наступному: зпроектована система представляється у вигляді безлічі сутностей чи акторів, що взаємодіють із системою за допомогою так званих варіантів використання або прецедентів [15-17].

Прецедент – це модель можливості ІС, завдяки якій користувач може отримати конкретний, потрібний йому результат. Прецедент відповідає окремому сервісу системи, визначає один із варіантів її використання та типовий спосіб взаємодії користувача з системою.

Клієнт – це користувач, який може переглядати інформацію, оновлювати данні, додавати інформацію та видаляти данні в таблицях. Функціональність клієнта відображена на UseCase діаграмі (рис. 2.2).

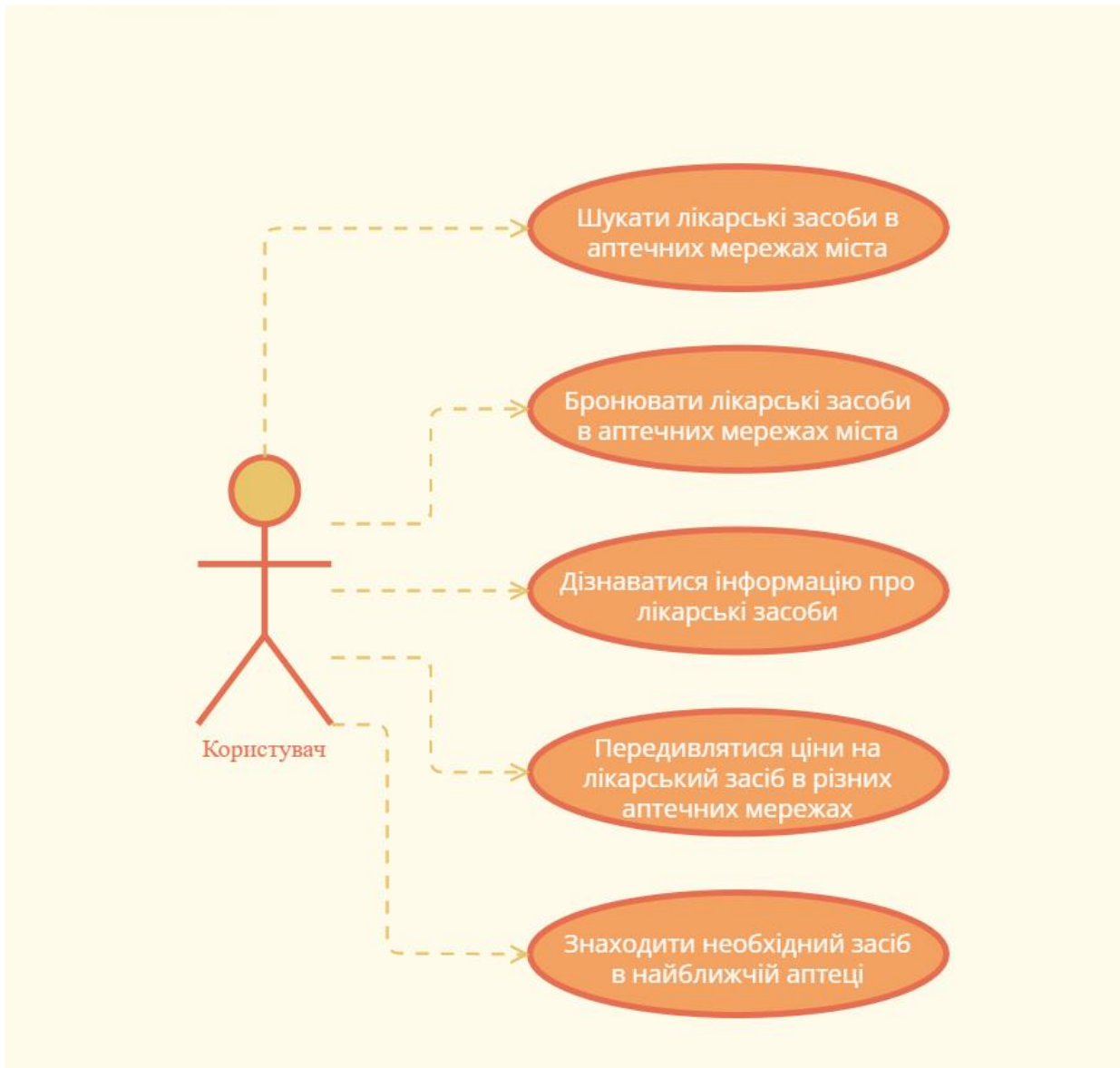


Рисунок 2.2 – Діаграма користувача

2.5 Побудова логічної моделі даних

Для переходу з концептуальної моделі даних до логічної потрібно створити транзитні слабкі сутності, які будуть поєднувати сутності, що мають зв'язок М:Н. Слабка сутність має обов'язково містити первинні ключі таблиць, між якими встановлюється зв'язок.

Логічні моделі даних повинні ґрунтуватися на структурах, визначених у попередній концептуальній моделі даних, оскільки вони описують семантику інформаційного контексту, яку логічна модель повинна також відображати.

Вміст логічної моделі даних коригується для досягнення певної ефективності [18].

Логічна модель є основою бази даних, вона повинна відображати взаємозв'язки між реляційними таблицями. На цьому етапі між сутностями можуть бути наступні типи зв'язків 1:1 (один до одного), 1:Б (один до багатьох) та Б:Б (багато до багатьох) [19].

Перевіривши концептуальну модель кваліфікаційної роботи було виявлено, що база даних не знаходиться в третій нормальній формі, згідно до визначення нормальних форм Бойса-Кодда:

- відношення знаходиться в 1НФ, якщо всі його атрибути містять атомарні значення, та всі використані домени містять тільки скалярні значення. Всі кортежі сутності мають бути унікальними;

- відношення знаходиться в 2НФ, якщо воно знаходиться в 1НФ і кожен не ключовий атрибут залежить від Первинного Ключа (ПК);

- відношення знаходиться в 3НФ, коли знаходиться у 2НФ і кожен не ключовий атрибут нетранзитивно залежить від первинного ключа [20, 21].

Після аналізу концептуальної моделі було модифіковано схему таким чином: зв'язок багато до багатьох модифіковано за допомогою додавання нових сутностей: для відношень Medicine і Pharmacy створено слабку сутність Storage, для відношень Medicine і User створено слабку сутність Search_history. Крім того атрибут Country було винесено до нової таблиці Address, з метою попередження введення помилкових даних.

Була побудована логічна модель у синтаксисі Чена (рис. 2.3).

У ER-діаграмі наявні 6 таблиць – «User», «Pharmacy», «Medicine», «Form», «Storage», «Search_history». Кожна з таблиць має свій список атрибутів. Зробимо розкриття для цих сутностей (табл. 2.3-2.8):

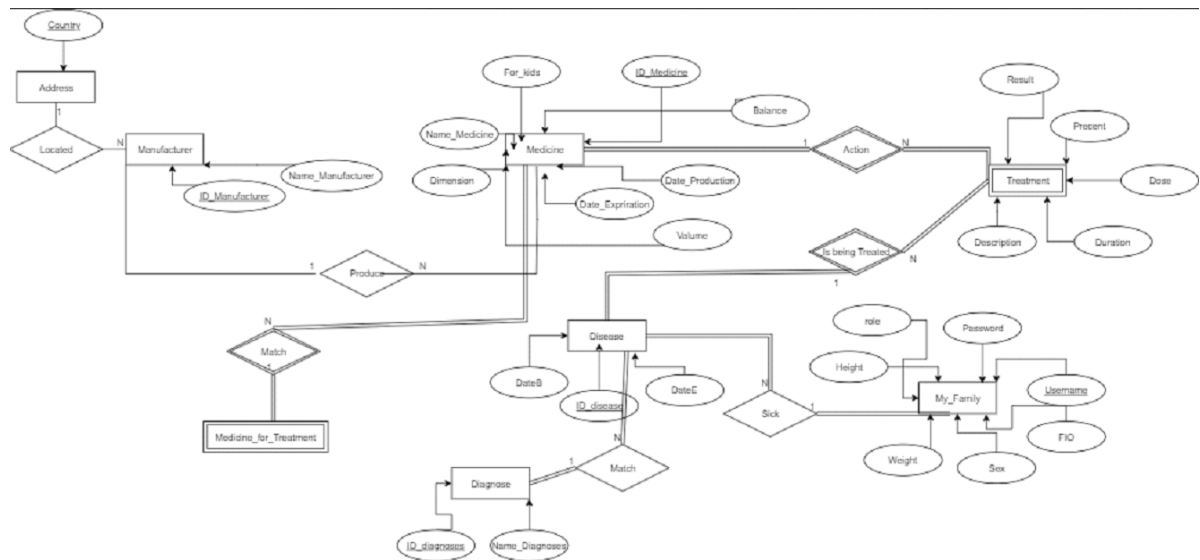


Рисунок 2.3 – Логічна модель даних

Таблиця 2.3 – Опис структури таблиці «User» («Користувач»)

Ключ	Ім'я поля	Тип даних	Розмір поля	Опис
PK	id_channel	int	4	Унікальний код чату користувача
	current_address	varchar	100	Поточна адреса користувача
FK	address_geo_code	varchar	20	Геокод адреси користувача

Таблиця 2.4 – Опис структури таблиці «Pharmacy» (Аптека)

Ключ	Ім'я поля	Тип даних	Розмір поля	Опис
1	2	3	4	5
PK	id	varchar	20	Ідентифікатор аптеки
	name	varchar	100	Назва аптеки
FK	address	varchar	100	Адреса аптеки
FK	address_geo_code	varchar	20	Геокод аптеки

Продовження таблиці 2.4

1	2	3	4	5
	start_work_at	time		Час початку роботи аптеки
	finish_work_at	time		Час завершення роботи аптеки

Таблиця 2.5 – Опис структури таблиці «Medecine» («Медичний засіб»)

Ключ	Ім'я поля	Тип даних	Розмір поля	Опис
PK	id	varchar	20	Унікальний ідентифікатор медичного засобу
	name	varchar	100	Назва медичного засобу
	instruction	varchar		Опис застосування медичного засобу
FK	form	varchar	20	Форма медичного засобу

Таблиця 2.6 – Опис структури таблиці «Storage» («Склад»)

Ключ	Ім'я поля	Тип даних	Розмір поля	Опис
PK	id	varchar	20	Ідентифікатор складу
FK	medicine_id	varchar	20	Ідентифікатор медичного засобу
FK	pharmacy_id	varchar	20	Ідентифікатор аптеки
	price	bigDecimal		Ціна за медичний засіб в аптеці
	amount	int		Кількість медичного засобу в аптеці в наявності

Таблиця 2.7 – Опис структури таблиці «Form» («Форма»)

Ключ	Ім'я поля	Тип даних	Розмір поля	Опис
PK	id	varchar	20	Ідентифікатор форми медичного засобу
	name	varchar	20	Назва форми медичного засобу

Таблиця 2.8 – Опис структури таблиці «Search_history» («Пошуковий запит»)

Ключ	Ім'я поля	Тип даних	Розмір поля	Опис
PK	id	varchar	20	Ідентифікатор пошукового запиту
FK	medicine_id	varchar	20	Ідентифікатор медичного засобу
FK	user_id	varchar		Ідентифікатор користувача
	search_at	time		Час пошукового запиту

Таким чином, в доменній моделі виділено 6 сутностей:

- User(id_channel, current_address, address_geo_code);
- Pharmacy(id, name, address, address_geo_code, start_work_at, finish_work_at);
- Medicine(id, name, instruction, form);
- Storage(id, medicine_id, pharmacy_id, price, amount);
- Form(id, name);
- Search_history(id, medicine_id, user_id, search_at).

2.6 Побудова моделі даних за допомогою CASE-засобу візуального проектування ERWIN

Для атрибутивного підходу моделювання ІС характерна наявність повної інформації про предметну область – об'єкти, їх атрибути і значення цих атрибутів [22, 23].

На основі концептуальної та логічної моделі побудуємо модель даних за допомогою програми візуального проектування ERWIN (рис. 2.4)

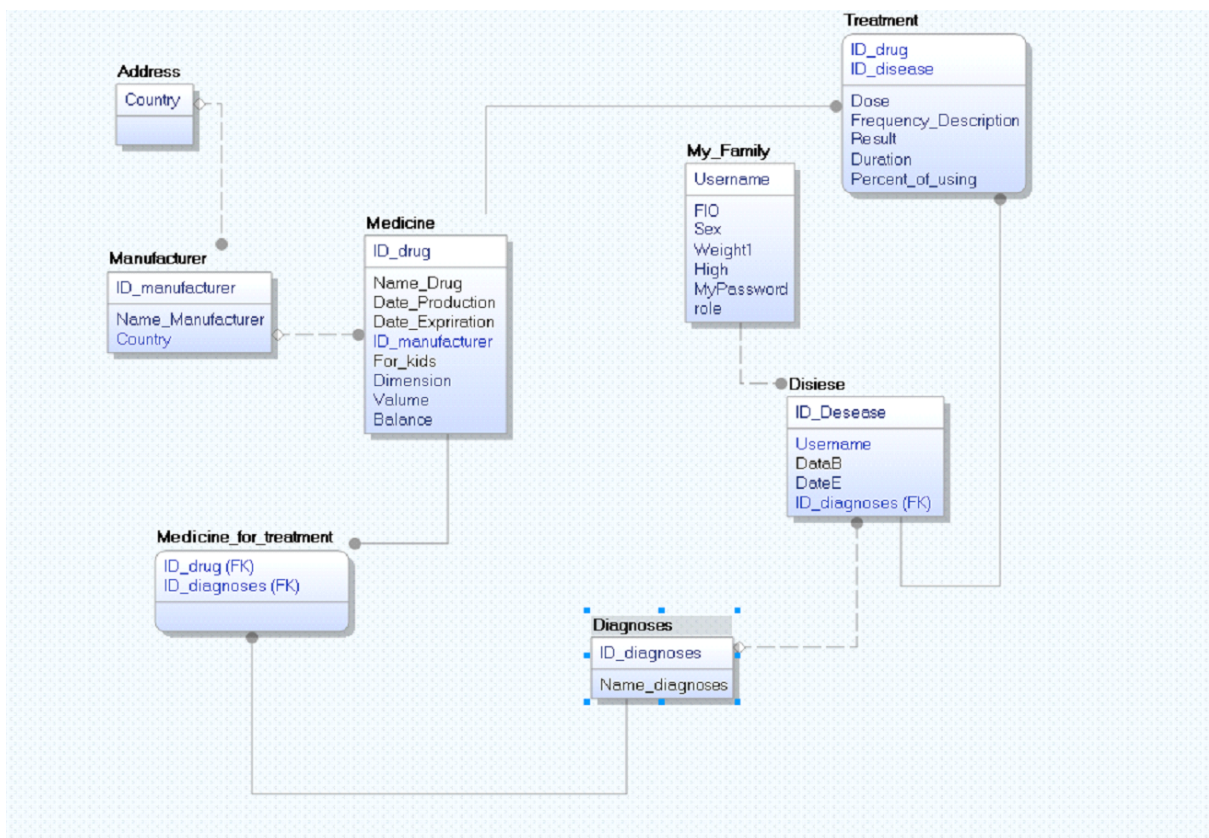


Рисунок 2.4 – Логічна модель в ERWIN

2.7 Фізичне проектування

Фізичне проектування бази даних - процес підготовки опису реалізації бази даних на вторинних запам'ятовуючих пристроях; на цьому етапі розглядаються основні відносини, організація файлів і індексів, призначених для забезпечення

ефективного доступу до даних, а також пов'язані з цим обмеження цілісності і засоби захисту інформації. Фізичне проектування є третім і останнім етапом створення бази даних, при виконанні якого проєктувальник приймає рішення про способи реалізації розробленої бази даних [24].

Результатом переносу моделі даних кваліфікаційної роботи з ERWIN у SQL Management Studio є діаграма, яка зображена на (рис. 2.5).

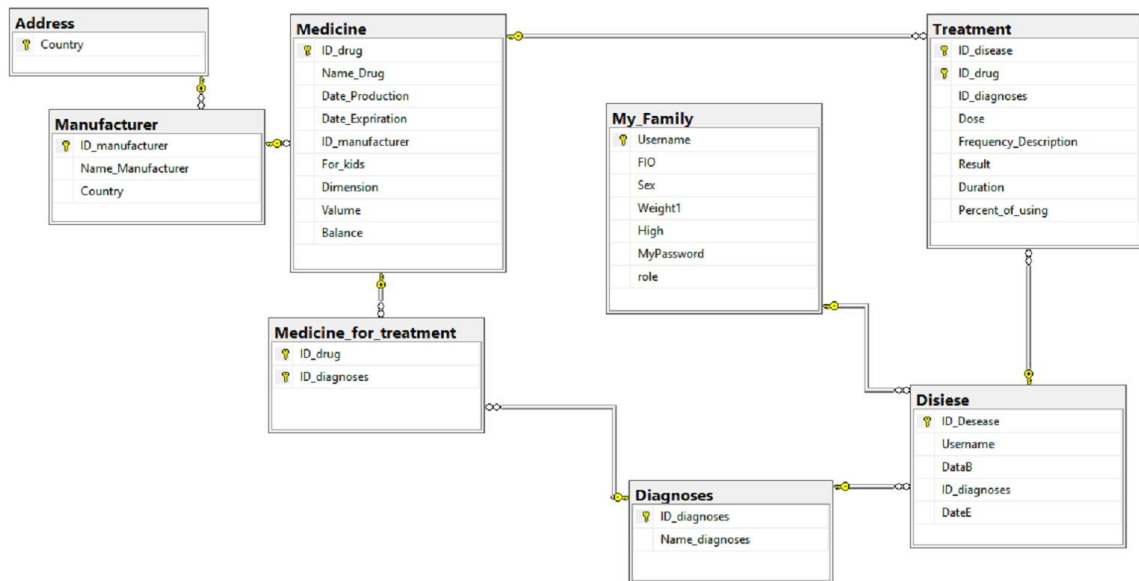


Рисунок 2.5 – Згенерована модель даних в SQL Server

Фізична структура бази даних для інформаційної системи пошуку та бронювання лікарських засобів в аптечній мережі наведена нижче за допомогою відповідних команд CREATE TABLE:

Лістинг 2.1 Створення таблиці «Аптека»:

```
create table pharmacies(
id uuid not null primary key,
name varchar not null,
address varchar unique not null,
address_geocode varchar,
start_work_at time not null,
finish_work_at time not null);
```

Лістинг 2.2 Створення таблиці «Користувач»:

```
create table users(
  chat_id bigint not null primary key,
  current_address varchar,
  current_address_geocode varchar);
```

Лістинг 2.3 Створення таблиці «Склад»:

```
create table medicine_in_pharmacy (
  medicine_id uuid not null,
  pharmacy_id uuid not null,
  price numeric(6,2) not null,
  number bigint not null,
  primary key (medicine_id, pharmacy_id),
  constraint medicine_in_pharmacy_medicine_fk foreign key (medicine_id)
  references medicines(id) on delete cascade,
  constraint medicine_in_pharmacy_pharmacy_fk foreign key (pharmacy_id)
  references pharmacies(id) on delete cascade);
```

Лістинг 2.4 Створення таблиці «Ліки»:

```
create table medicines (
  id uuid not null primary key ,
  name varchar not null,
  instruction text,
  form_id int not null,
  constraint medicine_forms foreign key (form_id)
  references forms(id) on delete cascade);
```

ЛІСТИНГ 2.5 Створення таблиці «Форма»:

```
create table forms(
id int not null primary key,
name varchar not null);
insert into forms (id, name) values (0, 'TABLET');
insert into forms(id, name) values (1, 'LIQUID');
insert into forms(id, name) values (2, 'CAPSULES');
insert into forms(id, name) values (3, 'TOPICAL');
insert into forms(id, name) values (4, 'INJECTION');
insert into forms(id, name) values (5, 'OTHER');
```

ЛІСТИНГ 2.6 Створення таблиці «Пошук»:

```
create table searches (
id serial primary key ,
medicine_id uuid not null,
user_id bigint not null,
searched_at timestamp default now(),
constraint search_medicine_fk foreign key (medicine_id)
references medicines(id),
constraint search_user_fk foreign key (user_id)
references users(chat_id) on delete cascade);
```

3 МОДЕЛЮВАННЯ ТА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ ПОШУКУ ТА БРОНЮВАННЮ ЛІКАРСЬКИХ ЗАСОБІВ В АПТЕЧНІЙ МЕРЕЖІ

3.1 Вибір інструментальних засобів для реалізації

Для розробки інформаційної системи було використано інтегроване середовище, яке зберігає в собі комплекс функцій програмного забезпечення для реалізації, тестування, налагоджування комп'ютерного коду [25-27].

Сучасне інтегроване середовище надає розробнику стек функціональних можливостей і надає такі інструменти як:

- виконання глибокого аналізу та створення віртуальної карти проєкту;
- автоматичне доповнення коду з огляду на контекст;
- проведення валідації (перевірка на відповідність стандартам коду);
- виконання перепроєктування коду;
- підтримка роботи зі вставками, написаними іншими мовами програмування;
- використання шаблонів для вставки фрагментів коду, що повторюються;
- рекомендація оптимізації даних за допомогою спеціальних інструментів, які аналізують продуктивність коду та оцінюють навантаження на процесор та оперативну пам'ять;
- рекомендація рішень тестування програмного коду;
- інструменти для проведення автоматичних тестів та формування аналітики, яка показує, який обсяг коду виконується;
- вбудований декомпілятор – інструмент для перетворення виконуваного двійкового на код, ораної мови програмування.

3.2 Етапи реалізації інформаційної системи для пошуку та бронювання лікарських засобів в аптечній мережі

Для початку використання вебзастосунку пошуку та бронювання лікарських засобів треба увійти до системи «Телеграм», знайти в пошуку pharm_ua_bot (рис. А.1) та скористатися командою за замовчуванням, щоб розпочати роботу з телеграм ботом (рис. А.2).

Основні можливості інформаційної системи приведені у додатках, використаних в програмі:

- перегляд найближчих аптек згідно з геолокацією користувача (рис. А.3, А.4);
- перегляд детальної інформації щодо медичних засобів (рис. А.5);
- порівняння цін на лікарські засоби (рис. А.6).

3.3 Тестування розробленої інформаційної системи та аналіз результатів

Тестування ІС проводилось завдяки реалізації модульного та функціонального тестування системи.

Модульне тестування забезпечило коректну роботу програми з технічної точки зору впродовж процесу розробки програмного коду.

Функціональне тестування склало більш складну частину процесу перевірки працездатності програми та забезпечило тестування роботи застосунку в поєднанні з прямим підключенням до бази даних.

В майбутньому планується проведення інтеграційного тестування для перевірки коректності роботи взаємопов'язаних систем [28-30].

3.4 Перспективи подальшої роботи

Перспективою подальшої роботи є удосконалення та розширення інформаційної системи, її масштабування та інтеграційне тестування.

В подальшому слід забезпечити зручний інтерфейс адміністрування даних в мережі інтернет – реалізувати вебсторінку для оновлення, додавання, видалення та регулювання даних, використовуючи інструменти гнучкого відображення даних.

Важлива задача буде полягати в налаштуванні серверів для різного навантаження системи. Для цього буде потрібно обрати хостинг і запровадити адміністративні заходи зручного користування інструментами налаштування серверів, реалізувати скрипти доступу до даних, налагодити конфігурацію файлів середовища.

Також слід подбати про масштабування системи. В майбутньому слід додати декілька мов та специфікацій для роботи інформаційної системи з пошуку та бронюванню медичних засобів у інших регіонах світу.

Важливим залишається тестування системи – запровадження нових технологій і методів тестування, удосконалення якості функціональних та модульних тестів.

ВИСНОВКИ

У рамках кваліфікаційної роботи було змодельовано та реалізовано інформаційну систему для пошуку та бронювання лікарських засобів.

Розробка електронної медичної інформаційної системи для швидкого пошуку лікарських засобів, згідно з дослідженнями фармацевтичних компаній, залишається актуальною для багатьох регіонів світу, через високий попит на ринку медичних засобів і зростаючий попит на користування вбудованими в глобальні кросплатформені системи застосунками.

Дана інформаційна система дозволяє переглядати список найближчих аптек, знаходити необхідні лікарські засоби, дізнаватися актуальну інформацію щодо рецепту та наявності ліків в найближчих аптеках, переглядати історію запитів.

ІС взаємопов'язана з зовнішніми системами, такими як Telegram API та Google API.

Під час тестування та подальшого аналізу застосунку був визначений функціонал, який потребує подальшого вдосконалення та опрацювання команди спеціалістів.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Bilonoh, B., Bodyanskiy, Y., Kolchygin, B., & Mashtalir, S. (2021, May). Tunable Activation Functions for Deep Neural Networks. In *International Scientific Conference “Intellectual Systems of Decision Making and Problem of Computational Intelligence”* (pp. 624-633). Springer, Cham.
2. Bilonoh, B., & Mashtalir, S. (2020, August). Parallel multi-head dot product attention for video summarization. In *2020 IEEE Third International Conference on Data Stream Mining & Processing (DSMP)* (pp. 158-162). IEEE.
3. Bodyanskiy, Y. V., Tyshchenko, O. K., & Mashtalir, S. V. (2019, June). Fuzzy clustering high-dimensional data using information weighting. In *International Conference on Artificial Intelligence and Soft Computing* (pp. 385-395). Springer, Cham.
4. Mashtalir, S., & Mashtalir, V. (2020). Spatio-temporal video segmentation. In *Advances in Spatio-Temporal Segmentation of Visual Data* (pp. 161-210). Springer, Cham.
5. Kinoshenko, D., Kobylin, O., Mashtalir, S., & Stolbovyi, M. (2019, March). Metric video retrieval speedup by irrelevant data elimination. In *Eleventh International Conference on Machine Vision (ICMV 2018)* (Vol. 11041, pp. 176-183). SPIE.
6. Mashtalir, S. V., Stolbovyi, M. I., & Yakovlev, S. V. (2019). Clustering video sequences by the method of harmonic k-means. *Cybernetics and Systems Analysis*, 55(2), 200-206.
7. Машталир, С. В., Столбовой, М. И., & Яковлев, С. В. (2019). Кластеризация последовательностей видеоданных на основе гармонических k-средних. *Кибернетика и системный анализ*.
8. Mashtalir, S. V., Stolbovoi, M. I., & Yakovlev, S. V. (2019). Hybrid Approach to Clustering Various Lengths Video. *Journal of Automation and Information Sciences*, 51(3).

9. Mashtalir, S., Mikhnova, O., & Stolbovyi, M. (2018, August). Sequence matching for content-based video retrieval. In *2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP)* (pp. 549-553). IEEE.
10. Mashtalir, S., Stolbovyi, M., & Mikhnova, O. (2019). Multidimensional sequence clustering with adaptive iterative dynamic time warping. *International Journal of Computing*, 18(1), 53-59.
11. Mashtalir, S., Mashtalir, V., & Stolbovyi, M. (2018, August). Representative based clustering of long multivariate sequences with different lengths. In *2018 IEEE second international conference on Data Stream Mining & Processing (DSMP)* (pp. 545-548). IEEE.
12. Kinoshenko, D., Mashtalir, S., Shlyakhov, V., & Stolbovyi, M. (2018, January). Video shots retrieval with use of pivot points. In *International Conference on Computer Science, Engineering and Education Applications* (pp. 102-111). Springer, Cham.
13. Information Resources Management Association (Ed.). (2018). *Computer Vision: Concepts, Methodologies, Tools, and Applications*. IGI Global.
14. Машталир, С. В., & Столбовой, М. И. (2018). Адаптивные матричные модели в задаче контроля потоков видео. *Радіоелектроніка, інформатика, управління*, (4 (47)), 188-194.
15. Богучарский, С. И., Машталир, С. В., & Столбовой, М. И. (2018). Быстрое обнаружение изменения свойств многомерных временных рядов на основе идентификационного подхода к ансамблю моделей. *Системи обробки інформації*, (3), 74-78.
16. Mashtalir, S. V., & Stolbovyi, M. I. (2018). ADAPTIVE MATRIX MODELS IN THE VIDEO STREAMS CONTROL PROBLEM. *RADIO ELECTRONICS COMPUTER SCIENCE CONTROL*, (4), 188-194.
17. Hu, Z., Mashtalir, S. V., Tyshchenko, O. K., & Stolbovyi, M. I. (2018). Clustering matrix sequences based on the iterative dynamic time deformation procedure. *International Journal of Intelligent Systems and Applications*, 10(7), 66-73.

18. Hu, Z., Mashtalir, S. V., Tyshchenko, O. K., & Stolbovyi, M. I. (2017). Video shots' matching via various length of multidimensional time sequences. *International Journal of Intelligent Systems and Applications*, 9(11), 10.
19. Oleg, K., Sergii, M., & Mykhailo, S. (2017, October). Video clustering via multidimensional time-series analysis. In *Proceedings of the 9th International Conference on Information Management and Engineering* (pp. 60-63).
20. Yarahmadi, A., Creemers, M., Qabbaah, H., Vanhoof, K., Palagin, O., Romanov, V., ... & Kurkchiyan, V. (2017). Unraveling Bi-Lingual Multi-Feature Based Text Classification: A Case Study. *International Journal of Information Theories and Applications*, 24(4), 3-18.
21. Mashtalir, S., Mashtalir, V., & Stolbovyi, M. (2017). Video shot boundary detection via sequential clustering. *International Journal "Information Theories and Applications*, 24(1), 50-59.
22. Mashtalir, S., & Mikhnova, O. (2017). Detecting significant changes in image sequences. In *Multimedia Forensics and Security* (pp. 161-191). Springer, Cham.
23. Gaulton, A., Hersey, A., Nowotka, M., Bento, A. P., Chambers, J., Mendez, D., ... & Leach, A. R. (2017). The ChEMBL database in 2017. *Nucleic acids research*, 45(D1), D945-D954.
24. Castano, S., Fugini, M., Martella, G., & Samarati, P. (1995). *Database Securi*. Addison—Wesley.
25. Yeo, K. T. (2002). Critical failure factors in information system projects. *International journal of project management*, 20(3), 241-246.
26. Sonderegger, T., Berger, M., Alvarenga, R., Bach, V., Cimprich, A., Dewulf, J., ... & Young, S. B. (2020). Mineral resources in life cycle impact assessment—part I: a critical review of existing methods. *The International Journal of Life Cycle Assessment*, 25(4), 784-797.
27. San Cristóbal, J. R., Carral, L., Diaz, E., Fraguera, J. A., & Iglesias, G. (2018). Complexity and project management: A general overview. *Complexity*, 2018.

28. Li, H., Xie, M., Wang, Y., Yang, L., Xie, Z., & Wang, H. (2021). riboCIRC: a comprehensive database of translatable circRNAs. *Genome biology*, 22(1), 1-11.
29. Rogers, R. (2020). Deplatforming: Following extreme Internet celebrities to Telegram and alternative social media. *European Journal of Communication*, 35(3), 213-229.
30. Chetthamrongchai, P., & JERMSITTIPARSERT, K. (2020). The Impact of Artificial Intelligence Outcomes on the Performance of Pharmacy Business in Thailand. *Systematic Reviews in Pharmacy*, 11(1).