

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Сумський державний університет
Центр заочної, дистанційної та вечірньої форм навчання
Кафедра інформаційних технологій

«До захисту допущено»

В.о. завідувача кафедри

_____ Світлана ВАЩЕНКО

_____ 2023 р.

КВАЛІФІКАЦІЙНА РОБОТА
на здобуття освітнього ступеня магістр

зі спеціальності 122 «Комп'ютерні науки»,
освітньо-професійної програми «Інформаційні технології проектування»
на тему: «Web-орієнтована система підтримки діяльності офтальмологічної клініки»

Здобувачки групи ІТ.мдн-21к Дюхіної Наталії Іллівни
(шифр групи) (прізвище, ім'я, по батькові)

Кваліфікаційна робота містить результати власних досліджень.
Використання ідей, результатів і текстів інших авторів мають посилання на відповідне джерело.

(підпис) Наталія ДЮХІНА
(Ім'я та ПРІЗВИЩЕ здобувача)

Керівник доцент кафедри ІТ, к.т.н., доцент Володимир НАГОРНИЙ
(посада, науковий ступінь, вчене звання, Ім'я та ПРІЗВИЩЕ) (підпис)

Сумський державний університет
Центр заочної, дистанційної та вечірньої форм навчання
Кафедра інформаційних технологій
Спеціальність 122 «Комп'ютерні науки»
Освітньо-професійна програма «Інформаційні технології проектування»

ЗАТВЕРДЖУЮ

В.о. завідувача кафедри ІТ

Світлана ВАЩЕНКО

«_____» _____ 2023 р.

ЗАВДАННЯ

на кваліфікаційну роботу магістра студентів

Дюхіній Наталії Іллівні

(прізвище, ім'я, по батькові)

1 Тема кваліфікаційної роботи «Web-орієнтована система підтримки діяльності офтальмологічної клініки»

затверджена наказом по університету від «08» листопада 2023 р. № 1249-VI

2 Термін здачі студентом кваліфікаційної роботи « 11 » __ грудня __ 2023 р.

3 Вхідні дані до кваліфікаційної роботи реєстрація пацієнтів, історія відвідин офтальмолога, збережені рецепти лінз, модуль підбору оправ з віртуальною приміркою оправ на фото пацієнта

4 Зміст розрахунково-пояснювальної записки (перелік питань, що їй належить розробити)

1) Аналіз предметної області

2) Постановка мети і задачі

3) Моделювання та проектування

4) Практична реалізація

5 Перелік графічного матеріалу (з точним зазначенням обов'язкових слайдів презентації) контекстна діаграма процесу розробки, діаграма декомпозиції процесу розробки, діаграма варіантів розробки прототипу, розробка модулю, публікація сайту в мережі інтернет

6. Консультанти випускної роботи із зазначенням розділів, що їх стосуються:

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв

Дата видачі завдання _____.

Керівник _____
(підпис)

Завдання прийняв до виконання _____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів кваліфікаційної роботи	Термін виконання етапів роботи	Примітка
1	Дослідження предметної області	до 19.09.23	
2	Формування мети і задач	до 19.09.23	
3	Аналіз аналогів та проблем використання	до 27.09.23	
4	Виявлення вимог до проекту	до 02.10.23	
5	Планування робіт	до 06.10.23	
6	Вибір засобів реалізації	до 09.10.23	
7	Проектування інформаційної системи	до 19.10.23	
8	Розробка інформаційної системи	до 09.11.23	
9	Тестування та завершення роботи	до 13.11.23	
10	Оформлення пояснювальної записки	до 01.12.23	

Магістрант _____ Наталія ДЮХІНА

Керівник роботи _____ к.т.н., доц. Володимир НАГОРНИЙ

АНОТАЦІЯ

Тема кваліфікаційної роботи магістра «Web-орієнтована система підтримки діяльності офтальмологічної клініки».

Пояснювальна записка складається зі вступу, 4 розділів, висновків, списку використаних джерел із 43 найменувань, додатків. Загальний обсяг роботи – 82 сторінки, у тому числі 43 сторінок основного тексту, 4 сторінки списку використаних джерел, 29 сторінок додатків.

Актуальність роботи полягає в швидких темпах розвитку комп'ютерних технологій і автоматизації різноманітних процесів усіх сфер життя. Робота та організація процесів діяльності офтальмологічних клінік не є виключенням, оскільки створення інформаційних web-орієнтованих систем даного напрямку дозволить підвищити якість обслуговування пацієнтів клініками.

Мета роботи: полягає у розробці web-орієнтованої системи підтримки діяльності офтальмологічної клініки.

Для дослідження було обрано емпіричні та експериментальні методи дослідження, результати яких показали, що більшість сучасних web-додатків обмежені та орієнтовані на комерційну діяльність інтернет-магазинів. Тому було прийнято рішення розробити функціональну систему підтримки діяльності клініки з обліком пацієнтів, їх історії захворювання, рекомендацій лікарів та функцією підбору оправи під власний формат обличчя.

Результати роботи опубліковано на Всеукраїнській науково-практичній конференції: «Крок у науку: дослідження у галузі природничоматематичних дисциплін та методик їх навчання», 7 грудня 2023 року, м. Чернігів на тему: «Web-орієнтована система підтримки діяльності офтальмологічної клініки».

Ключові слова: WEB-ОРІЄНТОВАНА СИСТЕМА, IDEF0, КЛІНІКА, ПАЦІЄНТ, БАЗА ДАНИХ, ІСТОРІЯ ХВОРОБ, РЕЦЕПТИ, ПІДБІР ОПРАВИ.

ЗМІСТ

ВСТУП.....	4
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	6
1.1 Огляд останніх досліджень і публікацій	6
1.2 Аналіз програмних продуктів – аналогів	8
2 ПОСТАНОВКА ЗАДАЧІ ТА МЕТОДИ ДОСЛІДЖЕННЯ.....	15
2.1 Мета та задачі дослідження	15
2.2 Методи дослідження	16
3 МОДЕЛЮВАННЯ ТА ПРОЕКТУВАННЯ.....	21
3.1 Моделювання процесу роботи.....	21
3.1.1 Моделювання процесу роботи у нотації IDEF	21
3.1.2 Модель варіантів використання.....	23
3.2 Проектування web-орієнтованої системи	28
3.3 Проектування моделі бази даних.....	32
4 ПРОГРАМНА РЕАЛІЗАЦІЯ.....	35
4.1 Архітектура web-орієнтованої системи.....	35
4.2 Реалізація програмного коду.....	37
4.3 Використання web-системи.....	43
ВИСНОВКИ.....	48
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	49
ДОДАТОК А. ПЛАНУВАННЯ РОБІТ.....	53
ДОДАТОК Б. ЛІСТИНГ ПРОГРАМНОГО КОДУ	66

ВСТУП

В Україні активно проводиться медична реформа, основою якої є діджиталізація процесу діяльності медичних закладів та установ з метою поліпшення швидкості та рівня обслуговування пацієнтів. Діяльність офтальмологічних клінік не є виключенням. Тому розробка web-орієнтованої системи для підтримки діяльності офтальмологічної клініки є актуальною задачею в сучасному світі.[1-3]

Підходи використані в web-системі дозволять підвищити ефективність ведення медичних облікових записів пацієнтів, запис до лікаря, ведення історії рецептів та зручність підбору оправ з примірюванням на власне фото. Дана web-орієнтована система офтальмологічної клініки дозволить оперативно отримувати доступ до історії пацієнта, аналізувати дані та вносити зміни в режимі реального часу.[4-8] Впровадження подібних систем у роботу офтальмологічної клініки допоможе підвищити її привабливість для обслуговування клієнтів.

Об'єктом дослідження є підтримка діяльності офтальмологічної клініки.

Предметом дослідження є web-орієнтована система для підтримки діяльності офтальмологічної клініки.

Мета роботи полягає у розробці web-орієнтованої системи підтримки діяльності офтальмологічної клініки.

Для виконання кваліфікаційної роботи, необхідно виконати наступні завдання:

- проаналізувати предметну область та сформулювати актуальність роботи;
- проаналізувати існуючі рішення web-додатків офтальмологічних клінік;
- визначити функціональні та нефункціональні вимоги до web-орієнтованої системи;
- провести моделювання процесів, що здійснюватимуться системою;
- розробити програмне рішення у вигляді web-орієнтованої системи.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Огляд останніх досліджень і публікацій

Поняття дистанційної медицини не нове, і широке поширення його в усіх сферах медицини дозволила наявність доступних апаратних і програмних засобів. Дистанційна консультація в режимі реального часу є найбільш ефективним способом надання медичної допомоги, але вона вимагає наявності у лікаря та пацієнта відповідного обладнання. Також мобільна медицина дозволяє пацієнтам отримувати доступ до медичної інформації та послуг у будь-який час і в будь-якому місці.[1-7]

Офтальмологія також розвивається швидкими темпами і обслуговування клієнтів переходить на новий рівень, а web-системи, що автоматизують частину процесів дозволяють конкурувати клінікам офтальмологічного напрямку діяльності.

Обслуговуючих web-орієнтованих систем підтримки діяльності офтальмологічних клінік є велике різноманіття, від простого онлайн запису до лікаря до популярних інтернет-сервісів підбору аксесуарів, що призвело до потреби створення інструментів для онлайн-примірок [2]. Один із таких сервісів, популярний серед інтернет-магазинів, є Fitting Box [3]. Ця програма використовує спеціальний сканер StudioBox для цифрових фотографій окулярів. Після цього вона моделює та візуалізує їх зображення. Отриману 3D-модель можна накласти на завантажену фотографію чи відео з web-камери в режимі реального часу. Для правильної примірки, нейронна мережа розпізнає обличчя користувача і шукає, де саме повинні бути розміщені окуляри. Під час обертання голови користувача в режимі відео-примірки також обертається 3D-модель окулярів, щоб максимально наблизити прикладку до реальності. [3]

Переважаюча технологія Fitting Box є її адаптивністю до різних інтернет-магазинів, що пропонують окуляри. Кожен магазин може розрахувати свій

асортимент окулярів для використання в цій системі. Крім того, сервіс дає можливість реального часу для примірки, але досить комерційно орієнтований і переважно інтернет-магазинами для продажу окулярів.[3]

Аналогами таких розробок є велика різноманітність українських web-сервісів, які дозволяють виконувати функції вибору оправы, підбору лінз, також можливе ведення особистого кабінету користувача. [4]

«Офтальмологічні клініки із використанням подібних систем можуть відкрити для себе нові можливості, які дозволять:

- автоматизувати прийом та обробку звернень в клініку. За інтеграції IP-телефонії процес стає ще більш зручним та швидким для адміністраторів;
- ведення прийому, з можливістю внесення діагнозів, направлень та призначень;
- здійснювати інтеграцію з офтальмологічним обладнанням;
- підключення до програми ПРРО для роботи каси;
- направлення, призначення, електронні рецепти - все це можна зробити доступним в web-додатках.» [2]

Крім того, такі системи регулярно оновлюються та доповнюються новими сучасними функціями з метою забезпечення діяльності лікарів офтальмологічних клінік. [5]

«Хмарне рішення програми web-сервісів дозволяє організувати ефективний робочий простір для усього персоналу офтальмологічних клінік. Понад 15 основних модулів реалізовано в програмі, більше 36 додаткових, постійні оновлення і допрацювання програмних можливостей (як за вимогами eHealth, так і згідно побажанням наших клієнтів, для оптимізації робочих процесів і зручності роботи із системою).» [2]

1.2 Аналіз програмних продуктів – аналогів

Під час виконання кваліфікаційної роботи було переглянуто велику кількість типових додатків та програм, для детального аналізу з цього спектру обрано два сервіси з найбільш схожою тематикою.[5-17]

Першим розглянемо сайт alensa.ua представлений на рисунку 1.1.

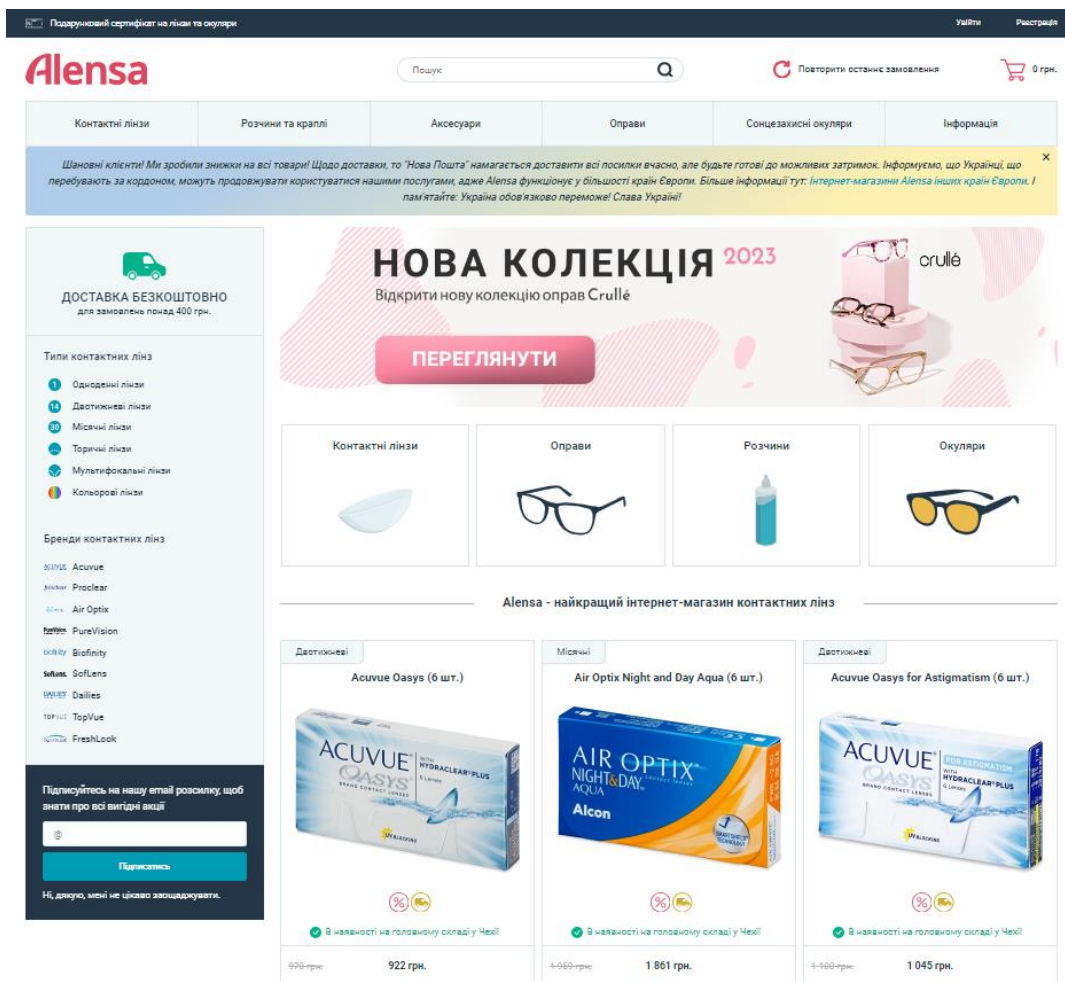


Рисунок 1.1 – Головна сторінка сайту alensa.ua

Джерело: [5]

Даний сайт – це лише один з різновидів сервісів такого типу. Сервіс надає можливість підбору контактних лінз, розчинів та крапель, акcesуарів, оправ, сонцезахисних окулярів.

Приклад вікна для підбору розчинів та крапель показано на рисунку 1.2. в даному вікні можна налаштувати фільтри:

- бренд;
- тип розчину;
- об'єм розчину;
- містить;
- виробник.

The screenshot shows the Alensa website interface. At the top, there is a search bar and navigation tabs for 'Контактні лінзи', 'Розчини та краплі', 'Акcesуари', 'Оправи', 'Сонцезахисні окуляри', and 'Інформація'. A sidebar on the left contains filters for 'Бренд', 'Тип розчину', 'Знайти окуляри', 'Об'єм розчину', 'Містить', and 'Виробник'. The main content area is titled 'Розчини для контактних лінз та краплі для очей' and displays a grid of product cards. Each card includes a product image, name, and price. The products shown are:

- Розчин ReNu MultiPlus 360 ml (355 грн)
- Розчин OPTI-FREE Express 2 x 355 ml (702 грн)
- Розчин ReNu MPS Sensitive Eyes 360 ml (295 грн)
- Розчин OPTI-FREE PureMoist 300 ml
- Розчин ReNu MPS Sensitive Eyes 2 x 360 ml
- Розчин OPTI-FREE Express 355 ml

Рисунок 1.2 –Сторінка сайту alensa.ua для підбору розчинів та крапель

Джерело: [5]

Основною специфікою даного сервісу є аналог інтернет-магазину, але орієнтованого на офтальмологічні ліки та акcesуари. Цікавим аспектом даного

сервісу є функція підбору окулярів, приклад сторінки підбору окулярів подано на рисунку 1.3.

Плюсами даного сервісу є можливість підбору оправ по багатьох пунктах:

- для кого призначені;
- виробник;
- тип оправы;
- форма лінз окулярів;
- колір оправы;
- основний матеріал окулярів;
- оправы з іншими функціями;
- ширина оправы;
- ширина лінзи;
- віртуальна примірка.

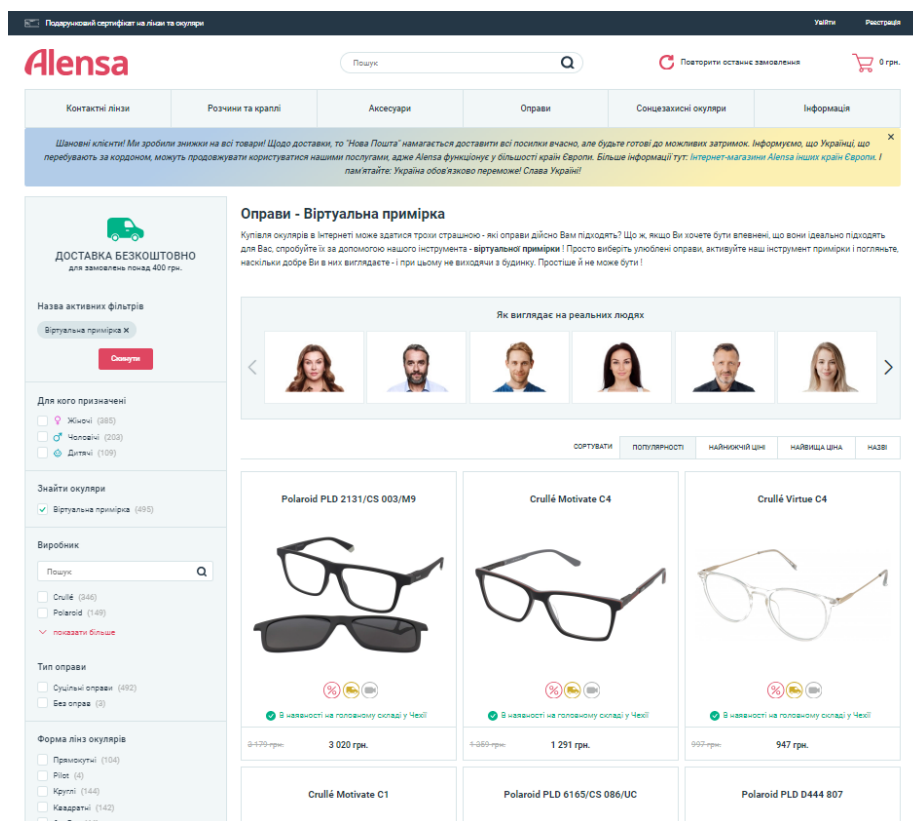


Рисунок 1.3 –Сторінка сайту alensa.ua для вибору окулярів

Джерело: [5]

Недоліком даного сервісу є відсутність можливості примірки окулярів на власне зображення, також вона має ряд обмежень по клієнтському самообслуговуванню і великій подібності до інтернет-магазину.

Іншим видом онлайн-сервісу є Люксоптика <https://luxoptica.ua/ua/virtual-try-on/> [6].

Даний варіант web-орієнтованої системи відрізняється дизайном, кольоровою палітрою, активною панеллю та можливим функціоналом. Оскільки тут активно підтримується функціонал підбору оправ та типу окулярів, але немає підтримки особистого кабінету користувача. Зовнішній вигляд початкової сторінки представлено на рисунку 1.4.

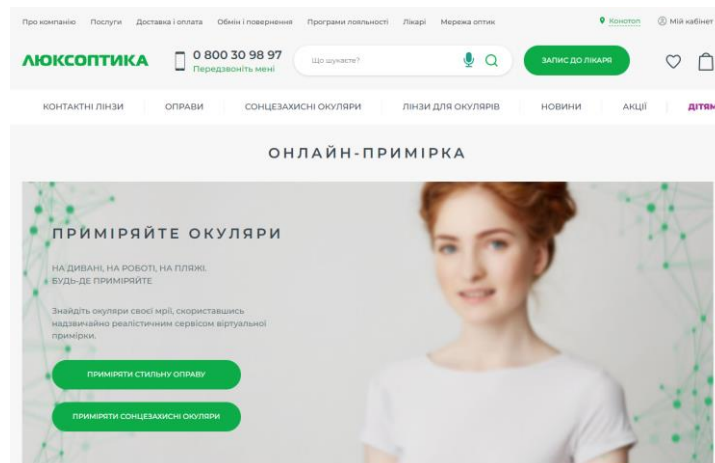


Рисунок 1.4 – Головна панель інформаційної web-сторінки «Люксоптика»

Джерело: [6]

Зовнішній вигляд функціональної сторінки web-системи представлено на рисунку 1.5. Операція підбору оправ окулярів складається з п'яти кроків:

- подивитися в камеру. Після чого з'являється підказка, яка допомагає записати відео селфі свого обличчя, завдовжки п'ять секунд. Для зчитування особистих біологічних характеристик вашого обличчя;

- слідкуючи за промінчиком, повертаємо голову вправо, з метою врахування всіх параметрів обличчя для підбору окулярів он-лайн;

- слідкуючи за промінчиком, потрібно повернути голову ліворуч, щоб зчитати параметри лівого боку обличчя;
- взяти банківську карту та прикласти до чола. Ця складова дозволяє врахувати пропорції обличчя відносно картки з метою вірного підбору розміру оправы;
- складається 3d-модель обличчя з приміряними окулярами, результат цієї примірки доступний користувачу.

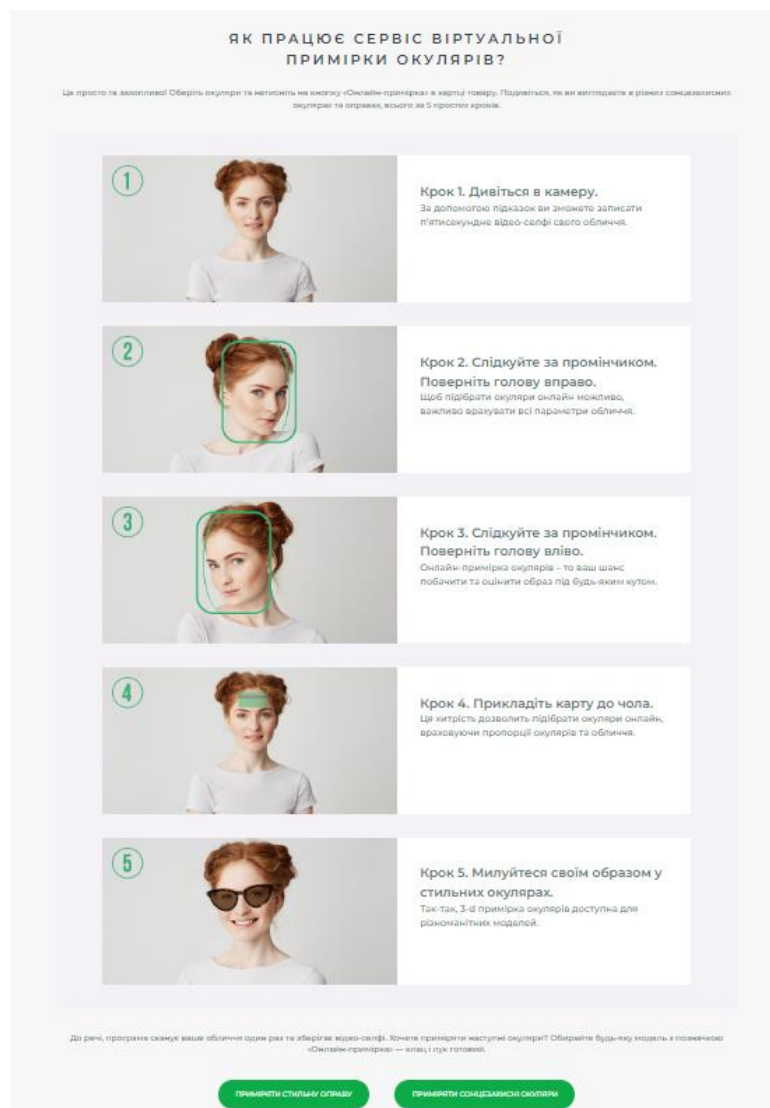


Рисунок 1.5 – Етапи роботи сервісу віртуальної примірки окулярів

Джерело: [6]

На наступному етапі обираємо дизайн оправы для примірки (рис. 1.6) при цьому є можливість налаштувати наступні фільтри:

- стать;
- бренд;
- колекцію;
- знижки та акції;
- форма оправы;
- ціна;
- тип оправы;
- матеріал оправы;
- довжина заушника;
- **МОЖЛИВІ ЗНИЖКИ.**

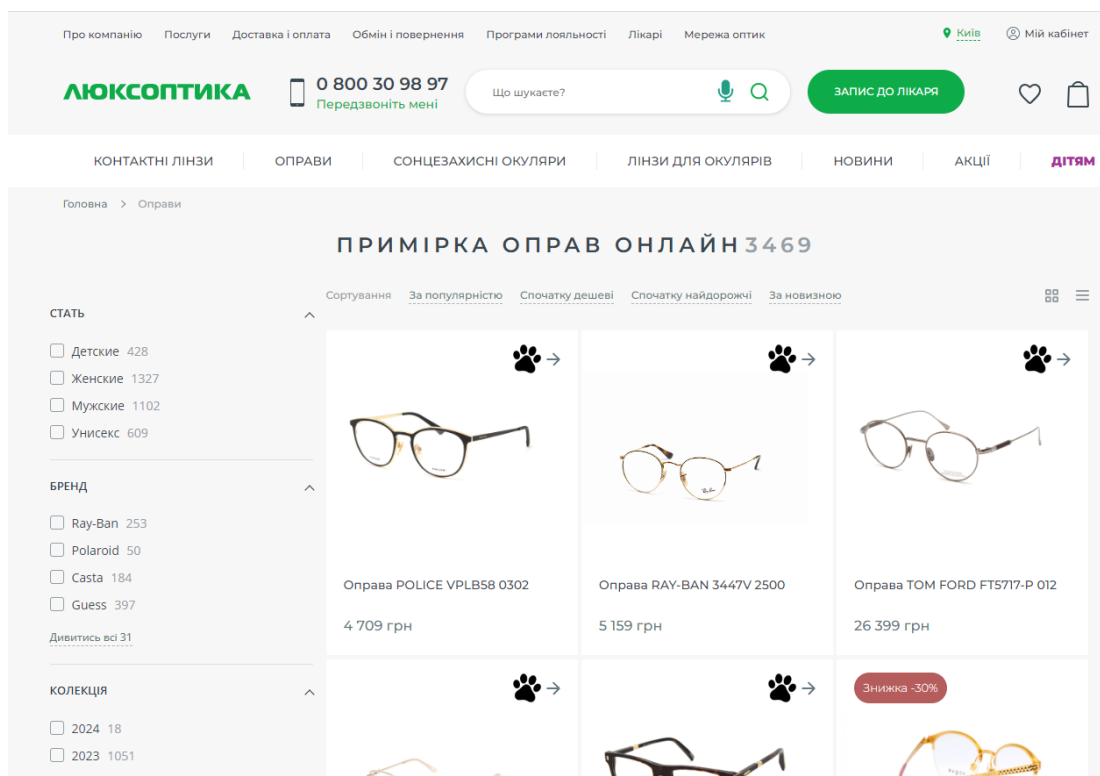


Рисунок 1.6 – Фільтри для підбору оправы

Джерело: [6]

Тобто система організована по прикладу інтернет-магазину.

Необхідно провести аналіз всіх модулів аналогів, розглянуті параметри приведено в таблиці 1.1.

Таблиця 1.1 – Аналіз додаткових модулів

Критерій	web-додаток alensa	web-додаток luxoptica	Розроблювана web- орієнтована система
Електронний обліковий запис	+	+	+
Система запису на прийом	-	+	+
Електронні рецепти	-	-	+
Модуль підбору оправ для окулярів	-	+	+
Підбір з приміркою на власне фото	-	-	+
Захист особистих даних	+	+	+
Адаптивність доступу	+	-	+

Джерело: побудовано автором

На основі аналізу були виявлені переваги та недоліки відомих web-додатків обслуговування офтальмологічних клінік.

Всі системи мають можливість реєстрації нових користувачів з підтримкою безпеки особистих даних, через можливість встановлення власного логіна й пароля. Основними недоліками відомих аналогів електронних сервісів підтримки діяльності офтальмологічних клінік є відсутність бази даних електронних рецептів та можливості віртуальної примірки певного типу оправ на власне фото.

Також по проведеному аналізу було прийнято рішення створення бази даних архіву рецептів та звернень користувачів, а також додати доступ користувачам до перегляду власних даних.

2 ПОСТАНОВКА ЗАДАЧІ ТА МЕТОДИ ДОСЛІДЖЕННЯ

2.1 Мета та задачі дослідження

Мета дослідження полягає у розробці web-орієнтованої системи підтримки діяльності офтальмологічної клініки.

Для досягнення визначеної мети, необхідно виконати наступні завдання:

- проаналізувати предметну область та сформулювати актуальність роботи.

В рамках даної задачі планується проаналізувати наукову та технічну літературу в пошуках відомостей, щодо підбору якомога доповненого варіанту реалізації web-орієнтованої системи підтримки діяльності офтальмологічної клініки, що дозволить організувати більш зручне рішення для користувачів;

- проаналізувати існуючі рішення web-орієнтованих систем підтримки діяльності офтальмологічних клінік. Що дозволить виявити недоліки існуючих рішень та створити web-орієнтовану систему з найбільш повним спектром можливостей функціоналу [5-19];

- визначити функціональні на нефункціональні вимоги до web-орієнтованої системи. Серед яких обов'язковими є створення захищених облікових записів з підключенням до баз даних з рецептами та оглядами лікарями, а також функції примірки оправ на зображення власного обличчя;

- провести моделювання бізнес-процесів. Тобто скласти IDEF0 модель, яка деталізує перелік виконуваних операцій і дій, що здійснюються у бізнес-процесі та принцип взаємозв'язку між ними. Use case діаграму, яка є графічним інструментом для моделювання взаємодій між різними акторами (користувачами чи іншими системами) та системою чи програмним продуктом. Скласти Sequence Diagram для показу взаємодій між об'єктами в певному порядку та інші;

- розробити програмне рішення у вигляді web-орієнтованої системи. Для цього планується використання відкритої фреймворк Laravel для web-розробки.

2.2 Методи дослідження

Протягом останнього десятиліття web-додатки розвинули свій функціонал від статичних web-сторінок до динамічних інтерактивних порталів з широкою номенклатурою функцій і багатозадачними системами управління інформацією [9]. В сучасних умовах web-додатки вже є достойними конкурентами багатьом не лише спеціалізованим програмам для комп'ютерів, а й продовжують розвиватися та привласнювати все нові горизонти шляхом переваг хмарних сервісів та можливостей. Більше того, майже кожна офісна спеціалізована програмна має аналог у вигляді web-сторінки або web-системи. А це дозволяє користувачам без великих втрат часу створювати, редагувати і поширювати інформацію через web-браузери незалежно від фінансової складової апаратного забезпечення, що використовується, звільняючись від вимог до офісного і настільного ПК. [20]

Через те, що більшість web-додатків використовують SQL-запити для обробки даних користувачів, таких як редагування особистих даних або заповнення анкет на сайтах, існує ризик SQL-ін'єкцій. Якщо не проводиться достатньої перевірки даних, зловмисник може вставити код SQL-запиту через web-інтерфейс, що може призвести до отримання несанкціонованого доступу до бази даних та видалення або зміни інформації. [21-24]

Крім того, загроза ін'єкцій існує і в інших контекстах, де дані отримуються ззовні. Наприклад, ін'єкції команд в операційну систему (OS injections) можуть виникнути, якщо не відбувається належна фільтрація введених даних.[20]

Міжсайтовий скриптинг (XSS) представляє іншу загрозу, коли вбудовується в код JavaScript для виконання в браузері користувача. Атаки такого типу часто відомі як HTML-ін'єкції, оскільки їхній механізм подібний до SQL-ін'єкцій, але відмінність полягає в тому, що впроваджуваний код виконується на боці клієнта.[24-25]

Для забезпечення безпеки web-додатків важливо належно налаштувати всі компоненти інфраструктури, включаючи фреймворки, web-сервери, сервери баз даних та платформу, оскільки налаштування за замовчуванням часто може стати вразливим до атак. [20]

Для проектування і розробки web-орієнтованих систем використовуються специфічні методичні дослідження, які складаються з ряду процедур:

1) web-дизайн – початковий етап планування, концептуалізації та розміщення контенту в Інтернеті. Сучасний дизайн web-сайтів часто виходить за рамки естетики і включає базову функціональність web-сайту. До web-дизайну відносяться web-додатки, мобільні програми та розробка інтерфейсу користувача.

На етапі web-дизайну закладається великий вплив на роботу системи в пошукових системах, тому вимогою сьогодення є створення web-орієнтованої системи, які не лише мають гарний вигляд, але й якісно функціонують, займаючи високі позиції в пошуку.

Стандартні способи розробки web-систем включають два варіанти: за допомогою спеціалізованого додатка або за допомогою конструктора web-сайтів. Основна відмінність даних варіантів полягає у розмірі вашої команди, вашого бюджету, типу сайту, який планується створювати, і вимог до нього. [26]

Також ефективність сайту напряму пов'язана з його швидкістю, рейтингом, можливістю пошуку та здатністю охопити потрібну аудиторію.

Розміщення контенту також має величезний вплив, як на комфорт використання, так і на функціональність сайту. Немає жодних детальних правил, які дозволять стовідсотково створити рентабельний та продуктивний сайт. Але є ряд рекомендацій, врахувавши які, можна досягнути гарних результатів:

- враховувати запити цільової аудиторії;
- уникати надмірно стимулюючих макетів, які відволікають від основного змісту сайту.[27]

У сучасних умовах, орієнтованих на мобільні пристрої, серйозним питанням стає параметр адаптивність web-сайту, тобто властивість підлаштовуватися під розміри екрану.

Адаптивність web-сайту визначається його здатністю ефективно адаптуватися до різних розмірів екранів на різних пристроях[17]. Правильна адаптація передбачає не лише коректне відображення, але й зручний перегляд без необхідності використання додаткових дій з боку користувача, таких як масштабування чи прокрутка. Існує кілька причин, чому ця характеристика є важливою, розглянемо три основних:

1) Зручність для користувача: Адаптивний дизайн дозволяє користувачам отримувати однаково комфортний та зручний доступ до інформації незалежно від того, який пристрій вони використовують. Це поліпшує загальний досвід взаємодії з web-сайтом; [28-31]

2) Пошукова оптимізація (SEO): Враховуючи важливість адаптивності для користувачів на мобільних пристроях, пошукові системи, такі як Google, враховують цей фактор при ранжуванні сторінок. Адаптивні сайти можуть мати кращі показники в пошукових результатах, що позитивно впливає на їхню видимість;

3) Розширення аудиторії: З урахуванням широкого розповсюдження мобільних та планшетних пристроїв, адаптивний web-дизайн розширює аудиторію сайту, сприяючи залученню користувачів з різних пристроїв та різних розмірів екранів.

Щоб перевірити адаптивність сайту, необхідно:

- перевірити коректність відображення у всіх популярних браузерах;
- перевірити коректність відображення на різних пристроях з різними розмірами екранів та операційними системами;
- перевірити коректність відображення при всіх можливих дозволах екрану.

Ці перевірки можна виконати самостійно, використовуючи різні пристрої та браузери. Також можна скористатися онлайн-сервісами перевірки адаптивності сайтів.[32]

«PHP - це мова програмування загального призначення з відкритим вихідним кодом, яка використовується для розробки web-сайтів.»[33] Синтаксис PHP орієнтований на більш ранні мови програмування C, Java і Perl. PHP відрізняється більшою простотою для освоєння новачком.

Переваги PHP:

- можливість швидко та зручно створювати динамічні web-сторінки;
- можливість працювати з базами даних;
- можливість створювати інтерактивні web-додатки.

Відмінності PHP від інших мов програмування:

PHP-скрипти виконуються на стороні сервера, а не на стороні клієнта.

PHP використовує власний синтаксис, який відрізняється від синтаксису інших мов програмування. [33,34]

Існують три основні області застосування PHP:

1) Написання серверних скриптів: PHP набув широкого використання для написання скриптів, які функціонують на боці сервера. Для цих цілей використовують лише парсер PHP, web-сервер і інтернет-браузер. Ця конфігурація дозволяє переглядати результати виконання PHP-скриптів у браузері;

2) Використання в командному рядку: PHP дає можливість писати скрипти, які зможуть виконуватися в командному рядку, незалежно від типу web-сервера та інтернет-браузера. Для цього потрібно скористатися лише парсером PHP. Такий варіант вигідно допомагає вирішувати регулярно виконувані завдання, такі як використання утиліти `cron` на операційних системах Unix або Linux, або використання календаря завдань Task Scheduler на операційній системі Windows. Даний тип скриптів можуть допомогти при обробці текстової інформації.

3) Розробка GUI-додатків для клієнтського використання: Навіть якщо PHP не є найоптимальнішою мовою для цієї області, використовуючи PHP-GTK, є можливість використовувати функціонал PHP для розробки графічного інтерфейсу клієнтського додатку. [34]

В роботах [30-38] зустрічаються відомості, що остання версія мови PHP виконує підтримку більшості відомих баз даних: mSQL; Informix; Direct MS-SQL; IBM DB2; Adabas D; ODBC; Ingres; InterBase; Oracle (OCI7 і OCI8); dBase; Ovrimos; Empress; FrontBase; PostgreSQL; FilePro (тільки читання); MySQL; Unix dbm.

Мова програмування PHP має мови аналоги, такі як: Python, Java.

Python, найбільш популярна мова програмування, що може бути варіантом заміни PHP для великої кількості задач web-проектів. Потужність даної мови програмування особливо проявляється при обробці великих масивів даних та керуванні їх потоками, що є дуже важливим якщо проект з великими кількостями сценаріїв автоматизації та системами аналітики проектів.[28]

Java, подібно до інших значущих конкурентів, вирізняється своєю дисципліною та можливістю структурувати код, що часто відсутнє у PHP. Java дозволяє написати вірний код, що дозволяє зменшити ймовірність виникнення помилок, в порівнянні з PHP, особливо при спільній роботі з розробниками, які не повністю ознайомлені з кодом.[29]

Однак індивідуальна розробка web-сайтів на PHP може стати можливою завдяки великій гнучкості налаштувань. PHP надає розробникам широкі можливості налаштування, що дозволяє досягти необхідного зовнішнього вигляду та функціоналу для web-сайту. Це надає достатньо гнучкості для створення особистого інтернет-простору, де враховується індивідуальний смак та потреби користувача.

Порівняння Java і PHP в контексті дисципліни коду та гнучкості налаштувань важливе для вибору оптимального інструменту в залежності від вимог і конкретних завдань розробки web-сайтів. [34-40]

3 МОДЕЛЮВАННЯ ТА ПРОЕКТУВАННЯ

3.1 Моделювання процесу роботи

3.1.1 Моделювання процесу роботи у нотації IDEF

Ще до початку виконання кваліфікаційної роботи важливо визначити основні елементи структурно-функціональної моделі процесів, які має в результаті виконувати web-орієнтована система підтримки діяльності офтальмологічної клініки.

IDEF0 - це комплекс процедур, правил і методів, що використовуються для побудови структурно-функціональної моделі web-системи.[31] Для виконання таких структур виокремлюється певна послідовність етапів: розробити узагальнений опис модулів системи та звязки взаємодії між ними[28].

Для зображення структурно-функціональної моделі процесів web-системи було розроблено контекстну діаграму в нотації IDEF0 представлену на рисунку 3.1, що відтворює процеси функціонування співпраці лікаря та пацієнта. Для реалізації діаграми було визначено: вхідні та вихідні дані, механізми взаємодії, елементи керування або обмеження.

Зліва від блоку відображені вхідні дані до web-орієнтованої системи підтримки діяльності офтальмологічної клініки є:

- запит реєстрації облікового запису;
- запис на прийом до лікаря;
- запит на замовлення окулярів по рецепту;
- запит на підбір розмірів оправ окулярів;
- запит прийомів пацієнта;



Рисунок 3.1 – Контекстна діаграма в нотації IDEF0

Джерело: побудовано автором

Праворуч від блоку відображено вихідні дані:

- створений обліковий запис;
- записи на прийом;
- отримані рецепти;
- результат підбору оправ у вигляді фото;
- виписка прийомів пацієнта;

Зверху від блоку відображено керуючі елементи процесу:

- графік роботи лікарів;
- завантажене фото пацієнта;
- форми рецептурних бланків;
- каталог оправ;

Знизу від блоку відображено механізми:

- web-орієнтована система;
- пацієнт;
- лікар;
- база даних;

– апаратне забезпечення.

Наступним кроком є діаграма декомпозиції 1 рівня проекту (рис. 3.2), вона складається з п'яти функціональних рівнів:

- авторизація;
- облік прийому лікарів;
- облік рецептів пацієнтів;
- підбір оправ;
- облік відвідин.

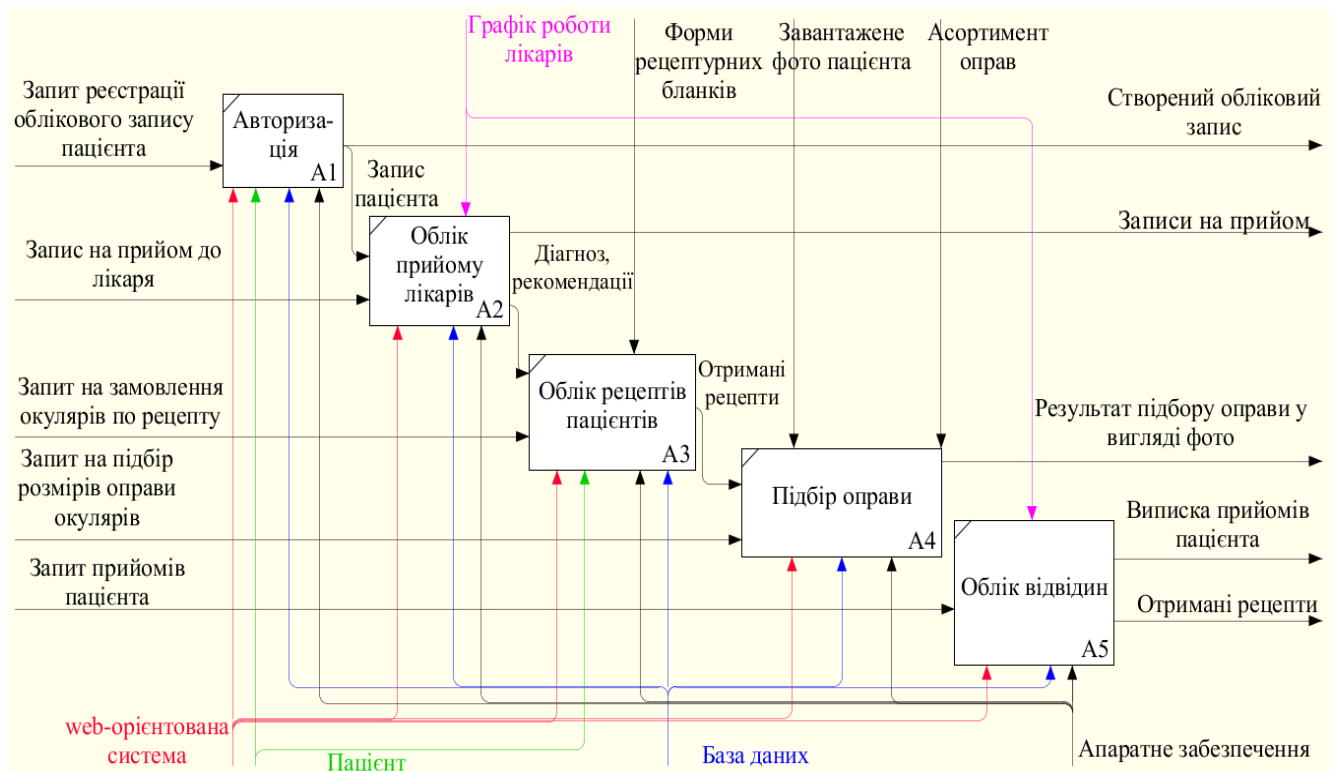


Рисунок 3.2 – Діаграма декомпозиції 1 рівня

Джерело: побудовано автором

3.1.2 Модель варіантів використання

Варіант використання — це можливі описи взаємодії між системою та користувачем або програмним продуктом. Кожен сценарій використання може

визначати різні умови, такі як винятки, зміни, або випадки успіху. Інструмент моделювання варіантів використання може бути використаний для опису цих сценаріїв як у вигляді тексту, так і візуально.

Кожен сценарій варіанта використання визначає функції, які повинні бути реалізовані в системі. При цьому, система може мати функції, які взаємодіють з іншими варіантами використання або виявляють свою унікальну поведінку.

У таблиці 3.1 відображено інформацію про акторів і опис варіантів використання (табл.3.2).

Таблиця 3.1 –Актори системи

Назва	Опис
Пацієнт	Користувач web-додатку
Лікар	Оформлення рекомендацій рецепту
База даних	Зберігає всю інформацію системи

Джерело: побудовано автором

Таблиця 3.2 –Варіанти використання опис

Назва	Описання
Авторизація пацієнта	Вхід до web-орієнтованої системи
Запис до лікаря	Підбір лікаря та отримання електронного талону
Відгуки щодо обслуговування	Пацієнт має можливість додати зворотній зв'язок про якість обслуговування.
Переглянути каталог	Можливість обирати тип оправи з каталогу представленого в додатку

Продовження таблиці 3.2

Назва	Опис
Обрати розмір оправы по розміру обличчя по фото	Можливість підбору оправы по параметрам обличчя за допомогою додаткового плагіну
Замовлення окулярів по рецепту та параметрах обличчя	Можливість оформити заявку на виконання окулярів по своїм параметрам
Зміна даних	Редагування даних і зміною виконується адміністратором.
Історія оглядів	Журнал оглядів та історія рецептів, ведеться адміністратором.
Виконання замовлення	Обробка та оформлення окулярів та фіскальний чек операції.

Джерело: побудовано автором

Для опису дій виконуваних web-орієнтованою системою, розроблено UseCase діаграму. Діаграма демонструє функціонування системи та шляхи взаємодії системи з користувачами. Дана діаграма має в своєму складі: акторів, елементи зовнішнього зв'язку з системою, варіантів використання, зв'язків між частинами діаграми. [22] Діаграма варіантів використання або use case діаграма зображено на рисунку 3.3.

В діаграмі визначено наступних акторів:

- лікар – це користувач, який входить через web-орієнтовану систему та надає даному користувачу надавати призначення пацієнту після огляду;
- пацієнт – це користувач, який має можливість реєструватися або авторизується в системі, після чого стає доступним запис на прийом до бажаного

лікаря на зручний пацієнту час, також формується особистий кабінет, де можна переглядати призначення, надані йому лікарем;

– база даних – це сховище в якому зберігаються дані, що використовуються або вносяться в систему впродовж роботи з нею.

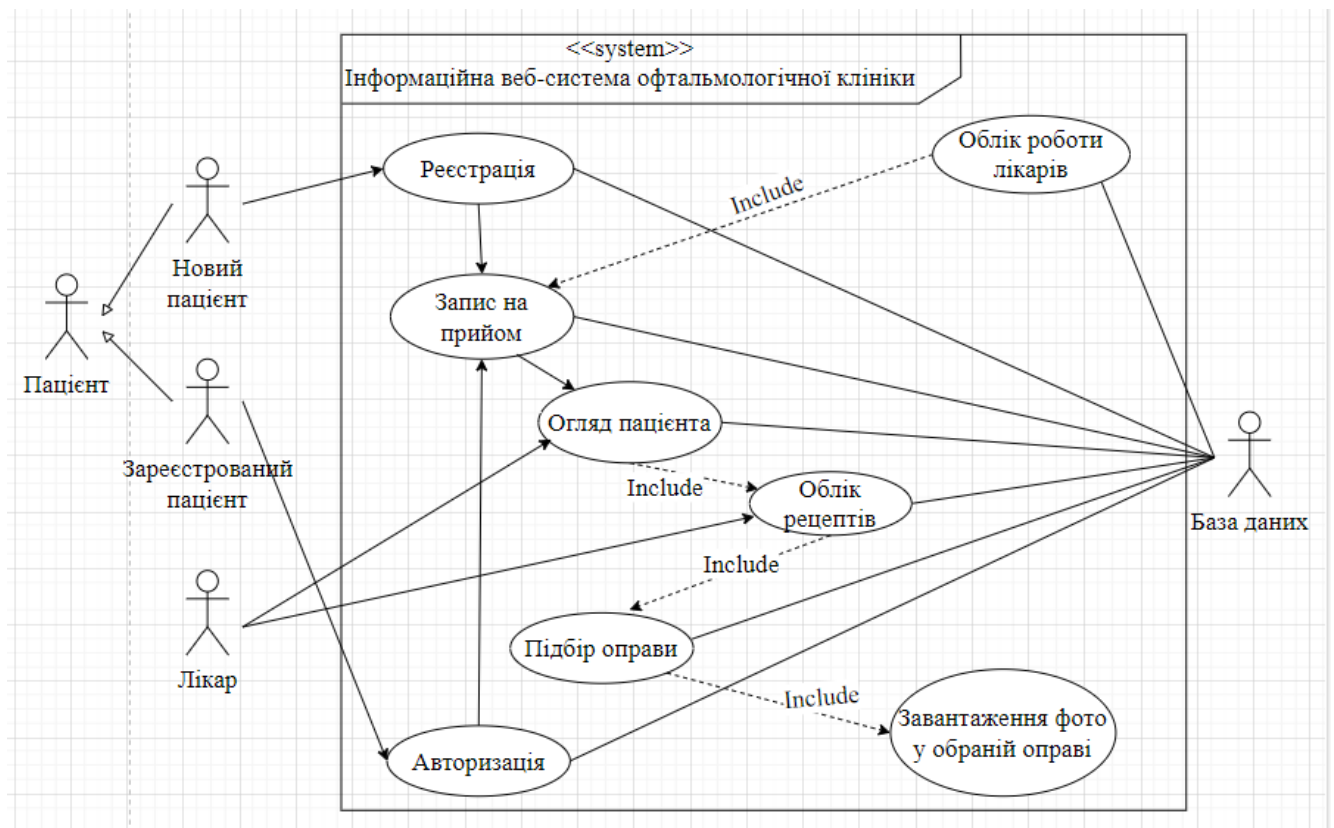


Рисунок 3.3 – Діаграма варіантів використання

Джерело: побудовано автором

Лікарі можуть переглядати записи пацієнтів, щоб отримати інформацію про стан здоров'я пацієнтів. Вони можуть також підтверджувати замовлення, щоб підтвердити, що пацієнти отримують необхідні окуляри. Крім того, вони можуть оформляти талони, щоб пацієнти могли записатися на прийом. І нарешті, вони можуть підбирати оправу, щоб допомогти пацієнтам знайти оправу, які підходять їм.

Пацієнти можуть реєструватися в системі, щоб створити обліковий запис. Вони можуть також вносити зміни до своїх даних, щоб оновити свою інформацію. Крім того, вони можуть переглядати записи, щоб отримати інформацію про свої недавні відвідування клініки. І нарешті, вони можуть оформляти замовлення, щоб замовити окуляри.[35-41]

Опис дій в модулі підбір оправи представлено на рисунку 3.4. Діаграма показує, як різні модулі взаємодіють один з одним, щоб забезпечити користувачам можливість підбирати оправу.

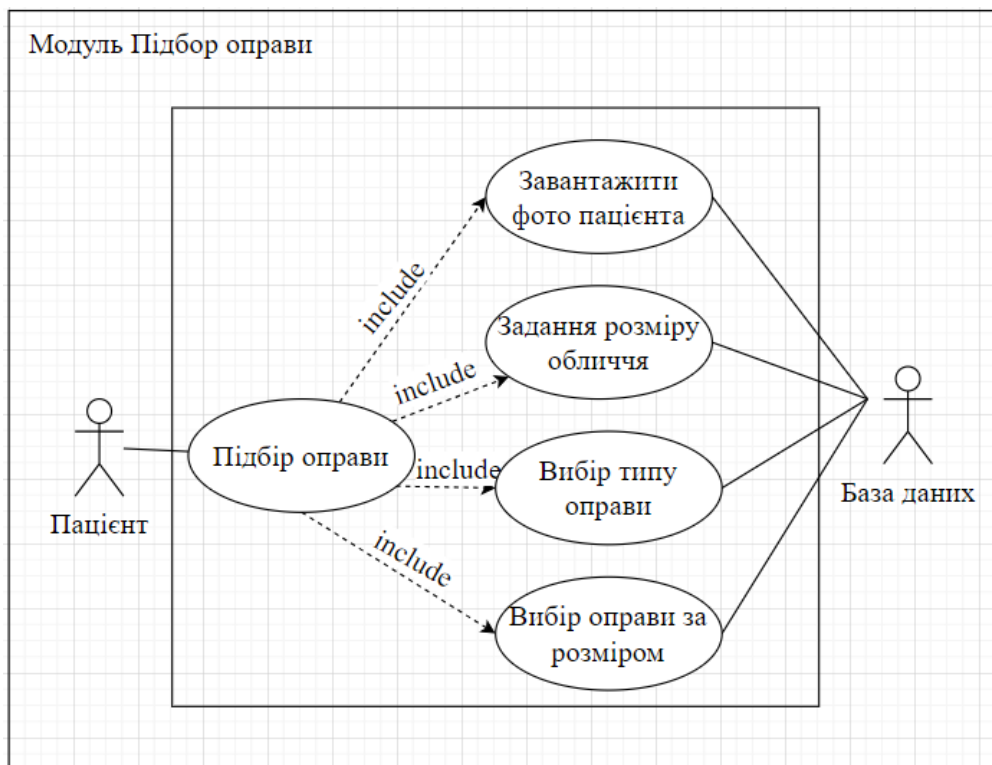


Рисунок 3.4 – Діаграма сценаріїв використання модулю «Підбору оправи»

Джерело: побудовано автором

Діаграма складається з двох частин. Верхня частина показує різні модулі додатку. Це включає в себе:

- модуль завантаження фотографії пацієнта: Цей модуль дозволяє пацієнту завантажити фотографію свого обличчя;

- модуль визначення розміру обличчя: Цей модуль аналізує фотографію пацієнта, щоб визначити його розмір обличчя;
- модуль вибору типу оправы: Цей модуль дозволяє пацієнту вибрати тип оправы, який йому подобається;
- модуль вибору оправы за розміром: Цей модуль дозволяє пацієнту вибрати оправу, яка підходить йому за розміром.

3.2 Проектування web-орієнтованої системи

Дизайн виконуваної web-орієнтованої системи є дуже важливим параметром, який дозволить мати високий рейтинг такої системи. Аналіз дозволив розробити дизайн web-системи, що зображено на рисунку 3.5.

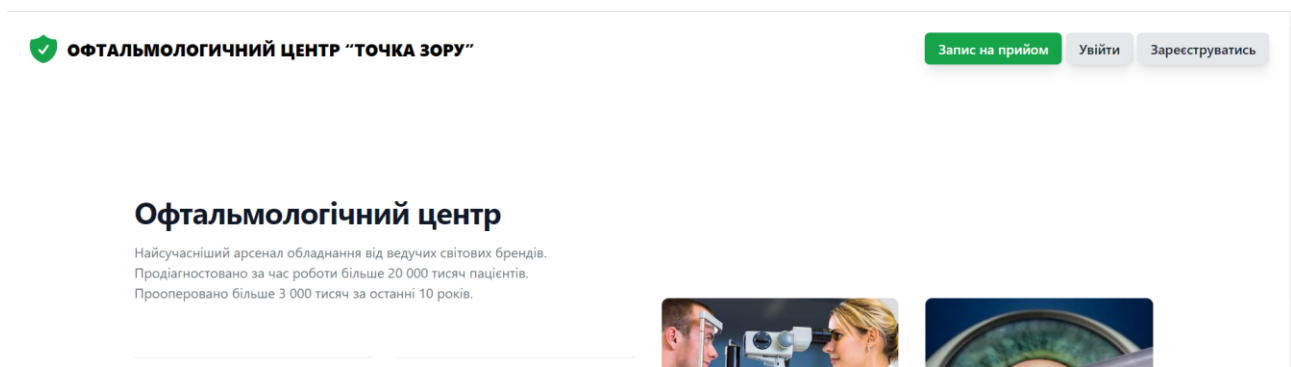


Рисунок 3.5 – Дизайн початкової сторінки системи

Джерело: побудовано автором

Після проходження процедури реєстрації або ж авторизації огляду відкривається внутрішня структура web-орієнтованої системи (рис. 3.6).

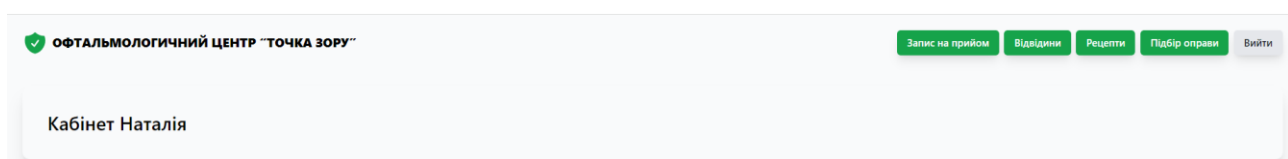


Рисунок 3.6 - Внутрішня структура web-орієнтованої системи

Джерело: побудовано автором

На цій панелі розміщено функції: запис на прийом; відвідини; рецепти; підбір оправ; та вийти.

При натисканні кнопки «Запис на прийом» відкривається меню вибору лікаря з випадаючого меню, дати прийому та опису проблеми (рис. 3.7).

The screenshot shows a web interface for an ophthalmology center. At the top, there is a navigation bar with buttons for 'Запис на прийом' (Appointment), 'Відвідини' (Visits), 'Рецепти' (Prescriptions), 'Підбір оправ' (Glasses selection), and 'Вийти' (Logout). Below this, the user is identified as 'Кабінет - Запис на прийом - Наталія'. The main form contains three input fields: 'Лікар' (Doctor) with a dropdown menu showing 'Оберіть лікаря', 'Дата' (Date) with a date picker showing 'дд.мм.рррр', and 'Опис проблеми' (Problem description) with a text area. A green 'Записатись' (Book) button is located at the bottom left of the form.

Рисунок 3.7 – Вікно запису до лікаря

Джерело: побудовано автором

Вигляд вікна перегляду відвідин пацієнтами-лікарів представлено на рисунку 3.8. В цьому вікні відкривається таблиця з інформацією про:

- дату відвідування;
- обраного лікаря;
- вказаними симптомами;
- висновком лікаря, щодо діагнозу.

The screenshot shows the 'Історія відвідин' (Visit history) section. It features a table with columns for 'Дата' (Date), 'Лікар' (Doctor), 'Коментар' (Comment), and 'Діагноз' (Diagnosis). The table contains four rows of visit records.

Дата	Лікар	Коментар	Діагноз
2023-11-29	Мельник Софія Ігорівна	почервоніння очей та незвичайний біль	Синдром "сухого ока": Недостаток слезної жидкості, що може вызвать чувство сухости, раздражения и покраснения глаз.
2023-11-24	Гіндім Ольга Андріївна	рипка в оці чорного кольору	Кератоконус: Рідке захворювання, при котором роговица постепенно становится конусообразной, что приводит к искажению зрения.
2023-10-11	Лібіцький Юрій Зіновійович	очіва слізьотеча зр	Стризм: Неправильное выравнивание глаз, что может привести к двоению и ухудшению глубины зрения.
2023-10-02	Ратушняк Наталя Ярославівна	біль двоєнок очей	Епісклерит: Запалення шару склери, зовнішньої оболонки ока, що викликає почервоніння і болісність ока.

Рисунок 3.8 - Вікно перегляду відвідин пацієнтами-лікарів

Джерело: побудовано автором

Наступним елементом буде модуль «Підбору оправы» (рис.3.9-3.13) після його вибору відкривається вікно завантаження особистого фото.

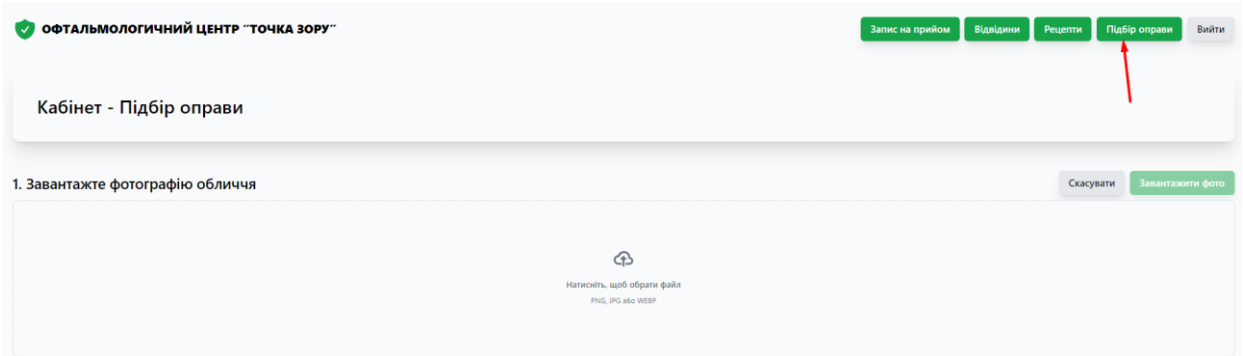


Рисунок 3.9 – Вікно завантаження особистого фото пацієнта

Джерело: побудовано автором

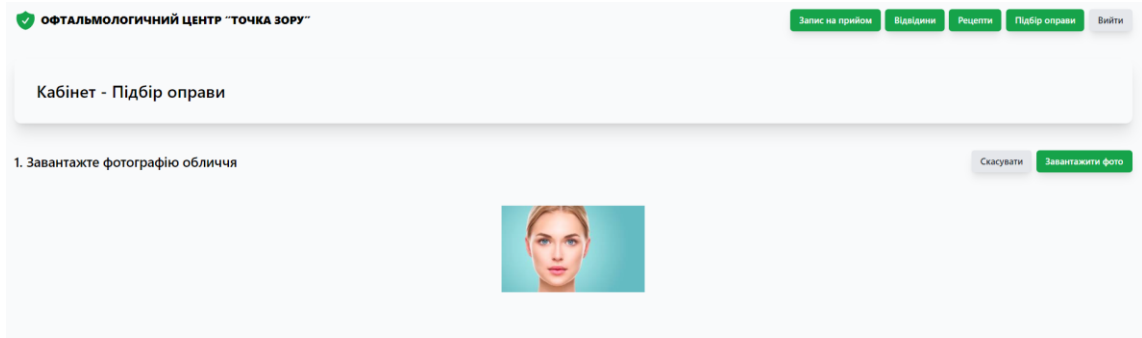


Рисунок 3.10 – Завантажуємо фото та натискаємо «Завантажити фото»

Джерело: побудовано автором

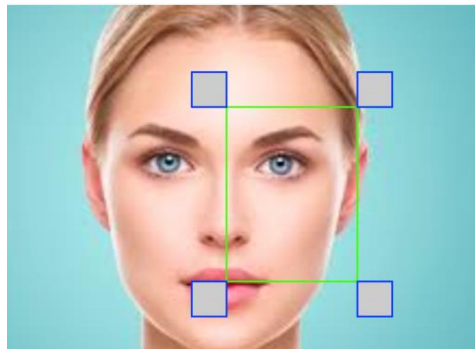


Рисунок 3.11 – Вікно задання параметрів обличчя

Джерело: побудовано автором

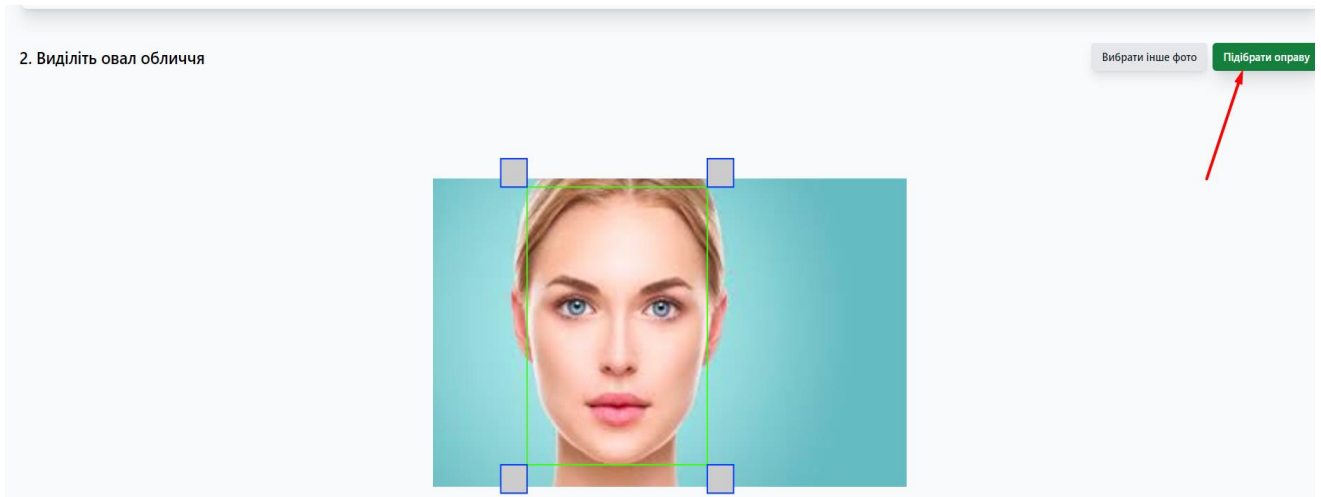


Рисунок 3.12 – Вікно після вказання розмірів, натискаємо «Підбір оправы»

Джерело: побудовано автором

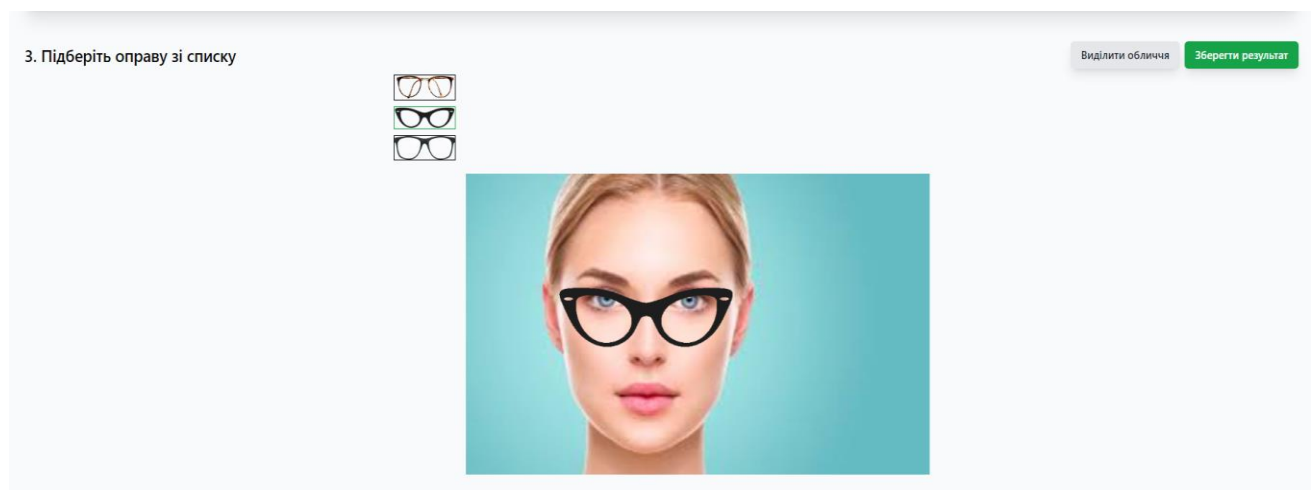


Рисунок 3.13 – Вікно з варіантами оправ

Джерело: побудовано автором

Після визначення найбільш підходящого варіанту оправы, який пасує пацієнту, натискаємо «Зберегти результат». Після виконання цих дій фото в окулярах автоматично завантажиться на особистий комп'ютер пацієнта в форматі .jpeg.

3.3 Проектування моделі бази даних

Головною складовою будь-якої web-орієнтованої системи є правильно розроблена база даних. Для представлення бази даних складають графічні або текстові моделі, на яких показують сутності, атрибути та зв'язки між ними. [35]

Відповідно до описаної раніше моделі варіантів використання, розроблювана інформаційна система зберігає всю інформацію в базі даних (БД).

База даних розроблюваної системи складається з 4 таблиць, які забезпечують їй відповідну роботу, логічна структура якої представлено на рисунку 3.14. [36]

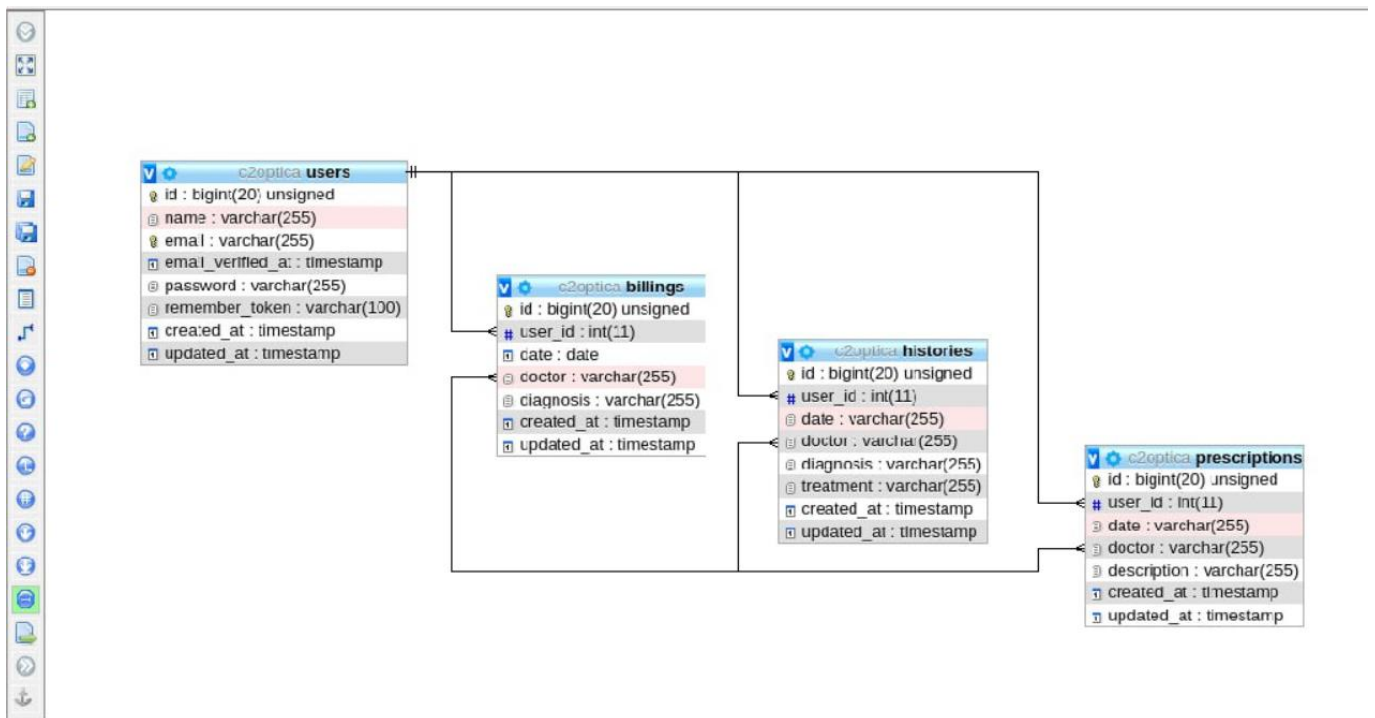


Рисунок 3.14 – Логічна структура бази даних

Джерело: побудовано автором

На рисунку представлена логічна структура бази даних організації роботи web-системи офтальмологічної клініки. Структура показує, як різні типи даних зберігаються в базі даних та їхні фізичні атрибути.

Рисунок складається з двох частин. Верхня частина показує різні типи даних, які зберігаються в базі даних. Це включає в себе:

1) Пацієнти: Цей тип даних містить інформацію про пацієнтів, включаючи їхні імена, дати народження, адреси та іншу особисту інформацію.

2) Лікарі: Цей тип даних містить інформацію про лікарів, включаючи їхні імена, спеціальності та інші професійні дані.

3) Записи: Цей тип даних містить інформацію про записи пацієнтів, включаючи дати та часи прийомів, діагнози та іншу медичну інформацію.

4) Окуляри: Цей тип даних містить інформацію про окуляри, які замовлені пацієнтами, включаючи тип, оправу та інші характеристики.

Нижня частина малюнка показує логічні атрибути кожного типу даних.

Таблиця пацієнтів:

- id: Первинний ключ, який ідентифікує кожного пацієнта;
- name: Ім'я пацієнта;
- email: Адреса електронної пошти пацієнта;
- date_of_birth: Дата народження пацієнта;
- address: Адреса пацієнта;
- phone_number: Номер телефону пацієнта.

Таблиця лікарів:

- id: Первинний ключ, який ідентифікує кожного лікаря;
- name: Ім'я лікаря;
- specialty: Спеціальність лікаря;
- years_of_experience: Кількість років досвіду лікаря.

Таблиця записів:

- id: Первинний ключ, який ідентифікує кожен запис;
- id_patient: Ключ зовнішнього посилання, який посилається на поле id в таблиці пацієнтів;

- `id_doctor`: Ключ зовнішнього посилання, який посилається на поле `id` в таблиці лікарів;

- `date`: Дата прийому;

- `time`: Час прийому;

- `diagnosis`: Діагноз;

- `treatment`: Лікування.

Таблиця окулярів:

- `id`: Первинний ключ, який ідентифікує кожну пару окулярів;

- `id_record`: Ключ зовнішнього посилання, який посилається на поле `id` в таблиці записів;

- `type`: Тип окулярів;

- `frame`: Оправа окулярів;

- `lens`: Лінзи окулярів.

Ця структура може бути використана для розуміння того, як різні типи даних зберігаються в базі даних для системи управління офтальмологічною клінікою. Це може бути корисно для розробників, які хочуть розробити або підтримувати цю систему.

Всі поля в таблицях мають типи даних, які вказують, який тип даних може зберігатися в кожному полі.

Поля `id_patient` та `id_doctor` у таблиці записів є ключами зовнішнього посилання, які посилаються на поля `id` в таблицях пацієнтів та лікарів відповідно. Це дозволяє системі відстежувати інформацію про пацієнтів, лікарів та записи.

4 ПРОГРАМНА РЕАЛІЗАЦІЯ

4.1 Архітектура web-орієнтованої системи

Магістерська робота спрямована на розробку web-додатку, призначеного для полегшення запису на прийом до лікаря-офтальмолога, переглянути особисту картку пацієнта з переліком звернень до лікаря, переглянуті рецепти, що були виписані протягом всього часу співпраці з клінікою, а також можливістю вибору пацієнтом найбільш підходящого типу оправи під контури власного обличчя без фізичної примірки. Архітектуру даного додатку приведено на рисунку 4.1.

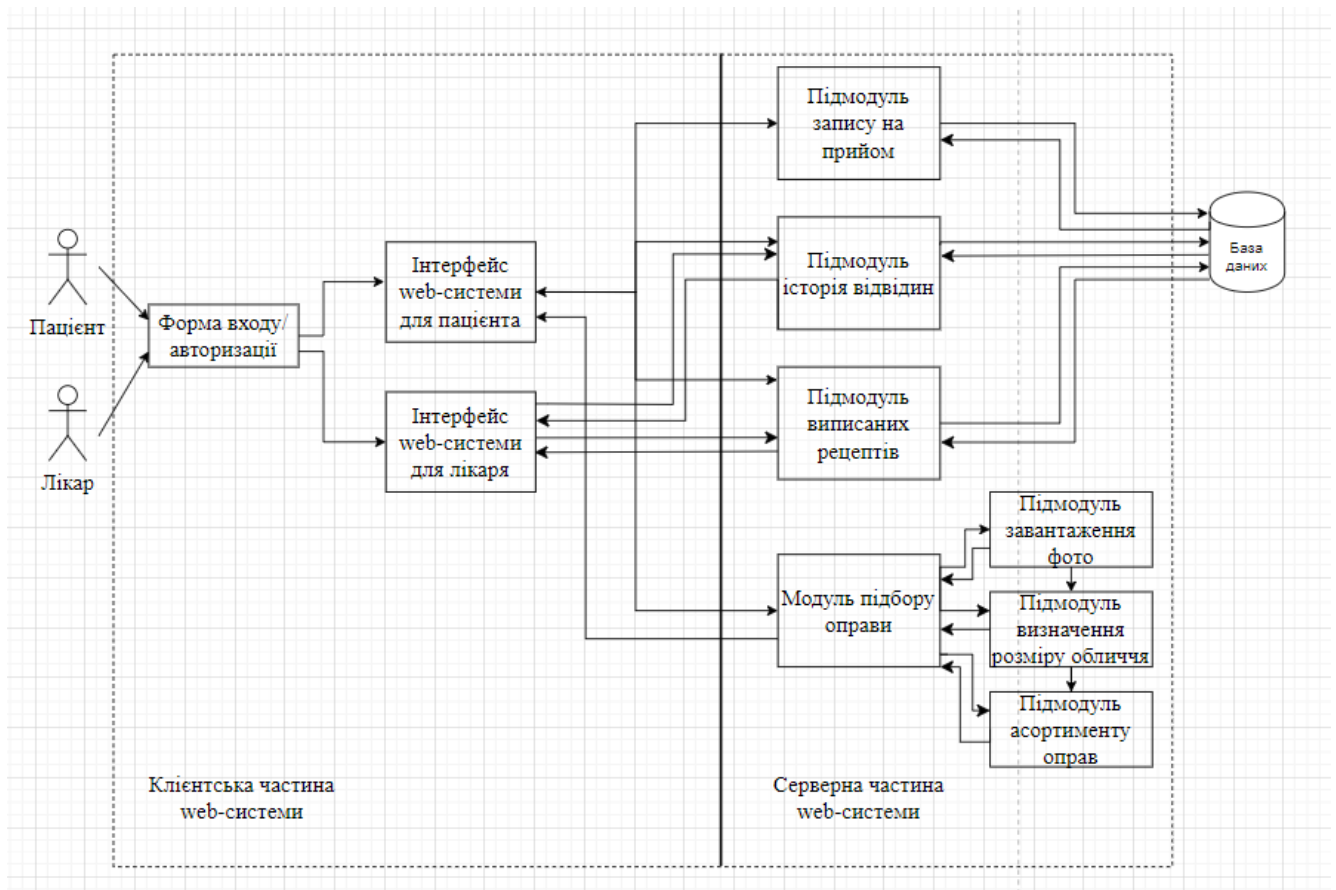


Рисунок 4.1 – Архітектура web-орієнтованої системи клініки

Джерело: побудовано автором

З рисунку зрозуміло, що після входу пацієнта в систему йому стає доступна користувацька панель. Достатньо лише пару кліків миші для оформлення запису до потрібного лікаря на певну дату та час, а також доступ до перегляду історії скарг пацієнта та звернень до лікаря, перегляд виписаних рецептів з рекомендаціями лікаря. Останнім є модуль системи призначений для зручності вибору оправы.

Даний модуль включає в себе ряд корисних функцій:

- можливість завантажити особисте фото пацієнта;
- визначення розміру для індивідуальних рис обличчя пацієнта;
- вибір оправы за зовнішнім виглядом;
- завантаження фото з результатом оправы.

Користувацький інтерфейс модулю "Підбір оправы" розроблений з логічною структурою та враховує зрозумілість для користувача. Дизайн модулю відповідає сучасним вимогам web-дизайну, що дозволяє навіть не досвідченому користувачеві легко користуватися всіма його можливостями.

Завдяки додатковим функціям модулю, навіть користувачі з обмеженим досвідом в роботі з комп'ютером зможуть ефективно використовувати цей інструмент для вибору та замовлення окулярів у потрібній оправі.

Головна сторінка web-системи - це його візитна картка, саме за першим враженням користувач вирішує, чи має бажання користуватися даною web-системою, чи простіше шукати аналоги з більш доступним інтерфейсом. [36-39]

Тому виділимо основні моменти головної сторінки, які відрізняють даний web-додаток від інших рисунк 4.2.

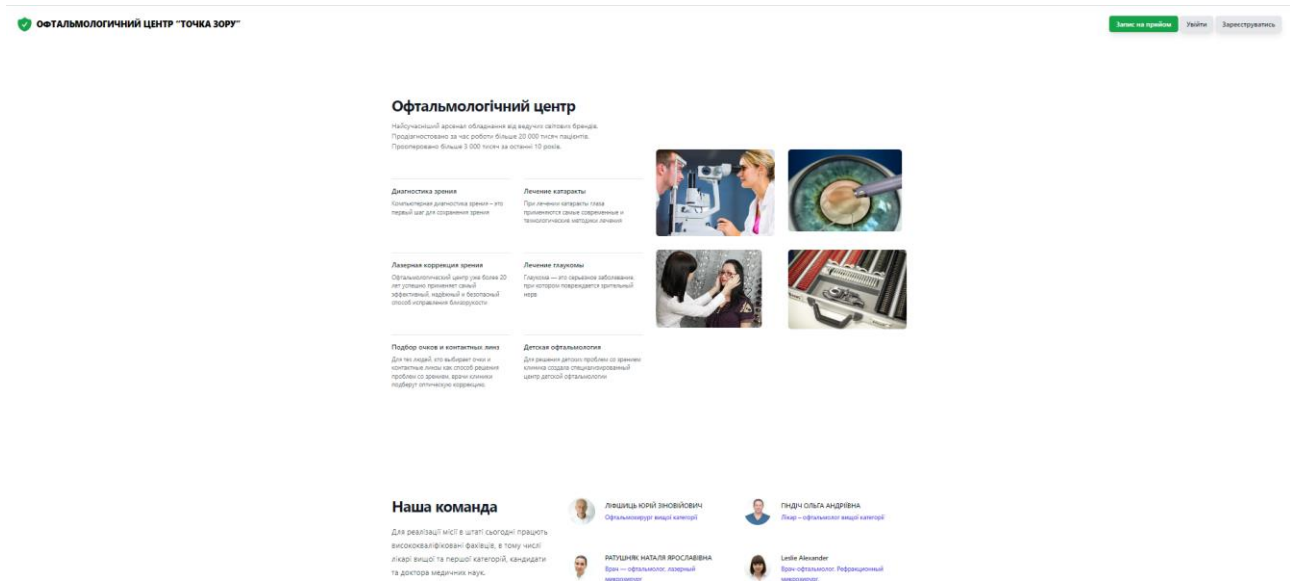


Рисунок 4.2 - Початкова сторінка web-системи офтальмологічної клініки

Джерело: побудовано автором

Початкова сторінка вирізняється інформативністю спеціалістів та процедур, які можна отримати в клініці та неперевантажена великою кількістю функціональних блоків. У верхньому правому куті пропонувані мінімалізовані варіанти співпраці з сайтом, а саме:

- записатися на прийом, де якщо пацієнт звертається вперше – буде запропоновано зареєструватися;
- увійти до особистого кабінету, якщо вже є обліковий запис;
- зареєструватися в особовому кабінеті.

4.2 Реалізація програмного коду

Для розроблюваної web-системи було розроблено дизайн головної сторінки (рис.4.3) для заохочення нових клієнтів. Для реалізації було обрано один з найбільш відомих конструкторів Laravel [40]. Даний фреймворк полегшує задачі розробника дозволяючи будувати легку комунікацію між головними модулями програмного коду.

Частина коду головної сторінки, що прописує процедуру реєстрації та взаємодію користувачького інтерфейсу з підмодулями системи описано нижче.

```
Route::middleware('guest')->group(function () {
    Route::get('/login', [LoginController::class, 'create'])
        ->name('login');

    Route::post('/login', [LoginController::class, 'store']);

    Route::get('/register', [RegisterController::class, 'create'])
        ->name('register');

    Route::post('/register', [RegisterController::class, 'store']);
});
Route::middleware('auth')->group(function () {
    Route::post('/logout', [LoginController::class, 'destroy'])
        ->name('logout');
    Route::get('/dashboard', [DashboardController::class, 'index'])
        ->name('dashboard');
    Route::get('/billing', [DashboardController::class, 'billing'])
        ->name('billing');
    Route::post('/billing', [DashboardController::class, 'billingStore']);
    Route::post('/confirm', [DashboardController::class, 'billingConfirm'])-
>name('confirm');
    Route::post('/cancel', [DashboardController::class, 'billingCancel'])-
>name('cancel');
    Route::get('/orders', [DashboardController::class, 'orders'])
        ->name('orders');
    Route::get('/history', [DashboardController::class, 'history'])
        ->name('history');
    Route::get('/check', [DashboardController::class, 'check'])
        ->name('check');
});
```

Джерело: побудовано автором

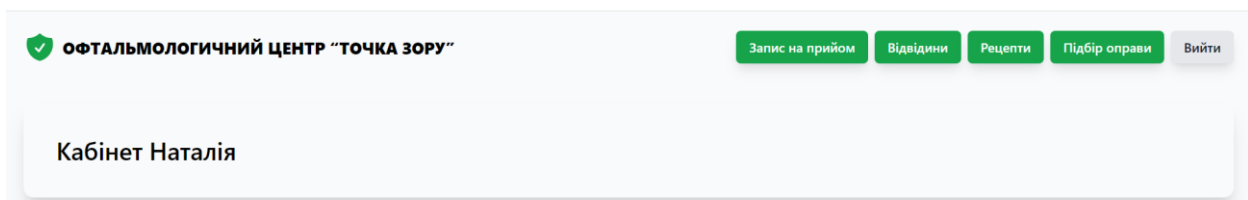


Рисунок 4.3 – Структура дизайну головної сторінки користувача

Джерело: побудовано автором

Laravel забезпечує легку взаємодію з системою керування масивними базами даних MySQL [41]. Для розроблюваної системи було виконано чотири таблиці бази даних:

- інформація про пацієнта зберігається в таблиці Users (рис.4.4);
- інформація про записи пацієнтів до лікаря з описом проблеми billings (рис.4.5);
- інформація про виписані рецепти та діагнози prescriptions (рис.4.6);
- історія відвідин зберігається в таблиці histories (рис.4.7);
- модуль підбору оправи.

Створюємо потрібні таблиці.

	id	name	email	email_verified_at	password	remember_token	created_at	updated_at
<input type="checkbox"/>	1	stlgsdsgsdg	igor.zabuza@gmail.com	NULL	\$2y\$10\$A5h6YkP9CZbHlLQK0ueEka5M3k35xcZw0TJYsw...	NULL	2023-11-07 10:53:22	2023-
<input type="checkbox"/>	2	Igor Zabuza	igor.zabuza1@gmail.com	NULL	\$2y\$10\$w0e8d0vnAnuXkEP0fBk6yG62CGacWNb6zQnEQeg...	7V5Z5yhEEA353FCXWPJM1dNoXXBJPocZ0hY13jhhblSarmY6...	2023-11-07 11:03:21	2023-
<input type="checkbox"/>	3	Сергій Гриценю	igor.zabuza2@gmail.com	NULL	\$2y\$10\$IXB9vyuT0uNW2LoNHivO7zdgjCnJ4Kack1LTTcW...	NULL	2023-11-08 12:19:44	2023-
<input type="checkbox"/>	4	Igor Zabuza	igor.zabuza4@gmail.com	NULL	\$2y\$12\$K5aG8uBIPiCK3.ZYRUTxxew7s0BlQ9yGuYkC2VVLW...	NULL	2023-11-12 23:00:01	2023-
<input type="checkbox"/>	5	john doe	vokavoka951@gmail.com	NULL	\$2y\$12\$g9HnCLAcnaVByxPInqk.Oltg9jCAWRjBZ5Hq5XS...	mTMZYTJ36BiZhCzqwq19vSuB44FqLa1ky8DLTeYpFrpLvy3u...	2023-11-15 11:01:39	2023-
<input type="checkbox"/>	6	Наталія	dyuhma42@gmail.com	NULL	\$2y\$12\$X6gCilQjFzQmUvG4VzCuXjXXWJnQhg0bag500QMK...	NULL	2023-11-19 17:40:57	2023-
<input type="checkbox"/>	7	Ivan Ivanov	ivanov@gmail.com	NULL	\$2y\$10\$R9CoRneWh.NDsetgtU6g.dYw5hzo10ctmb0eT84B4L...	NULL	2023-12-04 09:20:50	2023-

Рисунок 4.4 – Вигляд таблиці Users

Джерело: побудовано автором

У якості облікових даних пацієнта, як видно з рисунку зберігаються нікнейм, електронна пошта, пароль та дата реєстрації пацієнта та присвоєні пацієнтам id.

Сервер: localhost:3306 - База даних: c2optica - Таблиця: billings

Показано рядки 0 - 0 (всього 1, Залит виконувався 0.0007 секунди.)

SELECT * FROM `billings`

Число рядків: 25

id	user_id	date	doctor	diagnosis	created_at	updated_at
12	2	2023-12-14	Ратушняк Наталя Ярославівна	лагано банчу	2023-12-05 11:15:48	2023-12-05 11:15:48

Рисунок 4.5 – Вигляд таблиці billing

Джерело: побудовано автором

В даній таблиці зберігаються відомості про записи на огляд або ж талони, сюди входить інформація про id номеру огляду, id пацієнта, дата огляду, інформація про обраного лікаря, скарги пацієнта та дата створення запису.

Сервер: localhost:3306 - База даних: c2optica - Таблиця: prescriptions

Показано рядки 0 - 15 (всього 16, Залит виконувався 0.0005 секунди.) (date: 2023-12-01... - 2023-12-01...)

SELECT * FROM `prescriptions` ORDER BY `date` ASC

Число рядків: 25

Сортувати за ключем: Жодного

id	user_id	date	doctor	description	created_at	updated_at
1	6	2023-12-01	Гіндіч Ольга Андрівна	Діагноз: Міопія (кратковидність) Сфера (SPH): -2.0...	NULL	NULL
2	6	2023-12-01	Ліфшиць Юрій Зіновійович	Діагноз: Синдром сухого глаза Лекарственное средст...	NULL	NULL
3	6	2023-12-01	Ліфшиць Юрій Зіновійович	Діагноз: Гіперметропія (дальнозоркість) Тип линз: ...	NULL	NULL
4	6	2023-12-01	Гіндіч Ольга Андрівна	Діагноз: Глаукома Лекарственное средство: Глазные ...	NULL	NULL
5	6	2023-12-01	Гіндіч Ольга Андрівна	Діагноз: Міопія (кратковидність) Сфера (SPH): -2.0...	NULL	NULL
6	6	2023-12-01	Ліфшиць Юрій Зіновійович	Діагноз: Синдром сухого глаза Лекарственное средст...	NULL	NULL
7	6	2023-12-01	Ліфшиць Юрій Зіновійович	Діагноз: Гіперметропія (дальнозоркість) Тип линз: ...	NULL	NULL
8	6	2023-12-01	Гіндіч Ольга Андрівна	Діагноз: Глаукома Лекарственное средство: Глазные ...	NULL	NULL
9	2	2023-12-01	Гіндіч Ольга Андрівна	Діагноз: Міопія (кратковидність) Сфера (SPH): -2.0...	NULL	NULL
10	2	2023-12-01	Ліфшиць Юрій Зіновійович	Діагноз: Синдром сухого глаза Лекарственное средст...	NULL	NULL
11	2	2023-12-01	Ліфшиць Юрій Зіновійович	Діагноз: Гіперметропія (дальнозоркість) Тип линз: ...	NULL	NULL
12	2	2023-12-01	Гіндіч Ольга Андрівна	Діагноз: Глаукома Лекарственное средство: Глазные ...	NULL	NULL
13	2	2023-12-01	Гіндіч Ольга Андрівна	Діагноз: Міопія (кратковидність) Сфера (SPH): -2.0...	NULL	NULL

Рисунок 4.6 – Вигляд таблиці prescriptions

Джерело: побудовано автором

В даній таблиці зберігається інформація про результати огляду пацієнта, номер звернення, id пацієнта, дата огляду, інформація про лікаря, що проводив огляд та визначений діагноз та виписаний рецепт.

id	user_id	date	doctor	diagnosis	treatment	created_at	updated_at
5	2	2023-11-26	ГІНДІЧ ОЛЬГА АНДРІІВНА	уф ва ва I	Дальнозоркість пресбіопія: Утрата способности фоку...	2023-11-20 16:07:10	2023-11-20 16:07:10
6	2	2023-11-26	ГІНДІЧ ОЛЬГА АНДРІІВНА	уф ва ва I	Ретинопатія діабетическа: Ущерб судам сетчатк...	2023-11-20 16:08:21	2023-11-20 16:08:21
7	2	2023-11-26	ГІНДІЧ ОЛЬГА АНДРІІВНА	уф ва ва I	Гіперметропія (дальнозоркість): Трудности с фокуси...	2023-11-20 16:08:24	2023-11-20 16:08:24
8	2	2023-11-26	ГІНДІЧ ОЛЬГА АНДРІІВНА	уф ва ва I	Астигматизм: Розмитість зображення из-за неправи...	2023-11-20 16:08:27	2023-11-20 16:08:27
9	2	2023-12-01	Мельник Софія Ігорівна	Проблема зору	Гіперметропія (дальнозоркість): Трудности с фокуси...	2023-11-20 16:09:41	2023-11-20 16:09:41
10	2	2023-12-01	Мельник Софія Ігорівна	Проблема зору	Дальнозоркість пресбіопія: Утрата способности фоку...	2023-11-20 16:09:52	2023-11-20 16:09:52
11	2	2023-12-10	РАТУШНИК НАТАЛІЯ ЯРОСЛАВІВНА	Перевірка	Міопія (близорукість): Способность видеть ближе...	2023-11-20 16:12:13	2023-11-20 16:12:13
12	2	2023-12-22	Гіндіч Ольга Андріївна	Перевірити ліве око	Катаракта: Затемнение хрусталика глаза, что привед...	2023-11-20 16:29:25	2023-11-20 16:29:25
13	6	2023-11-24	Гіндіч Ольга Андріївна	риск в оді чорного кольору	Кератоконус: Редкое заболевание, при котором рого...	2023-11-20 16:30:21	2023-11-20 16:30:21
14	6	2023-11-29	Мельник Софія Ігорівна	покерення очей та незвичайні біль	Синдром "сухого глаза": Недостаток слезной жидко...	2023-11-20 16:31:32	2023-11-20 16:31:32
15	6	2023-10-11	Ліфшиць Юрій Євгенович	почав слабати зір	Стробизм: Неправильное выравнивание глаз, что мож...	2023-11-20 16:32:37	2023-11-20 16:32:37
16	2	2023-12-10	Гіндіч Ольга Андріївна	без проблем	Кератоконус: Рідкісне захворювання, при якому рог...	2023-11-20 16:33:43	2023-11-20 16:33:43
17	6	2023-10-02	Ратушник Наталія Ярославівна	біль дикола очей	Еписклерит: Запалення шару склери, зовнішньої обо...	2023-11-20 16:35:13	2023-11-20 16:35:13
18	7	2023-12-06	Ліфшиць Юрій Євгенович	subsd	Стробизм: Неправильное выравнивание очей, что може п...	2023-12-04 09:21:18	2023-12-04 09:21:18

Рисунок 4.7 – Вигляд таблиці histories

Джерело: побудовано автором

В таблиці histories зберігається інформація загального характеру про номер огляду, id пацієнта, дату огляду, лікаря, тип діагностики, визначений діагноз та вказівки лікаря.

Перед процедурою підбору оправи, користувачу потрібно зареєструватися на web-сайті клініки рисунок 4.8.

Рисунок 4.8 – Початкова сторінка реєстрації нового користувача

Джерело: побудовано автором

Використання web-орієнтованої системи підтримки діяльності офтальмологічної клініки дозволяє здійснювати:

- полегшений запис на прийом: Пацієнт може зручно обирати дату та час прийому лікаря-офтальмолога, використовуючи інтерактивний календар;
- вдалий вибір оправы: Пацієнт, завдяки функції визначення розміру та вибору за зовнішнім виглядом, може впевнено обирати оправу, яка ідеально підходить йому за всіма параметрами;
- збереження результатів та історії: Всі записи на прийом зберігаються в особистому кабінеті пацієнта для зручного відстеження та управління;
- дослідження меж працездатності: проведено тестування системи з великою кількістю одночасних користувачів для оцінки її швидкодії та масштабованості; використані сучасні методи шифрування та захисту даних для забезпечення конфіденційності медичної інформації; взаємодія із різними пристроями.[40-43]

Призначення блоків верстки приведено у таблиці 4.1.

Таблиця 4.1 - Перелік блоків макету додатку

№	Назва	Class	Призначення
1.	Заголовок	class="text-3xl font-semibold"	Назва модулю
2.	Блок гнучкого налаштування зображення	class="mb-2 flex flex-wrap md:flex-no-wrap items-center justify-between gap-2 w-full"	Налаштування гнучкої області робочого зображення
3.	Блок налаштування	class="text-2xl font-semibold"	Редагування текстових стилів
4.	Блок завантаження	class='input-medium'	Завантажує зображення з вашого пристрою в блок зображення

Джерело: побудовано автором

Продовження таблиці 4.1

№	Назва	Class	Призначення
5.	Вибір файлу зображення	class="mb-2 text-sm text-gray-500"><span class="font-semibold"	Дозволяє обрати необхідне зображення для підбору оправ
6.	Примірка окуляр	class="icon-picture"	При натисненні окуляри автоматично з'являються поверх блоку зображень

Джерело: побудовано автором

Код розробленого макету приведено у додатку Б.

4.3 Використання web-системи

Після завершення виконання web-системи, виконуємо тестування основних функцій, для початку тестується функція входу (рис. 4.9).

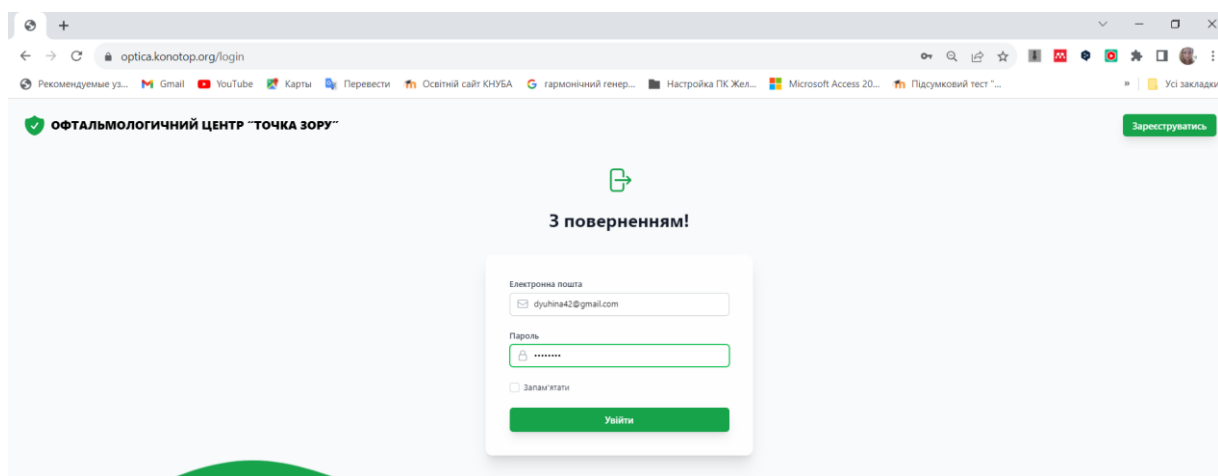


Рисунок 4.9 – Вікно активації облікового запису

Джерело: побудовано автором

Тестуємо функцію «Запис на прийом», де обираємо лікаря, що нам підходить з випадуючого списку (рис. 4.10), вказуємо дату та описуємо проблему. Результат запису зберігається в базу даних.

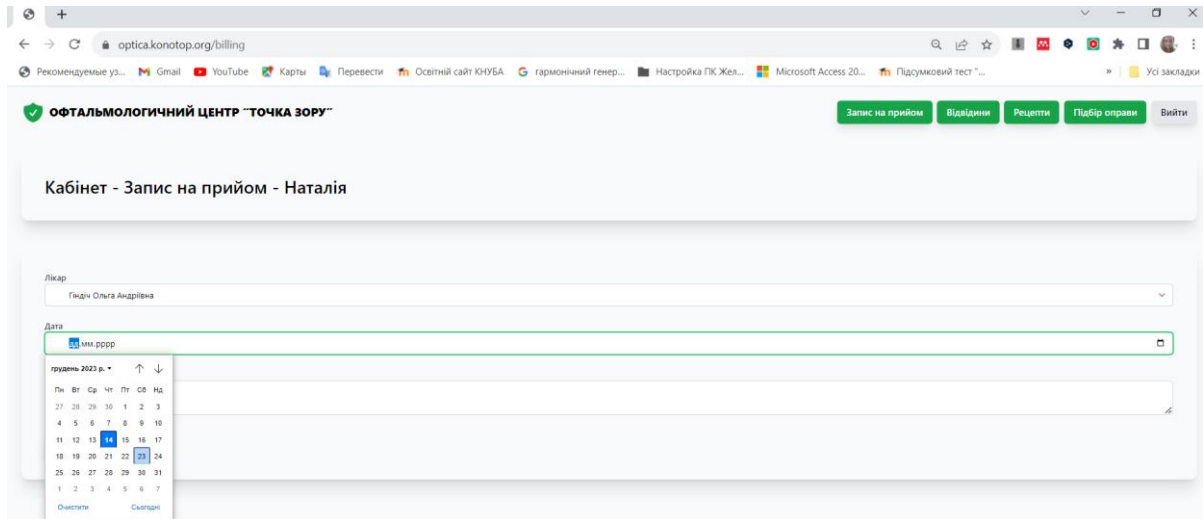


Рисунок 4.10 – Вікно запису на прийом

Джерело: побудовано автором

Тестуємо функцію «Відвідини», де відкривається таблиця звернень даного пацієнта до лікарів клініки (рис. 4.11). Для отримання даної інформації опрацьовується запит до бази даних.

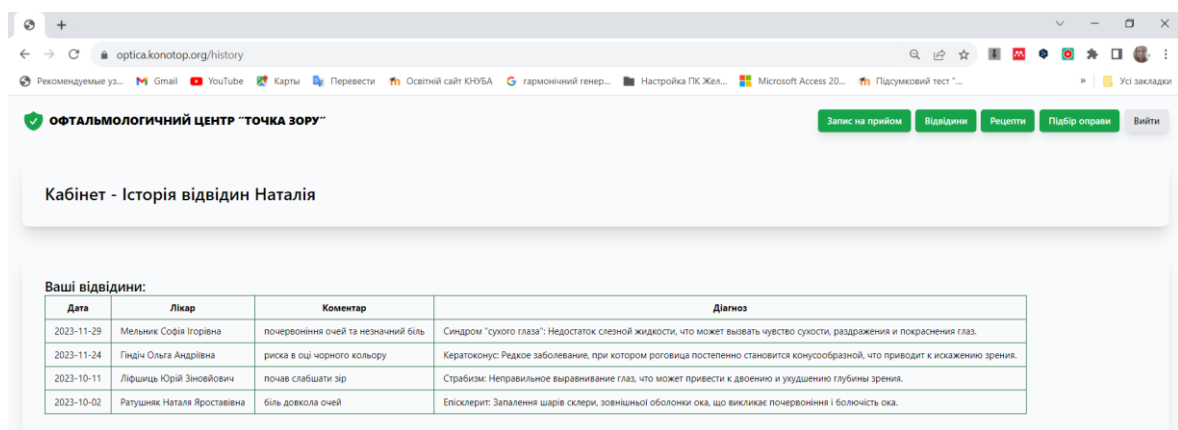


Рисунок 4.11 – Вікно запиту «Відвідини»

Джерело: побудовано автором

Тестуємо функцію «Рецепти», де відкривається таблиця рецептів з рекомендаціями по кожному зверненню даного пацієнта (рис. 4.12). Для отримання даної інформації опрацьовується запит до бази даних.

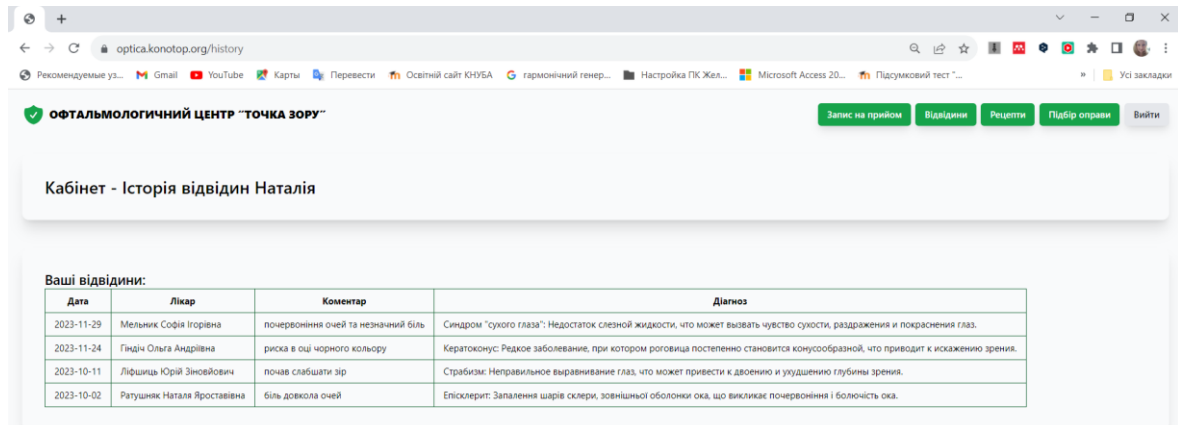


Рисунок 4.12 – Вікно запити «Рецепти»

Джерело: побудовано автором

Тестуємо функцію «Підбір оправ», після натискання на дану кнопку, натискаємо «Натисніть, щоб обрати файл (1)», обираємо улюблене фото та натискаємо «завантажити фото (2)» (рис. 4.13).

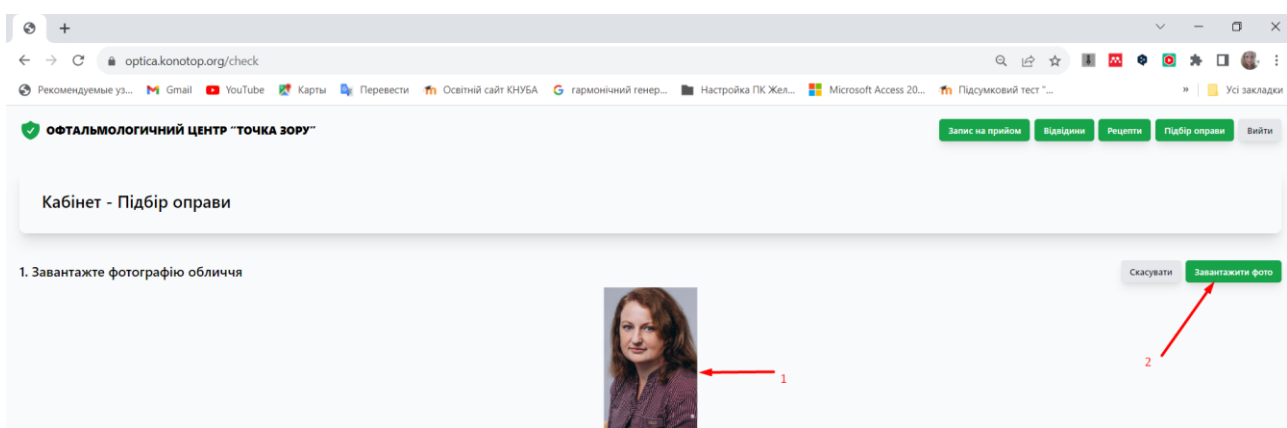


Рисунок 4.13 – Вікно «Підбір оправ»

Джерело: побудовано автором

Після завантаження фото пацієнта стає доступною рамка задання розмірів обличчя (рис. 4.14), далі простим перетягуванням розміщуємо розмірну сітку по контуру обличчя (рис. 4.15) та натиснути на кнопку «Підібрати оправу», або ж «Вибрати інше фото».

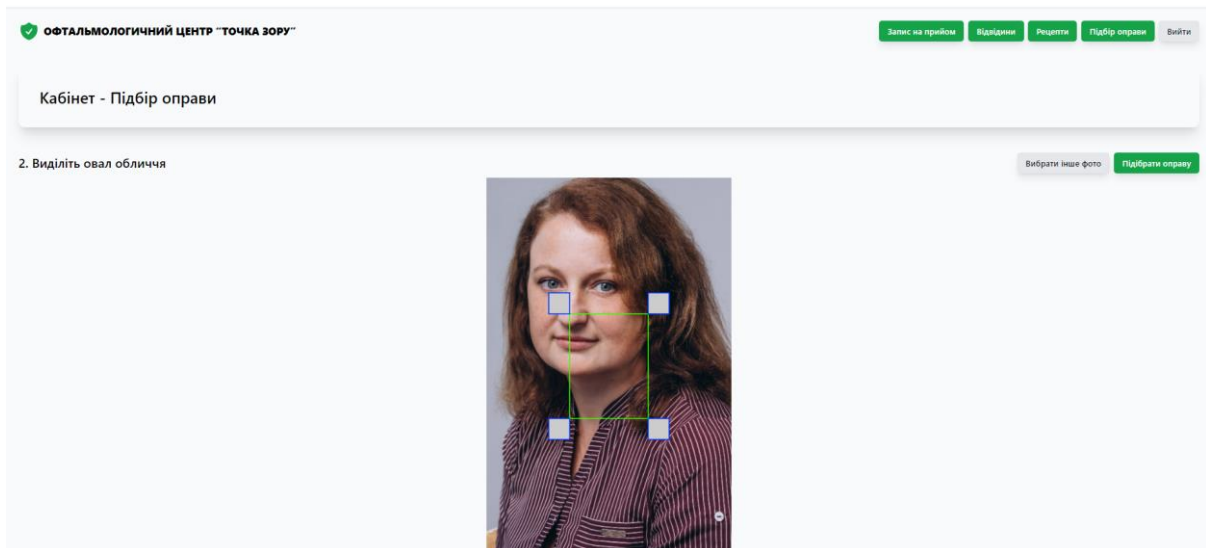


Рисунок 4.14 – Вікно «Підбір оправ», після завантаження фото пацієнта

Джерело: побудовано автором

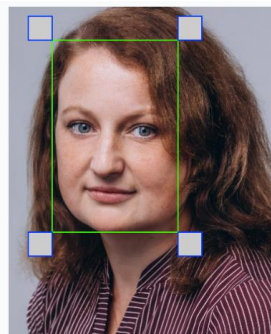


Рисунок 4.15 – Вікно «Підбір оправ», задання параметрів обличчя

Джерело: побудовано автором

З наявного асортименту оправ обираємо найбільш підходящий індивідуальний варіант (рис. 4.16). Далі натискаємо «Зберегти результати». Асортимент оправ можна розширити за вимогою клініки.

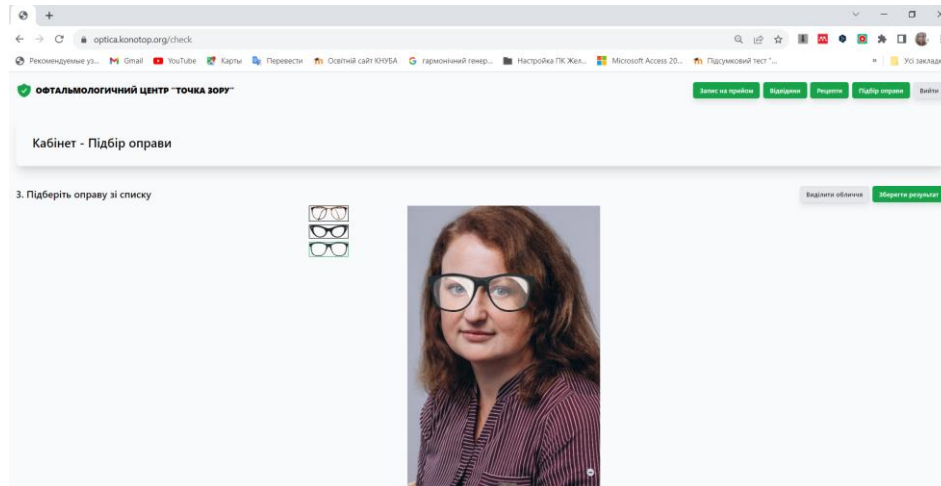


Рисунок 4.16 – Ваше зображення в обраній оправі

Джерело: побудовано автором

Далі натискаємо «Зберегти результати» і фото пацієнта буде завантажено у форматі картинки на комп'ютер пацієнта (рис. 4.17).

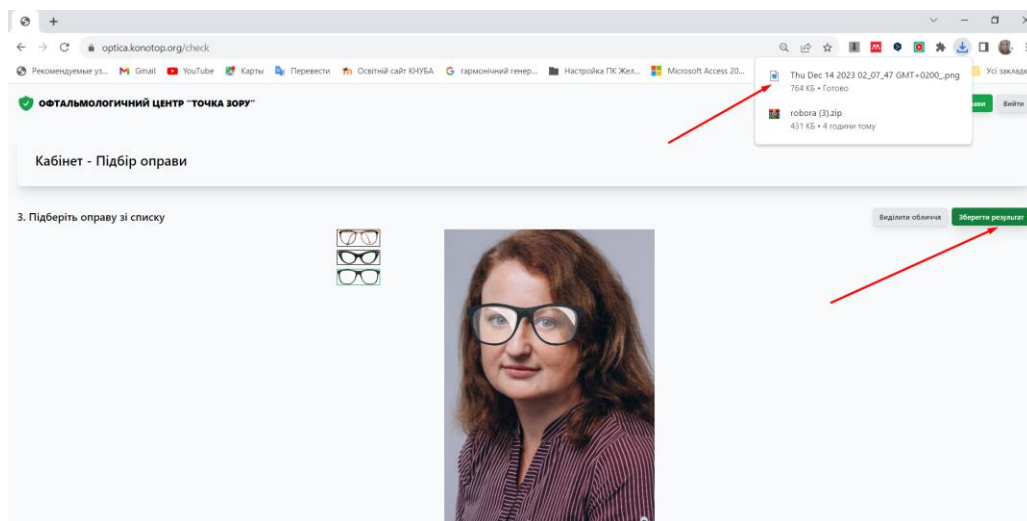


Рисунок 4.17 – Ваше зображення в обраній оправі

Джерело: побудовано автором

Отже, після проведеного тесту системи на реальному прикладі, можна стверджувати, що система працює незалежно від вибору пристрою, або ж варіанту браузера.

ВИСНОВКИ

В результаті виконання кваліфікаційної роботи магістра було проведено аналіз аналогів, призначених для дистанційного підбору окулярів. Однак, більшість з них орієнтовані на комерційну діяльність інтернет-магазинів і не мають функціоналу, необхідного для автоматизації діяльності офтальмологічних клінік. В ході аналізу визначено функціональні вимоги до системи такі, як: система реєстрації/авторизації пацієнтів; запис пацієнта на огляд; підключення бази даних історії хвороби пацієнта та оформлені рецепти; модуль «Підбір оправ». Нефункціональні вимоги: надійність, безпека, адаптованість та зручність використання. Виконано планування робіт системи.

Також, було проведено структурно-функціональне моделювання в нотації IDEF0 та її декомпозицією графічно представлено функціонування системи, що демонструє внутрішні зв'язки та функціонування системи. Проведено моделювання варіантів використання системи та розроблено модель бази даних. Було продумано дизайн майбутньої системи. Для реалізації всіх описаних функцій, було визначено фреймворк Laravel.

Перевагою розробленої web-орієнтованої системи є - реалізована функція ведення обліку відвідин та виписаних рецептів, можливість перегляду пацієнтами отриманих ними рецептів та рекомендацій. А також вбудований модуль підбору оправ з приміркою на власне фото, що надає можливість пацієнтам клініки без реальних примірок визначитися зі стилем оправ, який найбільше їм пасуватиме. Мета та всі задачі виконані, web-орієнтована система реалізована.

В перспективі є можливість розширення функціоналу системи, асортименту оправ, можливості налаштування фільтрів для різного типу оправ.

Розробка такої системи дозволить поліпшити якість обслуговування пацієнтів, підвищити ефективність роботи офтальмологічних клінік та знизити витрати.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Моделі надання телемедичних послуг. URL: https://sdh.com.ua/blog/mobile/Rozrobka_Telemedicine_Web/ (дата звернення: 10.10.2023).
2. Медична система для Офтальмології. URL: <https://www.clinica-web.ua/ophthalmology/> (дата звернення: 17.11.2023).
3. FittingBox Technology. URL: <https://www.fittingbox.com/en/fittingbox-technology> (дата звернення: 17.11.2023).
4. О.М. Черемухіна, С.О. Риков, І.В. Александров. Інформаційні системи і технології в офтальмології / УКР. МЕД. ЧАСОПИС, 2 (88) – III/IV 2012. URL: | www.umj.com.ua (дата звернення: 17.11.2023).
5. Alensa logo. URL: <https://www.alensa.ua/> (дата звернення: 17.11.2023).
6. Люксоптика. URL: <https://luxoptica.ua/ua/virtual-try-on/> (дата звернення: 17.11.2023).
7. Оптика. URL: <https://www.optika.ua> (дата звернення: 17.11.2023).
8. Київлінза. URL: <https://kyivlinza.ua/> (дата звернення: 17.11.2023).
9. Онлайн-консультація з професійними лікарями. URL: <https://doc.ua/ua/telemed> (дата звернення: 10.11.2023).
10. Онлайн-консультації лікарів-практиків . URL: <https://likaronline.com.ua/> (дата звернення: 17.11.2023).
11. Медична допомога від волонтерів України . URL: <https://check-up.in.ua/ukr/help> (дата звернення: 16.11.2023).
12. Онлайн-консультація . URL: лікаря <https://smartmedicalcenter.ua/ways/adults/onlajn-konsultatsiya-vracha/> (дата звернення: 17.11.2023).
13. Лікарі . URL: <https://e-likari.com.ua/doctors/ofthalmolog/> (дата звернення: 17.11.2023).

14. ОФТАЛЬМОЛОГ ОНЛАЙН . URL: https://medikit.ua/patient_consultations/oftalmoloh/oftalmoloh (дата звернення: 17.11.2023).
15. Експертна офтальмологія . URL: <https://znamenska.com/services/dystancijni-konsultacziyi/> (дата звернення: 22.11.2023).
16. Альтамедика. Медичний центр . URL: <https://altamedica.com.ua/pitannya-oftalmologu> (дата звернення: 17.11.2023).
17. Програма для окуліста . URL: <https://easyweek.com.ua/programa-onlajn-zapisu-dlya-oftalmologa.html>. (дата звернення: 19.11.2023).
18. Журавльова Л. В., Лопіна Н. А. Практично-орієнтований кейс-метод навчання в системі безперервної медичної освіти на основі інформаційно-освітніх веб-технологій як спосіб симуляційного навчання : навч.-метод. посібник для викладачів мед. освітніх закладів. – Харків : ХНМУ, 2019. – 76 с.
19. Медична інформаційна система «Доктор Елекс»: основи роботи: Навчальний посібник / під. ред. І. Березовської, Ю. Триуса. – Львів: Ліга Прес, 2018. – 186 с.
20. Г.Д. Жабоедов, та ін. Офтальмологія : підручник — К.: ВСВ “Медицина”, 2011. — 424 с.. 2011
21. Інформаційні системи і технології в офтальмології О.М. Черемухіна, С.О. Риков, І.В. АлександровУКР. МЕД. ЧАСОПИС, 2 (88) – III/IV 2012 | www.umj.com.ua.
22. УДК 004.415.5 С. Вишковський (Тернопільський національний технічний університет імені Івана Пулюя) ІНФОРМАЦІЙНИХ WEB-САЙТІВ ТА МЕТОДІВ ЇХ УСУНЕННЯ ДОСЛІДЖЕННЯ ВРАЗЛИВОСТЕЙ. URL: https://elartu.tntu.edu.ua/bitstream/lib/27194/2/IMST_2018_Vishkovskii_S-Formatsiinykh_veb_saitiv_12.pdf (дата звернення: 25.11.2023).
23. Медична система для Офтальмології. URL: <https://www.clinica-web.ua/ophthalmology/>(дата звернення: 25.11.2023).

24. Які помилки потрібно уникати при оптимізації сайту для просування послуг медичного центру. URL: <https://seo-evolution.com.ua/blog/poiskovaya-optimizatsiya/prodvizhenie-uslug-meditsinskogo-tsentra/> (дата звернення: 18.11.2023).

25. Просування медичних сайтів. URL: <https://ilion.digital/ua/medycina/> (дата звернення: 18.11.2023).

26. Контент-маркетинг як ключ до успішного SEO-просування медичного сайту: кейс Porcelain. URL: <https://web-promo.ua/ua/blog/kontent-marketing-kak-klyuch-k-uspeshnomu-seo-prodvizheniyu-mediczinskogo-sajta-kejs-porcelain/> (дата звернення: 18.11.2023).

27. Вебаналітика й робота з даними. URL: <https://mixdigital.com.ua/veb-analitika-ta-robot-a-z-danimi/> (дата звернення: 18.11.2023).

28. Веб-аналітика. URL: https://roman.ua/ua/analytics/?utm_content=%D0%90%D0%BD%D0%B0%D0%BB%D0%B8%D1%82%D0%B8%D0%BA%D0%B0_%D0%BE%D0%B1%D1%89%D0%B5%D0%B5&utm_campaign=Roman_Search_Services_Analytics_General_Ukraine&utm_term=%D0%B2%D0%B5%D0%B1%20%D0%B0%D0%BD%D0%B0%D0%BB%D0%B8%D1%82%D0%B8%D0%BA%D0%B0&gclid=Cj0KCQiA35urBhDCARIsAOU7QwmN2iS2xvZKoRUQfXLmGsKw4FtHTtQSaENy5MPwGAj6lfc16xNIIIYaAkwKEALw_wcB (дата звернення: 18.11.2023).

29. Просування сайту медичної клініки: поради та рекомендації. URL: <https://seo-evolution.com.ua/blog/seo-prodvizhenie/prosuvannya-saytu-medichnoyi-kliniiki> (дата звернення: 18.11.2023).

30. WEBSITE RESPONSIVENESS CHECK. URL: <https://seo-akademiya.com/en/knowledge-base/internal-optimization/website-responsiveness-check/> (дата звернення: 17.11.2023).

31. Адаптивний web-дизайн. URL: <https://www.ar25.org/article/adaptyvnyu-veb-dyzaun.html> (дата звернення: 17.11.2023).

32. Mobile-First: як адаптивність впливає на позиції ресурсу. URL: <https://dou.ua/forums/topic/45064/> (дата звернення: 17.11.2023).
33. Топ навичок php розробника. URL: <https://nt.ua/blog/top-php-developer-skills> (дата звернення: 17.11.2023).
34. PHP. URL: <https://programming.in.ua/web-design/allphp/30-about-php.html> (дата звернення: 17.11.2023).
35. PYTHON. URL: <https://www.python.org/> (дата звернення: 17.11.2023).
36. JAVA. URL: <https://www.java.com/ru/> (дата звернення: 17.11.2023).
37. МЕТОДОЛОГІЯ IDEF0. URL: https://stud.com.ua/87184/ekonomika/metodologiya_idef0 (дата звернення: 19.11.2023).
38. Варіанти використання та сценарії (Use Cases and Scenarios). URL: <https://www.maxzosim.com/use-cases-and-scenarios/> (дата звернення: 17.11.2023).
39. OpenCV Cascade Classifier. URL: https://docs.opencv.org/3.4/db/d28/tutorial_cascade_classifier.html (дата звернення: 17.11.2023).
40. Що таке Laravel?. URL: <https://asabix.com.ua/what-is-laravel>. (дата звернення: 13.12.2023).
41. База даних MySQL. URL: <https://promoter.net.ua/articles/baza-danix-mysql.html>. (дата звернення: 13.12.2023).
42. Training a custom dlib shape predictor. URL: <https://www.pyimagesearch.com/2019/12/16/training-a-custom-dlib-shapepredictor/> (дата звернення: 17.11.2023).
43. Flask's documentation. URL: <https://flask.palletsprojects.com/en/2.0.x/> (дата звернення: 17.11.2023).

ДОДАТОК А. ПЛАНУВАННЯ РОБІТ

1 ІДЕНТИФІКАЦІЯ ІДЕЇ ПРОЕКТУ

Мета роботи полягає у розробці Web-орієнтованої системи підтримки діяльності офтальмологічної клініки.

Дипломний проект розроблений для того, щоб пацієнт офтальмологічної клініки мав спрощений доступ до особистого кабінету, оформити талон до потрібного лікаря, вільний доступ до історії оглядів та рецептів, а також мати можливість віртуально приміряти оправу для того, щоб побачити, як вона буде виглядати на вашому обличчі.

Web-система реалізована як сайт, доступний через будь-який браузер. Система складається з підмодулів, доступ до яких реалізовано через верхню панель.

2 ДЕТАЛІЗАЦІЯ МЕТИ МЕТОДОМ SMART

Метою роботи є розробка Web-орієнтованої системи підтримки діяльності офтальмологічної клініки.

Дипломний проект орієнтований на розробку системи з спрощеним доступом до особистого кабінету пацієнта, де є можливість оформити талон до потрібного лікаря, вільний доступ до історії оглядів та рецептів, а також мати можливість віртуально приміряти оправу для того, щоб побачити, як вона буде виглядати на вашому обличчі не виходивши з дому.

Web-орієнтована система підтримки діяльності офтальмологічної клініки має бути удосконалена модулем «Підбору оправ» по різним параметрам користувачів та базою даних користувача з обліком відвідин клініки.

Технологія SMART (СМАРТ) є сучасним підходом для визначення якісних цілей, що дозволяє систематизувати всю доступну інформацію на етапі визначення мети. Вона спрямована на встановлення реалістичних строків для виконання завдань, визначення необхідних ресурсів та надання усім учасникам процесу чітких, точних і конкретних завдань. (табл. А.2.1).

Таблиця А.2.1 – Вичерпна мета методом SMART

Specific (конкретна)	Розробка «Web-орієнтованої системи підтримки діяльності офтальмологічної клініки».
Measurable (вимірювана)	Використовуючи мінімум ресурсів розробити якісний програмний продукт.
Achievable (досяжна)	Реалізація відбувається з допомогою мови програмування php та використання WHook.
Relevant (реалістична)	Всі потрібні апаратні та програмні засоби в наявності. Рівень кваліфікації фахівця достатній, щоб виконати поставлені задачі.
Time-framed (обмежена у часі)	Виконання роботи обмежене часовими рамками, задані замовником системи. Виконання проекту має відповідати календарному плану

Джерело: побудовано автором

3 ОПИСАННЯ ФАЗИ РОЗРОБКИ ІТ—ПРОЕКТУ

3.1 Планування змісту структури робіт ІТ—проекту (WBS)

Ієрархічна структура робіт (Work Breakdown Structure, WBS) є ефективним інструментом, спрямованим на розподіл завдань та функцій в межах проекту. Цей

інструмент визначає ієрархічну структуру, яка детально розкриває всі етапи та компоненти проекту, щоб забезпечити його успішне виконання. Кожен рівень ієрархії представляє собою підрозділ проекту на більш дрібні завдання, що робить його більш керованим та контрольованим.

Такий підхід дозволяє керівникам проектів та учасникам з легкістю розуміти структуру та обсяг проекту, а також визначати відповідальності за кожен конкретний елемент. Ієрархічна структура робіт є важливим інструментом для планування, виконання та моніторингу проектів, допомагаючи забезпечити організоване та ефективне виконання всіх завдань в рамках проекту.

Ієрархічна структура робіт переважно є переліком завдань проекту та може мати форму графічного зображення або текстового опису, який вказує на включення робіт. Цей інструмент організовує та визначає всі складові проекту.

Зробимо побудову ієрархічної структури робіт, вказуючи всі виконувані завдання відповідно до головних етапів. На рисунку А.3.1 представлена діаграма WBS, що складається з наступних складових:

- 1) Створення технічного завдання - процес розробки документа, який визначає основні цілі, значення якості, деталізовані вимоги технічного і економічного характеру до виконуваної системи. Оформлення технічного завдання складається з предметної області, вибором мови програмування, призначення цільової аудиторії та вимог до складових модулів web-системи;

- 2) Розробка web-системи – написання програмного коду всіх необхідних модулів, що необхідні для працездатності web-системи;

- 3) Тестування - етап, на якому проводиться коду програми та виявлення можливих помилок;

- 4) Завершення - заключний етап виконання проекту, який є заключним етапом розробки та випуску готової системи.

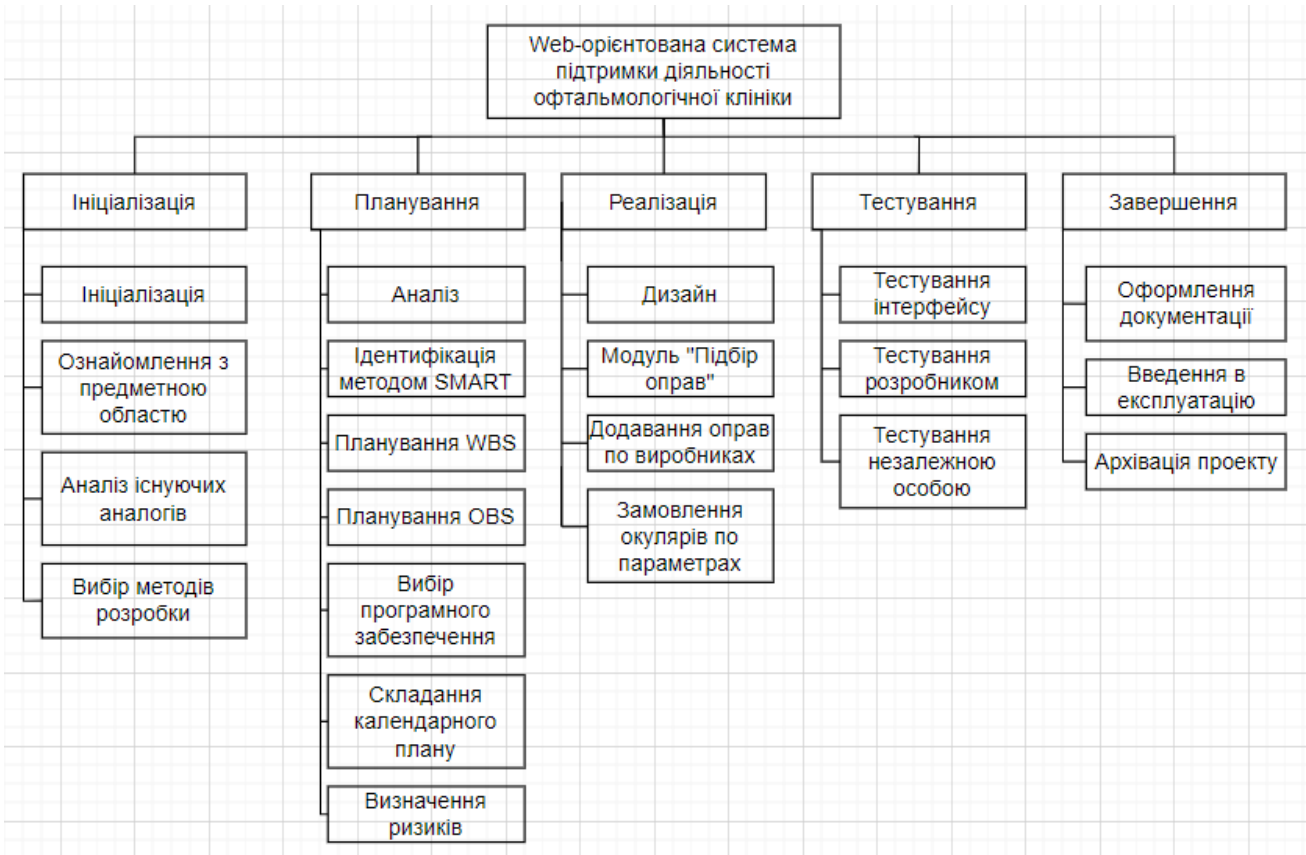


Рисунок А.3.1 – WBS-структура Web додатку офтальмологічної клініки

Джерело: побудовано автором

3.2 Планування структури організації готового проекту

OBS-структура проекту представляє собою організаційну структуру виконавців або організацій, які беруть участь у реалізації проекту. Ця структура формується на основі переліку етапів робіт кожного напрямку WBS (структури розбиття робіт). Вона визначає відповідальних за виконання конкретних пакетів робіт, де вагомими - є обов'язково керівники організацій, а скоріше ті особи, які прямо організують виконання задач.

Організаційна структура представляється графічно і відображає учасників проекту та їхні ролі в реалізації проекту. На верхньому рівні OBS знаходиться команда проекту. На наступних рівнях фіксуються виконавці, такі як організації

або відділи, і для кожного виконавця вказуються конкретні особи, відповідальні за елементарні роботи WBS.

Важливо зауважити, що відповідальні особи не обов'язково є керівниками; це можуть бути співробітники, які безпосередньо керують виконанням конкретних елементарних завдань WBS. Після побудови структури розбиття робіт WBS можна розробити організаційну структуру виконавців OBS, в якій відображено всі відповідальні особи та організації, задіяні у проекті. Важливо відзначити, що організаційна структура проекту стосується внутрішньої організації проекту і не враховує відносин між учасниками проекту та їхніми батьківськими організаціями. Діаграма OBS зображена на рисунку А.3.2.

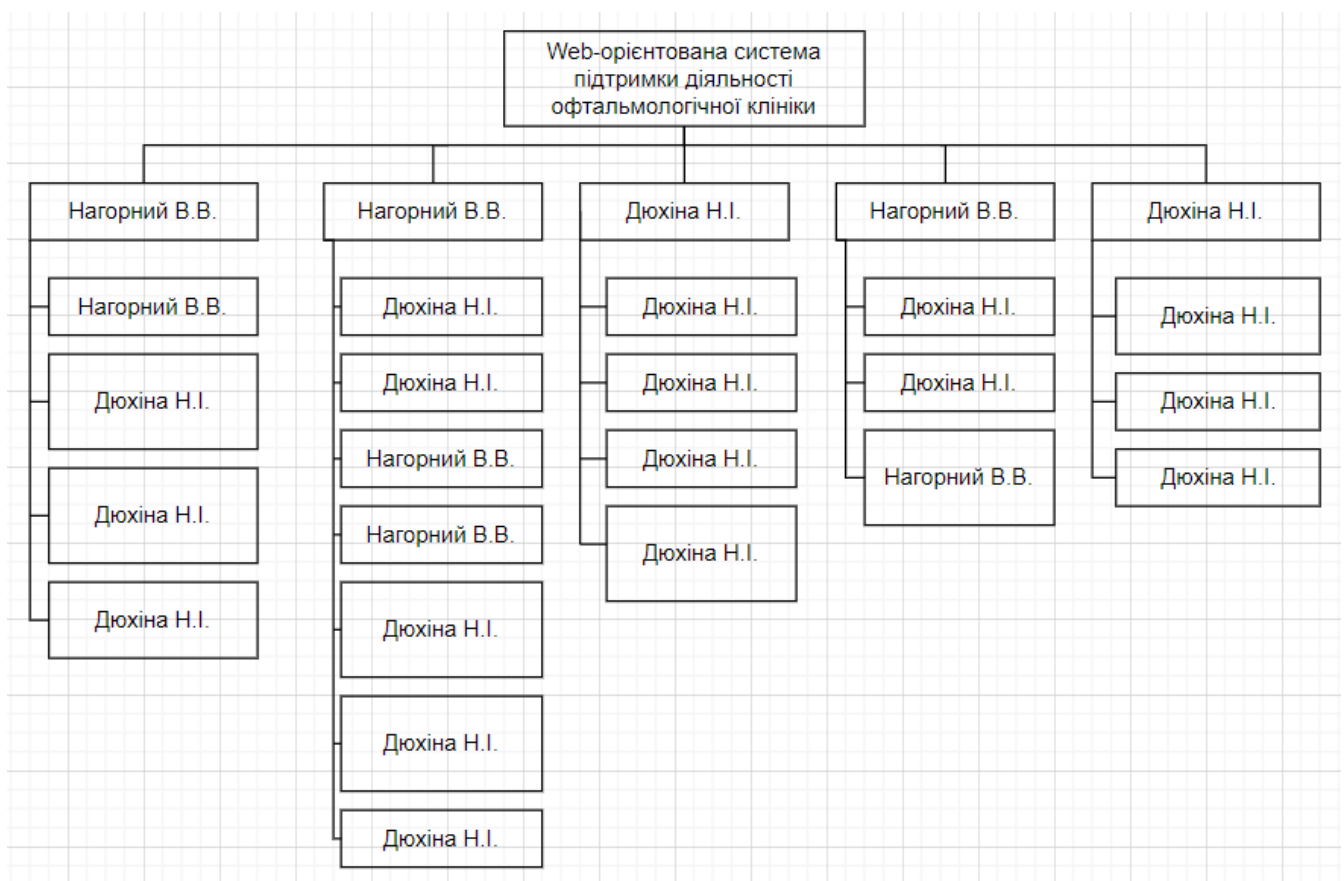


Рисунок А.3.2 – Організаційна структура виконавців

Джерело: побудовано автором

3.3 Побудова матриці виконавців пакетів робіт

На підставі OBS та WBS структур було створено список виконавців, що функціонують в проекті (табл. А.3.1).

Таблиця А.3.1 – Виконавці проекту

Фази	Виконавець	Виконавець
	Дюхіна Н.І.	Нагорний В.В.
Аналіз предметної області	+	-
Формування ТЗ	+	+
Мета проекту	+	-
WBS-структура	+	+
Календарний план проекту	+	+
Управління ресурсами	+	-
Управління ризиками	+	-
Управління якістю	+	-
Проектування дизайну інтернет магазину	+	-
Розробка шаблону сайту	+	-
Налаштування WordPress	+	-
Тестування	+	-
Інструкція користувача	+	-
Здача в експлуатацію	+	+

Джерело: побудовано автором

4 ПОБУДОВА КАЛЕНДАРНОГО ГРАФІКУ ВИКОНАННЯ ПРОЕКТУ

Діаграма Ганта - це інструмент управління проектами, який використовується для візуалізації та відстеження прогресу проекту. Діаграма Ганта складається з горизонтальної шкали часу, на якій відображаються завдання проекту. Кожне завдання представлено вертикальною смужкою, яка починається з дати початку завдання і закінчується датою закінчення. Довжина смужки відповідає тривалості завдання.

Для отримання реального уявлення про тривалість виконання робіт з урахуванням обмеженості у використанні ресурсів, було побудовано календарний графік робіт, що показано на рисунках А.4.1 та А.4.2.

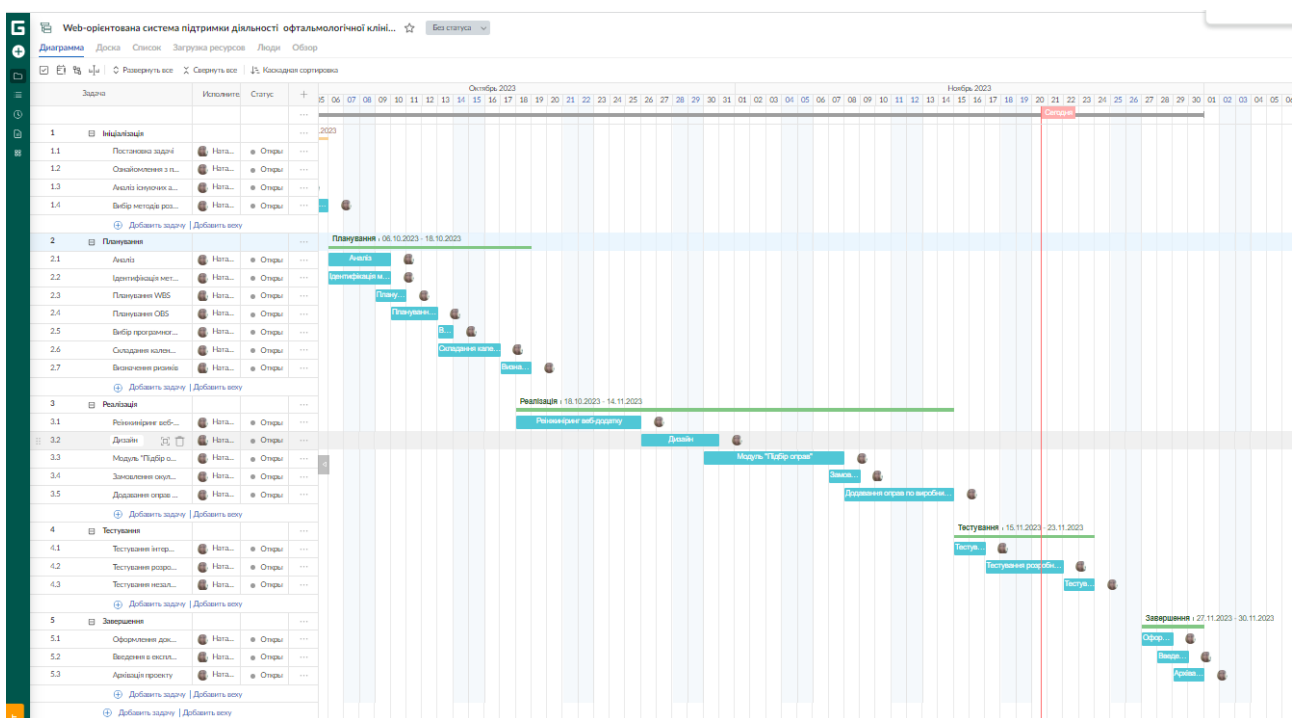


Рисунок А.4.1 – Діаграма Ганта

Джерело: побудовано автором

Web-орієнтована система підтримки діяльності офтальм

Діаграма Доска Список Загрузка ресурсов Люди Обзор

Развернуть все Свернуть все Каскадная сортировка

Задача	Исполнитель	Статус	
1	Иніціалізація		...
1.1	Постановка задачі	Ната...	Откры
1.2	Ознайомлення з п...	Ната...	Откры
1.3	Аналіз існуючих а...	Ната...	Откры
1.4	Вибір методів роз...	Ната...	Откры
+ Добавить задачу Добавить веку			
2	Планування		...
2.1	Аналіз	Ната...	Откры
2.2	Ідентифікація мет...	Ната...	Откры
2.3	Планування WBS	Ната...	Откры
2.4	Планування OBS	Ната...	Откры
2.5	Вибір програмног...	Ната...	Откры
2.6	Складання кален...	Ната...	Откры
2.7	Визначення ризиків	Ната...	Откры
+ Добавить задачу Добавить веку			
3	Реалізація		...
3.1	Рейтингірує веб...	Ната...	Откры
3.2	Дизайн	Ната...	Откры
3.3	Модуль "Підбір о...	Ната...	Откры
3.4	Замовлення скул...	Ната...	Откры
3.5	Додавання опрае...	Ната...	Откры
+ Добавить задачу Добавить веку			
4	Тестування		...
4.1	Тестування інтер...	Ната...	Откры
4.2	Тестування розро...	Ната...	Откры
4.3	Тестування незал...	Ната...	Откры
+ Добавить задачу Добавить веку			
5	Завершення		...
5.1	Оформлення док...	Ната...	Откры
5.2	Введення в експл...	Ната...	Откры
5.3	Архівація проекту	Ната...	Откры
+ Добавить задачу Добавить веку			
+ Добавить задачу Добавить веку			

Рисунок А.4.2 – Список робіт для побудови діаграми Ганта

Джерело: побудовано автором

5 ІДЕНТИФІКАЦІЯ РИЗИКІВ

Управління ризиками - це процес виявлення, оцінки, реагування та моніторингу ризиків, які можуть вплинути на проект.

Ідентифікація ризиків - це перший етап процесу управління ризиками. На цьому етапі необхідно визначити всі можливі ризики, які можуть вплинути на проект. Для цього можна використовувати різні методи, такі як:

1) Brainstorming - метод колективного обговорення, який дозволяє виявити широкий спектр ризиків;

2) Аналіз сценаріїв - метод, який передбачає розробку різних сценаріїв розвитку подій, що можуть вплинути на проект;

3) Аналіз статистичних даних - метод, який дозволяє оцінити ймовірність виникнення ризику на основі історичних даних;

4) Оцінювання ризиків - це другий етап процесу управління ризиками. На цьому етапі необхідно оцінити ймовірність виникнення кожного ризику та його потенційний вплив на проект. Для оцінки ймовірності виникнення ризику можна використовувати такі методи, як:

Для оцінки потенційного впливу ризику на проект можна використовувати такі методи, як:

- аналіз вартості реалізації (CV) - метод, який дозволяє оцінити вартість реалізації проекту з урахуванням ризику;

- аналіз часу реалізації (CT) - метод, який дозволяє оцінити час реалізації проекту з урахуванням ризику;

- аналіз якості реалізації (CQ) - метод, який дозволяє оцінити якість реалізації проекту з урахуванням ризику.

Аналіз різних видів ризиків представлено в таблиці А.5.1.[31]

Таблиця А.5.1 – Види ризику

№	Назва ризику	Опис ризику
1	Зовнішні ризики, які неможливо передбачити	Різноманітні природні катастрофи
		Різноманітна злочинна діяльність
		Непередбачувані зовнішні впливи соціальні або економічні
		Непередбачувані проблеми фінансового характеру
2	Зовнішні ризики, передбачувані	Зміна споживчих вимог; зміна ринкової привабливості та конкурентноспроможності.
		Ризики в середині системи: безпекові порушення; невідповідність проекту початковим цілям; зміна курсу валют, інфляція та інше.
3	Організаційні ризики.	Порушення терміну плану робіт
4	Технічні ризики системи	Технологічний прогрес
		Закладені ризики
		Помилки допущені в проектній документації
5	Інші типи ризики	Втрати фізичного майна
		Непередбачувані збитки пов'язані зі справністю обладнання та орендою.
6	За джерелами виникнення	Помилки розрахунків конкурентної та ринкової ситуації
		Не правильно визначена цільова аудиторія
		Порушення проектної документації
		Зміни орієнтирів, перепрофілювання

Джерело: побудовано автором

Періодичні ризики – повністю залежать від зовнішньої ситуації, внутрішніми суб'єктами не регулюються:

- ризики, пов'язані з змінами інвестиційного фонду (табл. А.5.2);
- зміни погодних умов, що викликаються катастрофами різного рівня;
- змінні соціально-політичні характеристики, та нестабільність політичних умов;
- постійні коливання рівня валют.[28]

Таблиця А.5.2 – Види прояву ризику

№	Назва ризику	Опис ризику
1	За галузями виявлення	економічні
		політичні
		соціальні
		екологічні
2	За видами втрат	трудові
		невідповідність часу виконання до плану проекту
		фінансові
		перевищення витрат в зв'язку з неправильними технічним розрахунками
		соціальні
		ризик нежиттєздатності проекту
		ризики невиплати боргів
		податкові
		виробничі

Джерело: побудовано автором

Головним параметром є ймовірність виникнення ступеня впливу. Даний розрахунок дозволить розрахувати ранг ризику, як можливий варіант впливу всіх

видів ризиків на функціонування системи. Ранг ризику розраховується, як добуток ймовірності виникнення та ступеню впливу:

$$RV = P * I,$$

де RV – ранг ризику;

P – ймовірність виникнення;

I – ступінь впливу.

У такій ситуації першим етапом є визначити прогнозовані ризики (табл. А.5.3).

Таблиця А.5.3 – Ризики проекту

№	Опис ризику	Вплив	Ймовірність	RV	Пом'якшення наслідків
1	Нестабільність роботи програми	5	3	С	У роботі користуватися лише перевіреним програмним забезпеченням
2	Некоректна робота web-системи	4	2	М	Перевірка та тестування web-системи
3	Неактуальність web-системи	3	2	М	Складати систему на основі аналізу сучасних підходів
4	Додаткові вимоги	4	3	М	Складання деталізованого технічного завдання відповідно до вимог замовника
5	Фінансові затримки	3	1	Л	Точний фінансовий план

Джерело: побудовано автором

- Зелений колір – допустимі ризики.
- Жовтий колір – оправдані ризики.
- Червоний колір – заборонені ризики.

Шкала оцінювання ризиків має п'ять головних критеріїв:

- 5 балів - критичний ризик.
- 4 бали - максимальний ризик.
- 3 бали - високий ризик.
- 2 бали - нормальний ризик.
- 1 бал - малий ризик.

На підставі отриманого значення індексу ризику класифікують: за рівнем ризику, що знаходиться в таблиці А.5.4.

Таблиця А.5.4 – Шкала оцінювання рівнів ризиків

№	Назва	Межі	Ризики, які входять (номера)
1	Допустимі	$1 \leq R \leq 2$	1,8,11,12,13
2	Оправдані	$3 \leq R \leq 4$	2,4,6,10,15
3	Заборонені	$6 \leq R \leq 9$	3,5,7,9,14

Джерело: побудовано автором

ДОДАТОК Б. ЛІСТИНГ ПРОГРАМНОГО КОДУ

1. Початкова сторінка web-системі

```

<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8"/>
  <meta http-equiv="X-UA-Compatible" content="IE=edge"/>
  <meta name="viewport" content="width=device-width, initial-scale=1.0"/>

  <link rel="stylesheet" href="{{ asset('build/app.css') }}">
  <script src="{{ asset('build/app2.js') }}"></script>
  <link rel="icon" type="image/x-icon" href="{{ asset('favicon.ico') }}">
  <title>ОФТАЛЬМОЛОГИЧНИЙ ЦЕНТР "ТОЧКА ЗОРУ"</title>
</head>

<body>
<header class="flex items-center justify-between p-6">
  <a href="" class="flex items-center gap-2">
    <svg class="h-10 text-green-600" xmlns="http://www.w3.org/2000/svg"
viewBox="0 0 24 24" fill="currentColor">
      <path fill-rule="evenodd"
d="M12.516 2.17a.75.75 0 00-1.032 0 11.209 11.209 0 01-7.877
3.08.75.75 0 00-.722.515A12.74 12.74 0 002.25 9.75c0 5.942 4.064 10.933 9.563
12.348a.749.749 0 00.374 0c5.499-1.415 9.563-6.406 9.563-12.348 0-1.39-.223-2.73-
.635-3.985a.75.75 0 00-.722-.516l-.143.001c-2.996 0-5.717-1.17-7.734-3.08zm3.094
8.016a.75.75 0 10-1.22-.872l-3.236 4.53L9.53 12.22a.75.75 0 00-1.06 1.06l2.25
2.25a.75.75 0 001.14-.094l3.75-5.25z"
clip-rule="evenodd"/>
    </svg>
    <span class="text-xl font-black">ОФТАЛЬМОЛОГИЧНИЙ ЦЕНТР "ТОЧКА
ЗОРУ"</span>
  </a>
  <div>
    @guest
      <a href="{{ route('billing') }}" class="rounded-md bg-green-600
py-2 px-4 font-semibold text-white shadow-lg transition duration-150 ease-in-out
hover:bg-green-700 hover:shadow-xl focus:shadow-xl focus:outline-none focus:ring-2
focus:ring-green-500 focus:ring-offset-2">Запис на прийом</a>
      <a href="{{ route('login') }}"
class="rounded-md bg-gray-200 py-2 px-4 font-semibold text-
gray-900 shadow-lg transition duration-150 ease-in-out hover:bg-gray-300
hover:shadow-xl focus:shadow-xl focus:outline-none focus:ring-2 focus:ring-gray-300
focus:ring-offset-2">Увійти</a>
      <a href="{{ route('register') }}"
class="rounded-md bg-gray-200 py-2 px-4 font-semibold text-
gray-900 shadow-lg transition duration-150 ease-in-out hover:bg-green-700
hover:shadow-xl focus:shadow-xl focus:outline-none focus:ring-2 focus:ring-green-
500 focus:ring-offset-2">Зареєструватись</a>
    @endguest
    @auth
      <a href="{{ route('dashboard') }}"
class="rounded-md bg-gray-200 py-2 px-4 font-semibold text-
gray-900 shadow-lg transition duration-150 ease-in-out hover:bg-gray-300
hover:shadow-xl focus:shadow-xl focus:outline-none focus:ring-2 focus:ring-gray-300
focus:ring-offset-2">Особистий

```

```

        кабинет</a>
    @endauth
</div>
</header>
<div class="bg-white relative">
    <div class="mx-auto grid max-w-2xl grid-cols-1 items-center gap-x-8 gap-
y-16 px-4 py-24 sm:px-6 sm:py-32 lg:max-w-7xl lg:grid-cols-2 lg:px-8">
        <div>
            <h2 class="text-3xl font-bold tracking-tight text-gray-900
sm:text-4xl">Офтальмологічний центр</h2>
            <p class="mt-4 text-gray-500">Найсучасніший арсенал обладнання від
ведучих світових брендів. <br/>Продіагностовано за час роботи більше 20 000 тисяч
пацієнтів. <br/>Прооперовано більше 3000 тисяч за останні 10 років.</p>

            <dl class="mt-16 grid grid-cols-1 gap-x-6 gap-y-10 sm:grid-cols-2
sm:gap-y-16 lg:gap-x-8">
                <div class="border-t border-gray-200 pt-4">
                    <dt class="font-medium text-gray-900">Діагностика
зору</dt>
                    <dd class="mt-2 text-sm text-gray-500">Комп'ютерна
діагностика зору - це перший крок для збереження зору</dd>
                </div>
                <div class="border-t border-gray-200 pt-4">
                    <dt class="font-medium text-gray-900">Лікування
катаракти</dt>
                    <dd class="mt-2 text-sm text-gray-500">При лікуванні
катаракти ока застосовуються самі сучасні і технологічні методики лікування</dd>
                </div>
                <div class="border-t border-gray-200 pt-4">
                    <dt class="font-medium text-gray-900">Лазерна корекція
зору</dt>
                    <dd class="mt-2 text-sm text-gray-500">Офтальмологічний
центр уже понад 20 років успішно застосовує самі ефективні, надійні і безпечні способ
виправлення короткозорості</dd>
                </div>
                <div class="border-t border-gray-200 pt-4">
                    <dt class="font-medium text-gray-900">Лікування
глаукоми</dt>
                    <dd class="mt-2 text-sm text-gray-500">Глаукома - це
серйозне захворювання, при якому пошкоджується зоровий нерв</dd>
                </div>
                <div class="border-t border-gray-200 pt-4">
                    <dt class="font-medium text-gray-900">Підбір окулярів і
контактних лінз</dt>
                    <dd class="mt-2 text-sm text-gray-500">Для тих людей, хто
обирає окуляри і контактні лінзи як спосіб вирішення проблем із зором, вчені клініки
підберуть оптичну корекцію.</dd>
                </div>
                <div class="border-t border-gray-200 pt-4">
                    <dt class="font-medium text-gray-900">Дитяча
офтальмологія</dt>
                    <dd class="mt-2 text-sm text-gray-500">Для вирішення
дитячих проблем із зором клініка створила спеціалізований центр дитячої
офтальмології</dd>
                </div>
            </dl>
        </div>
    </div>
</div class="grid grid-cols-2 grid-rows-2 gap-4 sm:gap-6 lg:gap-8">

```

```

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

<div class="bg-white py-24 sm:py-32">
    <div class="mx-auto grid max-w-7xl gap-x-8 gap-y-20 px-6 lg:px-8 xl:grid-cols-3">
        <div class="max-w-2xl">
            <h2 class="text-3xl font-bold tracking-tight text-gray-900 sm:text-4xl">Наша команда</h2>
            <p class="mt-6 text-lg leading-8 text-gray-600">Для реалізації місії в штаті сьогодні працюють висококваліфіковані фахівців, в тому числі лікарі вищої та першої категорій, кандидати та доктора медичних наук.</p>
        </div>
        <ul role="list" class="grid gap-x-8 gap-y-12 sm:grid-cols-2 sm:gap-y-16 xl:col-span-2">
            <li>
                <div class="flex items-center gap-x-6">
                    
                    <div>
                        <h3 class="text-base font-semibold leading-7 tracking-tight text-gray-900">Ліфшиць Юрій Зіновійович</h3>
                        <p class="text-sm font-semibold leading-6 text-indigo-600">Офтальмохірург вищої категорії</p>
                    </div>
                </div>
            </li>
            <li>
                <div class="flex items-center gap-x-6">
                    
                    <div>
                        <h3 class="text-base font-semibold leading-7 tracking-tight text-gray-900">Гіндіч Ольга Андріївна</h3>
                        <p class="text-sm font-semibold leading-6 text-indigo-600">Лікар - офтальмолог вищої категорії</p>
                    </div>
                </div>
            </li>
            <li>
                <div class="flex items-center gap-x-6">
                    
                    <div>
                        <h3 class="text-base font-semibold leading-7 tracking-tight text-gray-900">Ратушняк Наталя Яроставівна</h3>
                        <p class="text-sm font-semibold leading-6 text-indigo-600">Лікар - офтальмолог, лазерний мікрохірург</p>
                    </div>
                </div>
            </li>
        </ul>
    </div>
</div>

```

```

        </div>
    </li>
    <li>
        <div class="flex items-center gap-x-6">
            
            <div>
                <h3 class="text-base font-semibold leading-7
tracking-tight text-gray-900">Мельник Софія Ігорівна</h3>
                <p class="text-sm font-semibold leading-6 text-
indigo-600">Лікар – офтальмолог. Рефракційний мікрохірург.</p>
            </div>
        </div>
    </li>
    <!-- More people... -->
</ul>
</div>
</div>

</body>

</html>

```

2. Код сторінки пацієнта та функціональної панелі

```

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8"/>
    <meta http-equiv="X-UA-Compatible" content="IE=edge"/>
    <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
    <link rel="stylesheet" href="{{ asset('build/app.css') }}">
    <script src="{{ asset('build/app2.js') }}"></script>
    <link rel="icon" type="image/x-icon" href="{{ asset('favicon.ico') }}">
    <title>Security</title>
</head>

<body class="min-h-screen bg-gray-50">
<header class="flex items-center justify-between p-6">
    <a href="{{ route('welcome') }}" class="flex items-center gap-2">
        <svg class="h-10 text-green-600" xmlns="http://www.w3.org/2000/svg"
viewBox="0 0 24 24" fill="currentColor">
            <path fill-rule="evenodd"
                d="M12.516 2.17a.75.75 0 00-1.032 0 11.209 11.209 0 01-7.877
3.08.75.75 0 00-.722.515A12.74 12.74 0 002.25 9.75c0 5.942 4.064 10.933 9.563
12.348a.749.749 0 00.374 0c5.499-1.415 9.563-6.406 9.563-12.348 0-1.39-.223-2.73-
.635-3.985a.75.75 0 00-.722-.516l-.143.001c-2.996 0-5.717-1.17-7.734-3.08zm3.094
8.016a.75.75 0 10-1.22-.872l-3.236 4.53L9.53 12.22a.75.75 0 00-1.06 1.06l2.25
2.25a.75.75 0 001.14-.094l3.75-5.25z"
                clip-rule="evenodd"/>
            </svg>
            <span class="text-xl font-black">ОФТАЛЬМОЛОГИЧНИЙ ЦЕНТР "ТОЧКА
ЗОРУ"</span>
        </a>
    <div class="flex gap-2">
        <a href="{{ route('billing') }}">

```

```

        class="rounded-md bg-green-600 py-2 px-4 font-semibold text-white
shadow-lg transition duration-150 ease-in-out hover:bg-green-700 hover:shadow-xl
focus:shadow-xl focus:outline-none focus:ring-2 focus:ring-green-500 focus:ring-
offset-2">Запис
        на прийом</a>
        <a href="{{ route('history') }}"
        class="rounded-md bg-green-600 py-2 px-4 font-semibold text-white
shadow-lg transition duration-150 ease-in-out hover:bg-green-700 hover:shadow-xl
focus:shadow-xl focus:outline-none focus:ring-2 focus:ring-green-500 focus:ring-
offset-2">Відвідини</a>
        <a href="{{ route('orders') }}"
        class="rounded-md bg-green-600 py-2 px-4 font-semibold text-white
shadow-lg transition duration-150 ease-in-out hover:bg-green-700 hover:shadow-xl
focus:shadow-xl focus:outline-none focus:ring-2 focus:ring-green-500 focus:ring-
offset-2">Рецепти</a>
        <a href="{{ route('check') }}"
        class="rounded-md bg-green-600 py-2 px-4 font-semibold text-white
shadow-lg transition duration-150 ease-in-out hover:bg-green-700 hover:shadow-xl
focus:shadow-xl focus:outline-none focus:ring-2 focus:ring-green-500 focus:ring-
offset-2">Підбір
        оправи</a>
        <form method="post" action="{{ route('logout') }}" class="flex">
            @csrf

            <a href="{{ route('logout') }}" onclick="event.preventDefault();
this.closest('form').submit();"
            class="rounded-md bg-gray-200 py-2 px-4 font-semibold text-
gray-900 shadow-lg transition duration-150 ease-in-out hover:bg-gray-300
hover:shadow-xl focus:shadow-xl focus:outline-none focus:ring-2 focus:ring-gray-300
focus:ring-offset-2">Вийти</a>
        </form>
    </div>
</header>
<main>
    <div class="m-6 mb-12 rounded-xl p-6 shadow-xl sm:p-10">
        <h1 class="text-3xl font-semibold">Кабінет@if($type ==
'billing')
            - {{ $title }} -
            @endif {{ $user->name }}</h1>
    </div>
    <div class="m-6 mb-12 rounded-xl p-6 shadow-xl sm:p-10">
        <h2 class="text-2xl font-semibold">Ваші призначення:</h2>
        <table class="border-collapse border border-green-800">
            <thead>
                <tr>
                    <th class="border border-green-800 px-4 py-2">Дата</th>
                    <th class="border border-green-800 px-4 py-2">Лікар</th>
                    <th class="border border-green-800 px-4 py-
2">Призначення</th>
                </tr>
            </thead>
            <tbody>
                @foreach($orders as $order)
                    <tr>
                        <td class="border border-green-800 px-4 py-2">{{
$order->date }}</td>
                        <td class="border border-green-800 px-4 py-2">{{
$order->doctor }}</td>
                        <td class="border border-green-800 px-4 py-2">{{
$order->description }}</td>
                    </tr>
                </tbody>
            </table>
    </div>

```

```

        </tr>
      @endforeach
    </tbody>
  </table>
</div>
</main>
</body>

</html>

```

3. Модуль підбору оправи

```

<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <link rel="stylesheet" href="{{ asset('build/app.css') }}">
  <script src="{{ asset('build/app2.js') }}"></script>
  <link rel="icon" type="image/x-icon" href="{{ asset('favicon.ico') }}">
  <title>Security</title>
</head>

<body class="min-h-screen bg-gray-50">
<header class="flex items-center justify-between p-6">
  <a href="{{ route('welcome') }}" class="flex items-center gap-2">
    <svg class="h-10 text-green-600" xmlns="http://www.w3.org/2000/svg"
viewBox="0 0 24 24" fill="currentColor">
      <path fill-rule="evenodd" d="M12.516 2.17a.75.75 0 00-1.032 0
11.209 11.209 0 01-7.877 3.08.75.75 0 00-.722 5.15A12.74 12.74 0 002.25 9.75c0 5.942
4.064 10.933 9.563 12.348a.749.749 0 00.374 0c5.499-1.415 9.563-6.406 9.563-12.348
0-1.39-.223-2.73-.635-3.985a.75.75 0 00-.722-.516l-.143.001c-2.996 0-5.717-1.17-
7.734-3.08zm3.094 8.016a.75.75 0 10-1.22-.872l-3.236 4.53L9.53 12.22a.75.75 0 00-
1.06 1.06l2.25 2.25a.75.75 0 001.14-.094l3.75-5.25z" clip-rule="evenodd" />
    </svg>
    <span class="text-xl font-black">ОФТАЛЬМОЛОГИЧНИЙ ЦЕНТР "ТОЧКА
ЗОРУ"</span>
  </a>
  <div class="flex gap-2">
    <a href="{{ route('billing') }}" class="rounded-md bg-green-600 py-2
px-4 font-semibold text-white shadow-lg transition duration-150 ease-in-out
hover:bg-green-700 hover:shadow-xl focus:shadow-xl focus:outline-none focus:ring-2
focus:ring-green-500 focus:ring-offset-2">Запис на прийом</a>
    <a href="{{ route('history') }}" class="rounded-md bg-green-600 py-2
px-4 font-semibold text-white shadow-lg transition duration-150 ease-in-out
hover:bg-green-700 hover:shadow-xl focus:shadow-xl focus:outline-none focus:ring-2
focus:ring-green-500 focus:ring-offset-2">Відвідини</a>
    <a href="{{ route('orders') }}" class="rounded-md bg-green-600 py-2
px-4 font-semibold text-white shadow-lg transition duration-150 ease-in-out
hover:bg-green-700 hover:shadow-xl focus:shadow-xl focus:outline-none focus:ring-2
focus:ring-green-500 focus:ring-offset-2">Рецепти</a>
    <a href="{{ route('check') }}" class="rounded-md bg-green-600 py-2 px-
4 font-semibold text-white shadow-lg transition duration-150 ease-in-out hover:bg-
green-700 hover:shadow-xl focus:shadow-xl focus:outline-none focus:ring-2
focus:ring-green-500 focus:ring-offset-2">Підбір оправи</a>
    <form method="post" action="{{ route('logout') }}" class="flex">

```



```

@csrf

    <a href="{{ route('logout') }}" onclick="event.preventDefault();
this.closest('form').submit();" class="rounded-md bg-gray-200 py-2 px-4 font-
semibold text-gray-900 shadow-lg transition duration-150 ease-in-out hover:bg-gray-
300 hover:shadow-xl focus:shadow-xl focus:outline-none focus:ring-2 focus:ring-
gray-300 focus:ring-offset-2">Вийти</a>
    </form>
</div>
</header>
<main>
    <div class="m-6 mb-12 rounded-xl p-6 shadow-xl sm:p-10">
        <h1 class="text-3xl font-semibold">Кабінет - Підбір оправи</h1>
    </div>

    <div class="m-6 mb-12 rounded-xl">
        <div class="js-step-1">
            <div class="mb-2 flex flex-wrap md:flex-nowrap items-center
justify-between gap-2 w-full">
                <h2 class="text-2xl font-semibold">1. Завантажте фотографію
обличчя</h2>
                <div class="flex flex-wrap justify-end gap-2">
                    <button type="button" class="js-step-1-btn-cancel
rounded-md bg-gray-200 py-2 px-4 font-semibold text-gray-900 shadow-lg transition
duration-150 ease-in-out hover:bg-gray-300 hover:shadow-xl focus:shadow-xl
focus:outline-none focus:ring-2 focus:ring-gray-300 focus:ring-offset-
2">Скасувати</button>
                    <button type="button" class="js-step-1-btn-continue
rounded-md bg-green-600 py-2 px-4 font-semibold text-white shadow-lg transition
duration-150 ease-in-out hover:bg-green-700 hover:shadow-xl focus:shadow-xl
focus:outline-none focus:ring-2 focus:ring-green-500 focus:ring-offset-2
disabled:opacity-50" disabled>Завантажити фото</button>
                </div>
            </div>
            <div class="flex items-center justify-center w-full relative">
                <label for="dropzone-file" class="js-step-1-dropzone flex
flex-col items-center justify-center w-full h-64 border-2 border-gray-300 border-
dashed rounded-lg cursor-pointer bg-gray-50">
                    <div class="flex flex-col items-center justify-center pt-
5 pb-6">
                        <svg class="w-8 h-8 mb-4 text-gray-500" aria-
hidden="true" xmlns="http://www.w3.org/2000/svg" fill="none" viewBox="0 0 20 16">
                            <path stroke="currentColor" stroke-
linecap="round" stroke-linejoin="round" stroke-width="2" d="M13 13h3a3 3 0 0 0 0-
6h-.025A5.56 5.56 0 0 0 16 6.5 5.5 5.5 0 0 0 5.207 5.021C5.137 5.017 5.071 5 5 5a4
4 0 0 0 0 8h2.167M10 15V6m0 0L8 8m2-2 2 2"/>
                            </svg>
                            <p class="mb-2 text-sm text-gray-500"><span
class="font-semibold">Натисніть, щоб обрати файл</span></p>
                            <p class="text-xs text-gray-500">PNG, JPG або WEBP</p>
                        </div>
                    <input id="dropzone-file" type="file" class="js-step-1-
file-input hidden" />
                </label>
                <img class="js-step-1-preview hidden absolute w-auto h-full
max-h-full top-[50%] left-[50%] -translate-x-1/2 -translate-y-1/2" src="" />
            </div>
        </div>

        <div class="js-step-2 hidden">

```

```

        <div class="mb-2 flex flex-wrap md:flex-nowrap items-center
justify-between gap-2 w-full">
            <h2 class="text-2xl font-semibold">2. Виділіть овал
обличчя</h2>
            <div class="flex justify-end gap-2">
                <button type="button" class="js-step-2-btn-cancel
rounded-md bg-gray-200 py-2 px-4 font-semibold text-gray-900 shadow-lg transition
duration-150 ease-in-out hover:bg-gray-300 hover:shadow-xl focus:shadow-xl
focus:outline-none focus:ring-2 focus:ring-gray-300 focus:ring-offset-2">Вибрати
інше фото</button>
                    <button type="button" class="js-step-2-btn-continue
rounded-md bg-green-600 py-2 px-4 font-semibold text-white shadow-lg transition
duration-150 ease-in-out hover:bg-green-700 hover:shadow-xl focus:shadow-xl
focus:outline-none focus:ring-2 focus:ring-green-500 focus:ring-offset-2">Підібрати
оправу</button>
            </div>
        </div>
        <div class="flex items-center justify-center w-full">
            <canvas class="js-step-2-canvas bg-black aspect-square max-w-
full" width="720" height="720">
                Canvas api not supported
            </canvas>
        </div>
    </div>

    <div class="js-step-3 hidden">
        <div class="mb-2 flex flex-wrap md:flex-nowrap items-center
justify-between gap-2 w-full">
            <h2 class="text-2xl font-semibold">3. Підберіть оправу зі
списку</h2>
            <div class="flex justify-end gap-2">
                <button type="button" class="js-step-3-btn-cancel
rounded-md bg-gray-200 py-2 px-4 font-semibold text-gray-900 shadow-lg transition
duration-150 ease-in-out hover:bg-gray-300 hover:shadow-xl focus:shadow-xl
focus:outline-none focus:ring-2 focus:ring-gray-300 focus:ring-offset-2">Виділити
обличчя</button>
                    <button type="button" class="js-step-3-btn-continue
rounded-md bg-green-600 py-2 px-4 font-semibold text-white shadow-lg transition
duration-150 ease-in-out hover:bg-green-700 hover:shadow-xl focus:shadow-xl
focus:outline-none focus:ring-2 focus:ring-green-500 focus:ring-offset-2">Зберегти
результат</button>
            </div>
        </div>
        <div class="flex items-center justify-center w-full">
            <div class="flex flex-col sm:flex-row gap-4">
                <canvas class="js-step-3-canvas order-1 sm:order-2 bg-
black aspect-square max-w-full" width="720" height="720">
                    Canvas api not supported
                </canvas>
                <div class="js-step-3-frames flex md:flex-col order-2
sm:order-1 gap-2">
                    
                        
                            

```

```

        </div>
    </div>
</div>
</div>
</div>
</div>
<script type="text/javascript">
    (() => {
        let activeHandle, activeGlasses, glassesHeight = 0;
        let mouseX = 0, mouseY = 0, mouseDown = false;
        let step2LoopEnabled = false;
        let step3LoopEnabled = false;
        let userPhoto = new Image();
        let userPhotoRect = {};

        let userFrame = {
            x: 0,
            y: 0,
            width: 0,
            height: 0,
            sx: 0,
            sy: 0,
            active: false,
        };

        let userHandles = [
            { type: 'top-left', x: 0, y: 0, sx: 0, sy: 0, active: false
},
            { type: 'top-right', x: 0, y: 0, sx: 0, sy: 0, active: false
},
            { type: 'bottom-right', x: 0, y: 0, sx: 0, sy: 0, active:
false },
            { type: 'bottom-left', x: 0, y: 0, sx: 0, sy: 0, active: false
},
        ];

        const canvasW = 720;
        const canvasH = 720;
        const handleW = 40;
        const handleH = 40;
        const handleColor = '#cccccc';
        const handleBorder = '#0036ff';
        const handleBorderActive = '#ffc900';
        const frameColor = '#3bff00';

        function pointRectCollision(px, py, rx, ry, rw, rh) {
            return px >= rx && px <= rx + rw && py >= ry && py <= ry + rh;
        }

        function fitRectInScreen(screenW, screenH, rectW, rectH) {
            let rectAR = rectW / rectH;
            let screenAR = screenW / screenH;
            let width = screenW;
            let height = screenH;

            if (rectAR > screenAR) {
                height = width / rectAR;
            } else {
                width = height * rectAR;
            }
        }
    });

```

```

    return {
        width,
        height,
        x: (screenW - width) / 2,
        y: (screenH - height) / 2,
    };
}

function resetUserFrame() {
    userFrame.width = 150;
    userFrame.height = 200;
    userFrame.x = (canvasW / 2) - (userFrame.width / 2);
    userFrame.y = (canvasH / 2) - (userFrame.height / 2);
}

function resetUserHandles() {
    userHandles.forEach((handle) => {
        switch (handle.type) {
            case 'top-left':
                handle.x = userFrame.x - handleW;
                handle.y = userFrame.y - handleH;

                break;

            case 'top-right':
                handle.x = userFrame.x + userFrame.width;
                handle.y = userFrame.y - handleH;

                break;

            case 'bottom-right':
                handle.x = userFrame.x + userFrame.width;
                handle.y = userFrame.y + userFrame.height;

                break;

            case 'bottom-left':
                handle.x = userFrame.x - handleW;
                handle.y = userFrame.y + userFrame.height;

                break;
        }
    });
}

function setActiveGlasses(element) {
    if (!element) return;

    step3Frames.querySelectorAll('img').forEach((img) => {
        img.classList.remove('border-2');
        img.classList.remove('border-green-600');
        img.classList.add('border');
        img.classList.add('border-black');
    });

    activeGlasses = element;
    activeGlasses.classList.remove('border');
    activeGlasses.classList.remove('border-black');
    activeGlasses.classList.add('border-2');
}

```

```

        activeGlasses.classList.add('border-green-600');

        glassesHeight = userFrame.width / (activeGlasses.naturalWidth
/ activeGlasses.naturalHeight);
    }

    function updateCanvasMouse(event, canvas) {
        const rect = canvas.getBoundingClientRect();
        const scaleX = canvas.width / rect.width;
        const scaleY = canvas.height / rect.height;
        if (event.targetTouches) {
            mouseX = (event.targetTouches[0].clientX - rect.left) *
scaleX;
            mouseY = (event.targetTouches[0].clientY - rect.top) *
scaleY;
        } else {
            mouseX = (event.clientX - rect.left) * scaleX;
            mouseY = (event.clientY - rect.top) * scaleY;
        }
    }

    function handleMouseDown(event, canvas) {
        updateCanvasMouse(event, canvas);

        if (canvas.classList.contains('js-step-2-canvas')) {
            userHandles.forEach((handle) => {
                if (!handle.active && pointRectCollision(mouseX,
mouseY, handle.x, handle.y, handleW, handleH)) {
                    activeHandle = handle;
                    handle.active = true;
                    handle.sx = mouseX - handle.x;
                    handle.sy = mouseY - handle.y;
                }
            });
        }

        if (!activeHandle && pointRectCollision(mouseX, mouseY,
userFrame.x, userFrame.y, userFrame.width, userFrame.height)) {
            userFrame.active = true;
            userFrame.sx = mouseX - userFrame.x;
            userFrame.sy = mouseY - userFrame.y;
        }
    }

    function handleMouseMove(event, canvas) {
        updateCanvasMouse(event, canvas);

        if (canvas.classList.contains('js-step-2-canvas')) {
            userHandles.forEach((handle) => {
                if (handle.active) {
                    let newX = 0;
                    let newY = 0;

                    switch (handle.type) {
                        case 'top-left':
                            newX = (mouseX - handle.sx) + handleW;
                            newY = (mouseY - handle.sy) + handleH;

                            userFrame.width += userFrame.x - newX;
                            userFrame.height += userFrame.y - newY;

```

```

        userFrame.x = newX;
        userFrame.y = newY;

        break;

    case 'top-right':
        newY = (mouseY - handle.sy) + handleH;

        userFrame.width = (mouseX - handle.sx) -
userFrame.x;

        userFrame.height += userFrame.y - newY;
        userFrame.y = newY;

        break;

    case 'bottom-right':
        userFrame.width = (mouseX - handle.sx) -
userFrame.x;

        userFrame.height = (mouseY - handle.sy) -
userFrame.y;

        break;

    case 'bottom-left':
        newX = (mouseX - handle.sx) + handleW;

        userFrame.width += userFrame.x - newX;
        userFrame.height = (mouseY - handle.sy) -
userFrame.y;

        userFrame.x = newX;

        break;
    }

    resetUserHandles();
}
});
}

if (userFrame.active) {
    userFrame.x = mouseX - userFrame.sx;
    userFrame.y = mouseY - userFrame.sy;

    resetUserHandles();
}
}

function handleMouseUp(event, canvas) {
    updateCanvasMouse(event, canvas);

    if (canvas.classList.contains('js-step-2-canvas')) {
        userHandles.forEach((handle) => {
            if (handle.active) {
                handle.sx = mouseX - handle.x;
                handle.sy = mouseY - handle.y;
            }

            handle.active = false;
        });
    }
}

```

```

        activeHandle = false;
    }

    if (userFrame.active) {
        userFrame.active = false;
        userFrame.sx = mouseX - userFrame.x;
        userFrame.sy = mouseY - userFrame.y;
    }
}

function step2DrawLoop() {
    if (!step2LoopEnabled) return;

    requestAnimationFrame(step2DrawLoop);

    cx.fillStyle = '#f9fafb';
    cx.fillRect(0, 0, canvasW, canvasH);

    cx.drawImage(userPhoto, userPhotoRect.x, userPhotoRect.y,
userPhotoRect.width, userPhotoRect.height);

    cx.strokeStyle = frameColor;
    cx.lineWidth = 2;
    cx.strokeRect(userFrame.x, userFrame.y, userFrame.width,
userFrame.height);

    userHandles.forEach((handle) => {
        cx.fillStyle = handleColor;
        cx.strokeStyle = handle.active ? handleBorderActive :
handleBorder;

        cx.lineWidth = 2;
        cx.fillRect(handle.x, handle.y, handleW, handleH);
        cx.strokeRect(handle.x, handle.y, handleW, handleH);
    });
}

function step3DrawLoop() {
    if (!step3LoopEnabled) return;

    requestAnimationFrame(step3DrawLoop);
    cx2.fillStyle = '#f9fafb';
    cx2.fillRect(0, 0, canvasW, canvasH);
    cx2.drawImage(userPhoto, userPhotoRect.x, userPhotoRect.y,
userPhotoRect.width, userPhotoRect.height);

    if (activeGlasses) {
        const glassesY = userFrame.y + (userFrame.height / 2) -
(glassesHeight / 2);
        cx2.drawImage(activeGlasses, userFrame.x, glassesY,
userFrame.width, glassesHeight);
    }
}

function initListeners() {
    step1CancelButton.addEventListener('click', (event) => {
        URL.revokeObjectURL(step1Preview.src);
        userPhoto.src = '';
        userPhotoRect = {};
        step1Preview.classList.add('hidden');
        step1Preview.src = '';
        step1FileInput.value = null;
    });
}

```

```

        step1ContinueButton.disabled = true;
        step1Dropzone.classList.remove('invisible');
    });

    step1ContinueButton.addEventListener('click', (event) => {
        resetUserFrame();
        resetUserHandles();
        step1Wrapper.classList.add('hidden');
        step2Wrapper.classList.remove('hidden');
        step2LoopEnabled = true;
        step2DrawLoop();
    });

    step1FileInput.addEventListener('change', (event) => {
        if (!step1FileInput.files[0]) return;
        userPhoto.onload = () => {
            userPhotoRect = fitRectInScreen(canvasW, canvasH,
userPhoto.naturalWidth, userPhoto.naturalHeight);

            step1ContinueButton.disabled = false;
            step1Dropzone.classList.add('invisible');
            step1Preview.classList.remove('hidden');
            step1Preview.src = userPhoto.src;
        };
        userPhoto.src
URL.createObjectURL(step1FileInput.files[0]);
    });
    step2CancelButton.addEventListener('click', (event) => {
        resetUserFrame();
        resetUserHandles();
        step2LoopEnabled = false;
        step2Wrapper.classList.add('hidden');
        step1Wrapper.classList.remove('hidden');
    });
    step2ContinueButton.addEventListener('click', (event) =>
{
        step2LoopEnabled = false;
        step2Wrapper.classList.add('hidden');
        step3Wrapper.classList.remove('hidden');
        setTimeout(() => {
            step3Frames.querySelector('img').click()
        }, 25);
        step3LoopEnabled = true;
        step3DrawLoop();
    });
    step2Canvas.addEventListener('mousedown', (event) =>
handleMouseDown(event, step2Canvas));
    step2Canvas.addEventListener('touchstart', (event) =>
handleMouseDown(event, step2Canvas));
    step2Canvas.addEventListener('mousemove', (event) =>
handleMouseMove(event, step2Canvas));
    step2Canvas.addEventListener('touchmove', (event) =>
handleMouseMove(event, step2Canvas));
    step2Canvas.addEventListener('mouseup', (event) =>
handleMouseUp(event, step2Canvas));
    step2Canvas.addEventListener('touchend', (event) =>
handleMouseUp(event, step2Canvas));
    step2Canvas.addEventListener('mouseleave', (event) =>
handleMouseUp(event, step2Canvas));

```



```

        step2Canvas.addEventListener('touchcancel', (event) =>
handleMouseUp(event, step2Canvas));
        step3CancelButton.addEventListener('click', (event) => {
            step3LoopEnabled = false;
            step3Wrapper.classList.add('hidden');
            step2Wrapper.classList.remove('hidden');
            step2LoopEnabled = true;
            step2DrawLoop();
        });
        step3ContinueButton.addEventListener('click', (event) => {
            const dataUrl = step3Canvas.toDataURL('image/png');
            const anchor = document.createElement('a');
            anchor.href = dataUrl;
            anchor.download = new Date().toString().replace(/[/:
] (\ (. * ? \) ) / g , ' _ ' ) + '.png';
            anchor.click();
        });
        step3Frames.addEventListener('click', (event) => {
            if (event.target.tagName !== 'IMG') return;

            setActiveGlasses(event.target);
        });
        step3Canvas.addEventListener('mousedown', (event) =>
handleMouseDown(event, step3Canvas));
        step3Canvas.addEventListener('touchstart', (event) =>
handleMouseDown(event, step3Canvas));
        step3Canvas.addEventListener('mousemove', (event) =>
handleMouseMove(event, step3Canvas));
        step3Canvas.addEventListener('touchmove', (event) =>
handleMouseMove(event, step3Canvas));
        step3Canvas.addEventListener('mouseup', (event) =>
handleMouseUp(event, step3Canvas));
        step3Canvas.addEventListener('touchend', (event) =>
handleMouseUp(event, step3Canvas));
        step3Canvas.addEventListener('mouseleave', (event) =>
handleMouseUp(event, step3Canvas));
        step3Canvas.addEventListener('touchcancel', (event) =>
handleMouseUp(event, step3Canvas));
    }
    const step1Wrapper = document.querySelector(`.js-step-1`);
    const step2Wrapper = document.querySelector(`.js-step-2`);
    const step3Wrapper = document.querySelector(`.js-step-3`);
    const step1CancelButton =
document.querySelector(`.js-step-1-btn-cancel`);
    const step1ContinueButton = document.querySelector(`.js-step-1-
btn-continue`);
    const step1Dropzone = document.querySelector(`.js-step-1-
dropzone`);
    const step1FileInput = document.querySelector(`.js-step-1-file-
input`);
    const step1Preview = document.querySelector(`.js-step-1-
preview`);
    const step2CancelButton = document.querySelector(`.js-step-2-btn-
cancel`);
    const step2ContinueButton = document.querySelector(`.js-step-2-
btn-continue`);
    const step2Canvas = document.querySelector(`.js-step-2-canvas`);
    const cx = step2Canvas.getContext('2d');
    const step3CancelButton = document.querySelector(`.js-step-3-btn-
cancel`);

```

```
const step3ContinueButton = document.querySelector(`.js-step-3-  
btn-continue`);  
const step3Canvas = document.querySelector(`.js-step-3-canvas`);  
const step3Frames = document.querySelector(`.js-step-3-frames`);  
const cx2 = step3Canvas.getContext('2d');  
  
initListeners();  
})();  
</script>  
</main>  
</body>  
</html>
```