

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Інститут інформатики та радіoeлектроніки,
Факультет комп'ютерних наук і технологій
(повне найменування інституту, назва факультету)

Кафедра програмних засобів
(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)

бакалавр

(ступінь вищої освіти)

на тему ПРОГРАМНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ
ДЛЯ БАРБЕРШОПУ
SOFTWARE IMPLEMENTATION OF A BARBERSHOP
INFORMATION SYSTEM

Виконав: студент(ка) 4 курсу, групи КНТ-228

Спеціальності 122 Комп'ютерні науки

(код і найменування спеціальності)

Освітня програма (спеціалізація)

Комп'ютерні науки

Харитонов М.О.

(прізвище та ініціали)

Керівник Коцур М.І.

(прізвище та ініціали)

Рецензент Гофман Є.О.

(прізвище та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»
(повне найменування закладу вищої освіти)

Інститут, факультет ІРЕ, ФКНТ

Кафедра програмних засобів

Ступінь вищої освіти бакалавр

Спеціальність 122 Комп'ютерні науки

(код і найменування)

Освітня програма (спеціалізація) Комп'ютерні науки

(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ, д.т.н, проф.

С.О. Субботін

“ ” 2022 року

З А В Д А Н Н Я

НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

Харитоновна Максима Олександровича

(прізвище, ім'я, по батькові)

1. Тема проєкту (роботи) Програмна реалізація інформаційної системи для барбершопу. Software Implementation of a Barbershop Information System

керівник проєкту (роботи) Коцур Михайло Ігорович, к.т.н., доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від “27” квітня 2022 року № 102

2. Строк подання студентом проєкту (роботи) червня 2022 року

3. Вихідні дані до проєкту (роботи) рекомендована література, технічне завдання

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз предметної області. 2. Розробка архітектури програми. 3. Основні рішення щодо реалізації компонентів системи. 4. Експлуатація, тестування та експериментальне дослідження програми.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) Слайди презентації

6. Консультанти розділів проєкту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-4 Основна частина	Коцур М.І., доцент		
Нормоконтроль	Липовець М.В., асистент		

7. Дата видачі завдання “04” квітня 2022 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Постановка завдання роботи.	1 тиждень	Завдання, ТЗ
2	Аналіз предметної області.	1 тиждень	Розділ 1
3	Вибір мови програмування та інших технологій розробки.	2 тиждень	Розділ 2
4	Розробка архітектури програми.	2 тиждень	Розділ 3
5	Розробка програми.	3-4 тижні	Розділи 3, 4
6	Тестування та експериментальне дослідження програмного забезпечення.	5 тиждень	Розділ 4
7	Оформлення пояснювальної записки та документів до неї.	6 тиждень	Додатки
8	Нормоконтроль та рецензування.	7 тиждень	
9	Захист роботи.	8 тиждень	

Студент(ка)

(підпис)

Харитонов М.О.
(прізвище та ініціали)

Керівник проєкту (роботи)

(підпис)

Коцур М.І.
(прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи бакалавра: 95 с., 47 рис., 3 табл., 4 дод., 19 джерел.

ІНФОРМАЦІЙНА СИСТЕМА, САЛОННИЙ БІЗНЕС, КОРПОРАТИВНИЙ САЙТ, ПРОГРАМНА РЕАЛІЗАЦІЯ, РОЗРОБКА, ВЕБЗАСТОСУНОК.

Об'єкт дослідження – процес програмної реалізації інформаційної системи для салонного бізнесу.

Предмет дослідження – методи програмної реалізації інформаційної системи для салонного бізнесу.

Метою роботи є програмна реалізація інформаційної системи для салонного бізнесу на прикладі кроссплатформного вебзастосунку для барбешопу.

Матеріали, методи та технічні засоби: об'єктно-орієнтоване програмування, мова програмування PHP, система контролю версій Joomla, система керування базою даних MySQL.

Результати. Розроблено програмне забезпечення у вигляді інформаційної системи для салонного бізнесу на прикладі кроссплатформного вебзастосунку для барбешопу

Практична цінність роботи полягає у розробці кроссплатформного вебзастосунку для барбешопу, як інформаційної системи для салонного бізнесу.

Висновки. Розроблено інформаційну систему для барбершопу у вигляді кроссплатформного вебзастосунку.

Галузь використання – чоловіча аудиторія користувачів мережі Інтернет та професійна спільнота майстрів-перукарів.

ABSTRACT

Explanatory note to the diploma qualifying work of the bachelor: 95 pages, 47 figures, 3 tables, 4 appendixes, 19 sources.

INFORMATION SYSTEM, SALON BUSINESS, CORPORATE WEBSITE, SOFTWARE IMPLEMENTATION, DEVELOPMENT, WEB APPLICATION.

The object of research is the process of software implementation of an information system for the salon business.

The subject of the research is methods of software implementation of an information system for salon business.

The aim of this work is to programmatically implement an information system for salon business using the example of a cross-platform web application for barbershop

Materials, methods, and technical tools: Object-Oriented Programming, PHP programming language, Joomla version control system, MySQL database management system.

Results. Developed software in the form of an information system for salon business on the example of a cross-platform web application for barbershop

The practical value of the work lies in the development of a cross-platform web application for barbershop, as an information system for the salon business.

Conclusions. An information system for Barbershop has been developed in the form of a cross-platform web application.

Field of usage: the male audience of internet users and the professional community of Master hairdressers.

ЗМІСТ

	С.
Перелік скорочень та умовних познач.....	7
Вступ.....	8
1 Аналіз предметної області.....	9
1.1 Використання інформаційних систем для салонного бізнесу	9
1.2 Необхідність розробки окремої інформаційної системи.....	10
1.3 Огляд існуючих рішень для соціальної взаємодії всередині академічної групи.....	13
1.4 Постановка завдання	18
1.5 Висновки за розділом 1	21
2 Розробка архітектури програми	22
2.1 Вибір інструментарію розробки.....	22
2.1 Структурна схема розробки.....	30
2.2 Висновки за розділом 2	31
3 Основні рішення щодо реалізації компонентів системи	32
3.1 Проєктування моделей системи.....	32
3.2 Висновки за розділом 3	42
4 Експлуатація, тестування та експериментальне дослідження програми.....	43
4.1 Опис застосунку.....	43
4.2 Умови виконання програми.....	43
4.3 Інструкція по експлуатації програми	44
4.4 Методика та результати тестування	49
Висновки	57
Перелік джерел посилання	58
Додаток А Технічне завдання	61
Додаток Б Опис програми	67
Додаток В Текст програми	71
Додаток Д Слайди презентації.....	88

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

БД – база даних;

ОС – операційна система;

ПЗ – програмний засіб;

ПК – персональний комп'ютер;

CMS – системи управління контентом;

PHP – Hypertext Preprocessor;

MVC – model-view-control;

SEO – пошукова оптимізація;

UML – Unified Modeling Language.

ВСТУП

Барбершоп – заклад, спрямований на косметичне обслуговування клієнтів, при чому в більшості випадків саме чоловіків. Даний ринок не перенасичений, тому з кожним днем зростає число нових салонів. Для підвищення якості обслуговування клієнтів і автоматизації процесів і завдань салони вдаються до використання інформаційних технологій.

Використання інформаційних технологій у сфері послуг забезпечить функціонування сучасних салонів на вищому рівні внаслідок того, що буде знижено навантаження на адміністратора салону. Потенційний клієнт зможе не стояти в чергах, зайти на сайт, здійснити пошук потрібної послуги, отримати про неї велику інформацію і записатися на послугу.

Метою даної дипломної кваліфікаційної роботи бакалавра є розробка інформаційної системи для барбершопу у вигляді вебзастосунку.

Доцільність реалізації дипломної кваліфікаційної роботи бакалавра обумовлена наступним фактором. Як правило, в салонах будь-який працівник перевантажений у зв'язку з великою кількістю відвідувачів в черзі і немає можливості проконсультувати кожного швидко і на достатньому високому рівні. У зв'язку з цим необхідним є використання в салоні автоматизованої системи, яка надає необхідну інформацію, володіє можливістю запису на будь-яку цікаву послугу. Салонний бізнес – це досить новий вид діяльності, який в даний час активно розвивається.

Таким чином, можна зробити висновок, що тема дипломної кваліфікаційної роботи бакалавра є актуальною.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

В першому розділі буде проведено аналіз предметної області – інформаційна система для салонного бізнесу. Буде досліджено загалом поняття інформаційної системи для корпоративного використання в середньому бізнесі та особливості таких систем. Також особливу увагу буде приділено дослідженню існуючих аналогів – виявлення їх основних переваг та недоліків.

За результатами роботи буде сформовано список вимог до функціоналу та інші неспецифічні вимоги (вимоги до безпеки, продуктивності, експлуатації, тощо). В подальшому ці напрацювання буде використано для складання технічного завдання на розробку.

1.1 Використання інформаційних систем для салонного бізнесу

Сфера послуг – велике і малодосліджене поле прикладання суспільної праці. Тривалий час сфера послуг взагалі не викликала особливого інтересу. І лише в останні десятиліття, вона стала притягувати увагу багатьох економістів [1]-[3].

З переходом України до ринкової економіки, сфера послуг займає домінуюче становище. На даному етапі розвитку українського бізнесу, особливе становище займають перукарні, салони краси та барбершопи [1]-[3].

Попит на послуги даних підприємств носить сезонний характер, потреби в красі, неповторному іміджі не мають великого спаду в будь-який час року, що дозволяє успішно розвиватися даній галузі [1]-[3].

Малим підприємствам важко конкурувати з лідерами даної галузі – великими мережами салонів краси. Для того, щоб змагатися і підтримувати прийнятний рівень прибутковості, будь-якому малому підприємству потрібно мати ефективні стратегію і тактику розвитку [1]-[3].

По-перше, необхідна якісна реклама, завдяки якій збільшується число потенційних клієнтів [1]-[3].

По-друге, важливими факторами є хороше локаційне розташування салону, чуйний, комунікабельний і висококваліфікований персонал, а також атмосфера всередині салону [1]-[3].

По-третє, потрібне хороше технічне обладнання [1]-[3].

Крім того, необхідні оптимізація і автоматизація основних бізнес-процесів з метою підвищення сервісу та оперативності рівня обслуговування і поліпшення якості продукції [1]-[3].

Процес автоматизації салонного бізнесу полягає в розробці та впровадженні програмного забезпечення (ПЗ) для автоматизованого обліку клієнтів з різним типом послуг [1]-[3]. Система повинна дозволяти централізовано контролювати і управляти всією інформацією, процесами, пов'язаними з прийняттям і виконанням замовлення. Керівнику система повинна дозволяти своєчасно отримувати достовірну інформацію, на основі аналізу якої буде надалі будуватися правильна економічна політика [1]-[3].

1.2 Необхідність розробки окремої інформаційної системи

Проте лише 51 відсоток малих підприємств мають вебсайти, незважаючи на те, що 97 відсотків споживачів шукають товари та послуги в Інтернеті, згідно з даними, опублікованими раніше Асоціацією SCORE [4]-[7].

Розрив між невеликими компаніями та зростаючою кількістю споживачів, які перевіряють звичайні підприємства на основі їх присутності в Інтернеті, може утримати підприємців від мільйонів клієнтів – і доларів [4]-[7]. Віце-президент SCORE з маркетингу Бріджит Вестон Поллак зазначив, що якщо компанія чи малий бізнес не мають присутності в Інтернеті, вони втрачають величезний відсоток населення, яке може робити покупки в їхньому магазині [4]-[7].

Вебсайт не тільки допомагає малим підприємствам рекламувати та продавати свої товари та послуги, але й дозволяє їм виділятися на тлі конкурентів, особливо для клієнтів, які в значній мірі покладаються на Інтернет, щоб дізнатися все про компанію [4]-[7]. Це робить вебсайт одним з найважливіших бізнес-активів для обміну інформацією, зміцнення довіри та виділення на переповнених ринках [4]-[7].

Згідно з недавнім звітом Verisign, 97 відсотків малих підприємств США, які мають вебсайт, рекомендують іншим малим підприємствам також створити вебсайт.

Пошук клієнтів. Verisign виявила, що 88 відсотків американських малих підприємств, які мають веб-сайт, погоджуються, що це полегшує клієнтам пошук свого бізнесу [4]-[7]. Сьогодні клієнти постійно шукають інформацію на комп'ютерах, планшетах і смартфонах. Вебсайт дозволяє їм знайти вашу компанію і дізнатися про ваш бізнес і про те, що він пропонує, в будь-який час і в будь-якому місці [4]-[7]. У сьогоднішньому конкурентному середовищі онлайн-пошуку контент відіграє ключову роль. Сьогодні Google оновила свої алгоритми, щоб краще обслуговувати пошукові запити користувачів, включаючи корисні сайти, які надають відвідувачам корисний контент і хороший користувацький досвід [4]-[7]. Це означає, що якщо хтось шукає продукт або послугу, пропоновані вашим бізнесом, ваш веб-сайт повинен відображатися в результатах пошуку, інакше ви пропустите потенційних потенційних клієнтів [4]-[7].

Накопичення довіри та авторитету. Власник може створити присутність бренду для свого бізнесу в соціальних мережах, але платформа обмежує вас з точки зору дизайну, процесу та технології. Веб-сайт може допомогти малому бізнесу підвищити гнучкість, контроль, брендинг і довіру [4]-[7]. Вісімдесят чотири відсотки споживачів у США вважають, що бізнес з веб-сайтом заслуговує більшої довіри, ніж той, у якого є тільки сторінка в соціальних мережах. Щоб бути надійним і залучати нових клієнтів, веб-сайт може мати велике значення [4]-[7].

Залучення великої кількості клієнтів. Постійний успіх у бізнесі часто залежить від здатності вашої компанії залучати нових клієнтів [4]-[7]. Охоплення тисяч людей з використанням лише традиційних маркетингових методів, таких як пряма поштова розсилка або друкована реклама, може бути дуже дорогим і ненадійним [4]-[7]. Вихід в Інтернет дозволяє вам охопити більше людей на місцевому рівні або за межами вашого регіону, при цьому сплачуючи менше за експозицію-і ви навіть можете виміряти конкретні результати. Насправді, 77 відсотків малих підприємств США заявили, що веб-сайт – відмінний спосіб знайти нових клієнтів [4]-[7]. І мова йде не тільки про залучення нових клієнтів; ви також можете використовувати веб-сайт для кращого обслуговування існуючих клієнтів [4]-[7]. Можливість оновлювати бізнес-інформацію 24/7 в Інтернеті покращує спілкування з клієнтами, одночасно звільняючи час для того, щоб зосередитися на інших бізнес-пріоритетах [4]-[7]. А за допомогою фірмового електронного листа ви можете почати маркетинг по електронній пошті, повертаючи трафік на свої онлайн – і оффлайн-сайти і підвищуючи лояльність клієнтів до вашого бренду [4]-[7].

Розвиток власного бізнесу. Багато малих підприємств прагнуть розширитися на ринку і залучити нових клієнтів. Хоча спілкування віч-на-віч може бути важливим, а бізнес часто передається з вуст в уста, веб-сайт дає вам шанс роздати свою картку тисячам людей онлайн [4]-[7]. За допомогою магазину електронної комерції ви можете продавати товари онлайн і відправляти товар покупцям практично в будь-яку точку світу. Власник може виявити, що клієнти в сусідньому місті або штаті зацікавлені в ваших послугах, що призведе до розширення зони обслуговування і, можливо, навіть розширення регіональних офісів [4]-[7]. Насправді, 81 відсоток жителів США малі підприємства кажуть, що веб-сайт допоміг розвинути свій бізнес. За допомогою веб-сайту ви можете охопити більшу кількість потенційних клієнтів і можливостей [4]-[7].

Отримання конкурентної переваги. Нещодавно опублікований звіт Verisign показує, що 93 відсотки споживачів у США використовують Інтернет для проведення досліджень перед здійсненням покупки [4]-[7]. Споживачі хочуть бути поінформованими про свої рішення про покупку-що вони купують і у кого вони купують [4]-[7]. Без вебсайту ви ризикуєте втратити клієнтів для компаній, які вже користуються перевагами онлайн-ринку. Якщо ви перебуваєте в прямій конкуренції з іншою компанією, яка має вебсайт, ваш конкурент має явну перевагу - особливо якщо вони ефективно рекламують свій вебсайт. Вебсайт може допомогти вирівняти ігрове поле [4]-[7]. Вісімдесят чотири відсотки американських малих підприємств, опитаних Verisign, заявили, що їхній веб-сайт важливий або критичний для їхнього бізнесу [4]-[7]. Більше половини малих підприємств у США не мають веб-сайту, тому зараз немає кращого часу, ніж створити свій бізнессайт і випередити конкурентів [4]-[7].

1.3 Огляд існуючих рішень для соціальної взаємодії всередині академічної групи

На сьогоднішній день варто зазначити, що таке явище, як барбершоп – не звичайний салон краси, а тематично та клієнто-орієнтовний активно входить на ринок сфери послуг. Тому еталонних рішень не існує. Проте в мережі вже достатньо сайтів та рішень на які можна орієнтуватися для подальшої розробки власного рішення з низкою переваг, що якісно піднімуть рівень розробки.

1.3.1 Проєкт «Sborodoy»

Sborodoy це перший і єдиний український каталог барбершопів. Розробники запевняють, що створили сайт, щоб полегшити життя чоловікам. На сайті клієнти можуть легко вибрати для себе чоловічу перукарню у власному місті [8].

На сайті публікується інформація про барбершопи України. У нас ви знайдете докладний опис кожної з чоловічих перукарень, місце розташування, фотографії та відгуки про роботу барберів. Якщо клієнт в пошуку чоловічої перукарні – каталог допоможе вибрати барбершоп. Також на сайті клієнт знайде цікаві статті на тему носіння борід і вусів, поради по догляду за волоссям, поради професійних барберів, інтерв'ю з власниками борід [8].

Найближчим часом розробники планують додати на сайт можливість онлайн запису, так само вони працюють над унікальним і правдивим рейтингом чоловічих перукарень України. Розробники постійно додають нові барбершопи в каталог. Вже зараз каталог пропонує клієнтам вибрати ідеальну чоловічу перукарню з представлених барбершопів Києва, Одеси, Харкова, Дніпра, Львова, Миколаєва, Чернігова, Житомира, Полтави, Кривого Рогу та Херсона [8].

Приклад головної сторінки каталогу для міста Запоріжжя наведено на рисунку 1.1.

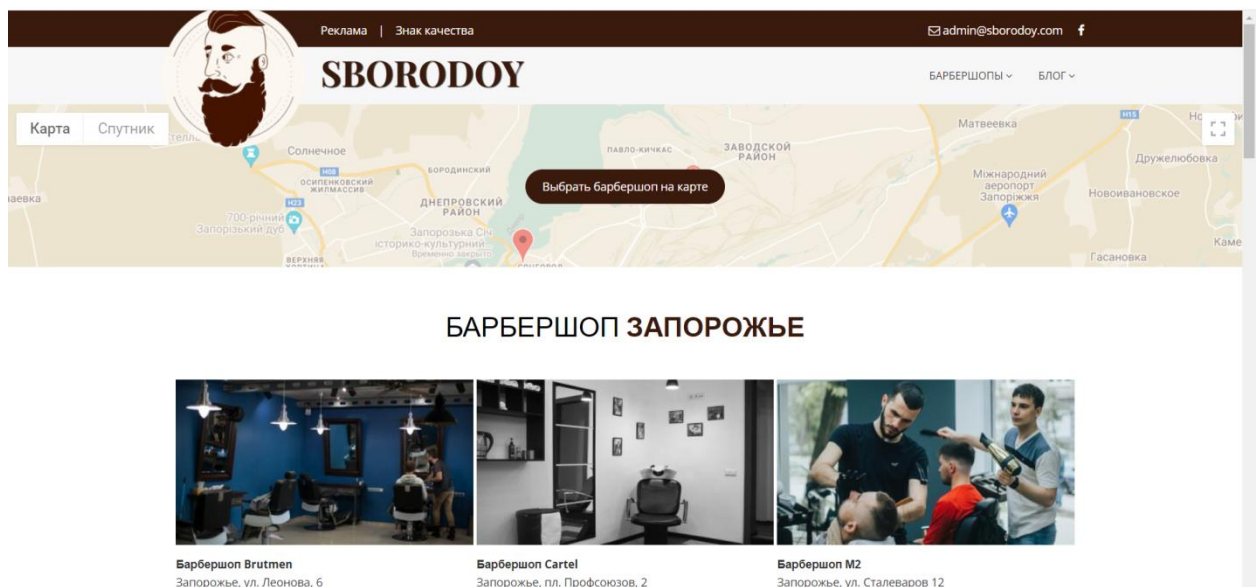


Рисунок 1.1 – Приклад головної сторінки каталогу для міста Запоріжжя [8]

1.3.2 Проєкт «Frisor Barbershop»

Перукарня Frisor поєднує в собі старомодні традиції з технологіями, випурне освітлення, наче в кіно для якісної геометрії стрижок і класичні технології з принципами архітектури та скульптури чоловічої стрижки, і тільки так майстри барбершопу допоможуть клієнтам розробити та підкреслити індивідуальний образ [9]. Власники запевняють, що їх чоловіча перукарня – перша, яка наважилася зламати всі стереотипи. Майстри салонів представляються певним поєднанням перукарів, технологів і стилістів, які прагнуть до досконалості [9]. Мережа Frisor проводить щотижневі зустрічі та семінари з метою поділитися секретами і хитрощами правильного домашнього гоління небезпечною бритвою і ритуалами королівського гоління [9]. Приклад головної сторінки мережі барбершопів наведено на рисунку 1.2.



Рисунок 1.2 – Приклад головної сторінки мережі барбершопів [9]

1.3.3 Проєкт «Firm»

Фірма Firm була заснована в 2012 році, коли низка майстрів об'єдналася навколо ідеї створити місце, якого раніше не існувало, місце, здатне поліпшити життя людей і міста [10].

Філософія фірми – строгість, лаконічність і стриманість [10].

Майстри салону поважають один одного, вони не сліднують чужим прикладам. Робітники знають, чого хочуть від життя, і заперечують умовності. На них не впливає громадська думка, і вони виступають за справедливість, тому що самі в неї віряють [10].

Салон Firm виступає за зміцнення справжніх чоловічих цінностей [10].

Приклад головної сторінки салону Firm наведено на рисунку 1.3.

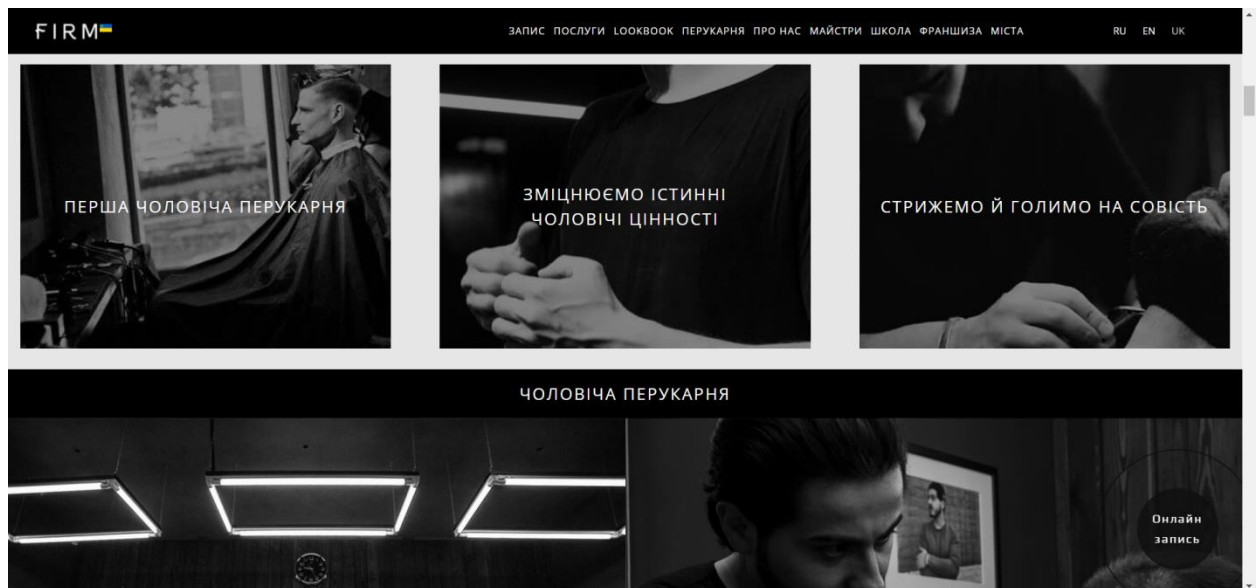


Рисунок 1.3 – Приклад головної сторінки салону Firm [10]

1.3.4 Порівняння існуючих аналогів

Проведемо порівняння за певними критеріями, хід якого представимо у вигляді таблиці (табл. 1.1). Серед критеріїв виділимо наступні:

- можливість онлайн запису до майстра – чи надає система можливість онлайн-запису до обраного майстра на бажаний час;
- перегляд інформації про майстрів – чи можливо попередньо переглянути інформацію про кожного майстра окремо;
- можливість залишити відгук або ознайомитися із відгуками – можливість залишити фото або текстовий відгук про роботу майстра та салону в цілому;
- перегляд каталогу засобів для догляду – можливість після отримання послуг також придбати спеціалізовані косметичні засоби.

Таблиця 1.1 – Порівняння існуючих аналогів

Критерій порівняння	Sborodoy	Frison Barbershop	Firm	Інформаційний вебресурс
Можливість онлайн запису до майстра	Не доступно	Можливо	Не доступно	Можливо
Перегляд інформації про майстрів	Відсутній	Відсутній	Відсутній	Можливо
Можливість залишити відгук або ознайомитися із відгуками	Можливо	Відсутня	Відсутня	Можливо
Перегляд каталогу засобів для догляду	Відсутній	Відсутній	Відсутній	Можливо

З результатів порівняння можна зробити висновок, що загалом більшість існуючих рішень достатньо обмежені в своєму функціоналі. Так, наприклад, Sborodoy – не надає більшість інтерактивних функцій для користувача, адже є загальним агрегатором салонів по Україні. Frison Barbershop в порівнянні виглядає найбільш закінченим проєктом на перший погляд, проте при тестуванні функцій більшість з них не працюють. Firm –

більш розрахований на інтерактивні потреби клієнта, що можна пояснити локальною орієнтованістю проєкту, проте не надає можливості для зворотного зв'язку із клієнтами.

Саме тому актуальною є задача розробки інформаційної системи для барбершопу у вигляді вебзастосунку із інтерактивними функціями для клієнтів.

1.4 Постановка завдання

Проаналізувавши усі найбільш поширені і можливі варіанти реалізації проєкту було прийнято рішення про організацію незалежного сайту на окремому хостингу з метою тонкого налаштування його функціоналу для інтерактивного зв'язку із клієнтами.

1.4.1 Межі системи, що розробляється

Система буде надавати послуги з перегляду інформації про послуги салону (вартість, майстри, що виконують дану послугу), крім того в інформації буде присутній опис її [11]-[12].

Також система буде надавати можливість реєструватись, авторизуватись на сайті. Дані про це будуть заноситись до бази даних (БД).

Система буде мати також інформацію про салон, а саме – місце розташування, часи роботи салону та певних майстрів, телефон. Записуватись на прийом можуть лише авторизовані користувачі [11]-[12].

У БД будуть зберігатися розклади днів майстрів, для легкості перегляду свого розпорядку майстрами та для уникнення помилок при запису на прийом до одного й того ж майстра декількох осіб на один і той же час.

У разі необхідності адміністратор може редагувати дані БД.

1.4.2 Функціональність системи

Функціональні вимоги до системи, що розробляється [11]-[12].

Робота з сайтом:

- а) реєстрація нових користувачів за допомогою форми (логін, пароль);
- б) авторизація на сайті;
- в) вилучення визначеного користувача з бази даних системи;
- г) перегляд контактної інформації салону;
- д) перегляд додаткової інформації про послуги;
- е) перегляд вартості послуг;
- ж) робота з інформацією обмеженого доступу:
 - 1) додавання та видалення коментарів до фотографій;
 - 2) додавання оцінок до фотографій;
 - 3) додавання відгуків та видалення про майстра;
 - 4) перегляд особистої інформації про майстра (телефон, часи роботи (розклад), список записаних на прийом)
 - 5) запис на прийом;
- з) редагування сайту адміністратором:
 - 1) реєстрація майстрів;
 - 2) видалення майстрів;
 - 3) редагування інформації про майстра;
 - 4) редагування інформації про салон (часи роботи, контактну інформацію, вартість послуг);
 - 5) додавання та видалення робіт майстрів (фото).

1.4.3 Вимоги до інтерфейсу

Інтерфейс системи повинен бути ергономічним, «легким» та зрозумілим для користувача. Усі основні елементи системи слід зробити інтуїтивно зрозумілими для користувачів та максимально спростити доступ до них з різних частин системи [11]-[12].

Інтерфейс блоку додавання матеріалів та коментарів повинен мати підказки, які допоможуть навіть невідготуваним користувачам здійснити свій намір [11]-[12].

1.4.4 Вимоги до продуктивності

Для великих інтернет – ресурсів важливо подбати про продуктивність системи, вони повинні витримувати великі навантаження. Редагувати, додавати або видаляти основну інформацію може лише адміністратор [11]-[12].

Інтернет – ресурс повинен відкриватись в усіх браузерях, чітко відтворювати необхідну інформацію і зображення. Програмний продукт переноситься на будь – якому комп'ютері, планшеті чи ноутбукі [11]-[12].

1.4.5 Вимоги до безпеки

Усі користувачі можуть переглядати інформацію про салон, майстрів та роботи майстрів [11]-[12]. Реєстрування відбувається за допомогою допоміжної форми з логіном, паролем та особистою інформацією про клієнта.

Авторизовані користувачі системи можуть переглядати більше інформації про салон та майстрів, залишати відгуки про майстрів, оцінювати та коментувати їх роботи, записуватись на прийом [11]-[12]. Користувачі, що є майстрами – можуть переглядати список записаних на прийом клієнтів. Тільки адміністратор може редагувати дані на сайті, крім того вносити до системи майстрів салону, редагувати їх особисту інформацію та видаляти їх.

1.4.6 Експлуатаційні вимоги

Висуваються такі вимоги:

- наявність браузера для перегляду сайту;
- наявність встановленого на клієнтській стороні Flashplayer;

- наявність вебсервера Denwer для тестування і налагодження.

1.4.7 Не функціональні вимоги до продукту

Мінімальні системні вимоги та програмне забезпечення , що необхідні для функціонування серверної частини мобільного застосунку:

- сервер із встановленою серверною операційною системою (ОС);
- розгорнутий вебсервер;
- інтерпретатор PHP5;
- розгорнутий MySQL-сервер.

Для експлуатації клієнтської частини необхідні такі програмно-технічні засоби:

- IBM-сумісний персональний комп'ютер (ПК);
- процесор із тактовою частотою від 1,5 GHz;
- обсяг оперативної пам'яті – від 3GB (для безперебійної роботи браузерів);
- підключення до мережі Internet;
- встановлений браузер останньої версії.

1.5 Висновки за розділом 1

В першому розділі було проведено аналіз предметної області – інформаційна система для салонного бізнесу. Особливу увагу було приділено дослідженню існуючих аналогів – виявлення їх основних переваг та недоліків.

За результатами роботи було сформовано список вимог до функціоналу та інші неспецифічні вимоги.

2 РОЗРОБКА АРХІТЕКТУРИ ПРОГРАМИ

Перший розділ було присвячено аналізу та дослідженню предметної області: інформаційні системи для салонного бізнесу. Особливу увагу було приділено дослідженню існуючих аналогів – виявлення їх основних переваг та недоліків. За результатами роботи було сформовано список вимог до функціоналу та інші неспецифічні вимоги.

Другий розділ буде присвячено висвітленню технічних питань розробки архітектури та реалізації бізнес логіки під час розробки. В розділі було проведено аналіз, вибір та його обґрунтування для технології застосунку, мови програмування та програмних надбудов. Визначення цих питань дозволить обґрунтувати наступні рішення з розробки застосунку.

2.1 Вибір інструментарію розробки

2.1.1 Вибір мови програмування

Як і у всіх мов програмування, у мови сценаріїв з Hypertext Preprocessor (PHP) є ряд плюсів і мінусів, які впливають на те, наскільки вона підходить для різних застосунків. Переваги PHP полягають у тому, що це безкоштовна мова програмування з відкритим кодом, відносно проста в освоєнні і гнучка. PHP також підтримує цілий ряд типів баз даних, що робить його хорошим вибором для додатків, яким необхідно взаємодіяти з базами даних [13]-[14]. Мінуси PHP включають в себе те, що він неефективний при створенні настільних додатків, відносно повільний в порівнянні з деякими іншими мовами і має погану підтримку для обробки помилок [13]-[14].

У PHP є багато плюсів і мінусів, як і у будь-якої мови програмування. Однією з головних переваг PHP є те, що він має відкритий вихідний код. Це означає, що він Безкоштовний у використанні і постійно вдосконалюється великою кількістю людей, а не однією компанією [13]-[14]. Розробники PHP створили великий онлайн-ресурс з усіма функціями мови, включаючи

приклади їх використання, що полегшує вивчення РНР, ніж вивчення альтернатив. РНР – настільки популярна мова програмування, що існує також ряд онлайн-груп підтримки, які допомагають прискорити процес навчання [13]-[14].

Ще однією перевагою РНР є те, що це гнучка мова програмування. Багато мов програмування в 2011 році розроблені як об'єктно-орієнтовані, що є загальноприйнятим правилом в програмуванні, яке допомагає більш ефективно організовувати дані [13]-[14]. Програмісти часто сперечаються, коли потрібно об'єктно-орієнтоване програмування. РНР підтримує об'єктно-орієнтоване програмування, але також може функціонувати і без нього [13]-[14].

РНР дуже ефективний при доступі до різних типів баз даних і взаємодії з ними. Це робить його ідеальним для вебскриптів, таких як системи управління контентом вебсайтів, яким необхідно зберігати дані в базі даних. Мова програмування підтримує кілька типів баз даних, що є одним з найбільш важливих плюсів і мінусів РНР [13]-[14].

Одним з головних недоліків РНР є те, що він, як правило, не підходить для створення настільних додатків [13]-[14]. Ось чому важливо зважити всі ЗА і проти РНР стосовно конкретного проекту, перш ніж приймати рішення про його використання. Крім того, в порівнянні з іншими мовами, додатки РНР, як правило, працюють повільніше. Чи є це проблемою для конкретного проекту, залежить від обчислювальної потужності, необхідної для його виконання [13]-[14].

Одним з найбільш часто обговорюваних плюсів і мінусів РНР є його обробка помилок [13]-[14]. Обробка помилок РНР традиційно вважається поганою в порівнянні з іншими мовами програмування, в основному через історію розробки РНР і його підтримки старіших функцій. Це означає, що може знадобитися більше часу, щоб з'ясувати, чому конкретний фрагмент коду працює не так, як очікувалося [13]-[14].

Проведемо порівняння мов програмування для реалізації вебзастосунків, результати порівняння представимо в табличній формі (табл. 2.1). У якості критеріїв будемо використовувати наступні:

- відкритість – загальна характеристика відкритості мови програмування, як проекту;
- кроссплатформність в рамках серверу – чи є мова кроссплатформною в рамках сервера на якому буде розгорнута відповідна частина застосунку;
- стабільність та надійність застосунків – загальна оцінка стабільності та надійності роботи застосунків на мові програмування;
- програмні надбудови для розробки – наявність спеціалізованих програмних надбудов (систем керування версіями (CMS) або фреймворків) для пришвидшення розробки;
- підтримка та масштабування застосунків – легкість та загальна можливість наступної підтримки застосунків.

Таблиця 2.1 – Порівняння мов програмування

Критерій	Python	JavaScript	PHP
Відкритість	++	++	++
Кроссплатформність в рамках серверу	+	++	++
Стабільність та надійність застосунків	+	±	++
Програмні надбудови для розробки	+	+	++
Підтримка та масштабування застосунків	+	±	++

З таблиці видно, що PHP є найоптимальнішим вибором для розробки, а наявна велика база фреймворків та CMS гарантує спрощення, пришвидшення та надійність розробки.

2.1.2 Система керування версіями

І ось тут-то і проявляється моя особиста перевага Joomla. WordPress-чудова платформа, і я ні в якому разі не проти неї. Я вже використовував його з багатьма клієнтами раніше, і він працює; просто WordPress, по суті, був розроблений для ведення блогів. Joomla, з іншого боку, була розроблена спеціально для веб-сайтів, а також для веб-сайтів, які повинні рости і розвиватися з плином часу. Все дуже просто [15]-[16].

Переваги Joomla.

Модулі. Joomla дозволяє легко позиціонувати модулі. Це значно спрощує адаптацію і розвиток веб-сайту і дозволяє веб-сайту рости, просто розміщуючи позиції модулів вище або нижче один одного або пліч-о-пліч, без необхідності повертатися і редагувати шаблон [15]-[16]. Можна налаштувати стиль для кожного модуля або просто налаштувати класи стилів, готові для застосування до кожного з них [15]-[16]. SEO. Joomla дозволяє створювати мета-описи, мета-ключові слова і заголовки сторінок для всіх ваших сторінок [15]-[16]. Він також має глобальні налаштування метаданих, якщо ви хочете призначити загальні метадані. З WordPress вам потрібен плагін; Yoast є популярним вибором [15]-[16]. Для створення простих перенаправлень зі старих URL-адрес на нові.

Для створення канонічних URL-адрес, якщо у вас є дублюючий контент на сторінках, і один повинен бути прочитаний пошуковими системами поверх іншого [15]-[16]. Повний контроль над заголовками сторінок, метаописами і ключовими словами за допомогою методу виділення, який дозволяє вам дізнатися, чи не занадто високо слово [15]-[16].

Контроль управління соціальними мережами. Це дозволяє вам керувати тим, що з'являється, коли Ви ділитесь URL-адресою в Інтернеті, починаючи з заголовка та опису до використовуваного зображення. (Знову ж таки, ви навіть можете встановити зображення для всіх видів використання сторінки для зручності) [15]-[16]. Можливість додавати необроблений код в

теги head і body на кожній сторінці; зручно для відстеження конкретних сторінок [15]-[16].

Стиснення та кешування. Знову ж таки, без необхідності в додаткових плагінах, ви можете контролювати завантаження вашого сайту і прискорити його, використовуючи вбудовану в Joomla опцію стиснення GZIP [15]-[16]. Ви також можете налаштувати різні параметри кешування, щоб керувати тим, як ваш веб-сайт зберігається в браузерх користувачів, щоб максимізувати швидкість завантаження. Для цього WordPress потрібен ще один пагін [15]-[16].

Статті. У той час як Wordpress використовує сторінки і повідомлення. Joomla просто використовує статті для обох [15]-[16]. Це дає вам перевагу при перетворенні записів в блозі на сторінки і навпаки. Joomla значно еволюціонував з перших днів свого існування, і створювати і класифікувати блоги ще ніколи не було так просто [15]-[16]. Однією з переваг статей на Joomla в порівнянні з постами на WordPress є той факт, що за допомогою WordPress ви можете вибрати вибране зображення для свого поста в блозі.

Шаблони або теми. Joomla дозволяє використовувати кілька шаблонів, тоді як WordPress дозволяє використовувати лише одну тему для кожного веб-сайту [15]-[16]. Це означає, що за допомогою Joomla ви можете мати сторінки на своєму веб-сайті з абсолютно іншим виглядом, якщо побажаєте, і додатковою перевагою є спільне використання позицій модулів в декількох шаблонах [15]-[16].

Безпека та плагіни. Оскільки WordPress є найпопулярнішою CMS, вона намалювала досить велику мету на своїй спині і привернула значну кількість спроб злому, ніж будь-яка інша платформа [15]-[16].

Те, що також не допомагає WordPress, полягає в тому, що для виконання основних вимог сьогоdnішнього веб-сайту потрібно багато різних плагінів [15]-[16]. Безліч різних плагінів надходить з безлічі різних джерел. Тільки для того, щоб привести веб-сайт у відповідний стан для того, що потрібно кінцевому користувачеві. Для кожного плагіна, який не

оновлюється, він стає вразливим для можливої спроби злому. Враховуючи, що WordPress є CMS для не-розробників, залишатися на висоті не просто [15]-[16].

Діаграма взаємодії між компонентами Joomla зображена на рис. 2.1.



Рисунок 2.1 – Діаграма взаємодії між компонентами Joomla

2.1.3 Розподіл та реалізація бізнес-логіки

Найбільш повно концепція бізнес-логіки відображена на діаграмі класів (рис. 2.2), завдяки розподілу функцій між типами користувачів [11]-[12].

До View (Представлення) – відноситься інтерфейс: сторінки з інформацією про салон, майстрів; посилання; кнопки; форми, що необхідні для реєстрації, запису на прийом, чи для видачі повідомлення про помилку; результати відображення запитів користувача.

Model зберігає дані класів «Запис», «Каталог послуг», «Послуга», «Каталог фотографій», «Фотографія».

Controller обробляє функції класів «Запис», «Каталог послуг», «Послуга», «Каталог фотографій», «Фотографія».

Функції класів розподілені між чотирма видами користувачів «Авторизований», «Майстер», «Адміністратор», «Не авторизований».

2.1.4 Обґрунтування вибору CMS для розробки вебресурсу

Порівняльна характеристика найпоширеніших CMS для розробки інтернет-магазинів наведена у таблиці 2.2.

JOOMLA MODEL VIEW CONTROLLER (MVC)

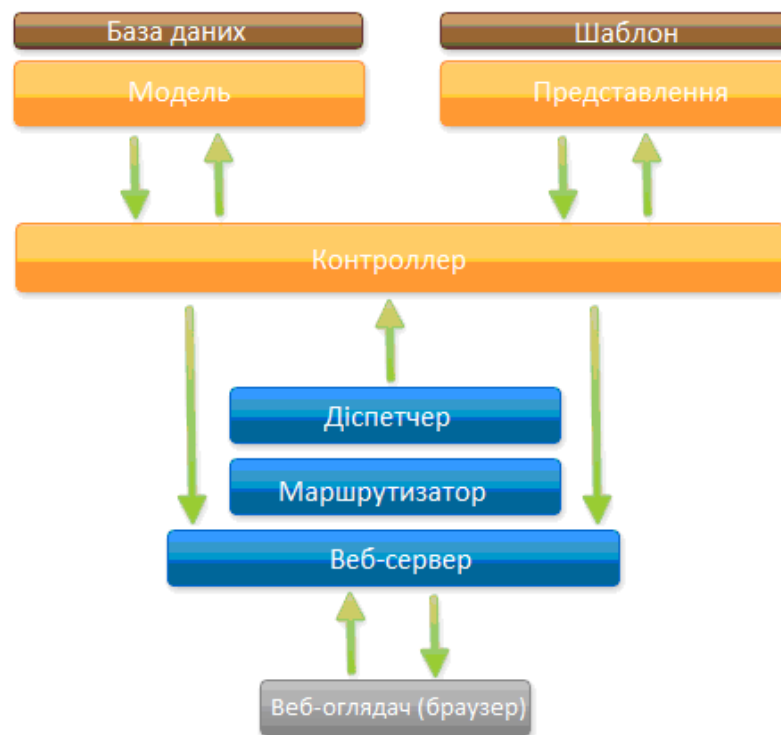


Рисунок 2.2 – Представлення патерну MVC в CMS Joomla

Серед критеріїв порівняння будемо розглядати наступні:

- початок роботи – простота розгортання та старту подальшої роботи із CMS;
- гнучкість налаштування функціоналу – можливість тонко налаштувати функціонал застосунку під особливі потреби та вимоги адміністратора;
- навантаження на хостинг – технічні вимоги CMS до хостингу під час розгортання застосунку;
- використання розширень – наявність та надійність розширень для спрощення розробки та захисту даних в застосунку.

Розподіл бізнес логіки застосунку в CMS Joomla наведено на рис. 2.3.

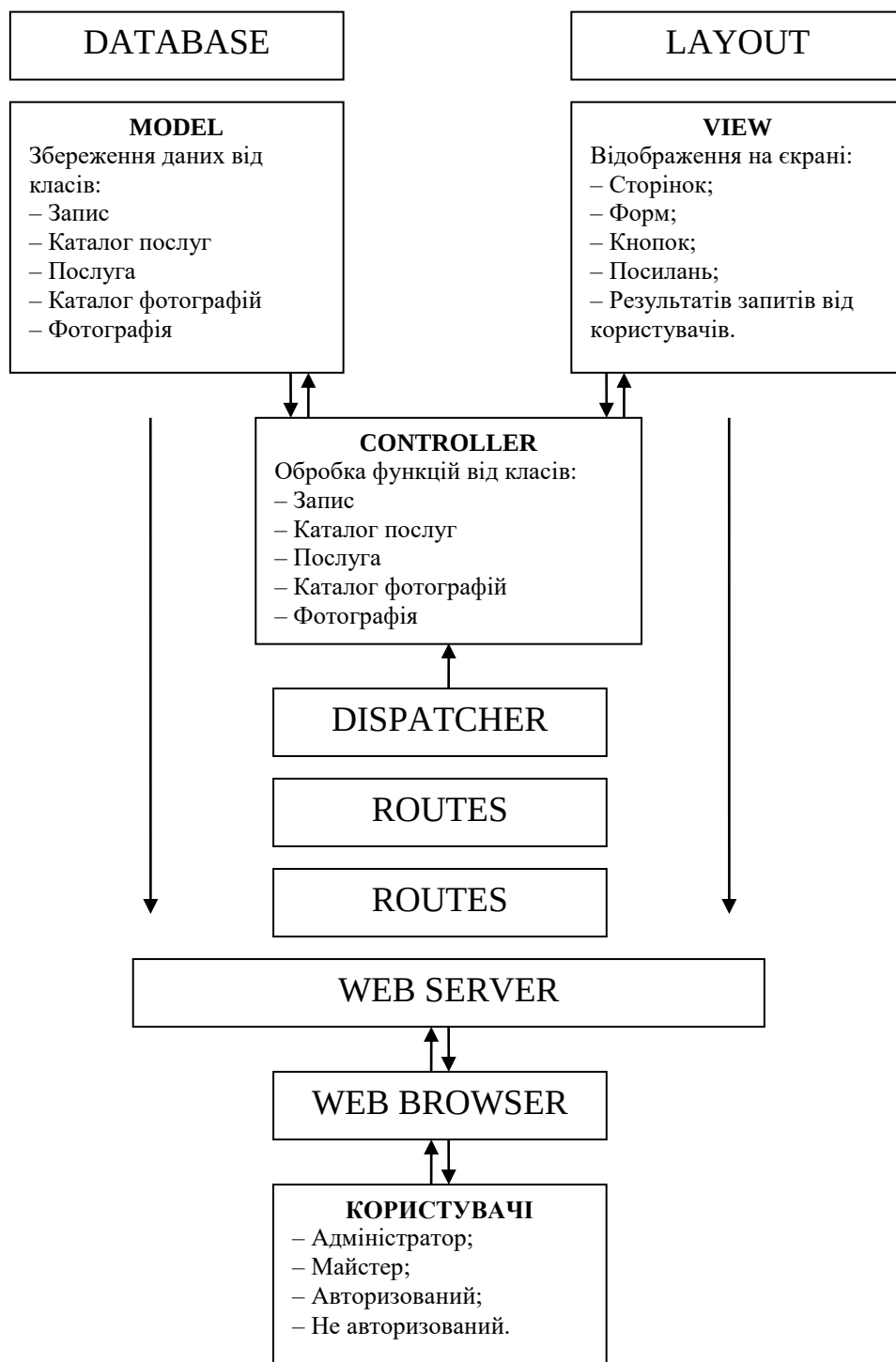


Рисунок 2.3 – Розподіл бізнес логіки застосунку в CMS Joomla

З порівняння наведеного в таблиці можна зробити висновок, що використання Joomla! є найбільш оптимальним варіантом. Це пов'язано із легкістю розгортання CMS та мінімальними вимогами до хостингу. Також

важливо відмітити низький рівень проблем із сумісністю плагінів на відміну від Wordpress, наприклад.

Таблиця 2.2 – Порівняння CMS

	WordPress	Drupal	Joomla!
Початок роботи	Простий та швидкий	Простий та швидкий	Простий та швидкий
Гнучкість налаштування функціоналу	Посередня	Посередня	Висока
Навантаження на хостинг	Високе	Низьке	Низьке
Використання розширень	Велика кількість, недостатня захищеність	Велика кількість, недостатня захищеність	Велика кількість, висока захищеність

2.1 Структурна схема розробки

Інтерфейс користувача містить тільки форми опитування даних, у вигляді полів, посилань, радіокнопок і т.д. безпосередньою їх обробкою займається ПЗ на стороні сервера [11]-[12].

Згідно спроектованої раніше концепції, показаної на діаграмі компонентів (рис. 2.4), попередня обробка даних користувацьких форм на стороні клієнта буде виконуватися модулями, які, в свою чергу, обробляються вебсервером як окремі інтерпретовані програми

Оброблені дані формують транзакції, що направляються в БД для збереження і формують відгук(у вигляді повідомлення або HTML-сторінки), який відправляється клієнту на відображення успішності\неуспішності скоєних дій, наприклад, входу на сайт або додавання коментаря [11]-[12].

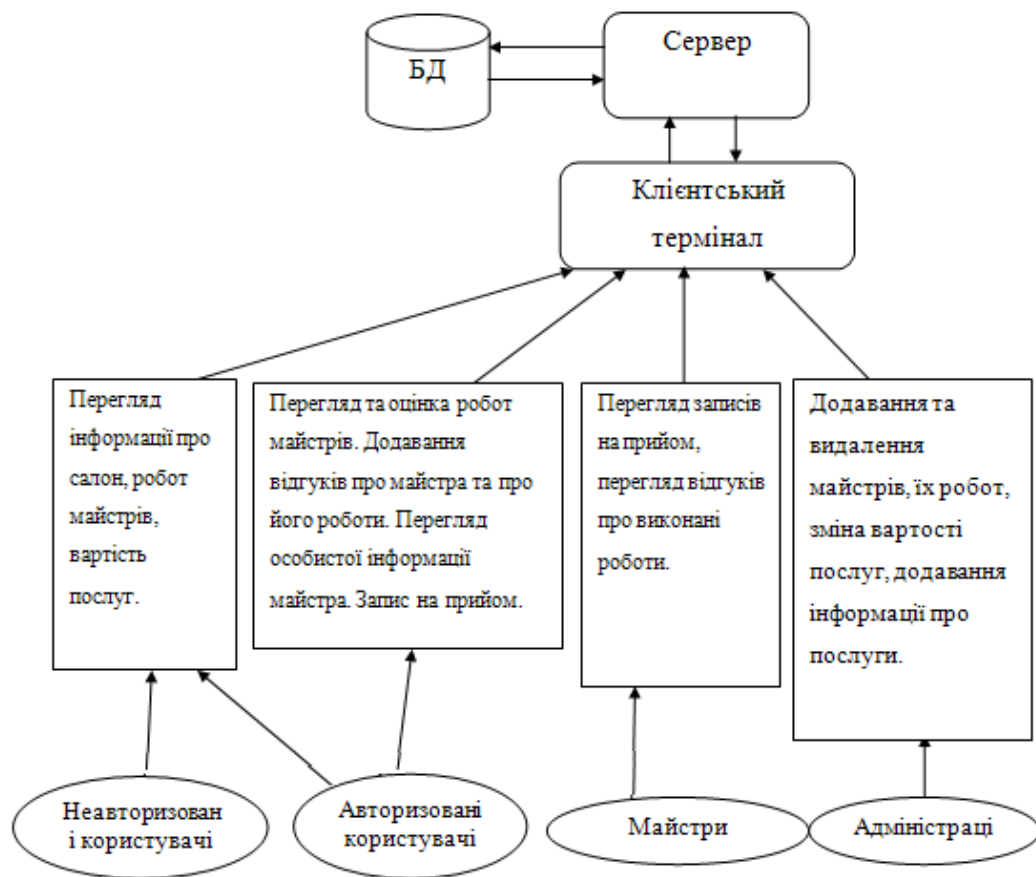


Рисунок 2.4 – Архітектура вебзастосунку

2.2 Висновки за розділом 2

В цьому розділі було досліджено питання вибору технології розробки застосунку:

- так у якості технології обрано – розробку вебзастосунку;
- у якості мови програмування – серверну мову PHP;
- у якості програмної надбудови – CMS Joomla.

Патерн проектування MVC в повній мірі дозволить організувати та розподілити бізнес логіку застосунку.

Визначення цих питань дозволяє обґрунтувати наступні рішення з розробки застосунку.

3 ОСНОВНІ РІШЕННЯ ЩОДО РЕАЛІЗАЦІЇ КОМПОНЕНТІВ СИСТЕМИ

3.1 Проєктування моделей системи

Unified Modeling Language (UML) — це мова моделювання загального призначення [17]-[18]. Основна мета UML - визначити стандартний спосіб візуалізації того, як була розроблена система. Він дуже схожий на креслення, які використовуються в інших галузях техніки [17]-[18].

UML – це не мова програмування, це скоріше візуальна мова. Ми використовуємо діаграми UML, щоб зобразити поведінку та структуру системи. UML допомагає інженерам програмного забезпечення, бізнесменам і системним архітекторам моделювати, проєктувати та аналізувати [17]-[18]. Група управління об'єктами прийняла уніфіковану мову моделювання як стандарт у 1997 році. Відтоді нею керує OMG. Міжнародна організація зі стандартизації опублікувала UML як затверджений стандарт у 2005 році. UML переглядався протягом багатьох років і періодично переглядається [17]-[18].

Діаграма класів UML – це графічне позначення, яке використовується для побудови та візуалізації об'єктно-орієнтованих систем [17]-[18]. Діаграма класів на UML – це тип діаграми статичної структури, яка описує структуру системи, показуючи:

- класи;
- їх атрибути;
- операції (або методи);
- і відносини між об'єктами [17]-[18].

Діаграма класів для застосунку зображена на рис. 3.1.

Діаграми пакетів – це структурні діаграми, які використовуються для показу організації та розташування різних елементів моделі у вигляді пакетів [17]-[18]. Пакет – це групування пов'язаних елементів UML, таких як діаграми, документи, класи чи навіть інші пакети. Кожен елемент вкладений

Діаграма пакетів зображена на рис. 3.2.

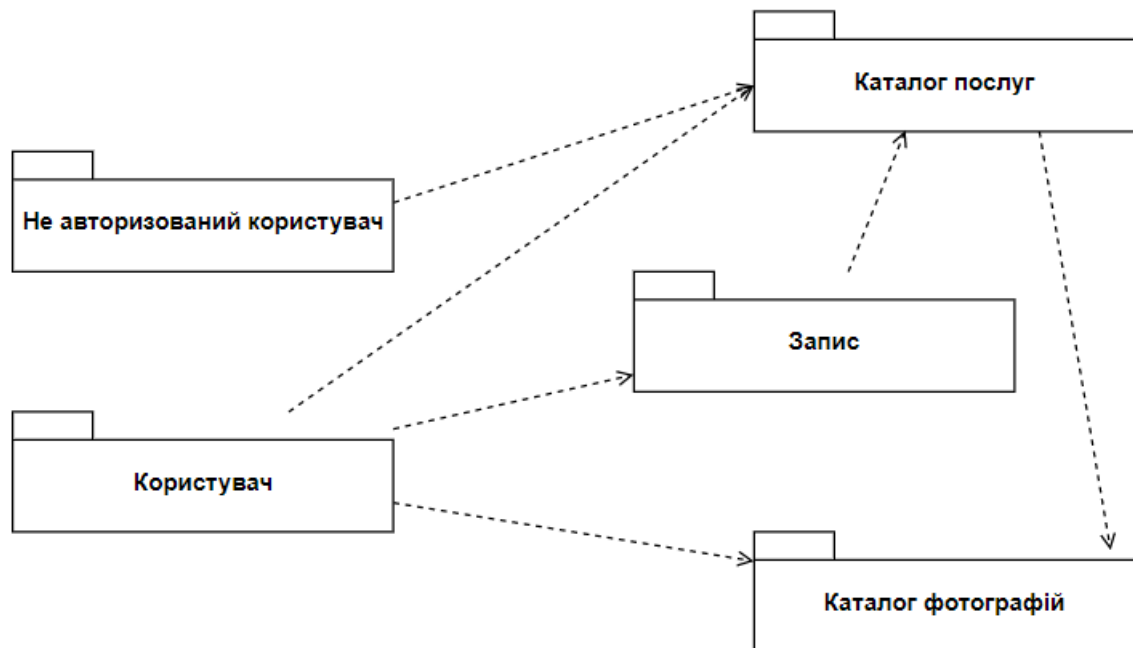


Рисунок 3.2 – Діаграма пакетів

Мета діаграми компонентів – показати взаємозв’язок між різними компонентами в системі. Для цілей UML 2.0 термін «компонент» відноситься до модуля класів, які представляють незалежні системи або підсистеми з можливістю взаємодії з рештою системи [17]-[18]. Існує цілий підхід до розробки, який обертається навколо компонентів: розробка на основі компонентів. У цьому підході діаграми компонентів дозволяють планувальнику ідентифікувати різні компоненти, щоб вся система виконувала те, що вона повинна робити. Частіше, в підході до програмування, діаграма компонентів дозволяє старшому розробнику групувати класи разом на основі спільної мети, щоб розробник та інші могли дивитися на проєкт розробки програмного забезпечення на високому рівні [17]-[18].

Діаграми компонентів для застосунку зображена на рис. 3.3.

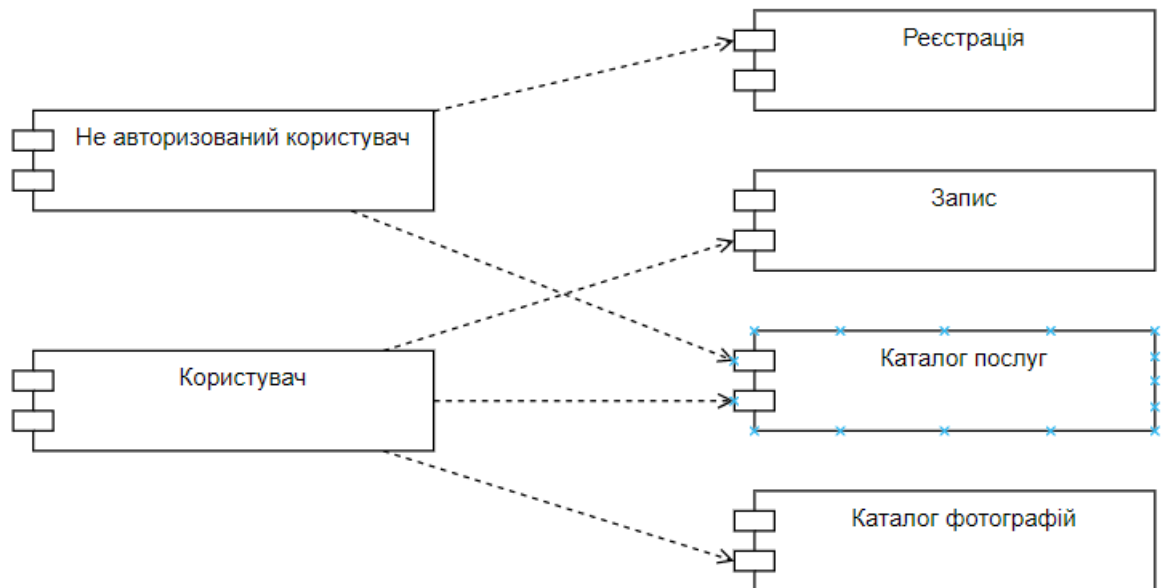


Рисунок 3.3 – Діаграма компонентів

Діаграми кооперацій використовуються, щоб показати, як об'єкти взаємодіють, щоб виконувати поведінку конкретного варіанту використання або частини варіанту використання [17]-[18]. Поряд із діаграмами послідовності, кооперація використовується дизайнерами для визначення й уточнення ролей об'єктів, які виконують певний потік подій у варіанті використання [17]-[18]. Вони є основним джерелом інформації, яка використовується для визначення обов'язків класу та інтерфейсів [17]-[18].

Діаграми кооперацій зображені на рис. 3.4 – 3.6.

Прецедент «Реєстрація на сайті салону краси»:

Основний потік:

- користувач натискає на посилання «Зареєструватися». Система відкриває нове віконце для реєстрації користувача;
- користувач вводить всі необхідні данні для реєстрації, а саме – логін, пароль, повторний пароль, ПІБ, дату народження, адресу, телефон та натискає «Ок», якщо користувач натискає «Скасувати» – альтернативний потік A1. Система перевіряє чи немає зареєстрованого користувача з таким логіном у системі, якщо є – альтернативний потік A2, якщо немає то система перевіряє чи паролі співпадають, якщо не співпали – альтернативний потік

A3, якщо співпали то система перевіряє чи всі поля заповнені, якщо ні – альтернативний потік A4, якщо так – то дані вносяться до БД клієнтів та видається повідомлення – «Реєстрація успішно завершена».

Альтернативні потоки:

A1: система видає повідомлення - «Реєстрація не завершена» та виходить з віконця реєстрації;

A2: система видає повідомлення – «Людина з таким логіном вже є!» і чекає введення іншого логіну;

A3: система видає повідомлення – «Введені різні паролі!» та чекає повторного введення паролей;

A4: система видає повідомлення – «Не всі поля заповнені!». Перехід на крок 2 основного потоку.



Рисунок 3.4 – Прецедент «Реєстрація на сайті салону краси»

Прецедент «Вхід на сайт салону краси».

Основний потік:

– користувач вводить логін та пароль;

– користувач натискає «Увійти». Система перевіряє чи всі дані введені, якщо ні – альтернативний потік A1, якщо так – то система перевіряє чи є зареєстрований користувач з таким логіном та паролем, якщо немає – альтернативний потік A2, якщо є – відбувається вхід на сайт.

Альтернативний потік:

A1: система видає повідомлення – «Не всі поля заповнені.» та повертається на крок 1 основного потоку;

A2: система видає повідомлення – «Користувача з такими даними немає» та повертається на крок 1 основного потоку.

Діаграми кооперації зображені на рис. 3.5 – 3.6.

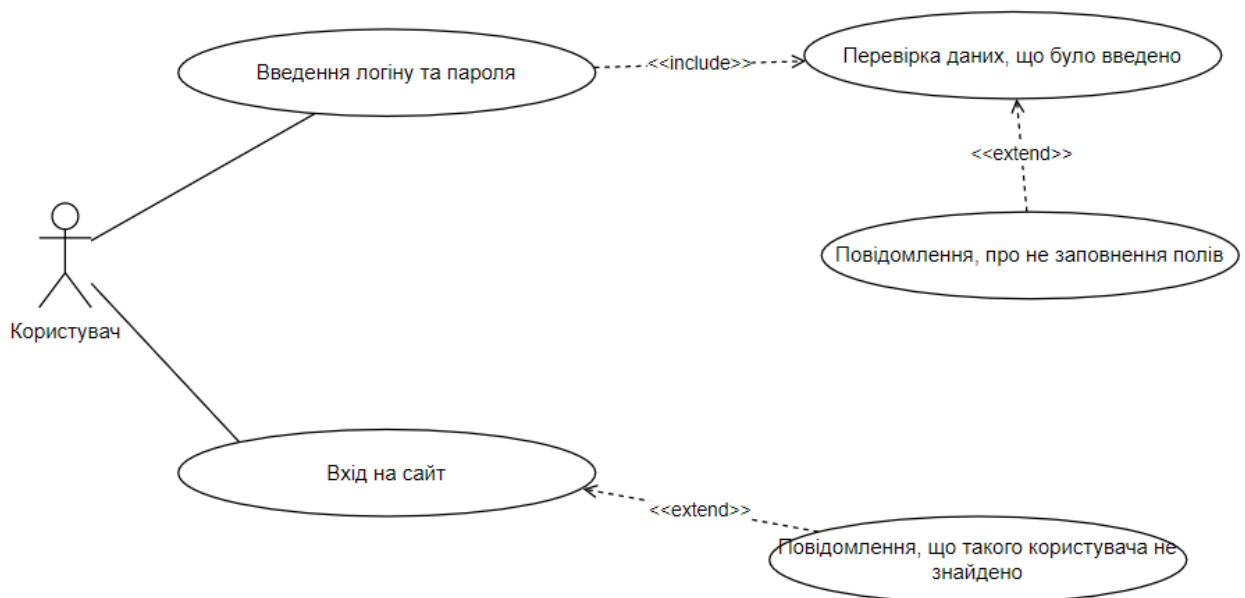


Рисунок 3.5 – Прецедент «Вхід на сайт салону краси»

Прецедент «Запис на прийом».

Основний потік:

– користувач обирає опцію «Записатись на прийом». Система відкриває вікно запису на прийом;

– користувач обирає дату, назву послуги, майстра та натискає «Ок», якщо користувач натискає «Скасувати» - альтернативний потік A1. Система

перевіряє чи всі поля заповнені, якщо ні – альтернативний потік A2, якщо так – система вносить до заявки ПІБ клієнта та зберігає її.

Альтернативний потік:

A1: система видає повідомлення – «Запис на прийом скасовано.» та закриває вікно запису на прийом;

A2: система видає повідомлення – «Не всі поля заповнені.» та повертається на крок 2 основного потоку.

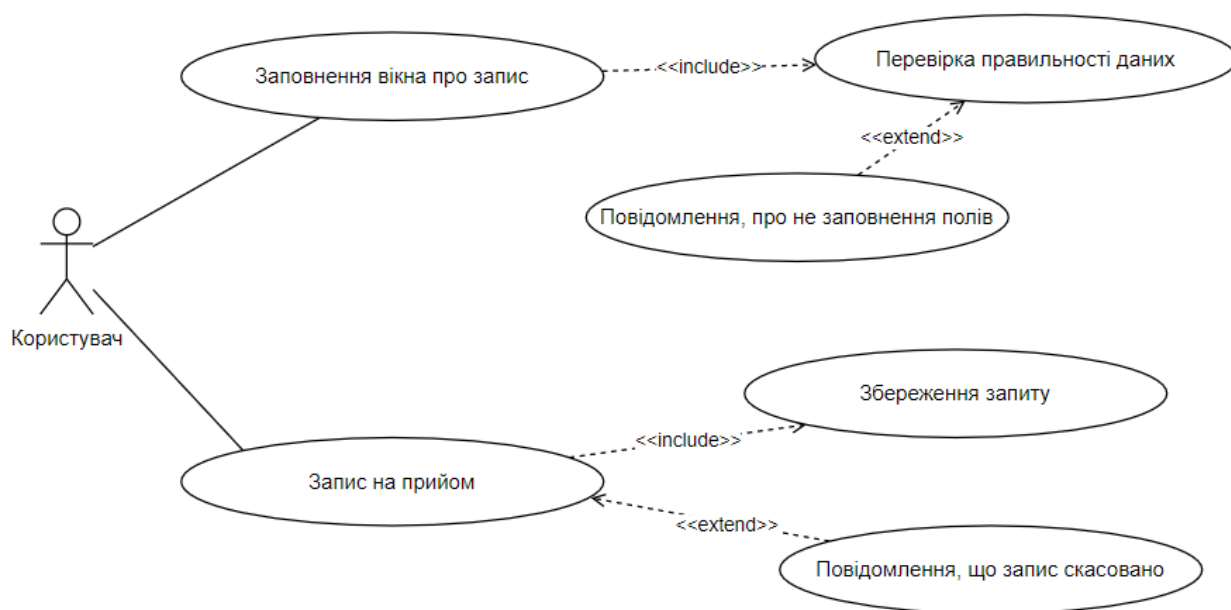


Рисунок 3.6 – Прецедент «Запис на прийом»

Діаграми послідовностей зображені на рис. 3.7–3.9.



Рисунок 3.7 – Обробка заявки на прийом

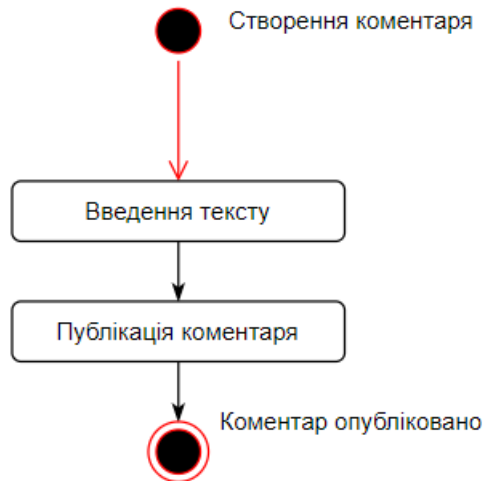


Рисунок 3.8 – Додавання коментарю

Діаграми послідовностей зображені на рис. 3.7–3.9.

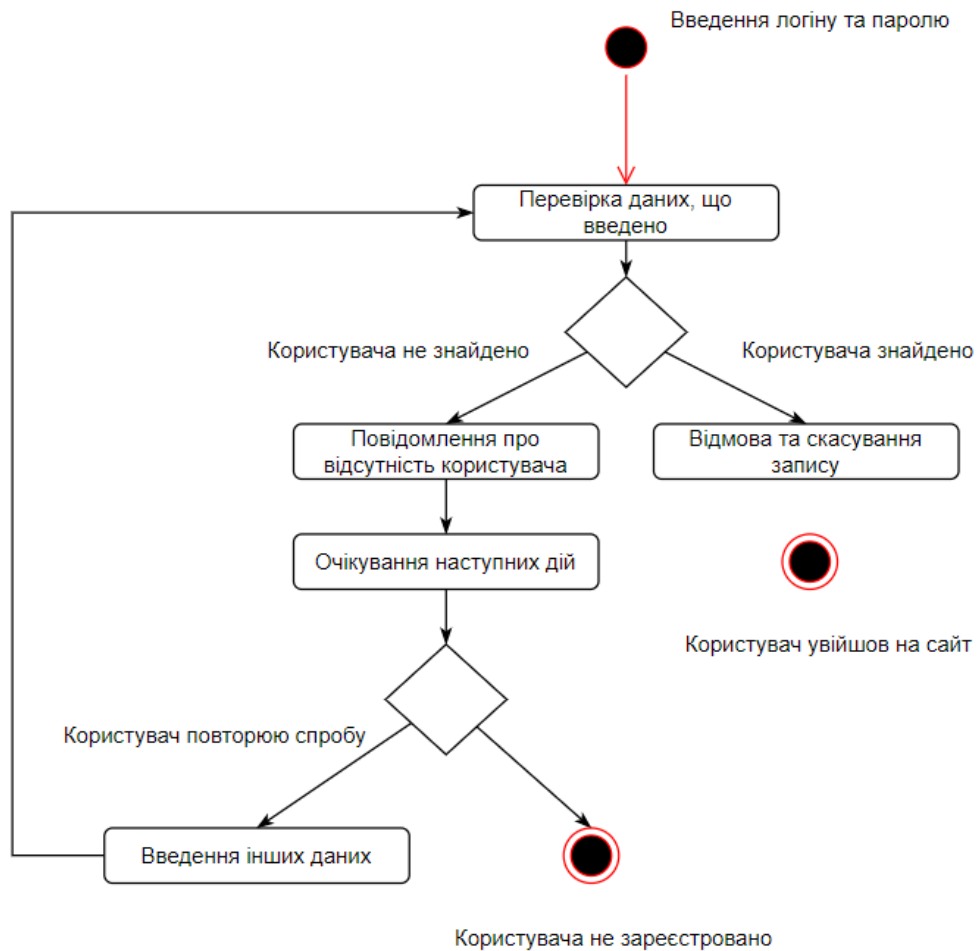


Рисунок 3.9 – Авторизація на сайті

Діаграма кооперації призначена для специфікації структурних аспектів взаємодії (рис. 3.10 та 3.11). Головна особливість діаграми кооперації полягає в можливості графічно уявити не тільки послідовність взаємодії, але і всі структурні відносини між об'єктами, що беруть участь у цій взаємодії.

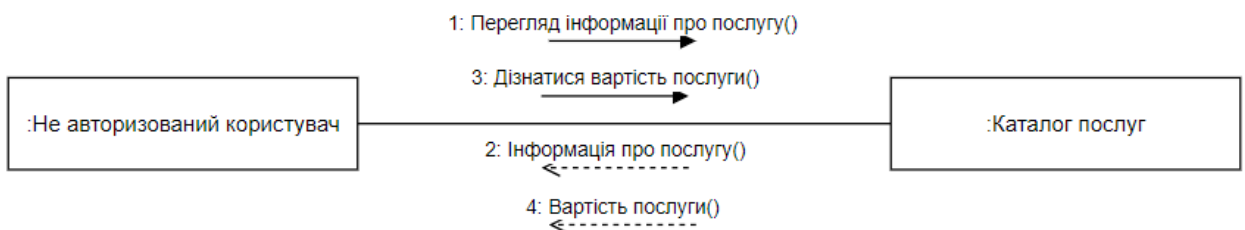


Рисунок 3.10 – Діаграма кооперацій перегляду інформації про послуги неавторизованим користувачем

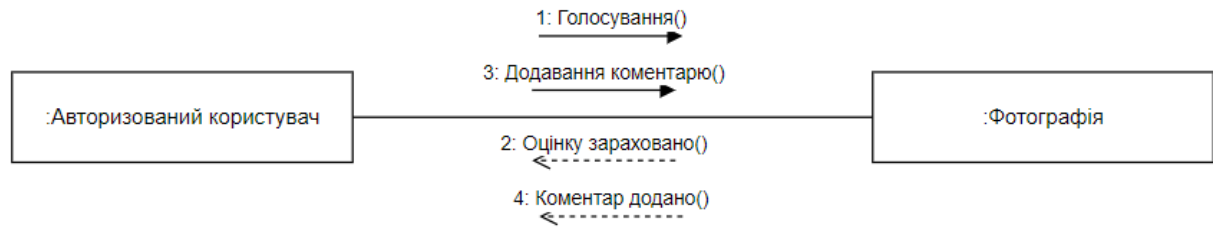


Рисунок 3.11 – Діаграма кооперацій додавання коментаря авторизованим користувачем

Перегляд інформації про послуги неавторизованим користувачем зображено на рис. 3.12.

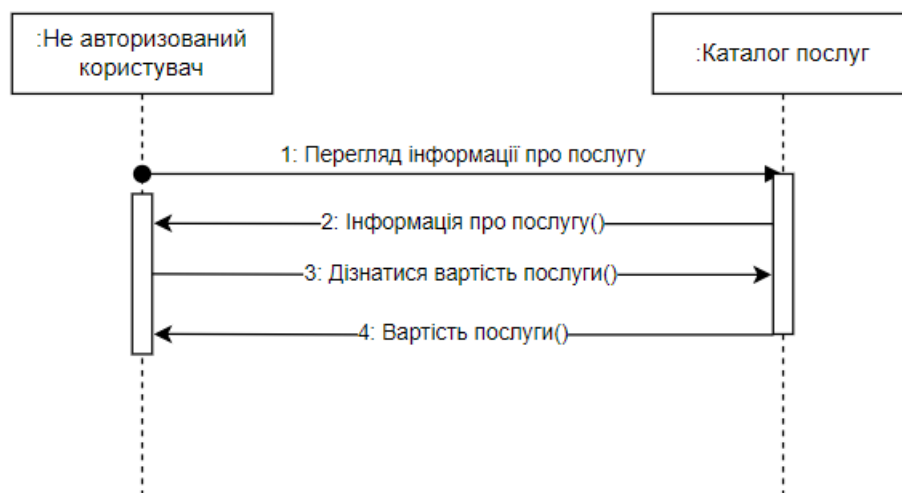


Рисунок 3.12 – Перегляд інформації про послуги неавторизованим користувачем

Додавання коментаря авторизованим користувачем зображено на рис. 3.13.

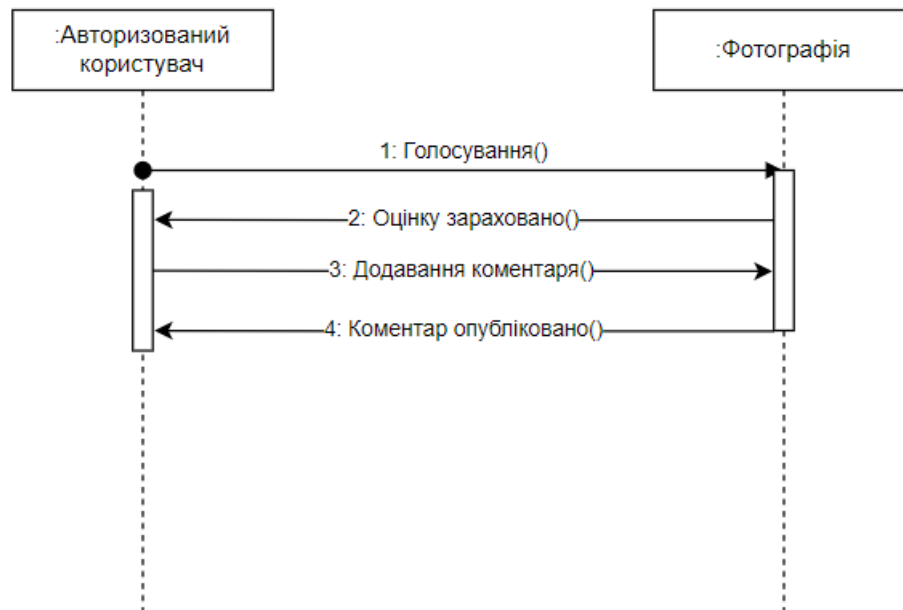


Рисунок 3.13 – Додавання коментаря авторизованим користувачем

3.2 Висновки за розділом 3

Було проведено ознайомлення з видами структурних діаграм та особливостями їх застосування, розроблено абстрактну модель системи мовою UML на основі структурних діаграм. Розроблено абстрактну модель системи за обраною темою на основі діаграму класів, пакетів та компонентів, на основі діаграм поведінки, прецедентів, діяльності та станів, кооперацій та послідовностей.

4 ЕКСПЛУАТАЦІЯ, ТЕСТУВАННЯ ТА ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ПРОГРАМИ

4.1 Опис застосунку

Функціональне призначення: інформаційна система у вигляді кроссплатформного вебзастосунку повинна надавати інформацію про барбершоп, послуги салону, вартість послуг та більш детальну інформацію про послуги.

Для зареєстрованих клієнтів – на сайті надається можливість переглядати особисту інформацію майстрів (ПІБ, телефон, розклад), надається можливість залишати відгуки про роботи майстрів та голосувати.

Кожен бажаючий може зареєструватись на сайті заповнивши відповідну форму, що з'являється при натисненні на посилання – «Зареєструватись».

Експлуатаційним призначенням сайту є збільшення кількості клієнтів барбершопу, надання їм необхідної інформації про послуги та роботи майстрів салону.

4.2 Умови виконання програми

Мінімальні системні вимоги та програмне забезпечення, що необхідні для функціонування серверної частини мобільного застосунку:

- сервер із встановленою серверною ОС;
- розгорнутий вебсервер;
- інтерпретатор PHP5;
- розгорнутий MySQL-сервер.

Для експлуатації клієнтської частини необхідні такі програмно-технічні засоби:

- IBM-сумісний ПК;
- процесор із тактовою частотою від 1,5 GHz;

- обсяг оперативної пам'яті – від 3GB (для безперебійної роботи браузерів);
- підключення до мережі Internet;
- встановлений браузер останньої версії.

4.3 Інструкція по експлуатації програми

4.3.1 Характеристики програми

Найбільш повно логічна взаємодія відображена на рис. 4.1, завдяки розподілу функцій між частинами застосунку.

До View (Представлення) – відноситься інтерфейс: сторінки з інформацією про салон, майстрів; посилання; кнопки; форми, що необхідні для реєстрації, запису на прийом, чи для видачі повідомлення про помилку; результати відображення запитів користувача.

Model зберігає дані класів «Запис», «Каталог послуг», «Послуга», «Каталог фотографій», «Фотографія».

Controller обробляє функції класів «Запис», «Каталог послуг», «Послуга», «Каталог фотографій», «Фотографія».

Функції класів розподілені між чотирма видами користувачів «Авторизований», «Майстер», «Адміністратор», «Не авторизований».

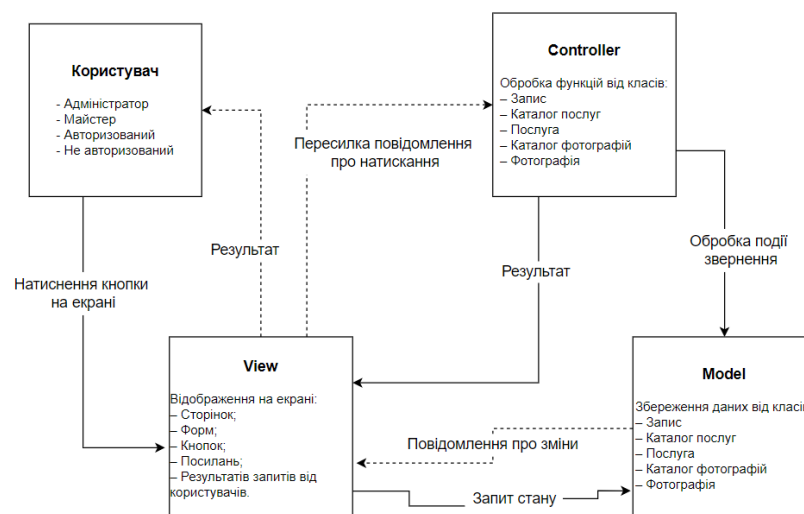


Рисунок 4.1 – Компоненти та зв'язки

4.3.2 Використані технічні засоби

Структура директорій і їх призначення в роботі Joomla – сайту [15]-[16]:

- а) cache – сюди централізована система кешування поміщає всі попередньо оброблені файли для прискорення видачі сторінок [15]-[16];
- б) administrator – директорія відповідає за панель управління сайту;
 - 1) components – місце установки компонентів для роботи адміністратора;
 - 2) includes – тут зберігаються файли, що реалізують Application Layer у додатку [15]-[16];
 - 3) language – в цій папці зберігаються локалізації панелі управління
 - 4) templates – директорія з шаблонами Back – End;
 - 5) modules – папка з адміністраторськими модулями, такими як toolbar, або панель швидкого доступу;
- в) components – в цій директорії зберігаються файли всіх встановлених в системі компонентів; Кожен компонент повинен мати певний набір файлів приписаний API і обраною моделлю; Інтерфейс прикладного програмування (іноді інтерфейс програмування додатків) (англ; Application Programming Interface, API) – набір готових класів, функцій, структур і констант, що надаються додатком (бібліотекою, сервісом) для її використання у зовнішніх програмних продуктах [15]-[16];
- г) images – директорія для зображень і файлів Joomla;
 - 1) stories – в яку зберігаються користувальницькі зображення;
 - 2) banners – тут розташовуються файли банерів (gif, jpg, swf і ін. формати) для стандартного компонента «Банери» [15]-[16];
 - 3) M_images – тут розташовуються іконки для сайту;

д) `includes` – папка з файлами, що реалізують Application Layer, проте безліч файлів в цій директорії залишені для сумісності зі старою версією Joomla;

е) `language` – локалізації для розширень користувальницької частини сайту;

ж) `libraries` – вміст цієї директорії реалізує Framework Layer CMS, в ній зберігається як сам фреймворк Joomla, так і сторонні бібліотеки, які необхідні для роботи;

з) `logs` – файли логів (рос; логов) розширень Joomla. Наприклад, сюди поміщаються записи подій, таких, як невірний логін на сайт;

и) `modules` – це модулі користувальницької частини сайту;

к) `plugins` – це обробники подій, вони викликаються компонентами в певний час генерації контенту, наприклад перед його розміщенням [15]-[16];

л) `templates` – користувальницькі шаблони Joomla, дуже сильні в своїй реалізації, при цьому вони залишаються досить прості для виготовлення навіть не фахівцем [15]-[16];

м) `temp` – папка для тимчасових файлів, наприклад – розпаковані архіви компонентів, тимчасові дані та інше;

н) `xmlrpc` – у цій папці зібрані файли, що реалізують доступ до сайту за протоколом XML-RPC (віддалений виклик процедур);

о) `index.php` – найголовніший файл. Саме цей файл запускає Joomla при звертанні користувача до сервера. Цей же файл запускає програму установки Joomla, якщо вона ще не була встановлена [15]-[16];

п) `error.php` – шаблон для помилки 404;

р) `templateDetails.xml` – описання шаблону; Налаштування шаблону;

Перелік доступних позицій;

с) `component.php` – шаблон виводу компонентів;

т) `params.ini` – файл для зберігання параметрів шаблону;

у) robots.txt – це файл, в якому записуються спеціальні інструкції для пошукових роботів;

ф) template.css – каскадна таблиця стилів шаблону;

Для створення вебзастосунку необхідно було реалізувати галерею зображень робіт майстрів. Для цього використовувався модуль JoomGallery [15]-[16].

JoomGallery – це галерея зображень для Joomla з великим і гнучким функціоналом.

Основні можливості галереї для Joomla – JoomGallery:

- різні способи завантаження зображень;
- величезна кількість налаштувань, які допоможуть відточити галерею під ваші потреби;
- можливість коментування зображень;
- можливість рейтингу зображень;
- налаштування прав доступу до категорій залежно від рівня доступу користувача;
- функція пошуку в галереї;
- можливість фільтрації за датою додавання, за рейтингом і т.п.
- можливість використання галереї всередині статей;
- необмежена вкладеність категорій.

Після встановлення, для виведення нашої галереї, для цього необхідно зробити кілька кроків [15]-[16]:

- створити категорії;
- завантажити фотографії в кожную з них;
- вивести за допомогою меню.

Типи меню в JoomGallery.

Галерея. Тут ми виводимо у вигляді блоків всі доступні галереї. При натисканні на одну з них ми можемо переглянути її вміст [15]-[16].

Детальний показ. Детальний відображення однієї з вибраних фотографій [15]-[16].

Категорії користувачів. Категорії створені окремим користувачем.

Моя галерея. Перегляд своїх зображень.

Топ списки. Перегляд найкращих зображень за певним критерієм.

Фаворити. Перегляд зображень , які ви додали до себе у вибране. Ви можете помітити всі потрібні вам зображення фаворитами і завантажити їх зір - архівом. Фаворити:табличний вигляд. Теж саме , що і звичайні фаворити, тільки список обраного відображається у вигляді таблиці [15]-[16].

Формуляр завантаження. Форма для завантаження зображень прямо з сайту [15]-[16].

Для розроблюваного сайту було використано перші два види.

Отже, нам було необхідно надати можливість голосувати за роботи майстрів та додавати коментарі, крім того переглядати всі роботи певного майстра. Для цього було створено окремі категорії, що належать певним працівникам салону та встановлена можливість виводу не лише самого списку працівників, а одразу ж і відображенні на обкладинці альбому – випадкову роботу майстра. Якщо ж відкрити альбом майстра то ми побачимо роботи майстра, кількість коментарів та оцінку. Для того, щоб додати власний коментар та проголосувати – необхідно відкрити зображення. Для переходу між зображеннями у цьому режимі можна використовувати стрілочки під фото [15]-[16].

4.3.3 Звернення до програми

Для завантаження сайту і взаємодії з ним досить набрати в адресному рядку браузера «barbershop.zp». Для входу в панель управління: «barbershop.zp\administrator».

4.3.4 Вхідні й вихідні дані

Програма не має конкретних вхідних даних для обробки, і ключем до доступу до роботи з нею є комбінація з логіна і пароля користувача чи адміністратора.

4.4 Методика та результати тестування

Тестування інформаційного порталу виконаємо в декілька етапів. На початку протестуємо функціональність застосунку, а на другому етапі виконаємо тестування за допомогою спеціального ПЗ –WAPT [19].

4.4.1 Тестування функціоналу

Основні сторінки застосунку зображені на рисунках 4.2 – 4.8.

На початку для користувача відкривається головна сторінка застосунку з якої він може перейти на окремі сторінки та розділи сайту.

Основна кольорова гама сайту поєднання чорного, білого та місцями кремових кольорів для акцентів, що гармонує із більшістю інтер'єрів таких закладів.

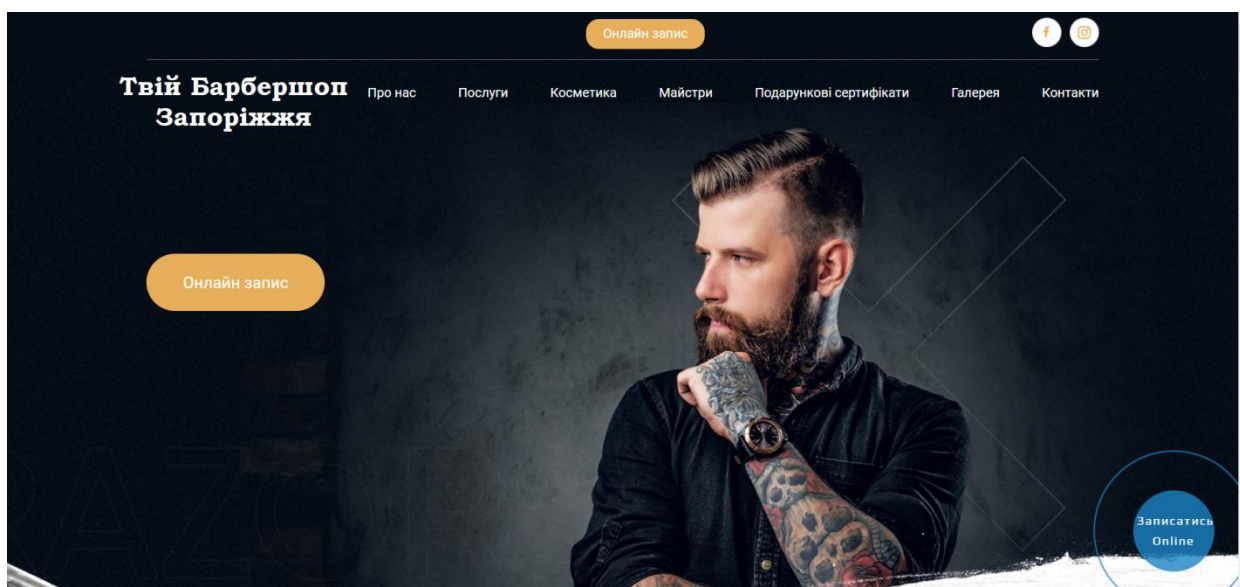


Рисунок 4.2 – Головна сторінка

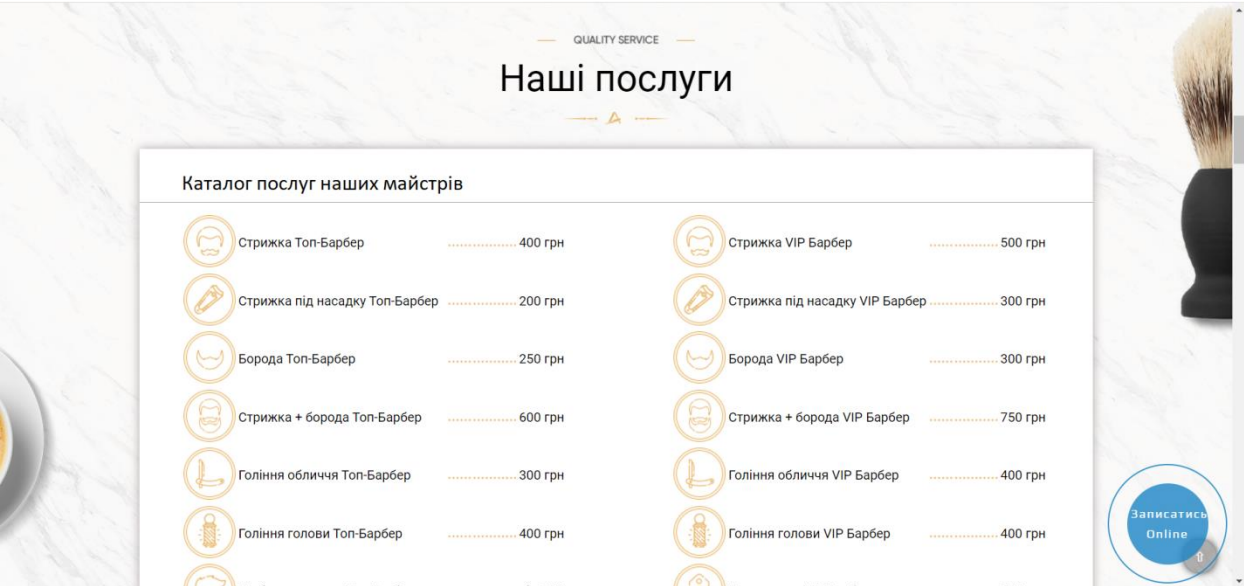
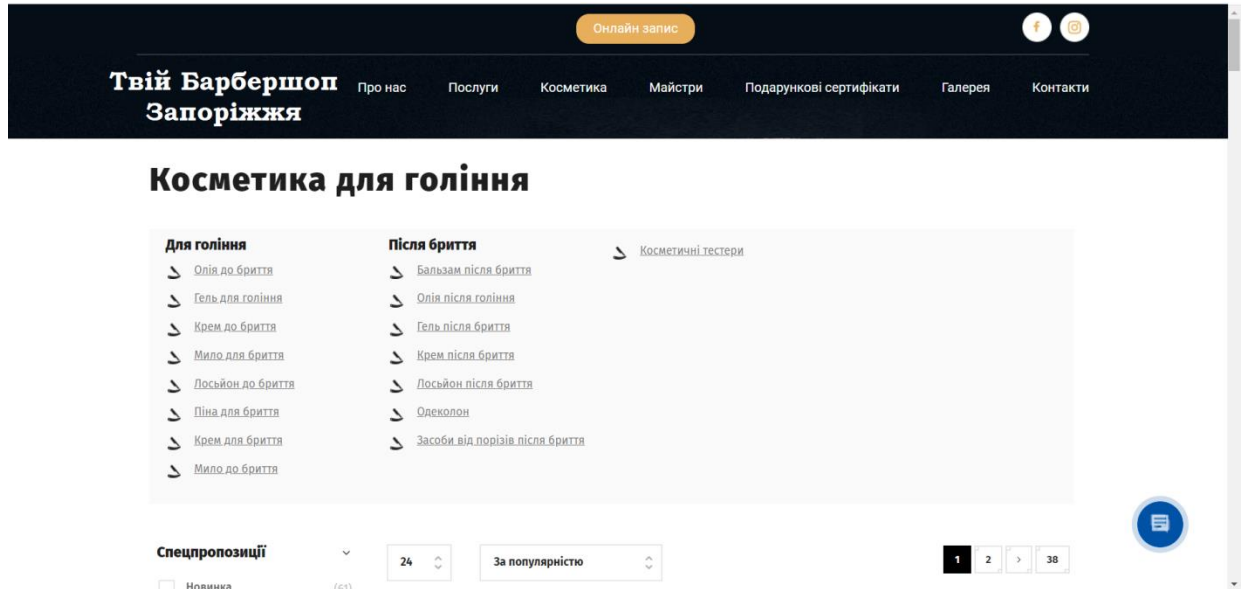


Рисунок 4.3 – Перелік послуг барбершопу



Риунок 4.4 – Сторінка із вибором косметики для докляду

Основні сторінки застосунку зображені на рисунках 4.5 – 4.6.

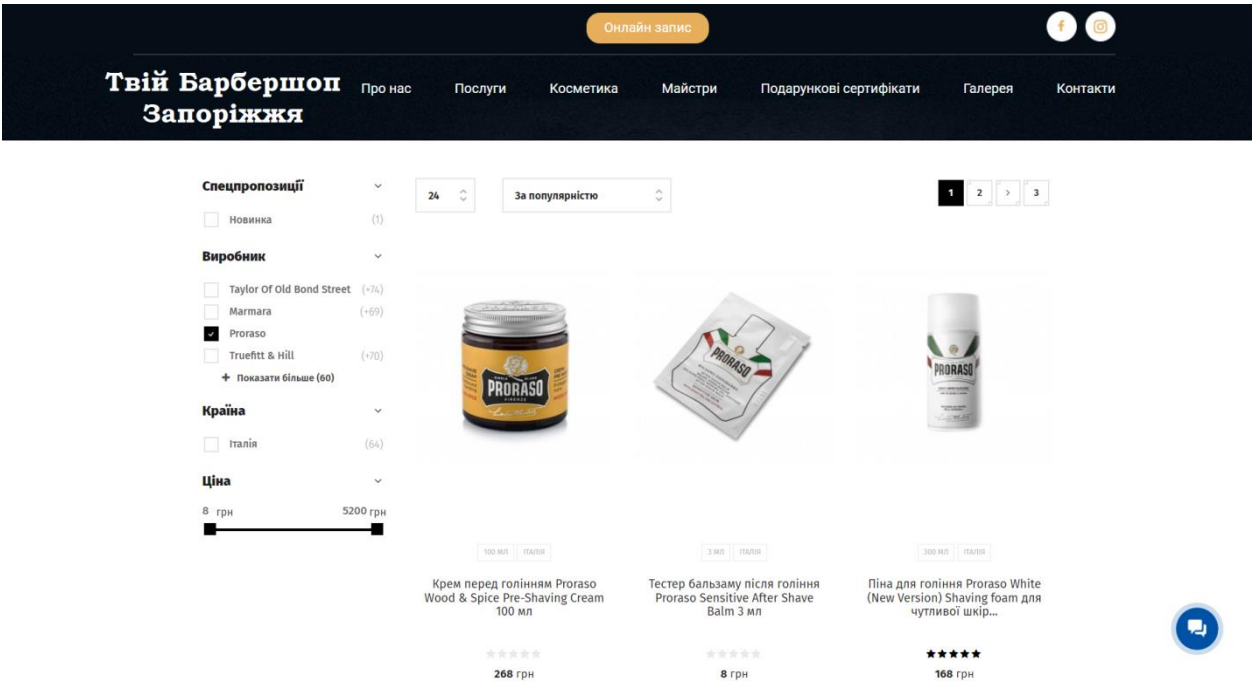


Рисунок 4.5 – Продовження каталогу косметики

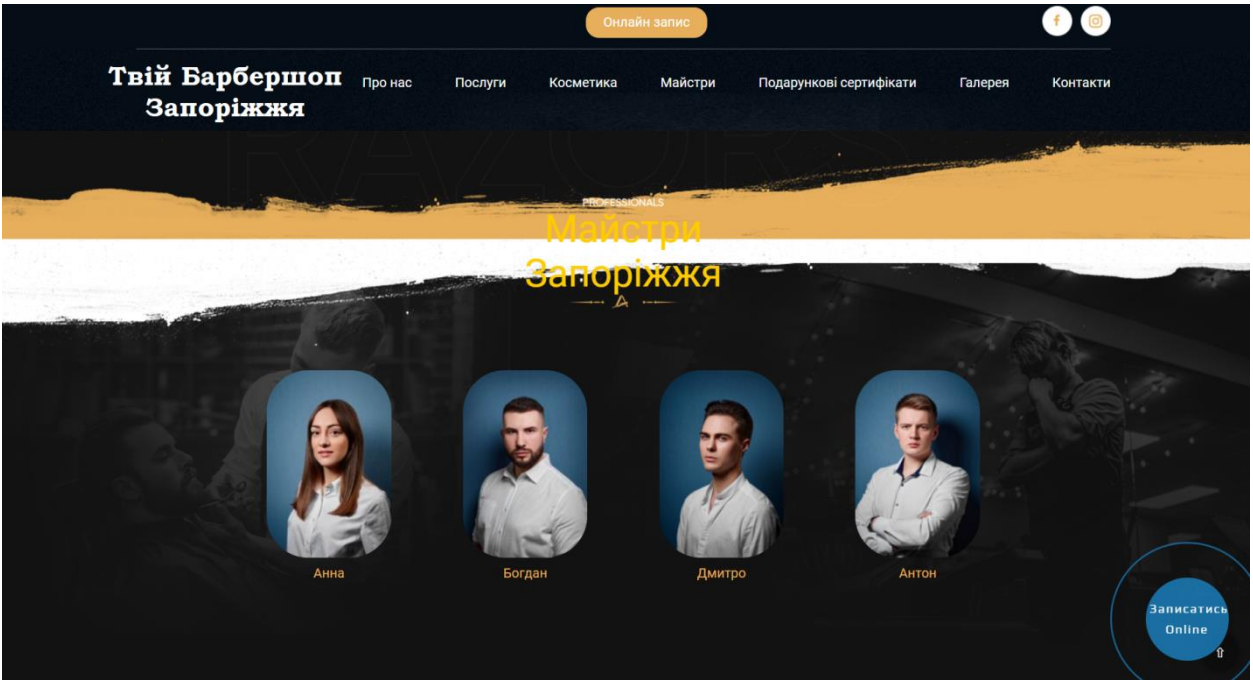


Рисунок 4.6 – Сторінка майстрів салону

Основні сторінки застосунку зображені на рисунках 4.7 – 4.8.

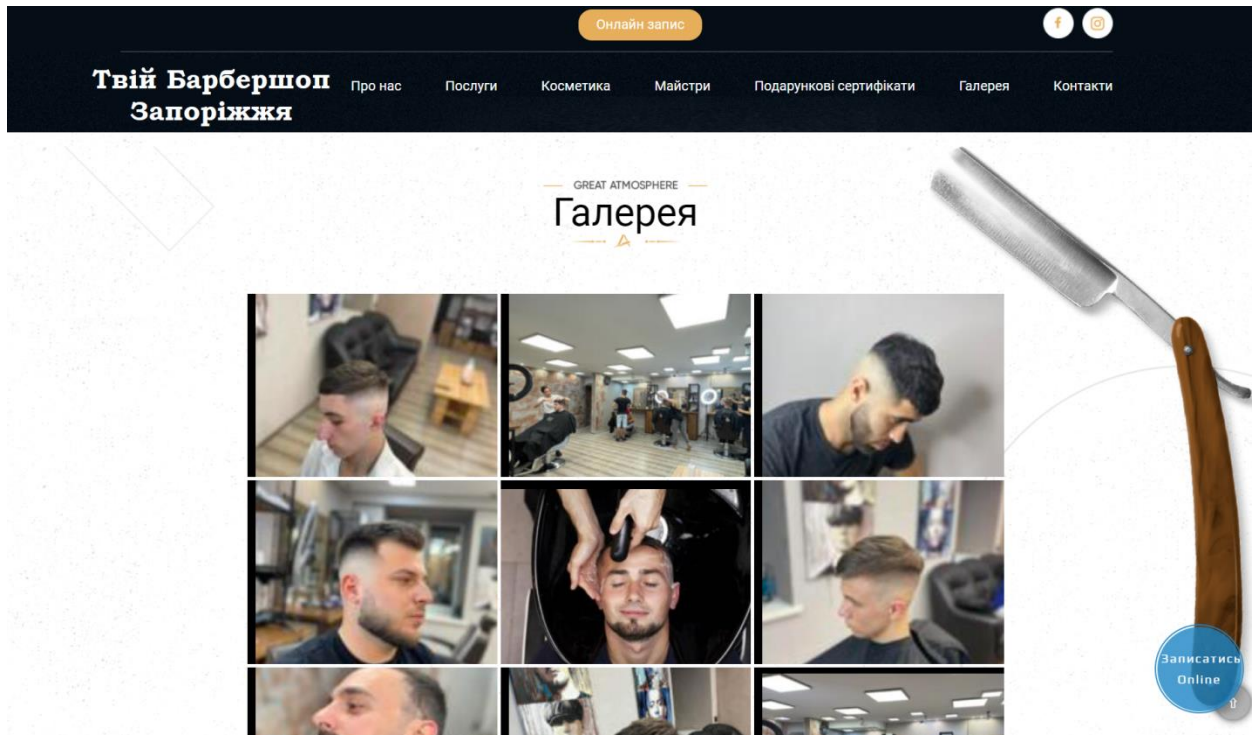


Рисунок 4.7 – Галерея робіт майстрів

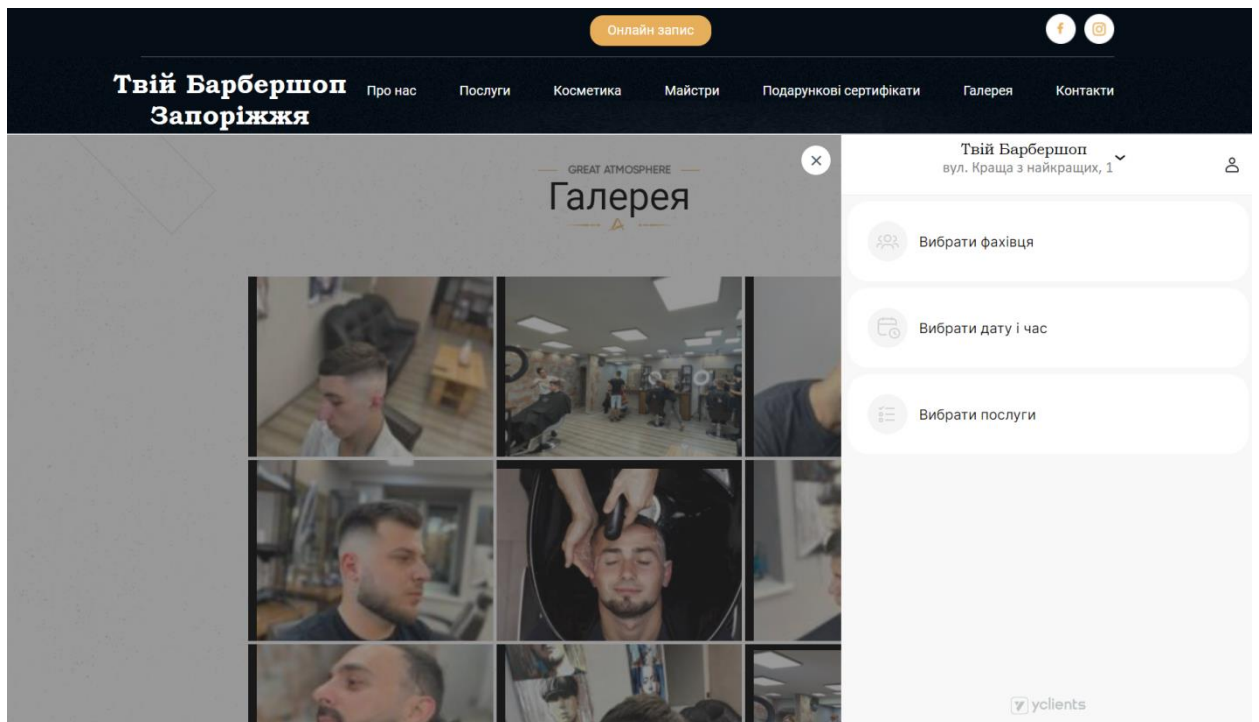


Рисунок 4.8 – Форма для запису на сайті

4.4.2 Тестування за допомогою WAPT

WAPT – це надійний, простий і економічний інструмент для навантажувального і стресового тестування будь-яких вебсайтів, включаючи бізнес-додатки, мобільні сайти, вебпортали і т.д. [19]. Він здатний дати відповіді на наступні питання продуктивності [19]:

- скільки одночасних користувачів може витримати вебсайт;
- скільки часу користувач чекає відповіді сервера на свій запит;
- як часто користувачі побачать помилки та перевантажують сторінки;
- чи витримає сайт стресове навантаження;
- чи зможе він повернутися до нормального функціонування після цього.

Кількісні результати наведено на рисунках 4.9 – 4.11.

Результати тестування:

Параметри виконання тесту:
Статус тесту: завершено
Тест розпочато: 12.04.2022 17:08:04
Коментарі до тесту:
Тест запущено на: МуРС
Тривалість тесту: 0:10:00

Остаточний звіт:

Профіль	Вдалих сесій	Помилков их сесій	Вдалих сторінок	Помилков их сторінок	Вдалих хітов	Помилков их хітов	Усього відправле но КБайт	Усього прийнято КБайт	Середній час відгуку (з урахуванн ям елементів сторінки)
Профіль1	10	0	182	0	1790	0	1271	37039	7.45(8.53)

Рисунок 4.9 – Загальні результати тестування

Дані продуктивності:

Вдалих сесій в секунду

Профіль	0:00:00- 0:01:00	0:01:00- 0:02:00	0:02:00- 0:03:00	0:03:00- 0:04:00	0:04:00- 0:05:00	0:05:00- 0:06:00	0:06:00- 0:07:00	0:07:00- 0:08:00	0:08:00- 0:09:00	0:09:00- 0:10:00	Усього
Профіль1	0	0	0	0.05	0.03	0	0.03	0.02	0.03	0	0.02
Усього	0	0	0	0.05	0.03	0	0.03	0.02	0.03	0	0.02

Вдалих сторінок в секунду

Профіль	0:00:00- 0:01:00	0:01:00- 0:02:00	0:02:00- 0:03:00	0:03:00- 0:04:00	0:04:00- 0:05:00	0:05:00- 0:06:00	0:06:00- 0:07:00	0:07:00- 0:08:00	0:08:00- 0:09:00	0:09:00- 0:10:00	Усього
Профіль1	0.20	0.37	0.32	0.30	0.37	0.28	0.33	0.30	0.33	0.23	0.30
Усього	0.20	0.37	0.32	0.30	0.37	0.28	0.33	0.30	0.33	0.23	0.30

Вдалих хітів в секунду

Профіль	0:00:00- 0:01:00	0:01:00- 0:02:00	0:02:00- 0:03:00	0:03:00- 0:04:00	0:04:00- 0:05:00	0:05:00- 0:06:00	0:06:00- 0:07:00	0:07:00- 0:08:00	0:08:00- 0:09:00	0:09:00- 0:10:00	Усього
Профіль1	2.53	3.17	3.60	2.97	2.77	2.72	3.12	2.40	3.38	3.18	2.98
Усього	2.53	3.17	3.60	2.97	2.77	2.72	3.12	2.40	3.38	3.18	2.98

Ресурси WAPT, %

Використання	0:00:00- 0:01:00	0:01:00- 0:02:00	0:02:00- 0:03:00	0:03:00- 0:04:00	0:04:00- 0:05:00	0:05:00- 0:06:00	0:06:00- 0:07:00	0:07:00- 0:08:00	0:08:00- 0:09:00	0:09:00- 0:10:00	Усього
Процесор	50	73	81	77	73	83	92	72	83	77	76
Пам'ять МБ(%)	178(7)	180(8)	180(8)	180(8)	179(8)	179(8)	180(8)	179(8)	179(8)	179(8)	179(7)
Мережа	0	0	0	0	0	0	0	0	0	0	0

Рисунок 4.10 – Результати тестування продуктивності

На рисунку 4.11 наведено тестування швидкості відгуку.

Час відгуку: Середній час відгуку (з урахуванням елементів сторінки)													
Ім'я	Час	0:00:00-0:01:00	0:01:00-0:02:00	0:02:00-0:03:00	0:03:00-0:04:00	0:04:00-0:05:00	0:05:00-0:06:00	0:06:00-0:07:00	0:07:00-0:08:00	0:08:00-0:09:00	0:09:00-0:10:00	Усього	
Профіль1.сторінка 1: http://barbershop.zp/		5	0	0	3	2	0	1	2	2	0	15	
	Мін	3.73 (3.98)	-	-	8.09 (8.88)	6.73 (8.56)	-	13.7 (14.5)	6.95 (7.58)	12.9 (15.2)	-	3.73 (3.98)	
	Макс	7.81 (8.41)	-	-	9.05 (9.88)	7.48 (9.08)	-	13.7 (14.5)	12.5 (13.4)	14.1 (17.1)	-	14.1 (17.1)	
	Серед не	4.90 (5.38)	-	-	8.43 (9.30)	7.11 (8.82)	-	13.7 (14.5)	9.74 (10.5)	13.5 (16.1)	-	8.28 (9.34)	
	Серед не90	7.08 (7.65)	-	-	9.05 (9.88)	7.48 (9.08)	-	13.7 (14.5)	12.5 (13.4)	14.1 (17.1)	-	12.8 (14.6)	
Профіль1.сторінка_2: http://barbershop.zp/templates/bee2_20/fonts/TitilliumMaps29L002.eo		4	1	0	3	2	0	1	2	2	0	15	
	Мін	0.02 (0.11)	0.05 (0.44)	-	0.05 (0.34)	0.03 (0.17)	-	0.05 (1.36)	0.03 (0.16)	0.06 (0.38)	-	0.02 (0.11)	
	Макс	0.05 (0.19)	0.05 (0.44)	-	0.27 (0.88)	0.03 (0.25)	-	0.05 (1.36)	0.16 (0.75)	0.13 (0.80)	-	0.27 (1.36)	
	Серед не	0.03 (0.13)	0.05 (0.44)	-	0.13 (0.55)	0.03 (0.21)	-	0.05 (1.36)	0.09 (0.45)	0.09 (0.59)	-	0.07 (0.43)	
	Серед не90	0.05 (0.18)	0.05 (0.44)	-	0.27 (0.88)	0.03 (0.25)	-	0.05 (1.36)	0.16 (0.75)	0.13 (0.80)	-	0.15 (0.90)	
Профіль1.сторінка_3: http://barbershop.zp/		2	3	0	0	3	2	0	3	0	2	15	
	Мін	5(5)	6.63 (6.63)	-	-	5.11 (5.11)	7.72 (7.72)	-	4.14 (4.14)	-	7.36 (7.36)	4.14 (4.14)	
	Макс	5.76 (5.76)	8.06 (8.06)	-	-	9.55 (9.55)	9.95 (9.95)	-	10.6 (10.6)	-	11.9 (11.9)	11.9 (11.9)	
	Серед не	5.38 (5.38)	7.16 (7.16)	-	-	7.00 (7.00)	8.84 (8.84)	-	8.33 (8.33)	-	9.63 (9.63)	7.68 (7.68)	
Серед дне9		5.76 (5.76)	8.06 (8.06)	-	-	9.55 (9.55)	9.95 (9.95)	-	10.6 (10.6)	-	11.9 (11.9)	10.7 (10.7)	

Рисунок 4.11 – Тестування швидкості відгуку

Візуальне представлення тестування створеного вебзастосунку приведено нижче на рисунках 4.12 та 4.13 .

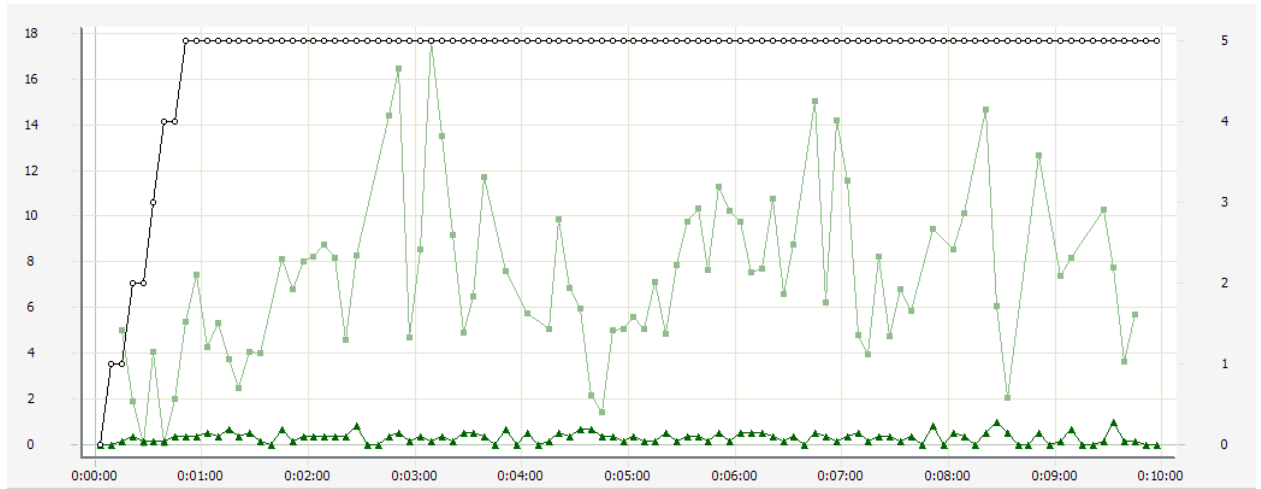


Рисунок 4.12 – Графік продуктивності

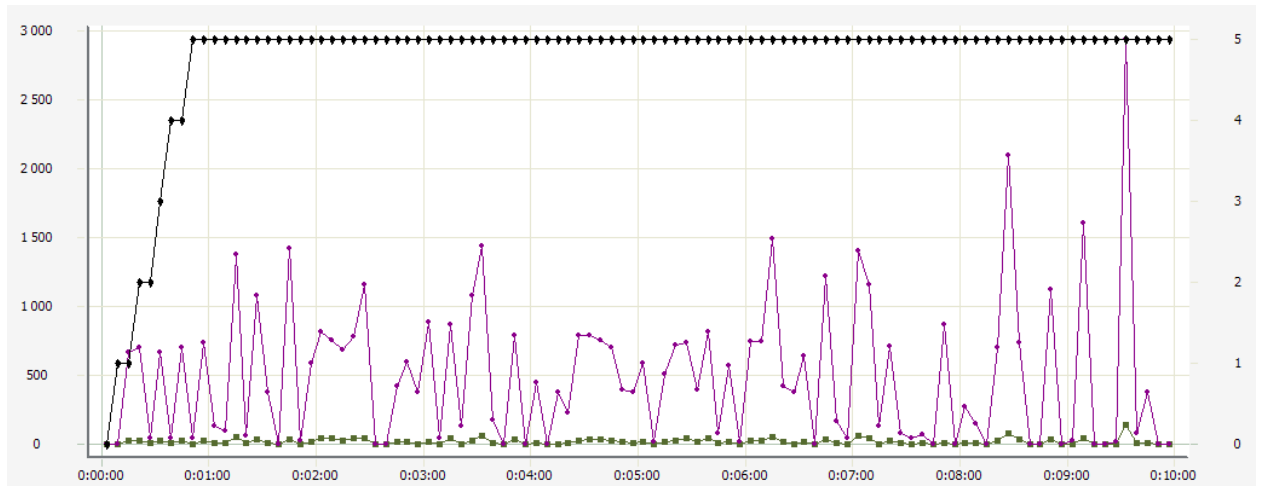


Рисунок 4.13 – Графік ширини каналу

ВИСНОВКИ

У дипломній кваліфікаційній роботі бакалавра виконано програмну реалізацію інформаційної системи для барбершопу у вигляді кроссплатформного вебзастосунку. Для досягнення головної мети роботи було виконано ряд задач. Так на початку було виконано аналіз та документування вимог до ПЗ. Визначено функціональність системи для розроблення технічного завдання. Визначили характеристику області застосування ПЗ, підстави розроблення системи та призначення розробки. Сформулювали технічне завдання на проєкт системи за обраною темою.

Пізніше, використовуючи засоби мови UML, було розроблено абстрактні моделі системи на основі діаграм: класів, пакетів, компонентів, поведінки, прецедентів, діяльності, станів, кооперацій та послідовностей.

Після цього використовуючи архітектуру шаблону MVC було розроблено та проведено аналіз архітектури вебзастосунку на основі предметної області. Розроблено архітектуру з використанням CMS Joomla. Використання модулів розширення функціональності дозволило спростити та прискорити розробку.

Проведене тестування створеного вебзастосунку за допомогою WAPT засвідчило прийнятні результати. Тестування функціональності також дало позитивний результат.

Останнім етапом стало складання та оформлення відповідної проєктної документації до власного ПЗ з метою фіксації усіх етапів розробки та напрацювань.

Усі завдання дипломної кваліфікаційної роботи бакалавра повністю виконано.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Стеф Ю. Салонний business. Все про організацію бізнесу та створення успішного проєкту [Текст] : [посібник] / Юлія Стеф. – Дніпро : Ліра, 2020. – 151 с.
2. Сівко Л. В. Аналіз характеристик вебсайтів з точки зору забезпечення комунікативної складової бізнес-діяльності підприємств [Текст] / Л. В. Сівко, О. В. Дорохов, М. Драшкович // Системи обробки інформації. – 2011. – Вип. 8. – С. 241-244.
3. Власова Т. Р. Власний сайт як фактор розширення можливостей фірми у веденні бізнесу [Текст] / Т. Р. Власова, Д. В. Затуловський // Вісник Чернівецького торговельно-економічного інституту. Економічні науки. – 2010. – Вип. 2. – С. 143-149.
4. Britvenko A. S. Internet-marketing as a tool for product promotion [Текст] / A. S. Britvenko, V. V. Bilousova, D. D. Poliah // Вісн. Бердян. ун-ту менеджменту і бізнесу. – 2018. – № 1. – С. 29-33.
5. Ліфінцев Д. С. Трансляція цінностей корпоративної культури як інструмент формування іміджу компанії [Текст] / Д. С. Ліфінцев // Бізнес Інформ. – 2014. – № 12. – С. 440-443.
6. Отенко І. П. Особливості корпоративної культури реалізації організаційних змін промисловими підприємствами [Текст] / І. П. Отенко, М. І. Чепелюк // Бізнес Інформ. – 2015. – № 10. – С. 396-402.
7. Інформаційні технології та інновації в економіці, управлінні проєктами і програмами : монографія [Текст] / М. В. Булаєнко, В. М. Вартанян, Г. М. Верясова, Н. В. Гибкіна, Ю. Ю. Гусєва, О. М. Гуца, Н. В. Довгопол, Н. В. Доценко, С. І. Доценко, А. М. Зеглам та ін.; ред.: В. О. Тімофєєв, І. В. Чумаченко; Харків. нац. ун-т радіоелектроніки. – Харків : Панов А. М., 2016. – 402 с.

8. Проект «Sborodoy» / [Електрон. ресурс] – Режим доступу: <https://sborodoy.com/>.
9. Frisor Барбершоп / [Електрон. ресурс] – Режим доступу: <https://frisor.ua/uk/>.
10. FIRM Barbershop / [Електрон. ресурс] – Режим доступу: <https://firmbarbershop.com/uk/zaporizhya/>.
11. Костенко О. П. Розробка методу захисту інформації у веббазованих додатках [Текст] / О. П. Костенко, С. М. Маркевич, Ю. О. Величай // Нові технології. – 2013. – № 1-2. – С. 78-81.
12. Канакін А. Е. Розробка елементів рекомендаційної системи на базі вебаналітики [Текст] / А. Е. Канакін, І. С. Скарга-Бандурова, М. Є. Щербакова // Вісник Східноукраїнського національного університету імені Володимира Даля. – 2018. – № 6. – С. 199-204.
13. What is PHP? The PHP Programming Language Meaning Explained / [Electronic resource]. – Access mode: <https://www.freecodecamp.org/news/what-is-php-the-php-programming-language-meaning-explained/>.
14. PHP Introduction / [Electronic resource]. – Access mode: https://www.w3schools.com/php/php_intro.asp.
15. North B.M. Joomla! A user's guide [Text] : building a successful Joomla! Powered website / Barrie M. North. – Upper Saddle River, New Jersey[etc.] : Prentice Hall, 2008.
16. Рамел Д. Joomla! для профессионалов [Текст] / Дэн Рамел ; [пер. с англ. и ред. И. В. Берштейна]. – Киев : Вильямс, 2014. – 441 с.
17. Гудзовата О. О. Використання уніфікованої мови візуального моделювання UML (Unified Modeling Language) як інструменту підтримки проєктування інформаційних систем [Текст] / О. О. Гудзовата, В. І. Костирко, І. В. Артищук // Підприємництво і торгівля. – 2019. – Вип. 24. – С. 108-114.

18. Богдан И. Классификация ошибок на UML-диаграммах, возникающих в ходе разработки IT-проектов [Текст] / И. Богдан, А. Задорожний // Технічні науки та технології. – 2018. – № 1. – С. 68-78.

19. WAPT / [Electronic resource]. – Access mode: <https://www.loadtestingtool.com/>.

ДОДАТОК А
Технічне завдання

Вступ

Даний проєкт – створення інформаційної системи для барбершопу у вигляді розробки кроссплатформного вебзастосунку. Створений інтернет-ресурс, дозволить збільшити популярність салону, полегшити роботу та оптимізувати її. За допомогою цього проєкту люди зможуть отримувати інформацію про послуги салону, переглянути роботи майстрів та зареєструвавшись залишити власні відгуки, як про роботи так і про майстрів, переглянути особисту інформацію майстрів та записатись на прийом.

A.1 Підстави для розробки

Підставою для розробки є завдання на дипломну кваліфікаційну роботу, яке було визначене за темою «Програмна реалізація інформаційної системи для барбершопу» і затверджене наказом Національного університету «Запорізька політехніка» № 102 від 27 квітня 2022 р.

A.2 Призначення розробки

Інтернет-ресурс повинен надавати інформацію про салон краси, послуги салону, вартість послуг та більш детальну інформацію про послуги.

Для зареєстрованих клієнтів – на сайті надається можливість переглядати особисту інформацію майстрів (ФІО, телефон, розклад), надається можливість залишати відгуки, як про самих майстрів так і про їх роботи.

Кожен бажаючий може зареєструватись на сайті заповнивши відповідну форму, що з'являється при натисненні на посилання – «Зареєструватись».

Блок авторизації (поля логіна і пароля) повинен бути на всіх сторінках сайту. Також, в цьому блоці присутнє посилання: «Забули пароль?», «Зареєструватися».

Після натиснення на «Забули пароль?» відкривається вікно в якому є поле для введення електронної адреси на яку будуть відправлені особисті дані клієнта.

А.3 Основні вимоги до програми, що розробляється

А.3.1 Вимоги до функціональних характеристик

Функціональні вимоги до системи, що розробляється.

Робота з сайтом:

- а) реєстрація нових користувачів за допомогою форми (логін, пароль);
- б) авторизація на сайті;
- в) видалення визначеного користувача з бази даних системи;
- г) перегляд контактної інформації салону;
- д) перегляд додаткової інформації про послуги;
- е) перегляд вартості послуг;
- ж) робота з інформацією обмеженого доступу:
 - 1) додавання та видалення коментарів до фотографій;
 - 2) додавання оцінок до фотографій;
 - 3) додавання відгуків та видалення про майстра;
 - 4) перегляд особистої інформації про майстра (телефон, часи роботи (розклад), список записаних на прийом)
 - 5) запис на прийом;
- з) редагування сайту адміністратором:
 - 1) реєстрація майстрів;
 - 2) видалення майстрів;
 - 3) редагування інформації про майстра;
 - 4) редагування інформації про салон (часи роботи, контактну інформацію, вартість послуг);
 - 5) додавання та видалення робіт майстрів (фото).

А.3.2 Вимоги до надійності

Для великих інтернет – ресурсів важливо подбати про продуктивність системи, вони повинні витримувати великі навантаження. Редагувати, додавати або видаляти основну інформацію може лише адміністратор.

Інтернет – ресурс повинен відкриватись в усіх браузерях, чітко відтворювати необхідну інформацію і зображення. Програмний продукт переноситься на будь – якому комп'ютері, планшеті чи ноутбукі.

А.3.3 Умови експлуатації

Висуваються такі вимоги:

- наявність браузера для перегляду сайту;
- наявність встановленого на клієнтській стороні Flashplayer;
- наявність вебсервера Denwer для тестування і налагодження.

Усі користувачі можуть переглядати інформацію про салон, майстрів та роботи майстрів.

Реєстрування відбувається за допомогою допоміжної форми з логіком, паролем та особистою інформацією про клієнта.

Авторизовані користувачі системи можуть переглядати більше інформації про салон та майстрів, залишати відгуки про майстрів, оцінювати та коментувати їх роботи, записуватись на прийом.

Користувачі, що є майстрами – можуть переглядати список записаних на прийом клієнтів.

Тільки адміністратор може редагувати дані на сайті, крім того вносити до системи майстрів салону, редагувати їх особисту інформацію та видаляти їх.

A.3.4 Вимоги до складу та параметрів технічних засобів

Мінімальні системні вимоги та програмне забезпечення, що необхідні для функціонування серверної частини мобільного застосунку:

- сервер із встановленою серверною ОС;
- розгорнутий вебсервер;
- інтерпретатор PHP5;
- розгорнутий MySQL-сервер.

Для експлуатації клієнтської частини необхідні такі програмно-технічні засоби:

- IBM-сумісний ПК;
- процесор із тактовою частотою від 1,5 GHz;
- обсяг оперативної пам'яті – від 3GB (для безперебійної роботи браузерів);
- підключення до мережі Internet;
- встановлений браузер останньої версії.

A.4 Стадії та етапи розробки

На цьому етапі необхідно сформулювати основну ідею майбутнього сайту.

Визначити мету сайту – музичний інтернет-магазин.

Визначити цільову аудиторію (стать, вік, освіта), тобто для кого буде цей сайт.

З'ясувати, яку інформацію очікують відвідувачі, що допоможе з визначенням контенту (текстовою інформацією) на сайті.

Визначитися з технічною стороною. Важливо враховувати середнє підключення до мережі, а так само програмним забезпеченням.

На цьому ж етапі складається ТЗ (Технічне завдання).

Проектування системи

Проектується система, враховуючи усі потреби замовника.

Проектування інтерфейсу

На цьому етапі проектується інтерфейс сайту. Він повинен бути зручний та зрозумілий у використанні.

Розробка програмного продукту

На цьому етапі відбувається аналіз структури та розгортання вебдодатків.

Програмується вебсайт з урахуванням усіх проектних рішень.

Тестування системи

Після завершення всіх робіт з інтеграції систем і налагодження обладнання інтернет-магазин буде надано на тестування експерту.

Наповнення контентом

Ефективність роботи сайту пов'язана з поняттям контенту, тобто текстової та графічної інформації. На цьому етапі сайт наповнюється контентом за бажанням Замовника.

Оформлення документації та передання замовнику

Після завершення всіх процедур і робіт з тестування сайту і навчання фахівців групи підтримки проєкт буде готовий до відкриття для відвідувачів. На даному етапі оформлюється документація та підписуються необхідні документи про приймання – здачі програмного продукту.

ДОДАТОК Б
Опис програми

Б.1 Загальні відомості

Даний проєкт – створення інформаційної системи для барбершопу у вигляді розробки кроссплатформного вебзастосунку. Створений інтернет-ресурс, дозволить збільшити популярність салону, полегшити роботу та оптимізувати її. За допомогою цього проєкту люди зможуть отримувати інформацію про послуги салону, переглянути роботи майстрів та зареєструвавшись залишити власні відгуки, як про роботи так і про майстрів, переглянути особисту інформацію майстрів та записатись на прийом.

Б.2 Функціональне призначення

Інтернет-ресурс повинен надавати інформацію про салон краси, послуги салону, вартість послуг та більш детальну інформацію про послуги.

Для зареєстрованих клієнтів – на сайті надається можливість переглядати особисту інформацію майстрів (ФІО, телефон, розклад), надається можливість залишати відгуки, як про самих майстрів так і про їх роботи.

Кожен бажаючий може зареєструватись на сайті заповнивши відповідну форму, що з'являється при натисненні на посилання – «Зареєструватись».

Блок авторизації (поля логіна і пароля) повинен бути на всіх сторінках сайту. Також, в цьому блоці присутнє посилання: «Забули пароль?», «Зареєструватися».

Після натиснення на «Забули пароль?» відкривається вікно в якому є поле для введення електронної адреси на яку будуть відправлені особисті дані клієнта.

Б.3 Опис логічної структури

Найбільш повно логічна взаємодія відображена на рис. Б.1, завдяки розподілу функцій між частинами застосунку.

До View (Представлення) – відноситься інтерфейс: сторінки з інформацією про салон, майстрів; посилання; кнопки; форми, що необхідні для реєстрації, запису на прийом, чи для видачі повідомлення про помилку; результати відображення запитів користувача.

Model зберігає дані класів «Запис», «Каталог послуг», «Послуга», «Каталог фотографій», «Фотографія».

Controller обробляє функції класів «Запис», «Каталог послуг», «Послуга», «Каталог фотографій», «Фотографія».

Функції класів розподілені між чотирма видами користувачів «Авторизований», «Майстер», «Адміністратор», «Не авторизований».

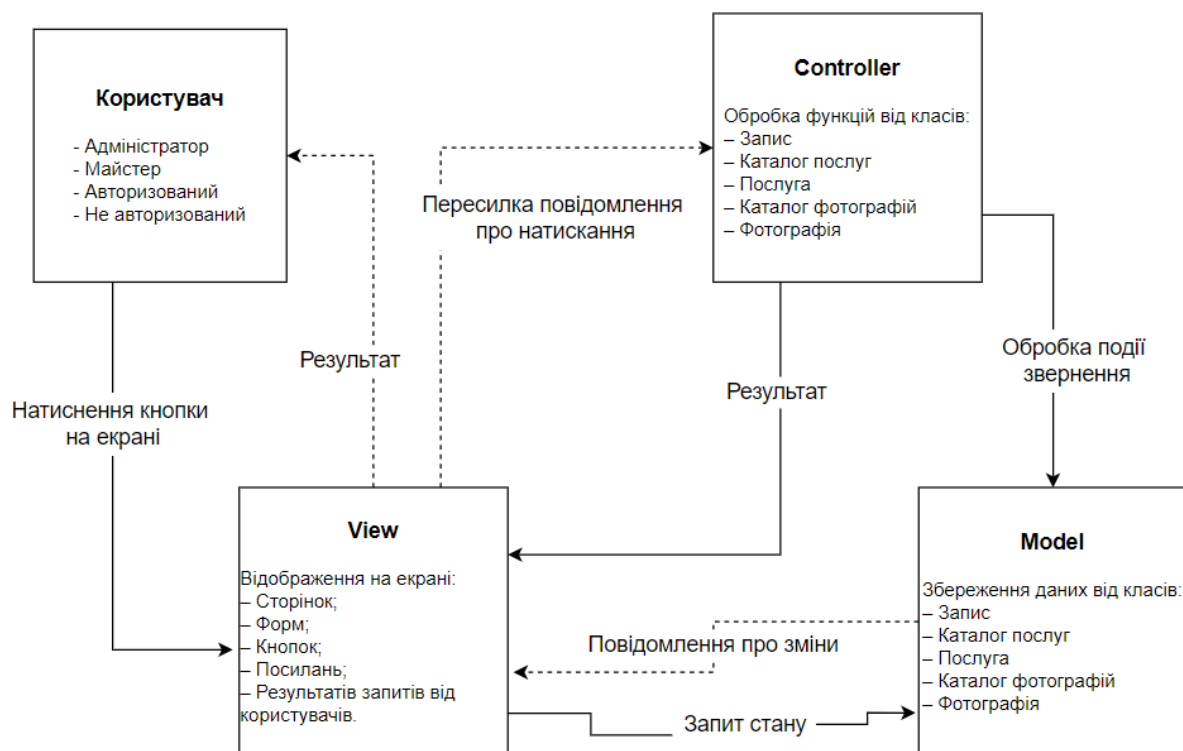


Рисунок Б.1 – Компоненти та зв'язки

Б.4 Використані технічні засоби

Мінімальні системні вимоги та програмне забезпечення, що необхідні для функціонування серверної частини мобільного застосунку:

- сервер із встановленою серверною ОС;

- розгорнутий вебсервер;
- інтерпретатор PHP5;
- розгорнутий MySQL-сервер.

Для експлуатації клієнтської частини необхідні такі програмно-технічні засоби:

- IBM-сумісний ПК;
- процесор із тактовою частотою від 1,5 GHz;
- обсяг оперативної пам'яті – від 3GB (для безперебійної роботи браузерів);
- підключення до мережі Internet;
- встановлений браузер останньої версії.

Б.5 Виклик та завантаження

Для завантаження сайту і взаємодії з ним досить набрати в адресному рядку браузера «barbershop.zp». Для входу в панель управління: «barbershop.zp/administrator».

Б.6 Вхідні та вихідні дані

Програма не має конкретних вхідних даних для обробки, і ключем до доступу до роботи з нею є комбінація з логіна і пароля користувача чи адміністратора.

ДОДАТОК В
Текст програми

```

<?php
//progect name: BARBERSHOP1

include_once '../models/BodyModel.php';
include_once '../models/ObjMod.php';
include_once '../models/UseMod.php';
include_once '../models/AvModel.php';
include_once '../models/OprtnMod.php';
include_once '../models/ShedMod.php';
/*
include_once '../models/OperationsVvModel.php';
include_once '../models/OprtnMod.php';
include_once '../models/ObjMod.php';
include_once '../models/ShedMod.php';
*/
function savetomainAction(){

$resultData['success'] = 1;
$resultData['message'] = 'savetomainAction';

$idOprtn = isset($_REQUEST['idOprtn']) ? $_REQUEST['idOprtn'] :
null;
$idWorker = isset($_REQUEST['idWorker']) ? $_REQUEST['idWorker']
: null;
$num = isset($_REQUEST['num']) ? $_REQUEST['num'] : null;

//d($num);

$_SESSION['user']['idWrkr']=$idWorker;

```

```

if (!count($idOprtn)){
    $resultData['success'] = 0;
    $resultData['message'] = 'Список операций пуст';
}
else{
    $idObj=$_SESSION['user']['idObj'];
    $idMaker=$_SESSION['user']['usr_id'];
    $nowDate=getMyDateTime();
    $idVisitor=getClienNo($idWorker);
    $salePerc1=getEmpsalePerc1ById($idWorker,'salary');
    $salePerc2=getEmpsalePerc1ById($idWorker,'salary_1');

    $op_id=array();

    foreach ($idOprtn as $value){
        $op_id[]=explode("_",$value)[1];
    }

    $i=0;
    foreach ($op_id as $value) {
        $summ=getsummByID($value);
        $summ=$summ*$nums[$i];
        //d($summ);
        appendOperation($idObj,$idWorker,$value,$idVisitor,$summ,$nums[$i],
            $nowDate,$idMaker,$salePerc1,$salePerc2);
        $i++;
    }
}

$resultData['success'] = 1;
$resultData['message'] = 'Все ОК';

```

```

$resultData['idWorker'] = $idWorker;
//d($resultData);
echo json_encode($resultData);
}

function getemployeeidAction() {
    $resultData['success'] = 1;
    $resultData['message'] = 'savetomainAction';
    $resultData['user_id'] = 0;

    $empid = isset($_SESSION['user']['idWrkr']) ?
    $_SESSION['user']['idWrkr'] : null;
    $resultData['user_id'] = $empid;

    echo json_encode($resultData);

}

function reportAction($smarty){
    if(! isset($_SESSION['user']['stat'])){
        redirect('/');
    }

    if($_SESSION['user']['stat'] === "admin"){
        //$admFIO=
getFIOById($_SESSION['user']['usr_id'],$_SESSION['user']['idObj']);

        $idAdmNS=getFIOById($_SESSION['user']['usr_id'],$_SESSION['user']['idObj'])
        ;

        $salePerc1=getEmpsalePerc1ById($_SESSION['user']['usr_id'],'salary');

        $salePerc2=getEmpsalePerc1ById($_SESSION['user']['usr_id'],'salary_1');
        $rsAv= getAv();
    }
}

```

```

$adm_base_salary = getbaseAdmSalary($rsAv,$salePerc1,$salePerc2);

//d($adm_base_salary);

$adminBaseSal=array(
    "idWorker"=>$_SESSION['user']['usr_id'],
    "idOprtn"=>0,
    "idVisitor"=>'0',
    "nowDate"=> getMyDateTime(),
    "custName"=>$idAdmNS[0]['custName'],
    "stat"=>'admin',
    "sirName"=>$idAdmNS[0]['sirName'],
    "prtnmc"=>$idAdmNS[0]['prtnmc'],
    "codeType"=>0,
    "codeName"=>0,
    "codeHall"=>0,
    "hllName"=>",
    "typeName"=>",
    "valName"=>'Основная зарплата администратору',
    "count"=>0,
    "summ"=>$adm_base_salary,
    //"summ"=>500,
    "@summa"=>0,
    "@salary"=>"
);

$obj_name_rus=getObjNameRusById($_SESSION['user']['idObj']);
$curdate= getMyDate();

```

```
// $idAdmNS=getFIOById($_SESSION['user']['usr_id'],$_SESSION['user']['idObj']
);
```

```
// $rsAllWorkToDay=array();
```

```
$rsAllWorkToDay=getAllWorkToDay();
```

```
if($rsAllWorkToDay){
```

```
array_unshift($rsAllWorkToDay,$adminBaseSal);
```

```
}
```

```
//d($rsAllWorkToDay);
```

```
$idVisitor=0;
```

```
$idWrkr=0;
```

```
$sum=0;
```

```
$rsSummaryByDay = array ();
```

```
$rsSalaryByDay = array ();
```

```
if($rsAllWorkToDay){
```

```
    insrtAdmBaseSalary($rsAllWorkToDay);
```

```
    foreach ($rsAllWorkToDay as &$value){
```

```
        if ( $value['idWorker']===$idWrkr){
```

```
            if(!($value['codeType']==20 || $value['codeType']==0)){
```

```
                $sum=$sum+$value['summ'];
```

```
            }else{
```

```
                $pr=getsummByID($value['idOprtn']);
```

```
                if($pr){
```

```
                    $value['summ']= round($pr);
```

```
                }
```

```

    }
}
else {
    $idWrkr=$value['idWorker'];
    $idVisitor=0;
    $sum=0;
    if(!($value['codeType']==20 || $value['codeType']==0)){
        $sum=$value['summ'];
    }else{
        $pr=getsummByID($value['idOprtn']);
        if($pr){
            $value['summ']= round($pr);
            //$value['summ']= round(getsummByID($value['idOprtn']));
        }
    }
}

```

```

$rsSummaryByDay[$value['idWorker']]=getSumAllEmpByID($value['idWorker'],
getMyDate());

```

```

$rsSalaryByDay[$value['idWorker']]=getSalaryEmpByID($value['idWorker'],
getMyDate());

```

```

    }
    if ( $value['idVisitor']===$idVisitor){$value['idVisitor']=' ';}
    else {$idVisitor=$value['idVisitor'];}
    $pars_date=date_parse($value['nowDate']);
    $value['nowDate']=$pars_date['hour'].":".$pars_date['minute'];
    if (strlen($value['codeType'])===1){
        $value['codeType'] = '0'.$value['codeType'];
    }
}

```

```

if (strlen($value['codeName'])===1){
    $value['codeName'] = '0'.$value['codeName'];
}

```

```

if ($value['count']>1){
    $value['valName']= $value['valName']." [".$value['count']."]шт.";
}

```

```

$value['@summa']=$sum;

```

```

//Вычисляем сумму по каждой операции

```

```

$value['@salary']=getSalaryByEmpIdOpId($value['idWorker'],$value['idOprtn'],$
value['count']);
    }
}

```

```

function reportAction($polynt){
if(! isset($_SESSION['client']['stat'])){
    redirect('/');
}
if($_SESSION['client']['stat'] ==="admin"){
    //$admFIO=
getFIOById($_SESSION['client']['idclient'],$_SESSION['client']['idObj']);

$idAdmNS=getFIOById($_SESSION['client']['idclient'],$_SESSION['client']['idO
bj']);

$salePerc1=getEmpsalePerc1ById($_SESSION['client']['idclient'],'paym');

$salePerc2=getEmpsalePerc1ById($_SESSION['client']['idclient'],'paym_1');
$rsAv= getAv();

```

```
$adm_base_paym = getbaseAdmpaym($rsAv,$salePerc1,$salePerc2);
```

```
//d($adm_base_paym);
```

```
$adminBaseSal=array(
    "idWorker"=>$_SESSION['client']['idclient'],
    "idOprtn"=>0,
    "idVisitor"=>'0',
    "nowDate"=> getMyDateTime(),
    "custName"=>$idAdmNS[0]['custName'],
    "stat"=>'admin',
    "sirName"=>$idAdmNS[0]['sirName'],
    "prtnmc"=>$idAdmNS[0]['prtnmc'],
    "codeType"=>0,
    "codeName"=>0,
    "codeHall"=>0,
    "hllName"=>",
    "typeName"=>",
    "valName"=>'Основная зарплата администратору',
    "count"=>0,
    "summ"=>$adm_base_paym,
    //"summ"=>500,
    "@summa"=>0,
    "@paym"=>"
);
```

```
$obj_name_ua=getObjNameuaById($_SESSION['client']['idObj']);
```

```
$curdate= getMyDate();
```

```
// $idAdmNS=getFIOById($_SESSION['client']['idclient'],$_SESSION['client']['id
Obj']);
```

```
// $rsAllWorkToDay=array();
```

```
$rsAllWorkToDay=getAllWorkToDay();
```

```
if($rsAllWorkToDay){
```

```
array_unshift($rsAllWorkToDay,$adminBaseSal);
```

```
}
```

```
//d($rsAllWorkToDay);
```

```
$idVisitor=0;
```

```
$idWrkr=0;
```

```
$sum=0;
```

```
$rsSummaryByDay = array ();
```

```
$rspaymByDay = array ();
```

```
if($rsAllWorkToDay){
```

```
    insrtAdmBasepaym($rsAllWorkToDay);
```

```
    foreach ($rsAllWorkToDay as &$value){
```

```
        if ( $value['idWorker']===$idWrkr){
```

```
            if(!($value['codeType']==20 || $value['codeType']==0)){
```

```
                $sum=$sum+$value['summ'];
```

```
            }else{
```

```
                $pr=getsummByID($value['idOprtn']);
```

```
                if($pr){
```

```
                    $value['summ']= round($pr);
```

```
                }
```

```

    }
}
else {
    $idWrkr=$value['idWorker'];
    $idVisitor=0;
    $sum=0;
    if(!($value['codeType']==20 || $value['codeType']==0)){
        $sum=$value['summ'];
    }else{
        $pr=getsummByID($value['idOprtn']);
        if($pr){
            $value['summ']= round($pr);
            //$value['summ']= round(getsummByID($value['idOprtn']));
        }
    }
}

$sale_to_admn_boss=$all_summary-$all_dm_paym-$admin_paym;

$polynt->assign('obj_name_ua',$obj_name_ua);
$polynt->assign('curdate',$curdate);
$polynt->assign('adm_custName',$idAdmNS[0]['custName']);
$polynt->assign('adm_sirName',$idAdmNS[0]['sirName']);
$polynt->assign('adm_prtnmc',$idAdmNS[0]['prtnmc']);
$polynt->assign('rsAllWorkToDay',$rsAllWorkToDay);
$polynt->assign('rsSummaryByDay',$rsSummaryByDay);
$polynt->assign('rspaymByDay',$rspaymByDay);
$polynt->assign('all_summary',$all_summary);
$polynt->assign('bs_summary',$bs_summary);
$polynt->assign('photo_summary',$photo_summary);
$polynt->assign('xerox_summary',$xerox_summary);
$polynt->assign('additional_summary',$additional_summary);

```

```

$polynt->assign('sale_summary',$sale_summary);
$polynt->assign('all_dm_paym',$all_dm_paym);
$polynt->assign('admin_paym',$admin_paym);
$polynt->assign('sale_to_admn_boss',$sale_to_admn_boss);
$polynt->assign('adm_id',$_SESSION['client']['idclient']);
$rsSummaryByDay[$value['idWorker']]=getSumAllEmpByID($value['id
Worker'], getMyDate());

```

```

$rspaymByDay[$value['idWorker']]=getpaymEmpByID($value['idWorker'],
getMyDate());

```

```

}
if ( $value['idVisitor']===$idVisitor){$value['idVisitor']=' ';}
else {$idVisitor=$value['idVisitor'];}
$pars_date=date_parse($value['nowDate']);
$value['nowDate']=$pars_date['hour'].":".$pars_date['minute'];
if (strlen($value['codeType'])===1){
    $value['codeType'] = '0'.$value['codeType'];
}
if (strlen($value['codeName'])===1){
    $value['codeName'] = '0'.$value['codeName'];
}

if ($value['count']>1){
    $value['valName']= $value['valName']." [".$value['count']."]шт.";
}
$value['@summa']=$sum;
//Вычисляем сумму по каждой операции

```

```

$value['@paym']=getpaymByEmpIdOpId($value['idWorker'],$value['idOprtn'],$value['count']);
    }
}
$all_summary= getAllSummary(getMyDate());
if(!$all_summary){$all_summary=0;}
$bs_summary=getAllBsSummary(getMyDate());
if(!$bs_summary){$bs_summary=0;}
$photo_summary=getAllPhotoSummary(getMyDate());
if(!$photo_summary){$photo_summary=0;}
$xerox_summary=getAllXeroxSummary(getMyDate());
if(!$xerox_summary){$xerox_summary=0;}
$additional_summary= getAllAdditionalSummary(getMyDate());
if(!$additional_summary){$additional_summary=0;}
$sale_summary= getAllSaleSummary(getMyDate());
if(!$sale_summary){$sale_summary=0;}

//d($rsAllWorkToDay);

$all_dm_paym=0;
$admin_paym=0;
$sale_to_admn_boss=0;

$rsSal=getSal($rspaymByDay);

$all_dm_paym=$rsSal['all_dm_paym'];
$admin_paym=getpaymEmpByID($_SESSION['client']['idclient'],
getMyDate());

```

```

foreach ($op_id as $value) {
    $summ=getsummByID($value);
    $summ=$summ*$nums[$i];
    //d($summ);
    appendOperation($idObj,$idWorker,$value,$idVisitor,$summ,$nums[$i],
        $nowDate,$idMaker,$salePerc1,$salePerc2);
    $i++;
}
}
$resultData['success'] = 1;
$resultData['message'] = 'Bce OK';
$resultData['idWorker'] =$idWorker;
//d($resultData);
echo json_encode($resultData);
}

function getemployeeidAction() {
    $resultData['success'] = 1;
    $resultData['message'] = 'savetomainAction';
    $resultData['client_id'] = 0;

    $sempid          =          isset($_SESSION['client']['idWrkr'])          ?
    $_SESSION['client']['idWrkr'] : null;
    $resultData['client_id'] = $sempid;

    echo json_encode($resultData);

}

function reportAction($polynt){
if(! isset($_SESSION['client']['stat'])){
    redirect('/');
}

```

```

    }
    if($_SESSION['client']['stat'] === "admin"){
        //$admFIO=
getFIOById($_SESSION['client']['idclient'], $_SESSION['client']['idObj']);

        $idAdmNS=getFIOById($_SESSION['client']['idclient'], $_SESSION['client']['idObj']);

        $salePerc1=getEmpsalePerc1ById($_SESSION['client']['idclient'], 'paym');

        $salePerc2=getEmpsalePerc1ById($_SESSION['client']['idclient'], 'paym_1');
        $rsAv= getAv();
        $adm_base_paym = getbaseAdmpaym($rsAv, $salePerc1, $salePerc2);
        $sale_to_admn_boss=$all_summary-$all_dm_paym-$admin_paym;

        $polynt->assign('obj_name_ua', $obj_name_ua);
        $polynt->assign('curdate', $curdate);
        $polynt->assign('adm_custName', $idAdmNS[0]['custName']);
        $polynt->assign('adm_sirName', $idAdmNS[0]['sirName']);
        $polynt->assign('adm_prtnmc', $idAdmNS[0]['prtnmc']);
        $polynt->assign('rsAllWorkToDay', $rsAllWorkToDay);
        $polynt->assign('rsSummaryByDay', $rsSummaryByDay);
        $polynt->assign('rspaymByDay', $rspaymByDay);
        $polynt->assign('all_summary', $all_summary);
        $polynt->assign('bs_summary', $bs_summary);
        $polynt->assign('photo_summary', $photo_summary);
        $polynt->assign('xerox_summary', $xerox_summary);
        $polynt->assign('additional_summary', $additional_summary);
        $polynt->assign('sale_summary', $sale_summary);
        $polynt->assign('all_dm_paym', $all_dm_paym);

```

```

$polynt->assign('admin_paym',$admin_paym);
$polynt->assign('sale_to_admn_boss',$sale_to_admn_boss);
$polynt->assign('adm_id',$_SESSION['client']['idclient']);
$polynt->assign('adm_base_paym',$adm_base_paym);

loadTemplate($polynt,'header');
loadTemplate($polynt,'admin/headerline');
loadTemplate($polynt,'admin/leftmenu');
if($rsAllWorkToDay){
loadTemplate($polynt,'admin/report');
}
loadTemplate($polynt,'footer');
}
}
function getSal($rspaymByDay){
    $rsRet = [
        "admin_paym" => 0,
        "all_dm_paym" => 0,
    ];
    $all_dm_paym=0;
    foreach ($rspaymByDay as $key=>$value){
        //d($key);

if($key==$_SESSION['client']['idclient']){$rsRet['admin_paym']=$value;}
        else{$all_dm_paym=$all_dm_paym+$value;}
    }
    $rsRet['all_dm_paym']=$all_dm_paym;
    return $rsRet;
}
function insrtAdmBasepaym($array_ins){

```

```

$adm=getAdm($_SESSION['client']['idclient']);
//d($adm);
array_unshift($array_ins, array (
'idWorker'=> $_SESSION['client']['idclient'],
'idOprtn'=> 1,
'idVisitor'=> 0,
'nowDate'=> getMyDateTime(),
'custName'=>$adm['0']['custName'],
'sirName'=>$adm['0']['sirName'],
'prtnmc'=>$adm['0']['prtnmc'],
'codeType'=>20,
'codeName'=>3,
'codeHall'=>6,
'hllName'=>'Работа',
'typeName'=>",
'valName'=>'lkjlkjlk',
'count'=>1,
'summ'=>300,
'@summa'=>0,
'@paym'=>0
));
}

```

ДОДАТОК Д
Слайди презентації

Міністерство освіти та науки України
Національний університет «Запорізька політехніка»

Програмна реалізація інформаційної системи для барбершопу

Харитонов М.О.
ст. групи КНТ-228
Коцур М.І.

к.т.н., доцент НУ «Запорізька політехніка»

Рисунок Д.1 – Слайд 1

Об'єкт, предмет та мета роботи

Об'єкт дослідження – процес програмної реалізації інформаційної системи для салонного бізнесу.

Предмет дослідження – методи програмної реалізації інформаційної системи для салонного бізнесу.

Метою роботи є програмна реалізація інформаційної системи для салонного бізнесу на прикладі кроссплатформного веб-застосунку для барбершопу.

[2]

Рисунок Д.2 – Слайд 2

Постановка завдання

Проаналізувавши усі найбільш поширені і можливі варіанти реалізації проекту було прийнято рішення про організацію незалежного сайту на окремому хостингу з метою тонкого налаштування його функціоналу для інтерактивного зв'язку із клієнтами.

Функціональні вимоги:

- реєстрація нових користувачів за допомогою форми (логін, пароль);
- авторизація на сайті;
- вилучення визначеного користувача з бази даних системи;
- перегляд контактної інформації салону;
- перегляд додаткової інформації про послуги;
- перегляд вартості послуг;
- робота з інформацією обмеженого доступу;
- редагування сайту адміністратором.

(3)

Рисунок Д.3 – Слайд 3

Порівняння існуючих аналогів

Критерій порівняння	Sborodoy	Frison Barbershop	Firm	Інформаційний веб-ресурс
Можливість онлайн запису до майстра	Не доступно	Можливо	Не доступно	Можливо
Перегляд інформації про майстрів	Відсутній	Відсутній	Відсутній	Можливо
Можливість залишити відгук або ознайомитися із відгуками	Можливо	Відсутня	Відсутня	Можливо
Перегляд каталогу засобів для догляду	Відсутній	Відсутній	Відсутній	Можливо

(4)

Рисунок Д.4 – Слайд 4

Порівняння мов програмування

Критерій	Python	JavaScript	PHP
Відкритість	++	++	++
Кроссплатформність в рамках серверу	+	++	++
Стабільність та надійність застосунків	+	±	++
Програмні надбудови для розробки	+	+	++
Підтримка та масштабування застосунків	+	±	++

(5)

Рисунок Д.5 – Слайд 5

Порівняння CMS

	WordPress	Drupal	Joomla!
Початок роботи	Простий та швидкий	Простий та швидкий	Простий та швидкий
Гнучкість налаштування функціоналу	Посередня	Посередня	Висока
Навантаження на хостинг	Високе	Низьке	Низьке
Використання розширень	Велика кількість, недостатня захищеність	Велика кількість, недостатня захищеність	Велика кількість, висока захищеність

(6)

Рисунок Д.6 – Слайд 6

Загальна схема системи за обраною темою

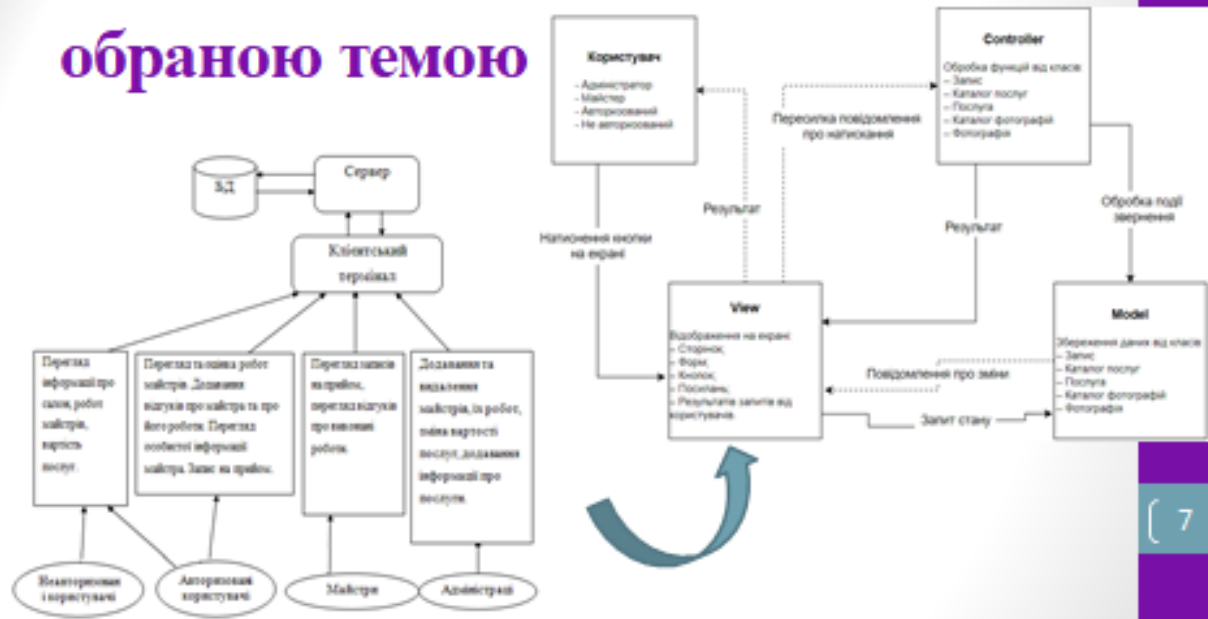


Рисунок Д.7 – Слайд 7



Архітектура веб-застосунку

JOOMLA MODEL VIEW CONTROLLER (MVC)

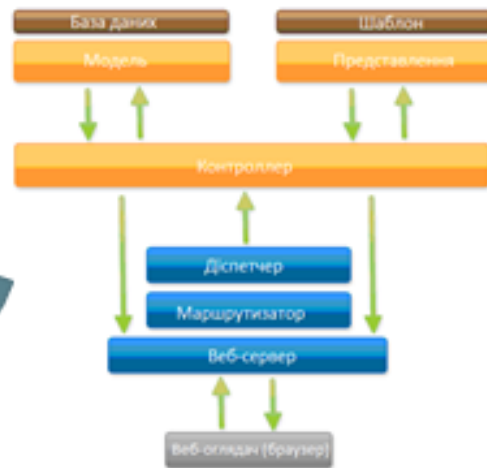


Рисунок Д.8 – Слайд 8

Діаграма класів системи



Рисунок Д.9 – Слайд 9

Логіка взаємодії

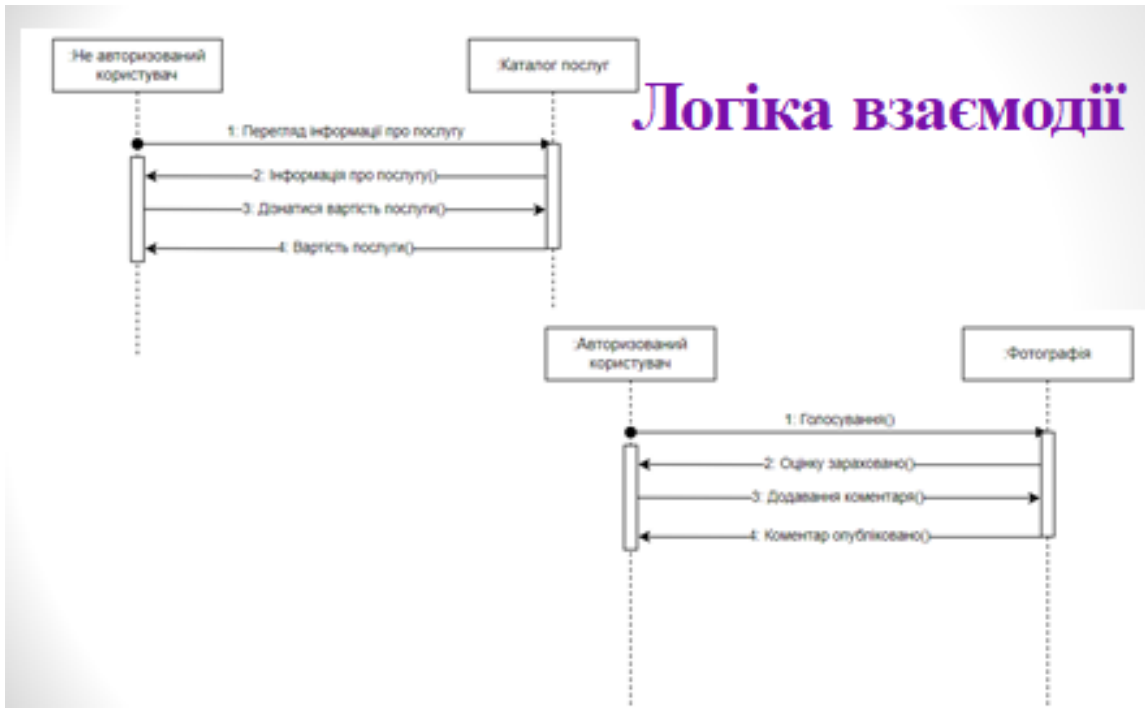


Рисунок Д.10 – Слайд 10

Приклади інтерфейсу застосунку

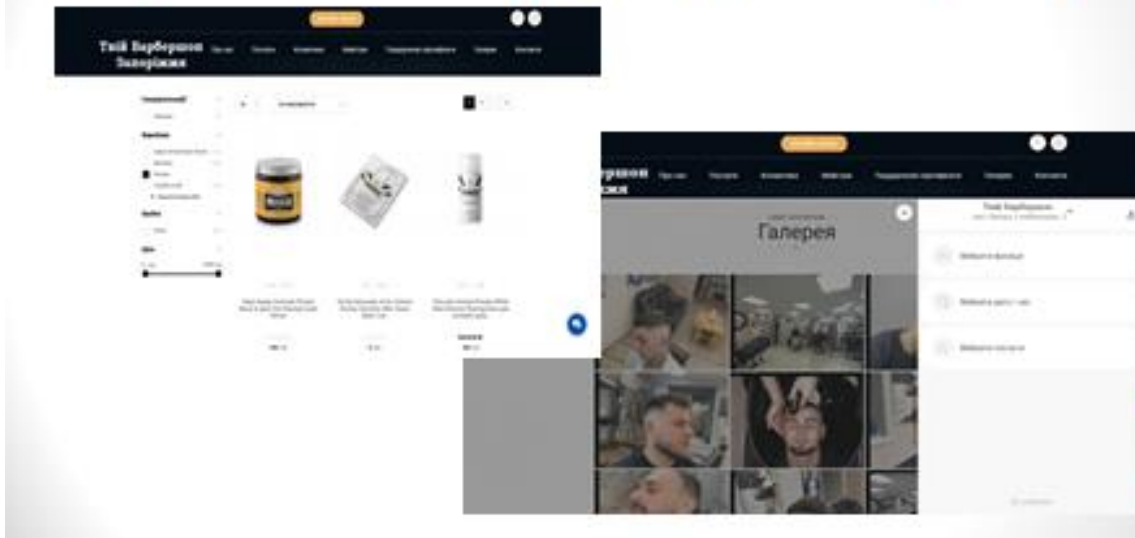


Рисунок Д.11 – Слайд 11

Тестування застосунку



Рисунок Д.12 – Слайд 12

Висновки

У дипломній кваліфікаційній роботі бакалавра виконано програмну реалізацію інформаційної системи для барбершопу у вигляді кроссплатформного веб-застосунку. Для досягнення головної мети роботи було виконано ряд задач. Так на початку було виконано аналіз та документування вимог до ПЗ. Визначено функціональність системи для розроблення технічного завдання. Визначили характеристику області застосування ПЗ, підстави розроблення системи та призначення розробки. Сформулювали технічне завдання на проект системи за обраною темою.

В результаті виконання завдання було отримано систему, що відповідає всім вимогам представленим в технічному завданні, зручна для використання та виконує всі необхідні функції.

Усі завдання випускної дипломної роботи бакалавра повністю виконано.

[13]

Рисунок Д.13 – Слайд 13