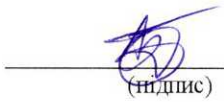
МІНІСТЕРСТВО ФІНАНСІВ УКРАЇНИ
ДЕРЖАВНИЙ ПОДАТКОВИЙ УНІВЕРСИТЕТ

ФАКУЛЬТЕТ ФІНАНСІВ ТА ЦИФРОВИХ ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНИХ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ І СИСТЕМ

КВАЛІФІКАЦІЙНА РОБОТА

на тему: РОЗРОБКА ВЕБСАЙТУ ТА ЧАТ-БОТУ ДЛЯ БРОНЮВАННЯ СТОЛИКІВ У РЕСТОРАНІ

Здобувача вищої освіти першого (бакалаврського) рівня,
IV курсу, групи КБ-21-1, спеціальності 122 «Комп'ютерні науки»,
освітня програма «Інформаційні управляючі системи і технології (за галузями)»


(підпис)

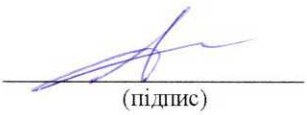
Азарянського Михайла Вікторовича
(прізвище, ім'я, по батькові)

Науковий керівник


(підпис)

Грищенко Світлана Миколаївна, к.пед.н., ст. досл.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

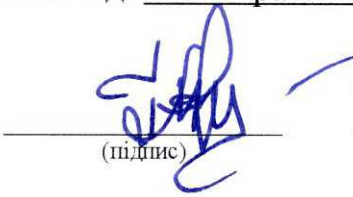
Керівник освітньо-професійної програми


(підпис)

Омельчук Антон Анатолійович, к.т.н.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Допущено до захисту за рішенням кафедри комп'ютерних та інформаційних технологій і систем протокол від «13» травня 2025 № 18

Завідувач кафедри


(підпис)

Філоненко Михайло Миколайович, к.ф.-м.н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Ірпінь – 2025

Реєстрація № 43 23.05.2025 _____
(дата) (підпис відповідальної особи)

Результати перевірки до захисту 23.05.2025 _____
(до захисту, на доопрацювання) (дата) (підпис)

Результат захисту 4 (добре) 04.06.2025 _____
(оцінка) (дата) (підпис)

Голова комісії: _____ В.М. Франчук
(підпис)

Члени комісії:

_____ М.М. Філоненко
(підпис)

_____ С.М. Грищенко
(підпис)

_____ А.А. Омельчук
(підпис)

БЛАНК ОЦІНЮВАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ

Об'єкти контролю	Діапазон оцінки в балах	Фактична кількість балів	Примітки	Дата, П.І.Б.: керівника, голови комісії членів комісії,секретаря, (підписи)
ОЦІНЮВАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ КОМІСІЄЮ ПО ЗАХИСТУ				
1. Зміст випускної роботи (якість теоретичного матеріалу, його зв'язок з практичним, вміння представляти матеріал в логічній послідовності, наявність висновків та пропозицій)	**від 0 до 30**	27		**Грищенко С.М.** Франчук В.М. Філоненко М.М. Грищенко С.М. Омельчук А.А. Суховецька Л.П.
2. Дотримання вимог щодо оформлення	**від 0 до 10**	9		
3. Вчасне виконання роботи на всіх етапах у встановлені кафедрою терміни	**від 0 до 5**	4		
4. Рецензент	**від 0 до 5**	4		
РАЗОМ БАЛІВ	**від 0 до 50**	44		

Оцінювання кваліфікаційної роботи науковим керівником - 40 балів.

ПЕРЕВЕДЕННЯ ОЦІНКИ ЗА 100-БАЛЬНОЮ ШКАЛОЮ ДО НАЦІОНАЛЬНОЇ ШКАЛИ ТА ШКАЛИ ЄКТС

Оцінка за 100-бальною шкалою	Оцінка за шкалоюЄКТС	Оцінка за національною шкалою	Дата	П.І.Б.: голови, членів комісії, секретаря, (підпис)
84	B	4 (добре)	04.06.2025	Франчук В.М. Філоненко М.М. Грищенко С.М. Омельчук А.А. Суховецька Л.П.

ПРАВИЛА ПЕРЕВЕДЕННЯ ОЦІНОК ДО НАЦІОНАЛЬНОЇ ШКАЛИ ТА ШКАЛИ ЄКТС

Оцінка за 100-бальною шкалою	Оцінка за шкалою ЄКТС	Оцінка за національною шкалою
90-100	**A**	**5 (відмінно)**
80-89	**B**	**4 (добре)**
70-79	**C**	
60-69	**D**	**3 (задовільно)**
50-59	**E**	
35-49	**FX**	**2 (незадовільно)**

Факультет фінансів та цифрових технологій
Кафедра комп'ютерних та інформаційних технологій і систем
Галузь знань: 12 «Інформаційні технології»
Спеціальність: 122 «Комп'ютерні науки»
Освітня програма: «Інформаційні управляючі системи і технології (за галузями)»

Затверджую
Завідувач кафедри _____ Філоненко М.М.
«14» листопада 2024

Завдання на кваліфікаційну роботу
здобувачу вищої освіти
Азарянському Михайлу Вікторовичу
(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи Розробка вебсайту та чат-боту для бронювання столиків у ресторані
Затверджена наказом ректора від «13» листопада 2024 № 1717
2. Строк здачі закінченої роботи «08» травня 2025
3. Мета роботи розробка веб-сайту та чат-боту для покращення процесів бронювання столиків у ресторані для користувачів та оптимізація процесів для бізнесу
Об'єкт дослідження Вебдизайн та веброзробка
Предмет дослідження Процес створення вебсайту та чат-боту
Орієнтовна база дослідження вебсайт та чат-бот для бронювання
4. Вихідні дані до роботи методи та підходи до розробки та створений вебсайт
5. Перелік графічного матеріалу головна сторінка вебсайту ресторану, сторінка меню, сторінка інформації про заклад, сторінка бронювання, чат-бот для бронювання
6. Зміст кваліфікаційної роботи (перелік питань за кожним розділом)

ЗМІСТ
ВСТУП
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.
1.1. Аналіз сучасних систем бронювання, тенденції в розробці вебсайтів та чат-ботів для ресторанів.
1.2 Основи веброзробки, принцип роботи чат-ботів, створення зручного та інтуїтивно зрозумілого інтерфейсу.
РОЗДІЛ 2. ТЕХНОЛОГІЯ ПРОЄКТУВАННЯ ВЕБСАЙТУ ТА ЧАТ-БОТУ.
2.1. Розробка чат-боту, створення діалогів, інтеграція з сайтом.
2.2 Структура та розробка вебсайту для бронювання столиків у ресторані.
РОЗДІЛ 3. АНАЛІЗ ТА ОЦІНКА РЕЗУЛЬТАТІВ
ВИСНОВКИ
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

7. Календарний план виконання роботи

№ п/п	Назва етапів кваліфікаційноїроботи	Строк виконання етапів роботи	
		За планом	Фактично
1	2	3	4
1	Вибір теми, підбір літературних джерел, формування структури кваліфікаційної роботи	До 30.10.2024	До 30.10.2024
2	**І розділ** (розкрити сутність і стан дослідження наукової проблематики роботи, після чого навести її загальну характеристику; огляд літератури за темою і вибір конкретних напрямів досліджень)	27.12.2024	27.12.2024
3	**ІІ розділ** (аналіз експериментальної частини про методики досліджень)	17.02.2025	17.02.2025
4	**ІІІ розділ** (проведені теоретичні і (або) експериментальні дослідження та проектні рішення)	16.04.2025	16.04.2025
5	**Висновки**	01.05.2025	01.05.2025
6	Доопрацювання роботи з урахуванням зауважень наукового керівника	09.05.2025	09.05.2025
7	Отримання відгуку керівника	27.05.2025	27.05.2025
8	Отримання зовнішньої рецензії	29.05.2025	29.05.2025

Дата видачі завдання «14» листопада 2024

Керівник кваліфікаційної роботи

_____ Грищенко С.М.
(підпис) (прізвище, ініціали)

Керівник освітньо-професійної програми

_____ Омельчук А.А.
(підпис) (прізвище, ініціали)

Завдання до виконання отримав здобувач вищої освіти

_____ Азарянський М.В
(підпис) (прізвище, ініціали)

АНОТАЦІЯ

____Розробка вебсайту та чат-боту для бронювання столиків у ресторані___

Мета роботи полягає в розробці вебсайту для покращення процесів бронювання столиків у ресторані для користувачів та оптимізація процесів для бізнесу.

Завданнями роботи є:

- Проведення аналізу предметної області
- Розробку естетичного та привабливого дизайну
- Створення функціонального вебсайту
- Створення функціонального чат-боту
- Впровадження адаптивного дизайну
- Оптимізація продуктивності та швидкодії

Предметом дослідження є розробка вебсайту та чат-боту.

Об'єктом дослідження є вебдизайн та веброзробка.

Базою дослідження виступає вебсайт та чат-бот для бронювання.

За результатами дослідження сформульовані методика та етапи реалізації вебсайту та чат-боту.

Одержані результати можуть бути використані при оптимізації бізнес процесів ресторану та залученню нових клієнтів завдяки використанню вебсайту та чат-боту.

Кваліфікаційна робота містить 59 сторінок, 10 рисунків, список використаних джерел 44 найменувань, 2 додатки.

Ключові слова: ресторан, бронювання, столик, онлайн, доступність, зручність, інформація, меню.

Рік виконання кваліфікаційної роботи 2025

Рік захисту роботи 2025

ABSTRACT

_____Website and chatbot development for restaurant table reservations_____

The aim of this work is to develop a website to improve restaurant reservation processes for users and optimize processes for businesses.

The tasks of the work are as follows:

- Conducting analysis of the subject area.
- Development of aesthetic and attractive design.
- Creation of a functional website.
- Creation of a functional chatbot.
- Implementation of responsive design.
- Optimization of performance and speed.

The subject of the research is the development of a website and a chatbot.

The object of the research is web design and web development.

The research is based on a website and a booking chatbot.

The results of the research include the methodology and stages of website implementation and chatbot were formulated.

The obtained results can be used to optimize the restaurant`s business processes and attract new customers through the use of a website and chatbot.

The qualification work consists of 59 pages, 10 images, a list of 44 references, and 2 appendices.

Keywords: restaurant, reservation, table, online, availability, convenience, information, menu.

Year of qualification work: 2025

Year of work defense: 2025

ЗМІСТ

ВСТУП..... 3
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ..... 6
1.1 Аналіз сучасних систем бронювання, тенденції в розробці вебсайтів та чат-ботів для ресторанів. 6
1.2. Основи веброзробки, принцип роботи чат-ботів, створення зручного та інтуїтивно зрозумілого інтерфейсу. 11
РОЗДІЛ 2. ТЕХНОЛОГІЯ ПРОЄКТУВАННЯ ВЕБСАЙТУ ТА ЧАТ-БОТУ 18
2.1 Розробка чат-бота, створення діалогів, інтеграція з сайтом..... 18
2.2 Структура та розробка вебсайту для бронювання столиків у ресторані.....
..... 24
РОЗДІЛ 3. АНАЛІЗ ТА ОЦІНКА РЕЗУЛЬТАТІВ..... 38
ВИСНОВКИ..... 43
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ..... 45
ДОДАТКИ..... 50

ВСТУП

Сучасний світ активно переходить до цифрових технологій, що значно спрощують взаємодію між компаніями та клієнтами. В ресторанному бізнесі автоматизація процесів є важливим фактором підвищення рівня обслуговування, оптимізації роботи персоналу та збільшення прибутків. Одним із ключових напрямів цифрової трансформації у цій сфері є впровадження вебсайтів та чат-ботів для бронювання столиків. Це дозволяє досить суттєво зменшити навантаження на адміністраторів, прискорити процес бронювання та покращити клієнтський досвід.

Актуальність даної теми дослідження зумовлена необхідністю автоматизації процесу бронювання у ресторанах. Традиційні методи бронювання, такі як телефонні дзвінки або особисте відвідування закладу, мають низку недоліків, зокрема тривалість обробки запиту та можливість людських помилок. Розробка вебсайту та чат-боту для бронювання столиків дозволяє мінімізувати ці проблеми, забезпечує зручність для клієнтів і підвищує ефективність роботи ресторану[1].

Проблематика розробки вебсайту та чат-боту для бронювання столиків у ресторані активно досліджується як українськими, так і зарубіжними науковцями. Серед українських авторів варто відзначити К.М. Чевгуза, який розглядає створення веб-орієнтованої системи бронювання, а також Тетяну Лисюк і Валентину Постову, що аналізують впровадження інноваційних технологій у ресторанному бізнесі. Зарубіжні дослідники, такі як Matthew Henderson і Xinshang Wang, працюють над створенням діалогових систем і онлайн-розкладів для автоматизації процесу вибору та бронювання закладів. Їхні напрацювання формують теоретичну й практичну базу для розробки ефективних ІТ-рішень у сфері ресторанного сервісу.

Метою кваліфікаційної роботи є розробка вебсайту та чат-боту для автоматизованого бронювання столиків у ресторані.

Для досягнення цієї мети необхідно виконати такі *завдання*:

- Проведення аналізу існуючих рішень у сфері бронювання столиків у ресторанах;

- Визначення вимог до вебсайту та чат-боту з урахуванням потреб клієнтів та ресторанного бізнесу;
- Розробка вебсайту з інтерактивним інтерфейсом для зручного бронювання столиків;
- Створення чат-боту для спрощення процесу бронювання через месенджер;
- Тестування розробленої системи та оцінка її ефективності.

Об'єктом кваліфікаційної роботи є процес бронювання столиків у ресторанах.

Предмет дослідження – методи та інструменти автоматизації цього процесу за допомогою вебтехнологій та чат-ботів.

Методи дослідження – аналіз існуючих програмних рішень, проєктування програмних систем, розробки веб-додатків та штучного інтелекту для створення чат-ботів. Також проводиться тестування розробленої системи для оцінки її працездатності та ефективності[2].

Базою дослідження виступає вебсайт та чат-бот ресторану Смакота.

Наукова-практична значимість одержаних результатів. Сприяє покращенню користувацького досвіду, ефективному бронюванню столиків, підвищенню конкурентоспроможності ресторану, оптимізації його роботи та забезпеченню доступної та актуальної інформації для клієнтів. Ці результати дозволяють ресторану залучати більше клієнтів, підвищувати ефективність управління та забезпечувати успішну роботу в конкурентному середовищі[3].

Практичне значення роботи полягає у тому, що розроблений у межах даної роботи вебсайт та чат-бот можуть бути використані в реальних ресторанах для автоматизації процесу бронювання. Це дозволить зменшити навантаження на персонал, підвищити якість обслуговування та зробити процес резервування столиків зручнішим та швидшим. Крім того, результати роботи можуть бути використані для подальшого вдосконалення систем автоматизації у ресторанному бізнесі[4].

Таким чином, дана кваліфікаційна робота спрямована на вирішення актуальної проблеми автоматизації ресторанного бізнесу шляхом створення сучасного цифрового рішення у вигляді вебсайту та чат-боту для бронювання столиків.

РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Аналіз сучасних систем бронювання, тенденції в розробці вебсайтів та чат-ботів для ресторанів.

Системи онлайн-бронювання займають важливе місце в розвитку ресторанного бізнесу, оскільки надають зручні інструменти для взаємодії з клієнтами та полегшують процес бронювання столиків у реальному часі. Оскільки існує багато платформ, що дозволяють організувати онлайн-бронювання, важливо детально розглянути найбільш популярні з них, зокрема *Booking.com* та *TripAdvisor*, щоб оцінити їх функціональність, переваги та недоліки, а також зрозуміти, які можливості вони надають ресторанам для інтеграції в свої вебсайти.

Booking.com

Booking.com відома здебільшого як платформа для бронювання готелів і житла, однак останніми роками вона активно розширює свої послуги і в ресторанну сферу. Система бронювання столиків, запропонована Booking.com, дозволяє ресторанам збільшити видимість та покращити обслуговування клієнтів. Платформа інтегрує онлайн-бронювання з процесом бронювання готельних номерів, що робить її зручним рішенням для туристів, які хочуть одночасно забронювати і місце для відпочинку, і столик у ресторані[5].

Основні функції Booking.com для ресторанного бронювання:

- *Інтеграція з іншими сервісами Booking:* ресторан може бути представлений разом з готелем або іншими туристичними об'єктами на загальному порталі. Це дає додаткову аудиторію, що може бути зацікавлена в бронюванні столиків.
- *Пошук за локацією та типом кухні:* користувачі можуть фільтрувати ресторани за різними параметрами, такими як місце розташування або тип кухні (італійська, японська, мексиканська тощо).
- *Рейтинг та відгуки:* клієнти можуть оцінити ресторан після відвідування, що додає достовірності та допомагає іншим користувачам прийняти рішення.

- *Підтвердження бронювання:* система автоматично підтверджує бронювання, що дає користувачеві впевненість в тому, що його замовлення було успішно оформлене.

Недоліки Booking.com:

- *Витрати на комісію:* ресторан змушений платити комісію за кожне бронювання, що може становити значну частину доходу, особливо для малих підприємств.
- *Відсутність контролю над брендом:* ресторан не має можливості на власний розсуд налаштовувати дизайн і функціональність своєї сторінки на платформі, що обмежує можливості персоналізації.
- *Обмеження доступу до клієнтів:* ресторан не має прямого контакту з клієнтом, оскільки Booking.com виступає як посередник, що знижує рівень взаємодії з кінцевим користувачем.

TripAdvisor

TripAdvisor є однією з найбільших платформ для обміну відгуками про туристичні послуги, включаючи ресторани, готелі та атракції. Хоча основна мета *TripAdvisor* — це допомога туристам у виборі кращих місць для відпочинку, платформа також надає можливість бронювати столи в ресторанах через своїх партнерів, зокрема через систему *OpenTable.*

Основні функції TripAdvisor для ресторанного бронювання:

- *Пошук за відгуками та рейтингами:* користувачі можуть легко знайти ресторани, що мають високі відгуки, що є важливим фактором для прийняття рішення про бронювання.
- *Інтеграція з OpenTable:* для тих ресторанів, які використовують OpenTable, TripAdvisor дозволяє здійснити бронювання прямо через платформу, що дозволяє зберегти час і мінімізувати помилки.
- *Додаткові функції для ресторану:* TripAdvisor надає ресторанам аналітику, що дозволяє відслідковувати кількість бронювань, часи найвищого попиту та загальну ефективність їхнього профілю.

Недоліки TripAdvisor:

- *Залежність від зовнішніх платформ:* ресторани, що інтегруються з OpenTable, не мають повного контролю над процесом бронювання і користуються умовами цієї платформи.

- *Негативні відгуки:* оскільки система рейтингу TripAdvisor дозволяє будь-кому залишати відгуки, навіть якщо вони не є клієнтами, це може призвести до ситуацій, коли ресторан отримує несправедливі негативні відгуки.

- *Комісія за бронювання:* як і в Booking.com, ресторани повинні сплачувати комісії за використання платформи для бронювання столиків.

Інші популярні платформи для онлайн-бронювання

Крім *Booking.com* і *TripAdvisor*, існують спеціалізовані платформи для бронювання столиків у ресторанах, які надають більше контролю рестораторам і пропонують зручніші інструменти для автоматизації бронювань.

OpenTable — одна з найбільш відомих платформ для бронювання столиків у ресторанах, що пропонує: інтуїтивно зрозумілий інтерфейс для клієнтів, повний контроль над дизайном та функціональністю вебсайту ресторану, аналітичні інструменти для відстеження популярності ресторанів і моніторингу запитів на бронювання.

Resy — платформа для бронювання столиків, яка пропонує інструменти для управління резервуваннями, дозволяє інтегрувати функції онлайн-оплати, а також має гнучку систему лояльності для постійних клієнтів.

Bookatable — система для бронювання столиків, яка забезпечує інтеграцію з онлайн-платформами та додатками, а також пропонує можливість автоматично підтверджувати бронювання через SMS та електронну пошту.

Переваги та недоліки існуючих рішень

Переваги:

- *Широке охоплення аудиторії:* платформи, такі як Booking.com і TripAdvisor, забезпечують доступ до великої кількості потенційних клієнтів.

- *Автоматизація процесу бронювання:* ресторани можуть легко відслідковувати й підтверджувати бронювання без необхідності ручного втручання.

- *Залучення нових клієнтів:* інтеграція з платформами дозволяє ресторану отримати більшу видимість у порівнянні з традиційними методами реклами.
- *Інструменти для аналізу:* використання таких платформ дозволяє ресторанам збирати дані про поведінку користувачів, що дає можливість для покращення стратегії маркетингу та оптимізації процесів.

Недоліки:

- *Комісії:* необхідність сплачувати значні комісії за кожне бронювання, що може знизити загальний прибуток ресторану.
- *Обмежений контроль над брендом:* ресторани не можуть повністю контролювати свій вигляд та функціональність на сторонніх платформах.
- *Невизначеність у взаємодії з клієнтами:* через наявність посередника ресторан може втратити безпосередній контакт із клієнтом, що важливо для створення персоналізованого досвіду.

Основні тенденції в розробці вебсайтів та чат-ботів для ресторанів у 2025 році.

Мобільна оптимізація та адаптивний дизайн

У більшості ресторанів клієнти використовують мобільні пристрої для перегляду меню, бронювання столиків або замовлення їжі. Тому важливо, щоб вебсайт був адаптований під всі типи екранів та мав інтуїтивно зрозумілий інтерфейс.

Основні аспекти мобільної оптимізації: Швидкість завантаження (мобільний інтернет часто може бути нестабільним, тому важливо оптимізувати вебсайти для швидкого завантаження), *Інтуїтивно зрозумілий інтерфейс* (логічна навігація, великі кнопки для мобільних пристроїв, зручні форми для заповнення, зокрема при оформленні замовлень чи бронюванні столика), *Адаптивний дизайн* (вебсайт має бути зручним і для смартфонів, і для планшетів, а також працювати на різних браузерах)[8].

Інтеграція чат-ботів

Чат-боти стали важливим елементом у взаємодії з клієнтами. Вони допомагають автоматизувати комунікацію, знижують навантаження на персонал і покращують досвід користувачів.

*Тренди в чат-ботах для ресторанів: Обробка замовлень (*чат-боти можуть приймати замовлення на доставку, бронювання столиків або відповідати на запити щодо наявності страв в меню), *Мовна підтримка (*боти, які підтримують кілька мов, що дозволяє ресторанам обслуговувати міжнародних клієнтів), *Підтримка штучного інтелекту* (чат-боти стають розумнішими завдяки використанню машинного навчання і можуть навчатися на основі попередніх взаємодій з клієнтами), *Персоналізація* (завдяки AI, чат-боти можуть запам'ятовувати уподобання клієнта і пропонувати персоналізовані рекомендації або нагадування про улюблені страви)[7].

Інтерактивне меню з можливістю замовлення

Вебсайти ресторанних закладів активно інтегрують інтерактивні меню, що дозволяють клієнтам не лише переглядати страви, але й швидко робити замовлення онлайн.

Особливості інтерактивних меню: Візуалізація страв (високоякісні фотографії страв, відео або навіть 3D-огляди, що дозволяють краще уявити продукт), *Фільтри* (можливість вибору страв за категоріями), *Динамічні оновлення* (автоматичне оновлення меню в реальному часі, що допомагає уникнути непорозумінь при замовленні недоступних страв).

Розширена реальність (AR) та віртуальні тури

Ресторанний бізнес також активно використовує технології доповненої реальності для підвищення інтересу клієнтів. Наприклад, за допомогою AR можна запропонувати віртуальний тур по ресторану або навіть демонстрацію страв у вигляді 3D моделей, щоб клієнти могли побачити, як виглядатиме їх замовлення.

До основних можливостей AR відносяться: перегляд страв у 3D, віртуальні тури по ресторану.

Інтеграція з соціальними мережами та платіжними системами

Сучасні вебсайти ресторанів повинні мати можливість інтеграції з популярними соціальними мережами (*Facebook, Instagram, Twitter*) та платіжними системами (*PayPal, Apple Pay, Google Pay*)[6].

Серед різних інтеграцій можна виділити: швидку оплату (зручні варіанти оплати, які не потребують від клієнтів додаткових кроків), спільні акції через соціальні мережі (спільні кампанії на *Facebook* або *Instagram* дозволяють ресторанам залучати нових клієнтів через публікації та спеціальні пропозиції), рейтинг і відгуки (можливість залишити відгуки на платформах, таких як *TripAdvisor* або *Google Reviews*, безпосередньо через сайт ресторану).

Інтелектуальні рекомендаційні системи

Завдяки використанню великих даних і алгоритмів машинного навчання ресторани можуть пропонувати клієнтам персоналізовані рекомендації на основі їхніх попередніх замовлень чи відгуків. Такі системи не лише підвищують клієнтську лояльність, а й збільшують середній чек.

Переваги для клієнтів: пропозиція страв на основі смакових уподобань, рекомендації страв для спеціальних випадків, таких як дієта або харчові обмеження.

Технології автоматизації та AI

Все більше ресторанів використовують автоматизацію для оптимізації роботи персоналу. Наприклад, AI може допомогти в управлінні запасами, прогнозуванні попиту на певні страви або навіть у створенні персоналізованих маркетингових кампаній.

Автоматизація включає: *системи*, які дозволяють автоматично приймати та обробляти замовлення без участі людей, та *алгоритми*, які допомагають прогнозувати попит на страви залежно від дня тижня, пори року або подій у місті.

1.2. Основи веброзробки, принцип роботи чат-ботів, створення зручного та інтуїтивно зрозумілого інтерфейсу.

Веброзробка є основною складовою частиною цифрової економіки, що забезпечує створення вебсайтів та вебдодатків для різноманітних сфер діяльності.

Веброзробка включає в себе роботу з HTML, CSS, JavaScript, а також різними фреймворками, які полегшують процес створення сучасних інтерактивних сайтів. Окрім того, в останні роки чат-боти стали важливою складовою веброзробки завдяки їх здатності автоматизувати комунікацію між бізнесами та користувачами.

Одним із важливих аспектів створення вебсайтів та вебдодатків є зручний та інтуїтивно зрозумілий інтерфейс. Від якості інтерфейсу залежить не лише ефективність роботи з продуктом, але й загальне враження користувача від сайту чи додатку.

Основи веброзробки

Основи веброзробки є важливим аспектом для кваліфікаційної роботи, оскільки ці технології є основою більшості сучасних веб-сайтів і веб-додатків.

HTML (HyperText Markup Language)

HTML є основною мовою розмітки для створення вебсторінок. Вона визначає структуру веб-сторінки, використовуючи теги для позначення різних елементів, таких як заголовки, параграфи, списки, зображення, посилання та інші.

Основні поняття HTML:

- *Теги:* Наприклад, <h1>, <p>, <a>, <img>, , .
- *Атрибути:* Атрибути додаються до тегів для визначення додаткової інформації, наприклад, атрибут src для зображень або атрибут href для посилань.
- *Семантичні елементи:* Наприклад, <header>, <article>, <section>, <footer> для кращої організації контенту [9].

CSS (Cascading Style Sheets)

CSS використовується для визначення стилів вебсторінки, таких як шрифти, кольори, розташування елементів та інші аспекти дизайну. CSS дозволяє змінювати вигляд HTML-елементів без змін у самій структурі сторінки.

Основні поняття CSS:

- *Селектори:* Визначають елементи на сторінці, до яких застосовуються стилі. Наприклад, p, .class-name, #id-name.
- *Властивості:* Це характеристики стилів, такі як color, font-size, margin, padding, display, position.

- *Медіа-запити:* За допомогою медіа-запитів можна створювати адаптивний дизайн для різних пристроїв (мобільних телефонів, планшетів, комп'ютерів).

JavaScript

JavaScript — це мова програмування, яка додає інтерактивність веб-сторінкам. Вона дозволяє працювати з елементами DOM (Document Object Model), обробляти події (наприклад, натискання кнопок), робити запити до серверів та багато іншого.

Основні поняття JavaScript:

- *Змінні:* let, const, var.
- *Функції:* Як створювати і викликати функції для виконання коду.
- *Події:* Наприклад, обробка кліків на кнопки, прокрутка сторінки, зміна значення в полях вводу.
- *Асинхронність:* Використання async та await, обробка промісів для асинхронних запитів [10].

Фреймворки для веб-розробки

Використання фреймворків дозволяє прискорити процес розробки та організувати код у більш зрозумілий і масштабований спосіб. Серед популярних фреймворків для веброзробки варто виділити: *React, Angular.*

React

React — це бібліотека JavaScript для створення інтерфейсів користувача. Вона дозволяє будувати компоненти, які можуть оновлюватися незалежно одна від одної, що робить додатки швидкими та інтерактивними [12].

Основні принципи React:

- *Компоненти:* Всі елементи інтерфейсу (кнопки, форми, таблиці) створюються як компоненти.
- *JSX:* Спеціальний синтаксис, що дозволяє використовувати HTML-подібний код в JavaScript.
- *Стан (State):* Управління даними компонента, що дозволяє динамічно змінювати вигляд сторінки.
- *Props:* Передача даних між компонентами.

Angular

Angular — це фреймворк для створення веб-додатків, що використовує TypeScript і надає потужний набір інструментів для розробки складних додатків [11].

Основні принципи Angular:

- *Модулі:* Організація додатку в модулі для кращої структури.
- *Директиви:* Спеціальні теги або атрибути для маніпулювання елементами DOM.
- *Сервіси:* Використовуються для логіки додатку, таких як запити до серверів.
- *Dependency Injection:* Механізм для управління залежностями між компонентами та сервісами.

Основи взаємодії між технологіями

У сучасній веброзробці часто використовуються різні технології, щоб створити інтерактивні вебдодатки: *HTML (*Структурує контент), *CSS* (Додає стилі до контенту), *JavaScript, React/Angular (*Додає функціональність і взаємодію з користувачем).

Принцип роботи чат-боту

Принцип роботи *чат-ботів* є важливою темою для кваліфікаційної роботи, оскільки чат-боти стають все більш популярними у різних сферах бізнесу та комунікації. *Чат-боти* використовуються для автоматизації процесів, взаємодії з клієнтами, надання послуг та допомоги. Розглянемо основні принципи роботи *чат-ботів*, а також технології, які використовуються для їх створення.

Чат-бот — це програмне забезпечення, яке імітує спілкування з людиною через текстовий або голосовий інтерфейс. Чат-боти використовуються на вебсайтах, в мобільних додатках, а також у месенджерах (WhatsApp, Facebook Messenger, Telegram) для автоматичної взаємодії з користувачами.

Основні принципи роботи чат-бота

Чат-боти можуть бути різними за складністю і функціоналом, але їх загальна структура та принципи роботи можна поділити на кілька ключових етапів. Чат-боти використовують текстовий або голосовий інтерфейс для взаємодії з користувачами.

Це може бути: *Текстовий чат* (наприклад, чат-боти у месенджерах, веб-сайтах), *Голосовий чат* (наприклад, віртуальні асистенти). Користувач вводить запит чи повідомлення в чат-бот, після чого бот обробляє інформацію і генерує відповідь. Після того, як користувач надіслав повідомлення, чат-бот здійснює обробку запиту. Це може відбуватися за допомогою кількох основних підходів:

1) *Статичні відповіді (правила):* Простий чат-бот працює за алгоритмами на основі набору правил. Якщо користувач вводить певну фразу, бот відповідає відповідно до заданого шаблону або переліку відповідей. Наприклад, якщо користувач питає "Яка погода?", бот відправляє стандартну відповідь: "Погода сьогодні сонячна".

2) *Машинне навчання (ML) та природна мова (NLP):* Більш складні чат-боти використовують технології машинного навчання (ML) і обробки природної мови (NLP) для розуміння запитів. Вони можуть аналізувати контекст повідомлення, визначати намір користувача та генерувати відповідь, враховуючи більше факторів[13].

Для того, щоб *чат-бот* міг точно зрозуміти, що запитує користувач, він використовує технології *обробки природної мови (NLP)***.** Це дозволяє чат-боту виконувати функції *розпізнавання наміру користувача:* (наприклад, чат-бот може зрозуміти, що користувач хоче дізнатися погоду або зробити замовлення, навіть якщо питання поставлене по-різному), *аналіз тексту та виділення сутностей (* наприклад, якщо користувач запитує: "Яка погода в Києві?", бот повинен розпізнати "Київ" як місто і "погода" як сутність для обробки).

Після аналізу запиту чат-бот формує відповідь. Це може бути: *статична відповідь (*наприклад, текстова відповідь, яка відповідає на запит користувача), *динамічна відповідь (* це може бути інтерактивний елемент, як-от кнопки для вибору, карусель з продуктами, або навіть карти та відео), *інтеграція з іншими системами (* бот може надсилати запити до сторонніх API для отримання актуальної інформації, наприклад, погоди, новин або здійснення покупок).

Чат-бот може підтримувати діалог з користувачем, задаючи уточнювальні питання або пропонуючи подальші дії. Також чат-бот може бути налаштований для

збирання зворотного зв'язку про свою роботу, щоб покращити взаємодію з користувачами.

Залежно від складності, чат-боти поділяються на:

- *Правилозалежні (на основі шаблонів):* Ці боти реагують на заздалегідь визначені фрази і команди. Вони простіші у створенні, але мають обмежену гнучкість.
- *Розумні чат-боти (на основі AI):* Використовують штучний інтелект, машинне навчання та обробку природної мови для більш точного аналізу запитів і надання відповідей.

Для розробки чат-ботів використовуються різноманітні технології та інструменти: *мови програмування (*Python, JavaScript, Java, PHP), *бібліотеки та фреймворки (*TensorFlow, PyTorch, Rasa, Dialogflow, Botpress), *платформи для чат-ботів (*ManyChat, Chatfuel, Microsoft Bot Framework, Facebook Messenger API).

Чат-боти застосовуються в різних *сферах*:

- *Клієнтська підтримка:* Автоматичні відповіді на часто задавані питання, обробка запитів клієнтів, допомога у вирішенні проблем.
- *Інтернет-магазини:* Персоналізовані рекомендації, оформлення замовлень, відповіді на запити щодо продуктів.
- *Розваги та медіа:* Чат-боти для надання новин, інтерактивних ігор, голосових або текстових розваг.
- *Освіта:* Допомога в навчанні, автоматизація викладання, навчальні курси та тести.

Серед основних *перспектив розвитку чат-ботів* можна виділити: *більш глибоку інтеграцію з AI* (покращення розуміння запитів користувачів, покращення генерації відповідей та створення чат-ботів, які можуть вести глибокі і змістовні діалоги), *мультимодальні чат-боти* (взаємодія не лише через текст, але й через зображення, відео, голосові команди), *чат-боти з покращеними емоційними навичками* (вміння розпізнавати емоції користувача та відповідно адаптувати стиль спілкування).

Створення зручного та інтуїтивно зрозумілого інтерфейсу (UX/UI) — це ключ до того, щоб користувачі із задоволенням працювали з вашим продуктом. Серед *основних принципів* зручного інтерфейсу є: *простота, зрозуміла навігація, візуальна ієрархія, консистентність (послідовність), зворотній зв'язок, мобільна адаптивність, доступність*[14].

РОЗДІЛ 2. ТЕХНОЛОГІЯ ПРОЄКТУВАННЯ ВЕБСАЙТУ ТА ЧАТ-БОТУ

2.1 Розробка чат-бота, створення діалогів, інтеграція з сайтом.

Чат-бот (*chatbot)* — це програмний додаток, який імітує спілкування з користувачем за допомогою текстових або голосових повідомлень. Основна мета чат-бота — забезпечити автоматизовану взаємодію з користувачем без участі людини-оператора. Він може бути реалізований у вигляді окремого модуля на вебсайті, у мобільному застосунку або інтегрований у месенджери (наприклад, Telegram, Viber, WhatsApp, Facebook Messenger)[15].

Історія та еволюція

Перші чат-боти з'явилися ще в 1960-х роках. Найвідомішим прикладом є бот ELIZA (1966), який симулював поведінку психотерапевта[16]. У 1990-х роках з'явилися перші боти для ICQ та IRC, а з розвитком штучного інтелекту у 2010-х роках відбувся стрімкий прорив: чат-боти почали використовувати алгоритми машинного навчання, обробки природної мови (NLP) та голосові технології[17].

Як працює чат-бот

Робота чат-бота базується на кількох етапах: розпізнавання запиту користувача (через ключові слова або NLP), аналіз наміру (intent) — визначення, що саме хоче зробити користувач (наприклад, забронювати стіл), виконання дії — бот надає інформацію або здійснює потрібну операцію, формування відповіді — повідомлення генерується відповідно до сценарію або за допомогою штучного інтелекту[18].

Типи чат-ботів: сценарні (rule-based) - *працюють за фіксованими діалогами, простота реалізації, не розпізнають відкриті запити, ідеальні для простих задач: бронювання, FAQ[19];* інтелектуальні (AI-based) - *використовують NLP і машинне навчання, можуть вести «вільну» бесіду, вчаться на попередніх взаємодіях, дорожчі у розробці, але гнучкіші[20];* гібридні чат-боти - *поєднують переваги обох типів, мають базовий сценарій + AI-модулі для гнучкості[21].*

У контексті кваліфікаційної роботи чат-бот використовується як інструмент оптимізації процесу бронювання столиків**.** Він дозволяє клієнту швидко вибрати

зручний час, кількість гостей, підтвердити бронювання та, за потреби, отримати відповіді на супутні запитання — без очікування відповіді адміністратора.

Одним із ключових елементів створеної системи є чат-бот, який дозволяє користувачам здійснювати бронювання столиків безпосередньо через мобільний додаток Telegram. Основна мета бота — спростити взаємодію клієнта з рестораном, автоматизувати бронювання та зменшити навантаження на персонал. Розробка чат-бота включала такі основні етапи: налаштування платформи, проєктування діалогів, реалізація функціоналу та інтеграція з вебсайтом[24].

pyTelegramBotAPI

pyTelegramBotAPI — це одна з найвідоміших і найпопулярніших Python-бібліотек для створення ботів у Telegram. Якщо говорити по-простому, це інструмент, який дозволяє писати бота буквально у кілька рядків коду, не занурюючись у складну технічну документацію Telegram API.

Бібліотека є *обгорткою над офіційним Telegram Bot API*, що значить: вона спрощує тобі життя, захоплюючи всі «низькорівневі» процеси — від отримання запитів і обробки оновлень, до форматування відповідей у потрібному Telegram-форматі. Весь «бруд» — всередині, а тобі лишається зосередитись на логіці.

Чому нею користуються? Причин насправді багато. Але головне — простота**,** гнучкість і зрозумілий стиль програмування**.** Тобі не потрібно вникати, як влаштовані HTTP-запити, як парсити JSON, або як працює webhook. Все, що тобі потрібно — це написати функцію, яка буде відповідати на повідомлення користувача[25].

```
@bot.message_handler(commands=['start'])
def welcome(message):
    bot.reply_to(message, "Привіт! Раді бачити тебе.")
```

Усе. Один декоратор і одна функція. Не потрібно нічого вигадувати — бібліотека сама подбає, щоб твій бот зрозумів, що користувач написав /start, і виконав потрібну дію.

Бібліотека організована дуже логічно. Основні компоненти: *telebot.TeleBot* — головний об'єкт, який ти створюєш, коли хочеш запустити бота, *message_handler* —

механізм, що дозволяє визначити, які саме повідомлення ловити й обробляти, *types* — це набір класів, які відповідають усім можливим Telegram-типам: повідомленням, користувачам, кнопкам, клавіатурам тощо, *util* — допоміжні інструменти, які іноді потрібні для розширених задач[22].

Для повного розуміння на що саме здатна бібліотека *pyTelegramBotAPI*:

Обробка повідомлень

Можна задати функції, які реагуватимуть на певні команди (/start, /help), або навіть на будь-який текст, фото, відео чи голосове повідомлення.

```
@bot.message_handler(content_types=['text'])
def echo(message):
    bot.send_message(message.chat.id, f"Ти сказав: {message.text}")
```

Клавіатури і кнопки

Бібліотека підтримує і reply-клавіатури, і inline-кнопки. Причому настільки просто, що навіть новачок зможе реалізувати меню з кнопками за 5 хвилин.

```
markup = types.ReplyKeyboardMarkup(resize_keyboard=True)
markup.add("Інформація", "Зв'язок")
bot.send_message(chat_id, "Що тебе цікавить?", reply_markup=markup)
```

Callback-запити

Коли користувач натискає кнопку в повідомленні, бот може миттєво реагувати — це реалізується через callback data.

```
@bot.callback_query_handler(func=lambda call: True)
def callback_handler(call):
    if call.data == "info":
        bot.send_message(call.message.chat.id, "Ось трохи інформації…")
```

Асинхронна підтримка

Для тих, хто працює з великим навантаженням або хоче будувати реактивні системи, є версія з asyncio. Вона дозволяє запускати бота в асинхронному режимі, що сильно підвищує ефективність.

Встановлення та запуск

Усе дуже просто — один рядок в терміналі:

pip install pyTelegramBotAPI

А далі в коді — імпортуєш як telebot:

import telebot

Найпростіший приклад, який можна прямо зараз вставити і протестувати:

Текст коду:

```
import telebot
bot = telebot.TeleBot("ВАШ_ТОКЕН")
@bot.message_handler(commands=['start'])
def start(message):
bot.send_message(message.chat.id, "Привіт! Я твій новий Telegram-бот.")
bot.polling()
```

polling() — це метод, який запускає нескінченний цикл і слухає всі вхідні повідомлення.

pyTelegramBotAPI — це не просто зручна бібліотека. Це справжній must-have інструмент для всіх, хто хоче почати писати Telegram-ботів на Python. Вона дозволяє робити складне — просто. І що ще важливіше — вона не заважає. Немає зайвої магії, ніяких прихованих процесів — усе прозоро, зрозуміло й підконтрольно[23].

Створення чат-боту для бронювання столиків у ресторані

Імпорт бібліотек та ініціалізація

```
import telebot
from telebot import types
from datetime import datetime, timedelta
import threading
```

Тут імпортуються:

telebot — для управління ботом

types — для створення кнопок та клавіатур

datetime — для роботи з датою та часом

threading — для запуску таймера скасування бронювання

Створюється об'єкт bot, через який відбувається взаємодія з Telegram.

```
TOKEN = '...'  # Твій унікальний токен від BotFather
bot = telebot.TeleBot(TOKEN)
```

Збереження даних бронювання

```
reservations = {}  # Збереження підтверджених бронювань
pending_reservations = {}  # Очікувані (ще не підтверджені)
```

Це словники, які використовуються як тимчасова база даних у пам’яті. Ключем є chat_id користувача.

Обробка команди /start

```
@bot.message_handler(commands=['start'])
def start(message):
...
```

Коли користувач пише /start, бот показує клавіатуру з двома варіантами:

- Забронювати столик
- Моє бронювання

Це зручно — користувачу не потрібно нічого вводити вручну.

Початок бронювання: вибір дати

```
@bot.message_handler(func=lambda message: message.text == "Забронювати столик")
def reserve_table(message):
...
```

Коли натиснута кнопка "Забронювати столик", бот просить ввести дату у форматі ДД.ММ.РРРР.

У наступному кроці функція get_date():

Перевіряє, щоб дата не була в минулому

Якщо все добре — запитує годину

Введення часу та кількості людей

```
def get_time(message, date):
...
```

Користувач вводить час (ГГ:ХХ). Якщо формат неправильний — бот просить повторити. Якщо правильний — далі запитується кількість людей.

Підтвердження бронювання (inline-кнопки)

def confirm_reservation(message, date, time):

...

Коли користувач вказав кількість людей, бот:

Формує повідомлення з інформацією про бронювання

Додає inline-кнопки: "Підтвердити" або "Скасувати"

Зберігає ID повідомлення в pending_reservations

Запускає таймер на 2 хвилини: якщо користувач нічого не натисне — бронювання скасовується[26].

Автоматичне скасування (таймер)

def auto_cancel(chat_id, message_id):

...

Через 120 секунд (2 хв) перевіряється, чи не підтверджене бронювання — і видаляється з повідомленням "Час на підтвердження вийшов".

Обробка підтвердження або скасування

@bot.callback_query_handler(func=lambda call: call.data.startswith('confirm') or call.data == 'cancel')

def callback_confirm(call):

...

Ця функція ловить натискання кнопок:

Якщо cancel — бот просто повідомляє, що бронювання скасовано

Якщо confirm — бот:

Зберігає бронювання в reservations

Виводить фінальне повідомлення з деталями

Перегляд поточного бронювання

@bot.message_handler(func=lambda message: message.text == "Моє бронювання")

def my_reservation(message):

...

Якщо користувач хоче переглянути своє бронювання — бот показує дані з reservations.

Запуск бота

bot.polling(none_stop=True)

polling() — запускає нескінченне опитування Telegram на нові повідомлення. Бот працює у реальному часі.

В додатку Б прописано повний код чат-боту на платформі Telegram для бронювання столиків у ресторані.

2.2 Структура та розробка вебсайту для бронювання столиків у ресторані.

Вебсайт — це електронний ресурс, доступний через мережу Інтернет, який складається з набору вебсторінок, пов’язаних між собою гіперпосиланнями та об'єднаних єдиною темою, дизайном або функціональністю[27]. Як правило, вебсайт має єдину домашню (головну) сторінку, від якої користувач може перейти до інших сторінок, що містять текстову, графічну, аудіо-, відеоінформацію чи інтерактивні елементи[28].

З технічного погляду, сайт є програмним продуктом, що працює на базі веб-сервера і обслуговується за допомогою серверного та клієнтського програмного забезпечення[29].

Основні функції вебсайту: інформаційна — подання інформації про компанію, послуги, контактні дані, комунікаційна — забезпечення зворотного зв’язку через форми, чати, email, комерційна — можливість здійснення замовлень, продажів, бронювань, іміджева — створення та підтримка позитивного іміджу компанії/бренду, інтерактивна — участь користувача у взаємодії з системою (вибір, введення даних, відповіді)[30].

Вебсайт ресторану як приклад сервісного сайту

Сайт ресторану поєднує в собі кілька ключових функцій: презентацію закладу, доступ до меню, а також можливість взаємодії з клієнтом у реальному часі — зокрема, через форми онлайн-бронювання або чат-бот.

Такі сайти, як правило, включають: головну сторінку з фото, описом концепції ресторану, акціями, меню з інтерактивним відображенням страв, описами та цінами, форму бронювання столиків (через віджет або вбудований сервіс), контакти із інтегрованою картою Google Maps, телефонами, соцмережами, відгуки клієнтів і

галерею зображень, чат-бот або онлайн-консультант для допомоги у виборі страв чи здійснення бронювання[31].

Роль вебсайту у бізнесі

У сучасних умовах вебсайт є не лише «візитною карткою» бізнесу, а й повноцінною платформою для комунікації, продажів та обслуговування клієнтів. У галузі громадського харчування (ресторани, кафе, бари) наявність зручного та функціонального сайту суттєво впливає на кількість клієнтів, швидкість обслуговування, зниження навантаження на персонал, а також на конкурентоспроможність закладу загалом[32].

Вимоги до сучасного вебсайту ресторану

У процесі розробки вебсайту для ресторану слід враховувати не лише функціональність, але й вимоги до дизайну, зручності користування, безпеки та технічної ефективності. Успішний сайт має забезпечувати позитивний досвід користувача (UX), бути візуально привабливим і підтримувати інтеграцію з додатковими сервісами (бронювання, замовлення, доставка)[33].

Основні вимоги до сайту ресторану: адаптивність - сайт має коректно відображатися на різних пристроях: комп'ютерах, планшетах, смартфонах, адаптивність забезпечує легкий доступ до функцій бронювання або меню з мобільних пристроїв, що є критично важливим для сучасних користувачів; *швидке завантаження* - оптимізація зображень, кешування та мінімізація скриптів забезпечують швидку роботу сайту, затримки понад 3 секунди значно знижують імовірність того, що відвідувач залишиться на сайті; *зрозуміла навігація* - інтерфейс має бути інтуїтивним (головне меню, кнопки дій, зрозумілі назви сторінок), користувач повинен швидко знаходити потрібну інформацію (меню, години роботи, контакти, форма бронювання); *SEO-оптимізація (пошукове просування)* - сайт має бути оптимізований для пошукових систем, щоб клієнти могли знайти ресторан через Google; *безпека* - підключення SSL-сертифіката (https), захист даних користувача при бронюванні чи оплаті, надійне зберігання персональних даних відповідно до вимог GDPR (для ЄС) або локального законодавства; *інтеграція онлайн-сервісів* - онлайн-бронювання столиків (через форму, віджет, чат-бот),

підключення до Google Maps, соцмереж, платіжних систем; якісний візуальний контент - професійні фото інтер'єру, страв, атмосфери ресторану, відеопрезентації, галереї, відгуки клієнтів підвищують довіру до закладу; *багатомовність (за потреби)* - у туристичних зонах бажано мати англійську версію сайту для іноземних гостей; *доступність* - вебсайт має бути доступним для людей з порушеннями зору (контрастність, масштабування, навігація з клавіатури)[34, 35].

Ці вимоги формують стандарти якості для вебсайтів у ресторанній галузі, дозволяючи ефективно представити заклад онлайн, забезпечити просту та зручну взаємодію з клієнтом і підвищити рівень обслуговування.

Wix — універсальна платформа для створення сайтів

Світ стрімко змінюється, і онлайн-присутність стала ключовим елементом успіху для будь-якого бізнесу, особистого бренду чи організації. Wix — це хмарна платформа для створення вебсайтів, яка надає користувачам можливість легко створювати, редагувати і публікувати сайти без знань у сфері програмування. Wix — це одна з найпопулярніших платформ для створення сайтів без необхідності володіння мовами програмування. Вона надає користувачам набір інструментів для побудови повноцінних веб-сайтів за допомогою візуального редактора, що працює за принципом «drag-and-drop». Платформа дозволяє створювати персоналізовані сторінки, підключати плагіни, додавати анімації, форми та інші інтерактивні елементи[36].

Створена у 2006 році, компанія Wix швидко перетворилася на одного з лідерів ринку, з більш ніж 200 мільйонами користувачів у всьому світі. Платформа орієнтована на доступність, гнучкість і креативність, що робить її особливо популярною серед фрілансерів, малих підприємств і творчих професіоналів. Wix пропонує готові шаблони сайтів, орієнтовані на конкретні галузі бізнесу, зокрема — ресторани, кафе та служби доставки. Це дозволяє значно зменшити час на розробку, оскільки основна структура сайту вже підготовлена, і потрібно лише змінити контент відповідно до потреб замовника.

Глибокий аналіз функціоналу Wix

Дизайн і шаблони

Wix пропонує понад 800 професійних шаблонів, створених дизайнерами з урахуванням актуальних трендів UX/UI. Шаблони охоплюють найрізноманітніші сфери: мода, техніка, їжа, медицина, події, освіта, нерухомість, благодійність тощо. Усі шаблони можна адаптувати під власні потреби: змінювати кольори, шрифти, макети, додавати анімації та інтерактивні елементи[44].

Редактор Wix Studio (Pro версія)

Wix не стоїть на місці. Нещодавно вони представили Wix Studio — професійне середовище для дизайнерів і агенцій з розширеними можливостями дизайну, включаючи: глобальні стилі і компоненти, підключення API і кастомних функцій, спільна робота в команді, підтримка анімацій, grid-верстки, автоматизації[42].

Бази даних і динамічні сторінки

За допомогою Velo by Wix, користувачі можуть створювати динамічні сторінки, які автоматично оновлюються з бази даних. Це особливо корисно для каталогів, блогів, списків подій, сторінок профілю користувача, портфоліо тощо.

SEO-інструменти

Wix значно покращив свою SEO-оптимізацію за останні роки. Сьогодні платформа підтримує: можливість редагування мета-тегів, заголовків, описів, генерацію sitemap.xml та редагування robots.txt., підключення Google Search Console, створення канонічних URL і редиректів, оптимізацію швидкості завантаження сторінок[39].

Блог і контент-маркетинг

Wix дозволяє легко вести блог, додаючи записи, рубрики, обкладинки, SEO-налаштування та навіть коментарі. Також можна підключити RSS, планувати публікації, створювати авторські профілі та інтегрувати соціальні мережі[40].

Wix ADI (Artificial Design Intelligence)

Цей інструмент використовує штучний інтелект для автоматичного створення сайту на основі відповідей користувача на кілька запитань. Це ідеально підходить для тих, хто хоче швидко запустити сайт без глибокого занурення в дизайн.

Мобільна оптимізація

Усі сайти, створені на Wix, автоматично адаптуються до мобільних пристроїв. Користувачі також можуть окремо налаштовувати мобільну версію сайту для кращого користувацького досвіду.

Інтеграції та додатки

Через Wix App Market користувачі можуть додавати на свій сайт різноманітні додатки, такі як онлайн-чати, форми зворотного зв'язку, системи бронювання, інструменти для маркетингу та багато іншого[43].

Wix для бізнесу та підприємництва

eCommerce на Wix

Інтернет-магазини на Wix отримують:Інтеграцію з платіжними системами (Stripe, PayPal, Wix Payments), налаштування варіантів доставки та податків, облік залишків, автоматичні нагадування про покинуті кошики, підтримку цифрових і фізичних товарів, маркетинг-автоматизацію (email, купони, акції).

Онлайн-запис і послуги

Для бізнесів, що надають послуги (косметологи, тренери, психологи тощо), Wix пропонує: Wix Bookings — система запису клієнтів онлайн, календар, сповіщення, прийом оплат, скасування записів, інтеграція з Google Calendar.

Інструменти просування: маркетинг і аналітика

Wix надає інструменти, які допомагають залучати відвідувачів і конвертувати їх у клієнтів: Email-маркетинг (розсилки, шаблони листів), Wix Ascend — CRM, автоматизація маркетингу, воронки продажів, інтеграція з Facebook Ads, Google Ads, Instagram, вбудована аналітика або підключення Google Analytics/Hotjar[38].

Безпека і хостинг

Wix забезпечує високий рівень безпеки: SSL-сертифікат для кожного сайту, захист від DDoS-атак, автоматичні резервні копії сайту, надійний хостинг на глобальній інфраструктурі[37].

Переваги Wix: Інтуїтивний інтерфейс дозволяє створювати сайти без технічних знань, можливість повного налаштування шаблонів та елементів сайту, понад 800 шаблонів для різних галузей, можливість додавання сторонніх додатків через App Market, автоматична адаптація сайту до мобільних пристроїв[41].

Недоліки Wix: Наявність реклами Wix та обмежений функціонал, для зміни шаблону потрібно створювати новий сайт, хоча Wix покращив свої SEO-інструменти, деякі користувачі все ще вважають їх менш гнучкими порівняно з іншими платформами.

Wix є ефективним інструментом для швидкої розробки веб-сайту ресторану з інтеграцією функцій бронювання. У межах даної роботи було реалізовано повнофункціональний сайт, що включає інтерактивне меню, онлайн-бронювання та чат-бота. Такий підхід дозволяє суттєво зекономити час на розробку та забезпечити якісний користувацький досвід.

Розробка вебсайту

У межах реалізації практичної частини кваліфікаційної роботи було створено функціональний вебсайт, що виконує роль онлайн-платформи для попереднього резервування столиків у ресторані. Вебсайт розроблено з використанням конструктора Wix, який забезпечив гнучкість, швидкість впровадження та адаптивність без необхідності поглибленого програмування.

Мета створення сайту

Головною метою є полегшення процесу комунікації між закладом та клієнтами, надання їм можливості онлайн-бронювання у зручний час, а також ознайомлення з меню, інтер’єром і розташуванням ресторану. Вебсайт служить також маркетинговим інструментом для просування бренду в онлайн-середовищі.

Структура сайту

Сайт має логічно побудовану структуру з чітко визначеними інформаційними блоками: *Головна сторінка,* яка містить короткий опис закладу, атмосферні зображення, кнопки швидкої дії («Забронювати столик», «Меню», «Про нас»), *Меню* - окремий розділ, що візуально представляє категорії страв та напоїв із описом,

Бронювання - інтерактивна форма, реалізована через Wix Bookings, дозволяє обрати дату та час бронювання, залишити контактні дані, *Про нас* - сторінка з короткими відомостями про заклад харчування.

Дизайн та адаптивність

Для оформлення сайту було обрано шаблон із категорії «Ресторани» на Wix (Рис 2.1).

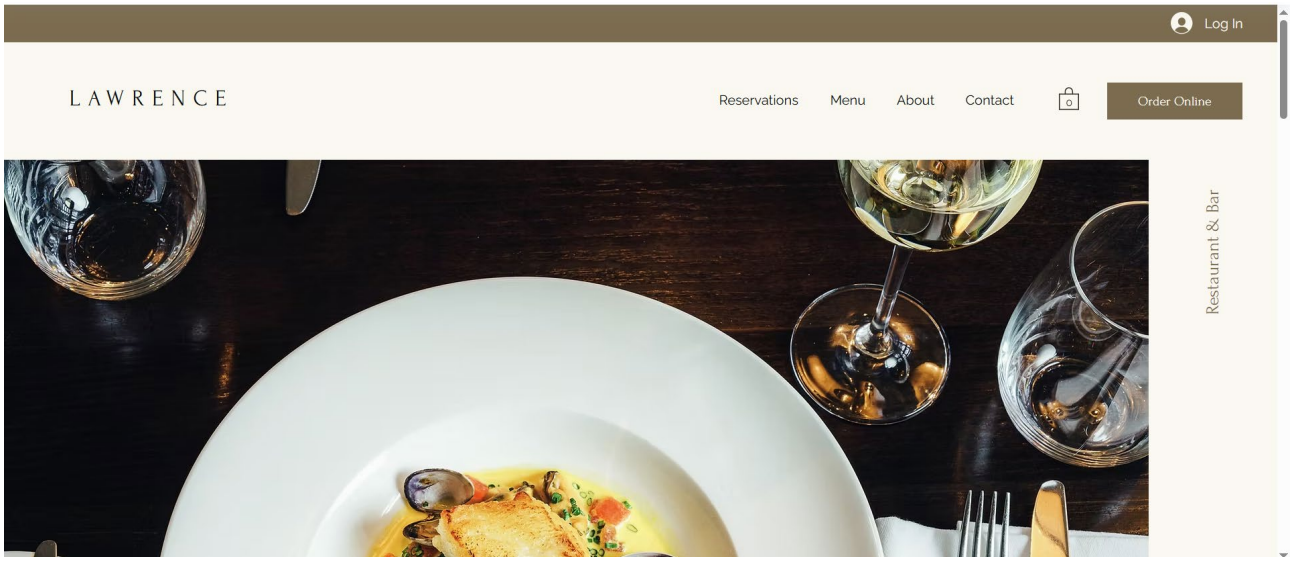


Рис 2.1 Шаблон вебсайту з категорії «Ресторани»

Його адаптовано під індивідуальні вимоги: кольорова гама витримана у теплих тонах, що створює затишну атмосферу, шрифти підібрано відповідно до стилістики закладу, адаптивний дизайн забезпечує коректне відображення на смартфонах і планшетах, елементи інтерфейсу зручні для користувача та інтуїтивно зрозумілі.

Опис сайту

Головна сторінка

Головна сторінка є візитівкою вебресурсу та першою точкою контакту користувача з рестораном. Вона виконує роль інформативного, естетичного та навігаційного хаба, забезпечуючи швидкий доступ до основних функцій сайту.

Структура та елементи головної сторінки:

Хедер (верхня панель): логотип ресторану з переходом на головну сторінку, меню навігації (посилання на розділи «Меню», «Бронювання», «Про нас»), мова

інтерфейсу (у разі багатомовної версії), кнопка швидкого бронювання у вигляді CTA ("Call to Action") (Рис 2.2).

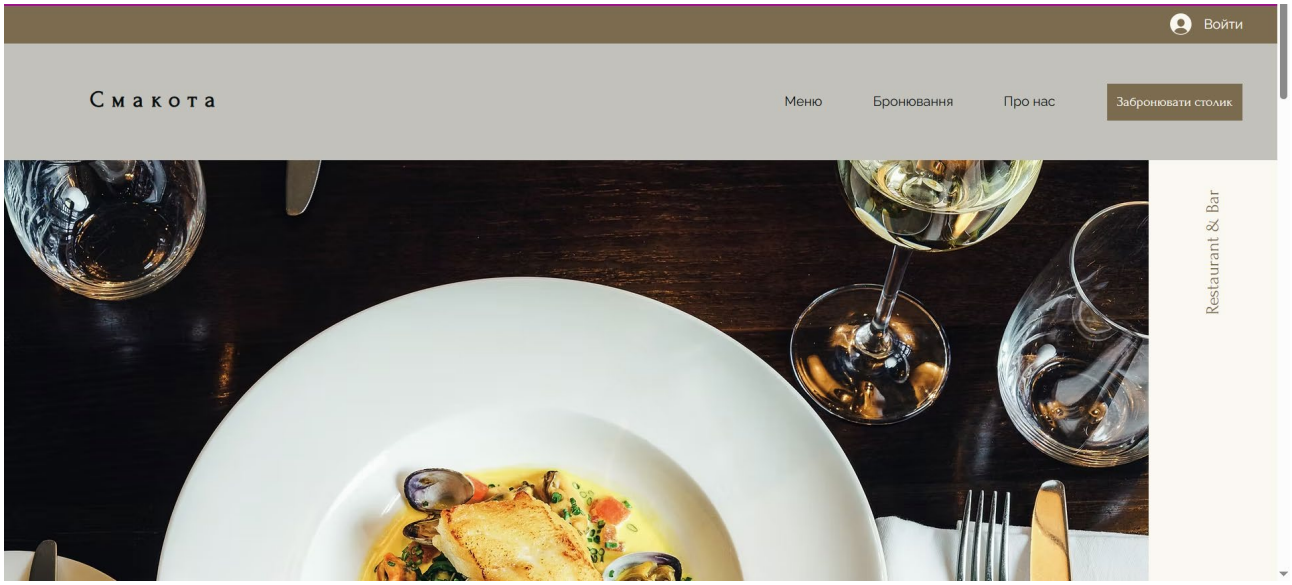


Рис 2.2 Хедер на головній сторінці сайту

Банер-слайдер: велике зображення або відеофон з елементами брендингу (Рис 2.3).



Рис 2.3 Банер-слайдер

Інформаційний блок про ресторан: короткий опис концепції, кухні, атмосфери, кнопка переходу на сторінку «Про нас», фото інтер'єру або процесу приготування страв (Рис 2.4).

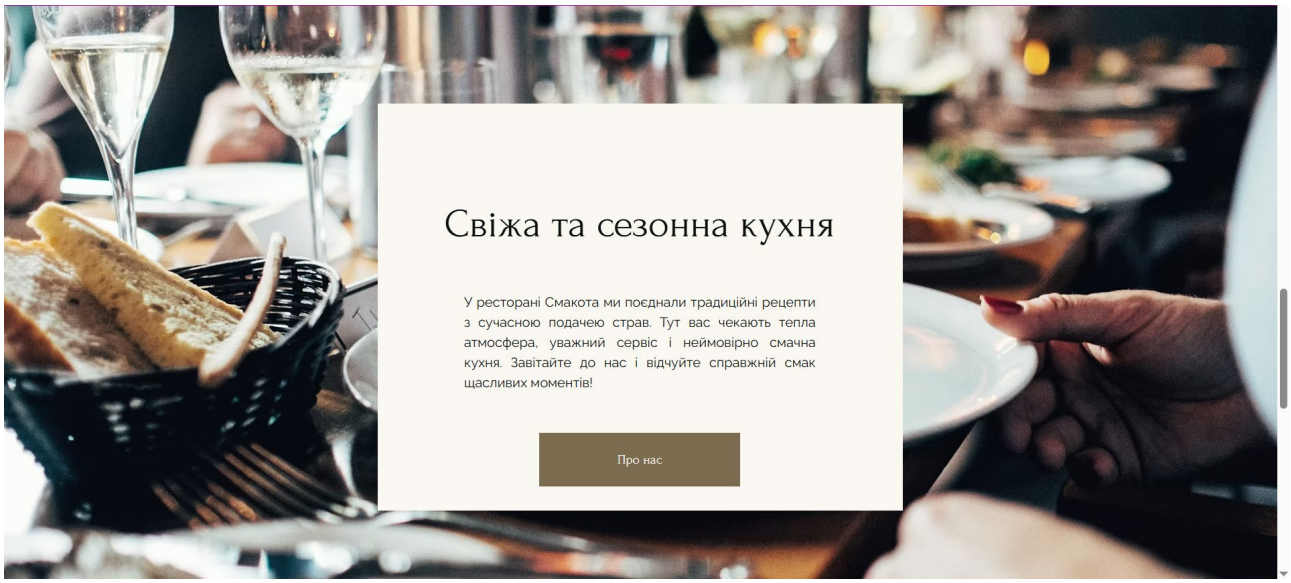


Рис 2.4 Інформаційний блок про ресторан

Перехід на сторінку «Меню»: короткий спонукаючий текст та фото однієї зі страв, кнопка переходу до повного меню (Рис 2.5).

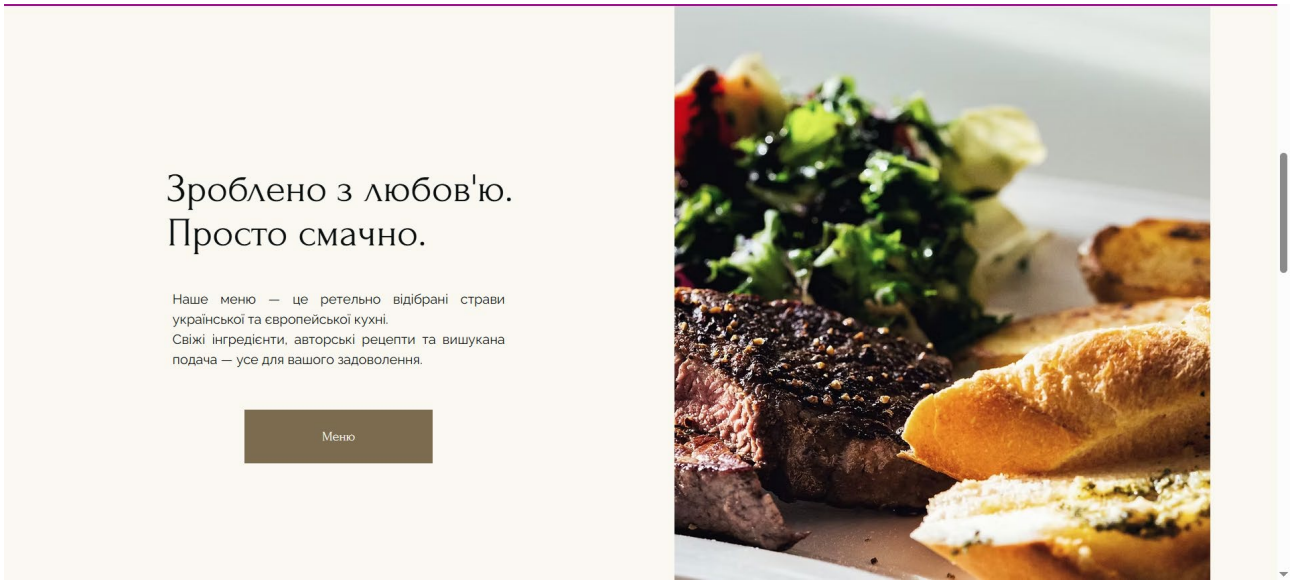


Рис 2.5 Інтеграція переходу до сторінки «Меню»

Футер (нижня частина сторінки): контактна інформація (телефон, e-mail, адреса), іконки соцмереж (Instagram, Telegram, Facebook, TikTok), робочий час закладу (Рис 2.6).

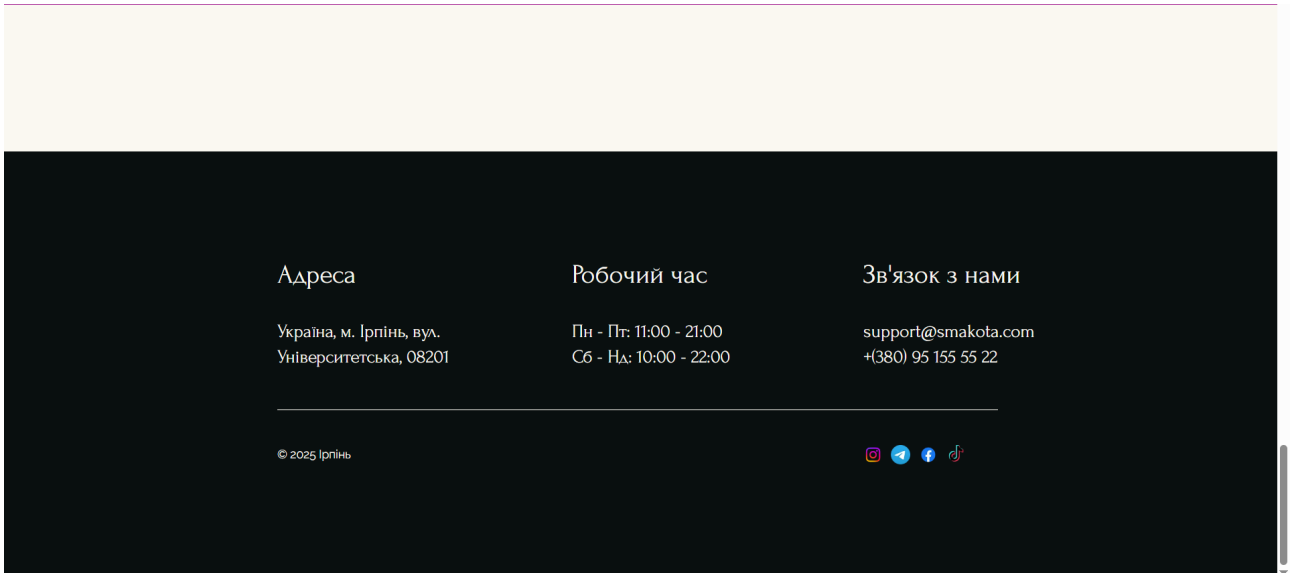


Рис 2.6 Футер сайту

Адаптивність

Сторінка оптимізована для мобільних пристроїв. Всі блоки масштабуються, кнопки мають достатню зону натискання, а меню трансформується в «гамбургер»-формат.

Сторінка «Меню»

Сторінка «Меню» виконує важливу функцію ознайомлення потенційних клієнтів з гастрономічною пропозицією ресторану. Вона впливає на прийняття рішення щодо бронювання та створює перше враження про рівень закладу.

Структура сторінки:

Верхній блок (шапка меню): назва сторінки «Меню», розміщена великим шрифтом.

Категорії страв: меню поділено на логічні розділи з інтерактивною навігацією або вкладками: Закуски, Головні страви, Десерти.

Карточки страв: кожна страва представлена окремим блоком – назва, опис (інгредієнти, спосіб приготування, особливості), ціна (в національній валюті) (Рис 2.7).

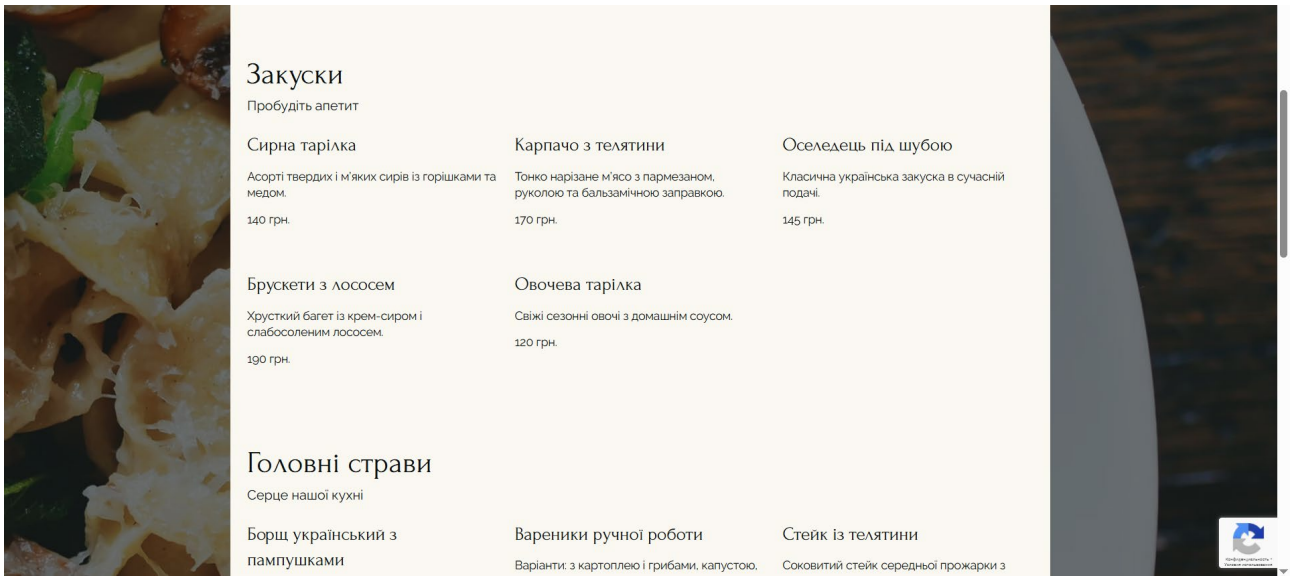


Рис 2.7 Інтерфейс сторінки «Меню»

Сторінка «Бронювання»

Сторінка «Бронювання» є ключовим функціональним елементом сайту, адже саме вона забезпечує інтерактивну взаємодію користувача із закладом. Завдяки вбудованому модулю на базі *Wix Bookings*, процес резервування спрощено до кількох кліків.

Структура сторінки

Заголовок: назва сторінки - «Забронювати столик».

Інтерактивна форма бронювання

Реалізована через *Wix Bookings*, форма містить: вибір дати та часу — інтерактивний календар із можливістю вибору дати та часу бронювання столика, контактна інформація — поля для введення імені, номера телефону, email, додаткові побажання — вікно для коментарів (наприклад, «столик біля вікна», «дитячий стілець»), кнопка з активним підсвічуванням: «Підтвердити бронювання» (Рис 2.8).

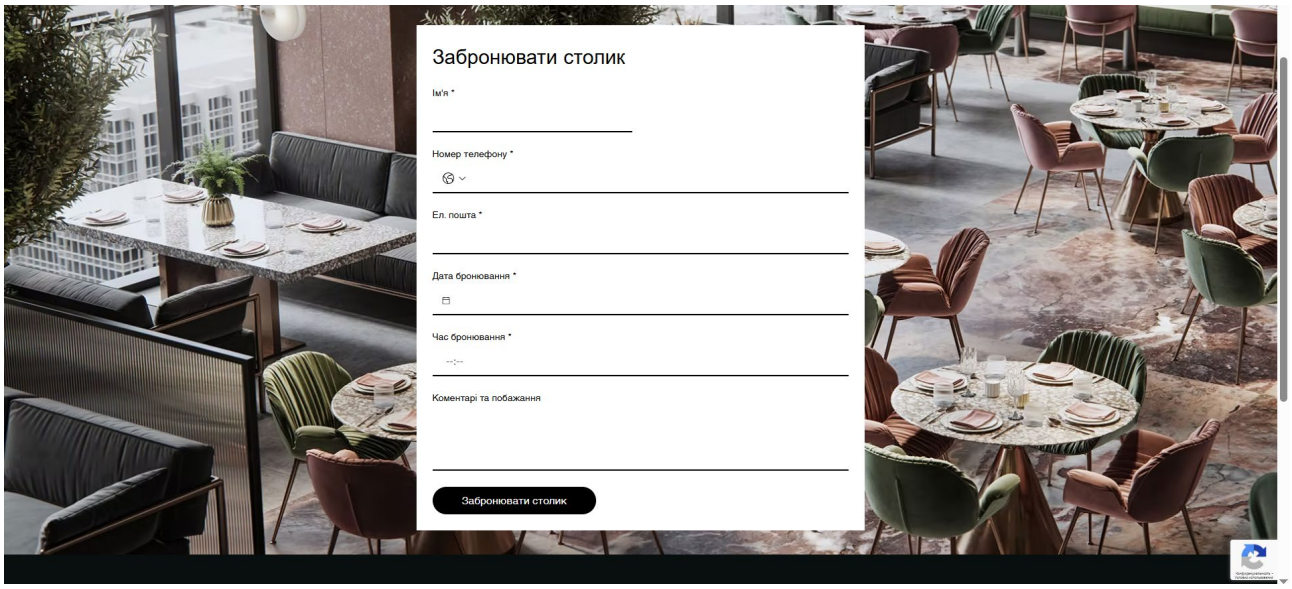


Рис 2.8 Інтерфейс сторінки «Бронювання»

Адаптивність

Сторінка бронювання повністю оптимізована для смартфонів: календар і поля легко прокручуються, форми мають автофокус та автозаповнення, кнопка підтвердження розміщена у зоні легкого доступу для пальця.

Сторінка «Про нас»

Сторінка «Про нас» є важливим інформаційним блоком, що формує довіру до закладу та створює емоційний зв’язок із відвідувачем. Вона презентує філософію ресторану, історію його створення, унікальність кухні та підхід до обслуговування.

Структура сторінки

Заголовок: текст «Про нас» великим шрифтом, атмосферне фото фону з кадрами із життя ресторану.

Історія та концепція: текстовий блок, що розповідає основну ідею або філософію закладу (Рис 2.9).

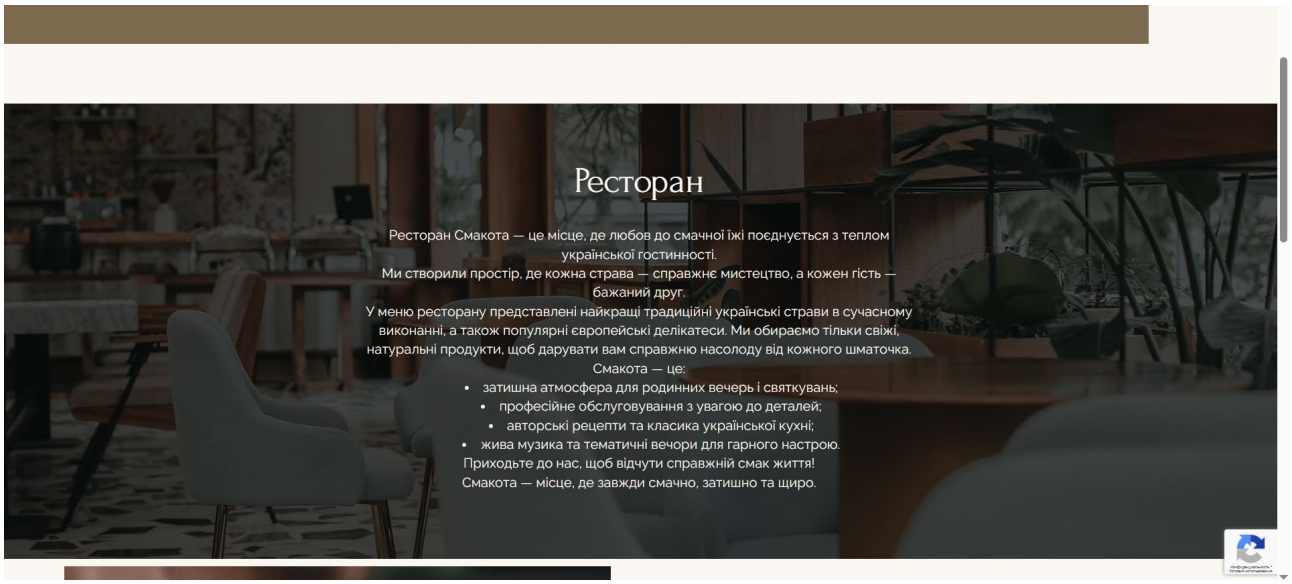


Рис 2.9 Ідея та філософія ресторану

Унікальні особливості закладу: кулінарна концепція (авторські страви, локальні інгредієнти), принцип обслуговування (турбота, індивідуальний підхід) (Рис 2.10).



Рис 2.10 Кулінарна концепція закладу харчування

Тестування та запуск

Перед запуском сайт пройшов повний цикл тестування: перевірено коректність заповнення форм та надсилання повідомлень, протестовано усі посилання й переходи між сторінками, оцінено швидкість завантаження на різних

пристроях, проведено коригування елементів дизайну та контенту згідно з виявленими недоліками.

В додатку А прописано текст коду вебсайту для бронювання столиків у ресторані.

Висновок

Розроблений вебсайт є ефективним інструментом для ресторану, який забезпечує якісну онлайн-взаємодію з клієнтами, оптимізує бронювання, підвищує впізнаваність бренду та покращує загальний користувацький досвід. Завдяки функціоналу Wix вдалося швидко створити адаптивний, візуально привабливий та зручний для користувача інструмент. Сайт сприяє автоматизації обслуговування клієнтів, розширенню присутності закладу в інтернет-просторі, підвищенню лояльності та якості сервісу. У майбутньому можливе масштабування функціоналу, зокрема додавання модуля онлайн-оплати або мобільного додатку.

РОЗДІЛ 3. АНАЛІЗ ТА ОЦІНКА РЕЗУЛЬТАТІВ

Розроблений вебсайт має сучасний дизайн, є інформативним та виконує функції цифрової вітрини ресторану. Основні реалізовані функції: головна сторінка — приваблює відвідувача слайдером з фото страв та гаслом ресторану. Включає навігаційне меню, що дозволяє зручно переходити між розділами, меню страв — динамічна галерея з фото, описом та цінами на популярні страви. У подальшому передбачено додавання можливості онлайн-замовлення страв, форма бронювання столика — дозволяє відвідувачу забронювати столик, вказавши дату, час, кількість осіб, ім'я та номер телефону. Дані надсилаються на сервер і зберігаються в базі даних або надсилаються електронною поштою адміністратору, сторінка «Про нас» — розповідає історію закладу, філософію обслуговування, відмінності від конкурентів, контактна інформація — містить інтерактивну карту (Google Maps), email, телефон, посилання на соцмережі, адаптивність — сайт коректно відображається на ПК, планшетах та смартфонах завдяки використанню медіазапитів CSS та фреймворків (наприклад, Bootstrap).

Реалізація чат-боту

Для автоматизації процесу взаємодії з клієнтами було створено чат-бот, який працює в месенджері Telegram (або Facebook Messenger / Viber — залежно від обраної платформи).

Основні функції чат-боту: привітання користувача та пропозиція здійснити бронювання, запит і збереження параметрів бронювання: дата, час, кількість осіб, контактна інформація, можливість скасування або зміни замовлення, відповіді на часті запитання (FAQ): години роботи, адреса, меню, відправлення підтвердження бронювання.

Технології: мова програмування - Python; платформа: Telegram Bot API.

Оцінка функціональності

Метою розробки системи було створення зручного та ефективного інструменту для бронювання столиків у ресторані як через вебсайт, так і за допомогою чат-боту. У процесі реалізації були визначені ключові функціональні

вимоги, відповідність яким є критерієм оцінки успішності розробки. Основні завдання, поставлені перед системою, включали: забезпечення користувачів можливістю переглядати доступні для бронювання столики, можливість здійснювати бронювання з вибором дати, часу та кількості осіб, створення чат-боту, здатного автоматично обробляти запити на бронювання, надання адміністрації ресторану інструментів для перегляду, редагування та скасування бронювань, збереження та обробка даних у захищеній базі даних, забезпечення підтвердження бронювання через e-mail або месенджер.

Проведене функціональне тестування показало, що система в повному обсязі виконує всі заявлені завдання. Зокрема: вебсайт надає користувачеві інтуїтивний інтерфейс для створення бронювання, перегляду доступних дат і часу, з урахуванням обмежень за кількістю місць, чат-бот у Telegram успішно розпізнає запити користувача, формує бронювання та надсилає підтвердження, адміністративна панель дозволяє персоналу ресторану керувати бронюваннями в режимі реального часу, база даних фіксує всі запити та забезпечує цілісність і безпеку збережених даних.

Функціональне тестування здійснювалося шляхом моделювання стандартних сценаріїв поведінки користувача та адміністратора. За підсумками тестування всі ключові функції пройшли перевірку успішно, без критичних помилок. Таким чином, система в повній мірі відповідає поставленим завданням, що свідчить про її функціональну повноцінність.

Оцінка юзабіліті

Юзабіліті — це якісна характеристика, яка визначає зручність інтерфейсу, його зрозумілість, швидкість освоєння та ефективність використання кінцевим користувачем. В межах цієї кваліфікаційної роботи було проведено повноцінний аналіз юзабіліті як вебсайту, так і чат-боту, що були створені для процесу онлайн-бронювання столиків у ресторані.

Мета оцінювання юзабіліті

Основна мета — виявити, наскільки просто та зручно користувачі можуть: знайти потрібну інформацію (дата, час, кількість осіб), здійснити бронювання з

мінімальною кількістю дій, отримати підтвердження, взаємодіяти з ботом без плутанини або повторення команд.

Методи оцінки

Для оцінки юзабіліті були використані такі методи: спостереження за користувачами (user testing) - 20 осіб виконували реальні завдання, фіксувалися час, кількість помилок, необхідність допомоги; опитування (анкетування) - після взаємодії кожен респондент оцінював 5 ключових аспектів системи за шкалою від 1 до 5; метод когнітивного walkthrough — дослідження поведінки користувача «очима новачка»; оцінка відповідності принципам Nielsen (зрозумілість, гнучкість, запобігання помилок тощо).

Основні результати

Вебсайт: час на оформлення бронювання: від 50 до 90 секунд, середня кількість кліків до завершення: 5, найчастіше використані елементи: кнопка «Забронювати», календар, форма контактних даних.

Чат-бот: час до завершення діалогу з бронюванням: 70–120 секунд, рівень розуміння команд ботом: 93% запитів оброблено коректно, позитивний досвід: користувачі хвалили швидкість і зручність, особливо через смартфон, недоліки: деякі користувачі очікували, що бот зрозуміє запити у вільній формі (наприклад: *«А можна на 19:00?»*), що не працювало.

Приклади користувацьких коментарів

«Дуже просто! Навіть моя мама змогла забронювати з першого разу» — (Користувачка, 38 роки)

«Хотілося б бачити фото столика перед бронюванням» — (Користувач, 42 рік)

«Бот трохи "дерев’яний" — просив писати все окремо. Хоча спрацювало швидко» — (Користувач, 21 років)

Порівняльний аналіз з іншими системами

Для порівняння були проаналізовані подібні рішення конкурентів. Основні переваги власної системи: менше кроків до оформлення бронювання, інтеграція з месенджерами — користувач не переходить на сайт, простий, локалізований інтерфейс без зайвих елементів.

Пропозиції щодо покращення

У ході реалізації даного проєкту було закладено фундамент для створення повноцінної цифрової платформи обслуговування клієнтів ресторану. Проте, враховуючи розвиток технологій, зміни у поведінці користувачів та конкурентне середовище, існує низка напрямків для подальшого вдосконалення та масштабування системи: інтеграція з внутрішніми базами даних і CRM-системами, розширення функціональності чат-бота, панель адміністратора (Back-office система), мобільний додаток, онлайн-оплата та передплата, AI-персоналізація та рекомендації, інтеграція з соціальними мережами, автоматизоване управління чергами, аналітика та звітність, безпека та захист даних.

Ці перспективи можуть бути реалізовані поетапно залежно від потреб бізнесу, бюджету та технічних ресурсів. Основна перевага розробленої системи — її масштабованість і гнучкість, що дозволяє легко додавати нові модулі та інтеграції.

На основі зворотного зв’язку сформовані такі пропозиції: додати зображення з планом залу, реалізувати «живий» пошук у бота — підтримку довільних запитів, додати функцію «Обране» або збережені дані користувача для повторного бронювання, увімкнути багатомовність інтерфейсу (українська, англійська).

Висновок

Після реалізації всіх етапів проєкту можемо зробити висновки, що: вебсайт повністю виконує функції сучасного онлайн-представництва ресторану, чат-бот спрощує взаємодію з клієнтом і скорочує час, необхідний для бронювання, розроблене рішення є гнучким, масштабованим, а також може бути впроваджене в інших закладах харчування, всі поставлені завдання були реалізовані, а функціональність протестована.

Результати оцінки юзабіліті показали, що як вебсайт, так і чат-бот є доступними, зрозумілими і приємними у користуванні. Користувачі швидко орієнтуються в інтерфейсі, легко виконують потрібні дії. Система має високий потенціал до подальшого вдосконалення й масштабування.

ВИСНОВКИ

У межах кваліфікаційної роботи було виконано повний цикл розробки та впровадження інформаційної системи, яка забезпечує ефективне онлайн-бронювання столиків у ресторані за допомогою вебінтерфейсу та чат-бота. Актуальність теми зумовлена зростаючим попитом на автоматизовані сервіси у сфері обслуговування, а також необхідністю оптимізації бізнес-процесів у закладах громадського харчування.

У ході виконання роботи досягнуто таких основних результатів:

Теоретичне обґрунтування

Проведено аналіз предметної області, виявлено недоліки традиційних методів бронювання (телефон, запис вручну). Вивчено сучасні підходи до автоматизації в ресторанному бізнесі та проаналізовано існуючі рішення на ринку.

Проєктування системи

Визначено вимоги до системи з урахуванням потреб клієнтів та адміністрації закладу. Окрему увагу приділено безпеці зберігання та обробки персональних даних користувачів.

Реалізація функціоналу

Розроблено вебсайт із можливістю швидкого бронювання, перегляду доступних слотів, підтвердженням по email або SMS.

Створено чат-бот, який дозволяє користувачу зручно бронювати столики за кілька кроків, отримувати нагадування, а також взаємодіяти з меню або отримувати інші довідкові дані.

Тестування та оцінка результатів

Проведено функціональне тестування, що підтвердило працездатність усіх модулів. Здійснено аналіз юзабіліті: інтерфейси виявилися інтуїтивно зрозумілими, а час на бронювання скоротився у 2–3 рази в порівнянні з телефонним способом.

Перспективи розвитку

Система є масштабованою: можливе розширення на мережу ресторанів з централізованим управлінням. Можливе впровадження модуля лояльності,

інтеграція з онлайн-оплатою, розширення можливостей аналітики. Також перспективним є підключення голосових асистентів (Google Assistant, Alexa), що відкриє нові канали взаємодії з клієнтом.

Загальний висновок

Розроблена система є успішною реалізацією сучасного цифрового інструменту для ресторанного бізнесу. Вона забезпечує ефективну автоматизацію процесу бронювання, покращує досвід користувачів, оптимізує роботу персоналу та демонструє економічну доцільність впровадження. Отримані результати підтверджують, що подібні рішення мають значний потенціал масштабування та впровадження у різних форматах закладів громадського харчування.

Сайт кваліфікаційної роботи можна переглянути за посиланням: https://azar2004misha.wixsite.com/smacota

Чат-бот Telegram для бронювання столиків у ресторані можна переглянути за посиланням: https://web.telegram.org/a/#8166849860

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Іванова, Т. А., & Грінштейн, А. С. (2021). *Веб-дизайн для малого бізнесу: розробка інтерфейсів та інтерактивних рішень*. Київ: Видавництво «ДіяТех» (дата звернення: 14.04.2025).
2. Wirtz, J., Zeithaml, V. A., & Gremlet, K. (2022). *Technology and the Service Industry: The Role of AI and Chatbots*. Journal of Service Research, 25(3), 29-45 (дата звернення: 14.04.2025).
3. Загороднюк, М. О., & Чорна, А. О. (2023). *Інформаційні технології в ресторанному бізнесі: автоматизація процесів та впровадження чат-ботів*. Київ: Видавництво «ІнфоТех» (дата звернення: 14.04.2025).
4. Bajwa, S. S., & Lamba, P. (2023). *Chatbots in the Hospitality Industry: Leveraging AI for Enhanced Customer Experience*. International Journal of Hospitality Management, 45(1), 17-32 (дата звернення: 14.04.2025).
5. Чевгуз К.М. Розвиток онлайн-бронювання в ресторанному бізнесі // *Матеріали конференції "Проблеми та перспективи розвитку підприємництва в Україні"*. Суми: СумДУ, 2023. С. 78–81. [Електронний ресурс]. Режим доступу: https://essuir.sumdu.edu.ua/handle/123456789/93067 (дата звернення: 14.04.2025).
6. Захарова Т.В. Цифрові технології як фактор розвитку HoReCa у 2025 році // *Економіка. Фінанси. Менеджмент: актуальні питання науки і практики*. 2025. №3. С. 91–97. [Електронний ресурс]. Режим доступу: https://econp.com.ua/index.php/journal/article/view/283 (дата звернення: 14.04.2025).
7. Сохань Ю.С. Використання чат-ботів у системах онлайн-замовлення // *Збірник наукових праць НАУ*. 2021. №12. С. 115–120. [Електронний ресурс]. Режим доступу: https://er.nau.edu.ua/handle/NAU/51813 (дата звернення: 14.04.2025).
8. Кирилова А.В. Веб-технології в ресторанному бізнесі: адаптивність і UX-дизайн // *Наукові записки ЧНУ ім. Петра Могили*. 2020. №2 (206). С. 102–105. [Електронний ресурс]. Режим доступу: https://krs.chmnu.edu.ua/jspui/handle/123456789/1433 (дата звернення: 14.04.2025).

9. Svekis, L. Front-End Web Development with Modern HTML, CSS, and JavaScript : [Електронний ресурс] / Laurence Svekis. Birmingham : Packt Publishing, 2022. 676 с. Режим доступу: https://www.oreilly.com/library/view/front-end-web-development/9781804618172 (дата звернення: 14.04.2025).

10. Mallach, E. G. Developing Web Sites with HTML, CSS and JavaScript / Efrem G. Mallach. Boca Raton : CRC Press, 2022. 268 с. ISBN 9781003467779. (дата звернення: 14.04.2025).

11. Faulkner, D., Bouillon, P. Modern Web Development with Angular / Duncan Faulkner, Pierre Bouillon. New Delhi : AVA®, 2024. 348 с. ISBN 9788196815097. (дата звернення: 14.04.2025).

12. Building Web Applications with Angular, TypeScript and React : [Електронний ресурс] / Packt Publishing. Режим доступу: https://www.packtpub.com/en-us/product/building-web-applications-with-angular-typescript-and-react-9781789340334?srsltid=AfmBOooa79BxLomcLIVxXPFYxuk6jtkk7XRzLIlcbpHd_m6gzs4RfMLF (дата звернення: 14.04.2025).

13. Roller, S. Recipes for building an open-domain chatbot / Stephen Roller, Emily Dinan, Da Ju et al. // arXiv preprint, 2020. arXiv:2004.13637. Режим доступу: https://arxiv.org/abs/2004.13637 (дата звернення: 14.04.2025).

14. Krug, S. Don't Make Me Think: A Common Sense Approach to Web Usability / Steve Krug. 3rd ed. Berkeley, CA : New Riders, 2014. 216 p. Режим доступу: https://eng317hannah.wordpress.ncsu.edu/files/2020/01/Krug_Steve_Dont_make_me_think_revisited___a_cz-lib.org_.pdf (дата звернення: 14.04.2025).

15. Investopedia. Chatbot Definition, Types, Pros & Cons, Examples. [Електронний ресурс]. Режим доступу: https://www.investopedia.com/terms/c/chatbot.asp (дата звернення: 06.05.2025).

16. Shrager, J. ELIZA Reinterpreted: The world's first chatbot was not intended as a chatbot at all. [Електронний ресурс]. Режим доступу: https://arxiv.org/abs/2406.17650 (дата звернення: 06.05.2025).

17. Khennouche, F., Elmir, Y., Djebari, N., Himeur, Y., Amira, A. Revolutionizing Customer Interactions: Insights and Challenges in Deploying ChatGPT and Generative Chatbots for FAQs. [Електронний ресурс]. Режим доступу: https://arxiv.org/abs/2311.09976 (дата звернення: 06.05.2025).

18. Rüdel, T., Leidner, J. Control in Hybrid Chatbots. [Електронний ресурс]. Режим доступу: https://arxiv.org/abs/2311.11701 (дата звернення: 06.05.2025).

19. ChatbotsPlus. Types of Chatbots: Rule-Based, AI-Powered, and Beyond. [Електронний ресурс]. Режим доступу: https://surl.li/daizpj (дата звернення: 06.05.2025).

20. Rezo.ai. Understanding Different Types of Chatbots for Your Business. [Електронний ресурс]. Режим доступу: https://www.rezo.ai/our-blogs/types-of-chatbots (дата звернення: 06.05.2025).

21. Peerdh.com. Building A Hybrid Chatbot: Merging Rule-based And AI-driven Approaches. [Електронний ресурс]. Режим доступу: https://surl.li/bykhde (дата звернення: 06.05.2025).

22. PyPI. pyTelegramBotAPI. [Електронний ресурс]. Режим доступу: https://pypi.org/project/pyTelegramBotAPI/ (дата звернення: 06.05.2025).

23. Vertsel, A., Rumiantsau, M. Hybrid LLM/Rule-based Approaches to Business Insights Generation from Structured Data. [Електронний ресурс]. Режим доступу: https://arxiv.org/abs/2404.15604 (дата звернення: 06.05.2025).

24. New Yorker. Can A.I. Treat Mental Illness? [Електронний ресурс]. Режим доступу: https://www.newyorker.com/magazine/2023/03/06/can-ai-treat-mental-illness (дата звернення: 06.05.2025).

25. Wired. A Chatbot Encouraged Him to Kill the Queen. It's Just the Beginning. [Електронний ресурс]. Режим доступу: https://www.wired.com/story/chatbot-kill-the-queen-eliza-effect (дата звернення: 06.05.2025).

26. Financial Times. AI agents: from co-pilot to autopilot. [Електронний ресурс]. Режим доступу: https://www.ft.com/content/3e862e23-6e2c-4670-a68c-e204379fe01f (дата звернення: 06.05.2025).

27. Гусєв О. В. Веб-дизайн: теорія та практика / О. В. Гусєв. Київ : Академвидав, 2020. 248 с. (дата звернення: 06.05.2025).

28. Іванова Н. М. Інтерактивні елементи на сайтах ресторанів: вплив на користувацький досвід / Н. М. Іванова. Харків : ХНУ ім. В. Н. Каразіна, 2021. 164 с. (дата звернення: 06.05.2025).

29. Петров А. І. Адаптивний дизайн у веб-розробці: підходи та інструменти / А. І. Петров. Львів : Видавництво Львівської політехніки, 2022. 192 с. (дата звернення: 06.05.2025).

30. Сидоренко В. С. SEO-оптимізація вебсайтів ресторанів: стратегії та інструменти / В. С. Сидоренко. Одеса : ОНУ ім. І. І. Мечникова, 2023. 180 с. (дата звернення: 06.05.2025).

31. Тимошенко О. В. Безпека вебсайтів у галузі громадського харчування / О. В. Тимошенко. Київ : КНЕУ, 2024. 206 с. (дата звернення: 06.05.2025).

32. Sekulic, N. (2022). Restaurant Website UX Design Trends. Loop11. [Електронний ресурс]. Режим доступу: https://www.loop11.com/restaurant-website-ux-design-trends/?utm_source=chatgpt.com (дата звернення: 06.05.2025).

33. O'Donnell, K. (2022). Web Design for Restaurants & Bars: Keep These 9 Best Practices in Mind. SFGate. [Електронний ресурс]. Режим доступу: https://marketing.sfgate.com/blog/web-design-tips-for-restaurants-and-barsmarketing.sfgate.com+1SFGATE+1 (дата звернення: 06.05.2025).

34. Currington, E. (2023). The Secret Sauce of Amazing Restaurant Website Design. The Digital Restaurant. [Електронний ресурс]. Режим доступу: https://thedigitalrestaurant.com/secret-sauce-of-amazing-restaurant-website-design/Цифровой Ресторан (дата звернення: 06.05.2025).

35. Weber, P., Krings, K., Schröder, L., Michel, L. K., & Ludwig, T. (2025). Rendezfood: A Design Case Study of a Conversational Location-based Approach in Restaurants. arXiv. [Електронний ресурс]. Режим доступу: https://arxiv.org/abs/2501.03862arXiv (дата звернення: 06.05.2025).

36. Currington, E. (2023). The Secret Sauce of Amazing Restaurant Website Design. The Digital Restaurant. [Електронний ресурс]. Режим доступу:

https://thedigitalrestaurant.com/secret-sauce-of-amazing-restaurant-website-design/ (дата звернення: 06.05.2025).

37. Elit-Web. Просування сайту на Wix — оптимізація та розкрутка Wix. [Електронний ресурс]. Режим доступу: https://elit-web.ua/ua/blog/seo-prodvizhenie-na-wixelit-web.ua+1AdverMedia+1 (дата звернення: 07.05.2025).

38. Advermedia. SEO просування сайту на WIX: Як розкрутити ваш сайт на Вікс? [Електронний ресурс]. Режим доступу: https://advermedia.ua/blog/prosuvannya-sajtu-na-wix/AdverMedia (дата звернення: 07.05.2025).

39. Webpromo. SEO-просування на Wix. [Електронний ресурс]. Режим доступу: https://web-promo.ua/ua/blog/seo-prosuvannya-na-wix-yak-pracyuvati-z-perevagami-ta-obmezhennyami-platformi/Webpromo (дата звернення: 07.05.2025).

40. Adwservice. Огляд конструктора сайтів Wix: переваги платформи. [Електронний ресурс]. Режим доступу: https://adwservice.com.ua/uk/ogljad-konstruktora-sajtiv-wixAdwservice (дата звернення: 07.05.2025).

41. Legit Digital. Розробка сайту на Wix - плюси та мінуси. [Електронний ресурс]. Режим доступу: https://www.legitdigital.org/post/wix-plusy-ta-minusyLegit Digital (дата звернення: 07.05.2025).

42. Kofman Marketing. Wix чи WordPress: Яку платформу вибрати для створення вебсайту? [Електронний ресурс]. Режим доступу: https://kofman-marketing.com/wix-vs-wordpress-which-one-is-better-for-seo-copy/KofmanMarketing (дата звернення: 07.05.2025).

43. Wix.com. Потужні функції для вашого сайту. [Електронний ресурс]. Режим доступу: https://uk.wix.com/features/mainwix.com (дата звернення: 07.05.2025).

44. Wix. Шаблони Wix (безкоштовні дизайни, які допоможуть вам надихнутися). [Електронний ресурс]. Режим доступу: https://www.wix.com/explore/websites?utm_source=chatgpt.com (дата звернення: 07.05.2025).

ДОДАТКИ

Додаток А

```
<!DOCTYPE html>
<html lang="ru" class="scroll">
  <head></head>
  <body class="">
    <script type="text/javascript"></script>
    <script type="wix/htmlEmbeds" id="pageHtmlEmbeds.bodyStart start"></script>
    <script type="wix/htmlEmbeds" id="pageHtmlEmbeds.bodyStart end"></script>
    <script id="wix-first-paint"></script>
    <div id="SITE_CONTAINER">
      <div id="main_MF" class="main_MF">
        <div id="SCROLL_TO_TOP" class="Vda6QZ ignore-focus SCROLL_TO_TOP" role="region" tabindex="-1" aria-label="top of page"></div>
        <button id="SKIP_TO_CONTENT_BTN" class="SKIP_TO_CONTENT_BTN UhrbPP has-custom-focus" tabindex="0">Перейти к основному контенту</button>
        <div id="WIX_ADS" class="EFlbOv czJOlz ytGGbW"></div>
        <div id="BACKGROUND_GROUP" class="backgroundGroup c1dmp BACKGROUND_GROUP"></div>
        <div id="site-root" class="overflow-x-clip-in-mobile site-root">
          <div id="masterPage" class="mesh-layout overflow-x-clip-in-mobile masterPage css-editing-scope">
            <header id="SITE_HEADER" class="xU8qfS SITE_HEADER wixui-header" tabindex="-1"></header>
            <main id="PAGES_CONTAINER" class="PAGES_CONTAINER" data-main-content="true">
              <div id="SITE_PAGES" class="JSxAz SITE_PAGES">
                <div id="SITE_PAGES_TRANSITION_GROUP" class="AnQBdR"></div>
                <div id="c1dmp" class="dBAkHl c1dmp">
```

<section id="comp-lt8r15ku" tabindex="-1" class="Oqnigsf comp-lt8r15ku wixui-section" data-block-level-container="ClassicSection">

<section id="comp-k8pz5alt" class="comp-k8pz5alt Cohh5y wixui-column-strip">

<wow-image id="img_comp-k8pz5118" class="jhvxbk RvIAhV qDLlPX yMGsuM bgImage"
data-image-info='{"containerId":"comp-k8pz5118","alignType":"center","displayMode":"fill","targetWidth":881,"targetHeight":761,"scalingMethod":"fit","imageData":{"width":6124,"height":4082,"uri":"46dc18_0998a8ce1f34483cbe31ace4f2f1608f~mv2.jpg"}}'

```
            data-image-part="BG_IMG comp-k8pz5118" data-bg-effect-name data-has-ssr-src="true">
              <img
src="https://static.wixstatic.com/media/46dc18_0998a8ce1f34483cbe31ace4f2f1608f~mv2.jpg"
               alt style="width: 881px; height: 761px; object-fit: cover; object-position: 50% 50%;" width="881"
               height="761" data-ssr-src-done="true" fetchpriority="high">
            </wow-image>
          </div>
        </div>
        <div data-mesh-id="comp-k8pz5118inlineContent" data-testid="inline-content" class="mwF7Xl"></div>
      </div>
      <div id="comp-k8pzhjdr" class="comp-k8pzhjdr YzqVVZ wixui-column-strip__column"></div>
    </div>
  </section>
 </div>
</div>
<section id="comp-lt8r15kw" tabindex="-1" class="Oqnigsf comp-lt8r15kw wixui-section" data-block-level-container="ClassicSection"></section>
<section id="comp-lt8r15kw" tabindex="-1" class="Oqnigsf comp-lt8r15kw wixui-section" data-block-level-container="ClassicSection"></section>
<section id="comp-lt8r15kw1" tabindex="-1" class="Oqnigsf comp-lt8r15kw1 wixui-section" data-block-level-container="ClassicSection">
</section>
</div>
</div>
</main>
```

<footer id="SITE_FOOTER" class="xU8fqS SITE_FOOTER wixui-footer" tabindex="-1">
</footer>

<script type="application/json" id="wix-fedops">
</script>
<script>
window.fedops = JSON.parse(document.getElementById('wix-fedops').textContent);
</script>
<script data-url="https://static.parastorage.com/services/wix-thunderbolt/dist/sendFedopsLoadStarted.inline.d72c147b.bundle.min.js"></script>
<!-- Polyfills check -->
<script></script>
<!-- initCustomElements #1 -->
<script data-url="https://static.parastorage.com/services/wix-thunderbolt/dist/fastdom.inline.fbeb2f28.bundle.min.js"></script>
<script data-url="https://static.parastorage.com/services/wix-thunderbolt/dist/custom-elements-utils.inline.90375295.bundle.min.js"></script>
<script data-url="https://static.parastorage.com/services/wix-thunderbolt/dist/initCustomElements.inline.d40a3ca5.bundle.min.js"></script>

<!-- preloading pre-scripts -->

<link href="https://siteassets.parastorage.com/pages/pages/thunderbolt?appDefinition-sha=wixsite.com.filesusr...masterPage" as="fetch" id="features_masterPage" rel="preload" crossorigin="anonymous">

<link href="https://siteassets.parastorage.com/pages/pages/thunderbolt?appDefinit...

<script id="SentryOnLoadSetup"> </script>

<script id="sentry"> </script>

<script src="https://browser.sentry-cdn.com/7.120.3/bundle.tracing.es5.min.js" crossorigin="anonymous" async></script>

<script data-url="https://static.parastorage.com/services/wix-thunderbolt/dist/externals-registry.inline.a7ae621e.bundle.min.js"></script>

<script type="application/json" id="wix-viewer-model"> {}</script>

<script type="application/json" id="used-platform-apis-data">["location","window","site","seo","user"]</script>

<script data-url="https://static.parastorage.com/services/wix-thunderbolt/dist/wixDropdownWrapper.inline.3e467f13.bundle.min.js"></script>

<script data-url="https://static.parastorage.com/services/wix-thunderbolt/dist/createPlatformWorker.inline.a6c9f5b0.bundle.min.js"></script>

<script data-url="https://static.parastorage.com/services/wix-thunderbolt/dist/windoidMessageRegister.inline.f24f8210.bundle.min.js"></script>

<script data-url="https://static.parastorage.com/services/wix-thunderbolt/dist/thunderbolt-commons.s14f4271.bundle.min.js"></script>

<script data-url="https://static.parastorage.com/services/wix-thunderbolt/dist/bi-common.inline.2733998a.bundle.min.js"></script>

<script data-url="https://static.parastorage.com/services/wix-thunderbolt/dist/main.rendereer.a2062f68.bundle.min.js"></script>

<script async src="https://static.parastorage.com/unpkg/lodash@4.17.21/lodash.min.js"></script>

<script async src="https://static.parastorage.com/unpkg/react@18.3.1/umd/react.production.min.js"></script>

<!-- lodash script -->

<script async onload="resolveExternalsRegistryModule('lodash')" src="https://static.parastorage.com/unpkg/lodash@4.17.21/lodash.min.js"></script>

<!-- react -->

<script crossorigin onload="resolveExternalsRegistryModule('react')" src="https://static.parastorage.com/unpkg/react@18.3.1/umd/react.production.min.js"></script>

<!-- react-dom -->

<script crossorigin defer onload="resolveExternalsRegistryModule('reactDOM')" src="https://static.parastorage.com/unpkg/react-dom@18.3.1/umd/react-dom.production.min.js"></script>

<!-- Browsers Deprecation -->

<script async src="https://static.parastorage.com/services/wix-thunderbolt/dist/browser-deprecation.bundle.es5.js"></script>

<script> window.clientSideRender = false; </script>

<script data-url="https://static.parastorage.com/services/wix-thunderbolt/dist/bi.inline.bee051e0.bundle.min.js"></script>

<script data-url="https://static.parastorage.com/services/wix-thunderbolt/dist/sendBeat12.inline.6e2b716d.bundle.min.js"></script>

<script defer src="https://static.parastorage.com/services/tag-manager-client/1.920.0/siteTags.bundle.min.js"></script>

<!--pageHtmlEmbeds.bodyEnd start-->

<script type="wix/htmlEmbeds" id="pageHtmlEmbeds.bodyEnd start"></script>

<script type="wix/htmlEmbeds" id="pageHtmlEmbeds.bodyEnd end"></script>

<!--pageHtmlEmbeds.bodyEnd end-->

<!-- warmup data start -->

<script type="application/json" id="wix-warmup-data">{}</script>

<!-- warmup data end -->

</body>

</html>

Додаток Б

```
import telebot
from telebot import types
from datetime import datetime, timedelta
import threading
# Токен від BotFather
TOKEN = '8166849860:AAGEHQ88KG8khuSxXzV7-JXtC7IWkTYVXFI'
bot = telebot.TeleBot(TOKEN)
# Словник для зберігання бронювань користувачів
reservations = {}
# Словник для зберігання непідтверджених бронювань
pending_reservations = {}
@bot.message_handler(commands=['start'])
def start(message):
    markup = types.ReplyKeyboardMarkup(resize_keyboard=True)
    btn1 = types.KeyboardButton("Забронювати столик")
    btn2 = types.KeyboardButton("Моє бронювання")
    markup.add(btn1, btn2)
    bot.send_message(message.chat.id, " Вітаємо в ресторані! Що бажаєте зробити?",
reply_markup=markup)
@bot.message_handler(func=lambda message: message.text == "Забронювати столик")
def reserve_table(message):
    msg = bot.send_message(message.chat.id, " На яку дату ви хочете забронювати
столик? (Формат: ДД.ММ.РРРР)")
    bot.register_next_step_handler(msg, get_date)
def get_date(message):
    date_text = message.text
    try:
        date_obj = datetime.strptime(date_text, "%d.%m.%Y")
        if date_obj.date() < datetime.now().date():
```

```
            bot.send_message(message.chat.id, " Ви не можете забронювати столик на минулий день. Спробуйте ще раз.")
            return reserve_table(message)
        else:
            msg = bot.send_message(message.chat.id, " На яку годину? (Формат: ГГ:ХХ)")
            bot.register_next_step_handler(msg, get_time, date_text)
    except ValueError:
        bot.send_message(message.chat.id, " Невірний формат дати. Спробуйте ще раз.")
        reserve_table(message)
def get_time(message, date):
    time_text = message.text
    try:
        datetime.strptime(time_text, "%H:%M")
        msg = bot.send_message(message.chat.id, " На скільки людей?")
        bot.register_next_step_handler(msg, confirm_reservation, date, time_text)
    except ValueError:
        bot.send_message(message.chat.id, " Невірний формат часу. Спробуйте ще раз (Формат: ГГ:ХХ)")
        msg = bot.send_message(message.chat.id, " На яку годину? (Формат: ГГ:ХХ)")
        bot.register_next_step_handler(msg, get_time, date)
def confirm_reservation(message, date, time):
    people = message.text
    markup = types.InlineKeyboardMarkup()
    callback_data = f"confirm|{date}|{time}|{people}"
    btn_yes = types.InlineKeyboardButton(text="Підтвердити", callback_data=callback_data)
    btn_no = types.InlineKeyboardButton(text="Скасувати", callback_data="cancel")
    markup.add(btn_yes, btn_no)
    sent = bot.send_message(message.chat.id, f" Ви хочете забронювати столик:\n Дата: {date}\n Час: {time}\n Кількість людей: {people}\nПідтвердити?",
```

```
reply_markup=markup)
    # Зберігаєм в очікування
    pending_reservations[message.chat.id] = sent.message_id
    # Встановлюємо таймер на 2 хвилини
    timer = threading.Timer(120.0, auto_cancel, args=[message.chat.id, sent.message_id])
    timer.start()
def auto_cancel(chat_id, message_id):
    if chat_id in pending_reservations and pending_reservations[chat_id] == message_id:
        bot.edit_message_text(chat_id=chat_id, message_id=message_id, text=" Час на
підтвердження вийшов. Бронювання скасовано.")
        pending_reservations.pop(chat_id, None)
@bot.callback_query_handler(func=lambda call: call.data.startswith('confirm') or
call.data == 'cancel')
def callback_confirm(call):
    pending_reservations.pop(call.message.chat.id, None)
    if call.data == 'cancel':
        bot.edit_message_text(chat_id=call.message.chat.id,
message_id=call.message.message_id, text=" Бронювання скасовано.")
    else:
        try:
            _, date, time, people = call.data.split("|")
            reservations[call.message.chat.id] = {
                'date': date,
                'time': time,
                'people': people
            }
            bot.edit_message_text(chat_id=call.message.chat.id,
message_id=call.message.message_id, text=f" Бронювання підтверджено!\n Дата:
{date}\n Час: {time}\n Кількість людей: {people}")
        except Exception as e:
```

```
        bot.send_message(call.message.chat.id, " Сталася помилка при підтвердженні бронювання. Спробуйте ще раз.")
@bot.message_handler(func=lambda message: message.text == "Моє бронювання")
def my_reservation(message):
    res = reservations.get(message.chat.id)
    if res:
        bot.send_message(message.chat.id, f" Ваше бронювання:\n Дата: {res['date']}\n Час: {res['time']}\n Кількість людей: {res['people']}")
    else:
        bot.send_message(message.chat.id, " У вас немає бронювань.")
bot.polling(none_stop=True)
```