

MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE

Sumy State University

Academic and Research Institute of Business, Economics and Management

(full name of the institute/faculty)

Economic Cybernetics Department

(full name of the department)

«Admitted to the defense»

Head of Department

_____ Vitaliia KOIBICHUK

(signature)

(First and LAST NAME)

_____ 20__ p.

QUALIFICATION WORK

to obtain an educational degree _____ bachelor

(bachelor / master)

from the specialty 051 Economics _____,

(code and name)

educational-professional _____ programs _____ Economic Cybernetics

(educational-professional / educational-scientific)

(the name of the program)

on the topic: Development of a Web-Based System for Reserving Tables in a Restaurant

Winner(s) of the group EK-91a

(group cipher)

_____ Chevhuza Karina Maximivna

(full name)

The qualification work contains the results of own research. The use of ideas, results and texts of other authors are linked to the corresponding source

_____ Karina CHEVHUZ

(signature)

(Name and SURNAME of the acquirer)

Head

docent, candidate of technical sciences

Kostiantyn HRYTSENKO

(position, academic degree, academic title, Name and SURNAME)

(signature)

Consultant _____

(position, academic degree, academic title, Name and SURNAME)

(signature)

Міністерство освіти і науки України
Сумський державний університет
Навчально-науковий інститут бізнесу, економіки та менеджменту
Кафедра економічної кібернетики

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

на тему «РОЗРОБКА WEBОРІЄНТОВАНОЇ СИСТЕМИ ДЛЯ БРОНЮВАННЯ
СТОЛИКІВ У РЕСТОРАНІ»

Виконала студентка 4 курсу групи ЕК-91а
(номер курсу) (шифр групи)

Спеціальності 051 «Економіка»
(Економічна кібернетика)

Чевгуз К.М.
(прізвище, ініціали студента)

Керівник

доцент, к.т.н. Гриценко К.Г.
(посада, науковий ступінь, прізвище, ініціали)

Суми – 2023 рік

АНОТАЦІЯ

кваліфікаційної роботи бакалавра на тему
«РОЗРОБКА ВЕБОРІЄНТОВАНОЇ СИСТЕМИ ДЛЯ БРОНЮВАННЯ
СТОЛИКІВ У РЕСТОРАНІ»

студентки Чевгуз Каріни Максимівни

(прізвище, ім'я, по-батькові)

Ресторанна галузь України пройшла великі зміни протягом багатьох років завдяки впровадженню технологій та Інтернету, які відіграють ключову роль у вдосконаленні роботи ресторанів. Актуальність теми, яку я обрала для дослідження, пояснюється однією з найвидатніших трансформацій: перехід від традиційних методів бронювання столиків в ресторані до онлайн-систем. Проаналізувавши статистику та конкурентність даного бізнесу, можна дійти висновку, що ринок не є переповненим, а отже існує велика вірогідність успіху даного виду діяльності. Таким чином, розробка сайту є важливим і актуальним рішенням, яке не тільки полегшить процес бронювання, а й привабить більшу кількість клієнтів. Для власників ресторанів стало простіше керувати бронюванням, оптимізувати розподіл столиків та підвищувати загальну ефективність роботи.

Метою кваліфікаційної роботи є розробка веборієнтованої системи для бронювання столиків у ресторані.

Об'єктом дослідження є процес бронювання столиків у ресторані.

Предметом дослідження є сучасні програмні засоби, моделі та технології створення вебсайту у якості веборієнтованої системи для бронювання столиків у ресторані.

Задачами є аналізу стану автоматизації процесу бронювання столиків в ресторанах; моделювання бізнес-процесу, розробка прототипу сайту бронювання столиків в ресторанах.

Для досягнення поставленої мети було проаналізовано матеріали наукових досліджень, присвячених питанням створення інформаційних систем онлайн бронювання, а також існуючих програмних рішень в даній галузі.

Інформаційною базою кваліфікаційної роботи є наукові дослідження, присвячені питанням процесу бронювання в ресторанах; існуючі програмні рішення в даній галузі.

Основним науковим результатом кваліфікаційної роботи є розробка прототипу сайту.

Отримані результати можуть бути використані для аналізу процесів бронювання, а також використання веборієнтованих систем у якості покращення ефективності ресторанної галузі.

Ключові слова: бронювання столиків, моделювання бізнес-процесів, веборієтована система, автоматизація.

Зміст кваліфікаційної роботи викладено на 25 сторінках. Список використаних джерел із 40 найменувань, розміщений на 4 сторінках. Робота містить 1 таблицю, 22 рисунки, а також 3 додатки, розміщених на 14 сторінках.

Рік виконання кваліфікаційної роботи – 2023 рік.

Рік захисту роботи – 2023 рік.

Міністерство освіти і науки України
Сумський державний університет
Навчально-науковий інститут бізнесу, економіки та менеджменту
Кафедра економічної кібернетики

ЗАТВЕРДЖУЮ
Завідувач кафедри
к.е.н., доцент
_____ В.В. Койбічук
“ ___ ” _____ 2023 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА
спеціальність 051 «Економіка (Економічна кібернетика)»
студенту IV курсу, групи EK-91a

Чевгуз Каріні Максимівні

(прізвище, ім'я, по батькові студента)

1. Тема роботи Розробка веборієнтованої системи для бронювання столиків у ресторані
2. Затверджена наказом по університету від «23» травня 2023 року № 0554-VI
3. Термін подання студентом закінченої роботи «16» червня 2023 року
4. Мета кваліфікаційної роботи створення веборієнтованої системи, яка дозволить автоматизувати процес бронювання столиків у ресторані
5. Об'єкт дослідження процес бронювання столиків у ресторані
6. Предмет дослідження сучасні програмні засоби, моделі та технології створення вебсайту у якості веборієнтованої системи для бронювання столиків у ресторані
7. Кваліфікаційна робота виконується на матеріалах наукових досліджень, присвячених питанням створення інформаційних систем онлайн бронювання; існуючих програмних рішень в даній галузі.
7. Орієнтовний план кваліфікаційної роботи, терміни подання розділів керівникові та зміст завдань для виконання поставленої мети

Розділ 1. Дослідження стану автоматизації процесу бронювання столиків у ресторані та проектування веборієнтованої системи – 23 травня 2023 року

(назва – термін подання)

У розділі 1 проаналізувати стан автоматизації процесу бронювання столиків у ресторані, сформулювати вимоги до веборієнтованої системи, розробити моделі процесу у нотаціях IDEF0, IDEF3 і DFD, описати архітектуру веборієнтованої системи, її функціональну структуру, підсистеми забезпечення функціональної частини та технології вирішення поставлених завдань.

Розділ 2. Реалізація прототипу веборієнтованої системи – 3 червня 2023 року
(назва – термін подання)

У розділі 2 розглянути структуру та особливості реалізації інформаційного та алгоритмічного забезпечення. Навести інструкцію по використанню.

(зміст конкретних завдань до розділу, які повинен виконати студент)

8. Консультації з роботи:

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1			
2			
3			

9. Дата видачі завдання: «03» квітня 2023 року

Керівник кваліфікаційної роботи

(підпис)

К.Г. Гриценко

(ініціали, прізвище)

Завдання до виконання одержав

(підпис)

К.М. Чевгуз

(ініціали, прізвище)

ЗМІСТ

ВСТУП	3
1. ДОСЛІДЖЕННЯ СТАНУ АВТОМАТИЗАЦІЇ ПРОЦЕСУ БРОНЮВАННЯ СТОЛИКІВ У РЕСТОРАНІ ТА ПРОЄКТУВАННЯ ВЕБОРІЄНТОВАНОЇ СИСТЕМИ	5
1.1. Аналіз стану автоматизації процесу бронювання столиків у ресторані....	5
1.2 Вимоги до веборієнтованої системи	6
1.3 Моделювання бізнес-процесу	11
2. РЕАЛІЗАЦІЯ ПРОТОТИПУ ВЕБОРІЄНТОВАНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ	16
2.1 Структура та особливості реалізації інформаційного та алгоритмічного забезпечення	16
2.2 Контрольний приклад та інструкції по використанню	23
ВИСНОВОК.....	27
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	29
ДОДАТКИ	33

ВСТУП

У сучасному цифровому світі технології здійснюють значний вплив на різні аспекти нашого життя, включаючи галузь ресторанного бізнесу. Одним із важливих аспектів цього сектору є процес бронювання столиків, який зазнає перетворень завдяки новим інноваціям. Однак, виникає проблема, зумовлена особливостями способу комунікації молоді.

Сучасне покоління молоді відрізняється від попередніх поколінь у своєму способі спілкування. Замість традиційних дзвінків, юні люди віддають перевагу перепискам за допомогою месенджерів та соціальних мереж. Цей феномен стає особливо помітним у контексті бронювання столиків у ресторанах. Молодь віддає перевагу зручній та швидкій комунікації через месенджери та онлайн-платформи, що створює проблему для рестораторів, які в основному використовують телефонні дзвінки для бронювання столиків.

Важливість онлайн бронювання столиків для ресторанів полягає в тому, що вони спрямовані на молоде покоління, яке є ключовою цільовою аудиторією для багатьох закладів. Незадовільне задоволення їх вимог може призвести до втрати клієнтів і негативно вплинути на прибуток. Отже, власники ресторанів і менеджери повинні активно шукати інноваційні рішення, щоб пристосуватися до змін у способах комунікації з молодим поколінням та задовольнити їхні потреби в бронюванні столиків.

Ця кваліфікаційна робота присвячена розробці та впровадженню системи онлайн бронювання столиків через вебсайт, яка допоможе ресторанам адаптуватись до змін у комунікаційних зв'язках та задовольнити потреби молодого покоління. Використання цієї веборієнтованої системи забезпечить зручну, швидку та сучасну комунікацію між клієнтами та рестораном, що сприятиме поліпшенню їхнього досвіду та забезпеченню високоякісного обслуговування. Інтеграція цієї інформаційної технології може стати важливим кроком для ресторанів у відповідь на потреби та вимоги молодого покоління.

Метою кваліфікаційної роботи є розробка веборієнтованої системи для бронювання столиків у ресторані.

Об'єктом дослідження є процес бронювання столиків у ресторані.

Предметом дослідження є сучасні програмні засоби, моделі та технології створення вебсайту у якості веборієнтованої системи для бронювання столиків у ресторані.

Задачами є аналізу стану автоматизації процесу бронювання столиків в ресторанах; моделювання бізнес-процесу, розробка прототипу сайту бронювання столиків в ресторанах.

Для досягнення поставленої мети було проаналізовано матеріали наукових досліджень, присвячених питанням створення інформаційних систем онлайн бронювання, а також існуючих програмних рішень в даній галузі.

1. ДОСЛІДЖЕННЯ СТАНУ АВТОМАТИЗАЦІЇ ПРОЦЕСУ БРОНЮВАННЯ СТОЛИКІВ У РЕСТОРАНІ ТА ПРОЄКТУВАННЯ ВЕБОРІЄНТОВАНОЇ СИСТЕМИ

1.1. Аналіз стану автоматизації процесу бронювання столиків у ресторані

Відповідно до тенденцій у ресторанній індустрії, автоматизація стає все більшою необхідністю, оскільки дефіцит робочої сили стає критичною проблемою, що спонукає галузь рухатися в цьому напрямку [2]. Прогнозується, що в найближчі роки приблизно половина всіх ресторанів у Сполучених Штатах мають намір використовувати автоматизаційні технології для подолання проблеми нестачі робочої сили, а 87% операторів промисловості та ресторанів вважають впровадження ресторанних технологій необхідним для свого виживання [22].

Одним з аспектів автоматизації є системи бронювання столиків через онлайн-платформи. Використання таких систем допомагає зменшити проблему неприбуття клієнтів та порожніх місць [4]. Більш як 80% ресторанів вже використовують онлайн-замовлення та програмне забезпечення для бронювання, що допомагає їм оптимізувати свої бізнес-процеси, і 95% з них повідомляють про зростання доходів після впровадження цих технологій. [9].

Такі системи бронювання мають ряд особливостей, включаючи безпечні варіанти оплати, зручний інтерфейс, зрозумілий календар, можливості попереднього замовлення, візуально привабливе меню з фотографіями, а також знижки та ваучери [11]. Сучасні онлайн-платформи для бронювання столиків мають простий у використанні інтерфейс та забезпечують ефективне керування столиками, що дарує клієнтам безпроблемний досвід [28].

Окрім бронювання, ресторани також можуть використовувати автоматизацію для обробки платежів клієнтів за допомогою цифрових платіжних систем. [36]. Інструменти автоматизації охоплюють широкий

спектр – від мобільних систем торгових точок і QR-кодів в меню до кіосків самозамовлення та цифрових списків очікування [33]. Незважаючи на те, що деякі з цих технологій можуть бути витратними, нові розробки та поліпшення роблять їх більш доступними та ефективними [31].

Україна, як країна, яка активно відстежує оновлення в галузі IT, не залишається осторонь. На сьогодні відомо 2 додатки для бронювання: RSRVIT APP, який має три основні розділи: ресторани, пошук і бронювання, та Resy, який є світовим партнером у сфері ресторанних технологій і пропонує інноваційне програмне забезпечення для бронювання [36]. Resy використовує Stripe як платіжну платформу, що спрощує процес повернення коштів та зменшує кількість випадків, коли клієнти не приходять. У додаток до цього, Resy дозволяє ресторанам отримувати доступ до бізнес-інформації, клієнтських даних і опитувань, а також допомагає у вдосконаленні організації розташування столів і зменшенні кількості випадків, коли клієнти не з'являються. [35].

Тож зараз, в Україні доступні програмне забезпечення для бронювання столиків для ресторанів і платформи для бронювання зручностей для готелів, які забезпечують точність, скорочують час простою та покращують враження від гостей [8].

1.2 Вимоги до веборієнтованої системи

Оскільки всі вищезазначені рішення для бронювання столиків в Україні створені для великих міст, я поставила за мету створити веборієнтовану систему для закладів м.Суми. Назва мого майбутнього сайту «Uti», яка є легкою для запам'ятовування.



Рис. 1 – Логотип для вебсайту

Ефективність будь-якої бізнес-дії залежить від конкретно поставленої мети, до якої вона спрямована. Без чітко визначених цілей, бізнес не може існувати. У контексті бізнесу, вебсайт є необхідним інструментом, який потребує визначення своїх власних цілей, що залежать від характеру компанії, її продукту і стратегії [29]. Основна ціль розробки вебсайту УТІ полягає в упрощенні процесу бронювання столиків в бажаному місці за кілька хвилин без необхідності телефонувати. Однак, бронювання – це не єдина функція сайту. Додатковими цілями можуть бути:

- ознайомлення користувачів із закладами харчування в м.Суми;
- залучення нових клієнтів або партнерів;
- стимулювання повторних покупок;
- підвищення рівня задоволеності та утримання клієнтів.

Розроблюваний сайт буде представлений у формі каталогу – самостійного ресурсу, де будуть розміщені всі заклади з детальним описом, меню, оцінками та коментарями відвідувачів. Таким чином, клієнт матиме можливість знайти відповіді на свої питання та зручно забронювати місце, яке йому сподобалось [15].

Під час здійснення бізнесу через Інтернет важливо враховувати вимоги законодавства та надавати на своєму вебсайті необхідну інформацію, що відповідає цим вимогам. Вхідною інформацією для веборієнтованої системи буде:

- нормативно-правова база України, яка повинна бути доступна для ознайомлення на вебсайті, а саме: закони, постанови, накази або інші нормативні акти, що стосуються здійснення ресторанного бізнесу.
- інформація про заклад, включаючи юридичний статус компанії і її адресу. Ця інформація повинна бути чітко вказана на вебсайті, щоб користувачі могли легко знайти ці дані та перевірити достовірність компанії.
- документи, що регулюють відносини між рестораном і відвідувачем, а саме: правила та обмеження, які відносяться до використання

вебсайту та замовлення послуг ресторану, політика конфіденційності. Сайт повинен мати політику конфіденційності, яка пояснює, як ресторан збирає, використовує та захищає особисті дані клієнтів. Одним з головних документів для закладів харчування є ліцензії та дозволи. Якщо ресторан працює з алкоголем або має спеціалізовані послуги, такі як кейтеринг, можуть знадобитися копії ліцензій або дозволів, що підтверджують право провадити ці види діяльності [16].

Основні користувачі веборієнтованої системи ресторану включають нових та потенційних відвідувачів, а також працівників, таких як адміністратори закладу та офіціанти. Повний доступ до програмної частини сайту надається лише адміністраторам.

Однією з основних функціональних частин веборієнтованої системи ресторану є власне бронювання столиків. У нього повинен бути сучасний дизайн, зрозумілий інтерфейс і простий процес обрання ресторану. На кожній сторінці має міститись коротка інформація про заклад: назва, адреса, контактні номери, посилання на соціальні мережі, вид кухні. Інформація доповнюється якісними фотографіями залу. На платформі мають бути прикріплені відгуки відвідувачів з фотографіями. Оскільки основною метою створення цієї платформи є спрощення процесу бронювання, на ній повинен бути показаний план розташування столиків, щоб клієнт міг обрати вільний столик, який йому подобається, та замовити страви заздалегідь. Також важливо забезпечити зручність оформлення замовлення: занадто багато кроків або полів, які потрібно заповнити, можуть здатися незручними для потенційних клієнтів і змусити їх передумати [27].

В Сумах до сьогоднішнього дня відсутній сайт, який би надавав можливість здійснювати онлайн-бронювання, що полегшує досягнення успіху на місцевому ринку. Приклади вебсайтів зі схожою функціональністю в Україні та за кордоном представлено на рисунках Б.1 та Б.2 додатка А. Обидва веб-сайти являють собою каталоги закладів, з детальною інформацією та можливістю замовлення.

Після визначення нового продукту, послуги, процесу або системи наступним етапом є встановлення вимог, які включають конкретні функціональні або нефункціональні характеристики [14].

Функціональні та нефункціональні вимоги, які встановлюються під час розробки програмного забезпечення, є ключовими критеріями для визначення відповідності нового продукту очікуванням замовника. Важливість наявності чітких вимог до системи проявляється не тільки в оцінці кінцевих результатів проєкту, але й відіграє суттєву роль у процесі розробки програмного забезпечення [11].

Функціональні вимоги – це вимоги, які описують, як продукт або послуга задовольняють потреби клієнтів [**Ошибка! Источник ссылки не найден.**]. Вони включають в себе опис функцій та функціональних можливостей у різних сценаріях використання, які документують, як користувачі будуть взаємодіяти з продуктом або послугою. Таким чином, виходячи з цілей веборієнтованої системи, що розробляється, в таблиці 1.2 було складено функціональні вимоги у вигляді user story і user case [32, 13].

Таблиця 1.2 – Функціональні вимоги користувача до веборієнтованої системи бронювання столиків

№	Тип вимоги	Роль користувача	Формулювання користувацької вимоги
1	2	3	4
1	user story	visitor	Як користувач матиме можливість бачити в окремій секції на головній сторінці список закладів із описом кухні, ціновим діапазоном.
2	user story	visitor	Як користувач повинен мати можливість оформити бронь за допомогою форми.
3	user story	administrator waitress	Як користувач повинен мати авторизований доступ до адміністративної панелі управління статичним сайтом.
4	user story	administrator waitress	Як користувач повинен мати авторизований доступ до адміністративної панелі налаштування лічильників аналітики та інших інтеграцій.
5	user case	visitor	При кліку на заклад повинна відкриватись сторінка з детальною інформацією
6	user case	visitor	При натисканні на іконки фейсбуку та інстаграму повинна відкриватись сторінка закладу в обраній соцмережі.
7	user case	visitor	При заповненні форми на сторінці товару та натисканні кнопки “Забронювати” надсилається замовлення до CRM системи.
8	user case	visitor	При заповненні форми на сторінці товару та натисканні кнопки “Забронювати” відкривається сторінка, що підтверджує оформлення заявки.

User story – це короткий опис того, що користувач буде робити, коли відкриє вебсайт або буде використовувати програмне забезпечення [30]. Історія користувача зазвичай пишеться в наступному форматі: “Я, як користувач, хочу виконати певну дію, щоб досягти певного результату”.

Use case являє собою вимогу у вигляді взаємодії користувача з системою [26]. Кожен варіант використання завжди формулюється з урахуванням специфічної мети користувача. Кожен варіант використання повинен включати згадку про користувача та дієслово. Наприклад, “онлайн-відвідувач” є користувачем, а “обрати дату та час” є дієсловом, пов’язаним з цим користувачем. Таким чином, користувачі веборієнтованої системи бронювання столиків характеризуються наступними ролями:

- відвідувач (visitor);
- адміністратор (administrator);
- офіціант (waitress).

Нефункціональні вимоги допомагають програмному забезпеченню працювати ефективно [34]. Ці вимоги не є обов’язковими для системи, хоча вони зазвичай підвищують загальну якість, швидкість і ємність програмного забезпечення. Нefункціональні вимоги дозволяють користувачам використовувати певні функції програмного забезпечення, що підвищують зручність їх використання. До нефункціональних вимог належать [24,38]:

1. Масштабованість та продуктивність. Продуктивність визначає кількість запитів користувачів, які система може обробити за секунду. Масштабованість сайту визначає найвищі робочі навантаження, при яких система все ще буде відповідати стандартам продуктивності.

2. Сумісність. Сумісність системи вимірюється готовністю працювати в різних середовищах. Система повинна бути сумісною з сайтами, які використовують старі версії операційних систем, а також підтримувати кросбраузерність.

3. Безпека. Система повинна бути стійкою до шкідливого програмного забезпечення та запобігати несанкціонованому доступу.

4. Доступність та надійність. Доступність системи визначається наявністю доступу для користувачів протягом певного часового періоду. Вимога до доступності часто пов'язана з надійністю системи, що означає її здатність працювати без відмови протягом тривалого періоду.

5. Локалізація. Нефункціональна вимога локалізації описує, наскільки система адаптована до місцевих вимог користувачів. Призначена виключно для українського населення, розроблювана система повинна мати інтерфейс українською мовою [37].

1.3 Моделювання бізнес-процесу

Початковий етап проектування будь-якої інформаційної системи включає вибір методології проектування. На сьогоднішній день існує кілька стандартів для моделювання бізнес-процесів, таких як нотація управління бізнес-процесами (BPMN), нотація мови виконання бізнес-процесів (BPEL), розширений ланцюжок, керований подіями (eEPC), методологія функціонального моделювання (IDF0, IDF3) [5,6]. Кожна з цих нотацій має свої переваги для різних цілей, а також переваги використання певного програмного забезпечення для опису бізнес-процесів у відповідних нотаціях.

Методологія IDEF0 є одним із популярних підходів до опису бізнес-процесів. Вона має наступні особливості:

- використання контекстної діаграми;
- підтримка декомпозиції процесів;
- фокус на ідентифікації головних елементів;
- виділення 4 типів стрілок для показу зв'язків (вхід, вихід, механізм, управління).

Таким чином, IDEF0 є методологією функціонального моделювання та графічного опису процесів, яка призначена для формалізації та опису бізнес-процесів [2]. Характерною особливістю IDEF0 є акцент на ієрархічне представлення об'єктів, що сприяє легкому розумінню предметної області. В

рамках методології IDEF0 розглядаються логічні зв'язки між роботами, а не послідовність робіт в часі.

За допомогою методології IDEF0 я покажу бізнес-процес бронювання столиків в ресторані на сайті. Основною метою цього процесу є забезпечення клієнтам можливості зручного та ефективного бронювання столиків для відвідування ресторану. Нижче наведена контекстна діаграма процесу (рис. 2):

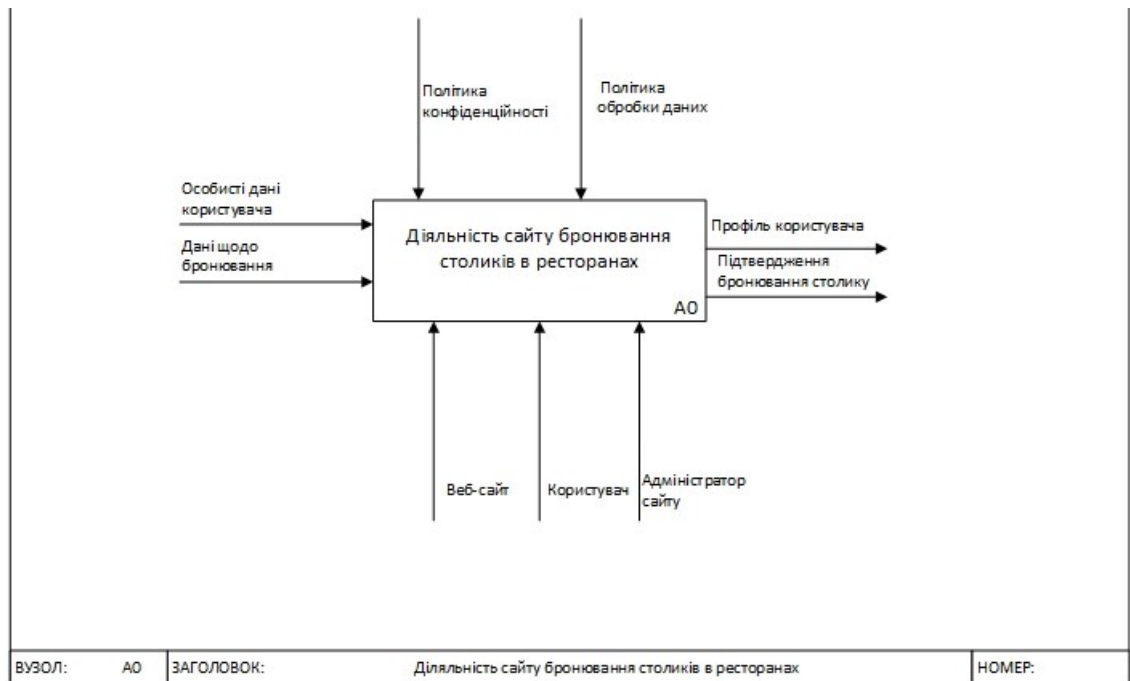


Рис. 2 – Контекстна діаграма IDEF0 верхнього рівня “Діяльність сайту бронювання столиків”

На наступній діаграмі відображено процеси, що пояснюють і демонструють усі функції системи (рис. 3)

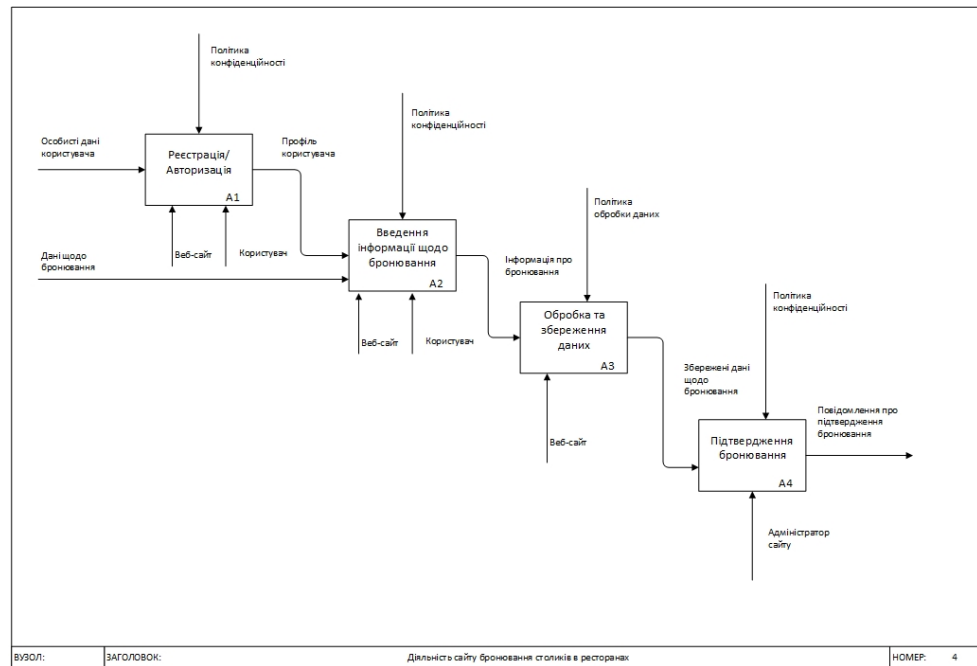


Рис. 3 - Декомпозиція контекстної діаграми роботи сайту бронювання столиків

Діаграма потоку даних (DFD) — це візуальне представлення потоку даних через систему або процес. Він включає чотири основні елементи:

- зовнішні сутності;
- процеси;
- сховища даних;
- потоки даних [3].

Метою DFD є надання інформації про входи та виходи кожної сутності та самого процесу. DFD мають кілька нотацій для їх відображення, але всі вони використовують узгоджені символи, такі як короткі текстові мітки, стрілки, кола та прямокутники для опису даних [10].

DFD розбиває процес на рівні, починаючи з загального опису системи на найвищому рівні і деталізуючи його на більш специфічні компоненти на нижчих рівнях. У цьому випадку, розглянемо 2 рівня DFD.

Рівень 0 DFD (Контекстний рівень):

На контекстному рівні DFD ми зображаємо саму систему, як один процес, з яким взаємодіють зовнішні сутності. Тут будуть наступні елементи (рис. 4):

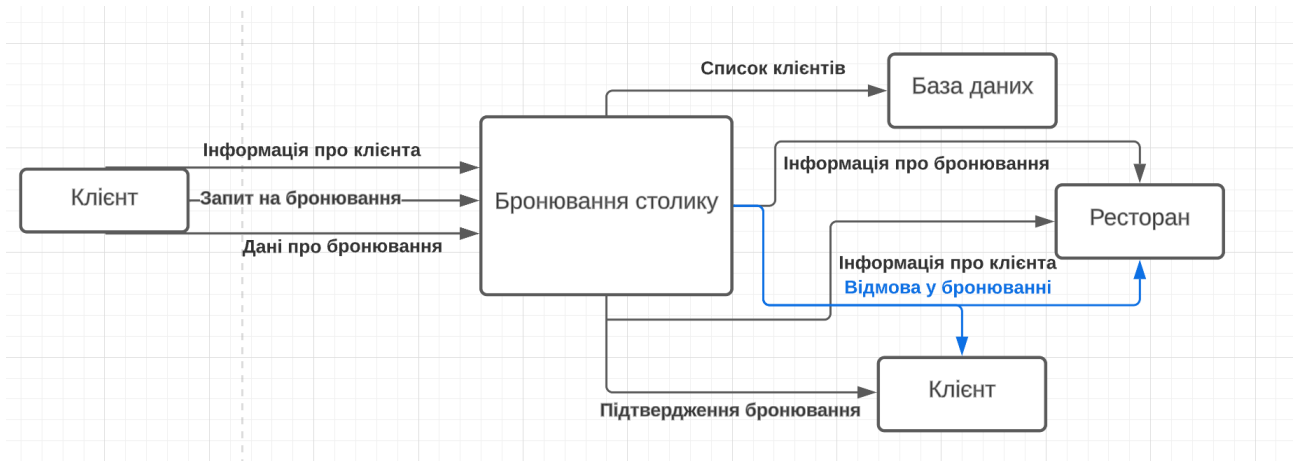


Рис. 4 - Контекстна діаграма DFD "Діяльність сайту бронювання столиків"

На рівні 1 DFD ми розкриваємо процес "Бронювання столика" на більш деталізовані компоненти. Тут будуть наступні елементи (рис.5):

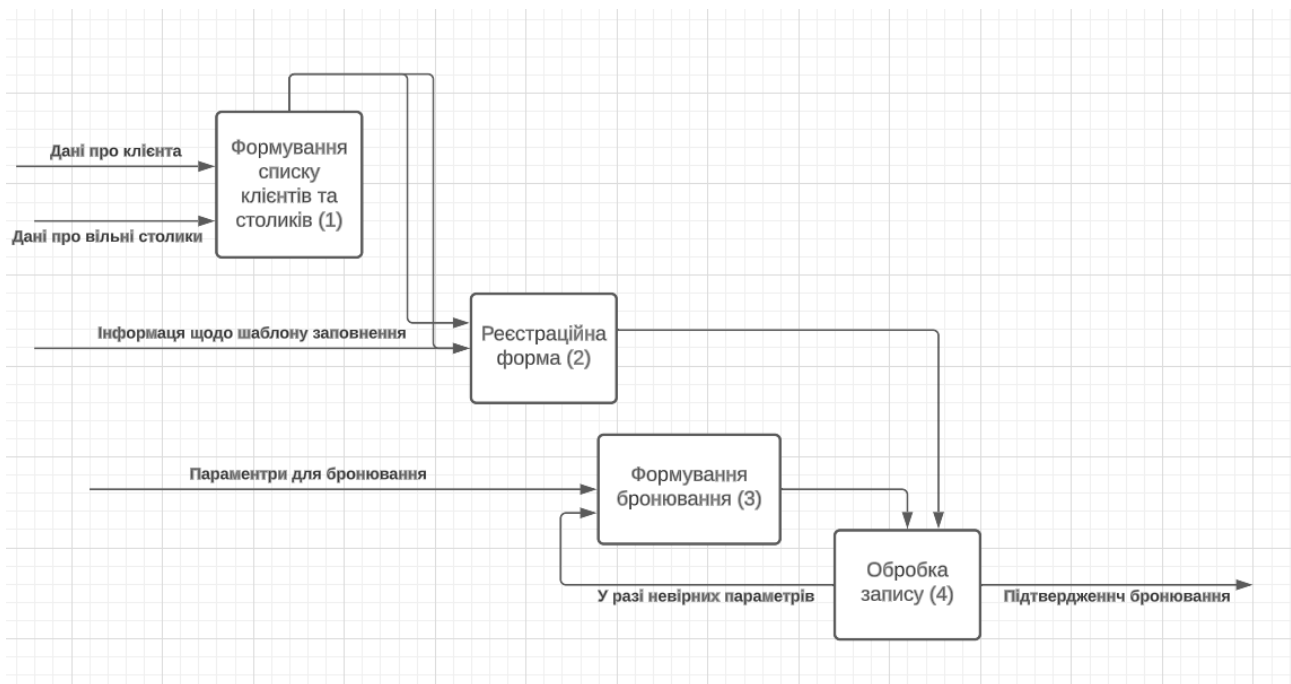


Рис. 5 - Декомпозиція контекстної діаграми роботи сайту бронювання столиків нотації DFD

Таким чином, DFD для бронювання столиків в ресторані допомагає проілюструвати потік даних та процеси, які відбуваються в системі. Він може

бути подальшим деталізованим і розширеним залежно від потреб проекту та архітектури системи [21].

2. РЕАЛІЗАЦІЯ ПРОТОТИПУ ВЕБОРІЄНТОВАНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ

2.1 Структура та особливості реалізації інформаційного та алгоритмічного забезпечення

Для забезпечення ефективної роботи веб-орієнтованої системи, яка використовується для бронювання столиків у закладах, необхідно мати відповідне алгоритмічне програмне забезпечення.

Алгоритмічне забезпечення – сукупність математичних методів, моделей і алгоритмів, які використовуються в системі для розв'язання задач і обробки інформації [37,39].

Алгоритмічне забезпечення інформаційної системи для бронювання столиків включає моделювання поведінки користувача та розробку алгоритмів для основних програмних функцій. Один з методів візуалізації цих алгоритмів є використання блок-схем. Блок-схема є графічним зображенням послідовності процесів, яке допомагає розуміти зв'язок між прийняттям рішень і кінцевим результатом. Цей інструмент може бути адаптований для різноманітних цілей і використовується для опису різних процесів, таких як виробничі, адміністративні або послугові процеси, а також для планування проектів. Блок-схеми використовують прості геометричні фігури для представлення процесів, прийняття рішень, зберігання даних та виведення результатів [40].

З урахуванням алгоритмів поведінки створюється інтерфейс користувача та програмується сама система. Якщо алгоритми розроблені правильно, то система буде працювати вдало, без помилок та добре продумано.

Приблизний алгоритм дій користувача на сайті бронювання столиків:

1. Зайти на сайт бронювання столиків
2. Ввести дату та час бронювання
3. Ввести кількість осіб
4. Вибрати меню завчасно (за бажанням)

5. Переглянути доступні варіанти столиків та їх розташування на плані ресторану
6. Вибрати бажаний столик та підтвердити бронювання
7. Заповнити контактну інформацію
8. Отримати підтвердження бронювання та інструкції щодо скасування бронювання у разі потреби.

Якщо користувач вирішує скасувати бронювання, він повинен повторити кроки 1-2 та знайти опцію відміни бронювання на сайті. Після цього йому буде надіслано підтвердження скасування бронювання.

На наведеному нижче рисунку представлено схему запропонованого алгоритму (рис. 6):

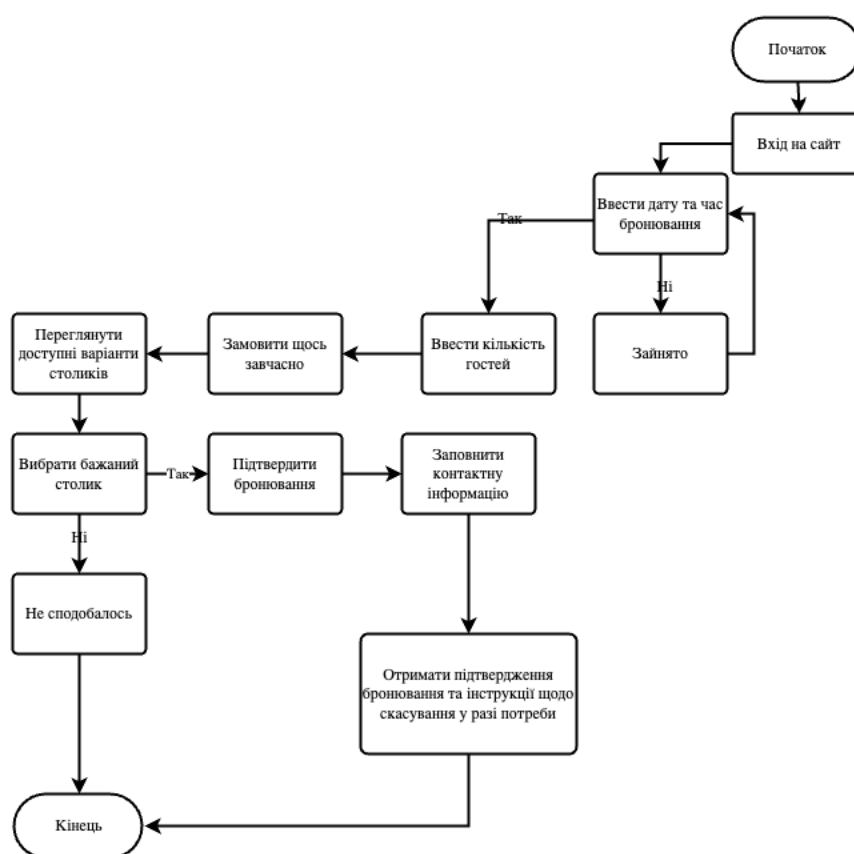


Рис. 6 – Блок-схема роботи веб-орієнтованої системи бронювання столиків

Розробку прототипу веб-орієнтованої системи для бронювання столиків я почала зі створення макету та дизайну в програмі Figma, але дозволяє експериментувати зі стилями [25]. Прототипи Figma дозволяють створювати

імітацію працюючого продукту, в якому працюють деякі елементи управління, є можливість переходити на сторінки і взаємодіяти з інтерфейсом (рис.7) [1, 17].

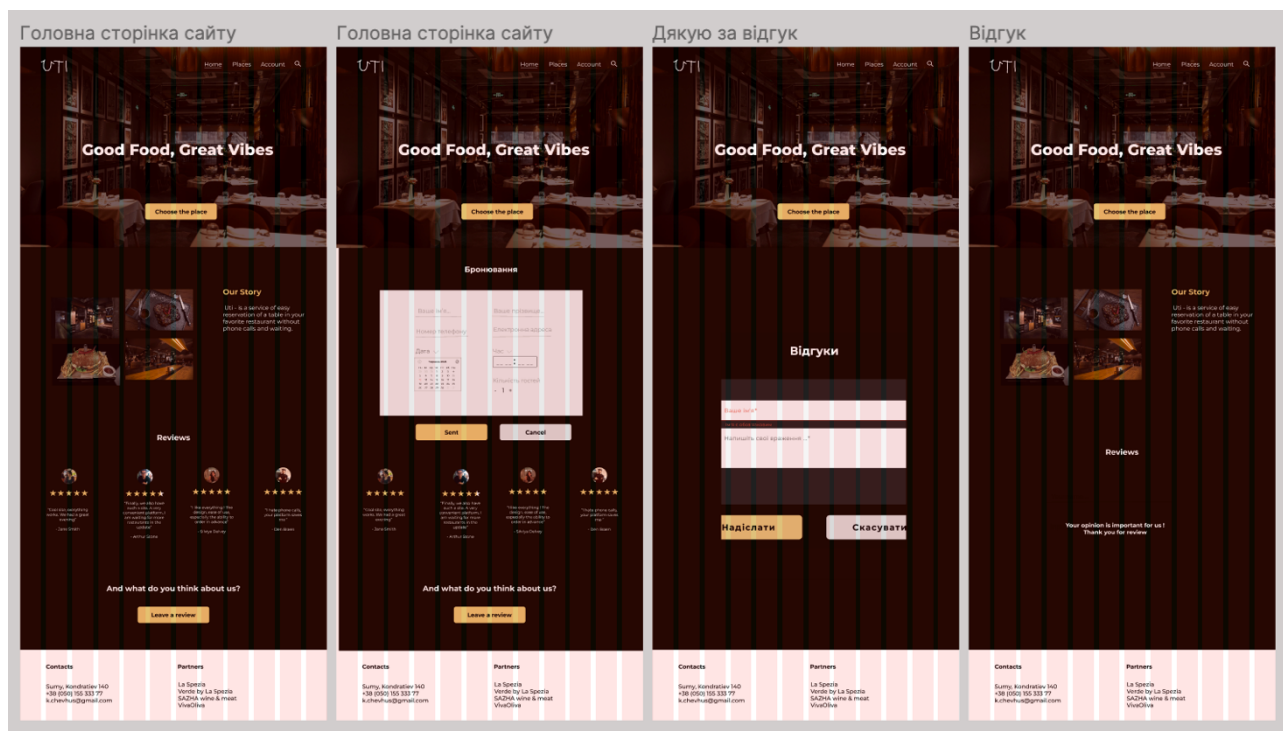


Рис. 7 – Прототип сайту, створений у Figma

Наступний етап – реалізація створеного прототипу в програмі. Для цього я використала технологію – Framework Angular, який дозволяє розроблювати веб-орієнтовані рішення, додавати на них логіку та, за необхідності, зв'язувати зі сторонніми сервісами такими як: backend, рі (рис. 8) [20].



Рис. 8 – Технологія Angular Framework

Наступна технологія - бібліотека Angular Materials, яка була використана для швидкості розробки. Вона має в собі достатню кількість готових компонентів, які легко модифікувати для логіки та стилів (рис. 9) [19].

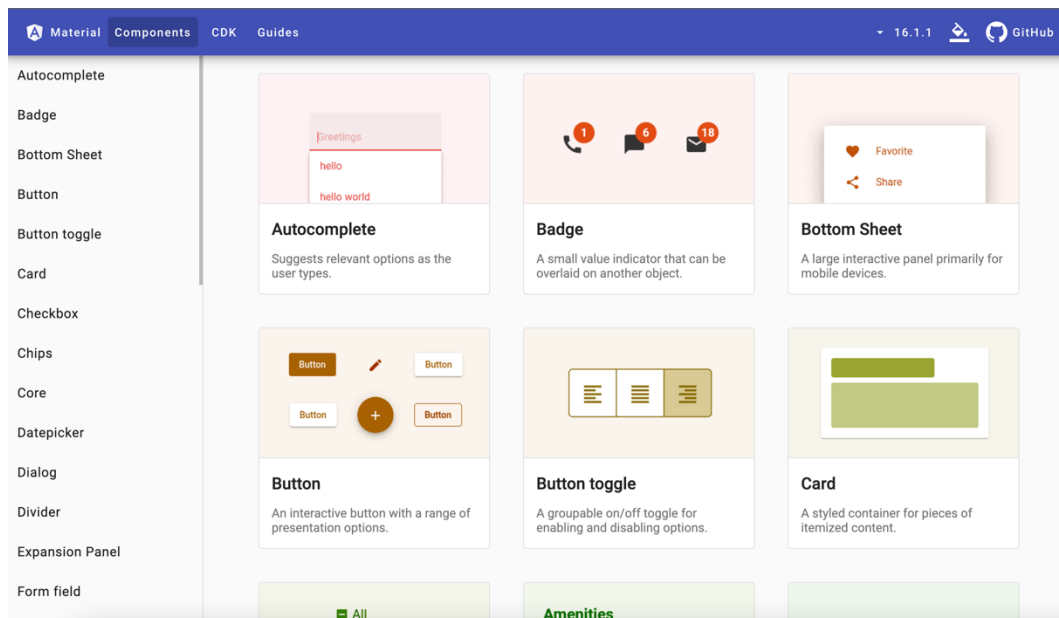


Рис. 9 – Технологія бібліотека Angular Materials

Також в Angular Materials були використані такі компоненти як кнопки, верхня шапка сайту – toolbar, форми з налаштованими валідаціями, а саме:

1. Поля імені та тексту є обов'язковими;
2. Для поля імені – мінімальна кількість символів - 3, максимальна – 5;
3. Для поля тексту – мінімум 50 [7].

При виникненні однієї з помилок під «input» ім'я буде показана помилка. Для надійності – кнопка «надіслати» є неактивною, якщо поточний стан форми не є валідним, тобто один з параметрів не відповідає вимогам валідації (рис. 10).


```

class="create-review-state" *ngSwitchCase="'Review create'">
  n [formGroup]="createReviewForm">
  iv class="form-content">
  <mat-form-field>
    <mat-label>Ваше ім'я</mat-label>
    <input matInput [formControl]="createReviewForm.controls.name">
    <mat-error *ngIf="createReviewForm.controls.name.invalid">
      {{createReviewForm.controls.name.hasError('required') ? 'Ім\`я є обов\`язковим' :
      createReviewForm.controls.name.hasError('minlength') ? 'Мінімальна довжина імені - 3' :
      createReviewForm.controls.name.hasError('maxlength') ? 'Максимальна довжина імені - 60' : null}}

```

Рис. 10 – Перевірка валідації даних

Розробка починалася зі створення шапки сайту, включаючи перший блок та блоку «Про нас», який містить фотографії та розповідь про сервіс (рис. 11, 12).

```

1 <mat-toolbar>
2   <h3>УТИ</h3>
3   <div class="nav-section">
4     <p>Головна</p>
5     <p>Міся</p>
6     <p>Профіль</p>
7     <mat-icon inline>search</mat-icon>
8   </div>
9 </mat-toolbar>
10 <div class="bd-header-img">
11   <p class="slogan">Смачна Їжа, Чудова Атмосфера</p>
12   <button mat-flat-button color="accent" class="choose-place-btn" >Забронювати місце</button>
13 </div>
14 <div class="story-section">
15   <div class="photos">
16     <img ngSrc="../assets/story-1.png" alt="Story photo" height="172" width="283">
17     <img ngSrc="../assets/story-2.png" alt="Story photo" height="172" width="283">
18     <img ngSrc="../assets/story-3.png" alt="Story photo" height="172" width="283">
19     <img ngSrc="../assets/story-4.png" alt="Story photo" height="172" width="283">
20   </div>
21   <div class="description">
22     <p>Наша історія</p>

```

Рис. 11 – Головна сторінка сайту з шапкою

```

21 <div class="description">
22   <p>Наша історія</p>
23   <p>Uti – це сервіс легкого <br> бронювання столика <br> в улюбленому ресторані <br> без телефонних д
24 </div>
25 </div>

```

Рис. 12 – Блок «Про нас»

На головній сторінці також розмістилась активна кнопка «Забронювати» (рис.11), яка відкриває форму для бронювання столика в ресторані (рис. 13).

```

<div class="book-form" #ngSwitchCase="book-place">
  <p>Бронювання</p>
  <form [formGroup]="bookPlaceForm">
    <div class="form-content">
      <mat-form-field>
        <mat-label>Ваше ім'я</mat-label>
        <input matInput [formControl]="bookPlaceForm.controls.firstName">
      </mat-form-field>
      <mat-form-field>
        <mat-label>Ваше прізвище</mat-label>
        <input matInput [formControl]="bookPlaceForm.controls.firstName">
      </mat-form-field>
      <mat-form-field>
        <mat-label>Номер телефону</mat-label>
        <input matInput [formControl]="bookPlaceForm.controls.phoneNumber">
        <mat-icon matSuffix>phone</mat-icon>
        <mat-hint>Введіть номер телефону</mat-hint>
      </mat-form-field>
      <mat-form-field>
        <mat-label>Електронна адреса</mat-label>
        <input matInput type="email" [formControl]="bookPlaceForm.controls.email">
        <mat-error #ngIf="bookPlaceForm.controls.email.hasError('email') && !bookPlaceForm.controls.em
          Введіть правильну електронну адресу

```

```

      <input matInput [matDatepicker]="picker" [formControl]="bookPlaceForm.controls.date">
      <mat-datepicker-toggle matIconSuffix [for]="picker">
        <mat-icon matDatepickerToggleIcon>keyboard_arrow_down</mat-icon>
      </mat-datepicker-toggle>
      <mat-datepicker #picker></mat-datepicker>
    </mat-form-field>
    <mat-form-field>
      <mat-label>Час</mat-label>
      <input matInput type="time" [formControl]="bookPlaceForm.controls.time">
      <mat-icon matIconSuffix>keyboard_arrow_down</mat-icon>
    </mat-form-field>
    <div class="empty"></div>
    <mat-form-field>
      <mat-label>Кількість гостей</mat-label>
      <input matInput type="number" min="1" max="12" [formControl]="bookPlaceForm.controls.guestsNum
    </mat-form-field>
  </div>

  <div class="form-actions">
    <button mat-flat-button type="submit" class="submit-btn" (click)="onBookSubmit()">Надіслати</bu
    <button mat-flat-button type="button" class="cancel-btn" (click)="onBookCancel()">Скасувати</but

```

Рис. 13 – Форма для бронювання столика в ресторані

Форма для бронювання була створена з валідаціями всіх inputs. При натисканні «Надіслати» або «Скасувати», відкриється головна сторінка.

Наступна сторінка, яка була створена – сторінка з відгуками. Для реалізації була використана директива ngSwitch, яка допомагає відобразити певний контент в залежності від значення виразу, а в нашому випадку – від значення змінної поточного стану (рис. 14) [18].

```

27 <p>Відгуки</p>
28 <div class="review-switch-container" [ngSwitch]="reviewsContainerState">
29   <div class="default-review-state" *ngSwitchCase="'Default'">
30     <div class="client-reviews">
31       <div class="client-review-item" *ngFor="let review of reviews">
32         <img [ngSrc]="review.avatarSrc" alt="Client avatar" width="65" height="65">
33         <div class="stars-rate">
34           <mat-icon inline *ngFor="let starIndex of [1,2,3,4,5]" [style.color]="review.stars >= starIn
35         </div>
36         <p class="review-text" [innerHTML]="review.text"></p>
37       </div>
38     </div>
39     <p>А що ви думаєте про нас?</p>
40     <button mat-flat-button (click)="leaveReviewClick()" >Залиште свій відгук</button>
41   </div>

```

Рис. 14 – Використання директиви ngSwitch в програмному коді

Власне на даний момент є три статуси: за замовчуванням, який відображає вже наявні відгуки на сайті (рис. 15), статус створення, де при натисканні кнопки «надіслати» буде відправлено відгук, який відображається на консолі. Після цього відкривається сторінка з повідомленням «Дякуємо за відгук» (рис. 16). При натисканні «скасувати» - буде повернено на екран за замовчуванням (рис. 15) [23].

```

26 <div class="review-section">
27   <p>Відгуки</p>
28   <div class="review-switch-container" [ngSwitch]="reviewsContainerState">
29     <div class="default-review-state" *ngSwitchCase="'Default'">
30       <div class="client-reviews">
31         <div class="client-review-item" *ngFor="let review of reviews">
32           <img [ngSrc]="review.avatarSrc" alt="Client avatar" width="65" height="65">
33           <div class="stars-rate">
34             <mat-icon inline *ngFor="let starIndex of [1,2,3,4,5]" [style.color]="review.stars >= starIndex
35           </div>
36           <p class="review-text" [innerHTML]="review.text"></p>
37         </div>
38       </div>
39       <p>А що ви думаєте про нас?</p>
40       <button mat-flat-button (click)="leaveReviewClick()" >Залиште свій відгук</button>
41     </div>

```

Рис. 15 – Блок відгуків за замовчуванням

```

    </form>
  </div>

  <div class="created-review-state" *ngSwitchCase="'Review created'">
    <p>Ваша думка важлива для нас! <br>
    Дякуємо за відгук</p>
  </div>
</div>
iv>

```

Рис. 16 – Повідомлення після відправки

2.2 Контрольний приклад та інструкції по використанню

Щоб скористатися сервісом бронювання, потрібно спочатку ввести URL-адресу: <https://uti-diploma.000webhostapp.com/>.

URL-адреса є спеціальним способом вказати унікальну адресу ресурсу [40].

Після переходу за цим посиланням відкриється головна сторінка, де знаходиться меню, пошук по сайту, кнопка та форма бронювання, вже наявні відгуки та форма для формування нових, контактна інформація, перехід на сторінку профіля (рис.17) [13].

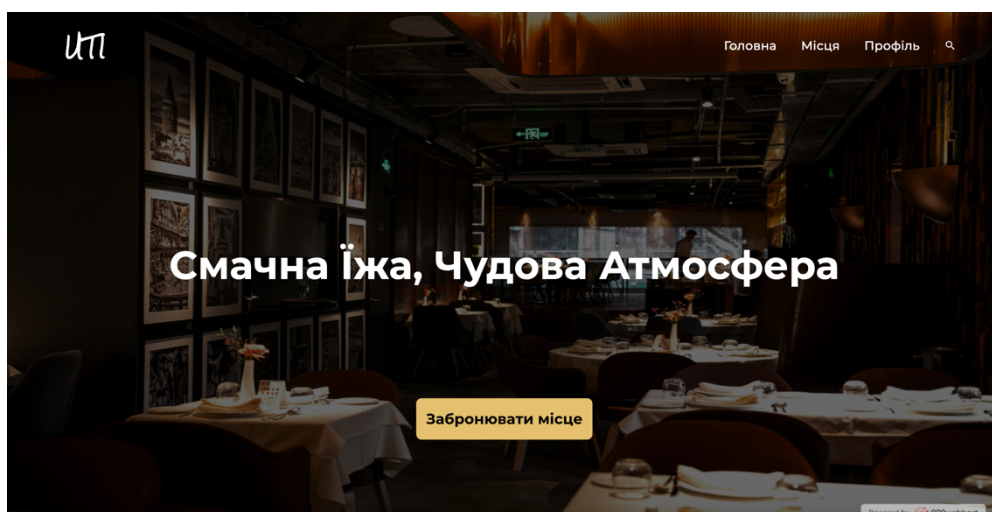


Рис. 17 – Головна сторінка сайту

На головній сторінці розташована кнопка «Забронювати». Натиснувши її, користувачу відкриється форма з бронюванням, де йому потрібно ввести особисті дані (рис. 18).

The image shows a booking form on a dark background. The form is titled "Бронювання" (Booking). It contains several input fields: "Ваше ім'я*" (Your name*) with the value "Каріна", "Ваше прізвище*" (Your surname*) with the value "Чевгуз", "Номер телефону*" (Phone number*) with the value "+38011122233", "Електронна адреса*" (Email address*) which is empty, "Дата" (Date) with a dropdown menu showing "6/22/2023", "Час" (Time) with a dropdown menu showing "17:30", and "Кількість гостей" (Number of guests) with the value "5". There are two buttons at the bottom: "Надіслати" (Send) in yellow and "Скасувати" (Cancel) in grey. To the right, a calendar widget is open, showing the month of June 2023. The date 22 is highlighted in purple. The calendar also shows the date 16 highlighted in white. Below the calendar, there is a date field showing "Дата 6/22/2023" with a dropdown arrow.

Рис. 18 – Форма для бронювання

Заповнивши форму, користувач переходить до блоку на головній сторінці, присвяченому розповіді про сам сервіс (рис. 19).

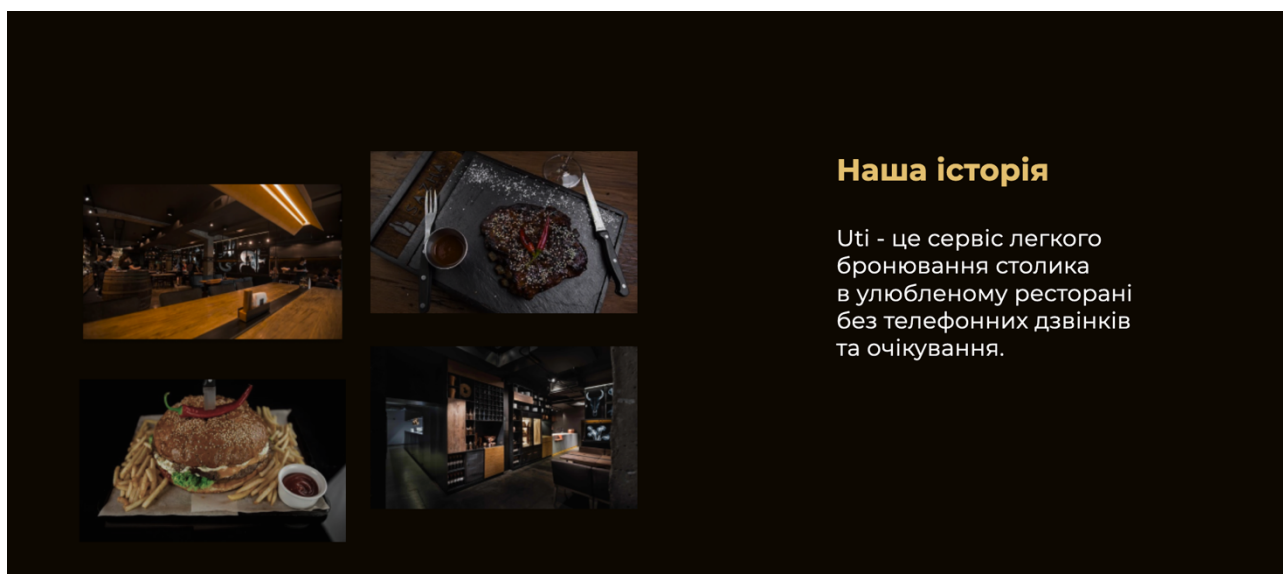


Рис. 19 – Блок «Наша історія»

Проскроливши нижче користувач переходить до блоку відгуків, він може ознайомитись з вже наявними та написати свій (рис. 20, 21, 22).

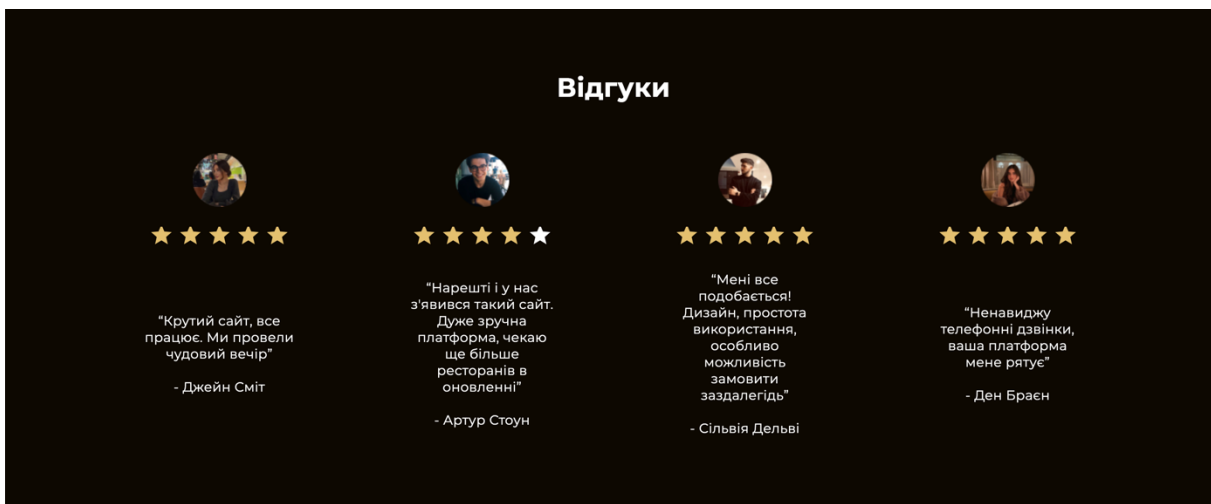


Рис. 19 – Наявні відгуки

Відгуки

Ваше ім'я*
ЕК

Мінімальна довжина імені - 3

Напишіть свої враження ...*
Каріна Чевгуз

Мінімальна довжина відгука - 50

Надіслати **Скасувати**

Рис. 20 – Створення відгуку

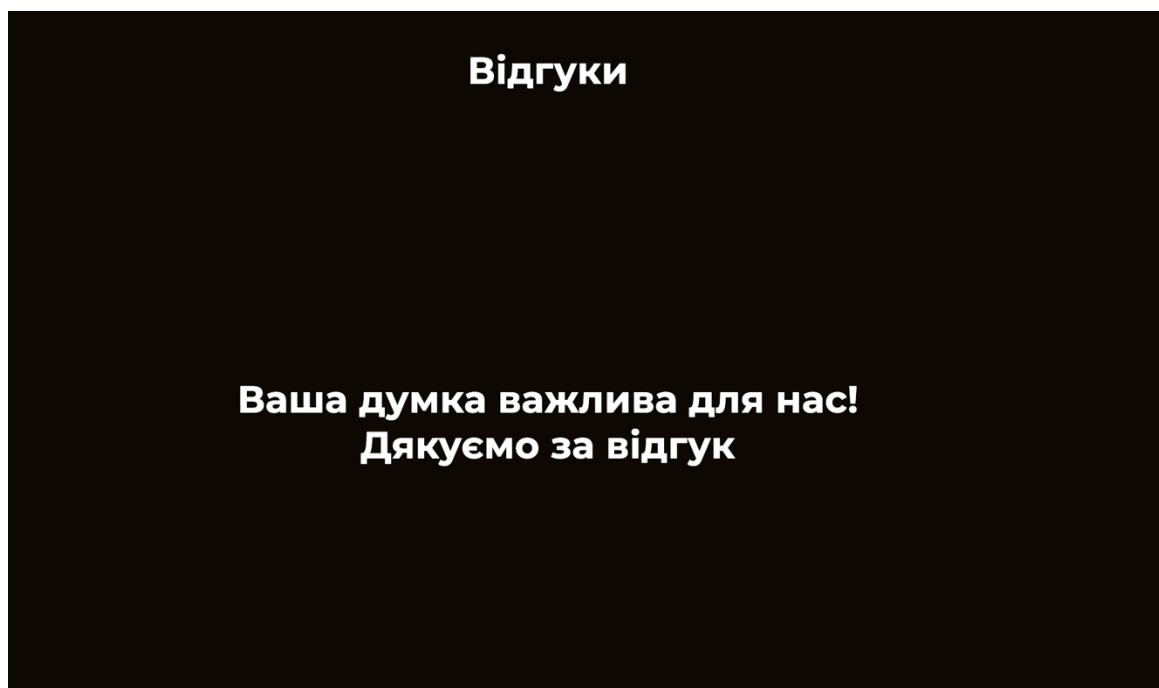


Рис. 21 – Сторінка після відправки відгуку

В кінці сайту користувач бачить контактну інформацію сервісу та список партнерів (рис. 22).

Контакти

Суми, Кондратьєва 140
+38 (050) 155 333 77
k.chevhus@gmail.com

Партнери

La Spezia
Verde by La Spezia
SAZHA wine & meat
VivaOliva

Рис. 22 – Футер сайту

ВИСНОВОК

Під час написання кваліфікаційної роботи, моя основна мета була розробити модель процесу та прототип веб-орієнтованої системи бронювання столиків у ресторані. Протягом цього часу я провів інтенсивне наукове дослідження, вивчивши ринок подібних рішень в Україні та за кордоном. Я також проаналізував статистику, яка підтверджує переваги онлайн бронювання порівняно з телефонними дзвінками.

- 9 із 10 споживачів кажуть, що хочуть, щоб компанії спілкувалися з ними за допомогою текстових повідомлень.
- 89% споживачів кажуть, що вони віддають перевагу текстовим повідомленням з бізнесом над будь-яким іншим способом спілкування.
- 63% споживачів кажуть, що перейшли б до компанії, яка пропонує текстові повідомлення
- обмін повідомленнями як канал зв'язку.
- 60% споживачів кажуть, що вони воліють записуватися на прийом для догляду за волоссям через текстове повідомлення.
- 72% кажуть, що вони б віддали перевагу бронюванню в туристичній компанії, яка надсилає їм повідомлення про плани подорожей, оновлення та нагадування.

Провівши аналіз можна дійти висновку, що покоління Z (2000 – 2002 рік народження), хто є головними респондентами та цільовою групою для бізнесу готові до автоматизації процесів, зокрема в ресторанному бізнесі. Більше того, ринок України, що пропонує сервіс бронювання столиків не є конкурентним на сьогодні, що означає легкість входження в бізнес-простір. На сьогодні в Україні діє одна компанія RestOn, що пропонує схожий сервіс для великих міст. З вищесказаного можна спрогнозувати успіх роботи веб-сайту UTI.

При розробці вебсайту було особливу увагу приділено дизайну та функціональності. Перше враження, яке відвідувач отримує від сайту, є дуже важливим, тому ми врахували різні правила вибору колірної гами та

розташування основних елементів на сторінках. Також ми дотримувалися мінімальних вимог до функціональності сайту. Виконання всіх цих правил та вимог є ключовим для успішного створення інформаційної системи.

Для створення прототипу веб-орієнтованої системи я використала Framework “Angular”, який дозволяє розроблювати веб-орієнтовані рішення, додавати на них логіку.

Описано архітектуру веб-орієнтованої системи: меню у хедері, футер, та форми зворотного зв'язку та відгуками та створений чіткий порядок дій, який утворюється в процесі обробки бронювання, головними етапами якого є заповнення клієнтом форми бронювання.

Результатом виконання індивідуального завдання стало створення і реалізація прототипу веб-орієнтованої системи для бронювання столиків в ресторанах.

Отже, після проведення всіх вище перерахованих робіт було отримано прототип веб-орієнтованої системи.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. 10 Elements of a Successful Website Homepage Design. URL: <https://themeisle.com/blog/website-homepage-design/>
2. 15+ Insightful Texting vs Calling Statistics for 2023. Cryptocurrency News, Reviews & Cryptocurrency Buying Guides | Bankless Times. URL: <https://www.banklesstimes.com/texting-vs-calling-statistics/>.
3. 54 business texting statistics you should know | Avochato. Avochato Blog | Text Messaging for Business. URL: <https://blog.avochato.com/54-text-messaging-stats-every-business-should-know-avochato>.
4. 9 Best Business Process Modeling Techniques (With Examples) - Tallyfy. Tallyfy. URL: <https://tallyfy.com/business-process-modeling-techniques/>.
5. Авраменко В. С., Авраменко А. С. Проектування інформаційних систем: навчальний посібник. Черкаси: Черкаський національний університет ім. Б. Хмельницького, 2017. 434 с.
6. Бегун А. В., Камінський О. Є. Web-програмування : навч. посіб. Київ: КНЕУ, 2011. 324 с.
7. Владилен Минин. Angular Основы. Полный Курс для начинающих, 2019. YouTube. URL: <https://www.youtube.com/watch?v=Rf54BH35yrY>.
8. Етапи створення сайту від ідеї до запуску – ITSPACE. Вебстудія ITSPACE: створення сайтів та додатків, які досягаю. URL: <https://itspace.company/blog/etapy-rozrobky-saytu-vid-ideyi-do-tekhnichnoyi-pidtrymky>.
9. Козак О. Л. Опорний конспект лекцій з курсу “Аналіз вимог до програмного забезпечення” для студентів напрямку підготовки програмна інженерія. Тернопіль, 2011. 56 с. URL: http://dspace.wunu.edu.ua/bitstream/316497/7281/1/FCIT_kKN_sPZS_dAVPZ_%20LEC.pdf.
10. Корзаченко О.В. Моделювання бізнес-процесів підприємств: методології, підходи та методи. Вісник Херсонського державного університету. 2015. Вип. 11. С. 171-175.

11. Краус К. М., Краус Н. М., Манжура О. В. Електронна комерція та Інтернет-торгівля: навчально-методичний посібник. Київ: Аграр Медіа Груп, 2021. 454 с.
12. Прототип сайту, як створення прототипу допомагає оцінити та розробити сайт. Evergreen - web розробка і діджиталізація бізнесу за допомогою AI продуктів. URL: <https://evergreens.com.ua/ua/articles/prototype-site.html>.
13. Рекомендації з управління нефункціональними вимогами. URL: <https://e5.ua/en/blogpost-2-en/rekomendatsiyi-z-upravlinnya-nefunktsionalnimi-vimogami/>.
14. Управління бізнес-процесами: Навчально-методичний комплекс дисципліни: навчальний посібник. Київ: КПІ ім. Ігоря Сікорського, 2021. 79 с.
15. Цифрова трансформація бізнесу: навіщо вона потрібна і ще 14 запитань. Главная - BusinessViews. URL: <https://businessviews.com.ua/ru/digital-transformation/id/cifrova-transformacija-biznesu-navischo-vona-potribna-i-sche-14-pitan-2046/>.
16. Цифрова трансформація: навіщо вона потрібна державі та бізнесу. Дія.Бізнес - Головна сторінка. URL: <https://business.dii.gov.ua/cases/tehnologii/cifrova-transformacia-navischo-vona-potribna-derzavi-ta-biznesu>.
17. юля і дизайн. Figma українською | Що таке Фігма і як вона працює?, 2022. YouTube. URL: <https://www.youtube.com/watch?v=cQSJrpF-t-w>.
18. Angular | Структурные директивы ngIf, ngFor, ngSwitch. METANIT.COM - Сайт о программировании. URL: <https://metanit.com/web/angular2/3.5.php>.
19. Angular. Angular Material. Angular Material. URL: <https://material.angular.io/>.
20. Angular. Angular. URL: <https://angular.io/>.
21. Automation in restaurants: how to streamline ordering. EngageBay - All-in-one marketing, sales, and service software for growing businesses. URL: <https://www.engagebay.com/blog/automation-in-restaurants/> (date of access: 20.05.2023).

22. Capturing functional requirements with use cases and user stories, 2022. URL: <https://www.cybermedian.com/ru/capturing-functional-requirements-with-use-cases-and-user-stories>.
23. Contributor T. What is block diagram? | Definition from TechTarget. WhatIs.com. URL: <https://www.techtarget.com/whatis/definition/block-diagram>.
24. Data Flow Diagram Symbols. Lucidchart. URL: <https://www.lucidchart.com/pages/data-flow-diagram/data-flow-diagram-symbols>).
25. Figma: the collaborative interface design tool. Figma. URL: <https://www.figma.com/>.
26. How to Find Your Target Audience | Marketing Evolution. A New Approach to Marketing Measurement & Optimization | Marketing Evolution. URL: <https://www.marketingevolution.com/marketing-essentials/target-audience>.
27. Hungry? Meet ukraine's first restaurant reservation app, RSRVIT APP is the first automated reservations app. Time Bulletin. URL: <https://www.timebulletin.com/hungry-meet-ukraines-first-restaurant-reservation-app-rsrvit-app-is-the-first-automated-reservations-app/> (date of access: 20.05.2023).
28. Online Restaurant Booking System / Reshma K R et al. International Journal of Advanced Research in Science, Communication and Technology. 2022. P. 757–769. URL: <https://doi.org/10.48175/ijarsct-7557>.
29. Pratt M. K., Roy M., McLaughlin E. What is Business Process? Definition from WhatIs.com. CIO. URL: <https://www.techtarget.com/searchcio/definition/business-process>. The benefits of using web-based applications | Geeks Insights. Digital Transformation for scale-up SMEs | Geeks. URL: <https://www.geeks.ltd.uk/about-us/blog/details/eQU5Ip/the-benefits-of-using-web-based-applications>.

30. Reshkova A. The difference and relationship between Use case and User story, 2018. URL: <https://medium.com/@a.reskova/the-difference-and-relationship-between-use-case-and-user-story-25e24df777a3>.
31. Restaurant reservation system know-how in 2020 | UpMenu. UpMenu. URL: <https://www.upmenu.com/blog/the-ultimate-guide-to-a-restaurant-reservation-system/> (date of access: 20.05.2023).
32. RestOn.ua: кращі заклади Києва на одному порталі | Резерв столика онлайн в ресторані, кафе, пабі. RestOn™. URL: <https://reston.ua/>.
33. Resy | restaurant management and table reservation software. ResyOS. URL: <https://resy.com/resyos/> (date of access: 20.05.2023).
34. Rymarenko B. Underestimated Power Of Non-Functional Requirements. URL: <https://www.uptech.team/blog/non-functional-requirements>.
35. Table reservations solution (TRS) | reservation software | infor. Infor. URL: <https://www.infor.com/products/table-management> (date of access: 20.05.2023).
36. Top automation tools to streamline restaurant operations. Eat App - Online Reservation System for Modern Restaurants. URL: <https://restaurant.eatapp.co/blog/automation-in-restaurant-industry> (date of access: 20.05.2023).
37. URL об'єкти. Сучасний підручник з JavaScript. URL: <https://uk.javascript.info/url>.
38. USA Text Message Statistics Updated for 2023. SMS Comparison. URL: <https://www.smscomparison.com/sms-statistics/#:~:text=Statistics%20have%20shown%20that%2098,3%20minutes%20of%20being%20delivered.&text=If%20given%20the%20option%20between,support%20rather%20than%20phone%20calls>.
39. What is a Use Case? Definition & Examples. URL: <https://study.com/academy/lesson/what-is-a-use-case-definition-examples.html>.
40. What is an algorithm? URL: <https://www.phgfoundation.org/briefing/what-is-an-algorithm>.

ДОДАТКИ

Додаток А
(обов'язковий)

SUMMARY

Chevhuз K.M. Development of a Web-Based System for Reserving Tables in a Restaurant. – Bachelor's thesis. Sumy State University, Sumy, 2023.

This work investigates the state of automation in the table reservation process in a restaurant and the relevance of this type of business in Ukraine. The paper also argues for the necessity of using a web-oriented system to automate this activity and defines the requirements for such a system. The main objective of the work is the development of a web-oriented table reservation system.

Keywords: web-oriented system, automation, reservation, restaurant, business process modeling, information system efficiency.

АНОТАЦІЯ

Чевгуз К.М. Розробка веб-орієнтованої системи бронювання столиків в ресторані. – Кваліфікаційна робота бакалавра. Сумський державний університет, Суми, 2023 р.

Дана робота досліджує стан автоматизації процесу бронювання столиків у ресторані та актуальність такого виду бізнесу в Україні. Також робота аргументує необхідність використання веб-орієнтованої системи для автоматизації цієї діяльності та визначає вимоги до такої системи. Основна мета роботи - розробка веб-орієнтованої системи бронювання столиків.

Ключові слова: веб-орієнтована система, автоматизація, бронювання, ресторан, моделювання бізнес-процесів, ефективність інформаційних систем.

Додаток Б
(інформаційний)
Аналогічні сайти, прототип
головної сторінки та сторінки бронювання

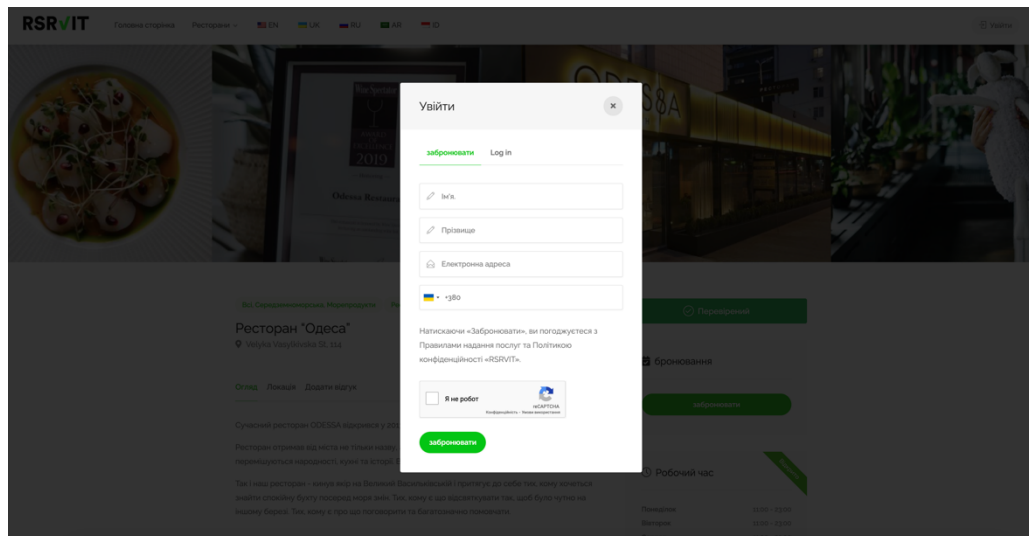
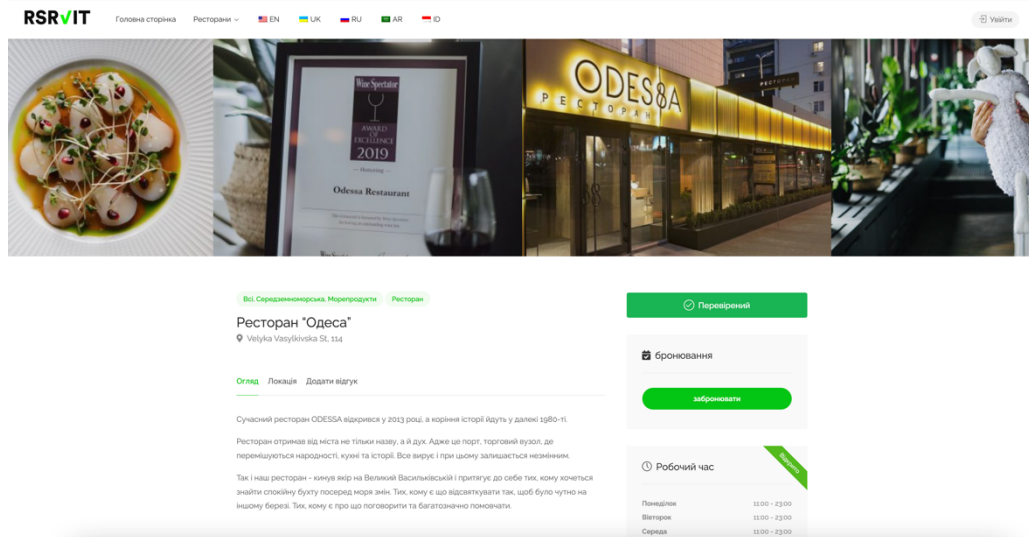


Рисунок Б.1 – Український сайт для бронювання столиків RSRVIT

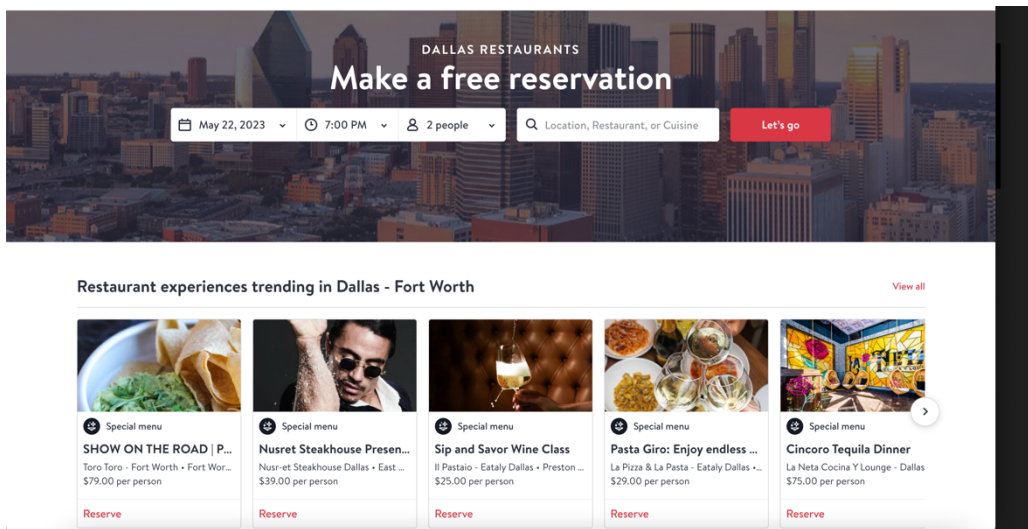
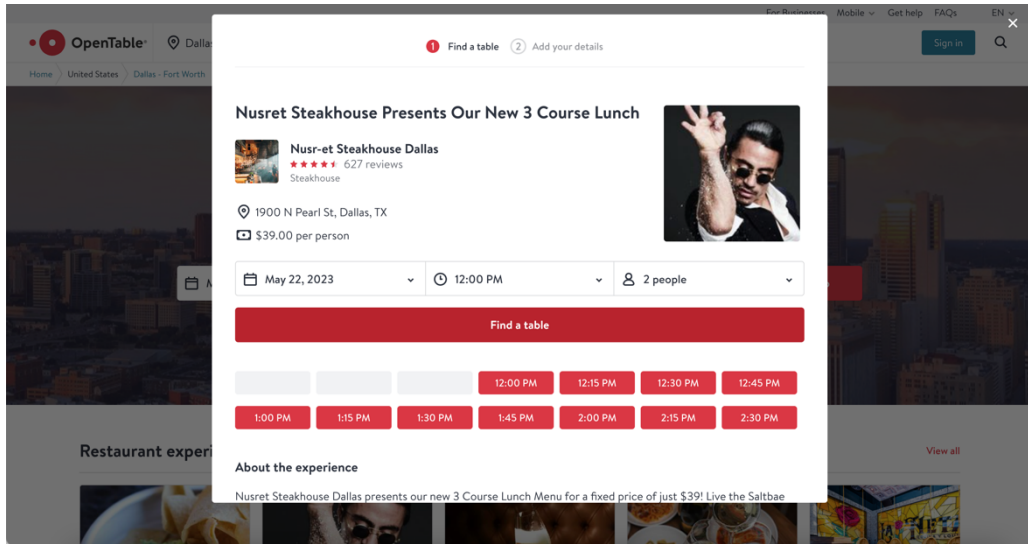


Рисунок Б.2 – Американський сайт для бронювання столиків OpenTable

Додаток В (інформаційний)

Код сайту

```

<mat-toolbar>
  <h3>UTI</h3>
  <div class="nav-section">
    <p>Головна</p>
    <p>Місця</p>
    <p>Профіль</p>
    <mat-icon inline>search</mat-icon>
  </div>
</mat-toolbar>

<div class="bd-header-img">
  <p class="slogan">Смачна Їжа, Чудова Атмосфера</p>

```

```

    <button mat-flat-button color="accent" class="choose-place-btn"
(click)="onBookClick()">Забронювати місце</button>
</div>

<div [ngSwitch]="bookPlaceState">
  <div class="story-section" *ngSwitchCase="'Default'">
    <div class="photos">
      <img ngSrc="../../assets/story-1.png" alt="Story photo" height="172" width="283">
      <img ngSrc="../../assets/story-2.png" alt="Story photo" height="172" width="283">
      <img ngSrc="../../assets/story-3.png" alt="Story photo" height="172" width="283">
      <img ngSrc="../../assets/story-4.png" alt="Story photo" height="172" width="283">
    </div>
    <div class="description">
      <p>Наша історія</p>
      <p>Uti – це сервіс легкого <br> бронювання столика <br> в улюбленому ресторані
<br> без телефонних дзвінків <br> та очікування.</p>
    </div>
  </div>

  <div class="book-form" *ngSwitchCase="'Book place'">
    <p>Бронювання</p>
    <form [formGroup]="bookPlaceForm">
      <div class="form-content">
        <mat-form-field>
          <mat-label>Ваше ім'я</mat-label>
          <input matInput [formControl]="bookPlaceForm.controls.firstName">
        </mat-form-field>
        <mat-form-field>
          <mat-label>Ваше прізвище</mat-label>
          <input matInput [formControl]="bookPlaceForm.controls.firstName">
        </mat-form-field>
        <mat-form-field>
          <mat-label>Номер телефону</mat-label>
          <input matInput [formControl]="bookPlaceForm.controls.phoneNumber">
          <mat-icon matSuffix>phone</mat-icon>
          <mat-hint>Введіть номер телефону</mat-hint>
        </mat-form-field>
        <mat-form-field>
          <mat-label>Електронна адреса</mat-label>
          <input matInput type="email" [formControl]="bookPlaceForm.controls.email">
          <mat-error *ngIf="bookPlaceForm.controls.email.hasError('email') &&
!bookPlaceForm.controls.email.hasError('required')">
            Введіть валідну електронну адресу
          </mat-error>
          <mat-error *ngIf="bookPlaceForm.controls.email.hasError('required')">
            Електронна адреса є <strong>обов'язковою</strong>
          </mat-error>
        </mat-form-field>
        <mat-form-field>
          <mat-label>Дата</mat-label>
          <input matInput [matDatepicker]="picker"
[formControl]="bookPlaceForm.controls.date">
          <mat-datepicker-toggle matIconSuffix [for]="picker">

```

```

        <mat-icon matDatepickerToggleIcon>keyboard_arrow_down</mat-icon>
      </mat-datepicker-toggle>
      <mat-datepicker #picker></mat-datepicker>
    </mat-form-field>
    <mat-form-field>
      <mat-label>Час</mat-label>
      <input matInput type="time" [formControl]="bookPlaceForm.controls.time">
      <mat-icon matIconSuffix>keyboard_arrow_down</mat-icon>
    </mat-form-field>
    <div class="empty"></div>
    <mat-form-field>
      <mat-label>Кількість гостей</mat-label>
      <input matInput type="number" min="1" max="12"
[formControl]="bookPlaceForm.controls.guestsNumber">
    </mat-form-field>
  </div>

  <div class="form-actions">
    <button mat-flat-button type="submit" class="submit-btn"
(click)="onBookSubmit()">Надіслати</button>
    <button mat-flat-button type="button" class="cancel-btn"
(click)="onBookCancel()">Скасувати</button>
  </div>
</form>
</div>
</div>
</div>

<div class="review-section">
  <p>Відгуки</p>
  <div class="review-switch-container" [ngSwitch]="reviewsContainerState">
    <div class="default-review-state" *ngSwitchCase="'Default'">
      <div class="client-reviews">
        <div class="client-review-item" *ngFor="let review of reviews">
          <img [ngSrc]="review.avatarSrc" alt="Client avatar" width="65" height="65">
          <div class="stars-rate">
            <mat-icon inline *ngFor="let starIndex of [1,2,3,4,5]"
[style.color]="review.stars >= starIndex ? '#E4C06F' : '#FFFFFF'">star</mat-icon>
          </div>
          <p class="review-text" [innerHTML]="review.text"></p>
        </div>
      </div>
      <p>А що ви думаєте про нас?</p>
      <button mat-flat-button (click)="leaveReviewClick()" >Залиште свій
відгук</button>
    </div>

    <div class="create-review-state" *ngSwitchCase="'Review create'">
      <form [formGroup]="createReviewForm">
        <div class="form-content">
          <mat-form-field>
            <mat-label>Ваше ім'я</mat-label>
            <input matInput [formControl]="createReviewForm.controls.name">
            <mat-error *ngIf="createReviewForm.controls.name.invalid">

```

```

        {{createReviewForm.controls.name.hasError('required') ? 'Ім'я є
обов'язковим' :
        createReviewForm.controls.name.hasError('minlength') ? 'Мінімальна
довжина імені - 3' :
        createReviewForm.controls.name.hasError('maxlength') ? 'Максимальна
довжина імені - 60' : null}}
    </mat-error>
</mat-form-field>
<mat-form-field>
    <mat-label>Напишіть свої враження ...</mat-label>
    <textarea matInput
        [formControl]="createReviewForm.controls.text"
        cdkTextareaAutosize
        #autosize="cdkTextareaAutosize"
        cdkAutosizeMinRows="5"
        (resize)="triggerResize()"></textarea>
    <mat-error *ngIf="createReviewForm.controls.text.invalid">
        {{createReviewForm.controls.text.hasError('required') ? 'Текст відгука є
обов'язковим' :
        createReviewForm.controls.text.hasError('minlength') ? 'Мінімальна
довжина відгука - 50' : null}}
    </mat-error>
</mat-form-field>
</div>

    <div class="form-actions">
        <button mat-flat-button type="submit" class="submit-btn"
[disabled]="createReviewForm.invalid" (click)="onSubmit()">Надіслати</button>
        <button mat-flat-button type="button" class="cancel-btn"
(click)="onCancel()">Скасувати</button>
    </div>
</form>
</div>

    <div class="created-review-state" *ngSwitchCase="'Review created'">
        <p>Ваша думка важлива для нас! <br>
        Дякуємо за відгук</p>
    </div>
</div>
</div>

<div class="footer">
    <div class="left-part">
        <p>Контакти</p>
        <p>Суми, Кондратьєва 140</p>
        <p>+38 (050) 155 333 77</p>
        <p>k.chevhus@gmail.com</p>
    </div>
    <div class="right-part">
        <p>Партнери</p>
        <p>La Spezia</p>
        <p>Verde by La Spezia</p>
        <p>SAZHA wine & meat</p>
    </div>
</div>

```

```
<p>VivaOliva</p>
</div>
</div>
```

Рис. В.1 – код html до головних сторінки сайту

```
background: transparent;
color: white;
position: absolute;
top: 0;
display: flex;
justify-content: space-between;
align-items: center;

h3 {
  font-size: 36px;
  font-family: 'Rock Salt', cursive;
  margin: 50px 0 0 80px;
}

.nav-section {
  margin: 60px 60px 0 0;
  display: flex;
  flex-direction: row;
  flex-wrap: nowrap;
  gap: 40px;

  p {
    &:hover {
      cursor: pointer;
    }

    &::after {
      content: '';
      width: 0;
      height: 2px;
      display: block;
      background: #ffffff;
      transition: 300ms ease-in-out;
    }

    &:hover::after {
      width: 100%;
    }
  }

  mat-icon {
    &:hover {
      cursor: pointer;
    }

    &::after {
```

```
        content: '';
        width: 0;
        height: 2px;
        display: block;
        background: #ffecb3;
        transition: 300ms ease-in-out;
    }

    &:hover::after {
        width: 100%;
    }
}
}

.bd-header-img {
    width: 100vw;
    height: 100vh;
    text-align: center;
    background-image: url("../assets/bg-header.png");
    background-repeat: no-repeat;
    background-size: 100% 100%;
}

.slogan {
    color: white;
    font-size: 64px;
    line-height: 78px;
    position: relative;
    text-align: center;
    top: 45%;
    font-weight: bold;
    overflow-wrap: break-word;
    height: 78px;
    overflow: unset;
}

button.choose-place-btn {
    position: relative !important;
    font-size: 24px;
    bottom: -65%;
    font-weight: bold;
    height: 68px;
    letter-spacing: normal;
    background: #E4C06F;
    border-radius: 9px;
}

.story-section {
    display: grid;
    grid-template-columns: calc(4 / 7 * 100%) calc(3 / 7 * 100%);
    padding-top: 176px;
}
```

```
.photos {
  display: grid;
  grid-template-columns: 283px 283px;
  column-gap: 26px;
  row-gap: 0;
  grid-template-rows: 172px + 35px;
  padding-left: 130px;
  padding-bottom: 150px;

  img:nth-child(2n + 1) {
    padding-top: 35px;
  }

  img:nth-child(2n) {
    padding-bottom: 35px;
  }
}

.description {
  p {
    font-weight: 400;
    font-size: 24px;
    line-height: 29px;
    color: #FFFFFF;
  }

  p:first-child {
    font-size: 32px;
    line-height: 39px;
    font-weight: 700;
    color: #E4C06F;
    margin-bottom: 39px;
  }
}

.review-section {
  padding-top: 30px;

  p:first-child {
    font-weight: 700;
    font-size: 32px;
    line-height: 39px;
    text-align: center;
    color: #FFFFFF;
    margin-bottom: 60px;
  }

  .review-switch-container {
    padding-bottom: 112px;

    .default-review-state {
      text-align: center;
    }
  }
}
```

```
p {
  font-weight: 700;
  font-size: 32px;
  line-height: 39px;
  text-align: center;
  margin-bottom: 57px;
  color: #FFFFFF;
}

button {
  height: 68px;
  background: #E4C06F;
  font-weight: 700;
  font-size: 24px;
  line-height: 29px;
  text-align: center;
  color: #000000;
  border-radius: 9px;
}

}

.client-reviews {
  padding: 0 100px;
  margin-bottom: 170px;
  display: grid;
  grid-template-columns: 1fr 1fr 1fr 1fr;

  .client-review-item {
    display: grid;
    justify-items: center;
    grid-template-rows: 65px 30px auto;
    row-gap: 24px;

    .stars-rate {
      display: flex;
      flex-direction: row;
      column-gap: 5px;
      margin-bottom: 12px;

      mat-icon {
        vertical-align: middle;
        height: 30px;
        width: 30px;
        font-size: 30px;
      }
    }
  }

  p {
    width: 183px;
    font-weight: 400;
    font-size: 16px;
    line-height: 20px;
```



```
        color: #FFFFFF;
        text-align: center;
        margin: auto;
    }
}
}

form {
    margin: auto;
    width: 60%;

    .form-content {
        display: flex;
        flex-direction: column;
        margin: auto;
        background: #202020;
        padding: 60px 100px 80px 100px;
    }

    .form-actions {
        display: grid;
        grid-template-columns: 1fr 1fr;
        justify-items: center;
        padding-top: 30px;

        button {
            width: 300px;
            border-radius: 9px;
            font-weight: 700;
            font-size: 24px;
            line-height: 29px;
            color: #000000;
            height: 68px;
        }

        .submit-btn {
            background: #E4C06F;
        }

        .cancel-btn {
            background: #D9D9D9;
        }
    }
}

.created-review-state {
    display: flex;
    align-items: center;
    justify-content: center;
    height: 500px;

    p {
        text-align: center;
    }
}
```

```

    }
  }
}

.footer {
  height: calc(295px - 43px);
  font-size: 24px;
  line-height: 29px;
  display: grid;
  grid-template-columns: 1fr 1fr;
  padding-top: 43px;
  background-color: white;

  .left-part {
    padding-left: 108px;

    p:first-child {
      font-weight: 700;
      margin-bottom: 29px;
    }
  }

  .right-part {
    padding-left: 20px;

    p:first-child {
      font-weight: 700;
      margin-bottom: 29px;
    }
  }
}
}

```

Рис. В.2 – код для задания стилю

```

import {Component, NgZone, ViewChild} from '@angular/core';
import {FormControl, FormGroup, Validators} from "@angular/forms";
import {CdkTextareaAutosize} from "@angular/cdk/text-field";
import {take} from "rxjs";

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.scss']
})
export class AppComponent {
  @ViewChild('autosize') autosize: CdkTextareaAutosize | undefined;

  title = 'uti';
  reviewsContainerState: 'Default' | 'Review create' | 'Review created' = 'Default';
  createReviewForm: FormGroup<{
    name: FormControl<string | null>,
    text: FormControl<string | null>

```

```

}> = new FormGroup({
  name: new FormControl<string>('', [Validators.required, Validators.minLength(3),
Validators.maxLength(60)]),
  text: new FormControl<string>('', [Validators.required, Validators.minLength(50)])
});
reviews: Review[] = [
  {
    avatarSrc: '../assets/review-1.png',
    stars: 5,
    text: '"Крутий сайт, все працює. Ми провели чудовий вечір" <br><br> – Джейн
Сміт'
  },
  {
    avatarSrc: '../assets/review-2.png',
    stars: 4,
    text: '"Нарешті і у нас з\'явився такий сайт. Дуже зручна платформа, чекаю ще
більше ресторанів в оновленні" <br><br> – Артур Стоун'
  },
  {
    avatarSrc: '../assets/review-3.png',
    stars: 5,
    text: '"Мені все подобається! Дизайн, простота використання, особливо можливість
замовити заздалегідь" <br><br> – Сільвія Дельві'
  },
  {
    avatarSrc: '../assets/review-4.png',
    stars: 5,
    text: '"Ненавиджу телефонні дзвінки, ваша платформа мене рятує" <br><br> – Ден
Браєн'
  }
]

constructor(private _ngZone: NgZone) {
}

leaveReviewClick() {
  this.reviewsContainerState = 'Review create';
}

triggerResize() {
  // Wait for changes to be applied, then trigger textarea resize.
  this._ngZone.onStable.pipe(take(1)).subscribe(() =>
this.autosize?.resizeToFitContent(true));
}

onSubmit() {
  console.log('ВІДГУК', ['Ім\`я користувача',
this.createReviewForm.controls.name.value], ['Текст відгука',
this.createReviewForm.controls.text.value]);
  this.reviewsContainerState = 'Review created';
}

onCancel() {

```

```
    this.reviewsContainerState = 'Default';  
  }  
}  
  
export interface Review {  
  avatarSrc: string;  
  stars: number;  
  text: string;  
}
```

Рис. В.3 – Код перевірки