

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
Національний університет «Запорізька політехніка»

Факультет комп'ютерних наук і технологій

(повне найменування факультету)

Кафедра програмних засобів

(повне найменування кафедри)

## Пояснювальна записка

до дипломного проєкту (роботи)

бакалавр

(ступінь вищої освіти)

на тему ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ  
ПІДТРИМКИ ДІЯЛЬНОСТІ РЕСТОРАНУ  
RESTAURANT SUPPORT SOFTWARE

Виконав: студент 4 курсу, групи КНТ-121сп

Спеціальності 121 Інженерія програмного

(код і найменування спеціальності)

забезпечення

Освітня програма (спеціалізація)

Інженерія програмного забезпечення

БОЙЧУК Д.С.

(ПРІЗВИЩЕ та ініціали)

Керівник КОЦУР М.І.

(ПРІЗВИЩЕ та ініціали)

Рецензент СКРУПСЬКИЙ С.Ю.

(ПРІЗВИЩЕ та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
**Національний університет «Запорізька політехніка»**

Факультет КНТ  
Кафедра програмних засобів  
Ступінь вищої освіти бакалавр  
Спеціальність 121 Інженерія програмного забезпечення  
(код і найменування)  
Освітня програма (спеціалізація) Інженерія програмного забезпечення  
(назва освітньої програми (спеціалізації))

**ЗАТВЕРДЖУЮ**

Завідувач кафедри ПЗ, д.т.н, проф.  
Сергій СУББОТІН  
“ ” 2024 року

**З А В Д А Н Н Я**

**НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)**

БОЙЧУКА Дмитра Сергійовича

(ПРИЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) Програмне забезпечення підтримки діяльності ресторану. Restaurant Support Software

керівник проєкту (роботи) к.т.н., доцент, КОЦУР Михайло Ігорович,  
(науковий ступінь, вчене звання, ПРИЗВИЩЕ, ім'я, по батькові)

затверджені наказом закладу вищої освіти від “ 24 ” квітня 2024 року № 168

2. Строк подання студентом проєкту (роботи) 10 червня 2024 року

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

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз предметної області. 2. Матеріали і методи. 3. Опис програми. 4. Експлуатація, тестування та експериментальне дослідження програми.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, кількість слайдів, плакатів)

Слайди презентації

6. Консультанти розділів проєкту (роботи)

| Розділ              | ПРІЗВИЩЕ, ініціали та посада консультанта | Підпис, дата   |                           |
|---------------------|-------------------------------------------|----------------|---------------------------|
|                     |                                           | завдання видав | прийняв виконане завдання |
| 1-4 Основна частина | КОЦУР М.І., доцент                        |                |                           |
| Нормоконтроль       | ЛИПОВЕЦЬ М.В., асистент                   |                |                           |

7. Дата видачі завдання “15” квітня 2024 року.

**КАЛЕНДАРНИЙ ПЛАН**

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

Студент(ка)

\_\_\_\_\_ Дмитро БОЙЧУК  
( підпис ) (Ім'я ПРІЗВИЩЕ)

Керівник проєкту (роботи)

\_\_\_\_\_ Михайло КОЦУР  
( підпис ) (Ім'я ПРІЗВИЩЕ)

## РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи бакалавра:  
86 с., 6 табл., 32 рис., 3 дод., 20 джерел.

ANGULAR, JAVASCRIPT, МОВА ПРОГРАМУВАННЯ,  
ПРОЄКТУВАННЯ, ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ, РЕСТОРАННА  
ДІЯЛЬНІСТЬ.

Об'єкт дослідження – процес розробки програмного забезпечення для підтримки діяльності закладів харчування.

Предмет дослідження – програмні засоби для підтримки діяльності ресторану.

Мета роботи – розробка програмного забезпечення для підтримки діяльності ресторану.

Матеріали, методи та технічні засоби: мова програмування JavaScript, фреймворк Angular.

Результати. Розроблено програмне забезпечення для підтримки діяльності ресторану за допомогою мови програмування JavaScript та фреймворку Angular.

Висновки. Виконано проєктування програмного забезпечення для підтримки діяльності ресторану. Розроблено програмне забезпечення для підтримки діяльності ресторану. Здійснено тестування розробленого програмного забезпечення для підтримки діяльності ресторану.

Галузь використання – програмні засоби для підтримки діяльності ресторану.

## **ABSTRACT**

Explanatory note to the diploma qualifying work of the bachelor: 86 pages, 6 tables, 32 figures, 3 appendixes, 20 sources.

ANGULAR, JAVASCRIPT, PROGRAMMING LANGUAGE, DESIGN, SOFTWARE, RESTAURANT BUSINESS.

The object of research is the process of developing software to support the activities of catering establishments.

The subject of the research is software for supporting restaurant activities.

The purpose of this work is to develop the software to support the restaurant's activities.

Materials, methods and technical tools: JavaScript programming language, Angular framework.

Results. The software tools to support restaurant operations using the JavaScript programming language and the Angular framework have been developed.

Conclusions. The software to support restaurant activities has been designed. The software to support restaurant activities has been developed. The developed software to support restaurant activities has been tested.

Scope of use – software tools for supporting restaurant operations.

## ЗМІСТ

|                                                                                                        |    |
|--------------------------------------------------------------------------------------------------------|----|
|                                                                                                        | С. |
| Перелік скорочень та умовних позначок .....                                                            | 8  |
| Вступ .....                                                                                            | 9  |
| 1 Аналіз предметної області .....                                                                      | 11 |
| 1.1 Процеси ресторанної діяльності та можливість їх автоматизації .....                                | 11 |
| 1.2 Аналіз програмного забезпечення для підтримки діяльності ресторану .....                           | 23 |
| 1.3 Висновки за розділом 1 .....                                                                       | 33 |
| 2 Матеріали і методи .....                                                                             | 36 |
| 2.1 Вибір мови програмування .....                                                                     | 36 |
| 2.2 Вибір фреймворку для розробки програмного забезпечення для<br>підтримки діяльності ресторану ..... | 38 |
| 2.3 Вибір системи керування базами даних .....                                                         | 40 |
| 2.4 Висновки за розділом 2 .....                                                                       | 43 |
| 3 Опис програми .....                                                                                  | 44 |
| 3.1 Структура програмного забезпечення для підтримки діяльності<br>ресторану .....                     | 44 |
| 3.2 Структура серверної частини програмного забезпечення для<br>підтримки діяльності ресторану .....   | 45 |
| 3.3 Структура клієнтської частини програмного забезпечення для<br>підтримки діяльності ресторану ..... | 47 |
| 3.4 Проєктування інтерфейсу програмного забезпечення для підтримки<br>діяльності ресторану .....       | 49 |
| 3.5 Висновки за розділом 3 .....                                                                       | 52 |
| 4 Експлуатація, тестування та експериментальне дослідження програми .....                              | 53 |
| 4.1 Призначення й умови застосування програми .....                                                    | 53 |
| 4.2 Характеристики програми для підтримки діяльності ресторану .....                                   | 54 |
| 4.3 Інструкція по експлуатації програми .....                                                          | 54 |
| 4.3.1 Звернення до програми .....                                                                      | 54 |
| 4.3.2 Вхідні й вихідні дані .....                                                                      | 54 |

|                                                                                    |    |
|------------------------------------------------------------------------------------|----|
| 4.3.3 Повідомлення.....                                                            | 55 |
| 4.4 Виконання програмного забезпечення для підтримки діяльності<br>ресторану ..... | 55 |
| 4.5 Тестування програмного забезпечення для підтримки діяльності<br>ресторану..... | 59 |
| 4.6 Висновки за розділом 4 .....                                                   | 59 |
| Висновки .....                                                                     | 60 |
| Перелік джерел посилань .....                                                      | 63 |
| Додаток А Технічне завдання .....                                                  | 65 |
| Додаток Б Фрагмент тексту програми .....                                           | 69 |
| Додаток В Слайди презентації .....                                                 | 79 |

## **ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК**

JS – JavaScript;

POS – Point of Sale;

БД – база даних;

ІТ – інформаційна технологія;

ПЗ – програмне забезпечення;

СУБД – система управління базами даних.

## ВСТУП

Програмне забезпечення для підприємств дрібної торгівлі, зокрема, підприємств, що ведуть ресторанну діяльність, дозволяє оптимізувати процеси, підвищити продуктивність та покращити якість обслуговування клієнтів. Програмні засоби для ресторанів можуть допомогти автоматизувати процес замовлень і ведення запасів, що полегшує контроль за наявністю продуктів та знижує ймовірність дефіциту або перевищення товарів [1]-[6].

Програмні засоби автоматизації ресторанної діяльності можуть включати функції онлайн-замовлень, програм лояльності та інші інструменти, що поліпшують якість обслуговування і забезпечують задоволення клієнтів. Програмне забезпечення (ПЗ) для підприємств дрібної торгівлі, зокрема, підприємств, що ведуть ресторанну діяльність, дозволяє вести облік фінансів, автоматизує процеси рахунків, податків і звітності, що сприяє точному фінансовому управлінню [1]-[6].

Програмні системи для ресторанів можуть забезпечувати аналітичні інструменти для вивчення даних про продажі, клієнтські вподобання та інші параметри, що дозволяє приймати обґрунтовані управлінські рішення. Застосування програмного забезпечення для обліку і контролю операцій в ресторанах може допомогти уникнути шахрайства або втрат в результаті недобросовісної діяльності працівників ресторанів. Програмне забезпечення для ресторанів може автоматизувати рутинні операції, такі як облік ресторанної продукції, замовлення клієнтів, продажі і ведення складу продукції для виготовлення ресторанних страв. Це дозволяє працівникам більше уваги приділяти стратегічним аспектам ресторанного бізнесу [1]-[6].

Проте впровадження програмного забезпечення у діяльність ресторану може бути досить дорогим завданням, особливо для дрібних ресторанів. Витрати на придбання, налаштування та навчання персоналу користуванню програмним забезпеченням можуть бути значними. Крім того, можуть виникнути проблеми інтеграції ресторанного програмного забезпечення з

іншими наявним ПЗ, що використовується на підприємстві. Це пов'язано з тим, що у випадку, коли ресторан вже використовує інші програмні системи, такі як бухгалтерське ПЗ чи ПЗ для управління персоналом, інтеграція нової програми може виявитися складною і вимагати додаткових витрат. Деякі програмні системи можуть вимагати високого рівня технічної компетенції працівників ресторану для належної експлуатації та обслуговування. Це може стати проблемою для ресторанів без великого ІТ-персоналу. Крім того, програмне забезпечення, зокрема і ПЗ для ресторанної діяльності, потребує регулярних оновлень та технічної підтримки. Якщо власник ресторану не забезпечить це, може виникнути ризик застарілості програми та проблем із сумісністю. Ресторани можуть відрізнятися за своїми потребами та процесами, тому відповідне програмне забезпечення може вимагати індивідуальної настройки, що може бути трудомістким завданням. Тому програмне забезпечення для ресторану повинно враховувати всі його аспекти, враховуючи конкретні потреби та обмеження ресторанного бізнесу [1]-[6].

Тому актуальною є розробка програмного забезпечення для підтримки діяльності ресторану. У дипломній кваліфікаційній роботі бакалавра розв'язується актуальне завдання розробки програмного забезпечення для підтримки діяльності ресторану.

Для досягнення поставленої мети у кваліфікаційній роботі бакалавра необхідно розв'язати такі задачі:

- виконати аналіз предметної області та програмних засобів для підтримки діяльності ресторану;
- здійснити проєктування програмного забезпечення для підтримки діяльності ресторану;
- створити програмне забезпечення для підтримки діяльності ресторану;
- виконати тестування розробленого програмного забезпечення для підтримки діяльності ресторану.

## 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

### 1.1 Процеси ресторанної діяльності та можливість їх автоматизації

Ресторани – це установи, які готують та подають їжу та напої клієнтам. Цей сегмент відіграє важливу роль в економіці та відображає культурне розмаїття, підтримує місцеве сільське господарство та є джерелом зайнятості. Тенденції в галузі можуть вказувати на ширші економічні та соціальні зміни [1]-[7].

Важливі аспекти ресторанної індустрії включають різноманітність ресторанів, вплив оцінювання на стандарти та вибір гостей, вплив пандемії, викликаной корона вірусною інфекцією, який призвів до росту послуг доставки та акценту на гігієні, фокус на стійкості з використанням місцевих інгредієнтів та екологічно чистих практик, а також економічна вигода через створення робочих місць та підтримку місцевої економіки [1]-[7].

Визначення ресторанів зосереджено на наданні послуг харчування для споживання на місці, вони є важливою частиною галузі гостинності [1]-[7].

Одним з ключових аспектів, який необхідно зрозуміти на початковому етапі автоматизації ресторанної діяльності, є концепція терміну ресторан. Простіше кажучи, цей термін використовується для опису сегмента сфери обслуговування, що фокусується на наданні їжі в ресторанах, барах, кафе та подібних закладах [1]-[7].

Такі заклади можуть варіюватися від незалежних ресторанів до мережових закладів, ресторанів швидкого харчування та інших форматів. Зазвичай будь-яка компанія, яка дозволяє клієнтам входити, замовляти їжу та вживати цю їжу на місці, може розглядатися як частина ресторанної індустрії, навіть якщо вона пропонує інші послуги [1]-[7].

Підприємства у галузі гостинності є важливими роботодавцями у загальній сфері гостинності. Посади в цьому виді бізнесу можуть включати офіціантів, кухарів, барменів, працівників обслуговування та менеджерів

ресторанів. Ресторанні підприємства часто розглядаються як складова галузі гостинності [1]-[7]. Хоча іноді використовуються як синоніми, терміни ресторанна індустрія та галузь гостинності мають визначення, що відрізняються і чіткі розходження. Основна відмінність полягає в тому, що ресторанна індустрія включає в себе підприємства, які фокусуються на наданні можливості пообідати клієнтам [1]-[7].

З іншого боку, поняття галузі гостинності має більший охоплення і включає всі підприємства, що пов'язані із сферою громадського харчування, незалежно від того, чи є можливість споживачеві обідати в ресторані. До цього відносяться ресторани, заклади громадського харчування, магазини роздрібних продуктів, спеціалізовані місця харчування і так далі [1]-[7].

Отже, ресторан є закладом роздрібною торгівлі, де готові страви надаються споживачам. Ресторани охоплюють різноманітні види закладів та стилів харчування та обслуговування. Ресторани іноді є частиною більшого комплексу, зазвичай готелю, де ресторан надається для зручності орендарів і, звісно ж, для збільшення їх потенційного доходу. Такі ресторани часто відкриті і для нерезидентів. В асортименті ресторанів можуть бути страви для тих, хто працює поруч, до простих і різноманітних меню, які доступні за доступними цінами в простому середовищі, або вищого класу закладів, які представляють фірмові страви та вишукані вина в елегантному інтер'єрі. У першому випадку клієнти часто обирають повсякденний стиль одягу. В останньому випадку, залежно від місцевих культур і традицій, клієнт може обирати напівформальний, формальний або неформальний стиль одягу [1]-[7].

При розробці програмного забезпечення підтримки ресторанної діяльності важливо розуміти, що клієнти, як правило, сидять за столиками, офіціант приймає їх замовлення, приносить їжу, коли вона готова, і виставляє рахунок перед тим, як клієнт залишить заклад [1]-[7].

Навіть при існуючому чіткому визначенні галузі гостинності при розробці програмного забезпечення для підтримки ресторанної діяльності

важливо враховувати, що існує різноманітність підприємств, які входять до загальної категорії гостинності [1]-[7].

Ресторани можна класифікувати за наявністю місць для сидіння, рівнем обслуговування та його якістю, атмосферою та ціновим діапазоном. Ресторани швидкого харчування пропонують обмежене меню, яке готується швидко. Зазвичай вони мають вікна для обслуговування на виїзд. Ресторани середнього рівня пропонують повноцінне харчування за помірними цінами, які клієнти вважають хорошими, це може бути повноцінне обслуговування, шведський стіл або обмежене обслуговування, коли клієнти роблять замовлення за стійкою та отримують їжу з доставкою, або шведський стіл. Елітні ресторани подають якісні страви за високими цінами, вони пропонують відмінний сервіс і створюють якісну атмосферу [1]-[7].

Проаналізуємо різні типи ресторанних закладів [1]-[7].

У сфері ресторанної індустрії елітний ресторан або ресторан високої кухні відноситься до закладів, які надають високоякісні послуги, включаючи вишукану кухню, витончену презентацію та вищий рівень формальності з досвідченим персоналом. Використовувані інгредієнти мають найвищу якість, а шеф-кухарі, що працюють у найкращих ресторанах, є професіоналами у відповідній галузі. Персонал є висококваліфікованим і готовим докладно пояснити меню, що обумовлює вищі ціни на продукцію [1]-[7].

Сегмент повсякденного харчування представляє заклади з більш спокійною атмосферою для обіду. Рівень формальності тут нижчий, порівняно з елітними ресторанами, але обслуговування за столиком може все ще присутнє. Повсякденні ресторани є більш доступними в порівнянні з елітними, але якість їжі може бути менш вищою [1]-[7].

Сучасні заклади для повсякденного харчування подібні до звичайних, проте акцентується на естетиці для привертання уваги сучасних споживачів, які хочуть ділитися своєю їжею та враженнями від ресторанів у соціальних мережах. Багато сучасних закладів для повсякденного харчування також мають своє місце в галузі гостинності і можуть фокусуватися на стійкості,

вишуканій кухні, сучасному дизайні та привабливості для молодшої аудиторії, а також для людей з середніми доходами [1]-[7].

Ресторани швидкого харчування операційно розташовані між швидкісними закладами та традиційними ресторанами. Основна увага приділяється швидкості обслуговування, і багато ресторанів цього типу приймають замовлення за прилавком, не надаючи повноцінного обслуговування. В порівнянні із сектором швидкого харчування в індустрії гостинності ціни вищі, але якість їжі може бути більш високою, хоча меню може бути більш обмеженим або фокусованим [1]-[7].

Ресторани сімейного стилю – це підприємства галузі харчової промисловості, які можна віднести до категорії "сімейні ресторани". Вони можуть бути формальними або неформальними, проте неформальні відзначаються більшою поширеністю та акцентом на подачі їжі на загальних тарілках, де клієнти діляться замовленою їжею і кладуть її на власні тарілки. В атмосфері панує сімейний характер, а вибір вин ймовірно буде більш обмеженим, ніж у вишуканих та неформальних закладах [1]-[7].

Ресторани швидкого харчування – це варіант швидкості і зручності. Хоча є можливість обідати в самому закладі, більшість ресторанів швидкого харчування також пропонують послугу доставки, а багато з них також надають послугу подання їжі в автомобілях. Їжа тут доступна за доступнішими цінами, але якість інгредієнтів може бути нижчою, ніж у інших ресторанах. Багато з цих підприємств працюють у форматі мереж чи франшиз та розташовані в різних місцях по всій країні або навіть у всьому світі [1]-[7].

Кав'ярня – це невеликий гастрономічний заклад, де пропонують відносно доступні та прості страви, такі як домашні страви, закуски, тістечка тощо. В той час як більшість ресторанів акцентується на приготуванні їжі, кав'ярні часто акцентують увагу на напоях, зокрема чаї та каві. Атмосфера тут розслаблена, а кафе може бути оформлені у формі кав'ярень, кав'ярень або чайні [1]-[7].

Ресторан-буфет – це заклад громадського харчування, де гості самостійно обслуговуються, вибираючи їжу і приносячи її на стіл. Типовий шведський стіл фокусується на конкретній страві, але деякі пропонують різноманіття страв. Зазвичай ці заклади працюють за принципом "все, що можеш з'їсти", де клієнти платять фіксовану суму, незалежно від обсягу спожитої їжі [1]-[7].

Фуд-траки та продуктові кіоски – це неформальні варіанти в ресторанній індустрії, які включають в себе мобільні фуд-траки та кіоски з їжею. У цих закладах простий стенд або фуд-трак, де приготується їжа та подається платникам. Вони часто мають обмежений вибір страв, фокусуючись, наприклад, на хот-догах, гамбургерах чи піці. Ці заклади можуть бути віднесені до ресторанів, якщо є можливість посидіти та спожити їжу [1]-[7].

Також існують тимчасові заклади, які можуть приймати різні форми. Вони можуть бути тимчасовими кулінарними стендами, де професійні кухарі готують страви, або ресторанами з повним сервісом, які існують лише на короткий період. У ресторанній галузі такі тимчасові заклади використовуються для оцінки інтересу, створення багатогранності або перевірки концепції ресторану [1]-[7].

Віртуальні ресторани – це нова концепція, яка переосмислює звичайне визначення ресторанної галузі. Це, головним чином, харчовий бізнес, який працює на винос або лише з можливістю виносу та функціонує як самостійний заклад, використовуючи наявну кухню та інфраструктуру ресторану. Ця ідея стала особливо популярною через пандемію COVID-19 [1]-[7].

Вплив пандемії COVID-19 на галузь гостинності був значний. Ця криза сильно вразила більшість галузей економіки, проте гостинна індустрія виявилася особливо вразливою через обмеження на громадські заходи, що призвело до неможливості відновлення звичайного режиму роботи. Багато компаній, щоб вижити, переорієнтувалися на позаміські послуги, такі як самовивіз та доставка, а також ресторани-привиди [1]-[7].

Внаслідок цього виникла потреба розробки додаткових програмних модулів для підтримки відповідної ресторанної діяльності. Світ вже виходить із найбільш складного періоду пандемії, і споживачі починають повертатися, але деякі з впроваджених під час кризи практик ймовірно залишаться в силі. Ресторани будуть продовжувати надавати послуги як на місці, так і за його межами, вживатимуть екстрених заходів та розглядатимуть нові методи організації роботи [1]-[7].

Також варто відзначити, що епідемія підвищила увагу споживачів до питань гігієни. Це означає, що ресторани можуть приділяти ще більше уваги забезпеченню чистоти та безпеки. Крім того, багато готелів і ресторанів укладають партнерства, щоб задовольняти потреби подорожуючих, забезпечуючи їх високоякісною їжею та забезпечуючи безпеку під час подорожей [1]-[7].

Проаналізуємо види та функції програмного забезпечення, що використовуються у ресторанній діяльності. В галузі гостинності все частіше використовують чат-боти, що базуються на штучному інтелекті. Ці чат-боти призначені для відповіді на стандартні запитання, збору даних та надання майже миттєвого зворотного зв'язку 24/7 [1]-[7].

Також важливими інвестиціями у галузі гостинності є програмні системи торгових точок, або POS-системи (Point of Sale, POS). Ці програмні системи є ключовим елементом інформаційних технологій, які можуть поліпшити роботу професіоналів у сфері гостинності. Однак, щоб максимально використати потенціал програмної POS-систем, важливо розглядати їх використання та цільове призначення, не обмежуючись лише базовими функціями обробки платежів [1]-[7].

У галузі гостинності підприємства використовують різноманітні програмні рішення для оптимізації процесів, підвищення продуктивності, поліпшення враження клієнтів та підвищення точності. Це охоплює різні програмні рішення, спрямовані на полегшення управління рестораном [1]-[7].

Проаналізуємо особливості програмного забезпечення управління рестораном. Таке програмне забезпечення призначено для допомоги користувачам у щоденному управлінні ресторанним бізнесом. До прикладів таких програмних інструментів відносяться системи торгових точок, системи управління замовленнями, системи управління взаємовідносинами з клієнтами та комплексні пакети "все в одному" [1]-[7].

Програмне забезпечення управління рестораном охоплює керування щоденними операціями ресторану або подібного середовища. Таке програмне забезпечення включає широкий спектр завдань та обов'язків, починаючи від фінансового нагляду за компанією та сприяння розвитку бізнесу до керування маркетингом, управління персоналом та забезпечення відмінного досвіду клієнтам [1]-[7].

Основна мета роботи програмного забезпечення управління рестораном полягає в забезпеченні балансу різних потреб бізнесу, встановленні лідерства, формулюванні стратегічних цілей, використанні програмного забезпечення для управління рестораном та впевненості, що всі відділи працюють належним чином. Узагалі, використання програмного забезпечення управління рестораном спрямоване на досягнення довгострокового успіху [1]-[7].

Програмна система управління рестораном представляє собою тип програмного забезпечення, спеціально розробленого для використання в галузі гостинності. Ці рішення, також відомі як програмне забезпечення для керування ресторанами, призначені для допомоги менеджерам та іншим працівникам у виконанні ключових завдань управління [1]-[7].

Такий тип програмного забезпечення може бути повністю інтегрованим програмним рішенням, яке включає елементи системи торгових точок, такі як обробка платежів, разом із більш складними функціями, такими як керування завданнями, управління запасами та замовленнями або система бронювання [1]-[7].

Сутність роботи програмного забезпечення управління рестораном полягає в забезпеченні стратегічних цілей та ефективної роботи всіх відділів, оскільки в кінцевому підсумку, управління рестораном є процесом керування рестораном з метою досягнення довгострокового успіху [1]-[7].

Проаналізуємо особливості програмного забезпечення для управління ресторанами [1]-[7]:

- комплексні можливості: це програмне забезпечення містить різноманітні функції, такі як обробка платежів, управління запасами та робочою силою, а також системи замовлення [1]-[7];

- автоматизація та ефективність: програмне забезпечення для управління ресторанами дозволяє автоматизувати процеси, прискорювати обслуговування та об'єднувати різні завдання, щоб покращити взаємодію з клієнтами та ефективність роботи [1]-[7];

- ключові функції включають хмарні рішення, звітність і аналітику, можливості інтеграції та технічну підтримку для ефективного управління ресторанною діяльністю [1]-[7];

- програмні засоби управління взаємовідносинами з клієнтами: програмне забезпечення для управління ресторанами має функції для керування даними клієнтів і підвищення залученості та лояльності клієнтів ресторану [1]-[7];

- гнучкість і адаптивність: програмне забезпечення для управління ресторанами можна легко розширювати та адаптувати до конкретних потреб і розміру ресторану [1]-[7].

Ефективні програмні системи для управління ресторанами характеризуються такими функціями та перевагами [1]-[7]:

- використання хмарних технологій. Хмарове програмне забезпечення для управління рестораном пропонує безліч переваг, які неможливо перевершити. Наприклад, отримання віддаленого доступу до програмного забезпечення стає набагато простіше, дані постійно оновлюються, створюються резервні копії, а операційні витрати зменшуються за

користування передбачуваними підписками. Використання програмного забезпечення, що базується на хмарній інфраструктурі, передбачає постійний доступ до Інтернету, але переваги також розширюються на підвищення безпеки та сумісності з аналогічними програмами [1]-[7];

– засоби аналізу та звітності є ще однією ключовою складовою систем управління рестораном. Зазвичай інформаційна панель надає основний огляд щоденних даних для тих, хто відповідає за управління рестораном. Одночасно існує можливість створення докладніших звітів. Ефективне програмне забезпечення для управління рестораном надає менеджерам, власникам та іншим керівникам змогу уважно контролювати ефективність ресторану від часу до часу. Інструменти аналітики допомагають ресторанам оцінити, розглянути та зрозуміти обсяги даних, які вони збирають [1]-[7];

– засоби інтеграції даних. Сучасні організації в основному використовують різноманітні види програм, і обмін даними між ними повинен бути якнайбільш плавним. Для цього ключовою є можливість інтеграції, тому ця характеристика є суттєвою для найкращих програмних рішень у сфері управління ресторанами. Постачальники послуг пропонують кілька хмарних рішень, які можуть налаштовувати систему управління рестораном з урахуванням потреб кожного бізнесу з урахуванням необхідної інтеграції. У випадках, коли це неможливо, рекомендується забезпечити базовий рівень інтеграції з системами різного типу, управління доходами та POS-системами для ресторанів [1]-[7];

– забезпечення технічної підтримки є ключовим аспектом ефективної програмної системи управління рестораном. В останньому випадку, коли виникають проблеми з програмним забезпеченням чи системні помилки, це може серйозно вплинути на ваш бізнес, тому важливо, щоб рішення було знайдено швидко. Функції технічної підтримки можуть набувати різних форм, включаючи доступ до чат-підтримки чи системи обробки заявок, надання допомоги з продажу, структурованої підтримки, усунення несправностей та доступ до посібників з програмного забезпечення [1]-[7].

Існує велика кількість різноманітних категорій програмних систем управління ресторанами або компонентів, які утворюють повну систему. Проаналізуємо основні категорії програмного забезпечення, що використовується у процесі підтримки ресторанної діяльності [1]-[7].

Програмне забезпечення POS-систем забезпечує можливість обробки платежів карткою, готівкою чи мобільним гаманцем, а також керування виставленням рахунків, друкування квитанції та автоматичне введення цієї інформації у великі дані компанії. Сучасні хмарні POS-системи сумісні з різними пристроями, включаючи мобільні телефони, iPad та інші планшети. Це змінило процес продажу для багатьох ресторанів, оскільки платежі можна обробляти за допомогою мобільних пристроїв, які також можна використовувати для виконання різних завдань [1]-[7].

Програмна система управління рестораном об'єднує ключові функції, пов'язані з керуванням рестораном. Наприклад, вона може синхронізувати дані з різних онлайн-каналів бронювання, дозволяючи переглядати всі замовлення та оновлювати інформацію для уникнення дублікатів. Програмна система управління рестораном може забезпечувати автоматизацію для автоматичного надсилання гостям електронної пошти або текстового повідомлення для підтвердження їх візиту в день резервації. Керівник ресторану або його адміністратор може генерувати звіти, які відображають досягнення певних критеріїв, а інформаційні панелі надають уявлення про поточний стан справ у ресторані [1]-[7].

Програмний компонент програмного забезпечення управління замовленнями в системі керування рестораном забезпечує ефективне управління замовленнями від клієнтів і від постачальників. При розміщенні замовлення клієнтом програмне забезпечення автоматично збирає інформацію та оновлює статус поточних запасів [1]-[7]. Зазвичай доступні опції перегляду історії замовлень клієнтів або відстеження активних замовлень, щоб гості могли знайти бажані страви. По мірі зменшення рівня запасів, постачальники можуть розміщувати кілька замовлень для певного продукту або компонента

та відстежувати їх протягом визначеного періоду часу, дозволяючи вам контролювати обсяги придбаного товару [1]-[7].

Програмне забезпечення управління персоналом в будь-якій високоякісній програмній системі керування рестораном дозволяє вирішувати завдання, пов'язані з ефективним управлінням трудовим колективом. Серед опцій можуть бути функції для створення графіка роботи, який гармонійно враховує потреби вашого бізнесу у персоналі та відповідає умовам трудових контрактів [1]-[7]. Можливі включення функцій для відстеження часу відвідування та контролю періодів відпустки чи інших відсутностей працівників. Цей програмний компонент дозволяє ефективно управляти оплатою праці, а функціональні можливості, пов'язані з управлінням людськими ресурсами і виплатами, також можуть містити різноманітні інтерфейсні можливості [1]-[7]. Програмне забезпечення управління персоналом підтримує різноманітні аспекти, такі як вирішення завдань ресторану, визначення посад співробітників та їх обов'язків, а також регулювання взаємин між ними. Таке програмне забезпечення сприяє установленню понять контролю, рівня управління, делегування обов'язків. Програмне забезпечення управління персоналом підтримує типову організаційну структуру відділу ресторанного господарства, проте в невеликих закладах можуть об'єднуватися різні функції в залежності від потреб конкретного підприємства [1]-[7]. Програмне забезпечення управління персоналом ресторану враховує, що різні види харчових операцій потребують різноманітних посад для успішного функціонування та ефективності закладу, більшість працівників працює у відділі харчування та напоїв [1]-[7]. Програмне забезпечення управління персоналом забезпечує взаємодію всіх відділів, щоб гості завжди отримували високоякісне обслуговування. Командна робота є важливою в будь-якому закладі ресторанної галузі. Лояльний та наполегливий персонал у поєднанні з ефективним управлінням за допомогою відповідного програмного забезпечення сприяють досягненню

головної мети закладу, пов'язаної із задоволенням харчових потреб відвідувачів ресторану [1]-[7].

Програмні засоби ефективного управління взаємовідносинами з клієнтами виявляються ключовим фактором для ресторанів, оскільки клієнти ресторанів можуть надходити з особливими запитаннями, дієтичними уточненнями чи скаргами. Функціонал такої програмної системи у системах управління рестораном дозволяє консолідувати всі комунікації в одному місці для легкого доступу та обробки [1]-[7]. Маркетингове програмне забезпечення ефективно вправляється з усіма аспектами рекламування ресторану, особливо в онлайн середовищі. За допомогою цього програмного забезпечення можна підтримувати контроль над рекламою, стратегією маркетингового контенту, спеціальними пропозиціями, акціями, розсилками маркетингових листів та іншими аспектами [1]-[7].

Визначено, що ресторан є закладом роздрібної торгівлі, де готові страви надаються споживачам. Проаналізовано процеси ресторанної діяльності та можливість їх автоматизації. Відзначено, що при розробці програмного забезпечення підтримки ресторанної діяльності важливо розуміти, що клієнти, як правило, сидять за столиками, офіціант приймає їх замовлення, приносить їжу, коли вона готова, і виставляє рахунок перед тим, як клієнт залишить заклад. Проаналізовано різні типи ресторанних закладів та їх особливості. Проаналізовано види та функції програмного забезпечення, що використовуються у ресторанній діяльності. Відзначено, що у ресторанній діяльності підприємства використовують різноманітні програмні рішення для оптимізації процесів, підвищення продуктивності, поліпшення враження клієнтів та підвищення ефективності. Таке програмне забезпечення призначено для допомоги користувачам у щоденному управлінні ресторанним бізнесом. До прикладів таких програмних інструментів відносяться програмні системи торгових точок, програмні системи управління замовленнями, програмні системи управління взаємовідносинами з клієнтами та комплексні програмні пакети. Відзначено, що програмна система управління рестораном

представляє собою тип програмного забезпечення, спеціально розробленого для використання в галузі гостинності. Отже програмне забезпечення для керування ресторанами призначено для допомоги менеджерам та іншим працівникам ресторану у виконанні їх ключових завдань.

## 1.2 Аналіз програмного забезпечення для підтримки діяльності ресторану

Проаналізуємо програмне забезпечення для підтримки діяльності ресторану [8]-[14].

Програмне забезпечення Restaurant365 (рис. 1.1) є інтегрованим програмним рішенням для ефективного управління ресторанним бізнесом, спрямованим на спрощення та оптимізацію різних аспектів діяльності закладу харчування. Програмне забезпечення Restaurant365 пропонує унікальну платформу, яка дозволяє ресторанам успішно вирішувати різноманітні завдання – від фінансового менеджменту і бухгалтерського обліку до ефективного контролю за запасами і розробки оптимального графіку роботи персоналу [9], [12].

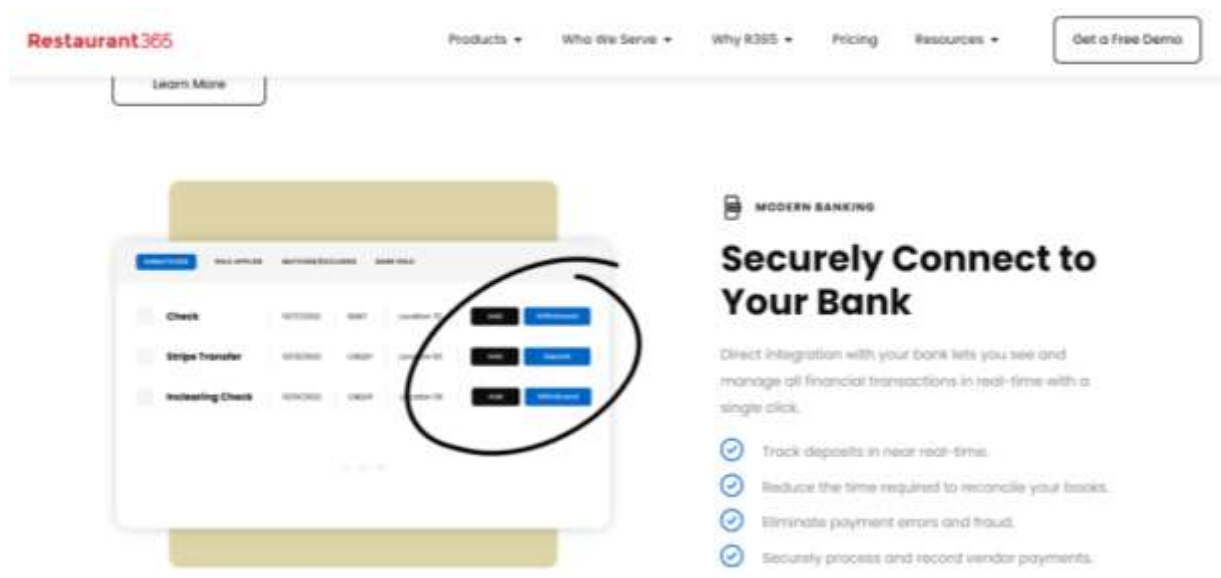


Рисунок 1.1 – Програмне забезпечення Restaurant365 [9], [12]

Шляхом забезпечення доступу до аналітики та реального часу для фінансових та операційних керівників, програмне забезпечення Restaurant365 допомагає приймати обґрунтовані рішення щодо ресторанної діяльності, спрямовані на зменшення витрат і підвищення загальної ефективності. Програмне забезпечення Restaurant365 спрощує складні завдання управління рестораном та підвищує його прибутковість, будь то один заклад чи мережа ресторанів з різними підрозділами [9], [12].

Програмне забезпечення Restaurant365 має програмний модуль для фінансового обліку ресторанів, який сприяє покращенню прибутковості та обсягу без додаткових витрат завдяки зручній інтеграції POS, безпосередній взаємодії з головною книгою, автоматизації обліку зобов'язань перед кредиторами та веденню звітності в реальному часі [9], [12].

Хмарна платформа програмного забезпечення Restaurant365 пропонує інноваційне та комплексне рішення для фінансового обліку, роботи з магазином та управління персоналом у ресторанах. Повна інтеграція з POS, банками, постачальниками тощо спрощує виконання завдань вручну та підвищує видимість і контроль над результатами [9], [12].

Програмне забезпечення Restaurant365 забезпечує автоматичне витягання даних про продажі та робочу силу безпосередньо з POS-системи, автоматизуючи створення щоденних записів у журналі продажів і трудових ресурсів для головної книги, економлячи час і підвищуючи точність [9], [12].

Програмне забезпечення Restaurant365 дозволяє керувати накладними витратами, інвентарем та операціями ресторану за допомогою надійних інструментів відстеження, а також забезпечує виконання адміністративних завдань ефективніше та точніше, отримуючи інформацію, необхідну для контролю витрат та збільшення маржі. Програма Restaurant365 забезпечує спрощення адміністративного процесу на кожному етапі роботи зі співробітниками за допомогою рішень для найму, адаптації, планування, нарахування заробітної плати та пільг. Програма Restaurant365 забезпечує

керівникам можливість зосередитися на підвищенні ефективності команди та продуктивності діяльності ресторану [9], [12].

Програмне забезпечення Restaurant365 дозволяє налаштовувати дані для максимальної продуктивності, отримувати точні та своєчасні дані, визначати, що працює, а що ні, і що відбувається в реальному часі за допомогою різноманітних звітів [9], [12].

Основними характеристиками програмного забезпечення Restaurant365 є такі [9], [12]:

- програмні модулі фінансового обліку: програмне забезпечення Restaurant365 дозволяє ресторанам вести бухгалтерію, включаючи відстеження прибутків, витрат та інших фінансових показників [9], [12];

- засоби інвентаризації: можливості відстеження запасів, автоматизоване оновлення інформації про інвентар, щоб уникнути дублювання та забезпечити належну управлінську інформацію [9], [12];

- POS інтеграція: програмне забезпечення Restaurant365 забезпечує взаємодії з точками продажу для ефективного оброблення замовлень та платежів [9], [12];

- засоби управління персоналом: програмне забезпечення Restaurant365 має функції для створення графіків роботи, відстеження відвідуваності та управління заробітною платою персоналу [9], [12];

- засоби звітності та аналітики даних: програмне забезпечення Restaurant365 надає деталізовані звіти та аналітику для допомоги в прийнятті обґрунтованих бізнес-рішень [9], [12];

- хмарна технологія: програмне забезпечення Restaurant365 підтримує можливість працювати через хмару, що полегшує доступ та оновлення для користувачів [9], [12];

- інтеграція з іншими системами: програмне забезпечення Restaurant365 підтримує можливість інтеграції з іншими програмами, такими як CRM, для оптимізації процесів [9], [12];

– технічна підтримка: наявність служби технічної підтримки для вирішення можливих проблем та питань користувачів [9], [12].

Серед недоліків програмного забезпечення Restaurant365 можна відзначити те, що деякі функції, які надає програмне забезпечення Restaurant365, можуть бути зайвими або надто розгалуженими для невеликих ресторанів та закладів харчування, що може призвести до надмірної складності та непотрібних витрат власників ресторану на придбання та впровадження програми Restaurant365 [9], [12].

Програмне забезпечення Foodics [9], [13] є комплексною програмною системою управління рестораном, спеціально розробленою для власників бізнесу в галузі харчування, включаючи ресторани, кафе та фургони з їжею. Програмне забезпечення Foodics (рис. 1.2) має на меті створити інтегровану екосистему управління рестораном, яка дозволяє власникам ефективно вести свою діяльність та розвивати бізнес [9], [13].

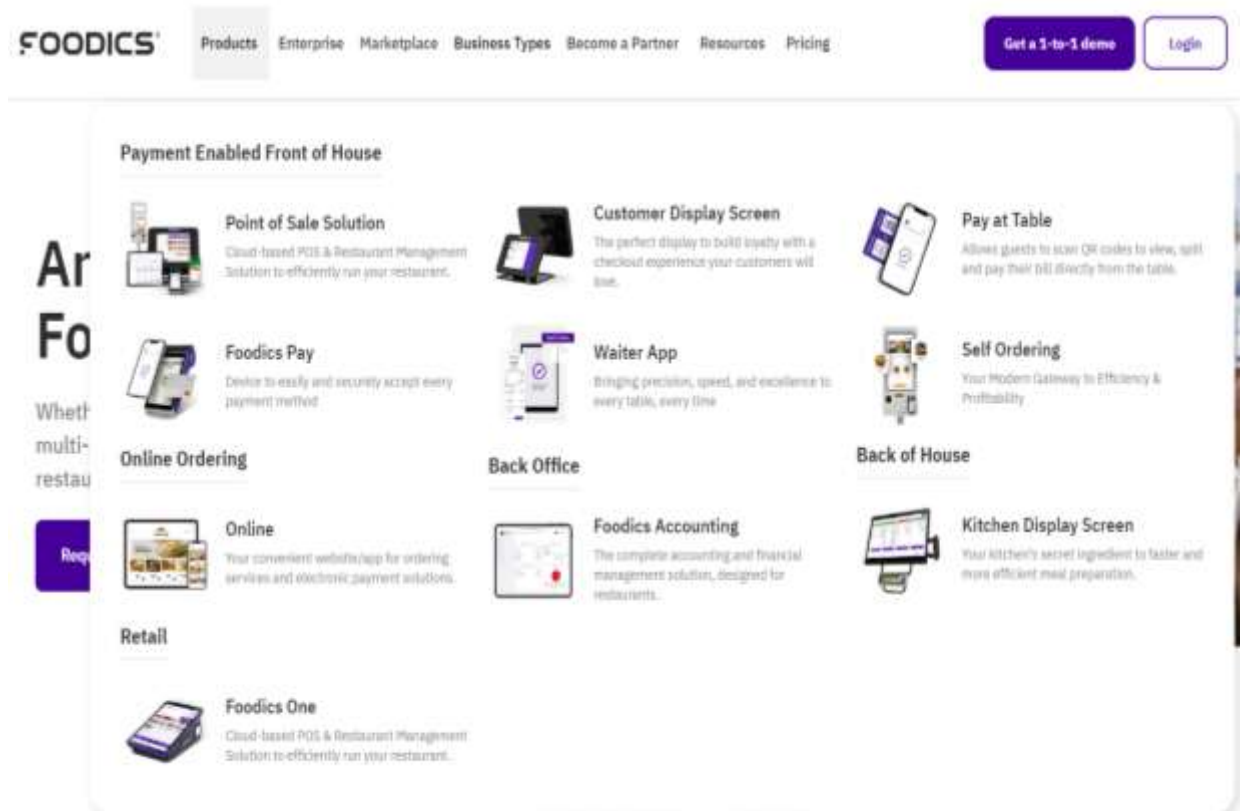


Рисунок 1.2 – Програмне забезпечення Foodics [9], [13]

Програмне забезпечення Foodics є комплексним рішенням для управління ресторанами та іншими закладами харчування, яке спрямоване на полегшення діяльності власників підприємств у будь-якому розмірі та сфері. Програмне забезпечення Foodics підтримує повну екосистему управління ресторанами, що дозволяє власникам ефективно керувати своїми операціями та розвивати бізнес. Основне завдання програмного забезпечення Foodics полягає у забезпеченні власників ресторанів та торговців всебічною хмарною POS-системою для покращення їхніх бізнес-процесів [9], [13].

Програмне забезпечення Foodics виступає в якості воріт до екосистеми, яка дозволяє власникам ресторанів скористатися численними інтеграціями сторонніх служб, таких як агрегатори доставки, бухгалтерія, фінанси, лояльність, аналітика, управління доставкою та інше [9], [13]. Програмне забезпечення Foodics підтримує технології для розвитку ресторанної спільноти, впроваджуючи ініціативи з підтримки галузі. Крім того, були введені два фінансові продукти: платіжне рішення, інтегроване з Foodics, що полегшує процес оформлення замовлення та завершення робочого дня, та програма мікрокредитування для малих підприємств з метою фінансування та підтримки їхнього розвитку [9], [13].

Програмний продукт Foodics Enterprise є рішенням для управління кількома відділеннями ресторану, яке дозволяє ефективно керувати всіма філіями ресторану з однієї платформи, піднімаючи бізнес на новий рівень та розвиваючи його [9], [13].

Основними характеристиками програмного забезпечення Foodics є такі [9], [13]:

– всеосяжність: програмне забезпечення Foodics є всеосяжною програмною системою, яка об'єднує різні аспекти управління рестораном в єдиному додатку. Ця особливість дозволяє забезпечити гармонійний підхід до керування ресторанными операціями [9], [13];

– хмарна POS-система: програмне забезпечення Foodics надає хмарну POS-систему, що дозволяє гнучко та зручно керувати рестораном з будь-якого місця [9], [13];

– легкість використання: користувачі програмне забезпечення Foodics відзначають його простоту використання, що дозволяє легко налаштовувати, категоризувати та індивідуалізувати елементи. Простота програмного забезпечення Foodics сприяє плавному робочому процесу, особливо під час активних годин роботи [9], [13];

– інтуїтивний дизайн і функціонал роблять програмне забезпечення Foodics ефективним засобом для збереження часу користувачів. Програмне забезпечення Foodics допомагає оптимізувати операційні процеси, забезпечуючи ефективність під час пік годин бізнесу [9], [13];

– функції управління запасами: програмне забезпечення Foodics надає потужні можливості управління запасами, що дозволяє користувачам слідкувати за залишками товарів та приймати обґрунтовані рішення щодо поповнення запасів. Ця функція сприяє кращому контролю та організації бізнесу за допомогою програмного забезпечення Foodics [9], [13];

– забезпечення підвищення продуктивності персоналу: програмне забезпечення Foodics допомагає покращити продуктивність персоналу, що є ключовим аспектом успішного ведення справ у сфері харчування. За допомогою Foodics власники можуть контролювати та покращувати ефективність свого колективу [9], [13];

– зручність наявності всіх необхідних функцій в одному програмному продукті, забезпечуючи ефективність управління рестораном [9], [13];

– легкість налаштування: програмне забезпечення Foodics забезпечує зручність налаштування, можливості індивідуалізації та категоризації елементів, що сприяє користувацькому комфорту [9], [13];

– доступність у хмарі: хмарна природа POS-системи програмного забезпечення Foodics отримує визнання за гнучкість, що дозволяє користувачам віддалено керувати своїми ресторанами [9], [13].

Серед недоліків програмного забезпечення Foodics можна відзначити те, що деякі функції, які надає програмне забезпечення Foodics, можуть бути зайвими або надто розгалуженими для невеликих ресторанів та закладів харчування, що може призвести до надмірної складності та непотрібних витрат власників ресторану на придбання програми Foodics [9], [13].

Програмне забезпечення Toast [9], [14] є технологічною програмною системою управління рестораном, створеною спеціально для ресторанів, що об'єднує багато рішень, починаючи від інтегрованого онлайн замовлення і закінчуючи програмами подарункових карток та лояльності, а також звітами про працю та продажі. Програмне забезпечення Toast (рис. 1.3) підтримує мобільні та хмарні POS-системи [9], [14].

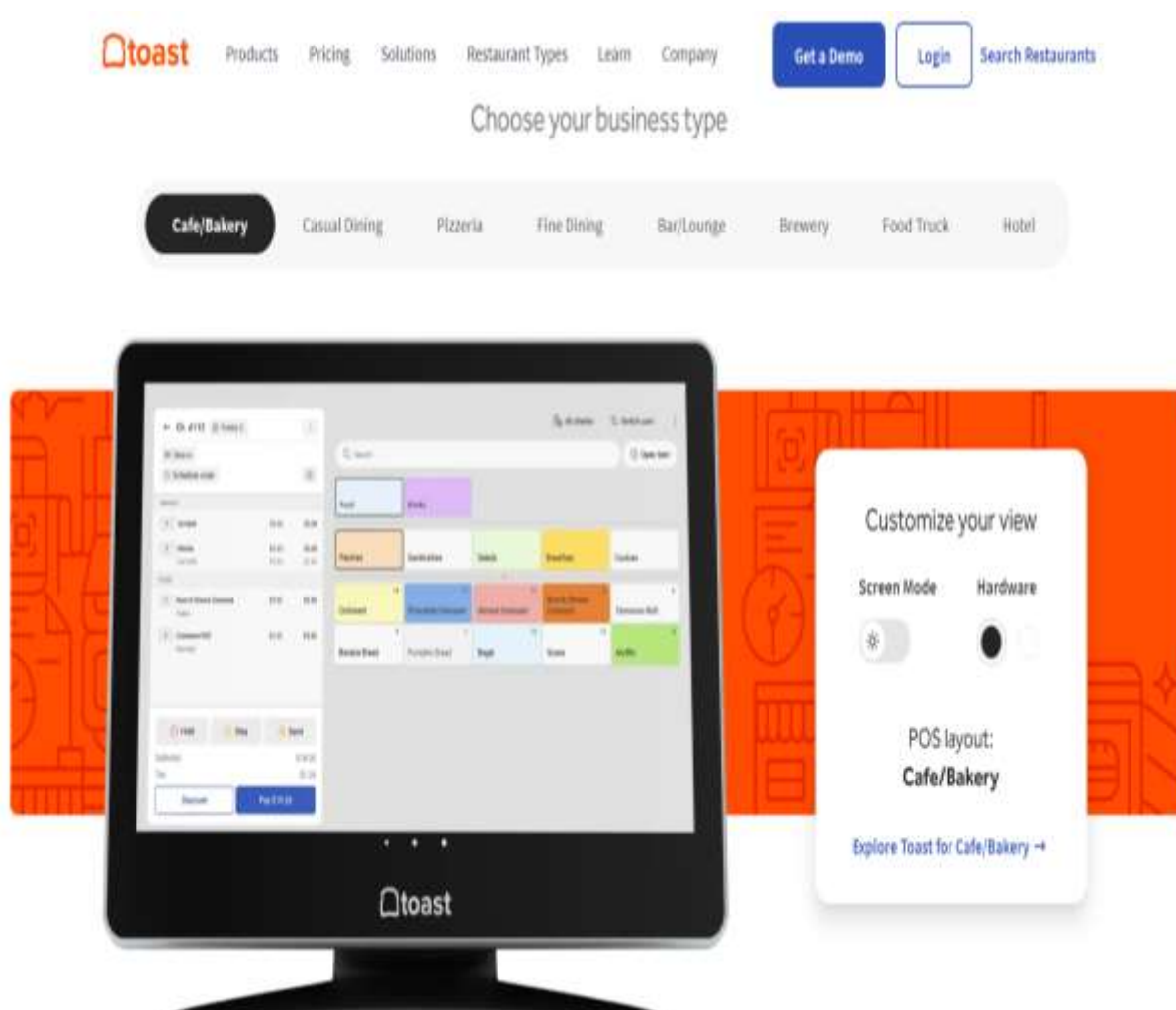


Рисунок 1.3 – Програмне забезпечення Toast [9], [14]

Програмне забезпечення Toast допомагає вдосконалити ресторанну діяльність, збільшити доходи і отримати більше бізнес-інсайтів. Місія програмного забезпечення Toast полягає у створенні найкращого можливого ресторанного підприємства, співпрацюючи з клієнтами для вирішення унікальних викликів ресторанного середовища [9], [14].

Підтримка клієнтів програмного забезпечення Toast завжди дружелюбна та ефективна, і дозволяє отримати відповіді на питання. Інтеграція програмного забезпечення Toast є досить простою. Весь системний блок програмного забезпечення Toast, як на робочому столі, так і на POS, простий та абсолютно зручний для користування як для ресторану, так і для роздрібних потреб. Для його використання практично не потрібне навчання, і на всі питання, що виникають, версія для робочого столу повна енциклопедії довідкових матеріалів та посібників. Всі дані, які користувачу потрібно відслідковувати, такі як продажі, робоча сила, розмір порції та лічильники, зберігаються та аналізуються програмним забезпеченням Toast. Програмне забезпечення Toast інтегрується з іншим програмним забезпеченням, щоб дозволити нам відстежувати та обробляти різні дані [9], [14].

Програмне забезпечення Toast дозволяє формувати звіти, які є дуже простими та зручними для використання. Програмне забезпечення Toast забезпечує легку імплементацію модулів. Програмне забезпечення Toast має велику кількість функцій, які можна додати до платформи, які повністю можуть змінити спосіб взаємодії з програмним забезпеченням POS. Прикладом такої інтеграції є модуль управління інвентарем Toast. Він може зекономити багато часу інвентаризації та процесу замовлення, а також забезпечити живий, динамічний інвентар, який оновлюється в реальному часі на основі поточної інформації. Програмне забезпечення Toast підтримує можливість зробити фотографію накладної, а також використовує засоби штучного інтелекту не лише для аналізу існуючого інвентарю, але й аналізу історії ціноутворення для всіх позицій [9], [14].

Програмне забезпечення Toast є дуже простою у використанні та сучасною програмною платформа. Співробітники ресторанів дуже швидко її освоюють. Програмне забезпечення Toast добре інтегрується з іншими системами, що дозволяє легкий доступ до даних [9], [14].

Основними характеристиками програмного забезпечення Toast є такі [9], [14]:

- інтегроване рішення: програмне забезпечення Toast пропонує все в одному підході, об'єднуючи інтегроване онлайн замовлення, програми для подарункових карток і лояльності, а також звітність про працю та продажі, всі ці функції доступні на мобільній, хмарній POS-системі [9], [14];

- мобільність: програмне забезпечення Toast працює на мобільних пристроях, що робить його зручним та доступним для використання в будь-якому місці в ресторані. Це полегшує прийом замовлень та керування операціями [9], [14];

- засоби підтримки зростання бізнесу: програмне забезпечення Toast спрямовано на те, щоб допомагати власникам ресторанів у вдосконаленні операцій, збільшенні доходів та отриманні більше інсайтів для прийняття обґрунтованих рішень [9], [14];

- засоби створення звітності: програмне забезпечення Toast надає функції звітності, включаючи дані про продажі, працю, порції та лічильники, забезпечуючи власникам необхідні дані для аналізу та управління [9], [14];

- зручний інтерфейс: програмне забезпечення Toast має простий та зручний інтерфейс, що полегшує використання як для ресторанів, так і для роздрібних точок [9], [14];

- підтримка клієнтів: програмне забезпечення Toast надає ефективну підтримку клієнтів з дружелюбним та ефективним підходом до вирішення питань та відповіді на запитання [9], [14];

- інтеграція з іншими програмними системами: програмне забезпечення Toast легко інтегрується з іншим програмним забезпеченням, що дозволяє вам відстежувати та керувати всім в одному місці [9], [14];

– надійність: програмне забезпечення Toast відоме своєю надійністю, забезпечуючи безперебійну роботу під час роботи ресторану [9], [14].

Загалом, програмне забезпечення Toast надає власникам ресторанів широкі можливості для поліпшення операцій та забезпечує ефективний інструмент для управління бізнесом у ресторанній галузі [9], [14].

Серед недоліків програмного забезпечення Toast можна відзначити те, що деякі функції, які надає програмне забезпечення Toast, можуть бути зайвими або надто розгалуженими для невеликих ресторанів та закладів харчування, що може призвести до надмірної складності та непотрібних витрат власників ресторану на придбання програми Toast [9], [14].

Порівняльну характеристику програмного забезпечення для підтримки діяльності ресторану наведено у таблиці 1.1.

Таблиця 1.1 – Порівняння програмного забезпечення для підтримки діяльності ресторану

| Критерій порівняння                      | Restaurant365 [12] | Foodics [13] | Toast [14] |
|------------------------------------------|--------------------|--------------|------------|
| Засоби інвентаризації                    | +                  | –            | +-         |
| POS інтеграція                           | +                  | +-           | +-         |
| Засоби управління персоналом             | +-                 | +            | +-         |
| Засоби встановлення та відстеження цілей | +                  | –            | –          |
| Ефективні засоби аналізу продуктивності  | +                  | –            | +-         |
| Зручність інтеграції з іншими системами  | –                  | +            | +          |
| Використання хмарних технологій          | +-                 | +            | +-         |

За результатами проведеного аналізу можна зробити висновок, що у наш час існує досить багато програмних засобів для підтримки діяльності ресторану. Проте впровадження програмного забезпечення у діяльність ресторану може бути досить дорогим завданням, особливо для дрібних ресторанів. Крім того, можуть виникнути проблеми інтеграції ресторанного програмного забезпечення з іншими наявним ПЗ, що використовується на підприємстві. Це пов'язано з тим, що у випадку, коли ресторан вже використовує інші програмні системи, такі як бухгалтерське ПЗ чи ПЗ для управління персоналом, інтеграція нової програми може виявитися складною і вимагати додаткових витрат. Деякі програмні системи можуть вимагати високого рівня технічної компетенції працівників ресторану для належної експлуатації та обслуговування. Крім того, ресторани можуть відрізнятися за своїми потребами та процесами, тому відповідне програмне забезпечення може вимагати індивідуальної настройки, що може бути трудомістким завданням. Тому актуальною є розробка програмного забезпечення для підтримки діяльності ресторану.

При розробці програмного забезпечення для підтримки діяльності ресторану необхідно забезпечити такі функціональні вимоги:

- можливість реєстрації та авторизації користувачів;
- підтримка процесів збереження та редагування інформації про користувачів;
- збереження інформації про транзакції користувачів;
- виведення інформації про меню ресторану;
- адміністрування новин ресторану;
- підтримка процесу оплати замовлення.

### **1.3 Висновки за розділом 1**

Визначено, що ресторан є закладом роздрібною торгівлі, де готові страви надаються споживачам. Проаналізовано процеси ресторанної

діяльності та можливість їх автоматизації. Відзначено, що при розробці програмного забезпечення підтримки ресторанної діяльності важливо розуміти, що клієнти, як правило, сидять за столиками, офіціант приймає їх замовлення, приносить їжу, коли вона готова, і виставляє рахунок перед тим, як клієнт залишить заклад. Проаналізовано різні типи ресторанних закладів та їх особливості. Проаналізовано види та функції програмного забезпечення, що використовуються у ресторанній діяльності. Відзначено, що у ресторанній діяльності підприємства використовують різноманітні програмні рішення для оптимізації процесів, підвищення продуктивності, поліпшення враження клієнтів та підвищення ефективності. Таке програмне забезпечення призначено для допомоги користувачам у щоденному управлінні ресторанним бізнесом. До прикладів таких програмних інструментів відносяться програмні системи торгових точок, програмні системи управління замовленнями, програмні системи управління взаємовідносинами з клієнтами та комплексні програмні пакети. Відзначено, що програмна система управління рестораном представляє собою тип програмного забезпечення, спеціально розробленого для використання в галузі гостинності. Отже програмне забезпечення для керування ресторанами призначено для допомоги менеджерам та іншим працівникам ресторану у виконанні їх ключових завдань.

За результатами проведеного аналізу зроблено висновок, що у наш час існує досить багато програмних засобів для підтримки діяльності ресторану. Проте впровадження програмного забезпечення у діяльність ресторану може бути досить дорогим завданням, особливо для дрібних ресторанів. Крім того, можуть виникнути проблеми інтеграції ресторанного програмного забезпечення з іншими наявним ПЗ, що використовується на підприємстві. Це пов'язано з тим, що у випадку, коли ресторан вже використовує інші програмні системи, такі як бухгалтерське ПЗ чи ПЗ для управління персоналом, інтеграція нової програми може виявитися складною і вимагати додаткових витрат. Деякі програмні системи можуть вимагати високого рівня технічної компетенції працівників ресторану для належної експлуатації та

обслуговування. Крім того, ресторани можуть відрізнятися за своїми потребами та процесами, тому відповідне програмне забезпечення може вимагати індивідуальної настройки, що може бути трудомістким завданням. Тому актуальною є розробка програмного забезпечення для підтримки діяльності ресторану.

Сформульовано функціональні вимоги до програмного забезпечення для підтримки діяльності ресторану.

## 2 МАТЕРІАЛИ І МЕТОДИ

### 2.1 Вибір мови програмування

При розробці програмного забезпечення для підтримки діяльності ресторану було використано мову програмування JavaScript [15], [16].

JavaScript – це мова програмування, спрямована на прототипи, що реалізує стандарт ECMAScript. Вона широко використовується в якості вбудованої мови для доступу до об'єктів програм та надає можливість додавання інтерактивності до вебсторінок у браузерях [15], [16].

Основні архітектурні особливості JavaScript включають динамічну та слабку типізацію, автоматичне керування пам'яттю, прототипне програмування, а також властивості об'єктів першого класу. Вплив на розвиток мови здійснили різноманітні мови, і JavaScript спроектований так, щоб нагадувати Java, але залишатися доступним для непрофесіоналів. Також, слід зазначити, що "JavaScript" є торговою маркою корпорації Oracle [15], [16].

JavaScript є об'єктно-орієнтованою мовою з особливим використанням прототипів, що робить його унікальним у порівнянні з традиційними об'єктно-орієнтованими мовами. Також, вона включає функції, характерні для функціональних мов, такі як об'єкти першого класу, списки об'єктів, анонімні функції та замикання [15], [16].

JavaScript можна розглядати як комбінацію трьох частин: базової (ECMAScript), об'єктної моделі браузера (BOM) та об'єктної моделі документа (DOM). Об'єктна модель браузера та об'єктна модель документа можуть не підтримуватися у середовищах без браузера [15], [16].

Написані програми можуть використовувати сервери Java або новіші для обробки JavaScript на стороні сервера. Це дозволяє створювати серверні програми з використанням JavaScript [15], [16].

У мові програмування JavaScript програмний код структурується у вигляді функцій та об'єктів, які можна об'єднувати у модулі. Ці модулі, в свою чергу, можна групувати у пакети програмного забезпечення [15], [16]. У світі

JavaScript існують різні реалізації, зокрема двигун, розроблений для підтримки різних платформ. Ліцензія Open Source, яку має ця мова, дозволяє використовувати її у різних програмах. Для різних середовищ виконання також існують інші реалізації, що розширюють можливості використання JavaScript у різних областях програмування. Організація JavaScript коду здійснюється через функції та об'єкти, які можна групувати у модулі, що, в свою чергу, можуть об'єднуватися в пакети [15], [16]. Наявність різних реалізацій JavaScript, поширених під ліцензією Open Source, робить цю мову високоякісною та гнучкою для використання в різних програмах. Універсальний менеджер пакетів сприяє ефективному використанню JavaScript, мінімізуючи труднощі при переходах між різними середовищами програмування та оволодінні новими можливостями [15], [16].

Мова програмування JavaScript у перші роки використовувалася переважно для розробки клієнтської частини вебзастосунків, проте в сучасному програмуванні ця мова є однією із найпоширеніших та ключових мов, що використовуються як на стороні клієнта, так і на сервері [15], [16]. JavaScript - це мова програмування, орієнтована на створення динамічних вебсайтів, здатна взаємодіяти з елементами вебсторінок, модифікувати їх зміст та реагувати на взаємодію з користувачем. Популярність JavaScript обумовлена його універсальністю: вона застосовується як для розробки інтерфейсу користувача, так і для обробки запитів на сервері. За підтримки активної спільноти розробників та постійного розвитку, JavaScript залишається однією з основних мов програмування у сфері веброзробки [15], [16]. У стандартній бібліотеці JavaScript реалізовано численні корисні функції. JavaScript підтримує різні парадигми програмування, такі як структурний, об'єктно-орієнтований, функціональний та імперативний [15]. Основні характеристики включають динамічну типізацію даних, автоматичне керування пам'яттю, повну обробку винятків, механізми взаємодії з винятками, підтримку операцій з багатьма потоками та зручну структуру даних високого рівня [15], [16].

Обґрунтування вибору мови програмування для розробки програмного забезпечення для підтримки діяльності ресторану наведено у таблиці 2.1.

Таблиця 2.1 – Обґрунтування вибору мови програмування для розробки програмного забезпечення для підтримки діяльності ресторану

| Критерій порівняння мов програмування                                          | Мова програмування |     |      |
|--------------------------------------------------------------------------------|--------------------|-----|------|
|                                                                                | JavaScript         | C++ | Java |
| Зручність синтаксису                                                           | +                  | +-  | +    |
| Активність спільноти розробників                                               | +                  | +-  | +-   |
| Безпека даних                                                                  | +                  | +-  | +    |
| Зручність розробки програмного забезпечення для підтримки діяльності ресторану | +                  | -   | -    |
| Застосування для розробки вебсайтів та вебзастосунків                          | +                  | +-  | +    |

Отже, для реалізації програмного забезпечення для підтримки діяльності ресторану обрано мову програмування JavaScript, яка широко використовується для створення вебзастосунків, в якості вбудованої браузерної мови для доступу до об'єктів програм та надає можливість додавання інтерактивності до вебсторінок у браузерах.

## 2.2 Вибір фреймворку для розробки програмного забезпечення для підтримки діяльності ресторану

В якості фреймворку для створення програмного забезпечення для підтримки діяльності ресторану було обрано Angular [17], [18].

Angular – це фреймворк із відкритим вихідним кодом, розроблений групою розробників від Google, який отримав активну підтримку від розробників спільноти та приватних компаній. Цей фреймворк є машинописним інструментом, що був повністю перероблений тією ж командою розробників, з метою надання розширених можливостей та поліпшеної продуктивності [17], [18].

Гнучка архітектура Angular визначає його як вискоєфективний інструмент для створення клієнтських програм. Фреймворк призначений для програмістів, які використовують JavaScript, і володіє розгалуженим набором корисних функцій, а також пропонує засоби для аналізу та оптимізації коду, що значно підвищує результативність розробки програмного забезпечення з різноманітними завданнями [17], [18].

Фреймворк також оснащений ефективними засобами для тестування модулів, сприяючи забезпеченню високої якості програмного продукту [17], [18]. Однією з ключових переваг Angular є вбудовані можливості візуального налагодження, що спрощують виявлення та усунення помилок у коді. Angular виявляє свою силу в підтримці веброзробки, роблячи його невід'ємним для тих, хто працює над створенням різноманітних вебзастосунків [17], [18].

Angular є машинописним фреймворком, орієнтованим на мову програмування JavaScript, і володіє розмаїттям корисних функцій. Цей фреймворк надає можливості для аналізу коду, візуального налагодження, проведення модульних тестів і підтримує веброзробку в рамках фреймворків. Він є доступним для використання на різних операційних системах [17], [18].

AngularJS пропонує різноманітні можливості, такі як маршрутизація, вбудована підтримка шаблонів, двостороннє зв'язування даних та інші програмні інструменти, які сприяють спрощенню розробки складних вебзастосунків [17], [18].

Обґрунтування вибору фреймворку для створення програмного забезпечення для підтримки діяльності ресторану наведено у таблиці 2.2.

Таблиця 2.2 – Обґрунтування вибору фреймворку для створення програмного забезпечення для підтримки діяльності ресторану

| Критерій порівняння фреймворків                               | Фреймворк |        |          |
|---------------------------------------------------------------|-----------|--------|----------|
|                                                               | Angular   | Vue.js | Ember.js |
| Доступність для використання на різних операційних системах   | +         | +–     | –        |
| Зручність для створення ПЗ для підтримки діяльності ресторану | +         | –      | –        |
| Двостороннє зв'язування даних                                 | +         | –      | –        |
| Зручність розробки на мові JavaScript                         | +         | +–     | +        |
| Активна спільнота розробників                                 | +         | +–     | +–       |

Таким чином, для створення програмного забезпечення для підтримки діяльності ресторану обрано фреймворк Angular, що є фреймворком із відкритим вихідним кодом, розроблений групою розробників від Google, який отримав активну підтримку від розробників спільноти та приватних компаній та є зручним для створення вебзастосунків.

### 2.3 Вибір системи керування базами даних

Для створення програмного забезпечення для підтримки діяльності ресторану обрано систему керування базами даних MySQL [19, 20].

MySQL є високопродуктивною та надійною системою управління базами даних (СУБД) з клієнт-серверною архітектурою, що надає можливість

здійснювати доступ до бази даних навіть з віддалених комп'ютерів за допомогою різноманітного клієнтського програмного забезпечення. Ця гнучкість дозволяє зручно управляти даними та виконувати операції з будь-якого місця [19, 20].

Однією з ключових переваг MySQL є його кросплатформенність, що робить його доступним для використання на різних операційних системах, таких як Windows, Linux, Mac OS, Solaris, HP-UX та інші. MySQL також підтримує кілька інтерфейсів прикладного програмування, включаючи C++, Eiffel, Java, Perl, PHP, Python, Tcl, ODBC та інші. Це робить MySQL універсальним рішенням для розробників, що використовують різні мови програмування [19, 20].

Заснована на концепції реляційної бази даних, MySQL дозволяє організовувати дані у вигляді таблиць з власними іменами. Кожна таблиця представляє собою збереження даних, які можна легко опрацьовувати та взаємодіяти з ними. Реляційний підхід спрощує взаємодію з базою даних та забезпечує консистентність даних [19, 20].

Окрім того, MySQL відзначається високим рівнем безпеки та обмежень доступу. Система обробки безпеки ґрунтується на ефективній системі аутентифікації, що забезпечує захист від несанкціонованого доступу та забезпечує конфіденційність даних. Таким чином, MySQL відповідає високим стандартам управління базами даних, надаючи засоби для ефективного та безпечного роботи з інформацією [19, 20].

MySQL знаходить застосування у створенні простих програм, де важливе легке збереження інформації, і при цьому не виникає потреба встановлювати та налаштовувати сервери баз даних [18, 19]. Однією з ключових переваг MySQL є його простота в експлуатації. Розробники можуть легко інтегрувати цю базу даних у свої проєкти, забезпечуючи зберігання даних в локальному середовищі, і в той же час уникнути установки та налаштувань серверів. Це особливо корисно для застосунків, які потребують

швидкого доступу до інформації та не потребують великих операцій з базою даних [19, 20].

Обґрунтування вибору системи керування базами даних для створення програмного забезпечення для підтримки діяльності ресторану наведено у таблиці 2.3.

Таблиця 2.3 – Обґрунтування вибору системи керування базами даних для створення програмного забезпечення для підтримки діяльності ресторану

| Критерій порівняння систем керування базами даних | Системи керування базами даних      |                          |                                                            |
|---------------------------------------------------|-------------------------------------|--------------------------|------------------------------------------------------------|
|                                                   | MySQL                               | PouchDB                  | LokiJS                                                     |
| Реляційна СУБД                                    | +                                   | –                        | –                                                          |
| Використання для створення вебзастосунків         | +                                   | +–                       | +–                                                         |
| Спосіб зберігання даних                           | Таблична модель з використанням SQL | JSON-документи для NoSQL | Об'єктно-орієнтована база даних, може використовувати JSON |
| Легкість використання                             | +                                   | +–                       | +–                                                         |
| Централізована архітектура зберігання даних       | +                                   | +                        | +–                                                         |

Таким чином, для створення програмного забезпечення для підтримки діяльності ресторану обрано систему керування базами даних MySQL, що є високопродуктивною та надійною СУБД з клієнт-серверною архітектурою, яка надає можливість здійснювати доступ до бази даних з віддалених комп'ютерів за допомогою різноманітного клієнтського програмного забезпечення та дозволяє зручно управляти даними і виконувати операції з будь-якого місця.

## **2.4 Висновки за розділом 2**

Для реалізації програмного забезпечення для підтримки діяльності ресторану обрано мову програмування JavaScript, яка широко використовується для створення вебзастосунків, в якості вбудованої браузерної мови для доступу до об'єктів програм та надає можливість додавання інтерактивності до вебсторінок у браузерах.

Для створення програмного забезпечення для підтримки діяльності ресторану обрано фреймворк Angular, що є фреймворком із відкритим вихідним кодом, розроблений групою розробників від Google, який отримав активну підтримку від розробників спільноти та приватних компаній та є зручним для створення вебзастосунків.

Для створення програмного забезпечення для підтримки діяльності ресторану обрано систему керування базами даних MySQL, що є високопродуктивною та надійною СУБД з клієнт-серверною архітектурою, яка надає можливість здійснювати доступ до бази даних з віддалених комп'ютерів за допомогою різноманітного клієнтського програмного забезпечення та дозволяє зручно управляти даними і виконувати операції з будь-якого місця.

### 3 ОПИС ПРОГРАМИ

#### 3.1 Структура програмного забезпечення для підтримки діяльності ресторану

Структуру програмного забезпечення для підтримки діяльності ресторану наведено на рис. 3.1.



Рисунок 3.1 – Структура програмного забезпечення для підтримки діяльності ресторану

Як видно з рисунку, програмне забезпечення для підтримки діяльності ресторану складається з таких компонентів:

- клієнтська частина програмного забезпечення для підтримки діяльності ресторану;
- серверна частина програмного забезпечення для підтримки діяльності ресторану;
- база даних програмного забезпечення для підтримки діяльності ресторану.

Проаналізуємо базовий функціонал клієнтської та серверної частин програмного забезпечення для підтримки діяльності ресторану (рис. 3.2).

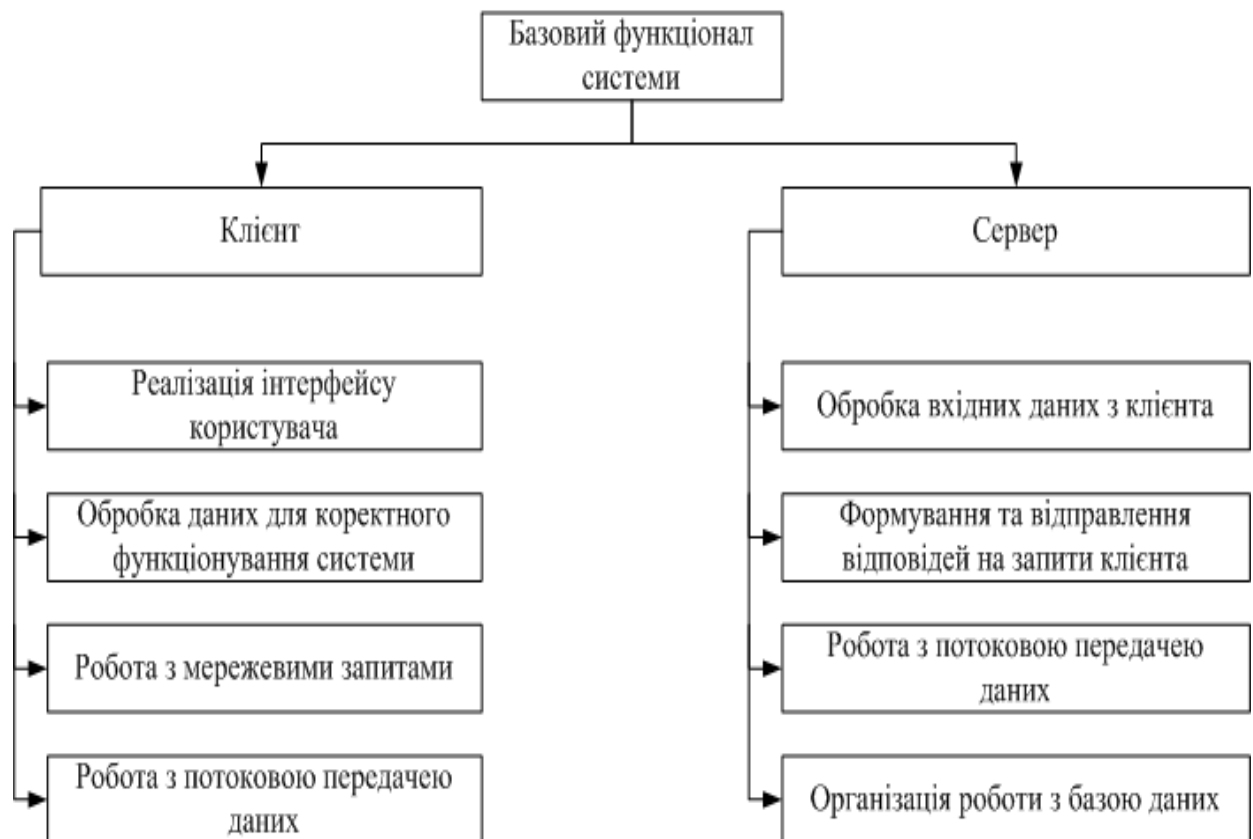


Рисунок 3.2 – Базовий функціонал клієнтської та серверної частин програмного забезпечення для підтримки діяльності ресторану

### 3.2 Структура серверної частини програмного забезпечення для підтримки діяльності ресторану

Фізичну структуру серверної частини програмного забезпечення для підтримки діяльності ресторану подано на рис. 3.3.

Опис фізичної структури компонентів серверної частини програмного забезпечення для підтримки діяльності ресторану наведено у табл. 3.1.

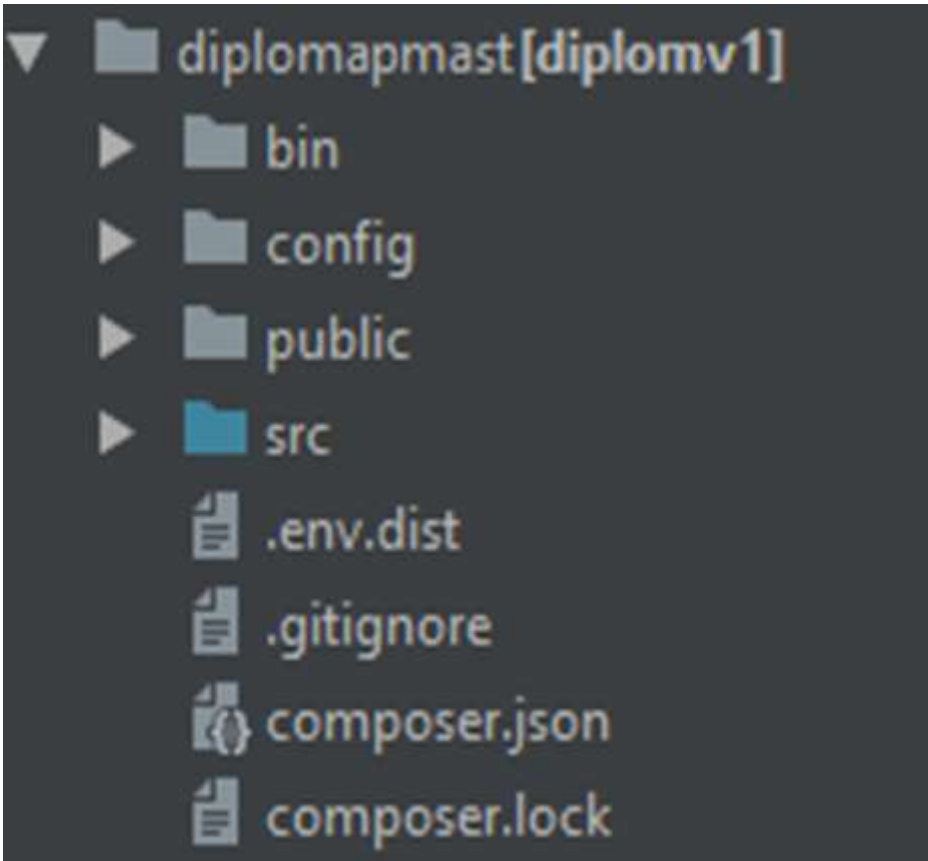


Рисунок 3.3 – Фізична структура серверної частини програмного забезпечення для підтримки діяльності ресторану

Таблиця 3.1 – Опис компонентів серверної частини програмного забезпечення для підтримки діяльності ресторану

| Компонент | Призначення                                                                                                                           |
|-----------|---------------------------------------------------------------------------------------------------------------------------------------|
| bin       | забезпечує створення та налаштування сервера                                                                                          |
| confif    | конфігураційні файли проєкту серверної частини програмного забезпечення для підтримки діяльності ресторану                            |
| public    | головний файл програми, через який проходять усі запити серверної частини програмного забезпечення для підтримки діяльності ресторану |

Продовження таблиці 3.1

| Компонент     | Призначення                                                                                                                       |
|---------------|-----------------------------------------------------------------------------------------------------------------------------------|
| src           | основний функціонал проєкту ПЗ для підтримки діяльності ресторану                                                                 |
| .env.dist     | файл конфігурації для інтерфейсу бази даних програмного забезпечення для підтримки діяльності ресторану                           |
| .gitignore    | список файлів проєкту програмного забезпечення для підтримки діяльності ресторану, які не проіндексовані системою контролю версій |
| composer.json | список сторонніх бібліотек, які використовуються в проєкті програмного забезпечення для підтримки діяльності ресторану            |
| composer.lock | сторонні бібліотеки, встановлених під час запуску проєкту програмного забезпечення для підтримки діяльності ресторану             |

### 3.3 Структура клієнтської частини програмного забезпечення для підтримки діяльності ресторану

Фізичну структуру клієнтської частини програмного забезпечення для підтримки діяльності ресторану подано на рис. 3.4.

Опис фізичної структури компонентів клієнтської частини програмного забезпечення для підтримки діяльності ресторану наведено у табл. 3.2.

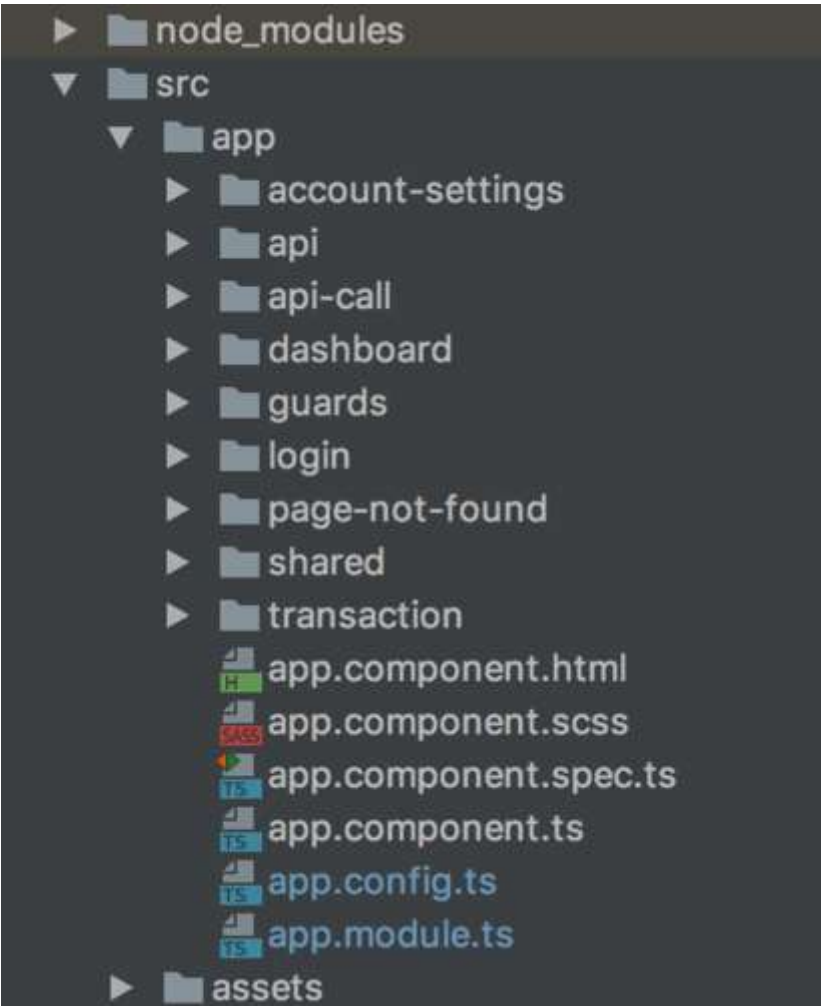


Рисунок 3.4 – Фізична структура клієнтської частини програмного забезпечення для підтримки діяльності ресторану

Таблиця 3.2 – Опис компонентів клієнтської частини програмного забезпечення для підтримки діяльності ресторану

| Компонент    | Призначення                                                                                                                 |
|--------------|-----------------------------------------------------------------------------------------------------------------------------|
| node_modules | пакети та бібліотеки, необхідні для запуску клієнтської частини програмного забезпечення для підтримки діяльності ресторану |
| app          | файли програмного модуля та служби клієнтської частини програмного забезпечення для підтримки діяльності ресторану          |

Продовження таблиці 3.2

| Компонент        | Призначення                                                                                                        |
|------------------|--------------------------------------------------------------------------------------------------------------------|
| account-settings | модуль для роботи з профілями користувачів програмного забезпечення для підтримки діяльності ресторану             |
| api              | файли конфігурації підключення API клієнтської частини програмного забезпечення для підтримки діяльності ресторану |
| api-call         | файли налаштувань підключення до API програмного забезпечення для підтримки діяльності ресторану                   |
| dashboard        | модулі основної частини клієнтської частини програмного забезпечення для підтримки діяльності ресторану            |
| guards           | файли захисту даних користувачів програмного забезпечення для підтримки діяльності ресторану                       |
| login            | модулі авторизації та реєстрації користувачів програмного забезпечення для підтримки діяльності ресторану          |
| shared           | сервіси клієнтської частини програмного забезпечення для підтримки діяльності ресторану                            |
| transaction      | модуль оплати рахунків клієнтом програмного забезпечення для підтримки діяльності ресторану                        |

### 3.4 Проєктування інтерфейсу програмного забезпечення для підтримки діяльності ресторану

Проєктування інтерфейсу взаємодії користувача з програмним забезпеченням для підтримки діяльності ресторану виконано з урахуванням вимог технічного завдання.

Інтерфейс користувача після запуску програмного забезпечення для підтримки діяльності ресторану наведено на рис. 3.5.



Рисунок 3.5 – Приклад інтерфейсу користувача після запуску програмного забезпечення для підтримки діяльності ресторану

Вікно аутентифікації користувача програмного забезпечення для підтримки діяльності ресторану наведено на рис. 3.6.

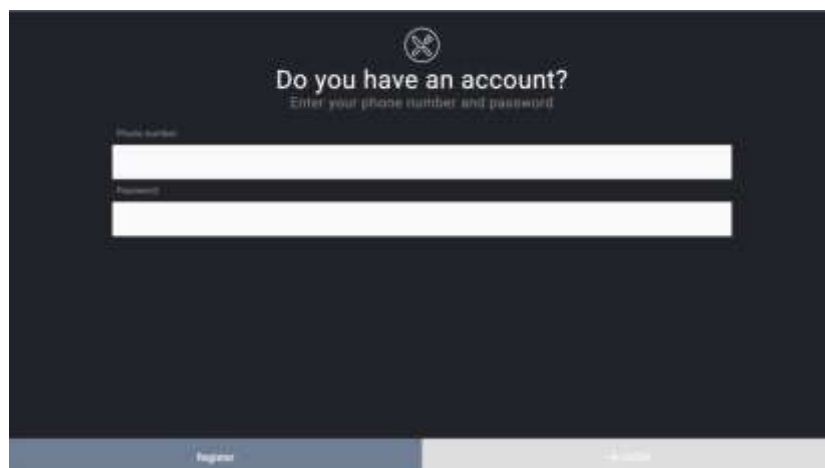


Рисунок 3.6 – Вікно аутентифікації користувача програмного забезпечення для підтримки діяльності ресторану

Вікно реєстрації користувача програмного забезпечення для підтримки діяльності ресторану наведено на рис. 3.7.

The screenshot shows a dark-themed registration form. At the top, there is a hamburger menu icon. Below it, the title 'Create your own account' is displayed in white, followed by the instruction 'Enter your phone number and password'. There are two white input fields: the first is labeled 'Phone number:' and the second is labeled 'Password:'. At the bottom, there is a light blue bar with a 'Login' button on the left and a '+ REGISTER' button on the right.

Рисунок 3.7 – Вікно реєстрації користувача програмного забезпечення для підтримки діяльності ресторану

Приклад сторінки зі списком замовлених страв наведено на рис. 3.8.

The screenshot shows a 'NEW ORDER' form with a light gray background. It contains a table with two rows of items. The first row has an unchecked checkbox, the text 'Coffee', and a price of '\$2'. The second row has a checked checkbox (indicated by a green checkmark), the text 'Soda', and a price of '\$1'. Below the table is a large green button with the text 'ADD' in white.

| NEW ORDER                           |        |     |
|-------------------------------------|--------|-----|
| <input type="checkbox"/>            | Coffee | \$2 |
| <input checked="" type="checkbox"/> | Soda   | \$1 |

ADD

Рисунок 3.8 – Приклад сторінки зі списком замовлених страв

Приклад сторінки зі списком страв, доступних у якості бонусу, наведено на рис. 3.9.

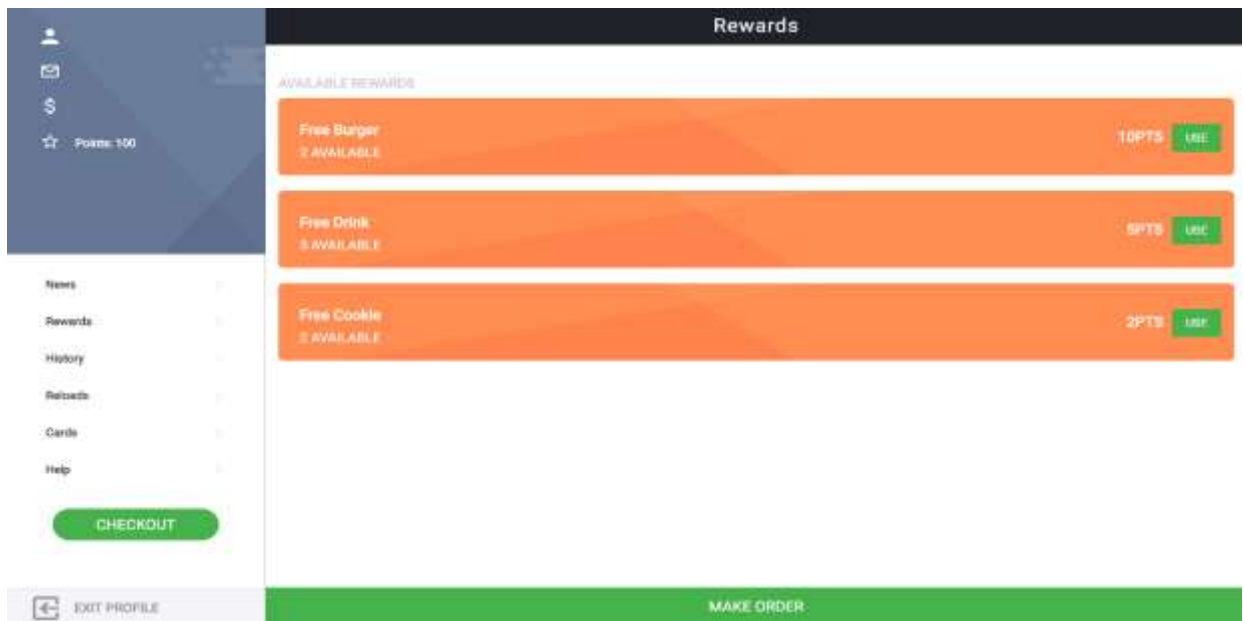


Рисунок 3.9 – Приклад сторінки зі списком страв, доступних у якості бонусу

### 3.5 Висновки за розділом 3

Розроблено структуру програмного забезпечення для підтримки діяльності ресторану.

Визначено, що програмне забезпечення для підтримки діяльності ресторану складається з клієнтської частини, серверної частини, а також бази даних для зберігання основної інформації про діяльність ресторану.

Розроблено програмне забезпечення для підтримки діяльності ресторану.

Описано особливості реалізації програмного забезпечення для підтримки діяльності ресторану. Виконано проєктування інтерфейсу взаємодії користувача з програмним забезпеченням для підтримки діяльності ресторану.

## **4 ЕКСПЛУАТАЦІЯ, ТЕСТУВАННЯ ТА ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ПРОГРАМИ**

### **4.1 Призначення й умови застосування програми**

Програма призначена для підтримки діяльності ресторану. Програма написана на мові JavaScript, яка широко використовується для створення вебзастосунків, в якості вбудованої браузерної мови для доступу до об'єктів програм та надає можливість додавання інтерактивності до вебсторінок у браузерах. В якості фреймворку для створення програми обрано Angular, що є фреймворком із відкритим вихідним кодом, розроблений групою розробників від Google, який отримав активну підтримку від розробників спільноти та приватних компаній та є зручним для створення вебзастосунків.

Програмне забезпечення для підтримки діяльності ресторану забезпечує виконання таких функцій:

- можливість реєстрації та авторизації користувачів;
- підтримка процесів збереження та редагування інформації про користувачів;
- збереження інформації про транзакції користувачів;
- виведення інформації про меню ресторану;
- адміністрування новин ресторану;
- підтримка процесу оплати замовлення.

Для експлуатації програмного забезпечення для підтримки діяльності ресторану необхідно таке апаратне та програмне забезпечення:

- оперативна пам'ять об'ємом 2Гб і більше;
- процесор із тактовою частотою не менш 1.0 ГГц;
- не менше 250 Мб вільного простору на запам'ятовуючому пристрою комп'ютера;
- операційна система.

Функціональні характеристики розробленого програмного забезпечення для підтримки діяльності ресторану наведено у технічному

завданні (додаток А). Фрагмент тексту програмного забезпечення для підтримки діяльності ресторану наведено у додатку Б.

## **4.2 Характеристики програми для підтримки діяльності ресторану**

Програмне забезпечення для підтримки діяльності ресторану може бути розгорнуто під керуванням різних операційних систем. Для функціонування серверної частини програмного забезпечення для підтримки діяльності ресторану необхідно встановити вебсервер. В операційній системі Windows для цього необхідно запустити встановлювач та керуватися вказівками майстра для встановлення програми. Крім того, необхідно встановити додаткове програмне забезпечення для коректної роботи пакетного менеджера сервера, який входить у встановлення разом із основною частиною. Цей менеджер пакетів необхідний для належної взаємодії зі сторонніми модулями, які використовуються у програмному забезпеченні для підтримки діяльності ресторану. Після успішного завершення процесу встановлення програми для підтримки діяльності ресторану вона готова до запуску.

## **4.3 Інструкція по експлуатації програми**

### **4.3.1 Звернення до програми**

Для запуску програмного забезпечення для підтримки діяльності ресторану необхідно запустити веббраузер та перейти на відповідну вебсторінку програми для підтримки діяльності ресторану.

### **4.3.2 Вхідні й вихідні дані**

Вхідними даними до програмного забезпечення для підтримки діяльності ресторану є інформація про користувача (номер телефону, пароль),

інформація про обрані користувачем страви, параметри для отримання історії замовлень користувача та ін.

Вихідними даними програмного забезпечення для підтримки діяльності ресторану є масив новин про діяльність ресторану, масив даних з інформацією про страви ресторану, повідомлення про успішне замовлення та ін.

#### **4.3.3 Повідомлення**

Користувач програмного забезпечення для підтримки діяльності ресторану може отримати такі повідомлення: повідомлення про успішну авторизацію чи реєстрацію, повідомлення про успішне оформлення замовлення та ін.

#### **4.4 Виконання програмного забезпечення для підтримки діяльності ресторану**

Після запуску програмного забезпечення для підтримки діяльності ресторану користувач повинен увійти до свого облікового запису. Для цього він вводить свої реєстраційні дані. Вікно аутентифікації користувача програмного забезпечення для підтримки діяльності ресторану наведено на рис. 4.1.

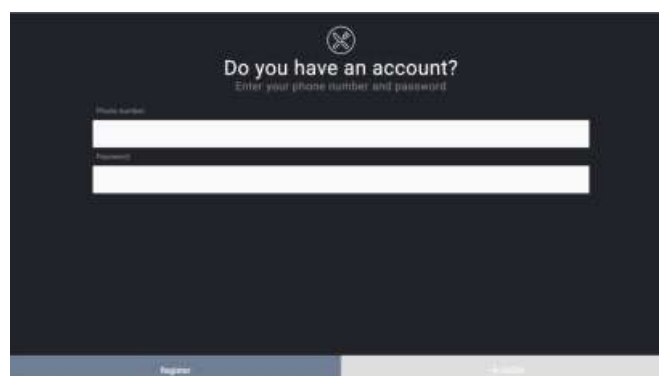


Рисунок 4.1 – Вікно аутентифікації користувача програмного забезпечення для підтримки діяльності ресторану

У випадку, якщо користувач незареєстрований, йому необхідно зареєструватися. Вікно реєстрації користувача програмного забезпечення для підтримки діяльності ресторану наведено на рис. 4.2.

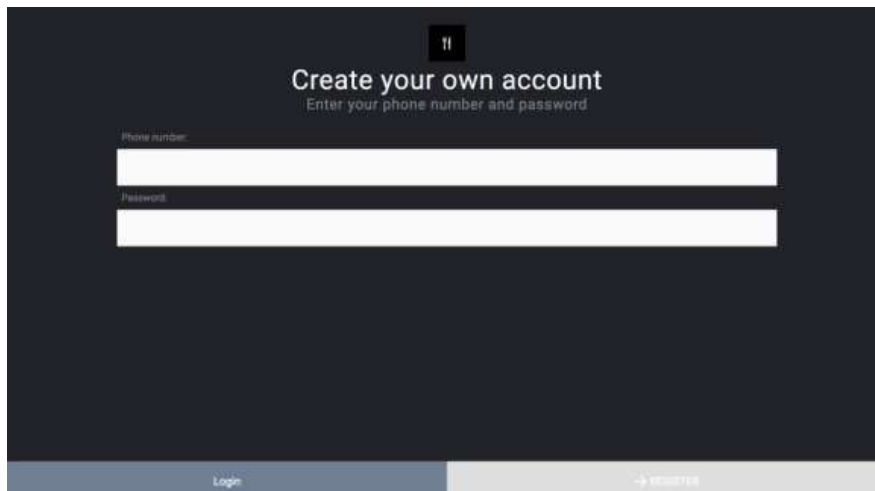
The image shows a registration form titled "Create your own account" with the subtitle "Enter your phone number and password". It features two input fields: "Phone number" and "Password". At the bottom, there are two buttons: "Login" and "→ REGISTER".

Рисунок 4.2 – Вікно реєстрації користувача програмного забезпечення для підтримки діяльності ресторану

Інтерфейс головної форми користувача після запуску програмного забезпечення для підтримки діяльності ресторану наведено на рис. 4.3.

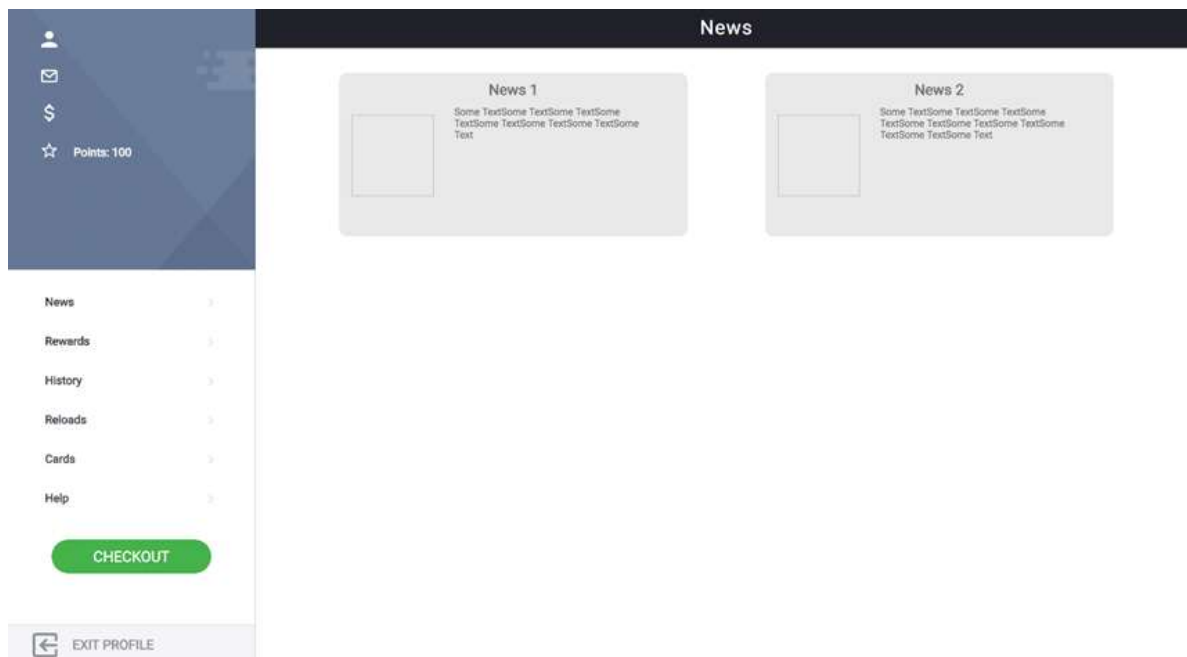


Рисунок 4.3 – Приклад інтерфейсу головної форми користувача після запуску програмного забезпечення для підтримки діяльності ресторану

Тут користувач може бачити меню для отримання інформації про новини про діяльність ресторану, його особисті відзнаки, історію замовлень, картки, а також пункт меню для отримання допомоги.

Для оформлення нового замовлення користувачу необхідно натиснути кнопку «Checkout», та сформувати нове замовлення, натиснувши потім кнопку «New Order» (рис 4.4).

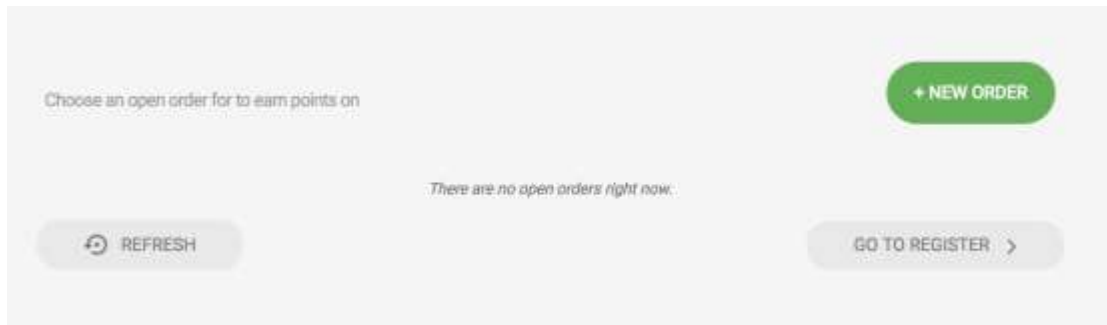


Рисунок 4.4 – Приклад сторінки створення нового замовлення

Після цього користувач програмного забезпечення для підтримки діяльності ресторану обирає страви зі списку страв (рис. 4.5).

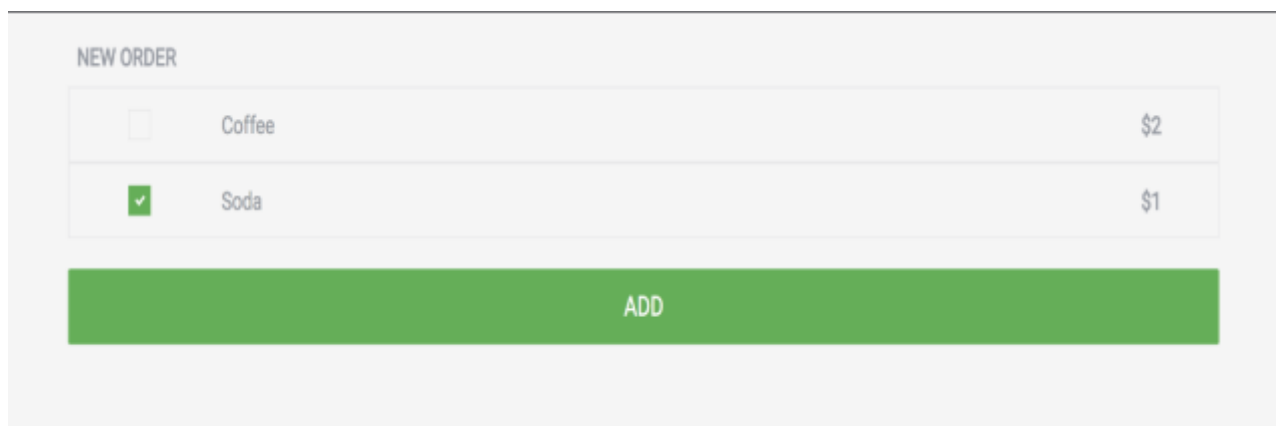


Рисунок 4.5 – Приклад формування користувачем замовлення шляхом вибору страв зі списку

Після цього користувач оплачує замовлення (рис. 4.6).

The image shows a payment interface. At the top, it displays '\$0 Balance' and a toggle switch for 'Pay with balance: \$0'. Below this is a text input field with a '\$' symbol and a placeholder 'Please enter the bill total'. Underneath the input field is a numeric keypad with digits 1 through 9, a decimal point, and a zero. To the right of the zero is a backspace button with an 'x' icon. At the bottom of the interface is a grey button labeled 'FROM CHECKOUT'.

Рисунок 4.6 – Форма для оплати замовлення

На головній сторінці у пункті меню Rewards користувач може ознайомитися з інформацією про страви, які є доступними його у якості бонусу за попередні замовлення. Приклад сторінки зі списком страв, доступних у якості бонусу, наведено на рис. 4.7.

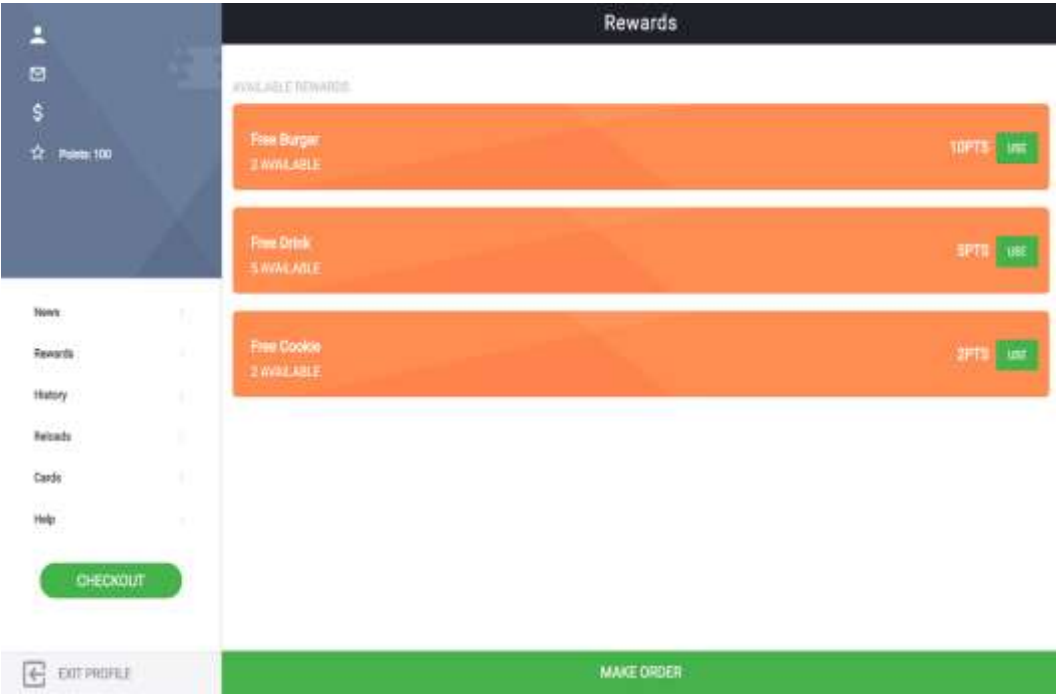


Рисунок 4.7 – Приклад сторінки зі списком страв, доступних у якості бонусу

#### **4.5 Тестування програмного забезпечення для підтримки діяльності ресторану**

Виконано тестування програмного забезпечення для підтримки діяльності ресторану.

Виявлено, що програма для підтримки діяльності ресторану функціонує правильно та злагоджено. Вона реалізує всі функціональні вимоги та успішно виконує свою основну задачу – підтримка діяльності ресторану.

Розроблене програмне забезпечення дозволяє забезпечувати автоматизацію процесів, пов'язаних з діяльністю ресторану.

#### **4.6 Висновки за розділом 4**

Описано програмне забезпечення для підтримки діяльності ресторану. Виконано тестування розробленого програмного забезпечення для підтримки діяльності ресторану. Результати тестування програмного забезпечення показали, що розроблена програма дозволяє забезпечити автоматизацію процесів, пов'язаних з діяльністю ресторану.

## ВИСНОВКИ

В ході виконання дипломної кваліфікаційної роботи бакалавра було проаналізовано та досліджено процес розробки програмного забезпечення для підтримки діяльності закладів харчування.

Визначено, що ресторан є закладом роздрібної торгівлі, де готові страви надаються споживачам. Проаналізовано процеси ресторанної діяльності та можливість їх автоматизації. Відзначено, що при розробці програмного забезпечення підтримки ресторанної діяльності важливо розуміти, що клієнти, як правило, сидять за столиками, офіціант приймає їх замовлення, приносить їжу, коли вона готова, і виставляє рахунок перед тим, як клієнт залишить заклад. Проаналізовано різні типи ресторанних закладів та їх особливості. Проаналізовано види та функції програмного забезпечення, що використовуються у ресторанній діяльності. Відзначено, що у ресторанній діяльності підприємства використовують різноманітні програмні рішення для оптимізації процесів, підвищення продуктивності, поліпшення враження клієнтів та підвищення ефективності. Таке програмне забезпечення призначено для допомоги користувачам у щоденному управлінні ресторанним бізнесом. До прикладів таких програмних інструментів відносяться програмні системи торгових точок, програмні системи управління замовленнями, програмні системи управління взаємовідносинами з клієнтами та комплексні програмні пакети. Відзначено, що програмна система управління рестораном представляє собою тип програмного забезпечення, спеціально розробленого для використання в галузі гостинності. Отже програмне забезпечення для керування ресторанами призначено для допомоги менеджерам та іншим працівникам ресторану у виконанні їх ключових завдань.

За результатами проведеного аналізу зроблено висновок, що у наш час існує досить багато програмних засобів для підтримки діяльності ресторану. Проте впровадження програмного забезпечення у діяльність ресторану може бути досить дорогим завданням, особливо для дрібних ресторанів. Крім того,

можуть виникнути проблеми інтеграції ресторанного програмного забезпечення з іншими наявним ПЗ, що використовується на підприємстві. Це пов'язано з тим, що у випадку, коли ресторан вже використовує інші програмні системи, такі як бухгалтерське ПЗ чи ПЗ для управління персоналом, інтеграція нової програми може виявитися складною і вимагати додаткових витрат. Деякі програмні системи можуть вимагати високого рівня технічної компетенції працівників ресторану для належної експлуатації та обслуговування. Крім того, ресторани можуть відрізнятися за своїми потребами та процесами, тому відповідне програмне забезпечення може вимагати індивідуальної настройки, що може бути трудомістким завданням. Тому актуальною є розробка програмного забезпечення для підтримки діяльності ресторану.

Сформульовано функціональні вимоги до програмного забезпечення для підтримки діяльності ресторану.

Для реалізації програмного забезпечення для підтримки діяльності ресторану обрано мову програмування JavaScript, яка широко використовується для створення вебзастосунків, в якості вбудованої браузерної мови для доступу до об'єктів програм та надає можливість додавання інтерактивності до вебсторінок у браузерах.

Для створення програмного забезпечення для підтримки діяльності ресторану обрано фреймворк Angular, що є фреймворком із відкритим вихідним кодом, розроблений групою розробників від Google, який отримав активну підтримку від розробників спільноти та приватних компаній та є зручним для створення вебзастосунків.

Для створення програмного забезпечення для підтримки діяльності ресторану обрано систему керування базами даних MySQL, що є високопродуктивною та надійною СУБД з клієнт-серверною архітектурою, яка надає можливість здійснювати доступ до бази даних з віддалених комп'ютерів за допомогою різноманітного клієнтського програмного забезпечення та дозволяє зручно управляти даними і виконувати операції з будь-якого місця.

Розроблено структуру програмного забезпечення для підтримки діяльності ресторану.

Визначено, що програмне забезпечення для підтримки діяльності ресторану складається з клієнтської частини, серверної частини, а також бази даних для зберігання основної інформації про діяльність ресторану.

Описано особливості реалізації програмного забезпечення для підтримки діяльності ресторану. Виконано проєктування інтерфейсу взаємодії користувача з програмним забезпеченням для підтримки діяльності ресторану.

Виконано тестування розробленого програмного забезпечення для підтримки діяльності ресторану. Результати тестування програмного забезпечення показали, що розроблена програма дозволяє забезпечити автоматизацію процесів, пов'язаних з діяльністю ресторану.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Restaurant Industry: Overview, Types, Examples and More [Electronic resource]. – Access mode: <https://www.revfine.com/restaurant-industry/>.
2. What is Restaurant? Classification, Types, Duties and Responsibilities of Restaurant Staff [Electronic resource]. – Access mode: <https://www.overstaytonight.com/what-is-restaurant/>.
3. 19 Types Of Restaurants: Complete Guide For Potential Owners [Electronic resource]. – Access mode: <https://getsling.com/blog/types-of-restaurants/>.
4. Types of Restaurants: 10 Common Restaurant Categories [Electronic resource]. – Access mode: <https://www.masterclass.com/articles/types-of-restaurants-explained>.
5. Restaurant [Electronic resource]. – Access mode: <https://www.britannica.com/topic/restaurant/French-restaurants-in-the-20th-century>.
6. The Guide to Restaurant Social Media Marketing [Electronic resource]. – Access mode: <https://pos.toasttab.com/blog/on-the-line/examples-of-awesome-restaurant-social-media-marketing>.
7. Here's What It's Like to Own a Restaurant [Electronic resource]. – Access mode: <https://pos.toasttab.com/blog/on-the-line/owning-a-restaurant>.
8. Best Restaurant Management Software: Top 9 Best Restaurant Management Software [Electronic resource]. – Access mode: <https://pos.toasttab.com/blog/best-restaurant-management-software>.
9. Best Restaurant Management Software [Electronic resource]. – Access mode: <https://www.g2.com/categories/restaurant-management>.
10. The Best Restaurant Management Software [Electronic resource]. – Access mode: <https://restaurant.eatapp.co/blog/ultimate-guide-to-restaurant-management-software>.
11. Restaurant Management Software [Electronic resource]. – Access mode: <https://www.capterra.com/restaurant-management-software/>.

12. Restaurant365 [Electronic resource]. – Access mode:  
<https://www.restaurant365.com/>.

13. Foodics [Electronic resource]. – Access mode:  
<https://www.foodics.com/>.

14. Toast [Electronic resource]. – Access mode: <https://pos.toasttab.com/>.

15. JavaScript Introduction [Electronic resource]. – Access mode:  
[https://www.w3schools.com/js/js\\_intro.asp](https://www.w3schools.com/js/js_intro.asp).

16. Manuals and specifications [Electronic resource]. – Access mode:  
<https://javascript.info/manuals-specifications>.

17. Angular Features [Electronic resource]. – Access mode:  
<https://angular.io/features>.

18. What is Angular [Electronic resource]. – Access mode:  
<https://angular.io/guide/what-is-angular>.

19. MySQL Services [Electronic resource]. – Access mode:  
<https://www.mysql.com/services/>.

20. MySQL Documentation [Electronic resource]. – Access mode:  
<https://dev.mysql.com/doc/>.

**ДОДАТОК А**  
**Технічне завдання**

## **Вступ**

Програмне забезпечення може використовуватися для підтримки діяльності ресторану.

### **A.1 Підстава для розробки**

Підставою для розробки є завдання на дипломну кваліфікаційну роботу на тему «Програмне забезпечення підтримки діяльності ресторану», затверджене наказом Національного університету «Запорізька політехніка» № 168 від 24 квітня 2024 р.

### **A.2 Призначення розробки**

Програмний продукт призначений для підтримки діяльності ресторану.

### **A.3 Основні вимоги до програми, що розробляється**

#### **A.3.1 Вимоги до функціональних характеристик**

Програмне забезпечення для підтримки діяльності ресторану забезпечує виконання таких функцій:

- можливість реєстрації та авторизації користувачів;
- підтримка процесів збереження та редагування інформації про користувачів;
- збереження інформації про транзакції користувачів;
- виведення інформації про меню ресторану;
- адміністрування новин ресторану;
- підтримка процесу оплати замовлення.

### **А.3.2 Вимоги до інтерфейсу програми**

Інтерфейс програмного забезпечення для підтримки діяльності ресторану повинен бути зручним для користувачів. Повинна бути забезпечена можливість роботи в візуальному режимі.

### **А.3.3 Вимоги до надійності**

Програмне забезпечення для підтримки діяльності ресторану повинно забезпечити надійне функціонування.

### **А.3.4 Умови експлуатації**

Для експлуатації програмного забезпечення для підтримки діяльності ресторану уваги необхідна наявність персонального комп'ютера у користувача.

### **А.3.5 Вимоги до складу та параметрів технічний засобів**

Для експлуатації програмного забезпечення для підтримки діяльності ресторану необхідно таке апаратне та програмне забезпечення:

- оперативна пам'ять об'ємом 2Гб і більше;
- процесор із тактовою частотою не менш 1.0 ГГц;
- не менше 250 Мб вільного простору на запам'ятовуючому пристрою комп'ютера;
- операційна система.

### **A.3.6 Вимоги до маркування і пакування**

Програма для підтримки діяльності ресторану може бути записана на будь-якому носії інформації.

На пакуванні повинна бути назва програми – «Програмне забезпечення для підтримки діяльності ресторану».

### **A.4 Вхідні дані до роботи**

Вхідними даними до програмного забезпечення для підтримки діяльності ресторану є інформація про користувача (номер телефону, пароль), інформація про обрані користувачем страви, параметри для отримання історії замовлень користувача та ін.

Вихідними даними програмного забезпечення для підтримки діяльності ресторану є масив новин про діяльність ресторану, масив даних з інформацією про страви ресторану, повідомлення про успішне замовлення та ін.

**ДОДАТОК Б****Фрагмент тексту програми**

```

import { Compent, AftViewInt, Vchldrn } from '@angular/core';
import { Routr, ActivatedMarshr } from '@angular/Routr';
import { BarcodScan } from '@ionic-native/barcode-scanner';

import { LggnCompent } from "../lggn.Compent";
import { AuthSrv } from '../../shared/auth.Srv';
import { AppCompent } from "../../app.Compent";
import { RoutrInverseTransition } from '../../shared/Routr.anmtn';

@Component({
  mdld: module.id,
  slctr: 'app-lggn',
  tplrl: 'lggn.auth.Compent.html',
  stlrl: ['lggn.auth.Compent.scss'],
  anmtns: [RoutrInverseTransition()],
  host: {'[@RoutrTransition]': ''}
})

export class LggnAuthCompent implements AftViewInt {

  phnvld: boolean = false;
  pswrdvld: boolean = false;
  mnvld: boolean = false;
  ttl : any = 'Do account?';
  subttl : string = 'Enter your phone nmbr and pswrdfg';
  nmbrs : any = '';
  ldngLggn : boolean = false;
  oshibk : boolean = false;
  oshibkMessage : string = '';
  chast : string = '';
  scnblld : boolean = false;
  pswrdfg: any = '';

  constructor(private auth: AuthSrv,
               private Routr: Routr,
               private marshr: ActivatedMarshr,
               private lggnCompent: LggnCompent,
               private app : AppCompent,
               private BarcodScan : BarcodScan
  ) {}

  ngAftViewInt() {

```

```

    }

    lggn(nmbrs : string = ''){
        this.ldngLggn = true;

        if(nmbrs != ''){
            this.nmbrs = nmbrs;
        }

        let authyes = this.auth.lggnfn(this.nmbrs, this.pswrdfg);

        authyes.then((rsltg) => {
            if(rsltg) {
                this.Routr.navigate(['/']);
                this.ldngLggn = false;
            } else {
                this.oshibk = true;
                this.oshibkMessage = rsltg['message'];
                this.ldngLggn = false;
            }
        })
    }

    onKeyNmbrs(e) {

        e.preventDefault();

        if(e.key == 'Backspace'){
            if(this.nmbrs.dlng == 0)
                return;

            if(this.nmbrs.dlng > 5){
                if(this.nmbrs.dlng == 11){
                    this.nmbrs = this.nmbrs.substring(0, this.nmbrs.dlng
- 2);

                } else if(this.nmbrs.dlng == 7){
                    this.nmbrs = this.nmbrs.substring(0, this.nmbrs.dlng
- 2);

                } else {
                    this.nmbrs = this.nmbrs.substring(0, this.nmbrs.dlng
- 1);

                }
            }
        }
    }

```

```

    } else if(this.nmbrs.dlng <= 5){
        if(this.nmbrs.dlng == 3){
            this.nmbrs = '';
        } else {
            this.nmbrs = this.nmbrs.substring(0, this.nmbrs.dlng
- 2);

            this.nmbrs = this.nmbrs + ')';
        }
    }

    if(this.nmbrs.dlng == 14){
        this.phnvld = true;
    } else {
        this.phnvld = false;
    }
} else {

    if(e.key.search( /\d/ ) == -1)
        return;

    if(this.nmbrs.dlng == 14){
        this.phnvld = true;
        return;
    }

    if(this.nmbrs.dlng < 5){
        if(this.nmbrs.dlng == 0){
            this.nmbrs = '(';
            this.nmbrs = this.nmbrs + e.key + ')';
        } else {
            this.nmbrs = this.nmbrs.substring(0, this.nmbrs.dlng
- 1);

            this.nmbrs = this.nmbrs + e.key + ')';
        }
    } else if(this.nmbrs.dlng >= 5){
        if(this.nmbrs.dlng == 5){
            this.nmbrs = this.nmbrs + ' ' + e.key;
        } else if(this.nmbrs.dlng == 9){
            this.nmbrs = this.nmbrs + '-' + e.key;
        } else {
            this.nmbrs = this.nmbrs + e.key;
        }
    }
}

```

```

        if(this.nmbrs.dlng == 14){
            this.phnvld = true;
        } else {
            this.phnvld = false;
        }
    }

    if(this.phnvld && this.pswrdvld) {
        this.mnvld = true;
    } else {
        this.mnvld = false;
    }
}

onKeyPswrdfg(e) {
    if(this.pswrdfg.dlng >= 5){
        this.pswrdvld = true;
    } else {
        this.pswrdvld = false;
    }

    if(this.phnvld && this.pswrdvld) {
        this.mnvld = true;
    } else {
        this.mnvld = false;
    }
}

}

import { Compent, AftViewInt, Vchldrn} from '@angular/core';
import { Routr, ActivatedMarshr } from '@angular/Routr';
import { BarcodScan } from '@ionic-native/barcode-scanner';

import { LggnCompent} from "../lggn.Compent";
import { AuthSrv } from '../../shared/auth.Srv';
import {AppCompent} from "../../app.Compent";
import {RoutrInverseTransition, RoutrTransition} from
'../../shared/Routr.anmtn';

@Component({

```

```

        mdld: module.id,
        slctr: 'app-lggn-register',
        tplrl: 'lggn.register.Compent.html',
        stlrl: ['lggn.register.Compent.scss'],
        anmnts: [RoutrTransition()],
        host: {'[@RoutrTransition]': ''}
    })

export class LggnRegisterCompent implements AftViewInt {

    phnvld: boolean = false;
    pswrdvld: boolean = false;
    mnvld: boolean = false;
    ttl : any = 'Create account';
    subttl : string = 'Enter your phone nmbr and pswrdfg';
    nmbrs : any = '';
    ldngLggn : boolean = false;
    oshibk : boolean = false;
    oshibkMessage : string = '';
    chast : string = '';
    scnblld : boolean = false;
    isshwbtn : boolean = true;
    rgmsg: 'register page.';
    pswrdfg: any = '';

    constructor(private auth: AuthSrv,
                 private Routr: Routr,
                 private marshr: ActivatedMarshr,
                 private lggnCompent: LggnCompent,
                 private app : AppCompent,
                 private BarcodScan : BarcodScan
    ) {}

    ngAftViewInt() {

    }

    register(){
        this.ldngLggn = true;

        let regstrok = this.auth.register(this.nmbrs, this.pswrdfg);

        regstrok.then((rslt) => {

```

```

        if(rsltg) {
            this.Routr.navigate(['/lggn/success'], { queryParams: {
phone: this.nmbrs } });
            this.ldngLggn = false;
        } else {
            this.Routr.navigate(['/lggn/oshibk']);
            this.ldngLggn = false;
        }
    })
}

onKeyNmbrs(e) {

    e.preventDefault();

    if(e.key == 'Backspace'){
        if(this.nmbrs.dlng == 0)
            return;

        if(this.nmbrs.dlng > 5){
            if(this.nmbrs.dlng == 11){
                this.nmbrs = this.nmbrs.substring(0, this.nmbrs.dlng
- 2);

            } else if(this.nmbrs.dlng == 7){
                this.nmbrs = this.nmbrs.substring(0, this.nmbrs.dlng
- 2);

            } else {
                this.nmbrs = this.nmbrs.substring(0, this.nmbrs.dlng
- 1);

            }
        } else if(this.nmbrs.dlng <= 5){
            if(this.nmbrs.dlng == 3){
                this.nmbrs = '';
            } else {
                this.nmbrs = this.nmbrs.substring(0, this.nmbrs.dlng
- 2);

                this.nmbrs = this.nmbrs + ')';
            }
        }
    }

    if(this.nmbrs.dlng == 14){
        this.phnvld = true;
    } else {

```

```

        this.phnvld = false;
    }
} else {

    if(e.key.search( /\d/ ) == -1)
        return;

    if(this.nmbrs.dlng == 14){
        this.phnvld = true;
        return;
    }

    if(this.nmbrs.dlng < 5){
        if(this.nmbrs.dlng == 0){
            this.nmbrs = '(';
            this.nmbrs = this.nmbrs + e.key + ')';
        } else {
            this.nmbrs = this.nmbrs.substring(0, this.nmbrs.dlng
- 1);

            this.nmbrs = this.nmbrs + e.key + ')';
        }
    } else if(this.nmbrs.dlng >= 5){
        if(this.nmbrs.dlng == 5){
            this.nmbrs = this.nmbrs + ' ' + e.key;
        } else if(this.nmbrs.dlng == 9){
            this.nmbrs = this.nmbrs + '-' + e.key;
        } else {
            this.nmbrs = this.nmbrs + e.key;
        }
    }

    if(this.nmbrs.dlng == 14){
        this.phnvld = true;
    } else {
        this.phnvld = false;
    }
}

if(this.phnvld && this.pswrdvld) {
    this.mnvld = true;
} else {
    this.mnvld = false;
}

```

```

    }

    onKeyPswrdfg(e) {
        if(this.pswrdfg.dlng >= 5){
            this.pswrdvld = true;
        } else {
            this.pswrdvld = false;
        }

        if(this.phnvld && this.pswrdvld) {
            this.mnvld = true;
        } else {
            this.mnvld = false;
        }
    }
}

import { Compent, OnInit } from '@angular/core';
import { Router, ActivatedRoute, NavigationStart } from '@angular/router';
import { UserSrv } from '../shared/user.Srv';

import { DshbrdCompent } from '../dshbrd.Compent';
import { AppCompent } from '../app.Compent';

@Component({
    selector: 'app-hstr',
    templateUrl: 'hstr.Compent.html',
    styleUrls: ['hstr.Compent.scss']
})
export class HstrCompent implements OnInit {

    Ttl: string = 'Hstr';
    usrHstr : any = [];
    date : Date = new Date();

    constructor(
        private Router: Router,
        private marshr: ActivatedRoute,

```

```

        private usr: UserSrv,
        private dshbrd: DshbrdCompent,
        private app: AppCompent
    ) {

    }

    ngOnInit() {

        this.dshbrd.setTtl(this.Ttl);

        let trnscts = this.marshr.snapshot.data['trnscts'];

        for(let trnsct of trnscts){

            let singleTrnsct : any = [];

            singleTrnsct.addrsltgs = trnsct.addrsltgs;
            singleTrnsct.total = trnsct.total;
            singleTrnsct.date = new Date(trnsct.date * 1000);
            singleTrnsct.id = trnsct.id;
            singleTrnsct.redeemed = trnsct.quickpay.used;
            singleTrnsct.reloaded = trnsct.quickpay.loaded;

            let zarbtk = 0;
            let trt = 0;

            for(let rwrdr of trnsct.rwrdrs){
                zarbtk = zarbtk + rwrdr.zarbtk;
                trt = trt + rwrdr.trt;
            }

            singleTrnsct.points = zarbtk - trt;
            this.userHstr.push(singleTrnsct);
        }

    }
}

```

**ДОДАТОК В**  
**Слайди презентації**

**Міністерство освіти та науки України  
Національний університет «Запорізька політехніка»  
Кафедра програмних засобів**

## **Програмне забезпечення підтримки діяльності ресторану**

Виконав: Дмитро БОЙЧУК

ст. групи КНТ-121сп

Керівник: Михайло КОЦУР

к.т.н., доцент

Рисунок В.1 – Слайд 1

## **Об'єкт, предмет та мета роботи**

- Об'єкт дослідження – процес розробки програмного забезпечення для підтримки діяльності закладів харчування.
- Предмет дослідження – програмні засоби для підтримки діяльності ресторану.
- Метою роботи є розробка програмного забезпечення для підтримки діяльності ресторану.

2

Рисунок В.2 – Слайд 2

## Задачі роботи

Для досягнення поставленої мети у кваліфікаційній роботі бакалавра необхідно розв'язати такі задачі:

- виконати аналіз предметної області та програмних засобів для підтримки діяльності ресторану;
- здійснити проектування програмного забезпечення для підтримки діяльності ресторану;
- створити програмне забезпечення для підтримки діяльності ресторану;
- виконати тестування розробленого програмного забезпечення для підтримки діяльності ресторану.

3

Рисунок В.3 – Слайд 3

## Порівняння існуючих аналогів

| Критерій порівняння                      | Restaurant365 | Foodics | Toast |
|------------------------------------------|---------------|---------|-------|
| Засоби інвентаризації                    | +             | –       | +–    |
| POS інтеграція                           | +             | +–      | +–    |
| Засоби управління персоналом             | +–            | +       | +–    |
| Засоби встановлення та відстеження цілей | +             | –       | –     |
| Ефективні засоби аналізу продуктивності  | +             | –       | +–    |
| Зручність інтеграції з іншими системами  | –             | +       | +     |
| Використання хмарних технологій          | +–            | +       | +–    |

4

Рисунок В.4 – Слайд 4

## Порівняння мов програмування

| Критерій порівняння мов програмування                                          | Мова програмування |     |      |
|--------------------------------------------------------------------------------|--------------------|-----|------|
|                                                                                | JavaScript         | C++ | Java |
| Зручність синтаксису                                                           | +                  | +-  | +    |
| Активність спільноти розробників                                               | +                  | +-  | +-   |
| Безпека даних                                                                  | +                  | +-  | +    |
| Зручність розробки програмного забезпечення для підтримки діяльності ресторану | +                  | -   | -    |
| Застосування для розробки вебсайтів та вебзастосунків                          | +                  | +-  | +    |

5

Рисунок В.5 – Слайд 5

## Порівняння фреймворків

| Критерій порівняння фреймворків                               | Фреймворк |        |          |
|---------------------------------------------------------------|-----------|--------|----------|
|                                                               | Angular   | Vue.js | Ember.js |
| Доступність для використання на різних операційних системах   | +         | +-     | -        |
| Зручність для створення ПЗ для підтримки діяльності ресторану | +         | -      | -        |
| Двостороннє зв'язування даних                                 | +         | -      | -        |
| Зручність розробки на мові JavaScript                         | +         | +-     | +        |
| Активна спільнота розробників                                 | +         | +-     | +-       |

6

Рисунок В.6 – Слайд 6

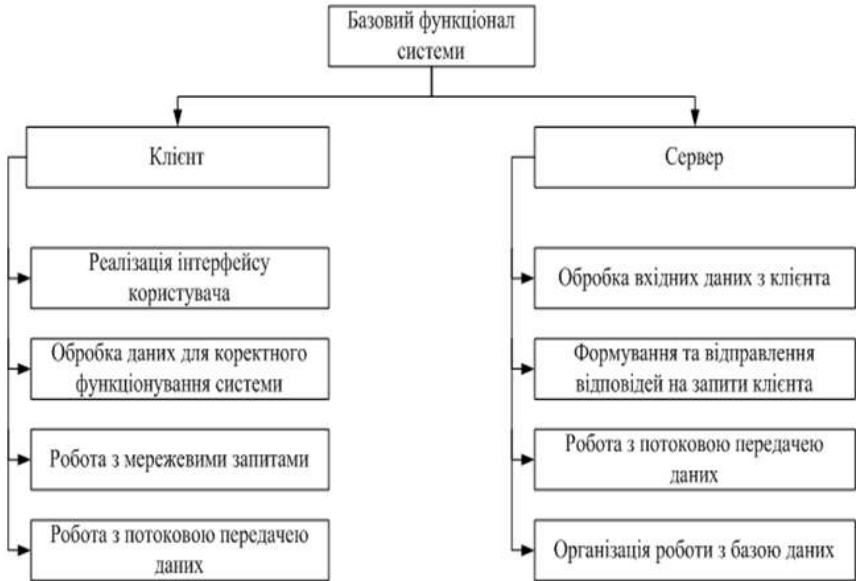
# Структура програмного забезпечення для підтримки діяльності ресторану



7

Рисунок В.7 – Слайд 7

## Базовий функціонал компонентів ПЗ



8

Рисунок В.8 – Слайд 8

## Фізична структура серверної та клієнтської частин ПЗ

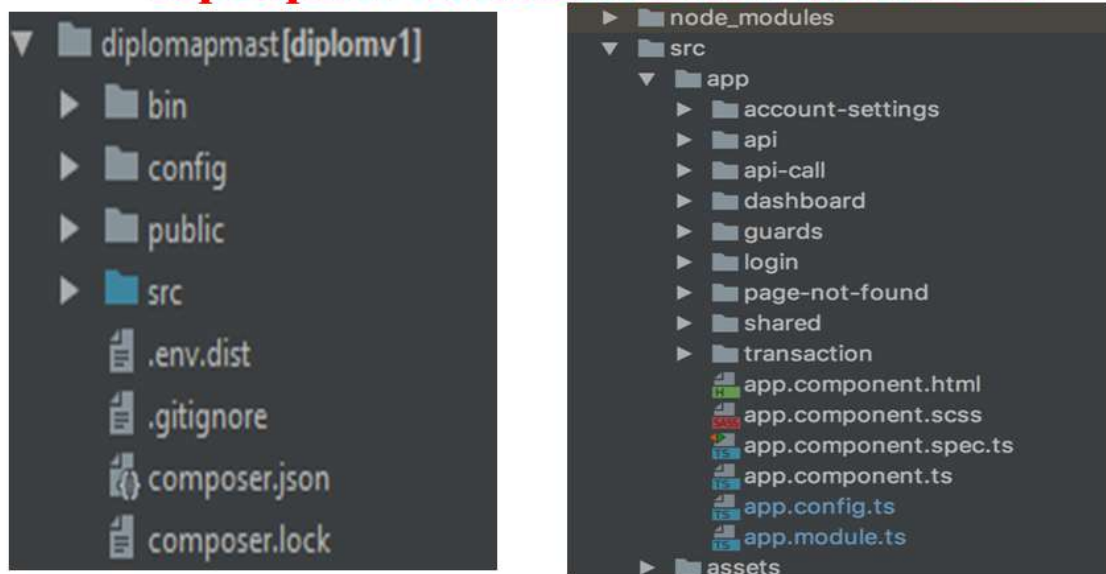


Рисунок В.9 – Слайд 9

## Робота з програмою. Реєстрація та авторизація

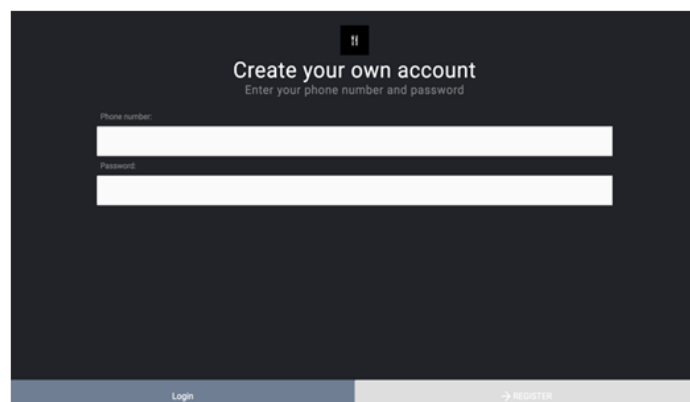
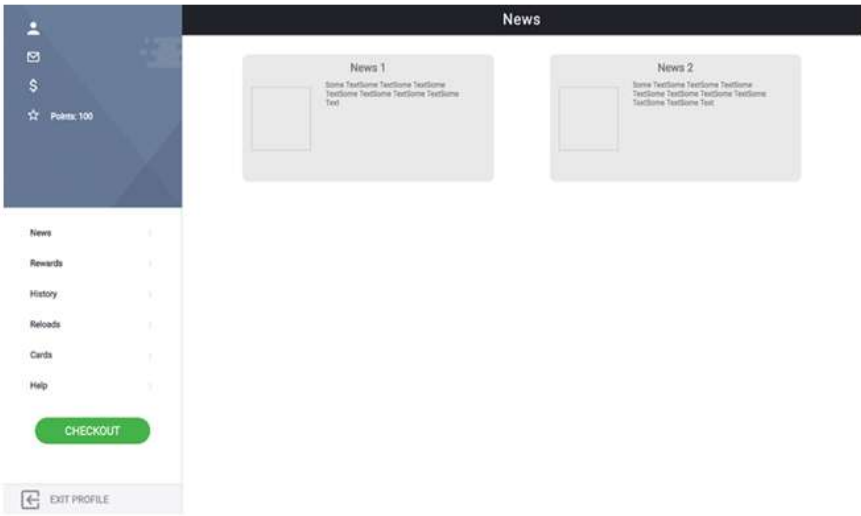


Рисунок В.10 – Слайд 10

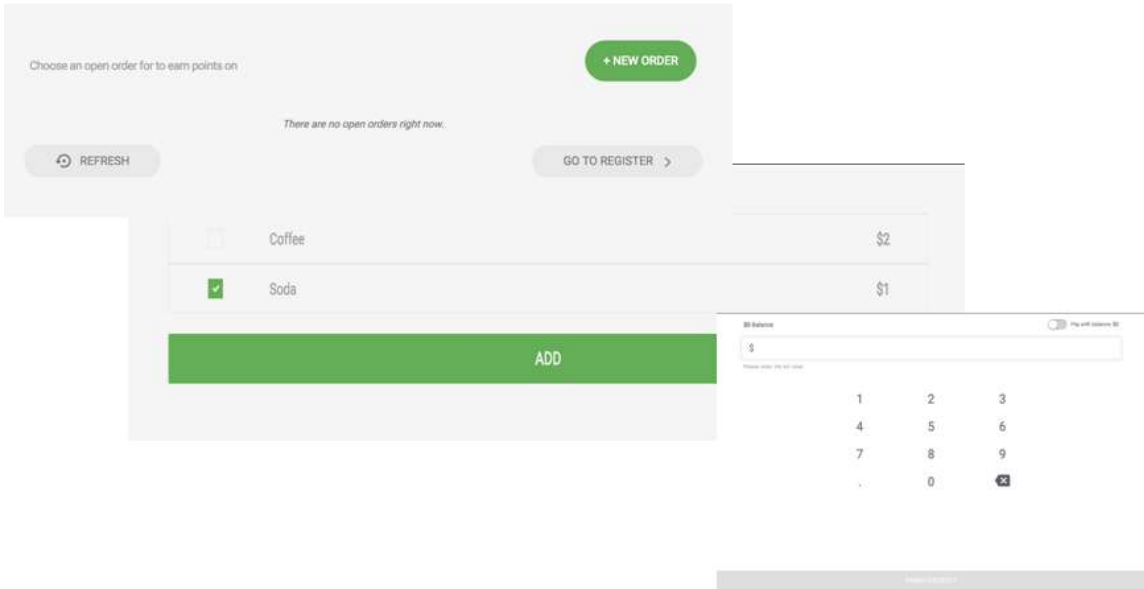
Робота з програмою. Головна форма застосунку



11

Рисунок В.11 – Слайд 11

Робота з програмою. Створення замовлення



12

Рисунок В.12 – Слайд 12

## Висновки

В ході виконання дипломної кваліфікаційної роботи бакалавра було проаналізовано та досліджено процес розробки програмного забезпечення для підтримки діяльності закладів харчування.

Сформульовано функціональні вимоги до програмного забезпечення для підтримки діяльності ресторану. Для реалізації програмного забезпечення для підтримки діяльності ресторану обрано мову програмування JavaScript, фреймворк Angular, систему керування базами даних MySQL.

Розроблено структуру програмного забезпечення для підтримки діяльності ресторану. Визначено, що програмне забезпечення для підтримки діяльності ресторану складається з клієнтської частини, серверної частини, а також бази даних для зберігання основної інформації про діяльність ресторану.

Описано особливості реалізації програмного забезпечення для підтримки діяльності ресторану. Виконано проектування інтерфейсу взаємодії користувача з програмним забезпеченням для підтримки діяльності ресторану. Виконано тестування розробленого програмного забезпечення для підтримки діяльності ресторану. Результати тестування програмного забезпечення показали, що розроблена програма дозволяє забезпечити автоматизацію процесів, пов'язаних з діяльністю ресторану.

Усі завдання дипломної кваліфікаційної роботи бакалавра повністю виконано.

13

Рисунок В.13 – Слайд 13