

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВАДИМА ГЕТЬМАНА
Навчально-науковий інститут
«Інститут інформаційних технологій в економіці»
Кафедра інформаційних систем в економіці

ОСВІТНЬО ПРОФЕСІЙНА ПРОГРАМА «КОМП'ЮТЕРНІ НАУКИ»

галузь знань 12 «Інформаційні технології»

спеціальність 122 «Комп'ютерні науки»

Форма навчання: денна

КВАЛІФІКАЦІЙНИЙ БАКАЛАВРСЬКИЙ ПРОЕКТ

на тему

СМАРТ-СЕРВІС ЗДОРОВОГО ХАРЧУВАННЯ

здобувача Мохаммада Даніеля Наїмовича

Науковий керівник:

д.е.н., проф.

Ріппа С.П.

Проект допущений до захисту

перед екзаменаційною комісією з

атестації здобувачів вищої освіти

завідувач кафедри:

к.е.н., доцент.

Тішков Б.О.

Київ 2024

Конкретні завдання, які здобувач повинен виконати для досягнення поставленої мети:

У розділі I

1. Дослідження і характеристика предметної області. Збір інформації та вивчення матеріалів за тематикою кваліфікаційного бакалаврського проекту.
2. Аналіз літератури та існуючих підходів до організації смарт-сервісів здорового харчування.

У розділі II

1. Аналіз і специфікація вимог до інформаційної системи смарт-сервісу здорового харчування.
2. Аналіз та визначення функціональних і нефункціональних вимог до смарт-сервісу.
3. Постановка задач і побудова алгоритму їх розв'язання.
4. Побудова інформаційної моделі задач смарт-сервісу.
5. Визначення діаграм блоків, класів, трасування та інших складових інф. моделі.

У розділі III

1. Визначення архітектура інформаційного забезпечення смарт-сервісу здорового харчування.
2. Визначення даталогічної моделі БД смарт-сервісу.
3. Розробка програмного і технічного забезпечення смарт-сервісу здорового харчування.
4. Апробація і тестування смарт-сервісу здорового харчування.

**Завдання підготував
науковий керівник**

(підпис)

Ріппа Сергій Петрович

(ініціали, прізвище)

« 06 » лютого 2024 р.

**Завдання одержав
здобувач**

(підпис)

Мохаммада Даніеля Наїмовича

(ініціали, прізвище)

« 07 » лютого 2024 р.

Відгук
на кваліфікаційний бакалаврський проект
здобувача вищої освіти КНЕУ
Мохаммада Даніеля Наїмовича
на тему: «РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ
СМАРТ-СЕРВІС ЗДОРОВОГО ХАРЧУВАННЯ»

Актуальність теми дослідження. Тема кваліфікаційного бакалаврського проекту є актуальною, оскільки присвячена дослідженню та практичному застосуванню сучасних технологій діджиталізації в сфері збалансованого і персоналізованого харчування, що є важливою складовою розвитку еко-системи орієнтованих на найкращі технології здорової дієтології.

Наукова новизна. Представлений проект містить елементи наукової новизни, зокрема: автор детально проаналізував сучасні ІТ-сервіси здорового харчування, їх переваги та недоліки і запропонував свій, персоналізований для користувачів підхід формування збалансованого харчування та вибору відповідних рецептів справ і меню з елементами смарт-сервісів.

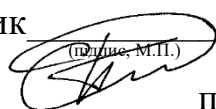
Якість проведеного аналізу проблеми, текстового та табличного матеріалу. Зміст представлено до захисту проекту, що містить результати опрацювання теоретичних аспектів розробки харчових ІТ-сервісів, а також практичні результати проектування та розробки архітектури ІС для пошуку страв, формування замовлень з врахуванням певних обмежень і рекомендацій дієтологів, генерації звіту, адміністрування замовлень тощо. Основний зміст роботи вдало доповнено таблицями, рисунками і зразками веб-орієнтованих документів.

Обґрунтованість пропозицій і висновків та їх практична цінність. Пропозиції щодо застосування методів та інструментів розробки смарт-сервісів здорового харчування є достатньо обґрунтованими; висновки, зроблені автором за результатами виконаної роботи, узагальнюють її основний зміст. Результати виконання кваліфікаційної роботи можуть знайти застосування при розробці веб-ресурсів установ та організацій в сфері здорового харчування, є працююча бета-версія системи.

Наявність недоліків. Суттєвих зауважень до проекту не має, є незначні відхилення від норм форматування.

Загальний висновок і оцінка роботи. Кваліфікаційний бакалаврський проект на тему «РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ СМАРТ-СЕРВІС ЗДОРОВОГО ХАРЧУВАННЯ» у цілому відповідає встановленим вимогам та рекомендується до захисту, а її автору Мохаммаду Даніелю Наїмовичу може бути присвоєний освітньо-кваліфікаційний рівень «бакалавр» в галузі знань 12 «Інформаційні технології» за спеціальністю 122 «Комп'ютерні науки» освітньо-професійної програми «Інформаційні управляючі системи і технології».

Науковий керівник


(Підпис: М.П.)

С. П. Ріппа, д.е.н., професор,

професор кафедри інформаційних систем в економіці
Навчально-наукового інституту "Інститут
інформаційних технологій в економіці" КНЕУ

РЕЦЕНЗІЯ

на кваліфікаційний бакалаврський проект
Мохаммада Даніеля Наїмовича на тему
«Розробка інформаційної системи смарт-сервіс здорового харчування»
за спеціальністю 122 «Комп'ютерні науки»,
ОПП «Інформаційні управляючі системи і технології»

Актуальність теми: Підготовлений до захисту кваліфікаційний бакалаврський проект присвячений актуальній темі, оскільки може бути використаний в процесі реформування інформаційного простору кулінарних сервісів і послуг в Україні і ця сфера ІТ-послуг є важливою в сучасних складних умовах для вітчизняної економіки.

Характеристика роботи. Рецензований дипломний проект складається стандартним чином із трьох розділів і десятих додатків. У першому розділі роботи наведений аналіз існуючого веб-середовища здорового харчування в Україні і відповідної літератури, другий розділ присвячений дослідженню постановці задач і моделюванню смарт-сервісу харчування, третій розділ зосереджений на проектних рішеннях щодо інформаційного, технічного та програмного забезпеченні проекту. За своїм змістом робота відповідає обраній темі та сформульованим задачам, її оформлення також вкладається в нормативи.

Дипломний проект не позбавлений окремих несуттєвих недоліків, до яких можна віднести певну незавершеність матеріалів веб-аналітики у сфері порівняння кулінарних ІТ-сервісів в Україні і в світі, хотілось би побачити більш розгорнуті висновки здобувача, орієнтовані на сучасний контекст медіа-простору кулінарних послуг, є несуттєві помилки і описки в оформленні дипломної роботи.

Загальний висновок. Можна вважати, що представлений на рецензування кваліфікаційний бакалаврський проект на здобуття ступеня бакалавра свідчить про достатню кваліфікацію автора щодо обраної тематики, проект виконаний на належному рівні, за змістом та практичною цінністю відповідає вимогам МОН, а її автору, Мохаммаду Даніелю Наїмовичу може бути присвоєний освітньо-кваліфікаційний рівень «бакалавр» за спеціальністю «Комп'ютерні науки» освітньо-професійної програми «Інформаційні управляючі системи і технології».

Рецензент:



Погореловська І.Д., к.е.н., доцент,
доцент кафедри комп'ютерних та інформаційних
технологій і систем факультету фінансів та
цифрових технологій Державного податкового
університету

«07» червня 2024 р.

АНОТАЦІЯ

випускного бакалаврського проекту студента 4 курсу
Навчально-наукового інституту «Інститут інформаційних технологій в
економіці» **Мохаммада Даніеля Наїмовича**, виконаної на тему:
«Розробка інформаційної системи смарт-сервіс здорового харчування»
Київ: кафедра інформаційних систем в економіці, 2024 р.

Випускний бакалаврський проект присвячений актуальній проблемі сьогодення, а саме здоровому та збалансованому харчуванню, підбору програм за потребами клієнтів.

Випускний бакалаврський проект складається з трьох розділів, логічно пов'язаних між собою.

В першому розділі дана характеристика предметної галузі й об'єкта дослідження, наведено аналіз задач, що розв'язуються, а також наводиться перелік існуючих програмних засобів даної предметної галузі.

Другий розділ є проектним і присвячений обґрунтуванню методу проектування системи, розробленню її архітектури, виконанню постановки та розробленню алгоритма розв'язання задач.

Третій розділ – конструктивний. Тут наведено інформаційне, технічне програмне та організаційне забезпечення. Висвітлені питання організації інформаційного забезпечення: розроблена структура інформаційного забезпечення, описані та наведені в додатках приклади сторінок та частина програмного коду. Вказані структури інформаційних масивів, які використовуються під час вирішення задач.

Висновки містять рекомендації щодо доцільності розроблення та впровадження системи.

РЕФЕРАТ

Випускний бакалаврський проект містить 80 сторінок, 3 таблиці, 45 рисунків, список літератури з 22 найменувань, 10 додатків.

РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ СМАРТ-СЕРВІС ЗДОРОВОГО ХАРЧУВАННЯ

Перелік ключових слів: Смарт-сервіс, веб-застосунок, веб-сервіс JavaScript, Vue, Pinia, Firestore.

Предметом дослідження є розробка та впровадження інноваційного смарт-сервісу здорового харчування.

Об'єктом дослідження є створення і функціонування веб-сервісу для замовлення здорового харчування.

Мета випускного бакалаврського проекту полягає в створенні веб-сервісу для замовлення їжі та контролю харчування.

Завданням випускного бакалаврського проекту є аналіз конкурентів у сфері здорового харчування, розробка інтерфейсу та функціоналу веб-застосунку.

Апаратні та програмні засоби, що використовувались на при проектуванні: ПК з ОС Windows 10, середовище розробки Visual Studio Code, графічний редактор Figma, браузер Google Chrome.

Результати досягнуті в процесі роботи: розроблений і реалізований веб-застосунок, який перебуває на етапі майже готового для впровадження на підприємстві.

Одержані результати можуть бути використані власниками підприємств які спеціалізуються на приготуванні здорового харчування для зручного замовлення клієнтами.

Рік виконання випускного бакалаврського проекту - 2024

Рік захисту випускного бакалаврського проекту - 2024

ЗМІСТ

ВСТУП	9
РОЗДІЛ 1 ХАРАКТЕРИСТИКА ТА АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ	11
1.1 Характеристика предметної галузі та об'єкта дослідження.....	11
1.2 Аналіз літературних джерел та практичного досвіду використання ІС і технологій в предметній галузі.....	14
РОЗДІЛ 2 РОЗРОБКА ВИМОГ І МОДЕЛЮВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ	18
2.1 Аналіз і специфікація вимог до інформаційної системи	18
2.2 Постановка задачі та алгоритм розв'язання задачі	21
2.3 Моделювання інформаційної системи.....	30
РОЗДІЛ 3 ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ КОМПОНЕНТІВ СИСТЕМИ	37
3.1 Інформаційне забезпечення	37
3.2 Технічне забезпечення.....	42
3.3 Програмне забезпечення	43
3.4 Організаційне забезпечення.....	44
3.5 Результати реалізації інформаційної системи.....	46
ВИСНОВКИ.....	53
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	55
ДОДАТКИ.....	58

ВСТУП

Метою проектування даної інформаційної системи є спрощення пошуку та купівлі здорової їжі через мережу інтернет. Розробка даної проблеми перебуває на рівні майже готового онлайн-сервісу.

З розвитком інформаційних технологій і зростаючим інтересом до здорового способу життя, питання правильного харчування набуло значного значення. Смарт-сервіси здорового харчування, які використовують алгоритми штучного інтелекту для надання персоналізованих рекомендацій, стають все більш актуальними в умовах сучасного суспільства. Такий підхід дозволяє враховувати індивідуальні потреби та особливості кожної людини, що сприяє ефективнішому контролю за якістю харчування та здоров'ям. Актуальність теми дослідження обумовлена зростаючою кількістю хронічних захворювань, пов'язаних з неправильним харчуванням, і необхідністю створення інструментів, які допоможуть людям приймати обґрунтовані рішення щодо свого раціону.

Об'єктом дослідження є смарт-сервіси, які спрямовані на забезпечення здорового харчування за допомогою технологій штучного інтелекту.

Предметом дослідження є методи та алгоритми, що використовуються для аналізу даних про стан здоров'я користувачів і формування персоналізованих рекомендацій щодо харчування.

Метою дослідження є розробка концепції та прототипу смарт-сервісу здорового харчування, який надаватиме користувачам індивідуальні рекомендації на основі аналізу їхніх медичних даних і харчових звичок.

Для досягнення цієї мети було поставлено наступні завдання:

- Провести огляд сучасних смарт-сервісів для здорового харчування та визначити їхні переваги й недоліки.
- Дослідити основні проблеми, з якими стикаються користувачі під час використання існуючих сервісів.
- Розробити методологію створення смарт-сервісу здорового харчування, що враховуватиме індивідуальні потреби користувачів.

- Створити функціональний прототип смарт-сервісу та провести його тестування.
- Оцінити ефективність запропонованого сервісу та розробити рекомендації щодо його вдосконалення.

Для виконання завдань дослідження було використано такі методи: аналіз літературних джерел, моделювання, програмування.

Робота складається з вступу, трьох розділів, висновків, списку використаної літератури та додатків. Перший розділ присвячений аналізу існуючих смарт-сервісів для здорового харчування та визначенню їх переваг і недоліків. Другий розділ містить опис методології розробки смарт-сервісу, включаючи алгоритми та моделі, що використовуються для аналізу даних. Третій розділ присвячений розробці прототипу смарт-сервісу, його тестуванню та оцінці ефективності.

РОЗДІЛ 1 ХАРАКТЕРИСТИКА ТА АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ

1.1 Характеристика предметної галузі та об'єкта дослідження

У світі з кожним днем все більше зростає кількість користувачів інтернету, що призводить до збільшення числа клієнтів, які віддають перевагу онлайн-замовленням. Цей розвиток інтернет-торгівлі має велике значення у формуванні інформаційного суспільства та його впливу на економіку.

Змагання на ринку означає конкурентну боротьбу за покупців, де товари і послуги намагаються привернути увагу клієнтів якістю та адекватною ціною. Після аналізу конкурентів стало очевидним, що інші служби доставки здорового харчування також становлять конкуренцію для цього інтернет-застосунку, спонукаючи потенційних клієнтів обирати їхні послуги [1].

Розробка смарт-сервісу стає все більш актуальною, оскільки надає можливість клієнтам отримувати зручний доступ до інформації про продукти і послуги компанії. Правильно спроектований дизайн інтерфейсу та користувацького досвіду сайту має велике значення для подальшого успіху компанії. Недостатньо професійно розроблений смарт-сервіс може негативно позначитися на бізнесі, тому що він не забезпечить задоволення клієнтів та може призвести до втрати репутації і конверсії. Інтернет-застосунки стають невід'ємною частиною життя, оскільки вони дозволяють зручно замовляти товари та послуги без виходу з дому. Окрім того, важливо звертати увагу на збалансоване харчування, яке має позитивний вплив на здоров'я та знижує ризик розвитку захворювань.

Здорове харчування — це не лише хороше самопочуття, позбавлення від багатьох недуг, міцна нервова система та нормалізація маси тіла, але й любов до себе, прагнення до здорового і тривалого життя [2].

Збалансоване харчування означає, що організм отримує достатню кількість поживних речовин, вітамінів і мінералів, щоб зберегти здоров'я та нормально працювати органам. Підтримка імунітету, збереження енергії, запобігання хронічним захворюванням і покращення загального самопочуття є важливими

результатами збалансованого харчування [3].

Дієтичне харчування — це найважливіша фізіологічна вимога нашого організму, що є основою здорового харчування. Людина зберігає здоров'я і не набирає зайвих кілограмів, включаючи дієтичні продукти в щоденний раціон. Зменшення загальної кількості споживаних калорій протягом дня є умовою дотримання низькокалорійного харчування. Для цього не обов'язково дотримуватися суворої дієти або купувати дорогі екзотичні продукти [4].

Важливо розуміти, що підхід до створення смарт-сервісу в галузі харчування повинен бути індивідуалізованим та орієнтованим на потреби людини. Цей підхід дозволяє створити смарт-сервіс, який не лише відповідає стандартам здорового та збалансованого харчування, але й ураховує специфіку та потреби користувачів - доступ до якісних та корисних харчових продуктів та послуг.

Моделювання предметної області - це процес створення абстрактної моделі системи, яку буде використовувати розробник програмного забезпечення. Цей процес передбачає дослідження та аналіз предметної області, що включає в себе ідентифікацію та опис сутностей, які взаємодіють у цій області, а також опис процесів та правил, які регулюють ці взаємодії.

За допомогою моделювання предметної області можна визначити функціональні та нефункціональні вимоги до програмного забезпечення, а також визначити його архітектуру та інтерфейси. Цей процес також дозволяє виявляти можливі проблеми та недоліки в системі до її розробки та запуску, що допомагає уникнути помилок та зберегти ресурси. Моделі, створені під час моделювання предметної області, можуть бути використані як для детального опису проекту програмного забезпечення, так і для комунікації зі стейкхолдерами та замовниками проекту. В результаті, моделювання предметної області є важливим етапом в розробці програмного забезпечення, що допомагає забезпечити успішне та ефективно виконання проекту.

В цьому проекті було використане саме методологія ОРМ. ОРМ (Object-Process Methodology) - це концептуальна мова моделювання та методологія для збору знань і проектування систем [1]. ОРМ діаграми дозволяють розглядати

систему як сукупність процесів та об'єктів, які взаємодіють між собою та зовнішніми системами.

ОРМ діаграми складаються з двох основних типів елементів - об'єктів (Objects) та процесів (Processes). Об'єкти відображають різні складові системи, такі як компоненти, ресурси, агенти або властивості, і зображаються у вигляді прямокутників. Процеси відображають дії, функції або операції, які здійснюються в системі, і зображаються у вигляді еліпсів. ОРМ діаграми використовують стрілки, які вказують на взаємозв'язки між об'єктами та процесами. Напрямок стрілок вказує на потік інформації, ресурсів або впливу між елементами системи.

ОРМ діаграми дозволяють забезпечити розуміння структури та поведінки системи, візуалізувати взаємодії між різними елементами системи та виявляти потенційні проблеми або недоліки. Вони також можуть служити основою для комунікації між різними зацікавленими сторонами, такими як розробники, клієнти, режисери, журналісти та організатори, що допомагає забезпечити взаєморозуміння та співпрацю між різними учасниками проекту.

В результаті моделювання було отримано наступну діаграму (див. рис. 1.1).

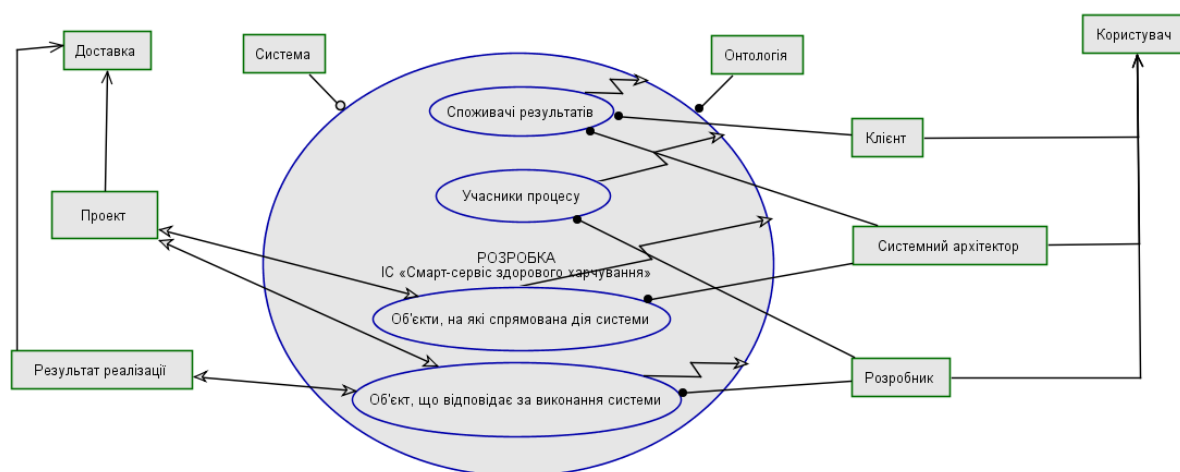


Рисунок 1.1 – Системна діаграма

Джерело: розроблено автором самостійно

1.2 Аналіз літературних джерел та практичного досвіду використання ІС і технологій в предметній галузі

Нещодавнє дослідження харчових звичок у 195 країнах показало, що неправильне харчування є причиною 20% передчасних смертей. Це більше, ніж куріння вбиває [5]. Дослідження показало, що причиною є надлишок солі, а також відсутність фруктів, цільнозернових злаків, горіхів і насіння.

Здоровий раціон задовольняє наші потреби в макро- та мікронутрієнтах (цукру, амінокислот, жирів, вітамінів, мінералів і рослинних антиоксидантах), а також певною мірою запобігає гіпертензії, діабету 2 типу, серцево-судинним хворобам, остеоартриту (нормальна маса тіла — менше навантаження на суглоби), депресії, деменції та деяким типам раку [6].

Для порівняння існуючих інформаційних систем, було взято 3 існуючих веб-сайти з доставки здорового харчування. Всі вони пропонують схожий функціонал та асортимент продуктів, але в жодному з них не було знайдено звичайного меню для клієнтів, які не зацікавлені в повноцінному раціоні. В таблиці 1.1 можна наглядно елементи функціоналу кожного із сервісів.

Таблиця 1.1 – Порівняльний аналіз конкурентів

	Елементи функціоналу	Perfect Balance	Healthy Food	Foodzi
1	Реєстрація з Google	●	●	●
2	Реєстрація з поштою	●	●	●
3	Особистий кабінет	●	●	●
4	Меню звичайне	●	●	●
5	Програми харчування	●	●	●
6	Поле пошуку страв	●	●	●
7	Калькулятор калорій	●	●	●
8	FAQ	●	●	●
9	Інтернаціоналізація (вибір мови)	●	●	●
10	Кошик	●	●	●
11	Про нас	●	●	●
Примітка*		● - присутній функціонал ● - відсутній функціонал		

Джерело: розроблено автором самостійно

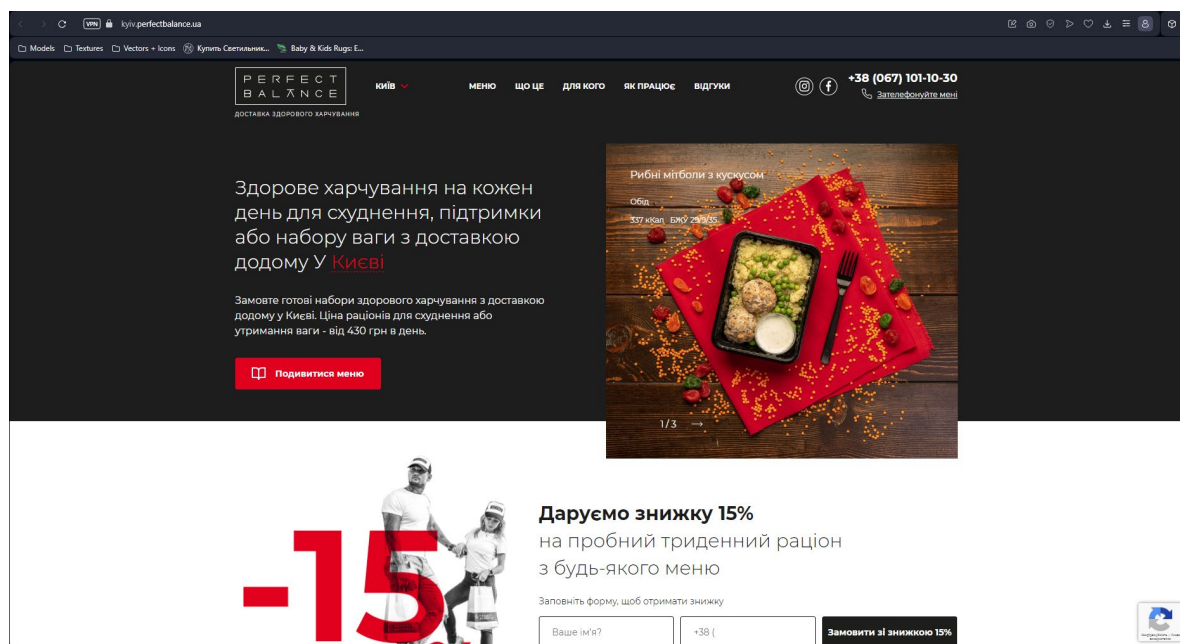


Рисунок 1.2 – Сторінка сайту Perfect balance

Джерело: <https://kyiv.perfectbalance.ua>

Сайт <https://kyiv.perfectbalance.ua> - це онлайн-платформа з збалансованим харчуванням, яка дозволяє замовляти готове меню з урахуванням уподобань клієнтів. Він спрямований на поліпшення здоров'я та підтримку збалансованого харчування, що допомагає знизити ризики захворювань і покращити фізичні та когнітивні можливості організму. На даному сайті присутній такий функціонал:

- замовлення готового меню: клієнти можуть обирати та замовляти готове меню з урахуванням своїх уподобань і потреб;
- урахування харчування: система ураховує збалансованість харчування та допомагає підібрати оптимальні страви для клієнтів;
- підтримка здоров'я: сайт спрямований на поліпшення здоров'я через збалансоване харчування, що допомагає знизити ризики захворювань.

Сайт <https://healthyfood.co.ua> - це онлайн-платформа з здоровим харчуванням, яка пропонує широкий вибір здорових та смачних страв. Основний функціонал сайту включає:

- вибір та замовлення страв: клієнти можуть обирати і замовляти різноманітні страви, які відповідають їхнім харчувальним потребам та уподобанням;

- інформація про продукти: сайт надає докладну інформацію про склад продуктів, їх калорійність, корисні властивості тощо, що допомагає клієнтам зробити осмислені вибори;



Рисунок 1.3 – Сторінка сайту Healthy food

Джерело: <https://healthyfood.co.ua>

- дієтичні рекомендації: крім страв, сайт також пропонує дієтичні поради та рекомендації щодо здорового способу життя та харчування;
- зручний замовлення: клієнти можуть зручно та швидко здійснювати замовлення онлайн, без зайвих турбот і зусиль;

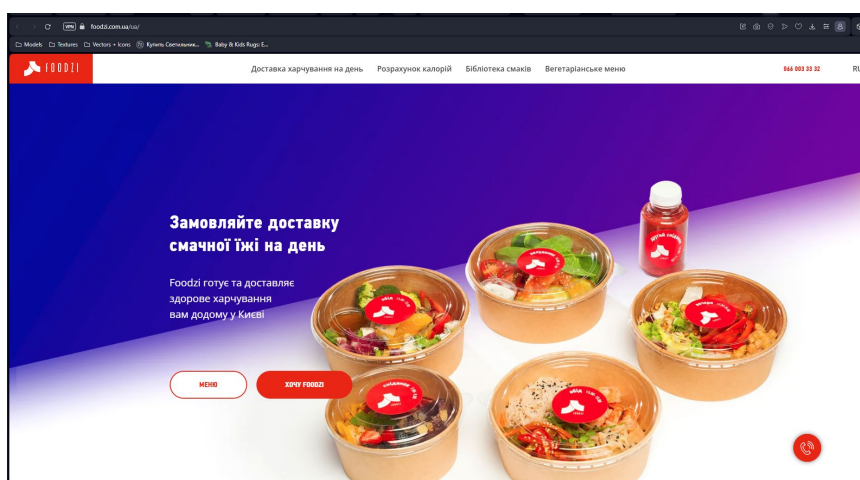


Рисунок 1.4 – Сторінка сайту Foodzi

Джерело: <https://foodzi.com.ua/ua/>

- доставка та обслуговування: сайт пропонує зручну доставку та високоякісне обслуговування клієнтів, що робить процес замовлення ще більш приємним та зручним.

Сайт <https://foodzi.com.ua/ua/> - це онлайн-платформа з продажу здорових та смачних страв, яка має наступний функціонал:

- широкий вибір страв: клієнти можуть обирати з різноманітного асортименту страв, які включають у себе різні дієтичні та здорові варіанти;
- калорійність та склад продуктів: сайт надає докладну інформацію про калорійність та склад кожної страви, допомагаючи клієнтам зробити свідомий вибір;
- зручне замовлення: клієнти можуть легко та швидко замовляти страви онлайн і вказувати свої уподобання та особливості харчування;
- доставка та обслуговування: сайт пропонує зручну доставку страв, а також високоякісне обслуговування клієнтів;
- акції та спеціальні пропозиції: foodzi також пропонує різні акції та спеціальні пропозиції для своїх клієнтів, що робить процес замовлення ще більш вигідним та привабливим.

РОЗДІЛ 2 РОЗРОБКА ВИМОГ І МОДЕЛЮВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

2.1 Аналіз і специфікація вимог до інформаційної системи

Бізнес-вимоги до предметної області смарт-сервісу доставки їжі можуть включати такі аспекти:

1. Платформа для замовлень: створення зручної та інтуїтивно зрозумілої платформи для клієнтів для оформлення замовлень онлайн, можливість реєстрації та авторизації користувачів для зберігання їхньої інформації та історії замовлень.
2. Управління замовленнями: система управління замовленнями для підтримки процесу від отримання замовлення до його виконання та доставки.
3. Управління меню та інгредієнтами: оновлення меню та інгредієнтів на основі наявності товарів та попиту клієнтів.
4. Оплата та фінансові операції: інтеграція з різними платіжними системами для забезпечення безпечної оплати замовлень онлайн, облік фінансових операцій, включаючи оплату замовлень, відшкодування, промо-акції та знижки.
5. Аналітика та звітність: система аналітики та звітності для відстеження продажів, попиту, ефективності маркетингових кампаній та іншої важливої інформації, звіти та аналізи для прийняття рішень щодо розвитку бізнесу, оптимізації процесів та покращення якості обслуговування.

Функціональна вимога - це формулювання того, як система повинна поводитися. Він визначає, що повинна робити система, щоб задовольнити потреби або очікування користувача. Функціональні вимоги можна розглядати як функції, які виявляє користувач. Вони відрізняються від нефункціональних вимог, які визначають, як система повинна працювати всередині (наприклад, продуктивність, безпека тощо).

Функціональні вимоги складаються з двох частин: функції та поведінки. Функція – це те, що виконує система (наприклад, «розрахувати податок з продажу»). Поведінка залежить від того, як система це робить (наприклад, «Система розраховує податок із продажу, помноживши ціну покупки на ставку податку») [7].

На діаграмі, що на рисунку 2.1, наведено основні функціональні вимоги для даної предметної області і специфікацію вимог до неї (рисунок 2.2).

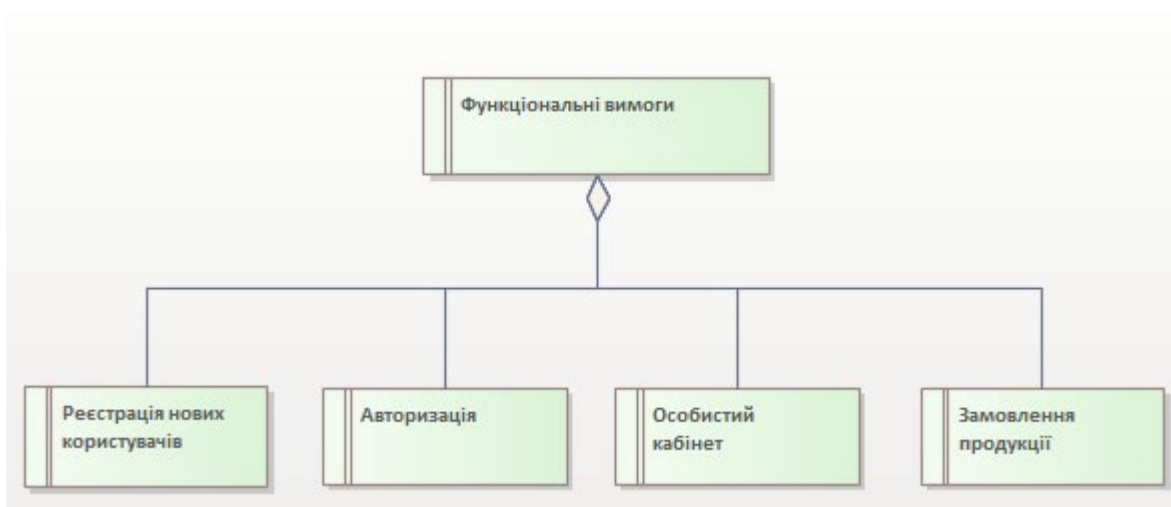


Рисунок 2.1 – Діаграма функціональних вимог

Джерело: розроблено автором самостійно

Item	Stereotype	Status	Difficulty	Priority
<input checked="" type="checkbox"/> Авторизація		Proposed	Medium	Medium
<input checked="" type="checkbox"/> Замовлення продукції		Proposed	Medium	Medium
<input checked="" type="checkbox"/> Особистий кабінет		Proposed	Medium	Medium
<input checked="" type="checkbox"/> Реєстрація нових користувачів		Proposed	Medium	Medium

Рисунок 2.2 – Специфікація вимог

Джерело: розроблено автором самостійно

Нефункціональні вимоги, також відомі як NFR, є набір вимог, які описують робочі можливості та обмеження системи з метою покращення її функціональності.

Це основні вимоги, які визначають продуктивність, якщо врахувати такі фактори, як швидкість, безпеку, надійність і цілісність даних. У структурі стандартів ISO/IEC 25000 нефункціональні вимоги визначаються як вимоги до якості системи та програмного забезпечення.

Функціональні та нефункціональні вимоги визначають конкретні характеристики продукту, необхідні для задоволення потреб зацікавлених сторін і бізнесу. Але, як видно з назви, вони зосереджені на двох різних речах:

- функціональні вимоги визначають функції та характеристики програмного продукту;
- нефункціональні вимоги визначають якість системи [8].

Діаграма, представлена на рисунку 2.3, відображає основні нефункціональні вимоги для даної предметної області, а специфікація вимог до неї наведена на рисунку 2.4.

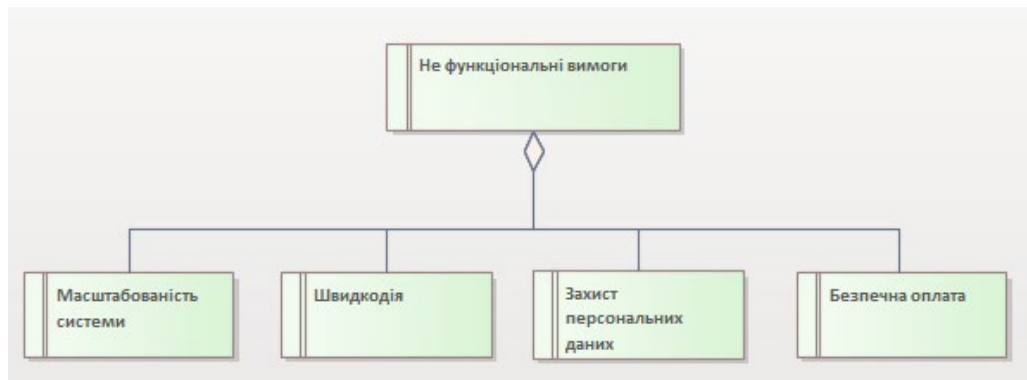


Рисунок 2.3 – Діаграма нефункціональних вимог

Джерело: розроблено автором самостійно

Item	Stereotype	Status	Difficulty	Priority
<input checked="" type="checkbox"/> Безпечна оплата		Proposed	Medium	Medium
<input checked="" type="checkbox"/> Захист персональних даних		Proposed	Medium	Medium
<input checked="" type="checkbox"/> Масштабованість системи		Proposed	Medium	Medium
<input checked="" type="checkbox"/> Не функціональні вимоги		Proposed	Medium	Medium
<input checked="" type="checkbox"/> Швидкодія		Proposed	Medium	Medium

Рисунок 2.4 – Специфікація вимог

Джерело: розроблено автором самостійно

Діаграм вимог з поданням ієрархії вимог, а також зв'язки між вимогами та іншими елементами моделі зображена на рисунку 2.5

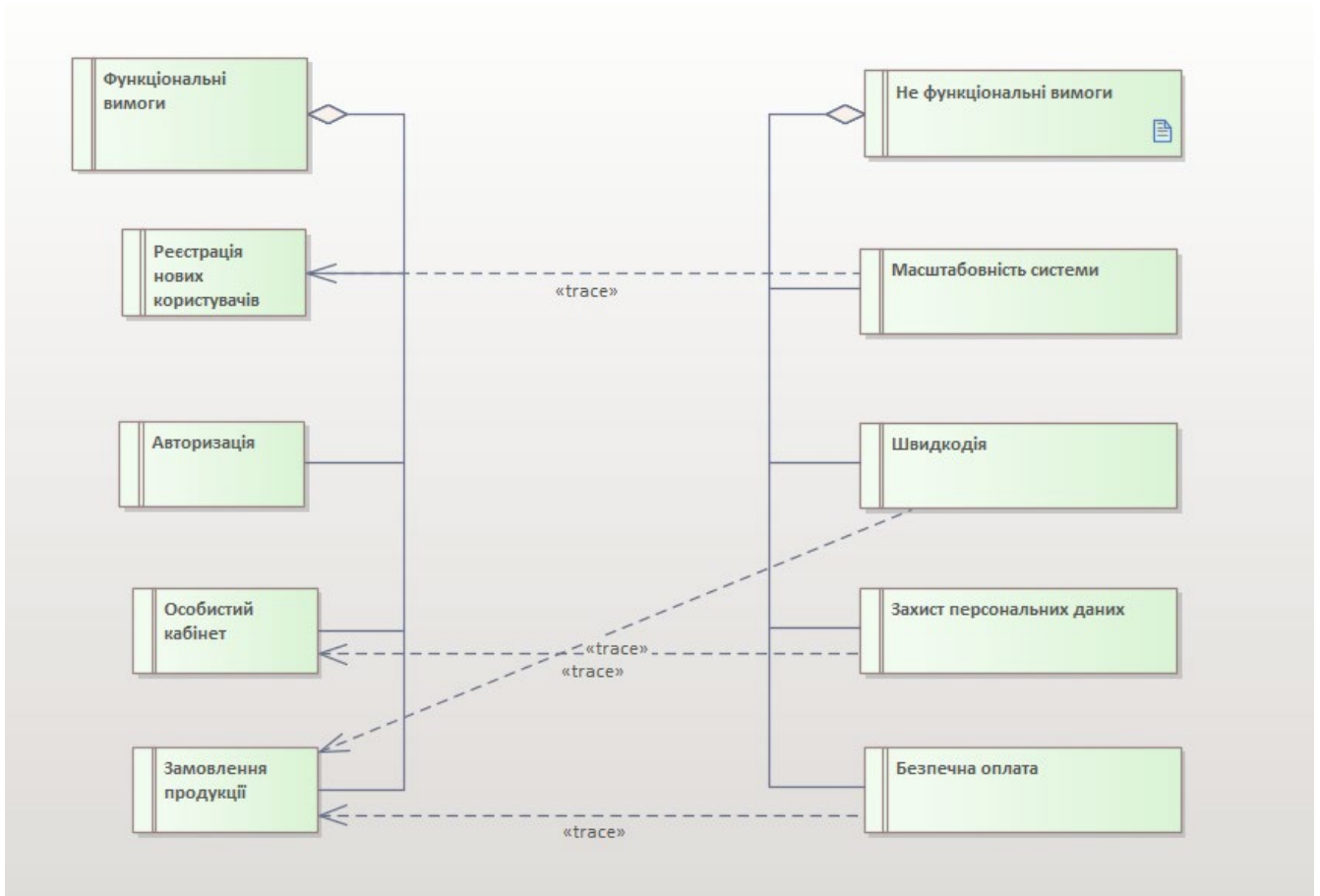


Рисунок 2.5 – Діаграма вимог з поданням ієрархії вимог, а також зв'язки між вимогами

Джерело: розроблено автором самостійно

2.2 Постановка задач та алгоритм розв'язання задач

Основне завдання смарт-сервісу здорового харчування є спрощення замовлення здорової їжі в інтернеті. При розв'язку даного завдання враховуються наступні об'єкти: клієнти, позиції меню (програми), замовлення. Взаємозв'язки між компонентами системи зображені на інформаційній моделі задачі, що зображена на рисунку 2.6.

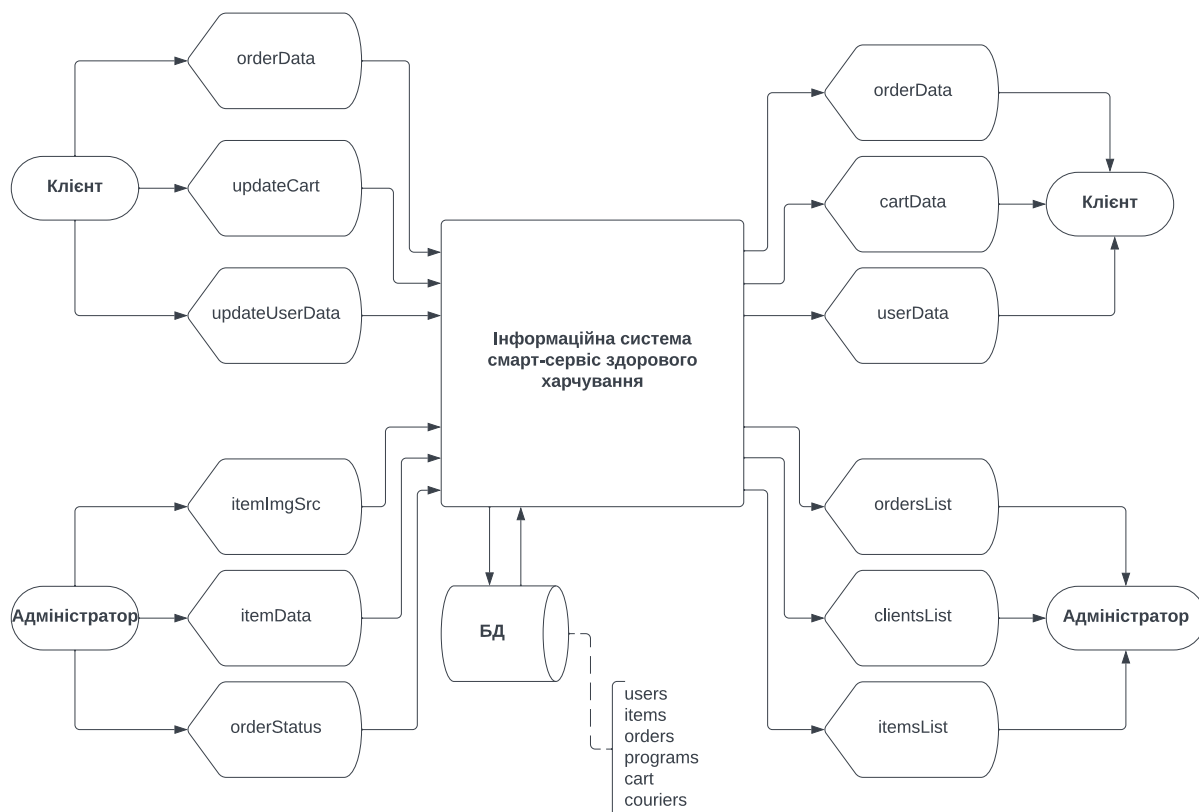


Рисунок 2.6 – Інформаційна модель задач

Джерело: розроблено автором самостійно

Таблиця 2.1 – Перелік і опис вихідних повідомлень

№ з/п	Назва вихідного повідомлення	Ідентифікатор	Форма подання і вимоги до неї	Періодичність видання	Термін видання і допустимий час затримки	Користувач і інформації
1	Меню страв	itemsList	Екранна форма	За запитом	Миттєво	Клієнт
2	Програми харчування	programsList	Екранна форма	За запитом	Миттєво	Клієнт
3	Інформація про замовлення	orderData	Екранна форма	За запитом	Миттєво	Клієнт
4	Інформація про користувача	userData	Екранна форма	За запитом	Миттєво	Клієнт
5	Кошик замовлень	cartData	Екранна форма	За запитом	Миттєво	Клієнт
6	Замовлення усіх клієнтів	ordersList	Екранна форма	За запитом	Миттєво	Адмін
7	Інформація про всіх клієнтів	usersList	Екранна форма	За запитом	Миттєво	Адмін

8	Усі позиції меню	itemsList	Екранна форма	За запитом	Миттєво	Адмін
9	Усі програми	programsList	Екранна форма	За запитом	Миттєво	Адмін
10	Панель для адміністратора	admin-panel	Екранна форма	За запитом	Миттєво	Адмін

Джерело: розроблено автором самостійно

1. Меню страв (itemsList) - система повинна надавати за запитом список страв певного типу, відфільтрований за вказаними клієнтом фільтрами, а також відсортований за ціною, якщо була вибрана відповідна опція.

2. Програми харчування (programsList) - система повинна надавати за запитом список програм харчування певного типу.

3. Інформація про замовлення (orderData) - система повинна надавати інформацію про конкретне замовлення, його деталі, статус, та товари, що були замовлені, їх кількість, суму, та загальну суму замовлення.

4. Інформація про користувача (userData) - система повинна надавати інформацію про користувача, його дані, контактну інформацію, історію замовлень та інші важливі дані.

5. Кошик замовлень (cartData) - система повинна надавати інформацію про товари в кошику замовлень, їх кількість, суму разом та загальну суму товарів у кошику.

6. Замовлення усіх клієнтів (ordersList) - система повинна надавати за запитом список замовлень усіх клієнтів, вивести список замовлень відсортований від нових до старих, та їх статус.

7. Інформація про всіх клієнтів (usersList) - система повинна надавати інформацію про всіх клієнтів, їх дані, контактну інформацію, історію замовлень та інші важливі дані.

8. Усі позиції меню (itemsList) - система повинна надавати за запитом список усіх позицій меню з можливістю фільтрації та сортування за різними параметрами.

9. Усі програми (programsList) - система повинна надавати за запитом

список усіх програм з можливістю фільтрації та сортування за різними параметрами.

Панель для адміністратора (admin-panel) - система повинна надавати адміністратору доступ до адміністративної панелі з можливістю управління замовленнями, користувачами та іншими параметрами системи.

Таблиця 2.1 – Перелік і опис вхідних повідомлень

№ з/п	Назва вхідного повідомлення	Ідентифікатор	Форма подання	Термін і частота надходження	Джерело
1	Форма реєстрації	userData	Екранна форма	При реєстрації	Клієнт
2	Форма авторизації	userData	Екранна форма	При авторизації	Клієнт
3	Пошук страв в меню	searchByName	Екранна форма	При пошуку страви в меню	Клієнт
4	Додавання в кошик	addToCart	Екранна форма	При додаванні позиції у кошик	Клієнт
5	Видалення з кошика	deleteFromCart	Екранна форма	При видаленні з кошика	Клієнт
6	Замовлення	addOrder	Екранна форма	При замовленні	Клієнт
7	Інформація про користувача	updateUserData	Екранна форма	При зміні в особистому кабінеті	Клієнт
8	Інформація про замовлення	orderData	Екранна форма	При замовленні	Клієнт
9	Форма додавання нової позиції в меню	addToItemsList	Екранна форма	При додаванні нової позиції	Адмін
10	Форма додавання нової програми	addToProgramsList	Екранна форма	При додаванні нової програми	Адмін
11	Форма видалення позиції в меню	deleteFromItemsList	Екранна форма	При видаленні позиції з меню	Адмін
12	Форма видалення програми	deleteFromProgramsList	Екранна форма	При видаленні програми	Адмін
13	Форма оновлення позиції в меню	updateItem	Екранна форма	При оновленні позиції	Адмін
14	Форма оновлення програми	updateProgram	Екранна форма	При оновленні програми	Адмін

Джерело: розроблено автором самостійно

1. Форма реєстрації (userData) - система повинна надавати екранну форму для реєстрації користувачів при реєстрації, і це повинно відбуватися за ініціативою клієнта.

2. Форма авторизації (userData) - система має надавати екранну форму для авторизації користувачів при авторизації, і це також відбувається за ініціативою клієнта.

3. Пошук страв в меню (searchByName) - система має екранну форму для пошуку страв у меню за назвою, і це відбувається за ініціативою клієнта.

4. Додавання в кошик (addToCart) - система має екранну форму для додавання позицій у кошик під час покупки, і це відбувається за ініціативою клієнта.

5. Видалення з кошика (deleteFromCart) - система має екранну форму для видалення позицій з кошика під час покупки, і це відбувається за ініціативою клієнта.

6. Замовлення (addOrder) - система має екранну форму для замовлення продуктів або послуг, і це відбувається за ініціативою клієнта.

7. Інформація про користувача (updateUserData) - система має екранну форму для зміни даних користувача в особистому кабінеті, і це відбувається за ініціативою клієнта.

8. Інформація про замовлення (orderData) - система має екранну форму для надання інформації про замовлення при замовленні продуктів або послуг, і це також відбувається за ініціативою клієнта.

9. Форма додавання нової позиції в меню (addToItemsList) - система має екранну форму для додавання нових позицій у меню, і це відбувається за ініціативою адміністратора.

10. Форма додавання нової програми (addToProgramsList) - система має екранну форму для додавання нових програм, і це відбувається за ініціативою адміністратора.

11. Форма видалення позиції в меню (deleteFromItemsList) - система має екранну форму для видалення позицій з меню, і це відбувається за ініціативою адміністратора.

адміністратора.

12. Форма видалення програми (deleteFromProgramsList) - система має екранну форму для видалення програм, і це відбувається за ініціативою адміністратора.

13. Форма оновлення позиції в меню (updateItem) - система має екранну форму для оновлення позицій в меню, і це відбувається за ініціативою адміністратора.

14. Форма оновлення програми (updateProgram) - система має екранну форму для оновлення програм, і це відбувається за ініціативою адміністратора.

Для розрахунку потрібної добової норми калорій можна використати наступні формули [9]:

для чоловіків

$$DCI = (w_{кг} \times 10 + h_{см} \times 6.25 - a_{років} \times 5 + 5) \times k$$

для жінок

$$DCI = (w_{кг} \times 10 + h_{см} \times 6.25 - a_{років} \times 5 - 161) \times k$$

w – вага в кілограмах;

h – ріст в сантиметрах;

a – вік в роках;

k -коефіцієнт активності.

В даному проекті для проектування інформаційних систем було обрано процесний підхід, який акцентує увагу на моделюванні та управлінні бізнес-процесами під час розробки. Цей підхід став популярним в галузі ІТ та бізнес-аналітики, оскільки він дозволяє забезпечити більшу взаємодію між інформаційними системами та реальними бізнес-процесами підприємства [17].

Основні принципи та аспекти процесного підходу включають:

1. Моделювання бізнес-процесів: Акцент робиться на аналізі, моделюванні та оптимізації бізнес-процесів, щоб зрозуміти, як підприємство функціонує та які процеси потребують автоматизації.

2. Взаємодія зі стейкхолдерами: Активна взаємодія зі стейкхолдерами, такими як клієнти та керівництво, допомагає визначити їхні потреби та очікування.
3. Стратегічне планування: Врахування стратегічних цілей підприємства та забезпечення відповідності інформаційних систем цим цілям.
4. Автоматизація бізнес-процесів: Ідентифікація процесів для автоматизації та розробка систем, які цю автоматизацію підтримують.
5. Моніторинг та оптимізація: Постійний моніторинг та внесення змін для покращення ефективності бізнес-процесів після впровадження інформаційних систем [17].

В даному проєкті також використовуються інструменти для моделювання бізнес-процесів, такі як Bizagi Process Modeler [18] та Figma [19]. Bizagi допомагає нам в аналізі, проектуванні та оптимізації процесів, а Figma - в створенні дизайну та прототипів інтерфейсів користувача. Використання цих інструментів допомагає нам ефективно впроваджувати процесний підхід та досягати більшої успішності у розробці інформаційних систем.

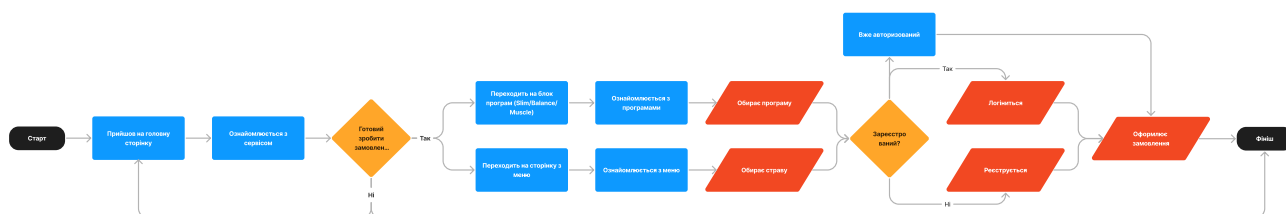


Рисунок 2.7 – Процес оформлення замовлення

Джерело: розроблено автором самостійно

На рисунку 2.7 зображено процес оформлення замовлення. Коли клієнт вже ознайомився із сервісом, він може перейти до сторінки Меню або Програми, де він може обрати запропоновані страви в меню або 1 із 3 програм. Після чого, якщо він вже авторизований може оформити замовлення або якщо не зареєстрований, спочатку застосунок направить його на сторінку авторизації.

На діаграмі (рисунок 2.8) представлені три ключові учасники процесу: клієнт, менеджер та кур'єр. Кожен з цих учасників має свою відповідну роль та взаємодіє

у рамках чотирьох основних підпроцесів, які ілюструють діаграму. Перший підпроцес – вибір позицій для замовлення – описує етап, на якому клієнт обирає необхідні товари чи послуги для покупки.

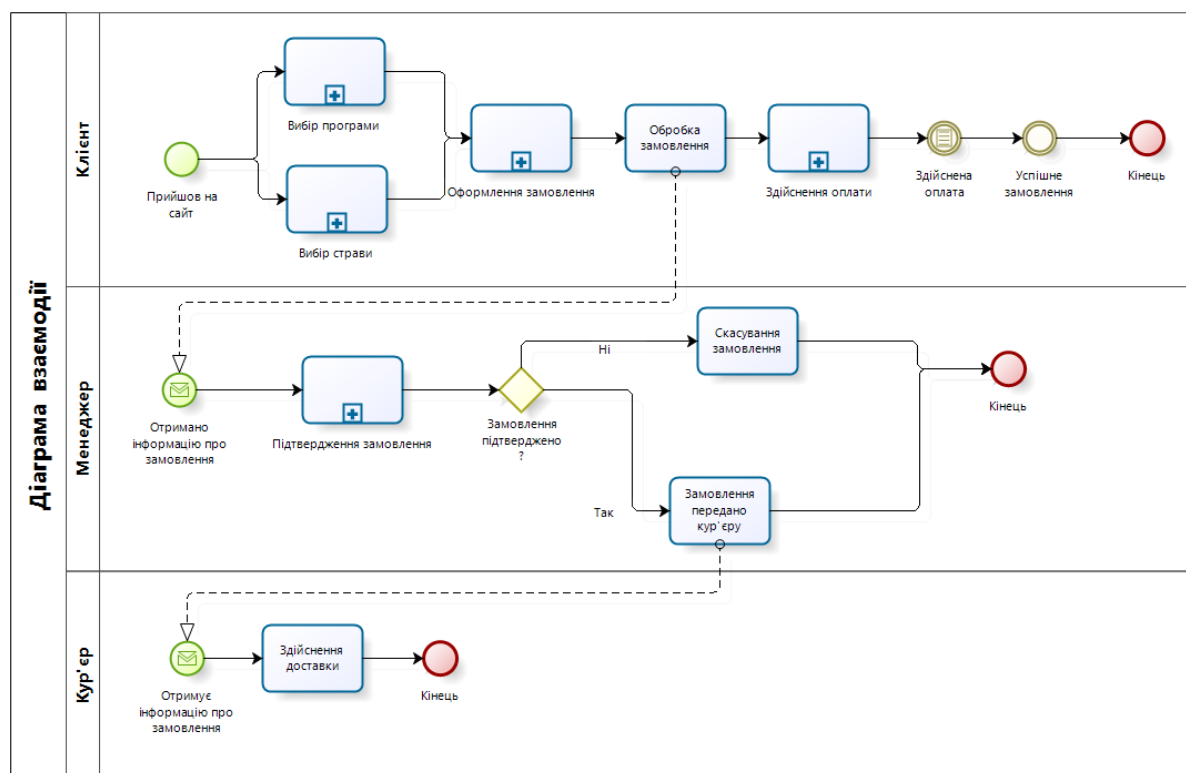


Рисунок 2.8 – Діаграма взаємодії

Джерело: розроблено автором самостійно

Другий підпроцес – оформлення замовлення – включає у себе усі дії, пов'язані з реєстрацією та оформленням замовлення клієнтом. Третій підпроцес – здійснення оплати – описує процес оплати за замовлення, який може включати в себе різні способи оплати, наприклад, оплата онлайн чи готівкою при отриманні товару. І, нарешті, четвертий підпроцес – підтвердження менеджером – описує етап, на якому менеджер системи підтверджує та обробляє замовлення, готовий до подальшої обробки кур'єром для доставки.

На рисунку 2.9 зображений алгоритм авторизації/реєстрації за допомогою облікового запису Google, який допомагає вирішувати завдання, що виникає під час роботи веб-застосунку.

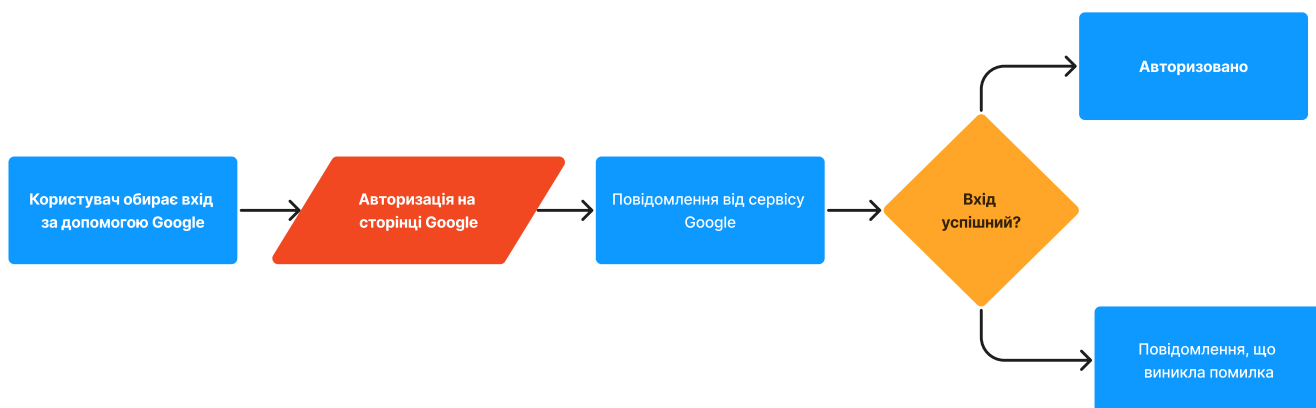


Рисунок 2.9 – Алгоритм авторизації/входу через Google

Джерело: розроблено автором самостійно

На рисунку 2.10 зображений алгоритм авторизації за допомогою електронної пошти і паролю, який допомагає вирішувати завдання, що виникає під час роботи веб-застосунку.

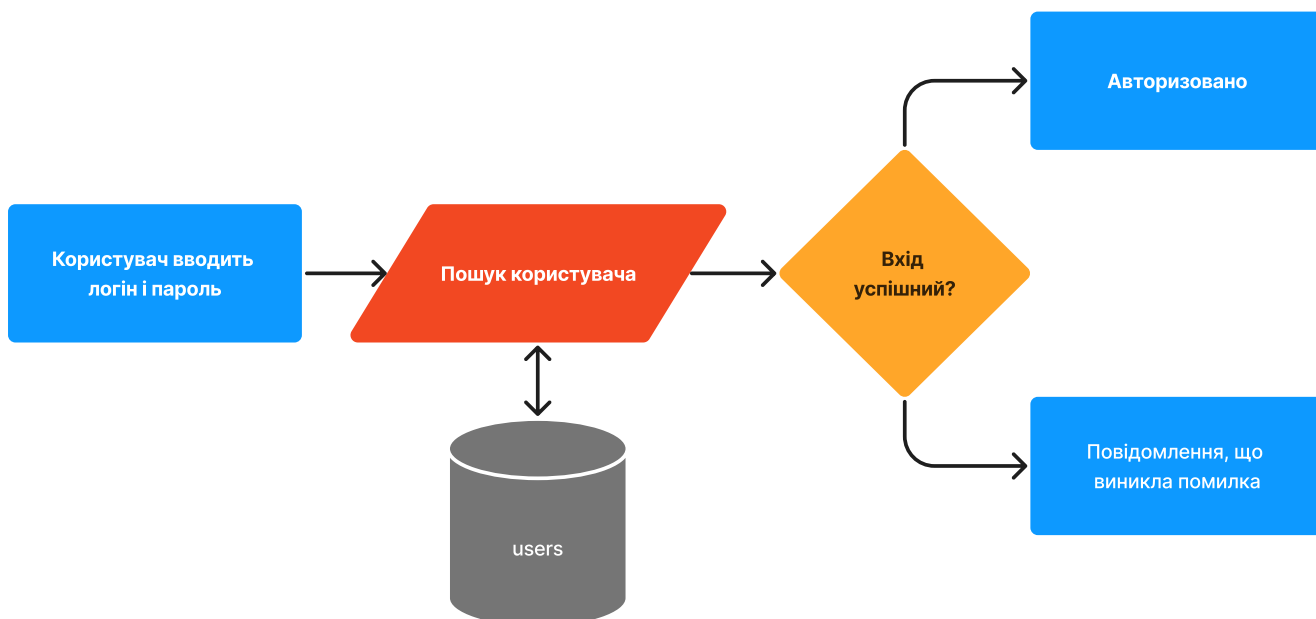


Рисунок 2.10 – Алгоритм авторизації/входу через Google

Джерело: розроблено автором самостійно

На рисунку 2.11 зображений деталізований алгоритм реєстрації облікового запису за допомогою електронної пошти і паролю.

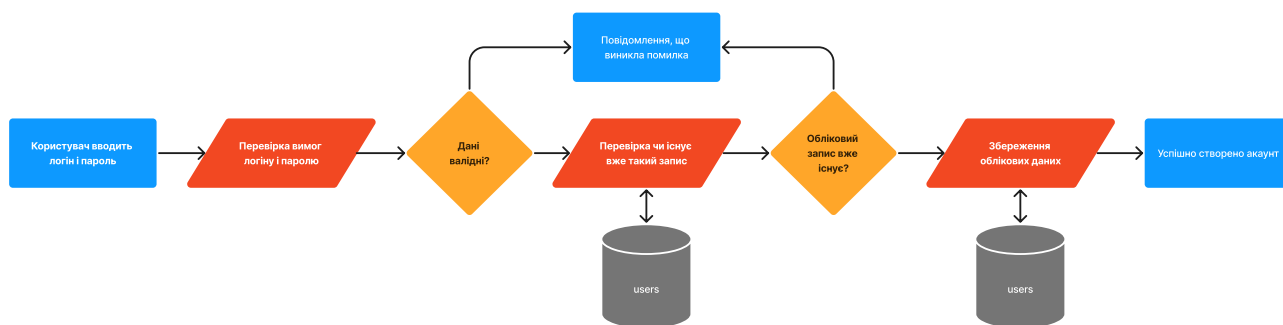


Рисунок 2.11 – Деталізований алгоритм авторизації/входу через Google
Джерело: розроблено автором самостійно

2.3 Моделювання інформаційної системи

Діаграма визначення блоку або «bdd» (block definition diagram) використовується для визначення блоків з точки зору їх функцій і їх структурних зв'язків з іншими блоками. Заголовок діаграми визначення блоку зображено таким чином: bdd [тип елемента моделі] ім'я елемента моделі [назва діаграми]

Діаграма визначення блоку може представляти пакет, блок або блок обмежень, як зазначено типом елемента моделі в квадратних дужках. Ім'я елемента моделі — це ім'я пакета, блоку або блоку обмежень, а ім'я діаграми визначається користувачем і часто використовується для опису призначення діаграми [10].

Діаграма визначення блоків для обраної предметної області зображена на рисунку 2.12.

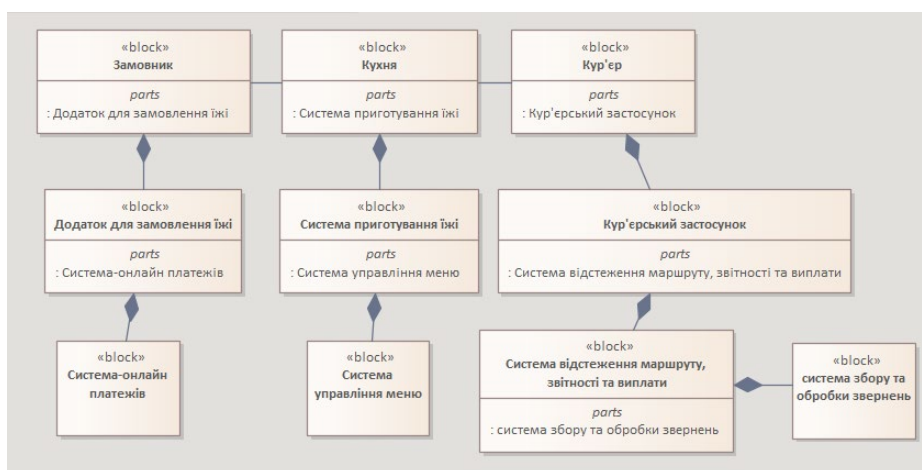


Рисунок 2.12 – Діаграма визначення блоків

Джерело: розроблено автором самостійно

Блок (нотація: прямокутник із ключовим словом = «блок») є системним компонентом, який представляє структурну модульну одиницю, яка в свою чергу інкапсулює свій вміст (властивості, поведінку, обмеження) і підтримує першокласність (тобто вона може бути нарисована та прямо керується в репозиторії моделі) поведінки, що інкапсулюється блоками, включно з: операціями, сигналами та кінцевими автоматами. Виняткові точки взаємодії для приєднання та з'єднання блочних інтерфейсів називають портами.

- Блоки можуть визначати програмні, апаратні, механічні компоненти та компоненти wetware (особи, організації, об'єкти).
- Блоки підтримують як надані (впроваджені або реалізовані), так і необхідні (використані) інтерфейси як для інформаційних, так і для фізичних потоків.
- Блоки можна рекурсивно розкласти на частини, де кожна частина також має бути визначена блоком [11].

Побудована діаграма внутрішніх блоків зображена на рисунку 2.13.

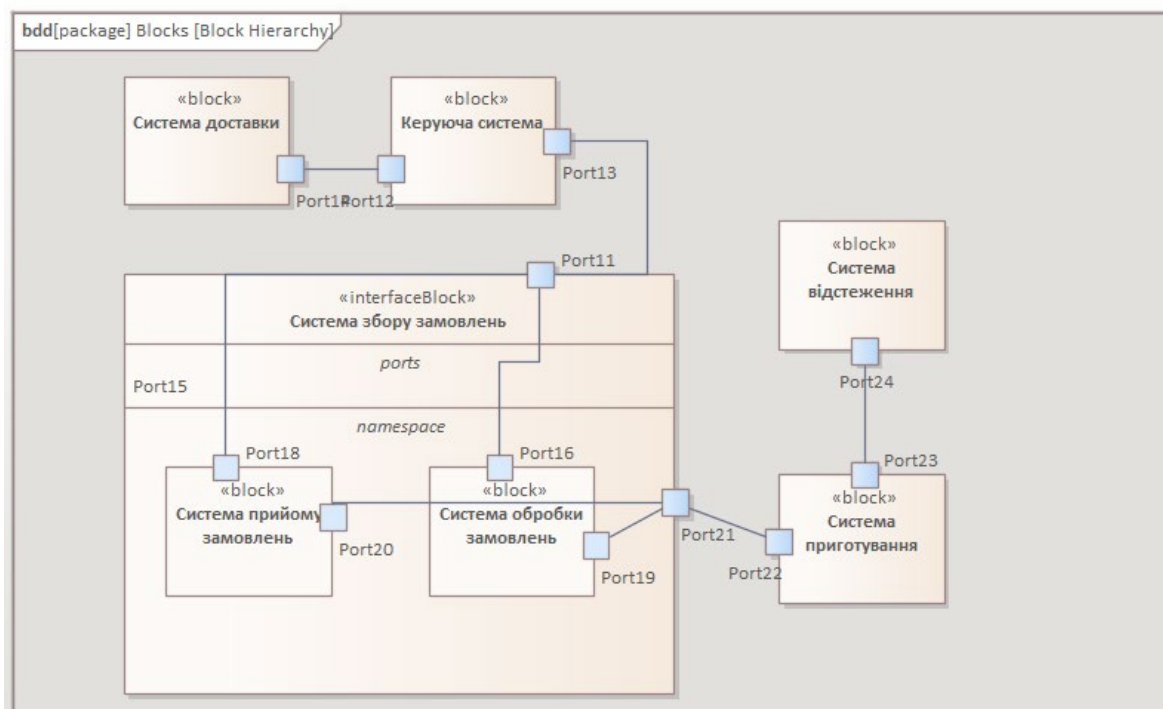


Рисунок 2.13 – Діаграма внутрішніх блоків

Джерело: розроблено автором самостійно

Діаграма класів, створена на уніфікованій мові моделювання (UML), є типом діаграми статичної структури, яка описує структуру системи, демонструючи класи системи, їхні характеристики, операції (або методи) і зв'язки між об'єктами [12]. Це використовується для розробки програмного забезпечення. Було побудовано діаграму класів, яка описує класи і атрибути для даної системи, що зображена на рисунку 2.14.

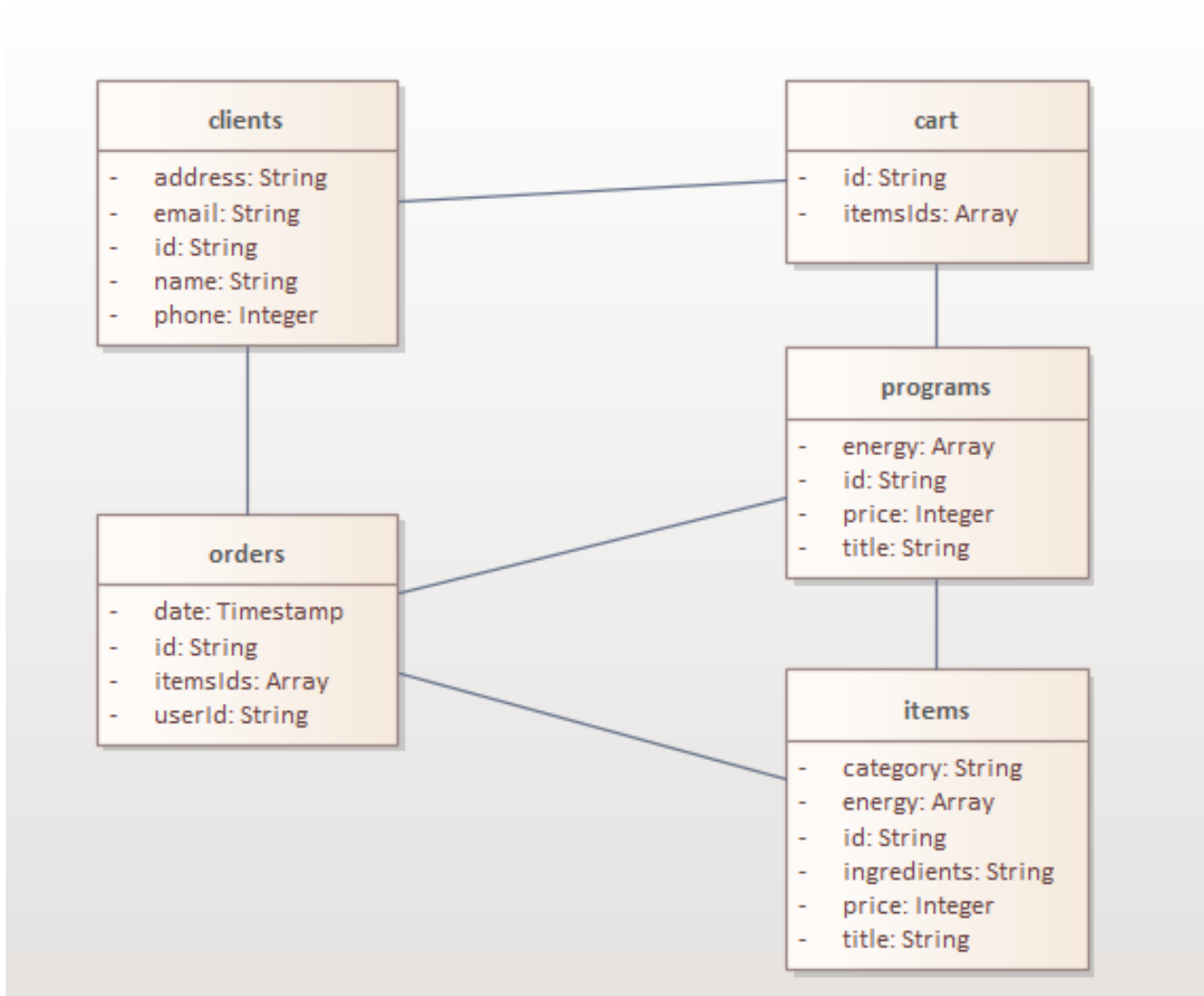


Рисунок 2.14 – Діаграма класів

Джерело: розроблено автором самостійно

Для вибраної предметної області була створена діаграма трасування, на якій присутні вимоги, прецеденти, варіанти тестування та елементи користувацького інтерфейсу та відповідні зв'язки між ними (рисунок 2.15), а також матрицю

взаємозв'язків які доповнюють дану діаграму (рисунок 2.16).

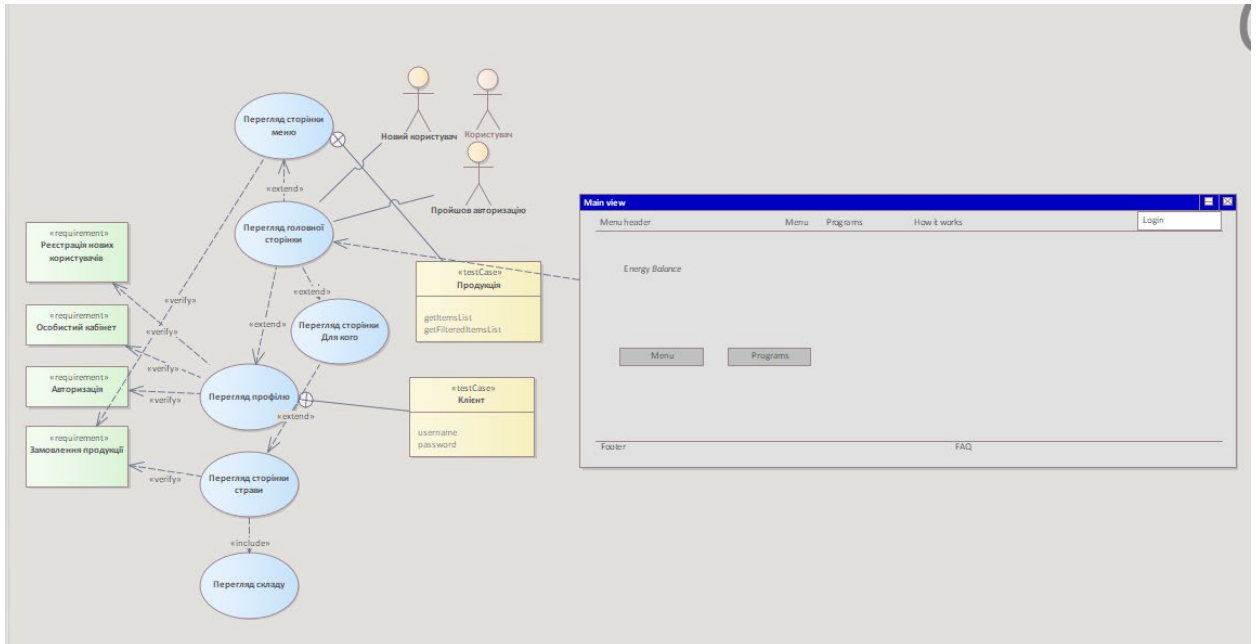


Рисунок 2.15– Діаграма трасування

Джерело: розроблено автором самостійно

Source	Requirements Traceability::Requirement1	Requirements Traceability::Авторизація	Requirements Traceability::Авторизація	Requirements Traceability::Замовлення продукції	Requirements Traceability::Замовлення продукції	Requirements Traceability::Особистий кабінет	Requirements Traceability::Особистий кабінет	Requirements Traceability::Реєстрація нових користув	Requirements Traceability::Реєстрація нових користув
Requirements Traceability::Перегляд головної сторінки		↑				↑		↑	
Requirements Traceability::Перегляд профілю						↑			
Requirements Traceability::Перегляд складу				↑					
Requirements Traceability::Перегляд сторінки Для кого									
Requirements Traceability::Перегляд сторінки меню				↑					
Requirements Traceability::Перегляд сторінки страви				↑					

Рисунок 2.16 – Матриця взаємозв'язків

Джерело: розроблено автором самостійно

Для моделювання поведінки системи, можна візуально зобразити її на діаграмах прецедентів і діаграмах послідовності.

Діаграма Прецедентів візуально зображає різноманітні сценарії взаємодії між акторами (користувачами) і прецедентами (випадками використання), описує функціональні аспекти системи (бізнес логіку) [21].

Побудовану діаграма прецедентів для обраної предметної області зображено на рисунку 2.17.

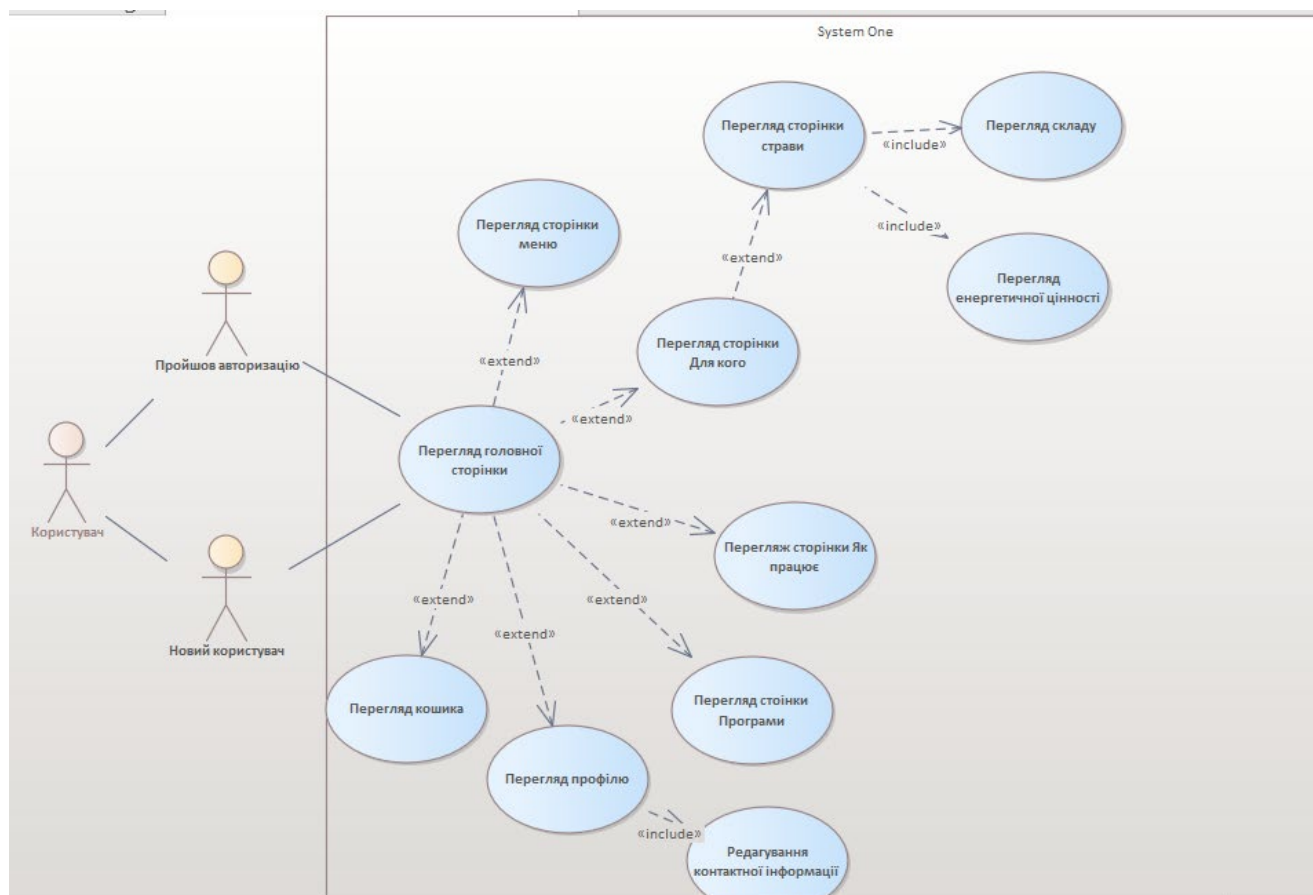


Рисунок 2.17 – Діаграма прецедентів

Джерело: розроблено автором самостійно

Діаграма послідовності (Sequence Diagram) - використовуються для моделювання логіки сценаріїв використання, показуючи інформацію, що передається між об'єктами в системі під час виконання сценарію.

В них описано, як процеси або об'єкти взаємодіють під час виконання сценарію. На діаграмі відображаються класи, необхідні для виконання сценарію, і

повідомлення, які вони передають один одному (викликані кроками у варіанті використання) [22].

Було побудовано діаграма послідовності, які відображають процес авторизації в даній системі (рисунок 2.18).

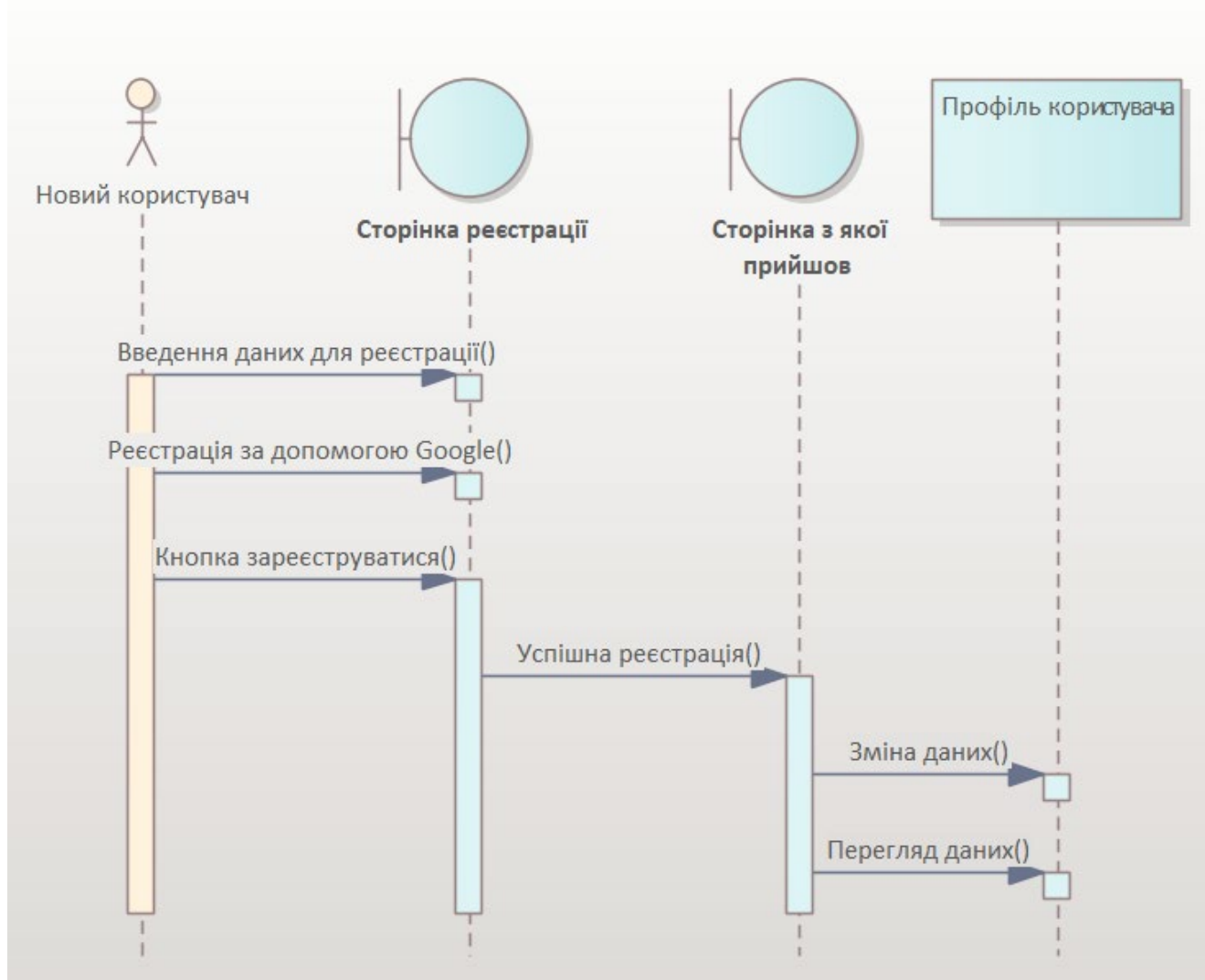


Рисунок 2.18 – Діаграма послідовності

Джерело: розроблено автором самостійно

Також, додатково ще було побудовано діаграму послідовності для процесу реєстрації за допомогою електронної пошти і паролю або за допомогою облікового запису Google (входу) в систему (рисунок 2.19).

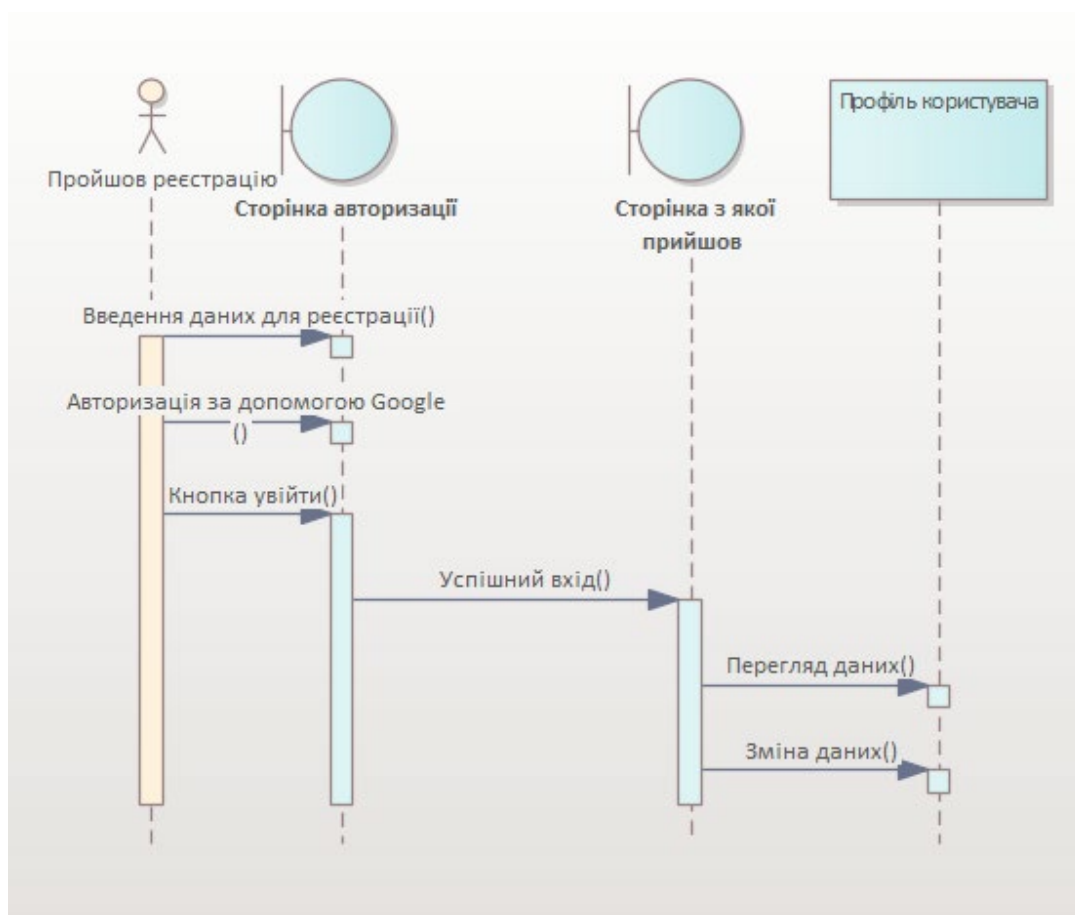


Рисунок 2.19 Діаграма послідовності

Джерело: розроблено автором самостійно

РОЗДІЛ 3 ПРОЕКТУВАННЯ ТА РЕАЛІЗАЦІЯ КОМПОНЕНТІВ СИСТЕМИ

3.1 Інформаційне забезпечення

Для реалізації зберігання даних в даному смарт-сервісі буде використовуватись документно-орієнтована база даних Google Cloud Firestore.

Cloud Firestore — це гнучка масштабована база даних, яка працює з Firebase і Google Cloud для мобільних, веб-розробок і серверних розробок. Вона підтримує синхронізацію даних між клієнтськими програмами, як і Firebase Realtime Database. Крім того, Firestore дозволяє легко інтегруватися з іншими продуктами Firebase і Google Cloud, такими як Cloud Functions [13].

Модель логічного рівня, відома як «даталогічна модель», показує логічні зв'язки між елементами даних, незалежно від їхнього змісту та середовища зберігання. Ця модель будується відповідно до інформаційних одиниць, які використовуються в СУБД, у якій ми проектуємо базу даних. Даталогічне проектування — це етап створення даталогічної моделі. Схема описує логічну структуру бази даних СУБД [14].

На рисунку 3.1 зображено даталогічну модель БД для смарт-сервісу здорового харчування.

Ця модель бази даних відображає систему управління послугами доставки їжі. Ось опис кожної таблиці:

1. `admins` (адміністратори): Зберігає інформацію про адміністраторів системи. Кожен адміністратор має ідентифікатор, ім'я та посаду в організації.
2. `items` (продукти): Представляє товари, доступні для замовлення. Кожен товар має ідентифікатор, назву, склад, енергетичну цінність (можливо, масив для різних харчових значень), категорію та ціну.
3. `couriers` (кур'єри): Містить дані про кур'єрів доставки. Кожен кур'єр має ідентифікатор та ім'я.
4. `cart` (кошик): Відстежує товари, додані до кошика користувачем. Він

має ідентифікатор та масив ідентифікаторів товарів, які є у кошику. Посилається на таблиці `items` та `clients` за допомогою зовнішніх ключів.

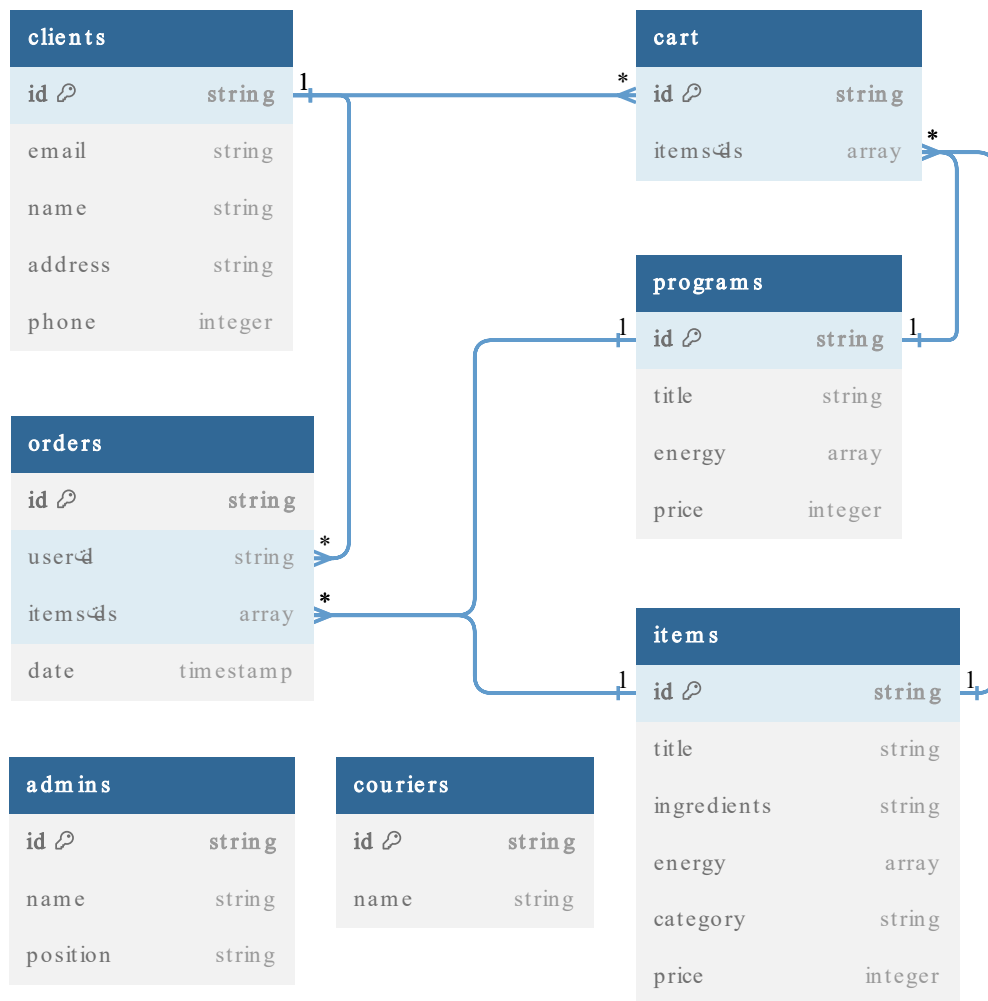


Рисунок 3.1 – Даталогічна модель БД

Джерело: розроблено автором самостійно

5. `programs` (програми): Представляє програми або пакети страв, які користувачі можуть замовляти. Кожна програма має ідентифікатор, назву, енергетичну цінність (масив) та ціну.

6. `clients` (клієнти): Зберігає інформацію про клієнтів або користувачів послуг доставки їжі. Кожен клієнт має ідентифікатор, електронну адресу, ім'я, адресу та номер телефону.

7. `orders` (замовлення): Записує деталі про замовлення, зроблені клієнтами. Включає ідентифікатор замовлення, ідентифікатор користувача

(посилається на clients), масив ідентифікаторів товарів у замовленні (посилається на items), та дату замовлення.

Схема бази даних включає посилання (зовнішні ключі), щоб встановлювати зв'язки між таблицями:

- items.id посилається на cart.itemsIds, programs.id та orders.itemsIds.
- clients.id посилається на cart.id та orders.userId.
- programs.id посилається на orders.itemsIds.

Ці зв'язки забезпечують цілісність даних та дозволяють виконувати запити, які враховують зв'язані дані з різних таблиць.

Опираючись на раніше створену діаграму БД, було реалізовано 4 колекції в БД Google Firestore.

На рисунку 3.2 зображено вигляд колекції cart, де зберігаються ідентифікатори страв які додав користувач в кошик і кількість даної страви.

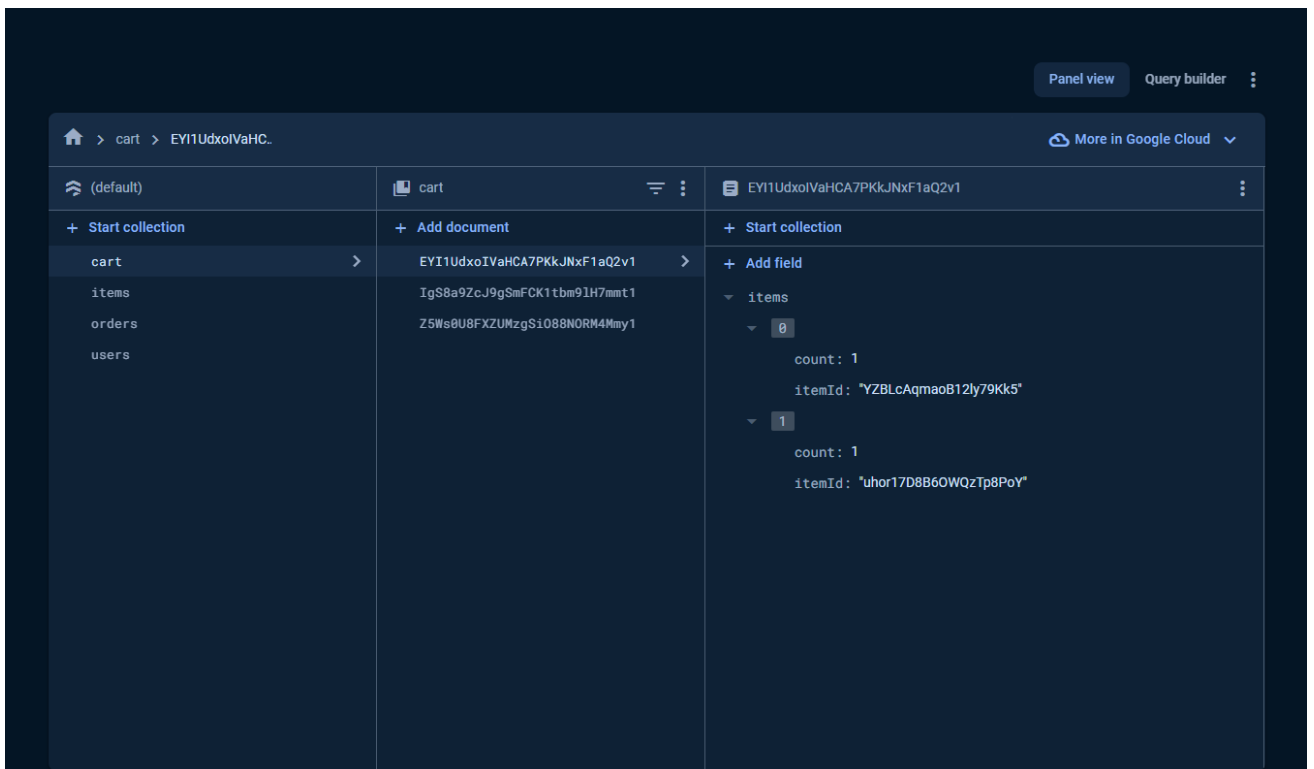


Рисунок 3.2 – Колекція cart

Джерело: розроблено автором самостійно

Колекція, що зображена на рисунку 3.3 містить документи з відомостями про страви які відображаються на сторінці меню, містить інформацію про назву, категорію в якій розміщується, енергетичну цінність КБЖВ, інгредієнти, ціну, а також зберігає картинку у форматі base64.

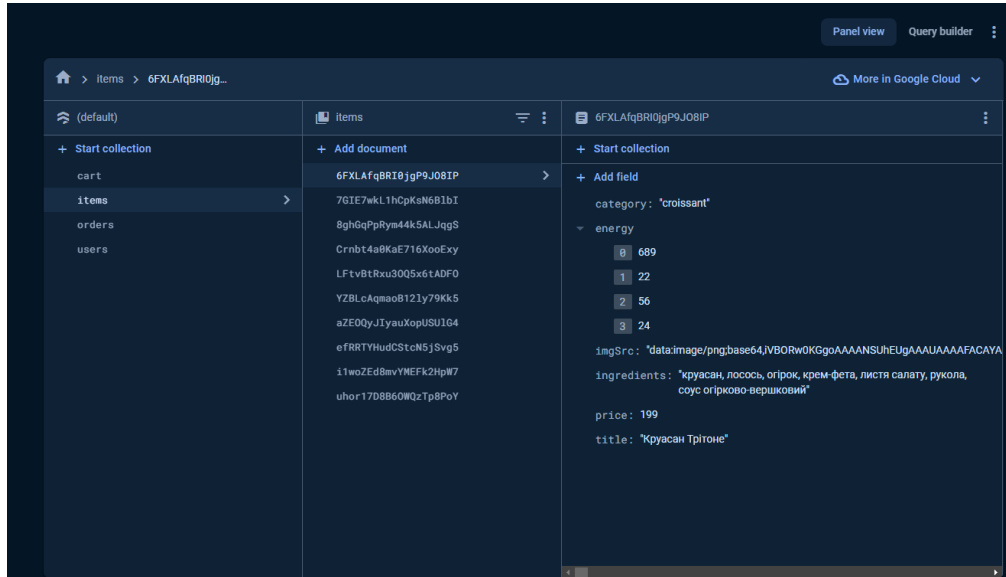


Рисунок 3.3 – Колекція items

Джерело: розроблено автором самостійно

На рисунку 3.4 зображено вигляд колекція яка зберігає всі замовлення клієнта, замовлення зберігається у вигляді об'єкту в списку і має дату замовлення, ідентифікатори страв і їхню кількість.

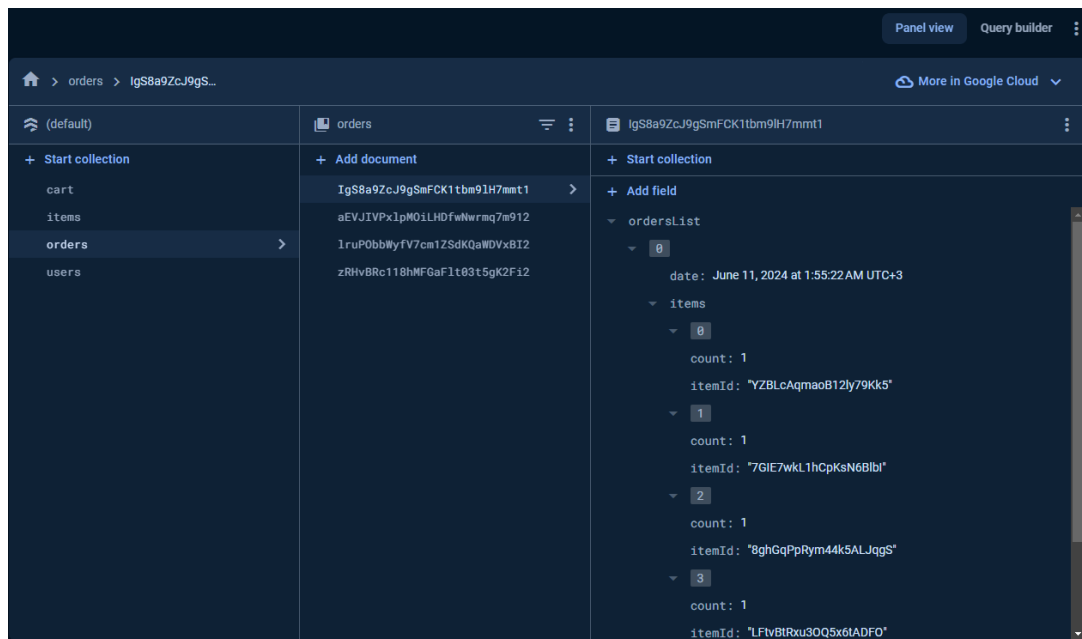


Рисунок 3.4 – Колекція orders

Джерело: розроблено автором самостійно

Колекція users, що зображена на рисунку 3.5 зберігає контактну інформацію, що клієнт надає в особистому кабінеті. В даній колекції зберігаються такі дані: ім'я, прізвище, контактний номер телефону, електронна пошта, місто, дата народження, вулиця, номер будинку, номер квартири, поштовий індекс і коментар або побажання.

На рисунку 3.6 зображено підключені провайдери для авторизації і реєстрації користувачів, за допомогою електронної пошти і паролю, а також авторизація через обліковий запис Google.

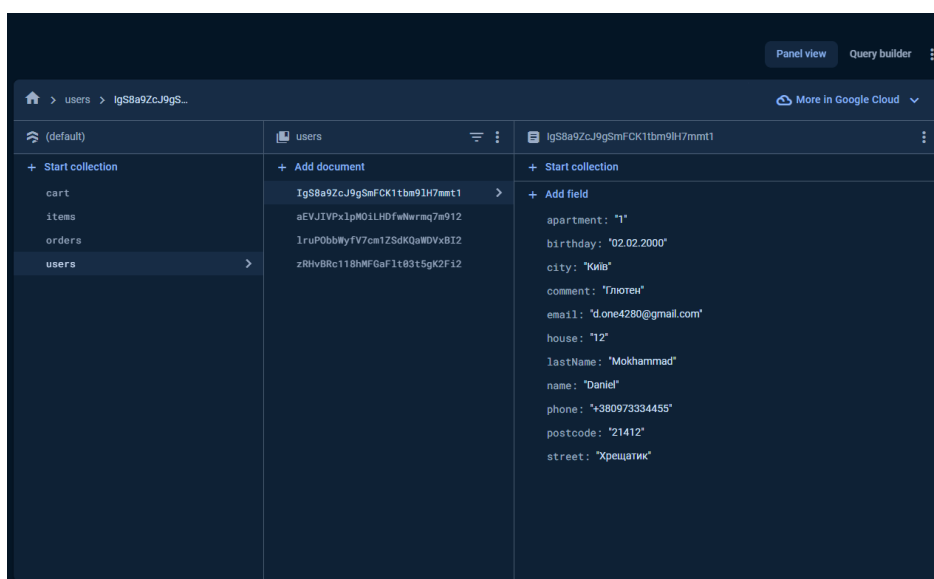


Рисунок 3.5 – Колекція users

Джерело: розроблено автором самостійно

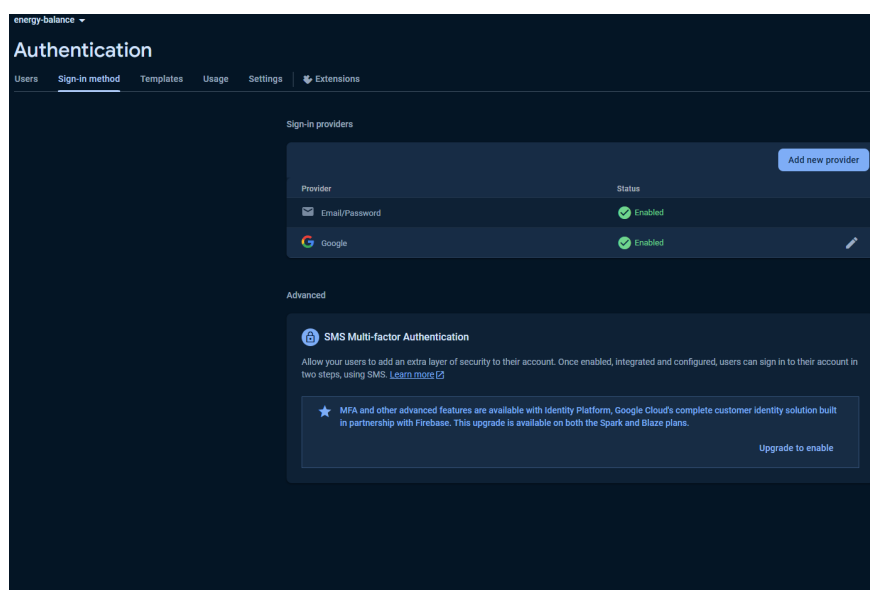


Рисунок 3.6 – Колекція users

Джерело: розроблено автором самостійно

3.2 Технічне забезпечення

Діаграми розгортання можна використовувати для опису фізичного обладнання, на якому працюватиме програмна система. Крім того, він описує, як розгортати програмне забезпечення на базовому обладнанні. Він встановлює компоненти програмного забезпечення системи на пристрій, який повинен їх виконувати.

Програмне забезпечення архітектури, створене в процесі розробки фізичної системи архітектури, представлено в діаграмі розгортання. Він імітує розподіл програмного забезпечення між фізичними вузлами в розподілених системах [15].

Діаграма розгортання (рис. 3.7) показує технічне забезпечення системи. Ця діаграма показує два пристрої: пристрій клієнта з веб-браузером, який керує взаємодією між сайтом Google Firestore та сервером.

Firestore розташована в хмарі та має доступ до неї через власний SDK для програм Apple, Android і веб-додатків.

Дані зберігаються в документах, які містять поля, які мають значення, згідно з моделями даних NoSQL Cloud Firestore. Усі ці документи зберігаються в колекціях, які служать контейнерами для документів, які можна використовувати для організації даних і побудови запитів [13].

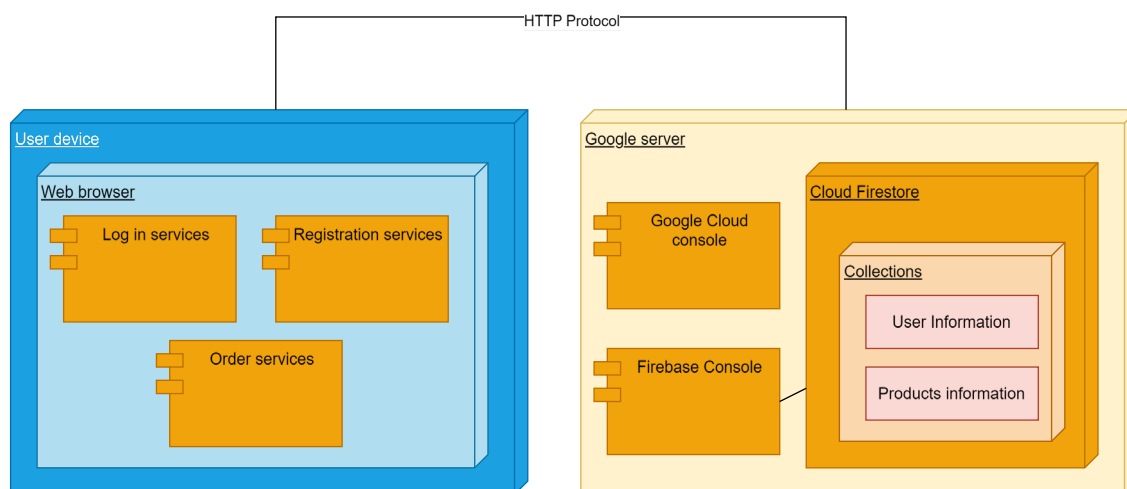


Рисунок 3.7 – Діаграма розгортання

Джерело: розроблено автором самостійно

3.3 Програмне забезпечення

Компонентна діаграма UML є важливим елементом в арсеналі діаграм UML, які призначені для моделювання систем. Ця діаграма створена з метою надання користувачам зрозумілого огляду структури певної системи. Вона дозволяє візуалізувати компоненти, їх взаємодію та залежності між ними, що допомагає в розумінні та аналізі роботи системи. Завдяки компонентній діаграмі, можна легше виявляти зв'язки між різними частинами системи, а також розуміти, як вони співпрацюють для досягнення поставлених цілей [16].

На діаграмі, що зображена на рисунку 3.8 можна побачити 5 інтерфейсів взаємодії: клієнтська частина, корзина, інтерфейси для авторизації і реєстрації, а також сторінку адміністратора.

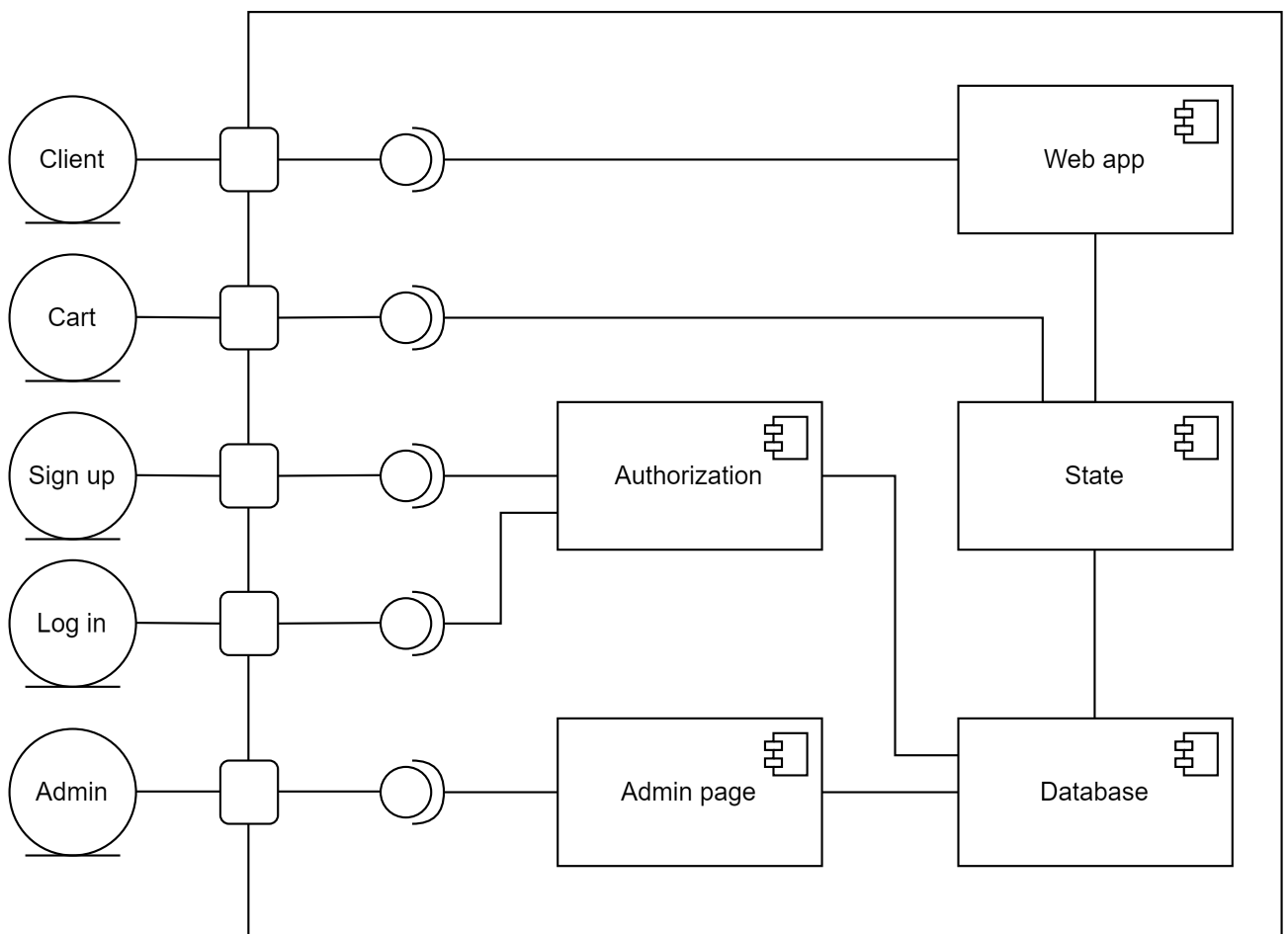


Рисунок 3.8 – Діаграма компонентів

Джерело: розроблено автором самостійно

3.4 Організаційне забезпечення

Рисунок 3.9 ілюструє всі можливі стани об'єкта замовлення та послідовність їх зміни за допомогою діаграми станів.

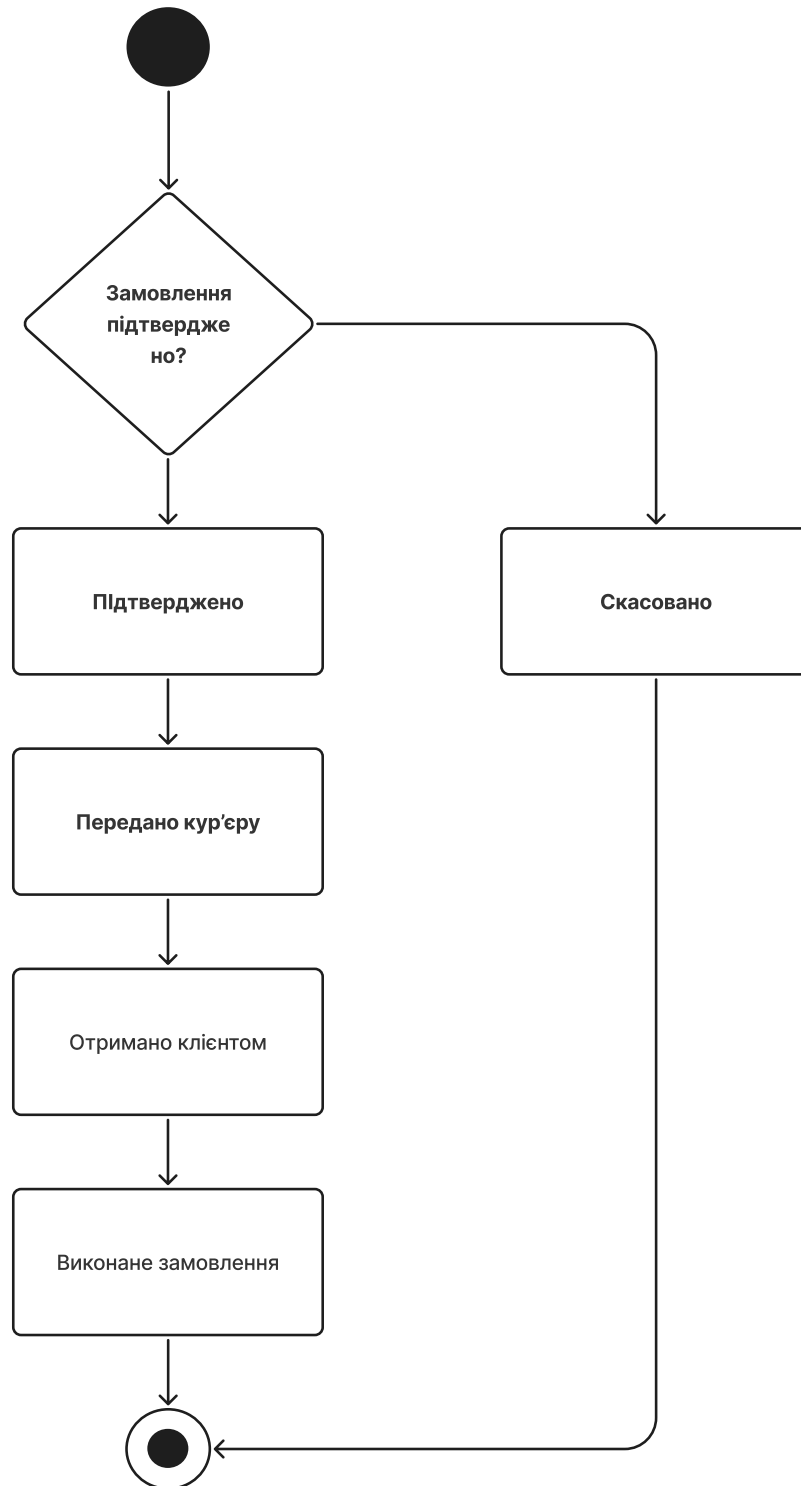


Рисунок 3.9 – Діаграма станів

Джерело: розроблено автором самостійно

На рисунку 3.10 подано діаграму дій двох ключових учасників процесу: клієнта і адміністратора/менеджера. Ця діаграма відображає всі можливі дії, які можуть виконувати ці учасники під час взаємодії [20], від початкового звернення клієнта до оформлення замовлення та його виконання працівником магазину.

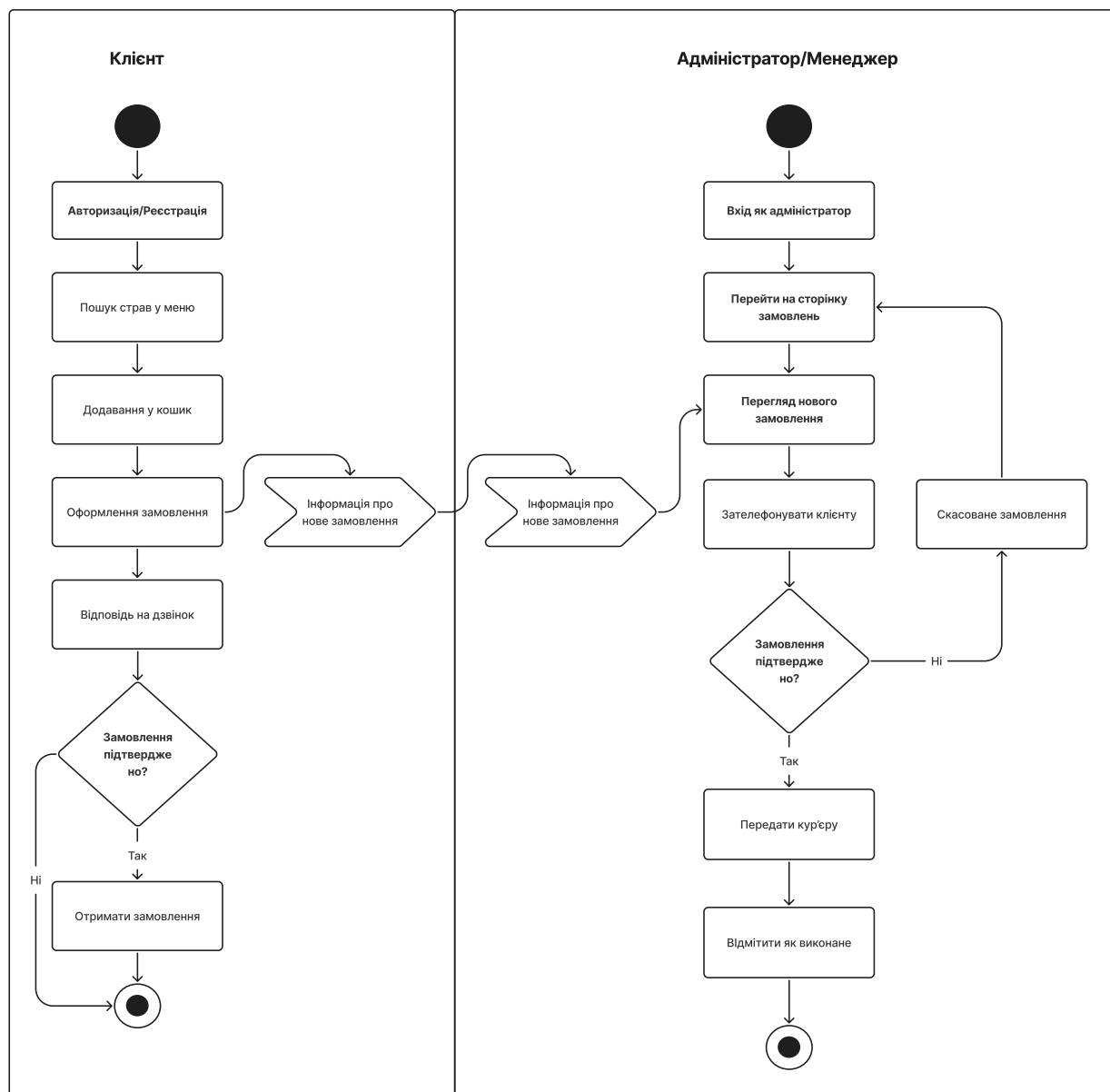


Рисунок 3.10 – Діаграма дій

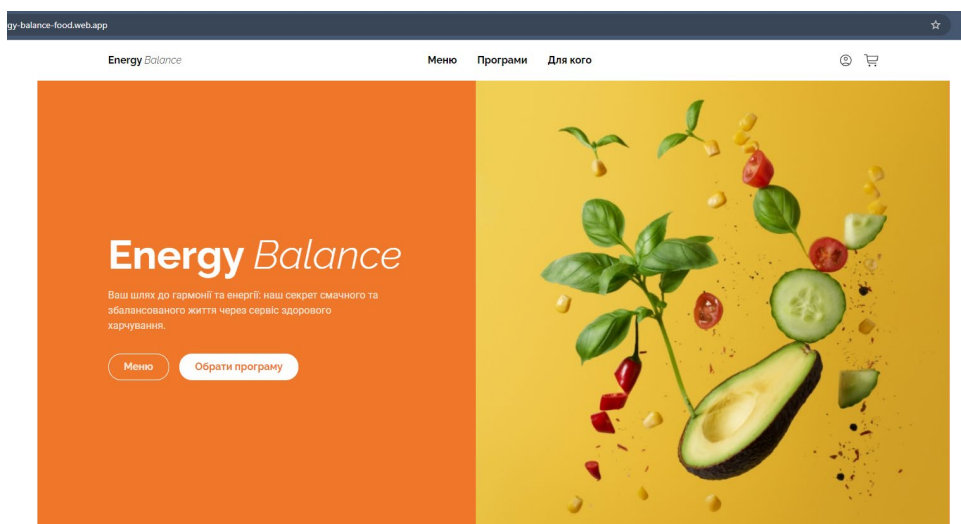
Джерело: розроблено автором самостійно

3.5 Результати реалізації інформаційної системи

В проєкті було реалізовано наступні сторінки:

- Головна сторінка;
- Меню;
- Програми;
- Для кого;
- Профіль;
- Кошик;
- Адмін-панель;
- Сторінка авторизації;
- Сторінка 404 (Not found).

Головна сторінка (рисунок 3.11) містить в собі інформацію яку користувач вперше бачить, зайшовши на сайт. На цій сторінці присутній опис даного сервісу, переваги, програми харчування, калькулятор для обчислення денної норми калорій з урахування різних персональних параметрів, поширені питання і відповіді (рисунок 3.12).



Про Energy Balance

Ваш провідник у світі збалансованого харчування. Смачні страви, персоналізовані плани та здоровий спосіб життя – все в одному сервісі для

Рисунок 3.11 – Головна сторінка сайту

Джерело: розроблено автором самостійно

The image shows a mobile application interface for 'Energy Balance'. At the top, there are three diet plan cards: 'Slim' (1200, 1400, 1600 kcal), 'Balance' (1800, 2000, 2200 kcal), and 'Muscle' (2400, 2600 kcal). Below these is a calculator section titled 'Розрахуйте вашу норму калорій на день' (Calculate your daily calorie norm). It includes a large orange button showing 'Ваш денний раціон складає: 2203 ккал' (Your daily diet consists of: 2203 kcal) and a 'Переглянути меню' (View menu) button. To the right of the calculator are input fields for gender, weight (68 kg), height (170 cm), and age (32 years), along with activity level and goal selection options. At the bottom, there is a 'Запитайте у нас що-небудь' (Ask us anything) section with four expandable questions.

Рисунок 3.12 – Блок калькулятора для обчислення потрібної денної норми калорійності

Джерело: розроблено автором самостійно

На рисунку 3.13 зображена сторінка меню. На цій сторінці присутнє навігаційне по категорія, що знаходиться зліва, а також для зручності пошук за назвою.

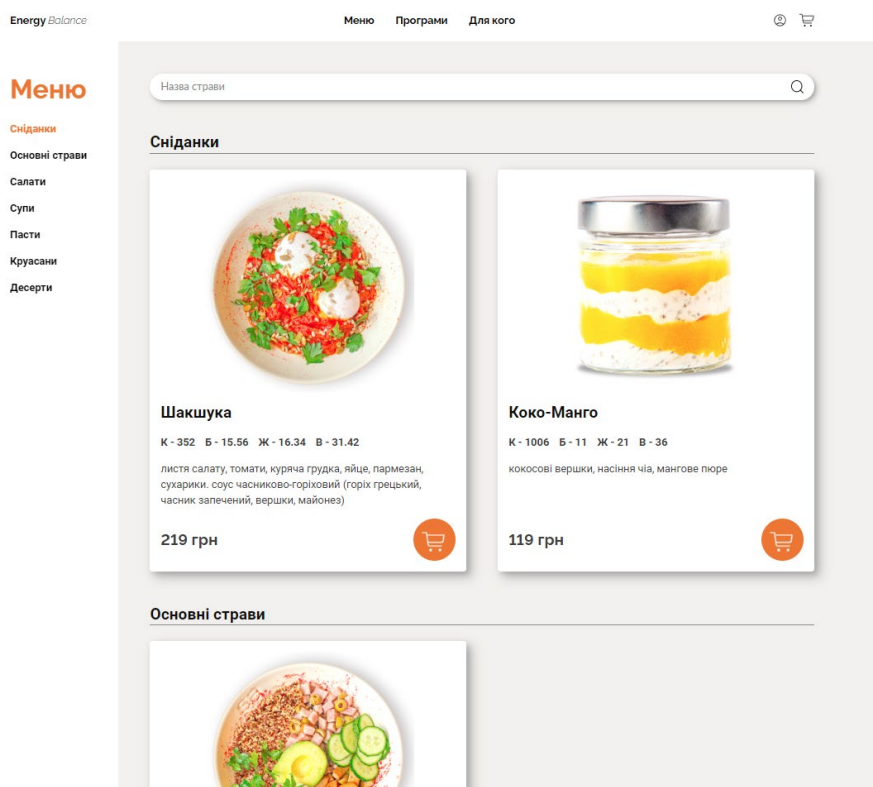


Рисунок 3.13 – Сторінка меню

Джерело: розроблено автором самостійно

Сторінка профілю клієнта містить 3 вкладки з контактною інформацією (рисунок 3.14), замовленнями (рисунок 3.15) і вкладку особливості, де клієнт може написати на який з продуктів він має алергію або просто особисті побажання щодо замовлень (рисунок 3.16).

The screenshot shows the 'Energy Balance' client profile page with the 'Контактна інформація' (Contact Information) tab active. The page contains several input fields for personal and contact details:










- Ваше ім'я:** Daniel
- Прізвище:** Mokhammad
- Емэйл / Логін:** d.one4280@gmail.com
- Дата народження:** 02.02.2000
- Телефон:** +380973334455
- Місто:** Київ
- Вулиця:** Хрещатик
- Будинок:** 12
- Квартира:** 1
- Індекс:** 21412

At the bottom of the form is a 'Зберегти' (Save) button. Below the form is a yellow banner with the text 'Харчуйся з користю разом з Energy Balance' and an image of sliced kiwi and other fruits.

Рисунок 3.14 – Сторінка профілю клієнта з активною вкладкою контактна інформація

Джерело: розроблено автором самостійно

The screenshot shows the 'Energy Balance' client profile page with the 'Мій замовлення' (My orders) tab active. It displays a list of orders with the following details:

Замовлення №	Дата та час	Всього	Дії												
1717775935	07.06.2024, 18:58:55	1623 грн	↑												
<table border="0"> <tr> <td></td> <td>Боул кіноа з тартаром тунця</td> <td>3 шт.</td> <td>349 грн</td> </tr> <tr> <td></td> <td>Коко-Манго</td> <td>3 шт.</td> <td>119 грн</td> </tr> <tr> <td></td> <td>Шахшука</td> <td>1 шт.</td> <td>219 грн</td> </tr> </table>					Боул кіноа з тартаром тунця	3 шт.	349 грн		Коко-Манго	3 шт.	119 грн		Шахшука	1 шт.	219 грн
	Боул кіноа з тартаром тунця	3 шт.	349 грн												
	Коко-Манго	3 шт.	119 грн												
	Шахшука	1 шт.	219 грн												
1717502274	04.06.2024, 14:57:54	119 грн	↓												

At the bottom of the page is a yellow banner with the text 'Харчуйся з користю разом з Energy Balance' and an image of sliced kiwi and other fruits.

Рисунок 3.15 – Сторінка профілю клієнта з активною вкладкою замовлень

Джерело: розроблено автором самостійно

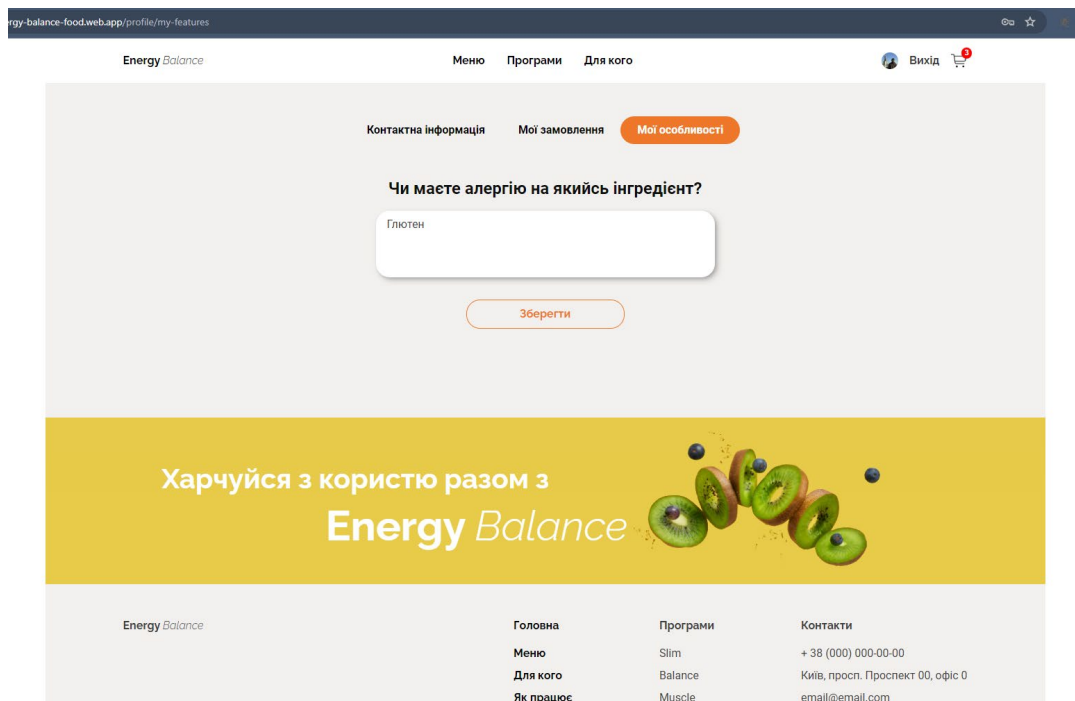


Рисунок 3.16 – Сторінка профілю клієнта з активною вкладкою «мої особливості»
Джерело: розроблено автором самостійно

Сторінка авторизації де клієнт може увійти або зареєструватися чи зайти за допомогою облікового запису Google (рисунок 3.17).

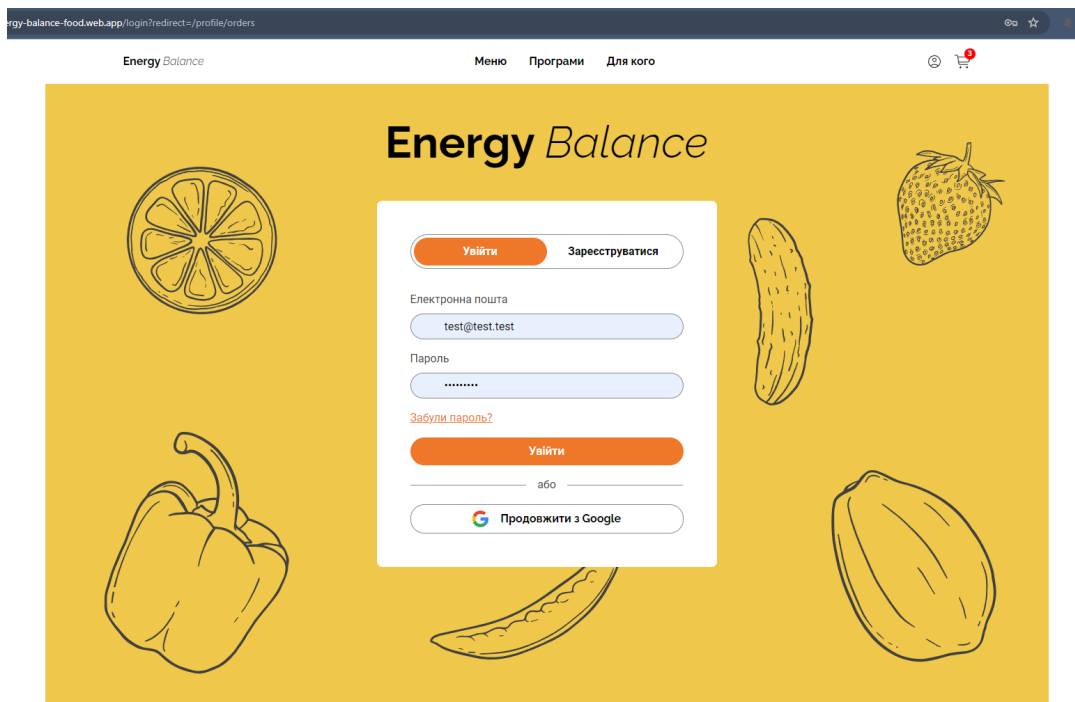
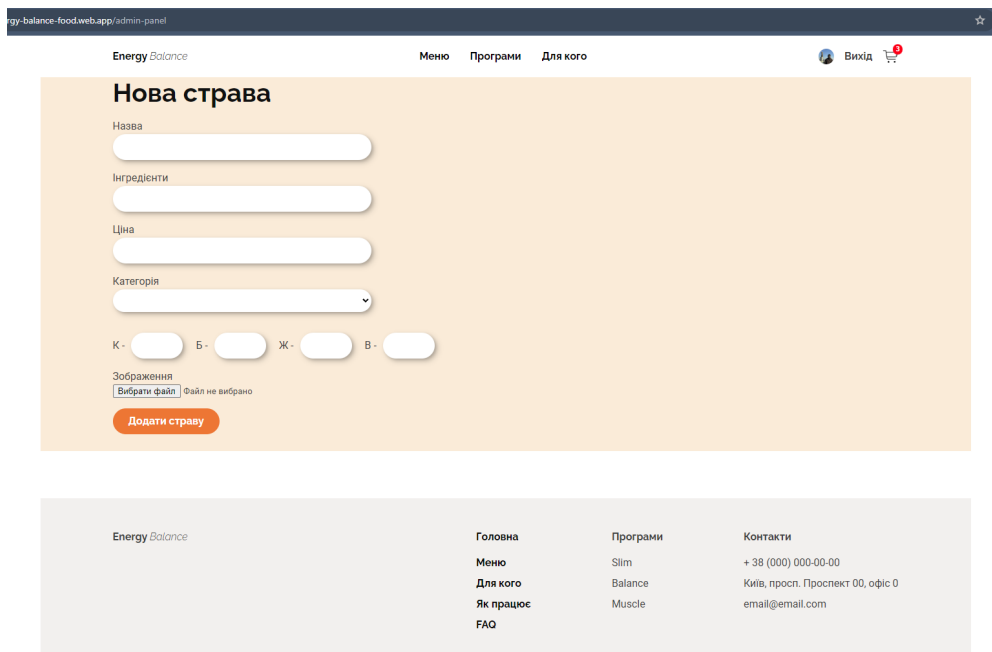


Рисунок 3.17 – Сторінка авторизація

Джерело: розроблено автором самостійно

Сторінка адміністрування (3.18) на якій призначений адміністратор може додавати нові страви до меню.



The screenshot shows the 'Energy Balance' admin panel. At the top, there is a navigation bar with 'Меню', 'Програми', and 'Для кого'. The main content area is titled 'Нова страв' and contains a form with the following fields: 'Назва' (text input), 'Інгредієнти' (text input), 'Ціна' (text input), 'Категорія' (dropdown menu), and 'К - Б - Ж - В' (radio buttons). Below the form is a 'Зображення' section with a 'Вибрати файл' button and a 'Додати страву' button. The footer contains the 'Energy Balance' logo, navigation links (Головна, Меню, Для кого, Як працює, FAQ), program links (Slim, Balance, Muscle), and contact information (+38 (000) 000-00-00, Київ, просп. Проспект 00, офіс 0, email@email.com).

Рисунок 3.18 – Сторінка адміністрування

Джерело: розроблено автором самостійно

Якщо сталась якась помилка, під час виконання запитів або помилка під час авторизації/реєстрації, на екран виводиться вікно помилки (рисунок 3.19), яка свідчить про це.

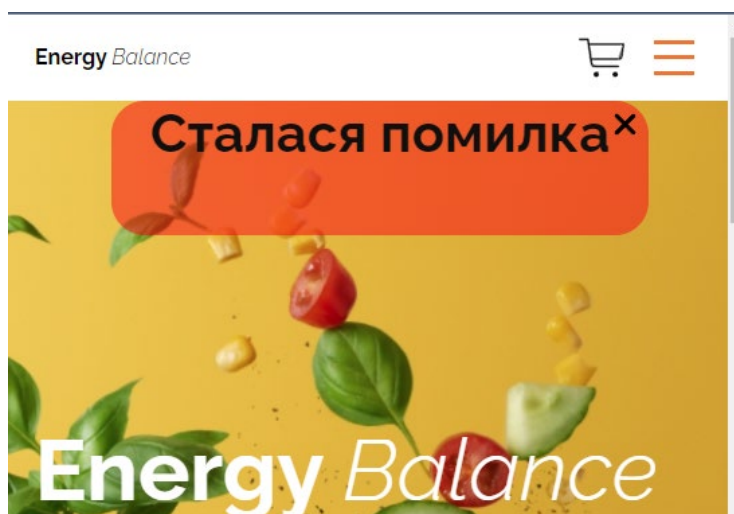


Рисунок 3.19 – Вікно помилки

Джерело: розроблено автором самостійно

Сторінка для кого (рисунок 3.20), на цій сторінці детально описано для кого може підійти даний сервіс.

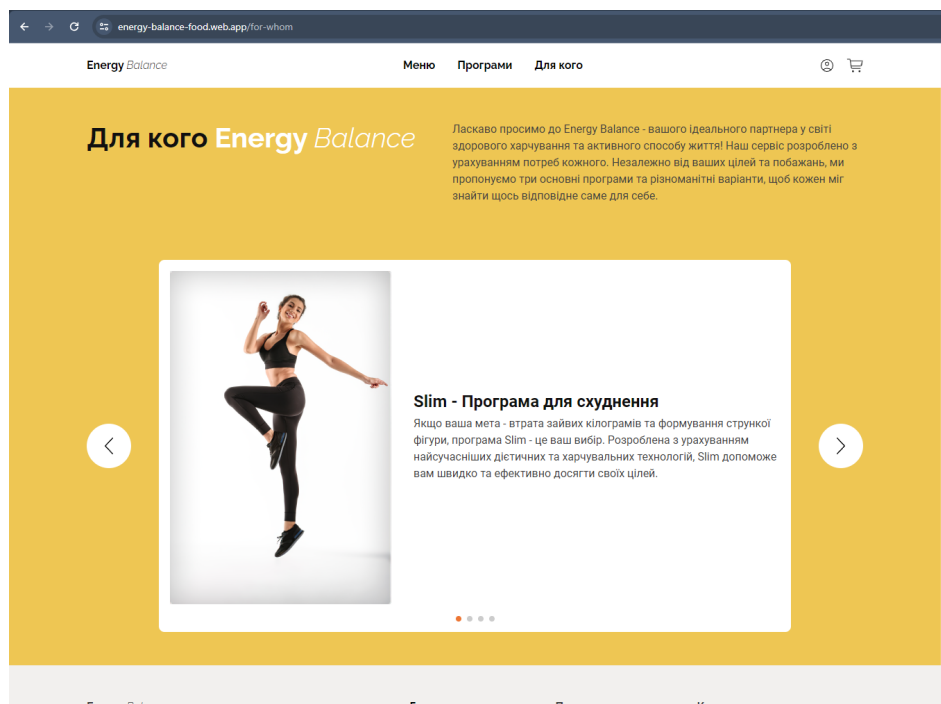


Рисунок 3.20 – Сторінка для кого

Джерело: розроблено автором самостійно

На сторінці, що зображено на рисунку 3.21, описано основні програми харчування, які клієнт може замовити.

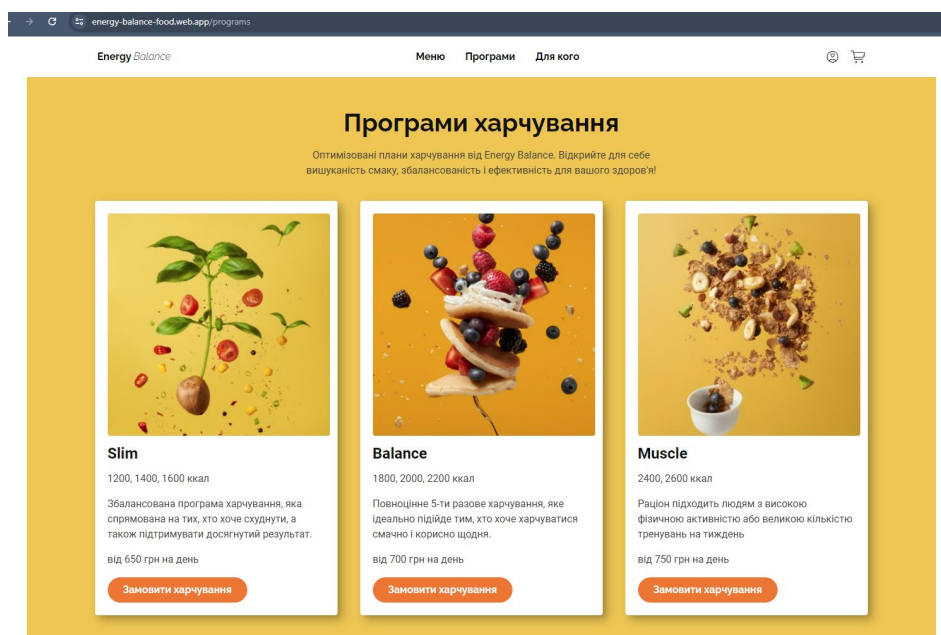


Рисунок 3.21 – Сторінка програм харчування

Джерело: розроблено автором самостійно

На сторінці кошику, що зображено на рисунку 3.22, зображено список страв, які клієнт додав до свого замовлення. На даній сторінці можна збільшити кількість страв за допомогою кнопок «+» та «-», а також при потребі, можна очистити кошик.

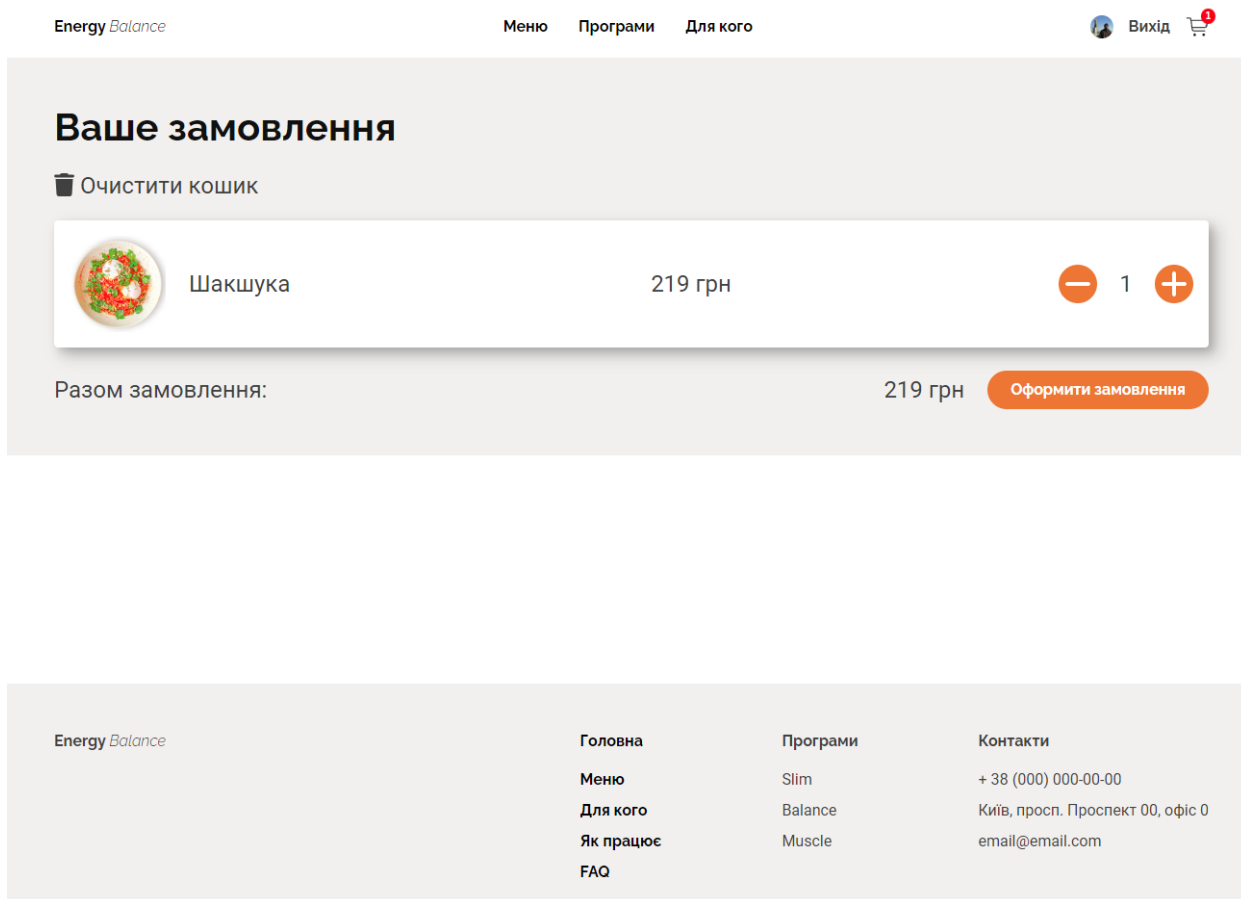


Рисунок 3.22 – Сторінка кошика

Джерело: розроблено автором самостійно

ВИСНОВКИ

У даному проекті було проведено всебічний аналіз існуючих рішень та виявлено основні проблеми, які потребують негайного вирішення для забезпечення ефективної роботи системи. На основі отриманих даних було розроблено та впроваджено ключові функції, такі як управління користувачами, обробка замовлень, генерація звітів та базові інструменти для адміністрування системи. Кожна з цих функцій була ретельно протестована на предмет надійності та ефективності, що дозволило досягти високих показників у їх роботі.

Одним з головних досягнень проекту є створення інтуїтивно зрозумілого інтерфейсу, який дозволяє користувачам швидко і легко виконувати всі необхідні операції. Завдяки цьому інтерфейсу користувачі можуть без зайвих зусиль управляти своїми замовленнями та іншими аспектами взаємодії з системою. Інтерфейс розроблений таким чином, щоб мінімізувати кількість помилок та спростити процес навчання нових користувачів, що є важливим аспектом для забезпечення високого рівня задоволеності клієнтів.

Однак, незважаючи на досягнуті успіхи, проект ще потребує значних вдосконалень для повної відповідності сучасним вимогам ринку. Зокрема, важливо впровадити систему оплати, яка дозволить користувачам здійснювати платежі безпосередньо через платформу. Така функціональність значно покращить користувацький досвід та підвищить зручність використання системи, дозволяючи користувачам швидко та безпечно здійснювати фінансові транзакції. Впровадження системи оплати також відкриє нові можливості для монетизації платформи, що є важливим аспектом для забезпечення її сталого розвитку та фінансової стійкості.

Додатково, рекомендовано розширити адміністративний функціонал, додавши інструменти для аналітики та моніторингу системи в режимі реального часу. Це дасть змогу адміністраторам оперативно реагувати на будь-які виникаючі проблеми та ефективніше управляти ресурсами системи. Аналітичні інструменти дозволять відслідковувати ключові показники ефективності та виявляти області,

що потребують покращення, забезпечуючи тим самим більш ефективне управління платформою.

Вдосконалення проекту передбачає збільшення функціональних можливостей, що дозволить підприємству ефективніше використовувати інформаційні технології у своїй діяльності. Особливу увагу слід звернути на інтеграцію системи оплати та вдосконалення адміністративного інтерфейсу для більш зручного управління системою. Реалізація цих заходів дозволить значно підвищити загальну ефективність та конкурентоспроможність впроваджуваного рішення, що сприятиме успішному розвитку підприємства.

Також варто зазначити, що подальший розвиток проекту може включати впровадження додаткових функцій, таких як автоматизація процесів обробки даних, інтеграція з іншими системами та сервісами, а також розробка мобільних додатків для зручного доступу до платформи з будь-яких пристроїв. Ці кроки дозволять забезпечити більш гнучке та ефективне управління бізнес-процесами, підвищити задоволеність користувачів та сприяти зростанню доходів підприємства.

Таким чином, даний проект є важливим кроком у розвитку інформаційних технологій підприємства, забезпечуючи його сучасними інструментами для ефективного управління ресурсами та покращення взаємодії з клієнтами. Вдосконалення та розширення функціоналу платформи дозволять підприємству досягти нових висот у своїй діяльності, забезпечуючи стабільний ріст та розвиток на ринку.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. ВПЛИВ МОЖЛИВОСТЕЙ ІНТЕРНЕТ-ТЕХНОЛОГІЙ НА РОЗВИТОК ПІДПРИЄМСТВ URL: <http://www.economy.nayka.com.ua/?op=1&z=2264> (дата звернення: 07.03.2024)
2. Здорове харчування: все, що необхідно знати. URL: (www.umj.com.ua/uk/novyna-163044-zdorove-harchuvannya-vse-shho-neobhidno-znati) (дата звернення: 07.03.2024)
3. Що таке збалансоване харчування? URL: <https://healthyfood.org.ua/ua/sho-take-zbalansovane-harchuvannya-i-chomu-vono-vazhlive-dlya-zdorovya/> (дата звернення: 07.03.2024)
4. Дієтичне харчування - <https://eko.zakaz.ua/uk/categories/dietary-eating-ekomarket/base=bran/> (дата звернення: 07.03.2024)
5. НАВИЩО ХАРЧУВАТИСЯ ЗБАЛАНСОВАНО. URL: <https://znaimo.gov.ua/navishcho-kharchuvatysia-zbalansovano/> (дата звернення: 07.03.2024)
6. Як харчуватися збалансовано. URL: <https://moz.gov.ua/article/health/jak-harchuvatisja-zbalansovano> (дата звернення: 07.03.2024)
7. Що таке функціональні вимоги: приклади, визначення, повний посібник. URL: <https://visuresolutions.com/uk/blog/functional-requirements/> (дата звернення: 01.04.2024)
8. НЕФУНКЦІОНАЛЬНІ ВИМОГИ: ПРИКЛАДИ, ТИПИ, ПІДХОДИ. URL: <https://training.qatestlab.com/blog/technical-articles/non-functional-requirements-examples-types-approaches/> (дата звернення: 01.04.2024)
9. Дієтичне харчування - <https://eko.zakaz.ua/uk/categories/dietary-eating-ekomarket/base=bran/> (дата звернення: 07.03.2024)
10. What is a SysML Internal Block Diagram? URL: <https://sysml.org/sysml-faq/what-is-internal-block-diagram.html> (дата звернення: 01.04.2024)
11. What is a SysML Internal Block Diagram? URL: <https://sysml.org/sysml-faq/what-is-internal-block-diagram.html> (дата звернення: 01.04.2024)
12. What is Class Diagram? URL: <https://www.visual->

paradigm.com/guide/uml-unified-modeling-language/what-is-class-diagram/ (дата звернення: 01.04.2024)

13. Cloud Firestore. URL: <https://firebase.google.com/docs/firestore> (дата звернення: 07.03.2024)

14. Даталогічна модель бази даних. URL: https://studwood.net/1553750/ekonomika/datalogichna_model_bazi_danih (дата звернення: 07.03.2024)

15. Що таке діаграма розгортання? URL: <https://www.guru99.com/uk/deployment-diagram-uml-example.html> (дата звернення: 07.03.2024)

16. Повне розуміння діаграми компонентів UML за допомогою легкого методу. URL: <https://www.mindonmap.com/uk/blog/uml-component-diagram/> (дата звернення: 07.03.2024)

17. Для чого потрібне впровадження процесного підходу в управління діяльністю організації? URL: <https://ukraine-oss.com/dlya-chogo-potribne-vprovadzhennya-prochesnogo-pidhodu-v-upravlinnya-diyalnistyu-organizaciyi/> (дата звернення: 07.03.2024)

18. Bring the power of process modelling into your business processes. URL: <https://www.bizagi.com/en/platform/modeler#:~:text=Bizagi%20Modeler%20enables%20organizations%20to,opportunities%20to%20increase%20organizational%20efficiency>. (дата звернення: 07.03.2024)

19. Figma. URL: https://www.figma.com/?gad_source=1&gclid=Cj0KCQjw1N6wBhCcARIsAKZvD5iQg-fh_gx0sEXPhPFoi0OtXVbtD_f1aM5KZRxwBardU5EzK9xc-VoaAjnKEALw_wcB (дата звернення: 07.03.2024)

20. What is Activity Diagram? URL: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-activity-diagram/> (дата звернення: 07.03.2024)

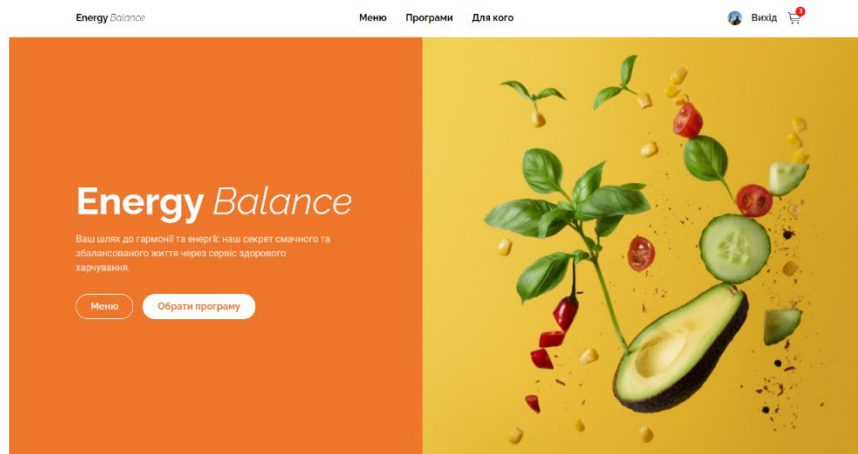
21. Діаграми Прецедентів (Use Case UML Diagram). URL: <https://lvivqaclub.blogspot.com/2008/10/use-case-uml-diagram.html> (дата звернення: 07.03.2024)

07.03.2024)

22. Діаграма послідовності (Sequence Diagrams). URL:
<https://www.maxzosim.com/sequence-diagrams/> (дата звернення: 07.03.2024)

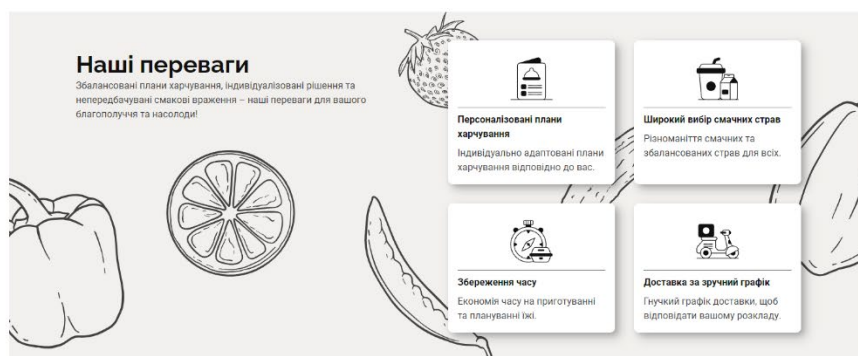
ДОДАТКИ

ДОДАТОК А. ГОЛОВНА СТОРІНКА



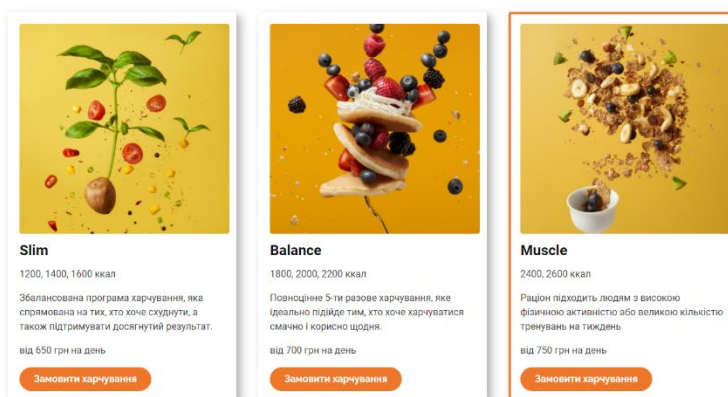
Про Energy Balance

Ваш провідник у світ збалансованого харчування. Смачні страви, персоналізовані плани та здоровий спосіб життя – все в одному сервісі для вашого енергійного та здорового життя!



Програми харчування

Оптимізовані плани харчування від Energy Balance. Відкрийте для себе вишуканість смаку, збалансованість і ефективність для вашого здоров'я!



Розрахуйте вашу норму калорій на день

Ваш денний раціон складає:
2203 ккал

[Переглянути меню](#)

Оберіть статі: чоловік жінка

Рівень активності: 1 2 3 4 5
Мінімум Слабо Середньо Високо Дуже високо

Мета: Зберегти вагу Зменшити вагу Підняти вагу

Ваша вага: кг


Ваш зріст: см

Ваш вік: роки

[Розрахувати калорії](#)

Запитайте у нас що-небудь

- [Як працює Energy Balance?](#) ▼
- [Чи можливий вибір дієтичних обмежень?](#) ▼
- [Які гарантії щодо якості інгредієнтів?](#) ▼
- [Як часто можна змінювати свій план харчування?](#) ▼



-20%
Energy Balance

Даруємо знижку 20%
На пробний традиційний раціон з будь-якого меню

Ваше ім'я:

+38 |

[Замовити зі знижкою](#)

Energy Balance

<p>Головна</p> <p>Меню</p> <p>Для кого</p> <p>Як працює</p> <p>FAQ</p>	<p>Програми</p> <p>Slim</p> <p>Balance</p> <p>Muscle</p>	<p>Контакти</p> <p>+ 38 (000) 000-00-00</p> <p>Київ, пр.Св.Проспект 00, офіс 0</p> <p>email@email.com</p>
--	--	---

[Terms & Conditions](#) [Privacy Policy](#) [Cookie Policy](#)

ДОДАТОК Б. СТОРІНКА МЕНЮ


Energy Balance Меню Програми Для кого Вікна

Назва страви

Меню


- Сніданки
- Основні страви
- Салати
- Супи
- Паста
- Круасани
- Десерти

Сніданки



Шакшука
К - 352 В - 15.56 Ж - 16.34 В - 31.42
листя салату, томати, курча грудка, яйце, пармезан, суроджені, соус чачическо-горизовий (органічний, часник запечений, вершки, майонез)


219 грн



Кокко-Манго
К - 1006 В - 11 Ж - 21 В - 36
кокосові вершки, часник ча, мангосові пюре

119 грн


Основні страви



Боул кіноа з тартаром тунця
К - 1427 В - 39.7 Ж - 77.1 В - 100.7
кіноа, тартар з тунця з оливками, авокадо, яйце пашот, огірок, гриби єршики, петрушка, соус оливково-фундуєвий


349 грн

Салати




Цезар
К - 1006 В - 52.4 Ж - 78.6 В - 23.9
листя салату, томати, курча грудка, яйце, пармезан, суроджені, соус чачическо-горизовий (органічний, часник запечений, вершки, майонез)

239 грн



Сімпл
К - 762 В - 19.2 Ж - 58.6 В - 46.3
кіноа, тофу, морква карамелізована, бобі садзимо, сільковаті, соус айвійсвий (органічний, часник, сирені соус, швейцарський, оліва, майонез, цукор, сіль)


239 грн



Греко
К - 616 В - 11.3 Ж - 57.9 В - 16.6
листя салату, томати, болгарський перець сирійський, огірок, фета в трав'яній олії, оліва оливкова

239 грн


Супи



Бульйон
К - 277 В - 34.8 Ж - 14.8 В - 1.2
бульйон курячий, лігвине, куряче філе запечене, яйце, кроп

119 грн


Паста



Песто Декузе
 К-1079 В-1014 Ж-011 В-160,7
 Салат, приготовленный с соусом песто из базилика, помидоров, чеснока, оливок, сыра пармезан и хлеба грубого помола.

255 грн


Круасани



Круасан Тритона
 К-1088 В-1079 Ж-038 В-124
 Круасан, наполненный итальянскими помидорами, мятой, базиликом, сыром, салатом фреско и соусом маринара.

199 грн

Десерти



Запеканка
 К-1087 В-1024 Ж-78,6 В-25,9
 Сладкий десерт с клубникой и шоколадом, украшенный свежими ягодами и сливками.

149 грн

Харчуйся з користю разом з **Energy Balance**


Енергетичне	Салон	Продукти	Комплекти
	Місто	01+	+38 (0)67 2564000
	Діагност.	01000	Київський Індустріальний Район, м. Київ
	Відділення	Київська	www.energybalance.com
	РАО		
		Київ, А.Сандієв	Рівненський, Сторожинецький

ДОДАТОК В. СТОРІНКА ПРОГРАМ

Energy Balance
Меню Програми Для кого
Вихід

Програми харчування

Оптимізовані плани харчування від Energy Balance. Відкрийте для себе вишуканість смаку, збалансованість і ефективність для вашого здоров'я!




Slim
1200, 1400, 1600 ккал

Збалансована програма харчування, яка спрямована на тих, хто хоче скинути, а також підтримувати досягнутий результат.

Від 650 грн на день

[Замовити харчування](#)




Balance
1800, 2000, 2200 ккал

Повноцінне 5-ти разове харчування, яке ідеально підійде тим, хто хоче харчуватися смачно і корисно щодня.

Від 700 грн на день

[Замовити харчування](#)



Muscle
2400, 2600 ккал

Раціон підходить людям з високою фізичною активністю або великою кількістю тренувань на тиждень.

Від 750 грн на день

[Замовити харчування](#)

Energy Balance

Головна

Меню

Для кого

Як працює

FAQ

Програми

Slim

Balance

Muscle

Контакти

+38 (000) 000-00-00

Київ, просп. Проспект 00, офіс 0

email@email.com

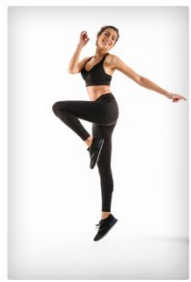
[Terms & Conditions](#) [Privacy Policy](#) [Cookie Policy](#)

ДОДАТОК Г. СТОРІНКА ДЛЯ КОГО

Energy Balance
Меню Програми Для кого
Вихід

Для кого Energy Balance

Ласкаво просимо до Energy Balance – вашого ідеального партнера у світі здорового харчування та активного способу життя! Наш сервіс розроблено з урахуванням потреб кожного. Незалежно від ваших цілей та побажань, ми пропонуємо три основні програми та різноманітні варіанти, щоб кожен міг знайти щось відповідне саме для себе.



Slim - Програма для схуднення

Якщо ваша мета - втрата зайвих кілограмів та формування стрункої фігури, програма Slim - це ваш вибір. Розроблена з урахуванням найсучасніших дієтичних та харчових технологій, Slim допоможе вам швидко та ефективно досягти своїх цілей.

Energy Balance

- Головна
- Меню
- Для кого
- Як працює
- FAQ

Програми

- Slim
- Balance
- Muscle

Контакти

+ 38 (000) 000 00 00

Київ, просп. Проспект 00, офіс 0

email@email.com

Terms & Conditions
Privacy Policy
Cookie Policy

ДОДАТОК Д. СТОРІНКА ПРОФІЛЮ

Energy Balance Меню Програми Для кого Вихід

Контактна інформація **Мі замовлення** Мої особливості

Ваше ім'я: Daniel

Прізвище: Mokhammad

Email / Логін: d.one428@gmail.com

Дата народження: 02.02.2000

Телефон: +380973334455

Місто: Київ

Вулиця: Хрециatik

Будинок: 12

Квартира: 1

Індекс: 21412

[Зберегти](#)

Харчуйся з користю разом з **Energy Balance**

Energy Balance

Головна
Меню
Для кого
Як працює
FAQ




Програми
Slim
Balance
Muscle

Контакти
+38 (000) 000-00-00
Київ, просп. Проспект 00, офіс 0
email@email.com

[Terms & Conditions](#) [Privacy Policy](#) [Cookie Policy](#)

Energy Balance Меню Програми Для кого Вихід

Контактна інформація **Мі замовлення** Мої особливості

Замовлення № 171775935, 07.06.2024, 18:58:55	1623 грн	↑
 Боул кіноа з тартаром тунця	3 шт.	349 грн
 Коко-Манго	3 шт.	119 грн
 Шакшука	1 шт.	219 грн
Замовлення № 1717502274, 04.06.2024, 14:57:54	119 грн	↓

Харчуйся з користю разом з **Energy Balance**



Energy Balance

Головна
Меню
Для кого
Як працює
FAQ

Програми
Slim
Balance
Muscle

Контакти
+38 (000) 000-00-00
Київ, просп. Проспект 00, офіс 0
email@email.com

[Terms & Conditions](#) [Privacy Policy](#) [Cookie Policy](#)


Energy Balance Меню Програми Для кого  Вихід 

[Контактна інформація](#) [Міі замовлення](#) [Міі особливості](#)

Чи маєте алергію на якийсь інгредієнт?

Глютен

[Зберегти](#)

Харчуйтеся з користю разом з **Energy Balance** 

Energy Balance

Головна
Меню
Для кого
Як працює
FAQ

Програми
Slim
Balance
Muscle

Контакти
+38 (000) 000 00 00
Київ, просп. Проспект 00, офіс 0
email@email.com




[Terms & Conditions](#) [Privacy Policy](#) [Cookie Policy](#)

ДОДАТОК Е. СТОРІНКА КОШИКУ

Energy Balance Меню Програми Для кого Вихід

Ваше замовлення

Очистити кошик

	Боул кіноа з тартаром тунця	349 грн	- 1 +
	Сімпл	239 грн	- 1 +
	Цезар	239 грн	- 1 +

Разом замовлення: 827 грн [Оформити замовлення](#)

Energy Balance

Головна	Програми	Контакти
Меню	Slim	+38 (000) 000-00-00
Для кого	Balance	Київ, просп. Проспект 00, офіс 0
Як працює	Muscle	email@email.com
FAQ		

[Terms & Conditions](#) [Privacy Policy](#) [Cookie Policy](#)

ДОДАТОК Є. СТОРІНКА 404 NOT FOUND

Energy Balance

Меню Програми Для кого



#404 Not found

Energy Balance

Головна

Меню

Для кого

Як працює

FAQ

Програми

Slim

Balance

Muscle

Контакти

+38 (000) 000-00-00

Київ, просп. Проспект 00, офіс 0

email@email.com

[Terms & Conditions](#)

[Privacy Policy](#)

[Cookie Policy](#)

ДОДАТОК Ж. СТОРІНКА АВТОРИЗАЦІЇ

Energy Balance

Меню Програми Для кого

Energy Balance

Увійти Зареєструватися

Електронна пошта
test@test.test

Пароль
.....

Забули пароль?

Увійти

або

Продовжити з Google

Energy Balance

Головна
Меню
Для кого
Як працює
FAQ

Програми
Slim
Balance
Muscle

Контакти
+ 38 (000) 000-00-00
Київ, просп. Проспект 00, офіс 0
email@email.com

Terms & Conditions Privacy Policy Cookie Policy

ДОДАТОК 3. ПРИКЛАД КОДУ РЕАЛІЗАЦІЇ ГОЛОВНОЇ СТОРІНКИ

```

<template>
  <main-master-page>
    <div class="hero">
      <div class="container">
        <div class="hero__body">
          <h1 class="hero__title">Energy <span>Balance</span></h1>
          <div class="hero__text">
            Ваш шлях до гармонії та енергії: наш секрет смачного та
            збалансованого життя через сервіс здорового харчування.
          </div>
          <div class="hero__buttons">
            <button @click="$router.push({ name: 'menu' })">Меню</button>
            <button><a href="#programs">Обрати програму</a></button>
          </div>
        </div>
      </div>
    </div>

    <div class="about">
      <div class="about__body">
        <h2 class="about__title">
          <span>Про</span> <span>Energy</span> <span>Balance</span>
        </h2>
        <div class="about__text">
          Ваш провідник у світі збалансованого харчування. Смачні страви,
          персоналізовані плани та здоровий спосіб життя – все в одному сервісі
          для вашого енергійного та здорового життя!
        </div>
      </div>
    </div>

    <div class="advantages">
      <div class="container">
        <div class="advantages__body">
          <div class="advantages__about">
            <h2 class="advantages__title">Наші переваги</h2>
            <div class="advantages__text">
              Збалансовані плани харчування, індивідуалізовані рішення та
              непередбачувані смакові враження – наші переваги для вашого
              благополуччя та насолоди!
            </div>
          </div>
          <div class="advantages__cards">
            <div class="advantages__card">
              <div class="advantages__card-img">
                
              </div>
              <div class="advantages__card-title">
                Персоналізовані плани харчування
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </main-master-page>
</template>

```

```

    </div>
    <div class="advantages__card-about">
        Індивідуально адаптовані плани харчування відповідно до вас.
    </div>
</div>

<div class="advantages__card">
    <div class="advantages__card-img">
        
    </div>
    <div class="advantages__card-title">
        Широкий вибір смачних страв
    </div>
    <div class="advantages__card-about">
        Різноманіття смачних та збалансованих страв для всіх.
    </div>
</div>

<div class="advantages__card">
    <div class="advantages__card-img">
        
    </div>
    <div class="advantages__card-title">Збереження часу</div>
    <div class="advantages__card-about">
        Економія часу на приготуванні та плануванні їжі.
    </div>
</div>

<div class="advantages__card">
    <div class="advantages__card-img">
        
    </div>
    <div class="advantages__card-title">
        Доставка за зручний графік
    </div>
    <div class="advantages__card-about">
        Гнучкий графік доставки, щоб відповідати вашому розкладу.
    </div>
</div>
</div>
</div>
</div>

<programs-block />
<energy-calc />

<div class="faq">
    <div class="container">
        <div class="faq__body">

```

```

<h2 class="faq__title">Запитайте у нас що-небудь</h2>
<div class="faq__questions">
  <o-collapse
    v-for="(collapse, index) of collapses"
    :key="index"
    class="card"
    animation="slide"
    :open="position == index"
    @open="
      position = index;
      isOpen = true;
    "
    @close="isOpen = false"
  >
    <template #trigger>
      <div class="card-header" role="button">
        <p class="card-header-title">
          {{ collapse.title }}
        </p>
        <a class="card-header-icon"
          ></a>
        </div>
      </template>
      <div class="card-content">
        <div class="content" v-html="collapse.text" />
      </div>
    </o-collapse>
  </div>
</div>
</div>

<div class="discount">
  <div class="container">
    <div class="discount__body">
      <div class="discount__label">
        <div class="discount__label-size">-20%</div>
        <div class="discount__label-title">Energy <span>Balance</span></div>
      </div>
      <div class="discount__info">
        <div class="discount__title">Даруємо знижку 20%</div>
        <div class="discount__about">
          На пробний триденний раціон з будь-якого меню
        </div>
        <div class="discount__inputs">
          <input type="text" placeholder="Ваше ім'я" class="main-input" />
          <input type="text" placeholder="+ 38 (" class="main-input" />
        </div>
      </div>
    </div>
  </div>

```

```

        </div>
        <button class="discount__button">Замовити зі знижкою</button>
      </div>
    </div>
  </div>
</div>
</main-master-page>
</template>

<script>
import MainMasterPage from "@/masterpages/MainMasterPage.vue";
import EnergyCalc from "@/components/EnergyCalc.vue";
import ProgramsBlock from "@/components/ProgramsBlock";
import collapses from "@/constants/collapses";

export default {
  name: "HomeView",
  components: { MainMasterPage, EnergyCalc, ProgramsBlock },
  data() {
    return {
      position: null,
      isOpen: false,
      collapses: collapses,
    };
  },
};
</script>

<style lang="scss" scoped>
.hero {
  &__body {
    background: url("@/assets/backgrounds/avocado.jpeg") right/auto 100%
      no-repeat,
      rgb(239, 119, 42);
    padding: 239px 112px;
  }
  &__title {
    span {
      font-style: italic;
    }
  }
  &__text {
    color: rgba(255, 255, 255, 0.8);
    max-width: 435px;
    margin: 16px 0 0 0;
  }
  &__buttons {
    display: flex;
    gap: 16px;
    margin: 32px 0 0 0;
  }
}

```

```
:first-child {
  border: 1px solid white;
  color: rgba(255, 255, 255, 0.8);
}
:last-child {
  background-color: white;
  color: rgb(239, 119, 42);
}
}
}
}
.about {
  &__body {
    text-align: center;
    padding: 80px 0;
  }

  &__title {
    :nth-child(2) {
      color: rgb(239, 119, 42);
    }
    :nth-child(3) {
      color: rgb(239, 119, 42);
      font-style: italic;
    }
  }

  &__text {
    margin: 0 auto;
    margin-top: 16px;
    max-width: 600px;
  }
}
.advantages {
  &__body {
    background: url("@/assets/backgrounds/advantages.svg") no-repeat,
      rgb(242, 240, 238);
    padding: 48px 112px;
    display: flex;
    justify-content: space-between;
    gap: 20px;
  }

  &__about {
    margin: 16px 0 0 0;
    max-width: 490px;
  }

  &__title {
  }
}
```

```
&__text {
}

&__cards {
  max-width: 600px;
  display: grid;
  grid-template-columns: 1fr 1fr;
  gap: 32px;
}

&__card {
  border-radius: 8px;
  box-shadow: 8px 8px 16px 0px rgba(0, 0, 0, 0.22),
    4px 4px 8px 0px rgba(0, 0, 0, 0.12);
  background-color: white;
  width: 280px;
  padding: 16px;
  &-img {
    border-bottom: 1px solid rgb(136, 136, 136);
    img {
      display: block;
      margin: 0 auto;
      padding: 8px 0;
    }
  }
}

&-title {
  color: rgb(16, 16, 16);
  font-weight: bold;
  margin: 8px 0 0 0;
}

&-about {
  margin: 8px 0 0 0;
}
}

.programs {
  padding: 80px 0 72px 0;
}

.faq {
  &__body {
    padding: 80px 0;
  }
}

&__title {
  text-align: center;
}

&__questions {
```

```
margin: 0 auto;
margin-top: 32px;
max-width: 800px;
}
.card {
  background-color: #fff;
  border-bottom: 1px solid rgb(136, 136, 136);
  max-width: 100%;
  position: relative;
}

.card-header {
  background-color: transparent;
  align-items: stretch;
  display: flex;
}

.card-header-title {
  font-size: 20px;
  font-weight: 700;
  color: rgb(16, 16, 16);

  align-items: center;
  display: flex;
  flex-grow: 1;
  padding: 0.75rem;
  margin: 0;
}

.card-header-icon {
  align-items: center;
  cursor: pointer;
  display: flex;
  padding: 0.75rem;
  justify-content: center;
}

.card-content {
  padding: 1.5rem;
  background-color: transparent;
}

.open {
  transform: rotate(180deg);
}
}

.discount {
  &__body {
    background: url("@/assets/img/qiwi.png") 330px no-repeat,
      linear-gradient(
        90deg,
```

```
    rgb(216, 180, 31),
    rgb(216, 179, 32) 22.727%,
    rgb(232, 203, 73) 56.79%,
    rgb(232, 203, 73) 100%
  );
padding: 56px 320px 31px 192px;
display: flex;
justify-content: space-between;
}

&__info {
  padding: 0 0 25px 0;
}

&__inputs {
  display: flex;
  flex-direction: column;
  gap: 16px;
  max-width: 280px;
  margin: 16px 0 0 0;
}

&__title {
  font-family: "Raleway", sans-serif;
  color: rgb(16, 16, 16);
}

&__about {
  margin: 8px 0 0 0;
  max-width: 280px;
}

&__button {
  width: 100%;
  margin: 40px 0 0 0;
}

&__label {
  align-self: flex-end;
  &-title {
    color: rgb(255, 255, 255);
    font-family: "Raleway", sans-serif;
    font-size: 41.89px;
    font-weight: 300;
    line-height: 50.27px;
    letter-spacing: 0%;
    text-align: left;
    span {
      font-style: italic;
      font-weight: 300;
    }
  }
}

&-size {
```

```
color: rgb(255, 255, 255);
font-family: "Raleway", sans-serif;
font-size: 78.18px;
line-height: 87.96px;
text-align: center;
}
}
}
</style>
```

ДОДАТОК І. ПРИКЛАД КОДУ РЕАЛІЗАЦІЇ ЛОКАЛЬНОГО СХОВИЩА ДОДАТКУ (STORE)

```

import DbOperations from "../helpers/DbOperations";
const collectionDB = new DbOperations("items");
import getItemsByCategory from "@/store/helpers/sortItemsList";
import isCorrespondToFilter from "@/store/helpers/filter";

import { defineStore } from "pinia";
import { useGeneralStore } from "../general";

export const useItemsStore = defineStore("items", {
  state: () => ({
    // Дані які можуть використати компоненти
    itemList: [],
    isLoading: null,
    searchByName: null,
  }),
  getters: {
    // Геттер отримання товару за ідентифікатором
    getItemById: (state) => (itemId) =>
      state.itemList.find((item) => item.id == itemId),

    // Геттер отримання відфільтрованого списку товарів
    getFilteredItemList: (state) =>
      state.itemList.filter((item) =>
        isCorrespondToFilter(item, state.searchByName)
      ),

    // Геттер отримання відфільтрованого списку товарів по категоріям
    getItemsByCategory: (state) =>
      getItemsByCategory(state.getFilteredItemList),
  },
  actions: {
    // Метод для зберігання списку товарів
    setItemList(itemList) {
      this.itemList = itemList;
      this.isLoading = true;
    },
    // Метод завантаження списку товарів з бази даних
    loadList() {
      const generalStore = useGeneralStore();
      generalStore.setError(null);
      generalStore.setLoading(true);

      collectionDB
        .loadItemList()
        .then((list) => {
          this.setItemList(list);
        })
    }
  }
});

```

```

        .catch((error) => {
            generalStore.setError(error);
        })
        .finally(() => {
            generalStore.setLoading(false);
        });
    },
    // Метод для додавання товару
    addItem(item) {
        const generalStore = useGeneralStore();
        generalStore.setError(null);
        generalStore.setLoading(true);
        collectionDB
            .addItem(item)
            .then(() => {
                this.loadList();
            })
            .catch((error) => {
                generalStore.setError(error);
            })
            .finally(() => {
                generalStore.setLoading(false);
            });
    },
    // Метод отримання товару по ідентифікатору
    getItemById(itemId) {
        const generalStore = useGeneralStore();
        generalStore.setError(null);
        generalStore.setLoading(true);
        collectionDB
            .getItemById(itemId)
            .catch((error) => generalStore.setError(error));
    },
    // Метод для встановлення фільтра за назвою
    setSearchByName(name) {
        this.searchByName = name;
    },
    },
    });

```

Имя пользователя:
Інформаційних систем в економіці Шкуратовська Те...

ID проверки:
1016345319

Дата проверки:
11.06.2024 04:11:44 EEST

Тип проверки:
Doc vs Internet + Library

Дата отчета:
11.06.2024 06:44:50 EEST

ID пользователя:
100005745

Название файла: Мохаммад_Д_ІН-404

Количество страниц: 30 Количество слов: 4089 Количество символов: 34246 Размер файла: 7.09 MB ID файла: 1016146102

Обнаружены модификации текста (могут влиять на процент совпадений)

13.2% Совпадения

Наибольшее совпадение: 3.91% с источником из Библиотеки (ID файла: 1015232085)

8.54% Источники из Интернета 312 Страница 32

11.5% Источники из Библиотеки 382 Страница 33

0% Цитат

Исключение цитат выключено

Исключение списка библиографических ссылок выключено

0% Исключений

Нет исключенных источников

Модификации

Обнаружены модификации текста. Подробная информация доступна в онлайн-отчете.

Замененные символы 8

Подозрительное форматирование 5 страниц

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ
ІМЕНІ ВАДИМА ГЕТЬМАНА
Навчально-науковий інститут
«Інститут інформаційних технологій в економіці»
Кафедра інформаційних систем в економіці

ОСВІТНЬО ПРОФЕСІЙНА ПРОГРАМА «КОМП'ЮТЕРНІ НАУКИ»
галузь знань 12 «Інформаційні технології»
спеціальність 122 «Комп'ютерні науки»

Форма навчання: денна

КВАЛІФІКАЦІЙНИЙ БАКАЛАВРСЬКИЙ ПРОЕКТ

на тему

РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ **SMART-СЕРВІС**
ЗДОРОВОГО ХАРЧУВАННЯ

здобувача Мохаммада Даніеля Наїмовича

Науковий керівник:

д.е.н., проф.

Ріппа С.П.

Проект допущений до захисту

перед екзаменаційною комісією з

атестації здобувачів вищої освіти

завідувач кафедри:

к.е.н., доцент.

Тішков Б.О.

Київ 2024