

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

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

Кафедра програмних засобів
(повне найменування кафедри)

Пояснювальна записка
до дипломного проєкту (роботи)

магістр

(ступінь вищої освіти)

на тему: ДОСЛІДЖЕННЯ ТА ПРОГРАМНА РЕЛІЗАЦІЯ ІНФОРМАЦІЙНО-
АНАЛІТИЧНОЇ СИСТЕМИ ДЛЯ ВИБОРУ ЗАКЛАДУ ХАРЧУВАННЯ ЗА
БАЖАНИМИ КРИТЕРІЯМИ
RESEARCH AND SOFTWARE IMPLEMENTATION OF DATA ANALYTICS
SYSTEM FOR SELECTING A CATERING ESTABLISHMENT REGARDING
THE DESIRED CRITERIA

Виконав: студент(ка) 2 курсу, групи КНТ-219м
Спеціальності _____

122 Комп'ютерні науки

(код і найменування спеціальності)

Освітня програма (спеціалізація) _____

Системи штучного інтелекту

Брильов Є.С.

(прізвище та ініціали)

Керівник Степаненко О.О.

(прізвище та ініціали)

Рецензент Гофман Є.О.

(прізвище та ініціали)

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

Інститут, факультет ПРЕ, ФКНТ
 Кафедра програмних засобів
 Ступінь вищої освіти магістр
 Спеціальність 122 Комп'ютерні науки
 (код і найменування)
 Освітня програма (спеціалізація) Системи штучного інтелекту
 (назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ
 Завідувач кафедри ПЗ, д.т.н, проф.
С.О. Субботін
 “ ” 20 року

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

Брильова Євгенія Сергійовича
 (прізвище, ім'я, по батькові)

1. Тема проєкту (роботи) Дослідження та програмна релізація інформаційно-аналітичної системи для вибору закладу харчування за бажаними критеріями.
Research and Software Implementation of Data Analytics System for Selecting a Catering Establishment Regarding the Desired Criteria

керівник проєкту (роботи) Степаненко Олександр Олексійович, к.т.н., доцент,
 (прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від “ 3 ” листопада 2020 року № 301

2. Строк подання студентом проєкту (роботи) 5 грудня 2020 року

3. Вихідні дані до проєкту (роботи) технічне завдання

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз предметної області. 2. Розробка інформаційної моделі програмних засобів для вибору харчування за бажаними критеріями. 3. Основні рішення щодо реалізації компонентів системи. 4. Керівництво програміста. 5. Керівництво оператора. 6. Економіко-організаційна частина. 7. Охорона праці та безпека в надзвичайних ситуаціях.

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

6. Консультанти розділів проєкту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-5	Степаненко О.О., доцент		
6	Пожуєва Т.О., професор		
7	Коробко О.В., ст. викладач		
Нормоконтролер	Калініна М.В., асистент		

7. Дата видачі завдання “ 4 ” вересня 2020 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Аналіз предметної області.	1 тиждень	
2	Розробка та затвердження технічного завдання.	2 тиждень	Технічне завдання
3	Проектування структури системи.	3-4 тиждень	Структурна схема
4	Розробка схеми функціонування системи та модулів програми.	5-6 тиждень	Функціональна схема. Модулі
5	Створення програмного коду системи.	7-8 тиждень	Текст програми
6	Тестування та відлагодження програми.	9 тиждень	Працездатна система
6	Тестування та відлагодження програми.	9 тиждень	Працездатна система
7	Розробка розділів пояснювальної записки.	10 тиждень	ПЗ
8	Розробка слайдів презентації.	11-12 тиждень	Слайди презентації
9	Захист роботи.	13 тиждень	

Студент(ка)

(підпис) Брильов Є.С.
(прізвище та ініціали)

Керівник проєкту (роботи)

(підпис) Степаненко О.О.
(прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи магістра:
127 с., 32 рис., 11 табл., 4 додатки, 37 джерел.

ВЕБСАЙТ, ВИБІР ЗАКЛАДУ ХАРЧУВАННЯ, ПРОГРАМНЕ
ЗАБЕЗПЕЧЕННЯ, РНР, КРИТЕРІЇ.

Об'єкт дослідження – процес розробки програмних засобів.

Предмет дослідження – програмні засоби для вибору та замовлення продуктів харчування.

Мета роботи – дослідження та реалізація програмних засобів для вибору харчування за бажаними критеріями.

Матеріали, методи та технічні засоби: структурне та об'єктно-орієнтоване програмування, мова програмування РНР, IBM-сумісний комп'ютер з монітором, клавіатурою та маніпулятором типу «миша», що функціонує під управлінням операційної системи.

Результати. Створено програмні засоби для вибору закладу харчування за бажаними критеріями. Розроблена програмна система може успішно експлуатуватися на комп'ютерах під управлінням операційних систем та використовуватися для вибору закладу харчування.

Висновки. Розроблено програмне забезпечення для вибору закладу харчування за бажаними критеріями. Проаналізовано предметну область, розроблено технічне завдання для реалізації програми, обрано середовище розробки, розроблено структурну схему програми, виконано конструювання програмного забезпечення для вибору закладу харчування за бажаними критеріями, здійснено тестування розробленого програмного забезпечення.

Галузь використання – підтримка процесів організації харчування клієнтів.

ABSTRACT

Explanatory note to the diploma qualifying work of the master: 127 pages, 32 figures, 11 tables, 4 appendixes, 37 sources.

WEBSITE, SELECTION OF FOOD INSTITUTION, SOFTWARE, PHP, CRITERIA.

Object of study is a process of software development.

Subject of study is a software for selecting and ordering food.

The purpose of the work is to study and develop of software for food selection according to the desired criteria.

Materials, methods, and tools: structured and object-oriented programming, programming language PHP, computer with a monitor, keyboard and mouse manipulator operating under the control of the operating system.

Results. The software tools for selection a catering establishment according to the desired criteria are developed. The developed software system can successfully operate on computers running operating system and can be used to select a food establishment.

Conclusions. Software for selection a catering establishment according to the desired criteria has been developed. The subject area is analyzed, the technical task for program realization is developed, the development environment is chosen, the structural scheme of the program is developed, the software for a food establishment by desirable criteria is executed, the developed software is tested.

The field of use is a support for customer catering processes.

ЗМІСТ

	С.
Перелік скорочень та умовних позначок	8
Вступ.....	9
1 Аналіз предметної області.....	11
1.1 Огляд існуючого програмного забезпечення	11
1.1.1 Каталог закладів «Tomato.ua».....	11
1.1.2 Каталог закладів «Guide-cafe»	12
1.1.3 NoRecs - Україна.....	13
1.1.4 Сайт Booze.com.ua	14
1.2 Порівняльна таблиця існуючих продуктів	15
1.3 Постановка завдання роботи.....	15
1.4 Висновки до розділу 1	16
2 Розробка інформаційної моделі програмних засобів для вибору харчування за бажаними критеріями	17
2.1 Розробка абстрактних моделей програмного забезпечення на основі структурних діаграм	17
2.2 Розробка абстрактних моделей програмного забезпечення на основі діаграм поведінки.....	19
2.3 Розробка абстрактних моделей програмного забезпечення на основі діаграм взаємодії	21
2.4 Висновки за розділом 2	23
3 Основні рішення щодо реалізації компонентів системи.....	24
3.1 Вибір мови програмування та інших засобів розробки	24
3.2 Вибір інструментарію розробки програмного забезпечення	30
3.3 Архітектура програмного забезпечення для вибору харчування за бажаними критеріями	32
3.4 Розробка інтерфейсу програмного забезпечення	33
3.5 Тестування розробленого програмного забезпечення для вибору харчування за бажаними критеріями	35

3.6 Висновки за розділом 3	37
4 Керівництво програміста.....	39
4.1 Призначення й умови застосування програми.....	39
4.2 Характеристики програми.....	39
4.3 Звернення до програми.....	40
4.4 Вхідні і вихідні дані	40
4.5 Повідомлення	40
5 Керівництво оператора	41
5.1 Призначення програми	41
5.2 Умови виконання програми	41
5.3 Виконання програми.....	42
5.4 Повідомлення оператору	42
6 Економіко-організаційна частина.....	43
6.1 Планування розробки програмного продукту.....	43
6.2 Визначення витрат на розробку програми	44
6.3 Розрахунок економічної ефективності програмного продукту.....	51
7 Охорона праці та безпека в надзвичайних ситуаціях.....	54
7.1 Аналіз потенційних небезпек.....	54
7.2 Заходи із забезпечення безпеки	55
7.3 Заходи з виробничої санітарії та гігієни праці.....	59
7.4 Заходи безпеки у надзвичайних ситуаціях	66
7.4.1 Заходи з пожежної безпеки	66
7.4.2 Заходи з цивільного захисту	67
Висновки	71
Перелік джерел посилання	73
Додаток А Технічне завдання	77
Додаток Б Опис програми	81
Додаток В Текст програми.....	84
Додаток Д Слайди презентації.....	119

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

PHP – Hypertext Preprocessor (гіпертекстовий препроцесор);

UML – Unified Modeling Language (уніфікована мова моделювання);

ОС – операційна система;

ПЗ – програмне забезпечення;

ПК – персональний комп'ютер;

ПП – програмний продукт.

ВСТУП

На фоні різкого розвитку ресторанної галузі та туризму все частіше почала підійматись тема пошуку, навігації та вибору страв і місця харчування у процесі відпочинку, віддаленої роботи, або проведення будь-яких заходів, як ділових так і культурних [1]-[6]. Цю проблему частково вирішив «Google», шляхом розробки своєї інтерактивної мапи. Але розвиток індустрії та взагалі світових тенденцій не змусив себе довго чекати [1]-[6].

Відомі сервіси з вибору та замовлення харчування мають певні недоліки, зокрема, вони часто є безнадійно застарілими (мають застарілий і не зовсім зручний інтерфейс), як правило, є орієнтованими під конкретний заклад харчування та не можуть бути модифікованими для роботи з іншими закладами (це обумовлює, окрім розробки програмного забезпечення, також і створення відповідних інформаційних моделей, що дозволять створювати подібне програмне забезпечення у майбутньому, а також легко модифікувати розроблюване ПЗ) [1]-[10]. Таким чином питання актуальності вирішується вже на етапі ідеї про подібний продукт. І потенційно, по ходу розвитку ідеї, сервіс може вийти за національні рамки і досягнути світового масштабу.

Такими чином, виникла необхідність в створенні програмного продукту, що зможе задовільнити запит суспільства на швидкий пошук та вибір ресторанних страв, а також одночасно мати можливість швидкої адаптації до появи нових умов.

Розроблюваний програмний продукт буде являти собою своєрідний онлайн-каталог, функціоналом якого буде:

- можливість вибору ресторанної справи за заданими критеріями;
- публікація головної інформації про усі можливості (страви) закладу харчування;
- власна інтерактивна мапа.

Метою роботи є дослідження та реалізація програмних засобів для вибору харчування за бажаними критеріями.

Для досягнення поставленої мети необхідно було розв'язати такі завдання:

- аналіз предметної області, зокрема вебсайтів для вибору харчування за бажаними критеріями;
- розробка технічного завдання для реалізації програми;
- вибір засобів розробки програмного забезпечення;
- розробка інформаційної моделі програмних засобів для вибору харчування за бажаними критеріями;
- конструювання програмного забезпечення для вибору харчування за бажаними критеріями;
- тестування розробленого програмного забезпечення.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Огляд існуючого програмного забезпечення

Різноманітність вибору продуктів харчування вже давно наштовхнула розробників програмного забезпечення на думку про необхідність подібних програмних продуктів. Тим не менш, їх кількість досить маленька, а функціонал залишає бажати кращого [1]-[10].

У виборі між мобільним додатком та вебсайтом, як демонструє практика, частіше обирають саме створення сайтів [1]-[6]. Вони і будуть розглянуті в рамках роботи.

1.1.1 Каталог закладів «Tomato.ua»

Головну сторінку сервісу «Tomato.ua» [7] наведено на рис. 1.1.

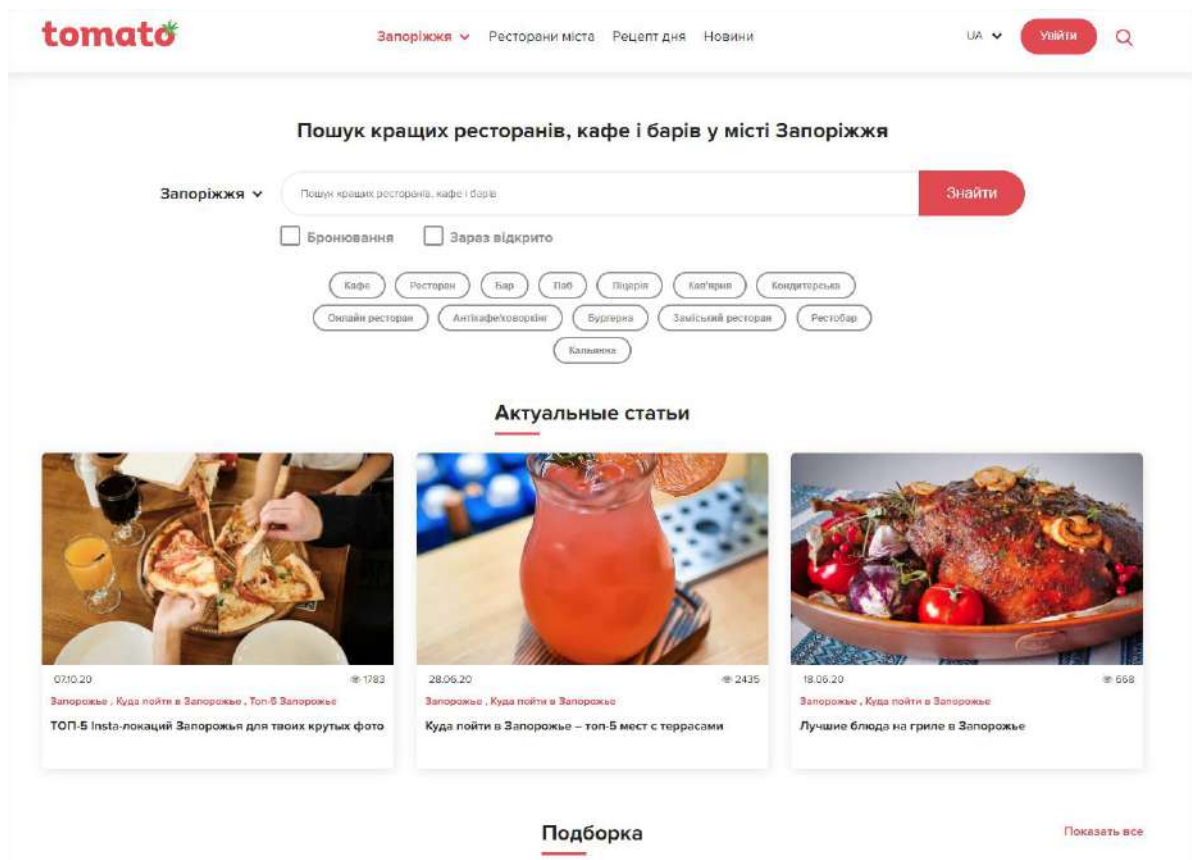


Рисунок 1.1 – Головна сторінка «каталог Tomato»

Це найбільш популярний сервіс, який пропонує подібні види послуг (рис 1.1).

Основні переваги :

- зручний інтерфейс;
- велика база закладів;
- безкоштовність;
- наявність власного кабінету з історією відгуків та оцінок;
- можливість роботи з необмеженою кількістю користувачів.

1.1.2 Каталог закладів «Guide-cafe»

Головну сторінку каталогу «Guide-cafe» [8] наведено на рисунку 1.2.

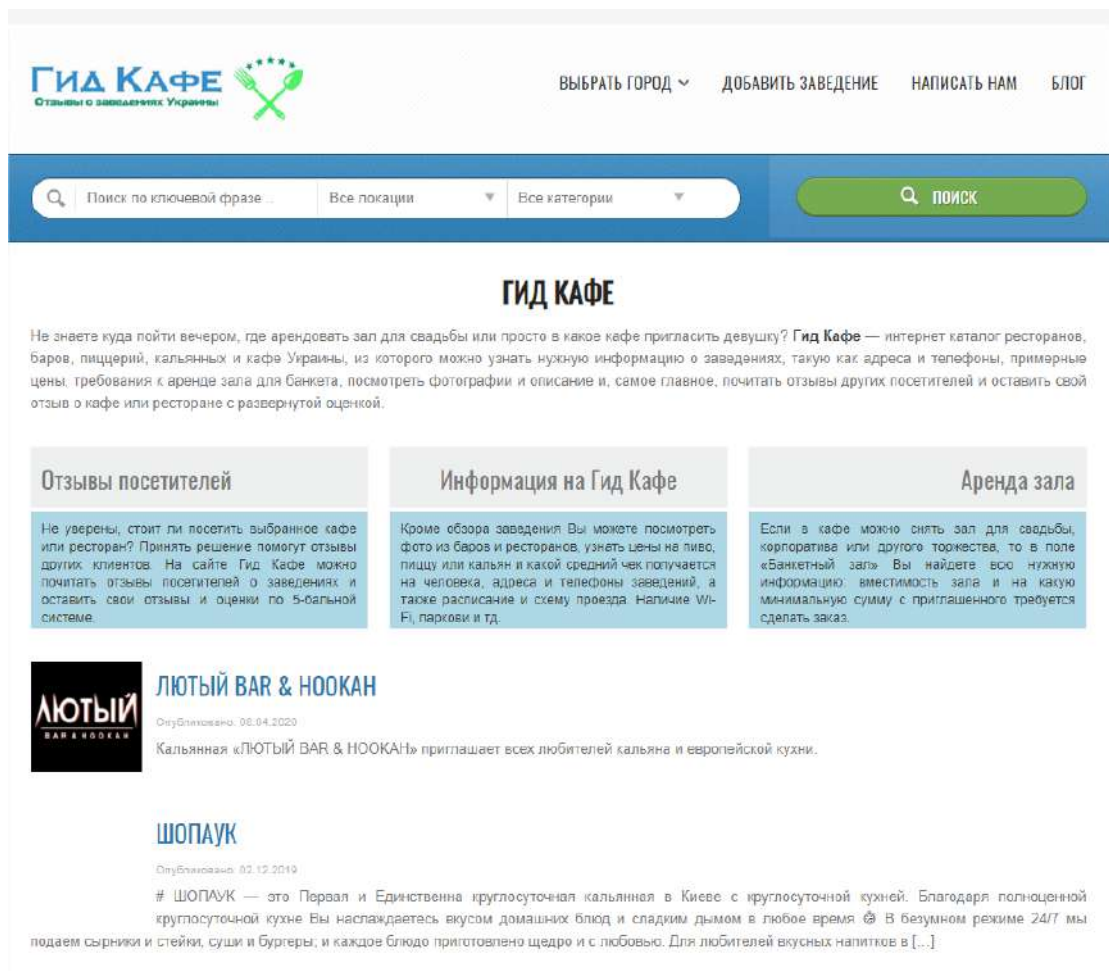


Рисунок 1.2 – Головна сторінка каталогу «Guide-cafe»

Даний продукт значно поступається попередньому. Серед недоліків цього сервісу виділимо такі:

- база закладів включає в себе лише 3 міста;
- відносно бідний інтерфейс;
- доволі малий список наявних фільтрів;
- повна відсутність функціоналу.

1.1.3 HoReCa - Україна

Головну сторінку сервісу «HoReCa - Україна» [9] наведено на рисунку 1.3.

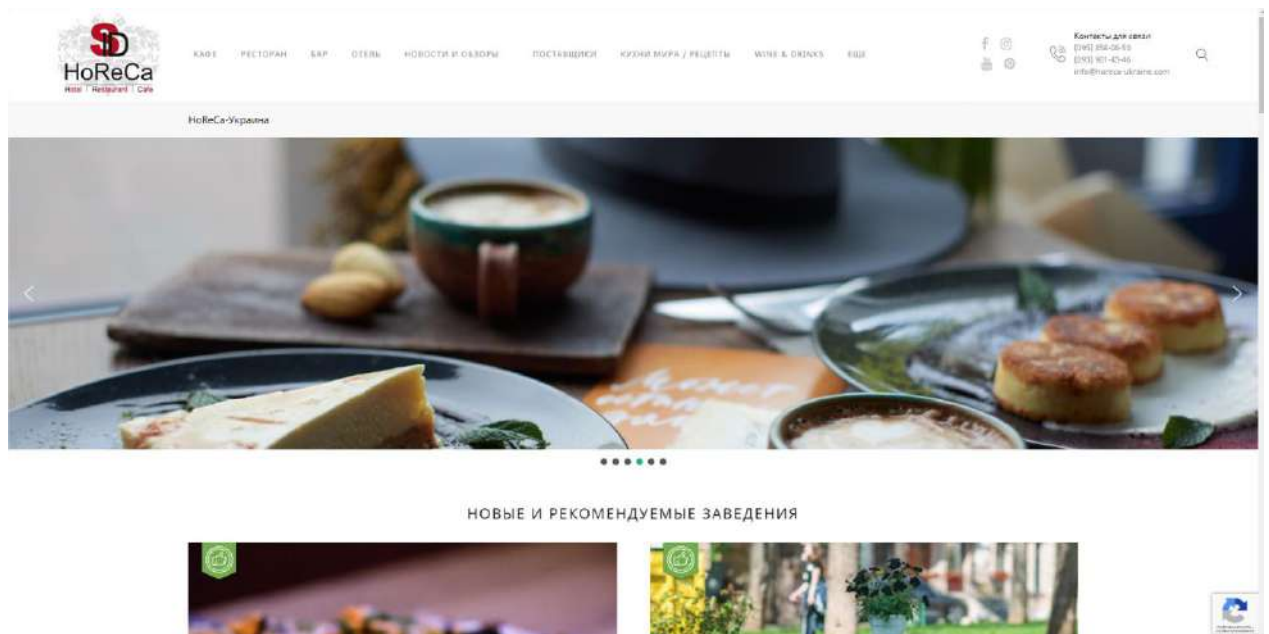


Рисунок 1.3 – Головна сторінка сайту «HoReCa»

Цей сайт виступає більше не як каталог, а скоріше як блог про гастрономію та заклади харчування.

Серед основних переваг цього сайту:

- цікаве інформаційне забезпечення;
- дуже велика база закладів;
- можливість приймати участь в розвитку проекту.

Основними недоліками програмного продукту є такі:

- при наявності взагалі всіх закладів, більшість з них не мають майже ніякої інформації;
- відсутність мапи;
- відсутність фільтрації окрім базових типів закладів

1.1.4 Сайт Booze.com.ua

Головну сторінку сайту Booze.com.ua [10] наведено на рисунку 1.4.

ЗАВЕДЕНИЯ ПО ВСЕЙ УКРАИНЕ У ВАС В ГАДЖЕТЕ

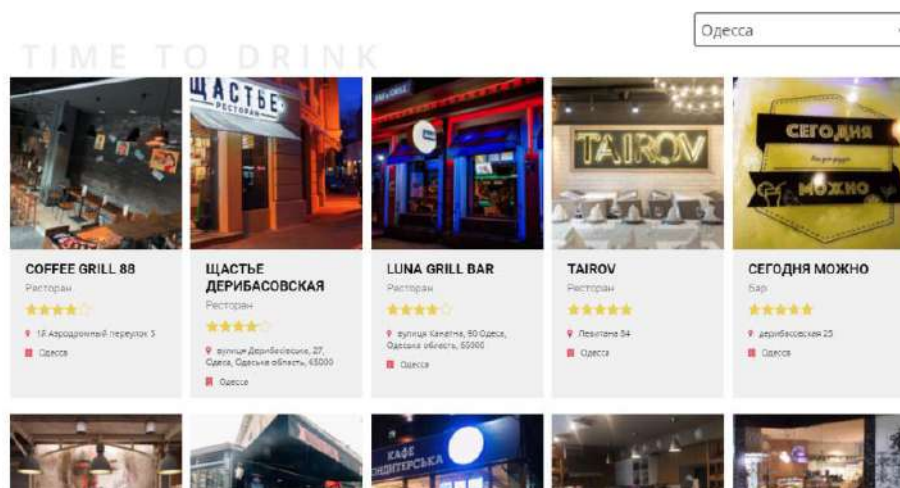


Рисунок 1.4 – Головна сторінка сайту «Booze.com.ua»

Цей сайт представляє себе також, як каталог і фактично їм і є (рис 1.4). Але, на відміну від ідеї цієї роботи, даний сайт фактично є лише списком закладів, без додаткової інформації, можливості фільтрації (окрім локації) та будь-якого функціонального супроводу.

В якості позитивних аспектів цього продукту можна виділити лише синхронізацію з відгуками в «Google», що надає більше початкової інформації користувачу та виступає непідробним джерелом.

Серед недоліків відзначимо такі:

- відсутність будь-якого функціоналу;
- незручний, хоч і сучасний інтерфейс;
- присутність програмних недоліків;
- відсутність можливості для реєстрації і подальших можливостей.

1.2 Порівняльна таблиця існуючих продуктів

В табл.1.1 виконано порівняння переваг та недоліків існуючих програм, що вирішують аналогічні завдання.

Таблиця 1.1 – Порівняння програмних продуктів

№ з/п	Критерій порівняння	Назва програми (сайту)			
		Tomato	Guide-cafe	HoReca	Booze
1	Реєстрація користувача	+	—	+	—
2	Наявність мапи для покращення навігації	—	—	—	—
3	Наявність мобільного додатку	+	—	—	+
4	Обширні можливості фільтрації	+	—	—	—
5	Якісне інформаційне оснащення	+	+	+	—
6	Можливість приймати участь в актуалізації контенту	—	+	+	—

1.3 Постановка завдання роботи

В якості результату виконаної роботи передбачається отримати продукт, задачею якого планується задовільнити усі можливі потреби

користувача, який буде потребувати допомоги в пошуку відповідного закладу.

Система повинна забезпечувати:

- можливість зареєструвати акаунт звичайного користувача;
- можливість створення акаунту власника закладу;
- наявність інтерактивної мапи;
- наявність широкого списку фільтрів;
- список усіх закладів, згідно з «Google» мапами;
- рейтингова система закладів;
- ведення статистики переглядів певних категорій;
- реалізація онлайн-бронювання з відправкою підтвердження на електронну пошту;
- реалізація системи відгуків.

1.4 Висновки до розділу 1

У рамках першого розділу було проаналізовано предметну область. Були виділені основні функції, що потребують реалізації та особливості, що потрібно взяти до уваги при розробці програмного продукту.

Окрім того, були розглянуті подібні програмні продукти, окреслені їх переваги та недоліки, внаслідок чого були сформульовані вимоги до розроблюваного програмного продукту.

Визначено, що відомі сервіси з вибору та замовлення харчування мають певні недоліки, зокрема, вони часто є досить застарілими (мають застарілий і не зовсім зручний інтерфейс), як правило, є орієнтованими під конкретний заклад харчування та не можуть бути модифікованими для роботи з іншими закладами. Це обумовлює, окрім розробки програмного забезпечення, також і створення відповідних інформаційних моделей, що дозволять створювати подібне програмне забезпечення у майбутньому, а також легко модифікувати розроблюване ПЗ.

2 РОЗРОБКА ІНФОРМАЦІЙНОЇ МОДЕЛІ ПРОГРАМНИХ ЗАСОБІВ ДЛЯ ВИБОРУ ХАРЧУВАННЯ ЗА БАЖАНИМИ КРИТЕРІЯМИ

З метою створення програмного забезпечення розроблено інформаційну модель програмних засобів для вибору харчування за бажаними критеріями.

Розроблена інформаційна модель подана у вигляді набору UML-діаграм та дозволяє створювати та модифікувати програмне забезпечення для вибору харчування за бажаними критеріями.

2.1 Розробка абстрактних моделей програмного забезпечення на основі структурних діаграм

На рисунках 2.1 – 2.2 наведено абстрактні моделі системи на основі структурних діаграм:

- діаграма класів програмного забезпечення для вибору харчування за бажаними критеріями (рисунок 2.1);
- діаграма пакетів програмного забезпечення для вибору харчування за бажаними критеріями (рисунок 2.2);

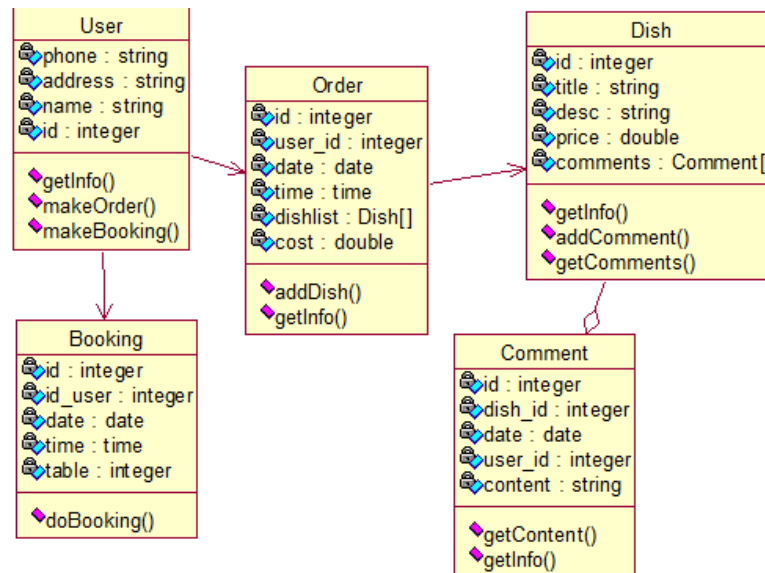


Рисунок 2.1 – Діаграма класів програмного забезпечення для вибору харчування за бажаними критеріями у нотації UML

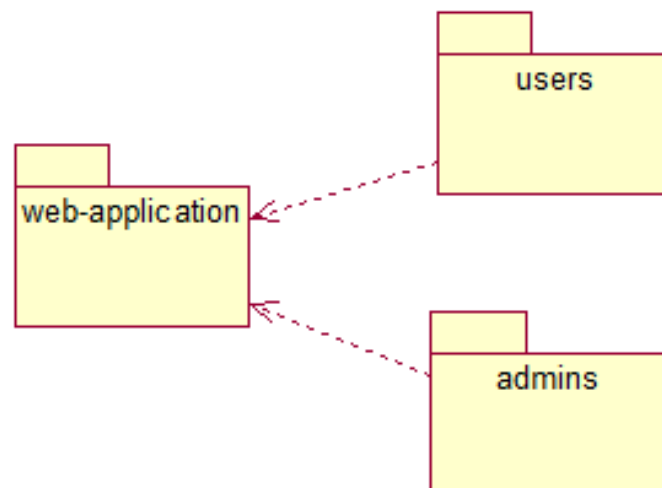


Рисунок 2.2 – Діаграма пакетів програмного забезпечення для вибору харчування за бажаними критеріями у нотації UML

На рисунку 2.3 наведено діаграму компонентів програмного забезпечення для вибору харчування за бажаними критеріями:

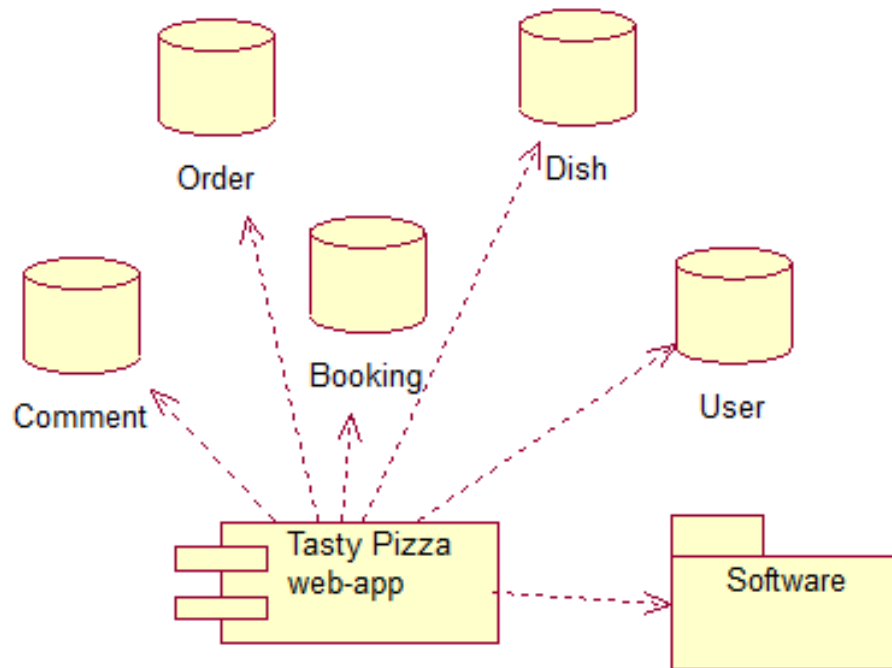


Рисунок 2.3 – Діаграма компонентів програмного забезпечення для вибору харчування за бажаними критеріями у нотації UML

2.2 Розробка абстрактних моделей програмного забезпечення на основі діаграм поведінки

На рисунках 2.4 – 2.5 наведено абстрактні моделі програмного забезпечення на основі діаграм поведінки:

- діаграма прецедентів програмного забезпечення для вибору харчування за бажаними критеріями (рисунок 2.4);
- діаграма станів програмного забезпечення для вибору харчування за бажаними критеріями (рисунок 2.5);

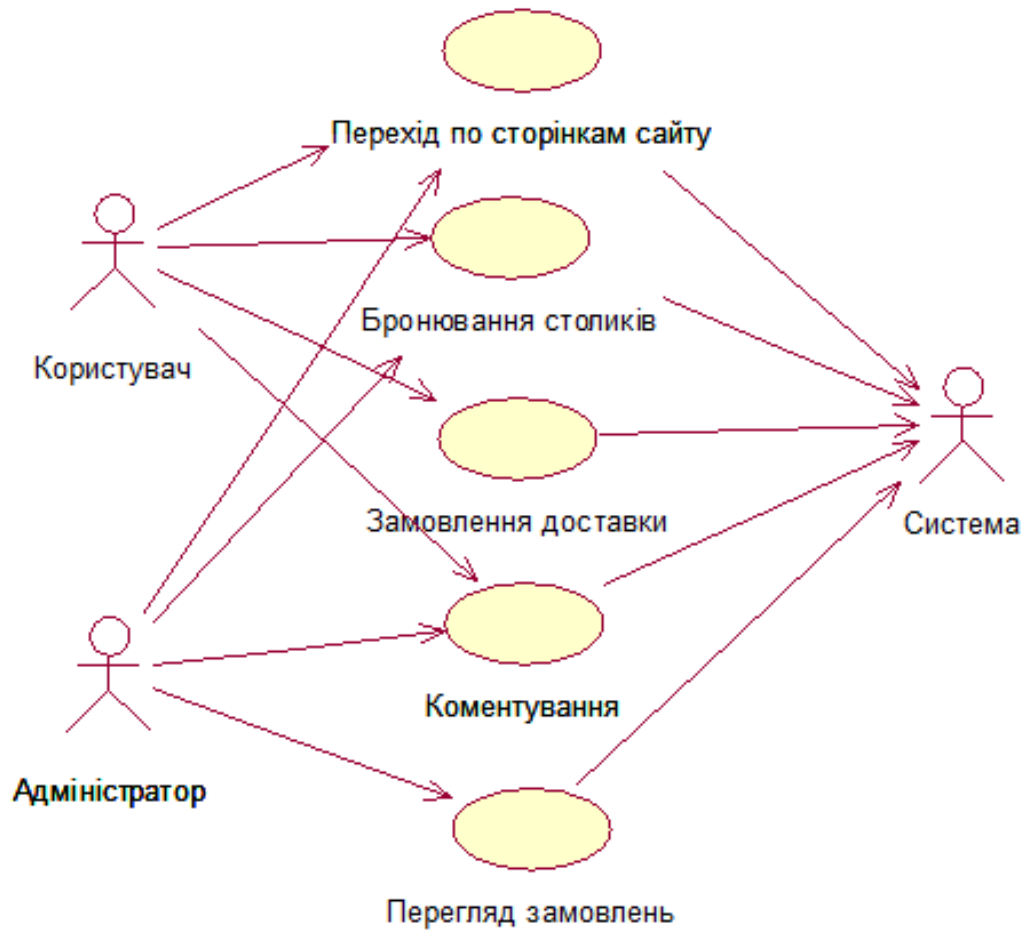


Рисунок 2.4 – Діаграма прецедентів програмного забезпечення для вибору харчування за бажаними критеріями у нотації UML

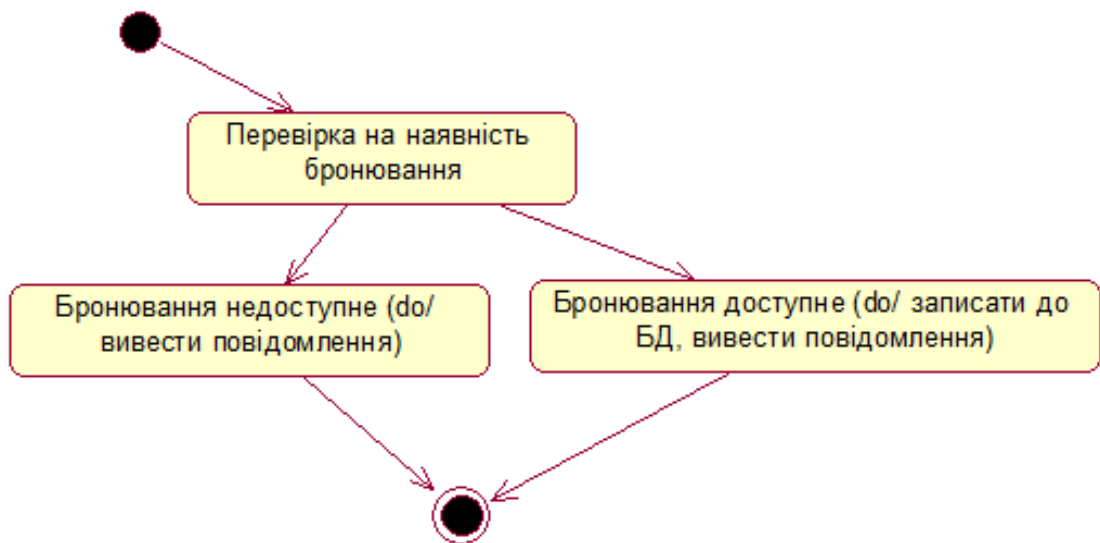


Рисунок 2.5 – Діаграма станів програмного забезпечення для вибору харчування за бажаними критеріями у нотації UML

– На рисунку 2.6 наведено діаграму діяльності програмного забезпечення для вибору харчування за бажаними критеріями (рисунок 2.6).

:

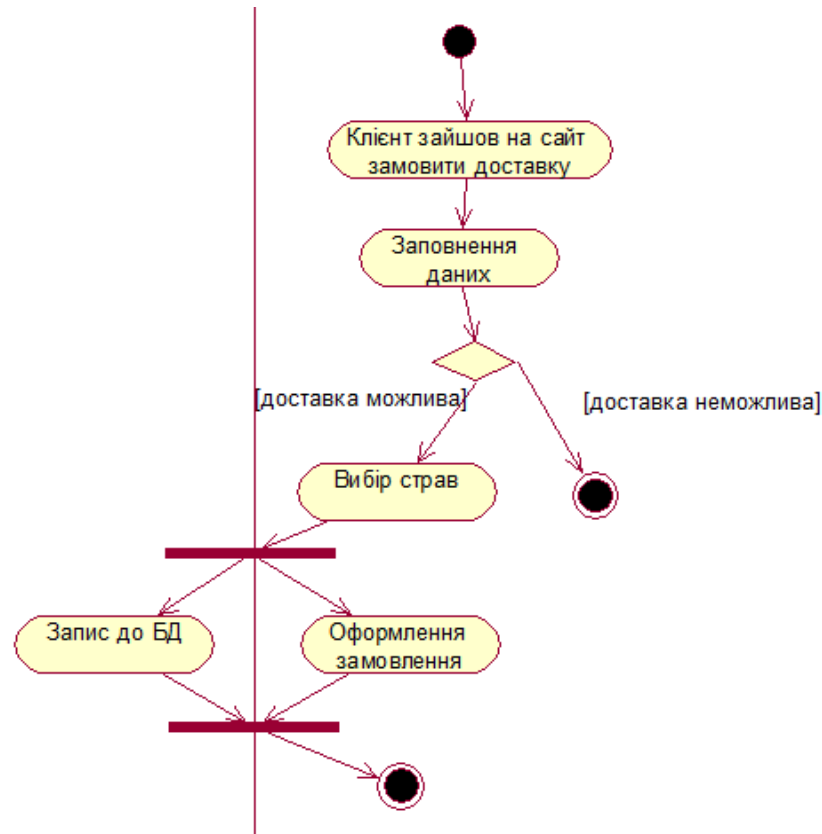


Рисунок 2.6 – Діаграма діяльності програмного забезпечення для вибору харчування за бажаними критеріями у нотації UML

2.3 Розробка абстрактних моделей програмного забезпечення на основі діаграм взаємодії

На рисунках 2.7 – 2.8 наведено абстрактні моделі програмного забезпечення на основі діаграм взаємодії:

– діаграма послідовності програмного забезпечення для вибору харчування за бажаними критеріями (рисунок 2.7);

– діаграма кооперації програмного забезпечення для вибору харчування за бажаними критеріями (рисунок 2.8).

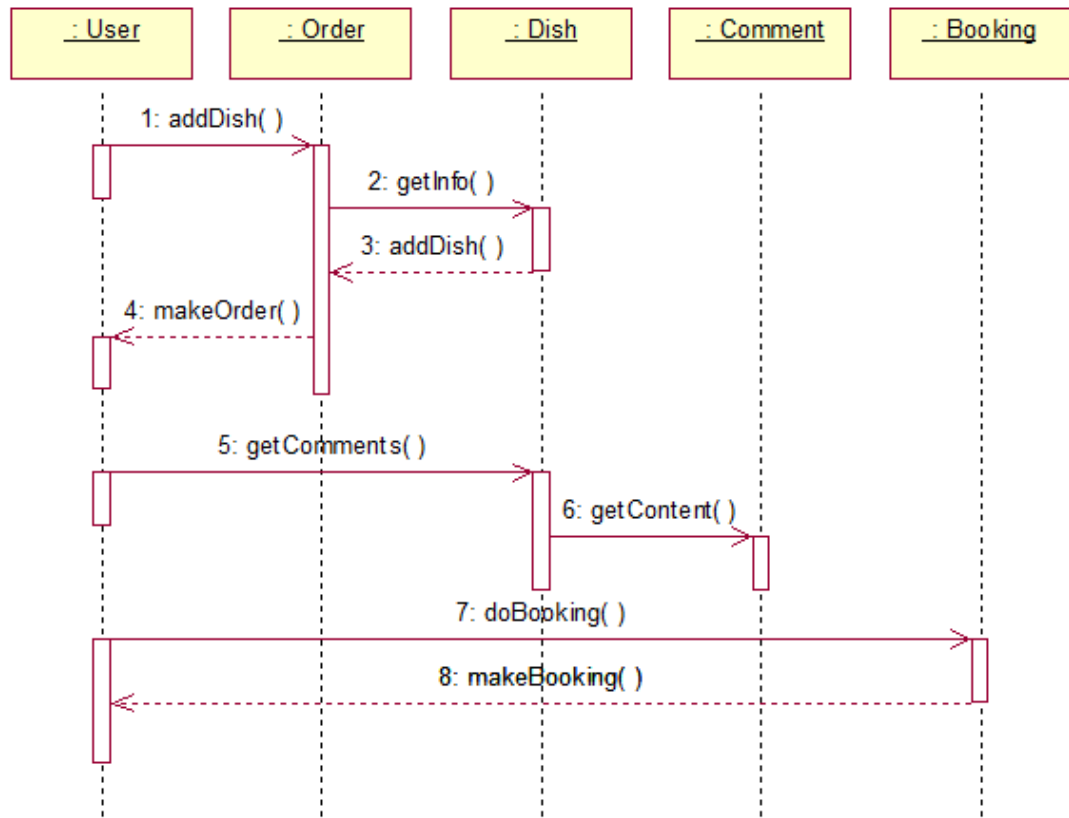


Рисунок 2.7 – Діаграма послідовності програмного забезпечення для вибору харчування за бажаними критеріями у нотації UML

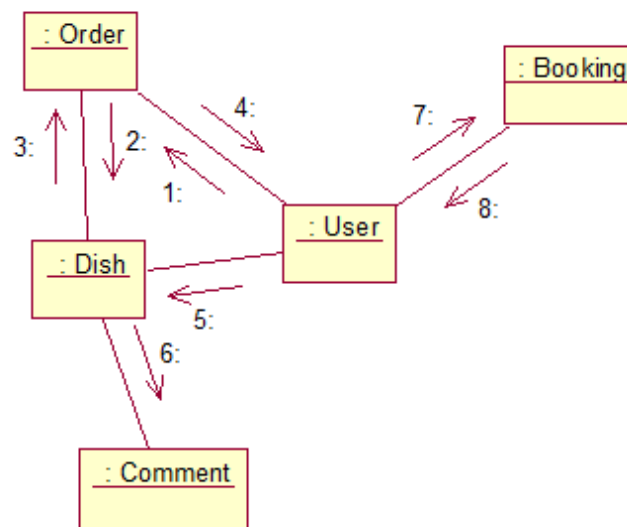


Рисунок 2.8 – Діаграма кооперації програмного забезпечення для вибору харчування за бажаними критеріями у нотації UML

2.4 Висновки за розділом 2

З метою створення програмного забезпечення розроблено інформаційну модель програмних засобів для вибору харчування за бажаними критеріями. Розроблена інформаційна модель подана у вигляді набору UML-діаграм та дозволяє створювати та модифікувати програмне забезпечення для вибору харчування за бажаними критеріями.

3 ОСНОВНІ РІШЕННЯ ЩОДО РЕАЛІЗАЦІЇ КОМПОНЕНТІВ СИСТЕМИ

3.1 Вибір мови програмування та інших засобів розробки

Для розробки програмного забезпечення для вибору харчування за бажаними критеріями необхідно обрати мову програмування [11]-[14]. У цьому проєкті обрано мову PHP (PHP: Hypertext Preprocessor - PHP: гіпертекстовий препроцесор), попередня назва: Personal Home Page Tools - скриптова мова програмування, була Створена для генерації HTML-сторінок на стороні вебсервера. PHP є однією з найпоширеніших мов, що Використовують у сфері веброзробок (разом із Java, .NET, Perl, Python, Ruby). PHP підтримується Переважно більшістю хостинг-провайдерів. PHP - проєкт відкритого програмного забезпечення. PHP інтерпретується вебсервером у HTML-код, Який передається на сторону клієнта. На Відміну Від скриптової мови JavaScript, користувач НЕ бачить PHP-коду, тому що браузер отримує готовий html-код. Це є перевага з точки зору безпеки, но погіршує інтерактивність сторінок. Але ніхто НЕ забороняє використовувати PHP для генерування JavaScript-кодів, Які виконуються Вже на стороні клієнта [15]-[21].

PHP — мова, у код якої можна вбудовувати безпосередньо html-код сторінок, які, у свою чергу, коректно оброблюватимуться PHP-інтерпретатором. Обробник PHP просто починає виконувати код після відкриваючого тегу (<?php) і продовжує виконання до того моменту, поки не зустрине закриваючий тег [15]-[21].

У PHP вбудовані бібліотеки для роботи з MySQL, PostgreSQL, SQLite, mSQL, Oracle, dbm, Hyperware, Informix, InterBase, Sybase [15]-[21].

Завдяки стандарту відкритого інтерфейсу зв'язку з базами даних (англ. Open Database Connectivity Standard, ODBC) можна підключатися до всіх баз даних, до яких існує драйвер [15]-[21].

Порівняльний аналіз різних мов програмування наведено у табл. 3.1.

Таблиця 3.1 – Порівняльний аналіз мов програмування

Критерій порівняння	C++	Java	PHP
Підтримка імперативної парадигми програмування	+	+	+
Зручність для створення вебсайтів	+ -	+	+
Наявність засобів для розробки клієнт-серверного програмного забезпечення	+ -	+	+
Зручність використання для створення webзастосунків	+ -	+	+
Зручність використання для створення клієнт-серверного програмного забезпечення	+ -	+ -	+
Кросплатформність	+	+	+

SQL (Structured query language — мова структурованих запитів) — декларативна мова програмування для взаємодії користувача з базами даних, що застосовується для формування запитів, оновлення і керування реляційними БД, створення схеми бази даних і її модифікація, система контролю за доступом до бази даних [22]-[23]. Сам по собі SQL не є ні системою керування базами даних, ні окремим програмним продуктом [22]-[23].

SQL — це діалогова мова програмування для здійснення запиту і внесення змін до бази даних, а також управління базами даних. Багато баз даних підтримує SQL з розширеннями до стандартної мови. Ядро SQL формує командна мова, яка дозволяє здійснювати пошук, вставку, оновлення, і вилучення даних, використовуючи систему управління і адміністративні функції. SQL також включає CLI (Call Level Interface) для доступу і управління базами даних дистанційно [22]-[23].

Не зважаючи на наявність діалектів і відмінностей в синтаксисі, більшість текстів SQL-запитів, що містять, DDL і DML, можуть бути досить легко перенесені з однієї СУБД в іншу. Існують системи, розробники яких спочатку орієнтувалися на застосування щонайменше кількох СУБД

(наприклад: система електронного документообігу Documentum може працювати як з Oracle, так і з Microsoft SQL Server та IBM DB2). Природно, що при застосуванні деяких специфічних для реалізації можливостей, такого рівня перенесення дуже важко досягти [22]-[23].

Наявність стандартів і наборів тестів для виявлення сумісності та відповідності конкретній реалізації SQL загальноприйнятому стандарту тільки сприяє «стабілізації» мови. Щоправда, слід звернути увагу на той факт, що сам по собі стандарт місцями занадто формалізований і має завеликі розміри, наприклад, Core-частина стандарту SQL:2003 містить понад 1300 сторінок тексту [22]-[23].

За допомогою SQL програміст описує лише дані, які потрібно витягнути або модифікувати. Яким саме чином це зробити, вирішує СУБД безпосередньо при обробці SQL-запиту. Не слід вважати, що це повністю універсальний принцип — програміст описує набір даних для вибірки або модифікації, проте йому корисно уявляти, як СУБД інтерпретуватиме текст його запиту. Такі моменти стають особливо критичними при роботі з великими базами даних та зі складними запитам — чим складніше сконструйований запит, тим більше варіантів виконання він припускає. Ці варіанти можуть дуже відрізнятися за швидкістю виконання та використаними ресурсами, хоча результат (набір даних) має бути однаковим [22]-[23].

Vue.js - це JavaScript бібліотека для створення вебінтерфейсів з використанням шаблону архітектури MVVM (Model-View-ViewModel). Оскільки Vue працює тільки на «рівні уявлення» і не використовується для проміжного програмного забезпечення і бекенда, він може легко інтегруватися з іншими проектами і бібліотеками. Vue.js містить широкую функціональність для рівня уявлень і може використовуватися для створення потужних односторінкових вебдодатків [14]-[23].

Функції Vue.js:

- реактивні інтерфейси;

- декларативний рендеринг;
- зв'язування даних;
- директиви (всі директиви мають префікс «V-». В директиву передається значення стану, а в якості аргументів використовуються html атрибути або Vue JS події);
- логіка шаблонів;
- компоненти;
- обробка подій;
- властивості;
- переходи і анімація CSS;
- фільтри.

Основна бібліотека Vue.js 2 дуже маленька (всього 17 кБ). Це гарантує, що навантаження на ваш проект, реалізований за допомогою Vue.js, мінімальна, а ваш сайт буде швидко завантажуватися [14]-[23].

Vue підходить для невеликих проектів, яким необхідно додати трохи реактивності, уявити форму за допомогою AJAX, відобразити значення при введенні даних користувачем, авторизація або інші аналогічні завдання. Vue легко масштабується і добре підходить для об'ємних проектів, тому його називають прогресивним фреймворком. Vue також відмінно підходить для великих односторінкових додатків завдяки своїм основним компонентам, таким як Router і Vuex. З Vue можна як використовувати загальнодоступні API для створення додатків, так і реалізовувати виконуються сервером додатка. Але Vue найкраще підходить для розробки рішень, які використовують зовнішні API для обробки даних. За допомогою Vue також можна створювати frontend блогу на популярних CMS. Vue.js відмінно підходить і для розробки динамічних інтерфейсів, які адаптуються під користувача [14]-[23].

Laravel це безкоштовний PHP 5.3 фреймворк з відкритим кодом, сам винахідник цього фреймворку описав його як "Фреймворк для Вебмайстрів" ("A Framework For Web Artisans"). Головні його особливості, це простота,

елегантність, та хороша документація. Також він звільняє від масивного і заплутаного коду, допомагає створювати чудові веб додатки, використовуючи простий і виразний синтаксис. Розробка повинна бути творчим і цікавим процесом, а не приносити незручності [14]-[23].

Пакети (bundles) в Laravel це як PEAR для PHP. Вони є доповненнями, які ви можете завантажити та підключити до Laravel. На даний момент існує досить багато загальнодоступних пакетів в репозиторії пакетів Laravel який постійно поповнюється. Laravel дозволяє використовувати командний рядок для швидкого встановлення пакетів [14]-[23].

ORM - це технологія програмування, що розроблена для полегшення програмістам роботи з БД шляхом надання методів API для типових операцій (вибірка, додавання, оновлення, видалення і т.д.). Eloquent ORM є найбільш передовою ActiveRecord реалізацією для PHP. Подібно Doctrine ORM дозволяє зробити будь-яку роботу з базою даних дуже простий. Можна з легкістю виконувати CRUD операції над вашою базою даних. А так же створювати різні зв'язки між таблицями [14]-[23].

Для пошуку за критеріями важливу роль відіграє синтаксичний аналіз, що представляє собою процес зіставлення лінійної послідовності лексем (слів, токенів) природного або формальної мови з його формальною граматиною. Результатом зазвичай є дерево розбору (синтаксичне дерево). Зазвичай застосовується спільно з лексичним аналізом.

Парсинг використовується для:

- мови програмування - розбір вихідного коду мов програмування, в процесі трансляції (компіляції або інтерпретації);
- структуровані дані - дані, мови їх опису, оформлення і т. Д. Наприклад, XML, HTML, CSS, JSON, ini-файли, спеціалізовані конфігураційні файли і т. п.;
- побудова індексу в пошуковій системі;
- SQL-запити (DSL-мову);
- математичні вирази;

- регулярні вирази (які, в свою чергу, можуть використовуватися для автоматизації лексичного аналізу);
- формальні граматики;
- лінгвістика - природні мови. Наприклад, машинний переклад і інші генератори текстів.
- витяг даних вебсторінок - вебскрейпінг, є окремим випадком парсинга.

Типи алгоритмів:

- метод рекурсивного спуску (англ. Top-down parsing) - це один з методів визначення приналежності вхідного рядка до деякого формального мови, описаному LL (k) контекстно-вільною граматиною. Це клас алгоритмів граматичного аналізу, де правила формальної граматики розкриваються, починаючи зі стартового символу, до отримання необхідної послідовності токенів. Синтаксичний LL-аналізатор (LL parser) - в інформатиці спадний синтаксичний аналізатор для деякого підмножини контекстно-вільних грамастик, відомих як LL-граматики. При цьому не всі контекстно-вільні граматики є LL-граматиками;
- LR-аналізатор (англ. LR parser) - синтаксичний аналізатор для вихідних кодів програм, написаних на деякій мові програмування, який читає вхідний потік зліва (Left) направо і виробляє найбільш праву (Right) продукцію контекстно-вільної граматики. Використовується також термін LR (k) -аналізатор, де k висловлює кількість непрочитаних символів предпросмотра у вхідному потоці, на підставі яких приймаються рішення при аналізі. Зазвичай k дорівнює 1 і часто опускається;
- GLR-алгоритм. Працює точно так само, як і LR-алгоритм, за винятком того, що для конкретної граматики GLR-парсер обробляє всі можливі трактування вхідної послідовності, використовуючи пошук в ширину. Генератори GLR-парсерів перетворюють вихідну граматику в таблиці парсеру, точно так же, як і генератори LR-парсерів. Але, тоді як таблиці LR-парсера допускають тільки один перехід стану (певне вихідним станом

парсеру і вхідним термінальним символом), таблиці GLR-парсера допускають безліч результуючих станів. В результаті GLR-алгоритм допускає конфлікти зрушення / згортка і згортка / згортка.

3.2 Вибір інструментарію розробки програмного забезпечення

Для розробки програмного забезпечення для вибору харчування за бажаними критеріями обрано інструментарій PhpStorm. Він являє собою інтелектуальний редактор для PHP, HTML і JavaScript з можливостями аналізу коду на льоту, запобігання помилок у сирцевому коді і автоматизованими засобами рефакторинга для PHP і JavaScript. Автодоповнення коду в PhpStorm підтримує специфікацію PHP 5.3/5.4/5.5/5.6/7.0/7.1 (сучасні і традиційні проекти), включаючи генератори, співпрограми, простори імен, замикання, типажі і синтаксис коротких масивів. Присутній повноцінний SQL-редактор з можливістю редагування отриманих результатів запитів [14]-[23].

PhpStorm включає в себе всю функціональність WebStorm, а також повноцінну підтримку PHP, баз даних і SQL [14]-[23].

PhpStorm розроблений на основі платформи IntelliJ IDEA, написаної на Java. Користувачі можуть розширити функціональність середовища розробки за рахунок установки плагінів, розроблених для платформи IntelliJ, або написавши власні плагіни. PhpStorm глибоко аналізує структуру коду і дійсно розуміє ваш код, підтримуючи всі можливості мови PHP як в нових, так і в legacy-проектах. Редактор підтримує автодоповнення коду і рефакторинг, запобігає помилки на льоту [14]-[23].

IDE інтегрована з системами контролю версій, підтримує віддалене розгортання, бази даних і SQL, інструменти командного рядка, Docker, Composer, REST-клієнт і багато інших інструменти. IDE інтегрована з системами контролю версій, підтримує віддалене розгортання, бази даних і

SQL, інструменти командної рядки, Docker, Composer, REST-клієнт і багато інших інструментів [14]-[23].

PhpStorm подбає про якість вашого коду за допомогою сотень інспекцій, які перевіряють код на льоту і аналізують весь проект цілком. Підтримка PHPDoc, code (re) arranger, інструмент форматування, швидкі виправлення і інші можливості допомагають розробникам писати акуратний код, який легко підтримувати. PhpStorm безпечно перетворює ваш код за допомогою надійних рефакторингов перейменування, переміщення та видалення, вилучення методів, введення змінних, переміщення елементів вгору / вниз, зміни сигнатури і інших. Рефакторингом, що враховують особливості конкретної мови допоможуть застосувати зміни по всьому проекту за пару кліків. При цьому будь-яке перетворення можна скасувати. Графічний відладчик PhpStorm не вимагає додаткової настройки. Він дуже наочно візуалізує, що відбувається у вашому додатку на кожному етапі налагодження. Отладчик працює з Xdebug і Zend Debugger і може використовуватися як локально, так і віддалено. IDE також підтримує модульне тестування з PHPUnit, BDD з Behat і інтегрується з профілювальником [14]-[23].

Одноманітні завдання зручно виконувати прямо в PhpStorm. IDE інтегрована з системами контролю версій, підтримує віддалене розгортання, бази даних і SQL, інструменти командного рядка, Docker, Composer, REST-клієнт і багато інших інструментів [14]-[23].

У PhpStorm ви можете працювати з найсучаснішими технологіями: HTML 5, CSS, Sass, Less, Stylus, CoffeeScript, TypeScript, Emmet і JavaScript. При цьому будуть доступні рефакторинг, налагодження та юніт-тестування. Завдяки функції Live Edit всі зміни можна тут же подивитися в браузері. В PhpStorm ви можете працювати з найсучаснішими технологіями: HTML 5, CSS, Sass, Less, Stylus, CoffeeScript, TypeScript, Emmet і JavaScript. При цьому будуть доступні рефакторинг, налагодження та юніт-тестування.

Завдяки функції Live Edit всі зміни можна тут же подивитися в браузері [14]-[23].

3.3 Архітектура програмного забезпечення для вибору харчування за бажаними критеріями

На рисунку 3.1 зображено архітектуру програмного забезпечення на основі предметної області.

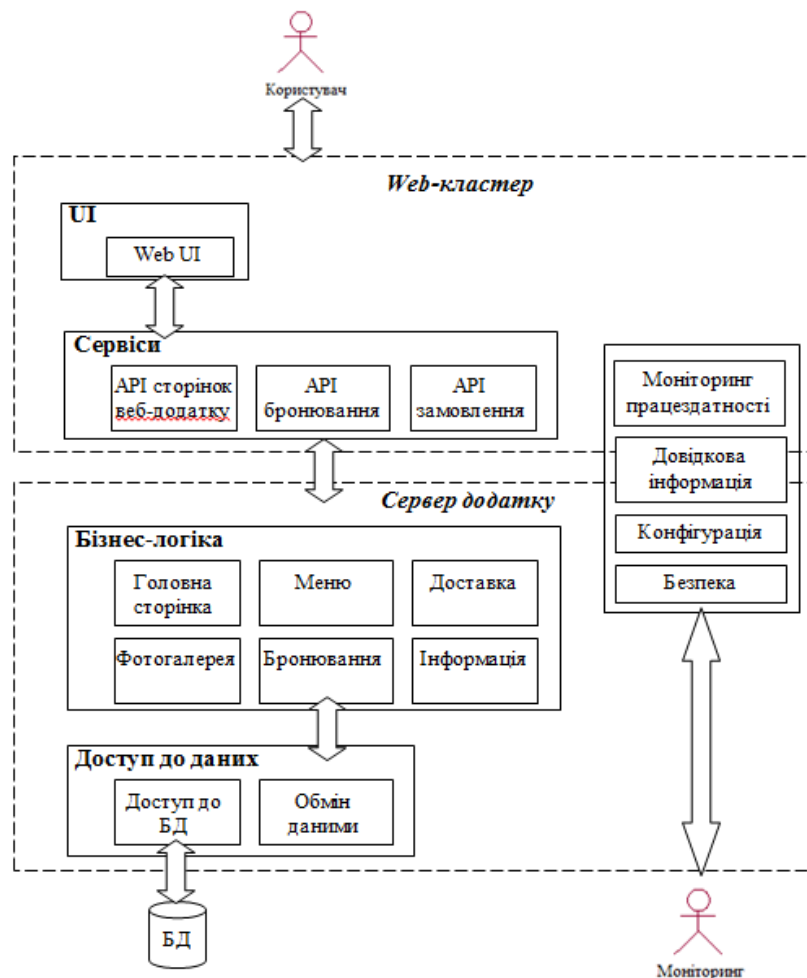


Рисунок 3.1 – Архітектура програмного забезпечення

Як видно, користувач через вебінтерфейс (Web User Interface, WebUI) звертається до сторінок вебсайту та отримує можливість замовлення продуктів харчування за заданими критеріями або для бронювання столику у

певному закладі. Логіка пошуку пропозицій для користувача відповідно його критеріїв реалізована на сервері. На сервері також організовано збереження інформації про пропозиції різних закладів харчування.

3.4 Розробка інтерфейсу програмного забезпечення

На рисунках 3.2 – 3.5 зображено інтерфейс розроблюваного вебзастосунку.

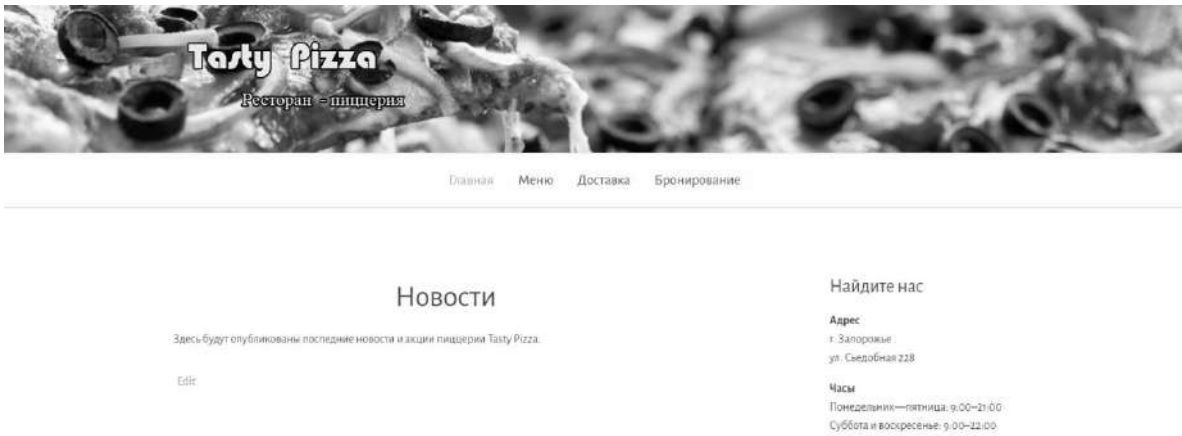


Рисунок 3.2 – Головна сторінка сайту

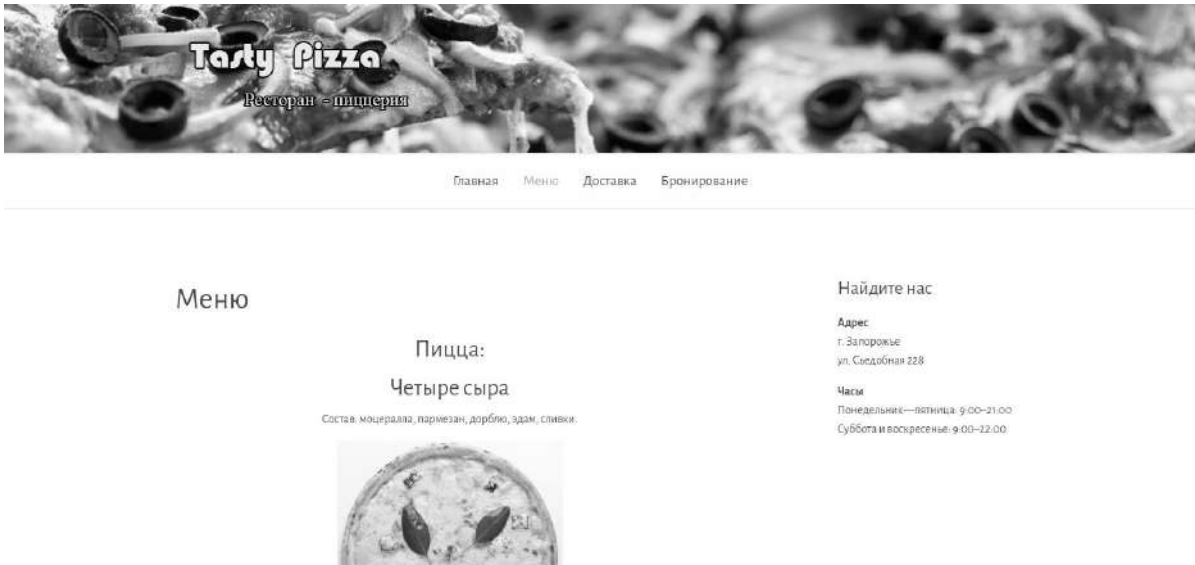


Рисунок 3.3 – Сторінка меню



Доставка

История пиццы столь глубокая, как и история всего нововещества. На сегодняшний день пицца попадает в наши дома быстрее, чем любые другие службы, а начинает свое происхождение с Древнего Египта, около шести тысяч лет тому назад. Впервые европейцами пицца была приготовлена в 1552 году, когда в Италию были завезены первые томаты. С того времени пицца стала активно вживаться во все Меню земного шара, за исключением Африки и стран столь отдаленных от цивилизации. Наша пиццерия со средневековым духом не стал тому исключением. Профессиональные пиццайолы готовы предложить более 20 видов пиццы с различными начинками, и в разнообразных вариациях. Мы доставим вкусную, сочную пиццу с пылу с жару по средствам собственной службы доставки. Бесплатная доставка пиццы в Запорожье осуществляется во всех районах города. Доставка пиццы в Запорожье от ресторана-пиццерии "Tasty Pizza" – это всегда быстро и вкусно! Наши операторы примут у Вас заказ, а курьер доставит Ваш заказ в течении 30-40 минут. Доставка пиццы в Запорожье от нашего ресторана пользуется колоссальным спросом у горожан, мы доставляем все заказанные блюда исключительно в термосумках и надлежащей пластиковой посуде.

- Минимальный заказ : 200 грн.

Найдите нас

Адрес
г. Запорожье
ул. Свободная 228

Часы
Понедельник—пятница: 9:00–21:00
Суббота и воскресенье: 9:00–22:00

Рисунок 3.4 – Сторінка доставки

Бронирование

Забронируйте столик!

Заказать стол

Дата

декабря 5, 2020

Время

Критерии

1

Контактная информация

Имя

Адрес электронной почты

Телефон

[Добавить сообщение](#)

Рисунок 3.5 – Форма бронювання

Завдяки формі бронювання користувач може здійснити бронювання столику, ввівши бажану дату та деякі інші параметри. Як такі параметри можуть бути введені також певні критерії, наприклад, наявність певного виду страв і т.ін. В такому випадку розроблена програма за допомогою внутрішніх засобів синтаксичного аналізу надасть список закладів, де на бажану дату існують вільні місця, та існує набір страв, що буде задовольняти користувача. Варто відзначити, що користувач також може здійснити замовлення з доставкою обраного харчування за бажаними критеріями.

3.5 Тестування розробленого програмного забезпечення для вибору харчування за бажаними критеріями

Результати тестування сумісності з різними веббраузерами наведено у таблиці 3.2.

Таблиця 3.2 – Тестування сумісності

Критерій оцінювання	Google Chrome	Opera	Mozilla Firefox
Сайт правильно відображається	Так	Так	Так
Версія HTML сумісна з версіями браузерів	Так	Так	Так
Картинки коректно відображаються у браузері	Так	Так	Так
Плагіни	Так	Так	Так

Результати тестування підтвердили працездатність розробленого програмного забезпечення для вибору харчування за бажаними критеріями, а також можливість його використання у різних веббраузерах.

Протоколи результатів тестування розробленого програмного забезпечення наведено у табл. 3.3-3.5.

Таблиця 3.3 – Протокол результатів тестування функціоналу програми

Критерій оцінювання	Значення критерію	Відповідність значення критерію технічному завданню
Наявність меню	Так	Так
Можливість забронювати столики	Так	Так
Можливість користування багатьма користувачами одночасно	Так	Так
Інтерфейс системи на заданій мові	Так	Так
Наявність графічного інтерфейсу	Так	Так

Таблиця 3.4 – Протокол результатів тестування ергономічності програми

Критерій оцінювання	Значення критерію	Відповідність значення критерію технічному завданню
Зміст сторінки без орфографічних помилок	Так	Так
Шрифт добре читається	Так	Так
Тексти правильно вирівняні	Так	Так
Наявність підказок	Ні	Так
Стандартний розмір і формат кнопок	Так	Так
Посилання на головну сторінку на кожній сторінці	Так	Так
Усі поля і кнопки доступні з клавіатури	Не всі	Так

Таблиця 3.5 – Протокол результатів тестування безпеки програми

Критерій оцінювання	Значення критерію	Відповідність значення критерію технічному завданню
Сторінки що містять важливі дані відриваються за допомогою HTTPS	Так	Так
Важлива інформація відображається у зашифрованому вигляді	Так	Так

Таким чином, результати тестування підтвердили працездатність програми та її відповідність технічному завданню.

3.6 Висновки за розділом 3

Для розробки програмного забезпечення для вибору харчування за бажаними критеріями обрано мову PHP, оскільки така мова дозволяє створювати вебсайти, що забезпечують ефективну взаємодію з користувачем.

Як інструментарій розробки обрано PhpStorm, який являє собою інтелектуальний редактор програмного коду для різних мов програмування, зокрема PHP, HTML і JavaScript, а також надає можливості аналізу коду, запобігання помилок у сирцевому коді і автоматизовані засоби рефакторингу.

Розроблено архітектуру програмного забезпечення. Визначено, що користувач через вебінтерфейс (Web User Interface, WebUI) звертається до сторінок вебсайту та отримує можливість замовлення продуктів харчування за заданими критеріями або для бронювання столику у певному закладі. Логіка пошуку пропозицій для користувача відповідно його критеріїв реалізована на сервері. На сервері також організовано збереження інформації про пропозиції різних закладів харчування.

Розроблено інтерфейс програмного забезпечення. Визначено, що завдяки формі бронювання користувач може здійснити бронювання столику, ввівши бажану дату та деякі інші параметри. Як такі параметри можуть бути введені також певні критерії, наприклад, наявність певного виду страв і т.ін. В такому випадку розроблена програма за допомогою внутрішніх засобів синтаксичного аналізу надасть список закладів, де на бажану дату існують вільні місця, та існує набір страв, що буде задовольняти користувача. Варто відзначити, що користувач також може здійснити замовлення з доставкою обраного харчування за бажаними критеріями.

Виконано тестування розробленого програмного забезпечення. Результати тестування підтвердили працездатність розробленого

програмного забезпечення для вибору харчування за бажаними критеріями, а також можливість його використання у різних веббраузерах. Виявлено, що функціонал програми, її ергономічність та функції безпеки відповідають технічному завданню.

4 КЕРІВНИЦТВО ПРОГРАМІСТА

4.1 Призначення й умови застосування програми

Програмний продукт призначений для вибору харчування за бажаними критеріями.

Для коректної роботи програму потрібно запускати у веббраузері при наявності операційній системі.

Програма не вимоглива до процесора та оперативної пам'яті, тому запуститься і швидко працюватиме на будь-якому комп'ютері з встановленою операційною системою та наявним веббраузером з використанням мінімуму ресурсів.

Для експлуатації програмного комплексу необхідні такі програмно-технічні засоби:

- ПК під керівництвом операційної системи;
- наявність веббраузеру;
- тактова частота процесора 1 ГГц або більше;
- обсяг оперативної пам'яті не менше 256 Мб;
- обсяг вільного простору на диску - не менше 20 Мб;
- дозвіл екрана монітора не менше 1024x768;
- маніпулятор миша.

4.2 Характеристики програми

Розроблене програмне забезпечення є вебдодатком, розміщеним на сервері, тобто працює цілодобово, поки сервер запущений та знаходиться в працездатному стані. Час роботи програми залежить від конфігурації комп'ютера, на якому вона виконується.

Програма контролює оператора від помилок.

4.3 Звернення до програми

Для виклику додатку слід у будь-якому браузері необхідно перейти за вебадресою, де встановлено програмне забезпечення. Для керування роботою додатку використовуються посилання, випадаючі списки, перемикачі та інші активні елементи.

4.4 Вхідні і вихідні дані

Вхідними даними для програми є інформація про запит користувача.

Вихідними даними для програми є такі: повідомлення про запис користувача у заклад харчування на певну дату та час, або повідомлення про замовлення певного виду харчування.

4.5 Повідомлення

Реалізована програмна система гарантує її стійке функціонування. Програма може виводити наступні повідомлення:

- повідомлення про некоректно введені дані;
- повідомлення про відсутні обов'язкові значення полів;
- повідомлення про помилку під час резервування столику у закладі харчування;
- повідомлення про успішне завершення резервування;
- повідомлення про успішне завершення замовлення.

5 КЕРІВНИЦТВО ОПЕРАТОРА

5.1 Призначення програми

Програмний продукт призначений для вибору харчування за бажаними критеріями.

Для коректної роботи програму потрібно запускати у веббраузері при наявності операційній системі.

Програма не вимоглива до процесора та оперативної пам'яті, тому запуститься і швидко працюватиме на будь-якому комп'ютері з встановленої операційною системою та наявним веббраузером з використанням мінімуму ресурсів.

Інтерфейс користувача забезпечує просту і ефективну взаємодію користувача з програмним продуктом, а також передбачає можливість його адаптації до вимог конкретного користувача.

5.2 Умови виконання програми

Для експлуатації програмного комплексу необхідні такі програмно-технічні засоби:

- ПК під керівництвом операційної системи;
- наявність веббраузеру;
- тактова частота процесора 1 ГГц або більше;
- обсяг оперативної пам'яті не менше 256 Мб;
- обсяг вільного простору на диску - не менше 20 Мб;
- дозвіл екрана монітора не менше 1024x768;
- маніпулятор миша.

Час роботи програми залежить від конфігурації комп'ютера, на якому вона виконується.

5.3 Виконання програми

Для запуску додатку потрібно виконати наступні дії:

- запустити встановлений веббраузер;
- у адресному рядку набрати адресу сайту, після чого з'явиться головна сторінка додатку.

Для замовлення харчування:

- запустити вебдодаток;
- вибрати на панелі фільтрів потрібні критерії харчування;
- переглянути доступні варіанти та натиснути кнопку «Замовити».

Для замовлення столику:

- запустити вебдодаток;
- вибрати необхідну дату та час відвідування закладу харчування;
- завершити резервування.

5.4 Повідомлення оператора

Реалізована програмна система гарантує її стійке функціонування.

Програма може виводити наступні повідомлення:

- повідомлення про некоректно введені дані;
- повідомлення про відсутні обов'язкові значення полів;
- повідомлення про помилку під час резервування столику у закладі харчування;
- повідомлення про успішне завершення резервування;
- повідомлення про успішне завершення замовлення.

6 ЕКОНОМІКО-ОРГАНІЗАЦІЙНА ЧАСТИНА

У дипломній кваліфікаційній роботі магістра проведено дослідження та програмно реалізовано інформаційно-аналітичну систему для вибору закладу харчування за бажаними критеріями.

Очікується, що впровадження в експлуатацію розробленого програмного забезпечення на підприємстві дозволить скоротити 0,25 ставки одного асистента керівника, що займається організацією харчування співробітників, з заробітною платою 9650 гривень на місяць.

У розробці беруть участь програміст із заробітною платою 11250 грн. в місяць і консультант з заробітною платою 9750 грн. у місяць. На розробку проєкту було витрачено 2 місяці. Розробка програми розпочалася першого вересня і повинна бути виконана до кінця жовтня 2020 року.

6.1 Планування розробки програмного продукту

Весь комплекс розробки програмного продукту можна розділити на етапи [24, 25]. Для кожного етапу вказуються трудомісткість, кількість виконавців і тривалість робіт. Інформація з характеристики робіт по розробці програмного забезпечення зведена у таблицю 6.1.

Таблиця 6.1 – Характеристика робіт по розробці програми

Найменування робіт	Трудомісткість		Виконавці	Тривалість, днів
	люди-дні	%		
Аналіз ПЗ	13	19,4	Програміст, Консультант	7 6
Визначення вимог до ПП	13	19,4	Програміст, Консультант	7 6

Продовження таблиці 6.1

Проектування структури програми	12	17,9	Програміст, Консультант	6 6
Розробка схеми функціонування програми	5	7,5	Програміст, Консультант	3 2
Створення програмного коду	18	26,9	Програміст	18
Тестування і налагодження програмного продукту	3	4,5	Програміст, Консультант	2 1
Складання програмної документації	3	4,5	Програміст, Консультант	2 1
Разом	67	100	Програміст, Консультант	45 22

6.2 Визначення витрат на розробку програми

Для визначення витрат на розробку програми складають калькуляцію кошторисної вартості робіт, що включає статті:

- основна заробітна плата;
- додаткова заробітна плата;
- відрахування на єдиний соціальний внесок (ЄСВ);
- матеріали та комплектуючі вироби;
- витрати на спеціальне обладнання (на оплату машинного часу ЕОМ

для написання і налагодження програмних засобів);

– накладні витрати.

Витрати за статтею «Основна заробітна плата» складаються з планового фонду зарплати всіх категорій працівників, зайнятих в розробці програми [25]. Розрахунок зарплати на підставі даних про трудомісткість, представлений в таблиці 6.2.

Таблиця 6.2 – Основна заробітна плата

Посада виконавця	Чисельність, чол	місячний оклад	Кількість місяців роботи	Сума ЗП, грн
Програміст	1	11250	2	22500
Консультант	1	9750	1	9750
Разом	2			32250

Додаткову заробітну плату приймають рівною 10% від основної зарплати. Вона розраховується за формулою (6.1):

$$ЗП_{\text{дод}} = ЗП_{\text{осн}} \cdot 10\%. \quad (6.1)$$

Підставивши величину основної заробітної плати в дану формулу отримаємо: $ЗП_{\text{дод}} = 32250 \cdot 0.1 = 3225$ грн.

Відрахування на єдиний соціальний внесок (ЄСВ) визначають у відсотковому відношенні від сум основної та додаткової зарплати з урахуванням премій і доплат [25]. Відрахування вираховуються за формулою (6.2):

$$ЄСВ = (ЗП_{\text{осн}} + ЗП_{\text{дод}}) \cdot 0,22, \quad (6.2)$$

$$ЄСВ = (32250 + 3225) \cdot 0,22 = 7804 \text{ грн.}$$

Використовується два найменування матеріалів: папір – 150 грн., флеш-носій – 350 грн. Витрати на матеріали розраховують за формулою (6.3):

$$M = \sum_{i=1}^n (\text{Ц}_i \cdot N_i \cdot (1 + K_{\text{т.з}}) - \text{Ц}_{io} \cdot N_{io}), \quad (6.3)$$

де M – витрати на матеріали, покупні напівфабрикати і комплектуючі вироби, грн.;

$K_{\text{т.з}}$ – коефіцієнт, що враховує транспортно-заготівельні витрати;

Ц_i – ціна i -го найменування матеріалу, напівфабрикату і комплектуючого, грн.;

N_i – потреба в i -му матеріалі, напівфабрикаті і комплектуючому;

Ц_{io} – ціна зворотних відходів i -го найменування матеріалу, грн;

N_{io} – кількість зворотних відходів i -го найменування;

n – кількість найменувань матеріалів, напівфабрикатів і комплектуючих.

Отримаємо:

$$\text{Ц}_{io} = 0; N_{io} = 0; K_{\text{т.з}} = 0,05;$$

$$M = (1 + 0,05) \cdot (150 + 350) = 525 \text{ грн.}$$

Таким чином, разом витрати на матеріали та комплектуючі вироби становлять 525 гривень.

У статтю «Витрати на спеціальне обладнання» входять витрати на придбання, транспортування, монтаж і налагодження нестандартного обладнання. Практично, в даному випадку, в цій статті враховуються витрати на оплату машинного часу ЕОМ для написання і налагодження програмних засобів. Для цього необхідно скласти кошторис «витрат на утримання і експлуатацію устаткування», виходячи з якого визначиться вартість одного машино-години роботи ПК, після множення якої на машинний час пішло на

написання і налагодження програми отримаємо витрати на оплату машинного часу. [25]

Амортизаційні відрахування визначають за формулою (6.4):

$$A = БВ \cdot \frac{H_a}{100}, \quad (6.4)$$

де БВ – балансова вартість обчислювальної техніки, грн;

H_a – норма амортизаційних відрахувань на повне оновлення обчислювальної техніки, %. Приймаємо, що техніка повинна працювати 4 роки, тому $H_a = 25\%$.

Балансова вартість обчислювальної техніки становить 18800 грн. Тому амортизаційні відрахування становлять:

$$A = 18800 \cdot 0,25 = 4700 \text{ грн.}$$

Статтю "Експлуатація обладнання" розраховують підсумовуванням витрат на силову електроенергію і допоміжні матеріали. Витрати на електроенергію визначають за формулою (6.5):

$$C_{\text{э}} = N_{\text{н}} \cdot \Phi_{\text{эф}} \cdot K_{\text{зм}} \cdot K_{\text{зв}} \cdot C_{\text{э}}, \quad (6.5)$$

де $N_{\text{н}}$ – номінальна потужність ЕОМ, кВт;

$\Phi_{\text{эф}}$ – річний ефективний фонд часу роботи ЕОМ, машино-год;

$K_{\text{зв}}$ – середній коефіцієнт завантаження за часом;

$K_{\text{зм}}$ – коефіцієнт завантаження по потужності;

$C_{\text{э}}$ – ціна одного кВт · год електроенергії, грн ./ (кВт · год).

Номінальна Потужність ЕОМ – 0,2 кВт. Річний ефективний фонд часу роботи ЕОМ ставити 1800 годин. Середні коефіцієнти завантаження за часом та за потужністю рівні відповідно 0,9 та 0,6. Ціна однієї кіловат-години електроенергії ставить 2,11 грн.

Отримуємо:

$$C_e = 0,2 \cdot 1800 \cdot 0,9 \cdot 0,6 \cdot 2,11 = 410,18 \text{ грн.}$$

Приймаємо, що $C_e = 410$ грн.

Витрати за статтею "Заробітна плата обслуговуючих робітників та відрахування на соціальне страхування в інші фонди" визначають за формулою (6.6):

$$ЗП_{\text{обсл}} = ФЗП_{\Gamma} \cdot (1 + K_{\text{відрах.}}) \cdot \frac{t_{\text{обсл}}}{\Phi_{\text{еф.обсл}}}, \quad (6.6)$$

де $ФЗП_{\Gamma}$ – річний фонд заробітної плати (основної та додаткової) обслуговуючих робітників, грн.;

$K_{\text{відрах}}$ – коефіцієнт, що враховує відрахування на соціальне страхування і в інші фонди;

$t_{\text{обсл}}$ – час протягом року, необхідне технічне обслуговування ЕОМ, годин / рік. Приймаємо $t_{\text{обсл}} = 36$ год.;

$\Phi_{\text{еф.обсл}}$ – річний ефективний фонд часу обслуговуючого персоналу, годин / рік. Приймаємо, що річний ефективний фонд часу обслуговуючого персоналу становить 1800 годин.

Заробітна плата обслуговуючого персоналу = 7250 грн/місяць.

Отримуємо

$$ФЗП_{\Gamma} = 7250 \cdot 12 = 87000 \text{ грн.}$$

$$ЗП_{\text{обсл}} = 87000 \cdot (1 + 0,22) \cdot 36 / 1800 = 2123 \text{ грн.}$$

Суму витрат за статтею "Поточний ремонт обладнання" обчислимо як 3% від балансової вартості обладнання, оскільки неможливо заздалегідь визначити, скільки коштів необхідно витратити на придбання запасних

частин наявного обладнання. Балансова вартість обчислювальної техніки становить 18800 грн. Тому сума витрат за статтею "Поточний ремонт обладнання" становить: $18800 \cdot 0,03 = 564$ грн.

Інших витрат за проектом не передбачено, тому сума витрат за цією статтею дорівнює нуль гривень.

Кошторис витрат на утримання і експлуатацію устаткування наведений в таблиці 6.3.

Таблиця 6.3 – Кошторис витрат на утримання і експлуатацію устаткування

Найменування статей витрат	Сума, грн.
Амортизація обладнання	4700
Заробітна плата основна і додаткова обслуговуючих робітників з відрахуваннями на соціальні заходи	2123
Експлуатація обладнання (крім витрат на поточний ремонт)	410
Поточний ремонт обладнання	564
Інші витрати	0
Разом	7797

Витрати на оплату машинного часу ЕОМ для написання і налагодження програмних засобів визначаються за формулою (6.7):

$$C_{\text{мо}} = P_{\text{екс}} \cdot t_{\text{мо}}, \quad (6.7)$$

де $C_{\text{мо}}$ – витрати на оплату машинного часу, грн;

$P_{\text{екс}}$ – експлуатаційні витрати на одну годину машинного часу цієї цифрової ЕОМ, грн. / машино–год.;
 $t_{\text{мо}}$ – машинний час цифрової ЕОМ для написання і налагодження даного програмного продукту, машино–год.

Розрахуємо:

$$P_{\text{екс}} = 7797 / 1800 = 4,33 \text{ грн. / машино–год.}$$

Для написання і налагодження даного програмного продукту ЕОМ експлуатується протягом 45 днів в одну зміну по 8 годин, що складає в сумі 360 годин. [25] Таким чином отримуємо суму за статтею «Витрати на спеціальне обладнання»:

$$C_{\text{мо}} = 4,33 \cdot 360 = 1559 \text{ грн.}$$

До накладних витрат відносяться витрати на загальне управління і загальногосподарські потреби (заробітна плата апарату управління, канцелярські витрати і т.д.), утримання та експлуатацію будівель. Накладні витрати включаються до вартості розробки програми непрямым шляхом – у відсотках до основної заробітної плати розробників. Для обчислення цієї статті видатків у дипломному проєкті приймемо, що накладні видатки становлять 30% від основної заробітної плати розробників, що складає 32250 грн. Тому сума накладних видатків становить: $32250 \cdot 0,30 = 9675$ грн.

Калькуляція кошторисної вартості робіт з розробки програми наведена в таблиці 6.4.

Таблиця 6.4 – Калькуляція кошторисної вартості робіт з розробки програми

Найменування статей витрат	Сума, грн.
Основна заробітна плата	32250
Додаткова заробітна плата	3225
ЄСВ	7804
Матеріали і комплектуючі	525
Витрати на спеціальне обладнання	1559
Накладні витрати	9675
Разом	55038

6.3 Розрахунок економічної ефективності програмного продукту

Річну економію від впровадження програми, що дозволяє знизити витрати машинного часу ЕОМ для рішення певного завдання в порівнянні із програмою, що застосовувалася раніше, визначають за формулою (6.8):

$$E = (T_{M1} - T_{M2}) \cdot B_{\text{екс.КТЗ}} + E_{\text{н}}^a \cdot \frac{\Phi_{\text{б.КТЗ}} \cdot (T_{M1} - T_{M2})}{\Phi_{\text{еф.КТЗ}}} - \frac{S_{\text{рп}}}{T_{\text{с}}} + E_{\text{зп}}, \quad (6.8)$$

де T_{M1}, T_{M2} – машинний час, необхідний для розв’язання поставлених завдань, відповідно в старому та у новому варіантах, машино–год./рік;

$B_{\text{екс}}$ – експлуатаційні витрати, що доводяться на 1 год машинного часу ЕОМ, грн/ машино–год.;

$E_{\text{н}}^a$ – нормативний коефіцієнт ефективності капітальних вкладень у засоби автоматизації;

$\Phi_{\text{б.КТЗ}}$ – балансова вартість ЕОМ (КТЗ), грн.;

$\Phi_{\text{еф.КТЗ}}$ – річний ефективний фонд часу роботи ЕОМ, машино–год.;

$S_{\text{рп}}$ – сумарні витрати на розробку програми, грн.;

T_c – термін служби впроваджуваної програми до її морального зношування, років. Приймаємо цей термін 5 років;

$E_{\text{зп}}$ – економія на заробітній платі.

До впровадження даного ПЗ дані функції виконувались співробітником, що відповідає за обробку рукописних символів, 1800 годин на рік, а після 1350 годин на рік (економія складає 0,25 ставки одного асистента керівника, що займається організацією харчування співробітників, з заробітною платою 9650 гривень на місяць).

Отримуємо

$$E = (1800 - 1350) \cdot 4,33 + 0,2 \cdot (18800 \cdot (1800 - 1350)) / 1800 - 55038/5 + 9650 \cdot 0,25(1 + 0,22) \cdot 12 = 27200 \text{ грн.}$$

Коефіцієнт ефективності капітальних вкладень E и строк їхньої окупності $T_{\text{ок}}$ розраховуються за формулами (6.9) і (6.10):

$$E\phi = \frac{E}{S_{\text{рп}}}, \quad (6.9)$$

$$T_{\text{ок}} = \frac{S_{\text{рп}}}{E}, \quad (6.10)$$

Отримуємо:

$$E\phi = 27200 / 55038 = 0,49,$$

$$T_{\text{ок}} = 55038 / 27200 = 2,02 \text{ року.}$$

Зведемо отримані техніко–економічні показники в таблицю 6.5.

Таблиця 6.5 – Техніко–економічні показники

Показники	Сума	Одиниця виміру
Витрати на розробку програми	55038	грн.
Річна економія	27200	грн.
Строк окупності програми	2,02	рік

Таким чином, впровадження в експлуатацію розробленого програмного забезпечення на виробничому підприємстві дозволить скоротити 0,25 ставки одного асистента керівника, що займається організацією харчування співробітників, з заробітною платою 9650 гривень на місяць. Економічні розрахунки показали, що строк окупності програми 2,02 року, що значно менше часу її нормативного строку окупності (5 років). Крім того, річна економія при використанні програми становить 27200 грн. Виходячи з виконаних розрахунків, можна зробити висновок, що розробка даного програмного продукту є економічно доцільною.

7 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

7.1 Аналіз потенційних небезпек

У ході виконання магістерської роботи проводиться дослідження та програмна реалізація інформаційно-аналітичної системи для вибору закладу харчування за бажаними критеріями. При роботі на розробників постійно або періодично діють наступні небезпечні і шкідливі чинники:

- невідповідність ергономічних характеристик устаткування нормованим величинам внаслідок нераціональної організації робочого місця;
- негативний вплив випромінювань відео дисплейних терміналів на органи зору людини має основною причиною перенапруження м'язового апарату ока, що пов'язане з недостатньою контрастністю та мерехтінням зображення, відблесками та відбитим світлом від моніторів. Крім того, втома очей настає при частій зміні огляду більш освітленого монітора та менш освітленої клавіатури, що призводить до «астенопії» (відчуття дискомфорту та біль в очах);
- негативний вплив при роботі з відео дисплейними терміналами на функціональний стан людини який обумовлений електростатичними полями, які створюються моніторами з електронно-променевою трубкою, що може призвести до зниження імунітету, захворювань перефронтальної нервової системи або бронхо-легеневих та шкіряних захворювань;
- можливість ураження електрострумом внаслідок небезпечного рівня напруги в електричній мережі, тому у приміщенні можуть виникнути аварійні ситуації: замикання, загоряння проводки, ураження електрострумом, що може призвести до електротравм або летальних наслідків;
- негативний вплив статичної електрики;
- невідповідність нормам параметрів мікроклімату;
- незадовільні умови освітлення робочого місця, в наслідок не рівномірного розподілу освітлювальних приладів;

- небезпеки, пов'язані з підвищеним рівнем шуму, який створюється устаткуванням та працюючими у приміщенні, що призводить до погіршення слуху;
- можливість загорянь внаслідок порушень правил пожежної безпеки;
- неправильні дії персоналу під час НС внаслідок відсутності управління та паніки, що може призвести до травм та летальних наслідків.

7.2 Заходи із забезпечення безпеки

Забезпечення безпеки праці, організації оптимальних умов праці, а також захисту навколишнього середовища досягається шляхом дотримання норм і проведення заходів, передбачених законодавчими актами України [26]-[37].

У НПАОП 0.00-7.15-18 «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями» [26] передбачені гігієнічні вимоги до організації устаткування робочих місць ВДТ ЕОМ і ПЭВМ. Конструкція робочого стола відповідає сучасним вимогам ергономіки й забезпечує оптимальне розміщення на робочій поверхні використовуваного устаткування (дисплея, клавіатури, принтера) і документів. Висота робочої поверхні робочого стола із ВДТ регулюється в межах 680...800 мм, а ширина й глибина - забезпечує можливість виконання операцій у зоні досяжності моторного поля (рекомендовані розміри: 600...1400 мм, глибина - 800..1000 мм). Приймаємо робочу поверхню з розмірами 1400x800. Щоб уникнути відблисків на екрані вікон робоче місце розміщене торцем до вікна не ближче чим на 1 м. Площа дозволяє нам розмістити додаткові меблі для зберігання документів, полки, стелажі, тумби й т.п.

Велике значення надається ергономічним та естетичним вимогам. Цього домоглися за допомогою колірної рішення приміщення й устаткування з урахуванням вимог ДСТУ 8604:2015 «Дизайн і ергономіка.

Робоче місце для виконання робіт у положенні сидячи. Загальні ергономічні вимоги» [27] - у приміщеннях стіни пофарбовані фарбами пастельних тонів.

Під час організації робочого простору необхідно враховувати індивідуальні антропометричні параметри користувача з відповідними допусками на можливі зміни робочих поз та потребу у переміщеннях.

Заходи мінімізації негативного впливу випромінювань від відео дисплейних терміналів на органи зору.

Передбачено такі внутрішньозмінні режими праці та відпочинку при роботі з ЕОМ при 8-годинній денній робочій зміні в залежності від характеру праці:

- для розробників програм із застосуванням ЕОМ слід призначати регламентовану перерву для відпочинку тривалістю 15 хвилин через кожную годину роботи за ВДТ;

- для операторів із застосування ЕОМ слід призначати регламентовані перерви для відпочинку тривалістю 15 хвилин через кожні дві години;

- для операторів комп'ютерного набору слід призначати регламентовані перерви для відпочинку тривалістю 10 хвилин після кожною години роботи за ВДТ.

У всіх випадках, коли виробничі обставини не дозволяють застосувати регламентовані перерви, тривалість безперервної роботи з ВДТ не повинна перевищувати 4 години.

При 12-годинній робочій зміні регламентовані перерви повинні встановлюватися в перші 8 годин роботи аналогічно перервам при 8-годинній робочій зміні, а протягом останніх 4-х годин роботи, незалежно від характеру трудової діяльності, через кожную годину тривалістю 15 хвилин [26].

З метою зменшення негативного впливу монотонності застосовуються чергування операцій усвідомленого тексту і числових даних (зміна змісту роботи).

Для зниження нервово-емоційного напруження, втомлення зорового аналізатору, поліпшення мозкового кровообігу, подолання несприятливих наслідків гіподинамії, запобігання втомі доцільні деякі перерви використовувати для виконання комплексу вправ.

В окремих випадках - при хронічних скаргах працюючих з ВДТ ЕОМ і ПЕОМ на зорове втомлення, незважаючи на дотримання санітарно-гігієнічних вимог до режимів праці і відпочинку, а також застосування засобів локального захисту очей - допускаються індивідуальних підхід до обмеження часу робіт з ВДТ, зміни характеру праці, чергування з іншими видами діяльності, не пов'язаними з ВДТ [26].

Активний відпочинок має полягати у виконанні комплексу гімнастичних вправ, спрямованих на зняття нервового напруження, м'язове розслаблення, відновлення функцій фізіологічних систем, що порушуються протягом трудового процесу, зняття втоми очей, поліпшення мозкового кровообігу і працездатності.

Згідно ПУЕ приміщення виробництва відноситься до категорії «Приміщення без підвищеної небезпеки» [28].

Апаратура, що перебуває під небезпечною для життя напругою, припускає забезпечення електробезпечності. Всі металеві частини заземлені на третій провід спеціальних триконтактних розеток. Струмопровідні кабелі розташовані під дерев'яною підлогою, зверху покритою лінолеумом, або у вигляді схованої проводки в стінах, таким чином, контакт із кабелем виключається. Над кожною розеткою попереджуючий надпис.

Електрична ізоляція забезпечується конструкціями пристроїв і способами їхнього підключення. У процесі експлуатації періодично проводиться перевірка якості ізоляції, шляхом виміру її опору. Відповідно до ПУЕ в електроустановках до 1000В мінімальне значення опору ізоляції силових і освітлювальних електропроводок на ділянці між суміжними запобіжниками або за останніми запобіжниками між будь-яким проведенням і землею становить не менше 0,5МОм

Відповідно до ПУЕ захисне заземлення не перевищує 4 Ом у всіх електроустановках змінного струму напругою 220 В і вище й постійного струму 440В і вище. У приміщеннях з підвищеною небезпекою заземлення використовується в електроустановках змінного струму напругою 42 В та постійного струму 110 В. Захисному заземленню підлягають корпуси трансформаторів, оболонки кабелів, металеві огороження електроустановок [29].

Вирівнювання потенціалів досягаються шляхом устрою контурних заземлювачів. У приміщеннях вирівнювання потенціалів здійснюється з'єднанням металевих стоек ЕОМ, корпусів апаратів світильників і іншого устаткування з усіма доступними для дотику металевими конструкціями будинку, спорудження, а також з контуром шиною захисного заземлення.

Електричний поділ мережі припускає розділ мережі на окремі електрично непов'язані між собою ділянки за допомогою розділових трансформаторів. Підвищення електробезпеки відбувається за рахунок зменшення ємкості мережі й підвищення опору ізоляції фаз мережі щодо землі.

Подвійна ізоляція забезпечує застосування крім основної, робочої ізоляції струмоведучих частин ще одного шару додаткової ізоляції, що ізолює людину від металевих неструмоведучих частин, які можуть випадково виявитися під напругою. При застосуванні електроустаткування з подвійною ізоляцією не потрібно ні заземлення, ні занулення їхніх корпусів [28].

Накопичена статична електрика, зокрема, на екрані монітора притягує пил, бруд та інші частини присутні в повітрі. Причому електризується не тільки екран, а і повітря на робочому місці, а також одяг користувача, особливо якщо він з синтетичного та шерстяного матеріалу.

В якості заходів захисту запропоновано:

- кілька разів на протязі робочого дня мити руки і обличчя водою, а після закінчення роботи вимити руки й лице з милом;

- щоденно протирати екран монітора, клавіатуру, пристрій «миша», а якщо є приєкранний фільтр, то і його антистатичною серветкою;
- щоденно в приміщенні з ПК проводити вологе прибирання;
- установити нейтралізатори статичної електрики;
- підтримувати у приміщенні вологість повітря зазначену в нормативних документах;
- користувачу ПК бажано носити одяг з природних (льняних) волокон, в т.ч. і шкарпетки [29].

7.3 Заходи з виробничої санітарії та гігієни праці

Оптимальні норми температури повітря в робочій зоні виробничих приміщень у холодний і перехідний періоди року при II категорії робіт згідно ДСН 3.3.6.042-99 Санітарні норми мікроклімату виробничих приміщень [30] – 22...24°C, відносна вологість – 60...40%, швидкість руху повітря – 0.2 м/сек. У теплий період року температура – 23...25°C, відносна вологість – 60...40%, швидкість руху повітря – 0.2 м/сек.

Для забезпечення вищевказаних параметрів передбачене застосування вентиляції - природної (аерації й провітрювання), механічної (загальобмінної, місцевої приточної і витяжної), кондиціонування повітря.

У приміщенні забезпечений приплив свіжого повітря, кількість якого визначається техніко-економічним розрахунком і вибором схеми системи вентиляції. Мінімальна витрата повітря визначається з розрахунку 50...60 м³/год на одну людину, але не менш дворазового повітрообміну в годину.

При невеликому забрудненні зовнішнього повітря кондиціонування приміщень здійснюється із перемінними витратами зовнішнього повітря й рециркуляційного.

Всі норми по освітленості беруться згідно ДБН В.2.5-28:2018. Природне та штучне освітлення [31]. Штучне освітлення в приміщеннях

здійснюється системою загального рівномірного освітлення. При цьому штучне освітлення застосовують не тільки в темний, але й у світлий час доби.

Найбільш прийнятними для приміщень є люмінесцентні лампи ЛБ (білого світла) потужністю 20, 40 або 80 Вт, які встановлюються в світильники типу ЛСП. Допускається застосування ламп накаливання у світильниках місцевого освітлення.

Загальне освітлення варто виконувати у виді суцільних чи переривчастих ліній світильників, розташованих збоку від робочих місць.

Розрахуємо систему загального рівномірного штучного освітлення в обчислювальному центрі. Приймаємо розміри приміщення: довжина $A = 12$ м, ширина $B = 9$ м, висота $H = 5$ м. Висота робочої поверхні складає $h = 0,7$ м.

Коефіцієнти відбиття поверхонь приміщення: колір стелі, стін - сірий, незначне пиловиділення, $\rho_c = 50\%$; $\rho_{ст} = 30\%$; $\rho_{п} = 10\%$,

Згідно з ДБН В.2.5-28:2018 «Природне та штучне освітлення» рівень нормованого загального освітлення в приміщенні нормуються як $E_n = 400$ лк.

Обираємо світильник типу ЛПП, призначений для приміщень висотою до 6 м. $IP = 65$ – ступінь захисту світильника, $[L/h] = 1,4$ – числове значення коефіцієнта світильника.

Коефіцієнт запасу k_z враховує зниження рівня освітленості з часом в результаті забруднення та старіння ламп, світильників і поверхонь приміщення, приймається в залежності від виробничих умов. Приймаємо $k_z = 1,6$.

Коефіцієнт нерівномірності (мінімального) освітлення z (відношення середньої освітленості до мінімальної освітленості), для люмінесцентних ламп низького тиску, як правило дорівнює $z = 1,1$.

Розрахуємо кількість рядів світильників у приміщенні за формулою (7.1):

$$N_p = \frac{B}{(H-h_p) \cdot [L/h]}, \quad (7.1)$$

$$N_p = \frac{9}{(5-0,7) \cdot 1,4} = 1,5 \approx 2 \text{ ряди світильників}$$

Визначимо максимально припустиму відстань між рядами світильників за формулою (7.2):

$$L_{\max} = \frac{B}{N_p}, \quad (7.2)$$

де B – ширина приміщення, N_p – кількість рядів.

Отримаємо:

$$L_{\max} = \frac{9}{2} = 4,5 \text{ м.}$$

Розраховуємо висоту підвісу світильника над робочою поверхнею за формулою (7.3):

$$h = \frac{L_{\max}}{[L/h]}. \quad (7.3)$$

Розрахуємо:

$$h = \frac{4,5}{1,4} = 3,2 \text{ м.}$$

Знайдемо висоту звисання світильника від стелі за формулою (7.4):

$$h_3 = H - h_p - h, \quad (7.4)$$

де H – висота приміщення, h_p – висота робочої поверхні, h – висота підвісу світильника над робочою поверхнею.

Отримаємо:

$$h_3 = 5 - 0,7 - 3,2 = 1,1 \text{ м.}$$

Чисельне значення індексу приміщення визначаємо за рівнянням (7.5):

$$i = \frac{AB}{h(A+B)}, \quad (7.5)$$

де A – довжина приміщення, м;

B – ширина приміщення, м;

h – висота розміщення світильників над робочою поверхнею, м.

Отримаємо

$$i = \frac{12 \cdot 9}{3,2 \cdot (12 + 9)} = 1,6.$$

Значення коефіцієнта використання світлового потоку η вибирається в залежності від виду джерела світла, типу обраного світильника, коефіцієнтів відбиття поверхонь приміщення (ρ_c , $\rho_{ст}$, $\rho_{п}$) та індексу приміщення. В нашому випадку $\eta = 52 \%$.

Визначаємо сумарний світловий потік освітлювальної установки у даному виробничому приміщенні за формулою (7.6):

$$\Phi_{\Sigma} = \frac{E_n \cdot S \cdot k_3 \cdot Z}{\eta}, \quad (7.6)$$

де Φ_{Σ} – розрахункове значення сумарного світлового потоку у приміщенні, лм.

Розрахуємо

$$\Phi_{\Sigma} = \frac{400 \cdot 12 \cdot 9 \cdot 1,6 \cdot 1,1}{0,52} = 146215,4 \text{ лм.}$$

Визначаємо умовну загальну кількість світильників у приміщенні на основі формули (7.7):

$$N^* = AB/L_{\max}^2. \quad (7.7)$$

Розрахуємо

$$N^* = \frac{12 \cdot 9}{4,5 \cdot 4,5} = 5,33 = 6 \text{ шт.}$$

Розрахуємо світловий потік умовного джерела світла за формулою (7.8):

$$\Phi_{\text{л}}^* = \frac{\Phi_{\Sigma}}{N_{\text{л}}}, \text{ лм,} \quad (7.8)$$

де Φ_{Σ} – розрахункове значення сумарного світлового потоку у приміщенні, лм;

$N_{\text{л}}$ – загальна кількість ламп у приміщенні, шт.

Розрахуємо за допомогою формули (7.9):

$$N_{\text{л}} = N^* \cdot n, \quad (7.9)$$

Де n – кількість ламп у світильнику (в ЛПП світильнику знаходиться 2 лампи).

Отримуємо

$$N_{\text{л}} = 6 \cdot 2 = 12 \text{ шт.}$$

Тоді:

$$\Phi_{\text{л}} = \frac{146215,4}{12} = 12185 \text{ лм.}$$

Обираємо тип стандартної лампи з найближчим значенням фактичного світлового потоку лампи $\Phi_{\text{л}}$ і знайти коефіцієнт m (співвідношення між розрахунковим світловим потоком лампи $\Phi_{\text{л}}^*$ та фактичним світловим потоком вибраної стандартної лампи $\Phi_{\text{л}}$).

Обираємо лампу ЛБ із $P_{\text{л}}=40$ Вт, $\Phi_{\text{л}} = 5400$ лм, використаємо формул (7.10):

$$m = \frac{\Phi_{\text{л}}^*}{\Phi_{\text{л}}}. \quad (7.10)$$

Розрахуємо:

$$m = \frac{12185}{5400} = 2,26.$$

Визначаємо оптимальну (фактичну) кількість світильників у приміщенні за формулою (7.11):

$$N = N_{\text{л}} \cdot m. \quad (7.11)$$

Розрахуємо:

$$N = 6 \cdot 2,26 \approx 14 \text{ шт.}$$

Фактична кількість ламп у приміщенні на основі формули (7.12):

$$N_{\text{фл}} = N_{\text{ф}} \cdot n. \quad (7.12)$$

Розрахуємо:

$$N_{\text{фл}} = 14 \cdot 2 = 28 \text{ шт.}$$

Визначаємо загальну розрахункову освітленість E_p у приміщенні, що створюється при застосуванні стандартних ламп за формулою (7.13):

$$E_p = \frac{\Phi_{\text{л}} N_{\text{фл}} \eta}{S k_{\text{з}} z} \text{ лк.} \quad (7.13)$$

Розрахуємо:

$$E_p = \frac{5400 \cdot 28 \cdot 0,52}{12 \cdot 9 \cdot 1,6 \cdot 1,1} = 414 \text{ лк.}$$

Тобто виконується умова $E_p = (-10\% \dots +20\%) \cdot E_{\text{н}}$, лк

Таким чином, на основі розрахунків для освітлення приміщення площею 108 м² встановлено 14 світильників, розташованих у 2 рядах по 7 світильників у кожному ряді, що дозволяє отримати освітленість на робочому місці 414 лк відповідно до ДБН В.2.5-28:2018 «Природне та штучне освітлення» [32].

– Згідно з ДСН 3.3.6.037-99 «Санітарні норми виробничого шуму, ультразвуку та інфразвуку» [33] зниження шуму, створюваного на робочих місцях внутрішніми джерелами, а також шуму проникаючого ззовні, є дуже важливим завданням. Зниження шуму в джерелі виникнення забезпечується

застосуванням пружних прокладок між основою машини, приладу й опорною поверхнею. Як прокладки використовуються гума, повсть, пробка, різної конструкції амортизатори. Під настільні шумливі апарати можна підкладати м'які килимки із синтетичних матеріалів, а під ніжки столів, на яких вони встановлені, - прокладки з м'якої гуми, повсті, товщиною 6 - 8 мм.

7.4 Заходи безпеки у надзвичайних ситуаціях

7.4.1 Заходи з пожежної безпеки

Заходи з протипожежної безпеки розроблені для виробничого приміщення обладнаного ПК з ВДТ розроблені згідно вимог НАПБ А.01.001-2014 «Правила пожежної безпеки в Україні».

Виходячи з аналізу речовин та матеріалів, які використовуються при роботі у приміщенні, обладнаному ПК з ВДТ:

- згідно ДБН В.1.1-7:2016 «Пожежна безпека об'єктів будівництва. Загальні вимоги» [34] у приміщенні, обладнаному ПК з ВДТ можлива пожежа класу – А та Е;

- відповідно до вимог ДСТУ Б В.1.1-36:2016 «Визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою» [35], воно належить до категорії «Д» з пожежної небезпеки – простір у приміщенні, у якому перебувають тверді горючі речовини та матеріали.

Оскільки приміщення належить до категорії «Д» з пожежної небезпеки, тому відповідно до вимог ДБН В.1.1-7:2016 «Пожежна безпека об'єктів будівництва. Загальні вимоги» [34] воно має II ступінь вогнестійкості.

У разі виникнення пожежі у приміщенні обладнаному ПК з ВДТ для евакуації персоналу відповідно до вимог ДБН В.1.1-7:2016 передбачені виходи, по обидві сторони приміщення, з одного боку вікно (на пожежні

сходи), а з іншого – вхідні двері. Відстань від найбільш віддаленого робочого місця до найближчого евакуаційного виходу не обмежується.

Обладнання, силові та освітленні мережі приміщення, обладнаного ПК з ВДТ відповідають вимогам пожежної безпеки, оскільки виконані відповідно до вимог НПАОП 40.1-1.32-01 «Правила будови електроустановок. Електрообладнання спеціальних установок» [36], та мають ступінь захисту ізоляції обладнання IP44 яка відповідає класу пожежанебезпечної зони П-Па до якої належить приміщення.

Для виявлення початкової стадії загоряння, швидкого й точного оповіщення служби пожежної охорони про час і місце виникнення пожежі встановлені системи автоматичної пожежної сигналізації (АПС). Вони самостійно пускають в хід установки пожежогасіння, коли пожежа не досягла великих розмірів. Системи АПС складаються з пожежних сповіслювачів, ліній зв'язку й прийомних пультав (станцій). Згідно вимог ДБН В.2.5-56:2014 «Системи протипожежного захисту» [36], в приміщенні обладнаному ПК з ВДТ встановлена система пожежної й охоронної сигналізації «Фотон-ОФІС», яка забезпечує виявлення теплових і димових ознак пожежі і місця виникнення пожежі з точністю до місця розміщення датчика.

Оскільки приміщення має площу 108 м², передбачені вуглекислотні вогнегасники типу ВВК-3,5 у кількості 6 штук (з розрахунку один вогнегасник с величиною заряду вогнегасної речовини 3,5 кг і більше, на 20 м² площі приміщення). Відстань між вогнегасниками та місцями можливих загорянь не перевищує 10 м.

7.4.2 Заходи з цивільного захисту

Оцінка надійності системи матеріально-технічного постачання і виробничих зв'язків.

У своїй діяльності підприємство використовує різноманітні матеріально-технічні ресурси (сировину, матеріали, паливо, енергію, комплектуючі вироби тощо). Вони в процесі виробництва перетворюються на продукцію (послуги) і підлягають постійному поповненню. Для цього організується матеріально-технічне забезпечення, яке включає: визначення потреби в матеріально-технічних ресурсах, пошук і купівлю ресурсів, організацію доставки, зберігання й видачі окремим споживачам на підприємстві. Постачання матеріально-технічних ресурсів має бути своєчасним, комплексним і з мінімальними витратами. Виконує цю роботу відділ матеріально-технічного постачання.

Матеріально-технічне забезпечення — це система принципів, форм, методів, важелів та структур, які спрямовані на виробництво і постачання технічних засобів, запасних частин, паливно-мастильних матеріалів, інших енергоносіїв, пестицидів, техніко-технологічне обслуговування та надання виробничих послуг.

Матеріально-технічні ресурси являють собою сукупність матеріально-речових засобів і предметів праці, що постачаються промисловістю, за допомогою яких виробляється продукція або надаються відповідні послуги.

Система матеріально-технічного забезпечення — це сукупність підприємств-виробників матеріально-технічних ресурсів, торгово-посередницьких організацій, сервісних підрозділів, формувань з машиновикористання та державних органів, що регулюють відносини у сфері техніко-технологічного забезпечення. Діяльність цієї системи має бути спрямована на відтворення і розвиток матеріально-технічної бази, комплексну механізацію і автоматизацію технологічних процесів, відновлення і збереження родючості ґрунту в сільському господарстві на основі вітчизняного машинобудування та виробництва матеріальних ресурсів.

Виробничими структурами у системі матеріально-технічного забезпечення є заводи сільськогосподарського, продовольчого і торгового

машинобудування, хімічні підприємства, торговельно-посередницькі організації, підприємства з ремонту і технічного обслуговування машин і обладнання, машинно-технологічні формування з надання виробничих послуг, центри випробування техніки, конструкторські та науково-дослідні установи. У межах свого регіону і сфери діяльності вони вивчають попит, купівельну спроможність споживачів, умови реалізації товарів, терміни поставок і на основі зібраної інформації налагоджують виробництво та реалізацію продукції.

Виробництво та реалізація матеріально-технічних засобів, технічне обслуговування і надання виробничих послуг набувають характеру цілісного технологічного процесу. Управління та координацію цим процесом здійснюють органи, діяльність яких визначається особливостями функціонування агропромислового комплексу, загальнодержавними і регіональними програмами соціально-економічного розвитку, що розроблені Кабінетом Міністрів України.

На ці органи покладаються такі функції:

- формування і проведення єдиної технічної політики виробництва та постачання матеріальних ресурсів, сервісного забезпечення підприємств; закупівля критичних видів ресурсів;
- координація діяльності державних і приватних структур для поліпшення матеріально-технічного забезпечення;
- державна підтримка пріоритетних напрямів технічного і технологічного переоснащення сільськогосподарського, продовольчого та торгового машинобудування і виробництва продукції, придбання ліцензій та освоєння наукоємних технологій;
- реалізація державних і регіональних програм розвитку сільськогосподарського, продовольчого і торгового машинобудування, технічного сервісу і виробничого обслуговування;
- координація замовлень на виконання та освоєння науково-дослідних, дослідно-конструкторських і проектно-технологічних розробок

для підвищення ефективності матеріально-технічного та сервісного забезпечення;

- формування єдиного інформаційного простору та координація маркетингу в системі матеріально-технічного забезпечення;

- проведення протекціоністської політики щодо вітчизняного товаровиробника матеріально-технічних ресурсів [37].

ВИСНОВКИ

У ході виконання дипломної кваліфікаційної роботи магістра було проаналізовано предметну область. Визначено, що відомі сервіси з вибору та замовлення харчування мають певні недоліки, зокрема, вони часто є досить застарілими (мають застарілий і не зовсім зручний інтерфейс), як правило, є орієнтованими під конкретний заклад харчування та не можуть бути модифікованими для роботи з іншими закладами. Це обумовлює, окрім розробки програмного забезпечення, також і створення відповідних інформаційних моделей, що дозволять створювати подібне програмне забезпечення у майбутньому, а також легко модифікувати розроблюване ПЗ.

Наукова новизна роботи полягає у тому, що розроблено інформаційну модель програмних засобів для вибору харчування за бажаними критеріями. Розроблена інформаційна модель подана у вигляді набору UML-діаграм та дозволяє створювати та модифікувати програмне забезпечення для вибору харчування за бажаними критеріями.

Для розробки програмного забезпечення для вибору харчування за бажаними критеріями обрано мову PHP, оскільки така мова дозволяє створювати вебсайти, що забезпечують ефективну взаємодію з користувачем.

Як інструментарій розробки обрано PhpStorm, який являє собою інтелектуальний редактор програмного коду для різних мов програмування, зокрема PHP, HTML і JavaScript, а також надає можливості аналізу коду, запобігання помилок у сирцевому коді і автоматизовані засоби рефакторингу.

Розроблено архітектуру програмного забезпечення. Визначено, що користувач через вебінтерфейс (Web User Interface, WebUI) звертається до сторінок вебсайту та отримує можливість замовлення продуктів харчування за заданими критеріями або для бронювання столику у певному закладі. Логіка пошуку пропозицій для користувача відповідно його критеріїв реалізована на сервері. На сервері також організовано збереження інформації про пропозиції різних закладів харчування.

Розроблено інтерфейс програмного забезпечення. Визначено, що завдяки формі бронювання користувач може здійснити бронювання столику, ввівши бажану дату та деякі інші параметри. Як такі параметри можуть бути введені також певні критерії, наприклад, наявність певного виду страв і т.ін. В такому випадку розроблена програма за допомогою внутрішніх засобів синтаксичного аналізу надасть список закладів, де на бажану дату існують вільні місця, та існує набір страв, що буде задовольняти користувача. Варто відзначити, що користувач також може здійснити замовлення з доставкою обраного харчування за бажаними критеріями.

Виконано тестування розробленого програмного забезпечення. Результати тестування підтвердили працездатність розробленого програмного забезпечення для вибору харчування за бажаними критеріями, а також можливість його використання у різних веббраузерах. Виявлено, що функціонал програми, її ергономічність та функції безпеки відповідають технічному завданню.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Скотт Б. Проектирование вебинтерфейсов : Пер. з англ. [Текст] / Б. Скотт, Т. Нейл. – СПб. : Символ-Плюс, 2010. – 352с. : ил.
2. Що таке вебсайт [Електрон. ресурс]. – Режим доступу: <http://ua.smart-ip.net/what-is-a-website>.
3. Види розробки [Електрон. ресурс]. – Режим доступу: <https://internetdevels.ua/blog/front-end-development-vs-back-end-development>.
4. Етапи проектування вебсайту [Електрон. ресурс]. – Режим доступу: <https://venweb.com.ua/etapy-rozrobky-veb-sajtu/>.
5. Что такое сайт [Электрон. ресурс]. – Режим доступа: <http://moolkin.ru/chto-takoe-sayt-internet-sayt-vidy-saytov/>.
6. Интернет-сайт [Электрон. ресурс]. – Режим доступа: <https://www.seonews.ru/glossary/internet-site/>.
7. Поиск лучших ресторанов, кафе и баров [Электрон. ресурс].. – Режим доступа: <https://tomato.ua/>.
8. Гид кафе [Электрон. ресурс]. – Режим доступа: <https://guide-cafe.in.ua/>.
9. Ресторанный консалтинг [Электрон. ресурс]. – Режим доступа: <https://horeca-ukraine.com/>.
10. Booze: заведение по всей Украине [Электрон. ресурс]. – Режим доступа: <https://booze.com.ua/>.
11. Bentley J. L. Writing Efficient Programs (Prentice-Hall Software Series) [Text] / J. L. Bentley. – Englewood Cliffs, NJ : Prentice Hall, 1982. – 170 p.
12. Кнут Д. Искусство программирования [Текст] / Д. Кнут. – 3-е изд. – М.: Вильямс. — Т. 2, 2000 — 832 с.
13. Гото К. Вебрeдизайн [Текст] / Келли Гото, Эмили Котлер. – 2-е изд. – СПб: Символ-Плюс, 2006. – 416 с.
14. Clarke A. Hardboiled Web Design [Текст] / Andy Clarke. –

Smashing Magazine, 2015. — 441 p.

15. Аргерих Л. Профессиональное PHP программирование [Текст] / Л. Аргерих, В. Чой, Д. Коггсхол. — СПб.: Символ. — 2007. — 1048 с.
16. Зандстра, М. PHP. Объекты, шаблоны и методики программирования [Текст] / М. Зандстра. — К. : Apress, 2011. — 560 с.
17. Шлоссейгл Д. Профессиональное программирование на PHP [Текст] / Д.Шлоссейгл. — М. : Вильям, 2008. — 624с.
18. Водолазкий, В. Эффективная работа: PHP 4 [Текст] / В. Водолазкий. — СПб.: Питер, 2008. — 416 с.
19. Хьюгс, С. PHP. Руководство разработчика: Пер.с англ. [Текст] / С. Хьюгс, А. Змиевский. — К. : ДияСофт, 2004. — 384 с.
20. Профессиональное PHP программирование [Текст] / Дж. Кастаньето, Х. Рават, С. Шуман, К. Сколло, Д. Велиаф. — СПб : Символ-Плюс, 2006. — 912 с.
21. Шлоссейгл, Дж. Профессиональное программирование на PHP [Текст] / Дж. Шлоссейгл. — М. : Вильям, 2008. — 624с.
22. MySQL [Электрон. ресурс]. — Режим доступа: <https://ru.wikipedia.org/wiki/MySQL>.
23. Дюбуа, П. MySQL. Сборник рецептов [Текст] / П. Дюбуа. — СПб : Символ-Плюс, 2004. — 1056 с.
24. Экономика и организация производства программных продуктов [Электрон. ресурс]. — 2019. — Режим доступа: <http://megaobzor.net/e-book/book.html>.
25. Методичні вказівки по економічному обґрунтуванню дипломних проектів для студентів спеціальностей 8.05010301 «Програмне забезпечення автоматизованих систем» та 8.05010201 «Інформаційні технології проектування» для усіх форм навчання [Текст] / В.В. Остапенко, Є.М. Сердюк. — Запоріжжя : ЗНТУ, 2015. — 22 с.
26. Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями. НПАОП 0.00-7.15-18. Державні санітарні

правила і норми [Текст]. – Харків : Форт, 2018. – 48 с.

27. ДСТУ 8604:2015 Дизайн і ергономіка. Робоче місце для виконання робіт у положенні сидячи. Загальні ергономічні вимоги [Текст]. – Наказ від 21.12.2015 № 204 Про прийняття нормативних документів України, гармонізованих з міжнародними та європейськими нормативними документами, національних стандартів України та змін до національних стандартів України.

28. Правила улаштування електроустановок [Текст]. Міненергосувугілля України, 2017. – 617с.

29. Основи професійної безпеки та здоров'я людини [Текст]. – підручник / В. В. Березуцький; під ред. проф. В. В. Березуцького. – Харків : НТУ «ХП», 2018. – 553 с.

30. ДСН 3.3.6.042-99 Санітарні норми мікроклімату виробничих приміщень [Текст]. – Харків : Форт, 2000. – 24 с.

31. ДБН В.2.5-28:2018. Природне та штучне освітлення. Посібник [Текст] : проблеми природного і штучного освітлення. – К. : Етін, 2019. – 180 с.

32. Методичні вказівки до лабораторного заняття «Розрахунок загального рівномірного штучного освітлення виробничих приміщень» з дисципліни «Цивільний захист і охорона праці в галузі» [Текст] : для студентів усіх спеціальностей та форм навчання / В.І. Шмирко, О.В. Коробко, Ю.І. Троян. – Запоріжжя : каф. ОПіНС. НУ «Запорізька політехніка», 2020. – 36 с., 3 рис., 12 табл.

33. ДСН 3.3.6.037-99 Санітарні норми виробничого шуму, ультразвуку та інфразвуку [Текст]. – К. : МОЗ України, 1999. – 24 с.

34. ДБН В.1.1-7:2016 Пожежна безпека об'єктів будівництва. Загальні вимоги [Текст]. – Київ : Міністерство регіонального розвитку, будівництва та житлово-комунального господарства України, 2017. – 39 с.

35. ДСТУ Б В.1.1-36:2016 Визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною

небезпекою [Текст]. – Київ : Мінрегіонбуд України, 2016. – 66 с.

36. Охорона праці: навч. посіб. / З. М. Яремко, С. В. Тимошук, О. І. Третяк, Р. М. Ковтун; за ред. проф. З. М. Яремка. [Текст]. – Львів: Видавничий центр ЛНУ імені Івана Франка, 2010. - 374 с.

37. Стеблюк М. І. Цивільна оборона та цивільний захист [Текст] : навч. посіб. для вузів / М. І. Стеблюк. – К. : Знання, 2013. – 487 с.

ДОДАТОК А
Технічне завдання

Вступ

Програма призначена для вибору харчування за бажаними критеріями.

A.1 Підстави для розробки

Підставою для розробки є завдання на дипломну кваліфікаційну роботу на тему «Дослідження та програмна реалізація інформаційно-аналітичної системи для вибору закладу харчування за бажаними критеріями», затверджене наказом по Національному університету «Запорізька політехніка».

A.2 Призначення розробки

Програма призначена для вибору харчування за бажаними критеріями.

A.3 Вимоги до програми

A.3.1 Вимоги до функціональних характеристик

Програма повинна виконувати такі функції:

- інформаційна - ресурс буде містити структуровану інформацію про страви та послуги що надають у піцерії;
- комерційна – додаток буде слугувати рекламною візитівкою;
- функція онлайн бронювання столиків та замовлення їжі на дім.

A.3.2 Вимоги до надійності

Програма повинна забезпечувати коректну обробку помилок у випадку введення користувачем некоректних даних.

A.3.3 Вимоги до експлуатації

Для роботи з програмою користувач повинен володіти достатніми знаннями для коректної роботи з програмно-апаратним комплексом. Вебдодаток повинен нормально функціонувати при безперебійній роботі Інтернету. У разі виникнення збою в роботі, відновлення нормальної роботи програми повинно проводитися після перезавантаження вебсторінки або перезапуску браузеру, а також повторного виконання дій, втрачених до останнього збереження інформації. Умовами експлуатації програми є наявність доступу до мережі Інтернет та гаджет або ПК з браузером, за допомогою якого буде здійснюватись доступ до вебдодатку.

Для нормального функціонування програми на стороні клієнта, повинні виконуватися наступні умови до програмної частини: наявність сучасної версії браузера (Google Chrome, Opera та інші); стабільне підключення до мережі Інтернет.

Розроблена програма безпосередньо встановлюється і налаштовується на сервері. Програма може надаватися і налаштовуватися через Інтернет. Крім того, файли програмного виробу поставляються на дистрибутивному (зовнішньому оптичному) носії (компакт-диску).

A.3.4 Вимоги до складу й параметрів технічних засобів

Для роботи з програмним продуктом рекомендується використовувати наступне апаратне забезпечення:

- ПК під керівництвом операційної системи;

- наявність веббраузеру;
- тактова частота процесора 1 ГГц або більше;
- обсяг оперативної пам'яті не менше 256 Мб;
- обсяг вільного простору на диску - не менше 20 Мб;
- дозвіл екрана монітора не менше 1024x768;
- маніпулятор миша.

A.3.5 Вимоги до інформаційної й програмної сумісності

Для роботи програми необхідна операційна система та наявність веббраузеру.

A.3.6 Вимоги до транспортування і зберігання

Програма повинна зберігатися на жорсткому або гнучкому магнітному диску і транспортуватися в спеціальних боксах.

A.3.7 Вимоги до програмної документації

Програма повинна поставлятися з технічним завданням, керівництвом оператора, керівництвом програміста.

A.4 Стадії та етапи розробки

Програмний продукт повинен бути виконаний у 16-ти тижневий строк з моменту отримання завдання.

ДОДАТОК Б
Опис програми

Б.1 Загальні відомості

Програма розроблена за допомогою мови програмування РНР.

Б.2 Функціональне призначення

Програма призначена для вибору харчування за бажаними критеріями.

Б.3 Опис логічної структури

Програму розроблено на основі клієнт-серверної моделі, з використанням підходів об'єктно-орієнтованого програмування.

Клієнт надає користувачеві зручну можливість заповнити форми для вводу даних, та надає доступ до результату роботи серверу. Має зручний та ергономічний інтерфейс з підказками для користувача та наданням користувачеві доступу до серверу з використанням мережевої технології, протоколу ТСР.

Підпрограма серверу реалізує основні функції програми, зокрема, забезпечує можливості бронювання столиків у закладах харчування та замовлення їжі за певними критеріями.

Б.4 Використані технічні засоби

Для роботи з програмою рекомендується використовувати наступне апаратне та програмне забезпечення:

- ПК під керівництвом операційної системи;
- наявність веббраузеру;
- тактова частота процесора 1 ГГц або більше;
- обсяг оперативної пам'яті не менше 256 Мб;

- обсяг вільного простору на диску - не менше 20 Мб;
- дозвіл екрана монітора не менше 1024x768;
- маніпулятор миша.

Б.5 Виклик і завантаження

Для запуску додатку потрібно виконати наступні дії:

- запустити встановлений веббраузер;
- у адресному рядку набрати адресу сайту, після чого з'явиться головна сторінка додатку.

Для замовлення харчування:

- запустити вебдодаток;
- вибрати на панелі фільтрів потрібні критерії харчування;
- переглянути доступні варіанти та натиснути кнопку «Замовити».

Для замовлення столику:

- запустити вебдодаток;
- вибрати необхідну дату та час відвідування закладу харчування;
- завершити резервування.

Б.6 Вхідні дані

Вхідними даними для програми є інформація про запит користувача.

Б.7 Вихідні дані

Вихідними даними для програми є такі: повідомлення про запис користувача у заклад харчування на певну дату та час, або повідомлення про замовлення певного виду харчування.

ДОДАТОК В
Текст програми

В.1 Текст файла .htaccess

```
RewriteEngine On
RewriteBase /magnitlast/
RewriteRule ^index\.php$ - [L]
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule ^(.*)$ /magnitlast/index.php?route=$1 [L]
```

В.2 Текст файла index.php

```
<?php
$route = $_GET['route'];

session_start();
include("templates/auth_cookie.php");

switch ($route) {
    case '':
        require 'templates/main.php';
        break;
    case 'about':
        require 'templates/about.php';
        break;
    case 'order':
        require 'templates/order.php';
        break;
    case 'article':
        require 'templates/article.php';
        break;
    case 'info':
        require 'templates/info.php';
        break;
    case 'goods':
        require 'templates/goods.php';
        break;
    case 'cart':
        require 'cart.php';
        break;
    case 'cart&action=confirm':
        require 'cart.php?action=confirm';
        break;
    case 'cart&action=sucsess':
        require 'cart.php?action=sucsess';
        break;
    case 'fav':
        require 'templates/fav.php';
        break;
    case 'search':
        require 'search.php';
        break;
    case 'viev_cat':
        require 'viev_cat.php';
        break;
    case 'viev_pages':
```

```

require 'view_pages.php';
break;
    case 'view_content':
require 'view_content.php';
break;
    case 'feedback':
require 'templates/feedback.php';
break;
    case 'registration':
require 'registration.php';
break;
    case 'profile':
require 'profile.php';
break;
    default;
require 'templates/404.php';
break;
}

```

В.3 Текст файла profile.php

```

<?php
    session_start();

if ($_SESSION['auth'] == 'yes_auth')
{
include("templates/db_connect.php");
include("templates/header.php");
$anton = '';
$q = $_GET["q"];
$id = $_GET["id"];

if ($_POST["save_submit"])
{

    $_POST["info_surname"] = $_POST["info_surname"];
    $_POST["info_name"] = $_POST["info_name"];
    $_POST["info_patronymic"] = $_POST["info_patronymic"];
    $_POST["info_address"] = $_POST["info_address"];
    $_POST["info_phone"] = $_POST["info_phone"];
    $_POST["info_email"] = $_POST["info_email"];

    $error = array();

    $pass    = md5($_POST["info_pass"]);
    $pass    = strrev($pass);
    $pass    = "9nm2rv8q".$pass."2yo6z";

    if($_SESSION['auth_pass'] != $pass)
    {
        $error[]='Неверный текущий пароль!';
    }else
    {
        if($_POST["info_new_pass"] != "")
        {

```

```

        if(strlen($_POST["info_new_pass"]) < 7 ||
strlen($_POST["info_new_pass"]) > 15)
        {
            $error[]='Укажите новый пароль от 7 до 15
СИМВОЛОВ!';
        }else
        {
            $newpass    = md5($_POST["info_new_pass"]);
            $newpass    = strrev($newpass);
            $newpass    = "9nm2rv8q".$newpass."2yo6z";
            $newpassquery = "pass='".$newpass.'",";
        }
    }

    if(strlen($_POST["info_surname"]) < 3 ||
strlen($_POST["info_surname"]) > 15)
    {
        $error[]='Укажите Фамилию от 3 до 15 СИМВОЛОВ!';
    }

    if(strlen($_POST["info_name"]) < 3 ||
strlen($_POST["info_name"]) > 15)
    {
        $error[]='Укажите Имя от 3 до 15 СИМВОЛОВ!';
    }

    if(strlen($_POST["info_patronymic"]) < 3 ||
strlen($_POST["info_patronymic"]) > 25)
    {
        $error[]='Укажите Отчество от 3 до 25 СИМВОЛОВ!';
    }

    if(!preg_match("/^(?:[a-z0-9]+(?:[-_]?[a-z0-9]+)?@[a-
z0-9_.-]+(?:\.[a-z0-9]+)?\.[a-
z]{2,5})$/i",trim($_POST["info_email"])))
    {
        $error[]='Укажите корректный email!';
    }

    if(strlen($_POST["info_phone"]) == "")
    {
        $error[]='Укажите номер телефона!';
    }

    if(strlen($_POST["info_address"]) == "")
    {
        $error[]='Укажите адрес доставки!';
    }

    }

    if(count($error))
    {
        $_SESSION['msg'] = "<p align='left' id='form-
error'>".implode('<br />',$error)."</p>";
    }else
    {

```

```

$_SESSION['msg'] = "<p align='left' id='form-
success'>Данные успешно сохранены!</p>";

$update = mysqli_query($link, "UPDATE reg_user SET
name='".$_POST['info_name']."' , surname='".$_POST['info_surname']."' ,
pass='$newpass', email='".$_POST['info_email']."' ,
phone='".$_POST['info_phone']."' ,
patronymic='".$_POST['info_patronymic']."' WHERE login =
'{"$_SESSION['auth_login']}'}");

    if ($newpass){ $_SESSION['auth_pass'] = $newpass; }

    $_SESSION['auth_surname'] = $_POST["info_surname"];
    $_SESSION['auth_name'] = $_POST["info_name"];
    $_SESSION['auth_patronymic'] = $_POST["info_patronymic"];
    $_SESSION['auth_address'] = $_POST["info_address"];
    $_SESSION['auth_phone'] = $_POST["info_phone"];
    $_SESSION['auth_email'] = $_POST["info_email"];

    }

    }

?>
<div class="body">
    <?php

        if($_SESSION['msg'])
        {
            echo $_SESSION['msg'];
            unset($_SESSION['msg']);
        }

    ?>
    <?php
    switch ($q) {
        case 'info':

            echo '
                <ul class="profile-steps">
                    <li> <a href="profile&q=info" style="border-bottom:1px solid
#E5E5E5">Изменение профиля</a></li>
                    <li><a href="profile&q=status">Мои заказы</a></li>
                </ul>

                <form method="post">

                <ul id="info-profile">
                    <li>
                        <label for="info_pass">Текущий пароль</label>
                        <span class="star">*</span>
                        <input type="text" name="info_pass" id="info_pass" value="">
                    </li>

                    <li>
                        <label for="info_new_pass">Новый пароль</label>
                        <span class="star">*</span>
                        <input type="text" name="info_new_pass" id="info_new_pass"
value="">

```

```

</li>

<li>
<label for="info_surname">Фамилия</label>
<span class="star">*</span>
<input type="text" name="info_surname" id="info_surname"
value="'.$_SESSION["auth_surname"].'">
</li>

<li>
<label for="info_name">Имя</label>
<span class="star">*</span>
<input type="text" name="info_name" id="info_name"
value="'.$_SESSION["auth_name"].' " >
</li>

<li>
<label for="info_patronymic">Отчество</label>
<span class="star">*</span>
<input type="text" name="info_patronymic" id="info_patronymic"
value="'.$_SESSION["auth_patronymic"].' ">
</li>

<li>
<label for="info_email">E-mail</label>
<span class="star">*</span>
<input type="text" name="info_email" id="info_email"
value="'.$_SESSION["auth_email"].' ">
</li>

<li>
<label for="info_phone">Телефон</label>
<span class="star">*</span>
<input type="text" name="info_phone" id="info_phone"
value="'.$_SESSION["auth_phone"].' ">
</li>

<li>
<label for="info_address">Адрес доставки</label>
<span class="star">*</span>
<textarea name="info_address"  > '$_SESSION["auth_address"].'
</textarea>
</li>

</ul>

<p><input type="submit" id="form_submit" name="save_submit"
value="Сохранить" /></p>
</form>';
break;

case 'status' :
echo '
<ul class="profile-steps">
<li> <a href="profile&q=info">Изменение профиля</a></li>
<li><a href="profile&q=status" style="border-bottom:1px
solid #E5E5E5">Мои заказы</a></li>
</ul>

```

```

<div id="header-list-orders">
    <div id="order-head1">Номер заказа</div>
    <div id="order-head2">Дата заказа</div>
    <div id="order-head3">Время заказа</div>
    <div id="order-head4">Сумма</div>
    <div id="order-head5">Статус заказа</div>
    <div id="order-head5">Информация</div>
</div>

';

    $result = mysqli_query($link, "SELECT * FROM orders WHERE
orders.order_login='{$_SESSION['auth_login']}'");
    if (mysqli_num_rows($result) > 0)
    {
        while ($row = mysqli_fetch_array($result)) {
            if ($row["order_status"] == "yes")
            {
                $status = '<span class="green">Обработан</span>';
            }else
            {
                $status = '<span class="red">Ожидает обработки</span>';
            }

            $x .= '
            <p style="float:left;clear:both;margin-left:40px">№
            '.$row["orders_id"].'</p>
            <p style="margin-left:125px;float:left">
            '.$row["order_datetime"].'</p>
            <p style="margin-left:115px;float:left">
            '.$row["order_time"].'</p>
            <p style="margin-left:110px;float:left; width:65px;text-
            align:center;"> '.$row["order_full_price"].' грн</p>
            <p style="margin-left:70px;float:left;width:150px;text-
            align:center"> '.$status.'</p>
            <p style="margin-left:60px;float:left;width:150px;text-
            align:left"><a href="profile&id='.$row["orders_id"].'">Подробнее</a>
            </p>
            '
            ;

        }

        echo ' '.$x.'';
    }
    else
    {
        echo '<h3 id ="clear-cart" align="center">Вы еще не совершали
заказов</h3>';
    }
    break;

default:

    echo '
    <ul class="profile-steps">
        <li> <a href="profile&q=info">Изменение профиля</a></li>

```

```

        <li><a href="profile&q=status" style="border-bottom:1px
solid #E5E5E5">Мои заказы</a></li>
    </ul>

```

```

<div id="header-list-orders">
    <div id="order-info-head1">Изображение</div>
    <div id="order-info-head2">Наименование товара</div>
    <div id="order-info-head3">Цвет</div>
    <div id="order-info-head4">Размер</div>
    <div id="order-info-head5">Кол-во</div>
    <div id="order-info-head6">Цена</div>
</div>
';
$result = mysqli_query($link, "SELECT * FROM orders,
orders_products, table_products WHERE
orders.order_login='{$_SESSION['auth_login']}' AND
orders_products.order_id=$id AND orders.orders_id=$id AND
table_products.products_id=orders_products.products_id");
if (mysqli_num_rows($result) > 0)
{
    while ($row = mysqli_fetch_array($result)){
        $tovar .= '
        <p style="float:left;clear:both;margin-left:40px"></p>
        <p style="text-align:center;width:150px;margin-
left:135px;float:left"><a
href="view_content&id='.$row["products_id"].'">
'.$row["title"].'</a></p>
        <p style="margin-left:130px;float:left;width:60px;text-
align:center"> '.$row["order_color"].'</p>
        <p style="margin-left:115px;float:left;width:20px;text-
align:center"> '.$row["order_size"].' </p>
        <p style="margin-left:165px;float:left;width:10px;text-
align:center"> '.$row["order_count"].'</p>
        <p style="margin-left:130px;float:left">
'.$row["order_product_price"].' грн</p>
        ';
    }
}
echo '$tovar.';
break;

}
?>
</div>

<?php
}
else { header("Location: index.php"); }

include("templates/footer.php");
?>

```

В.4 Текст файла registration.php

```

<?php
session_start();
include("templates/db_connect.php");
include("templates/header.php");
include("templates/auth_cookie.php");

if ($_SESSION['auth'] == 'yes_auth')
{

    header("Location: index.php");

}
else { echo' <div class="body">

<h2 class="h2-title">Registration</h2>
<form method="post" id="form_reg" action="reg/handler_reg.php">
<p id="reg_message"></p>
<div id="block-form-registration">
<ul id="form-registration">

<li>
<label>Login</label>
<span class="star" >*</span>
<input type="text" name="reg_login" id="reg_login" />
</li>

<li>
<label>Password</label>
<span class="star" >*</span>
<input type="text" name="reg_pass" id="reg_pass" />
<span id="genpass">Generate</span>
</li>

<li>
<label>Surname</label>
<span class="star" >*</span>
<input type="text" name="reg_surname" id="reg_surname" />
</li>

<li>
<label>Name</label>
<span class="star" >*</span>
<input type="text" name="reg_name" id="reg_name" />
</li>

<li>
<label>Patronymic</label>
<span class="star" >*</span>
<input type="text" name="reg_patronymic" id="reg_patronymic" />
</li>

<li>
<label>E-mail</label>
<span class="star" >*</span>
<input type="text" name="reg_email" id="reg_email" />

```

```

</li>

<li>
<label>Mobile phone</label>
<span class="star" >*</span>
<input type="text" name="reg_phone" id="reg_phone" />
</li>

<li>
<div id="block-captcha">


<input type="text" name="reg_captcha" id="reg_captcha" />

<p id="reloadcaptcha">Reload</p>
</div>
</li>

</ul>
</div>

<p><input type="submit" name="reg_submit" id="form_submit"
value="Registration" /></p>

</form>';}
?>

</div>

<?php
include("templates/footer.php");
?>

```

B.5 Текст файла registration.php

```

<?php
include("templates/db_connect.php");
include("templates/header.php");
session_start();
include("auth_cookie.php");
?>
<div class="body">
    <div class="slidershow-container">
        <div class="mySlides fade"></div>
        <div class="mySlides fade"></div>
        <div class="mySlides fade"></div>
        <div class="mySlides fade"></div>
        <div onclick="plusSlides(-1)" class="prev
knopki">&#10094</div>

```

```

        <div onclick="plusSlides(1)" class="next
knopki">&#10095</div>
    </div>
    <div class="dot-container">
        <span class="dot" onclick="currentSlide(1)"></span>
        <span class="dot" onclick="currentSlide(2)"></span>
        <span class="dot" onclick="currentSlide(3)"></span>
        <span class="dot" onclick="currentSlide(4)"></span>
    </div>
    <p class="categoriya1">Новости</p>
    <ul class="block-news">
        <?php
            $result = mysqli_query($link, "SELECT * FROM
`news` LIMIT 4");
            if(mysqli_num_rows($result) > 0) {
                $row = mysqli_fetch_array($result);
                do {
                    echo ( '
                        <li>
                            <a class="title-news"
href="">'. $row["title"].'</a>
                                <div class="images-news"></div>
                                    <p>'. $row["text"].'<a class="read-
more" href=""> читать далее..</a></p>
                                        </li>');
                                } while ($row = mysqli_fetch_array($result));
                            }
                        ?>
                    </ul>
                    <div class="categoriya">Новинки</div>
                    <ul id="block-tovar-grid1">
                        <?php
                            if ($_SESSION['auth'] != 'yes_auth') { $j="AND
favourite.fav_ip = '{$_SERVER['REMOTE_ADDR']}'"; }
                            else { $j="AND favourite.login =
'".$_SESSION['auth_login']."'"; }
                            $result = mysqli_query($link, "SELECT * FROM
table_products ORDER BY products_id DESC LIMIT 6 ");
                            if(mysqli_num_rows($result) > 0) {
                                $row = mysqli_fetch_array($result);
                                do {
                                    $scp = mysqli_query($link, "SELECT *
FROM uploads_images WHERE products_id='{ $row["products_id"]}' AND
main=1 LIMIT 1 ");
                                    $scolr=mysqli_fetch_array($scp);
                                    echo '
                                        <li>
                                            <div class="block-images-grid">
                                                
                                                    </div>';
                                            $result1 = mysqli_query($link, "SELECT *
from table_products, favourite WHERE table_products.products_id =
favourite.fav_id_products $j ");
                                            $i=0;
                                            $row1=mysqli_fetch_array($result1);
                                            do{

```

```

                                if(mysqli_num_rows($result1)>0 &&
$row["products_id"]== $row1["fav_id_products"]) {
                                    echo'<span class="add-cart-
style-grid krasniy" fid="'. $row["products_id"].'"></span>';
                                    break;
                                }
                                else { $i=$i++; }
                                } while ($row1 =
mysqli_fetch_array($result1));
                                if ($i==$row1) {echo '<span class="add-
cart-style-grid" fid="'. $row["products_id"].'"></span>';}
                                echo'
                                    <p class="style-title-grid"><a
href="view_content&id="'. $row["products_id"].'">'. $row["title"].'</a></
p>
                                    <p class="style-price-
grid">'. $row["price"].' грн.</p>
                                    </li>';
                                } while ($row =
mysqli_fetch_array($result));
                                }
                                ?>
                                </ul>
                                <div class="categoriya2">Блог</div>
                                <div class="article-block">
                                <?php
                                    $result = mysqli_query($link, "SELECT * FROM
articles LIMIT 4");
                                    if(mysqli_num_rows($result) > 0) {
                                        $row = mysqli_fetch_array($result);
                                        do {
                                            echo'
                                                <div class="statia">
                                                    <p
class="article_name">'. $row["article_name"].'</p>
                                                    <div class="images-news"></div>
                                                    <p>'. $row["article_description"].'<a
class="read-more" href=""> читать далее..</a></p>
                                                    </div>';
                                                } while ($row = mysqli_fetch_array($result));
                                            }
                                        ?>
                                    </div>

                                </div>
                                <?php
                                include("templates/footer.php");
                                ?>

```

В.6 Текст файла auth.php

```

<?php

if($_SERVER["REQUEST_METHOD"] == "POST")
{

```

```

include('db_connect.php');

$login = $_POST["login"];

$pass   = md5($_POST["pass"]);
$pass   = strrev($pass);
$pass   = strtolower("9nm2rv8q".$pass."2yo6z");

if ($_POST["rememberme"] == "yes")
{

setcookie('rememberme',$login.'+'.$pass,time()+3600*24*31, "/");

}

$result = mysqli_query($link, "SELECT * FROM reg_user WHERE
(login = '$login' OR email = '$login') AND pass = '$pass'");
If (mysqli_num_rows($result) > 0)
{
    $row = mysqli_fetch_array($result);
    session_start();
    $_SESSION['auth'] = 'yes_auth';
    $_SESSION['auth_pass'] = $row["pass"];
    $_SESSION['auth_login'] = $row["login"];
    $_SESSION['auth_surname'] = $row["surname"];
    $_SESSION['auth_name'] = $row["name"];
    $_SESSION['auth_patronymic'] = $row["patronymic"];
    $_SESSION['auth_phone'] = $row["phone"];
    $_SESSION['auth_email'] = $row["email"];
    echo 'yes_auth';

}
else
{
    echo 'no_auth';
}
}

?>

```

B.7 Текст файлу auth_cookie.php

```

<?php
if ($_SESSION['auth'] != 'yes_auth' && $_COOKIE["rememberme"])
{

    $str = $_COOKIE["rememberme"];

    $all_len = strlen($str);

    $login_len = strpos($str, '+');

    $login = substr($str, 0, $login_len);

    $pass = substr($str, $login_len+1, $all_len);

```

```

        $result = mysqli_query($link, "SELECT * FROM reg_user WHERE
(login = '$login' or email = '$login') AND pass = '$pass'");
        If (mysqli_num_rows($result) > 0)
        {
            $row = mysqli_fetch_array($result);
            session_start();
            $_SESSION['auth'] = 'yes_auth';
            $_SESSION['auth_pass'] = $row["pass"];
            $_SESSION['auth_login'] = $row["login"];
            $_SESSION['auth_surname'] = $row["surname"];
            $_SESSION['auth_name'] = $row["name"];
            $_SESSION['auth_patronymic'] = $row["patronymic"];
            $_SESSION['auth_phone'] = $row["phone"];
            $_SESSION['auth_email'] = $row["email"];

        }

    }
?>

```

B.8 Текст файла view_content.php

```

<?php
include("templates/db_connect.php");
include("templates/header.php");
$test = $_POST["test"];
$id = $_GET["id"];
?>
<div class="body">
    <?php

        $result1 = mysqli_query($link, "SELECT * FROM
table_products WHERE table_products.products_id='$id'");
        If (mysqli_num_rows($result1) > 0)
        {
            $row1 = mysqli_fetch_array($result1);
            $array = unserialize($row1["size"]);
            echo '
                <div id="fotkitovara">
                    ';

                echo '
            </div>';

            $result = mysqli_query($link, "SELECT * FROM uploads_images
WHERE products_id='$id'");
            If (mysqli_num_rows($result) > 0)
            {
                $row = mysqli_fetch_array($result);
                echo '<div id="block-img-slide">
                    <ul>';

            do
            {

                $img_path = 'uploads_images/'. $row["image"];

```

```

$max_width = 70;
$max_height = 70;
list($width, $height) = getimagesize($img_path);
$ratioh = $max_height/$height;
$ratiow = $max_width/$width;
$ratio = min($ratioh, $ratiow);

$width = intval($ratio*$width);
$height = intval($ratio*$height);

echo '
<li>
<a class="image-modal" href="#image'.$row["id"].'"></a>
</li>
<a style="display:none;" class="image-modal" rel="group"
id="image'.$row["id"].'" ></a>
';
}
while ($row = mysqli_fetch_array($result));
echo '
</ul>
</div>
';
}
echo'
<div id="tovarinfo">
<p>Имя товара: '.$row1["title"].'</p>
<p>Цена: '.$row1["price"].'</p>
<p>Цвет:</p>
<form method="POST" action="" tid="'.$row1["products_id"].'"
id="order_form">';

    $result1 = mysqli_query($link, "SELECT * FROM table_products,
color_products, color WHERE table_products.products_id='$id' and
table_products.products_id=color_products.products_id and
color_products.colors_id=color.color_id and
color_products.products_count>0");
    if (mysqli_num_rows($result1) > 0)
    {
        $row1 = mysqli_fetch_array($result1);
        $array = unserialize($row1["size"]);
        do{
            echo'
            <input required class="colorr" id="'.$row1["color_id"].'"
type="radio" style="display:none" name="order_color"
value="'. $row1["color_name"].'"><label for="'. $row1["colors_id"].'"
style="background-color: '.$row1["color_icon"].'"
class="color_lable"></label>';
        }

        while ($row1 = mysqli_fetch_array($result1));

        ;}

        echo'
        <p>Выберите нужный размер:</p>

```

```

        <p class="setka"><a href="info#gena"> (Размерная
сетка)</a></p>
';

```

```

        $result1 = mysqli_query($link, "SELECT * FROM size,
size_products, color, color_products, table_products WHERE
table_products.products_id='$id' and size_products.products_id=$id and
size_products.size_id=size.size_id and
color.color_id=size_products.colorss_id and
color.color_id=color_products.colors_id and
size_products.products_count>0 group by
colorss_id,size_products.size_id");

```

```

        if (mysqli_num_rows($result1) > 0)
        {
            $row1 = mysqli_fetch_array($result1);

            do{
                if ($row1["colorss_id"]!=$row1["colors_id"]) {
                    echo'<input required style="display:none" type="radio"
id="'. $row1["size_id"].' ' . $row1["colorss_id"].' " name="order_size"
value="'. $row1["size_name"].' "><label win="'. $row1["colorss_id"].' "
style="display:none" class="size_label" for="'. $row1["size_id"].' '
'. $row1["colorss_id"].' ">'. $row1["size_name"].' </label>';
                }
            }
            else{
                echo'<input required style="display:none" type="radio"
id="'. $row1["size_id"].' ' . $row1["colorss_id"].' " name="order_size"
value="'. $row1["size_name"].' "><label win="'. $row1["colorss_id"].' "
style="display:none" class="size_label" for="'. $row1["size_id"].' '
'. $row1["colorss_id"].' ">'. $row1["size_name"].' </label>';
            }
        }
        while ($row1 = mysqli_fetch_array($result1));
    };

```

```

        $result1 = mysqli_query($link, "SELECT * FROM
table_products WHERE table_products.products_id=$id");
        if (mysqli_num_rows($result1) > 0)
        {
            $row1 = mysqli_fetch_array($result1);

```

```

        echo '

        <input type="hidden" name="order_product"
value="'. $row1["products_id"].' ">
        <button type="submit" class="box">Добавить в </button>
        <p>Описание товара: '. $row1["description"].' </p>

        </form>
        </div>

```

```

';}

```

```

?>
</div>
</div>
<?php

```

```
include("templates/footer.php");
?>
```

В.9 Текст файла auth_cookie.php

```
<footer>
    <p class="feedback">Contacts: +380634232642
+380504231643</p>
    <a class="feedback" href="http://vk.com"></a>
    <a class="feedback" href="http://facebook.com"></a>
    <a class="feedback" href="http://instagram.com"></a>
    <br>
    <a class="feedback" href="feedback">Отзывы</a>
    <br>
    <a class="feedback"
href="http://Vashmail@gmail.com">Vashmail<span
class="sobaka">@</span>gmail.com</a>
    <br>
    <span>Copyright © 2017 MaGnit</span>
</footer>
<script type="text/javascript"
src="js/jquery.1.7.1.min.js"></script>
<script src="js/textchange.js"></script>
<script type="text/javascript"
src="fancybox/jquery.fancybox.js"></script>
<?php
    switch ($route) {
        case 'view_content':
            echo'<script type="text/javascript">
$(document).ready(function() {

    $(".image-modal").fancybox();
    $(".send-review").fancybox();
    $(".ul.tabs").click({content: ".tabs_content", animate: true,
effect:"fade"});

});
</script>
<script src="js/jsjsjs.js"></script>';
            break;

        case '':
            echo'<script src="js/script.js"></script>
<script src="js/jsjsjs.js"></script>';
            break;
        case 'feedback':
            echo'<script type="text/javascript">
$(document).ready(function() {

    $(".image-modal").fancybox();
    $(".send-review").fancybox();
```

```

    $(".ul.tabs").click({content: ".tabs_content", animate: true,
effect:"fade"});

});
</script>
<script src="js/jsjsjs.js"></script>';
    default:
        echo'<script src="js/jsjsjs.js"></script>';
    } ?>
<script type="text/javascript" src="js/jquery.form.js"></script>
<script type="text/javascript"
src="js/jquery.validate.js"></script>
</body>

</html>

```

В.10 Текст файла style.css

```

body {
    background: white;
    max-width: 1902px;
    margin: 0 auto;
    height: 100%;
    min-width: 780px;
}

html {height: 100%;}
.body {
    min-height: calc(100% - 120px);
    max-width: 1902px;
}
header {
    width: 100%;
    max-width: 1902px;
    min-width: 780px;
}
.logo {
    margin-top: 15px;
    width: 290px;
    padding-left: 10px;
    float: left
}
.golova {
    height: 80px;
    background-color: white;
}
.search {
    width: auto;
    cursor: pointer;
}

.logotd {
    height: 45px;
    text-align: top;
}

```

```

header {
    border-bottom: 1px solid rgb(80, 82, 80);
    z-index: 53245;
    background: rgba(255,255,255,0.9);
    position: fixed;
    top: 0;
    height: 110px;
    width: 100%
}
.feedback {
    color: rgb(80, 82, 80);
    margin-top: 2px;
    margin-bottom: 5px;
}
footer span {
    color: rgb(80, 82, 80);
}
.feedback img {
    width: 20px;
    height: 20px;
    margin-left: 20px;
}
.feedback img:hover {
    transition: 0.5s ease;
    transform: scale(1.5);
}
footer {
    clear: both;
    border-top: 1px solid rgb(80, 82, 80);
    max-width: 1902px;
    background-color: #f7f7f7;
    text-align: center;
    height: 120px;
}
footer a {
    text-decoration: none;
    color: rgb(80, 82, 80);
}
.agit {
    height: 150px;
    width: 100%;
    font-size: 40px;
    font-weight: bold;
    background: url('/images/test2.jpg') no-repeat;
}
.agittext {
    position: relative;
    text-decoration: none;
    color: white;
    left: 48%;
    top: 50px;
}
#agit {
    text-decoration: none;
}
#agit1 {
    width: 100%;
    margin-left: 3%;
}

```

```

}
.agit:hover {
    transition: 0.6s ease;
    opacity: 0.8;
}
.main-menu a {
    color: black;
    text-decoration: none;
    font-weight: bold;
    font-size: 16px;
}
.main-menu a:hover {
    transition: 0.35s ease;
    color: grey;
}
.navigation {
    background: white;
    z-index: 54353543;
    width: 100%;
    position: relative;
    height: 30px;
    padding-top: 90px;
}
.nomera {margin-bottom: 7px;}
.first, .second, .third {
    display: inline-block;
    background: url('../images/test4.jpg') no-repeat;
    height: 300px;
    width: 25%;
    background-size: 100% 100%;
    text-align: center;
    color: white;
    font-weight: bold;
    text-shadow: black 1px 1px 0, black -1px -1px 0,
    black -1px 1px 0, black 1px -1px 0;
    font-size: 21px;
    margin-left: 6%;
    margin-top: 40px;
}

#firsttt, #secondd, #thirdd {
    margin-top: 150px;
    text-align: center;
    width: content-width;
}
.firstttt, .secondddd, .thirdddd {
    margin-top: -25px;
}

#firsttt:hover, #secondd:hover, #thirdd:hover {
    transition: 0.6s ease;
    opacity: 0.7;
    transform: scale(1.3);
}
#firsttt a, #secondd a, #thirdd a {
    color: white;
    text-decoration: none;
}

```

```

.categoriya {
    clear: both;
    text-align: center;
    font-size: 40px;
}

.categoriya1 {
    text-align: center;
    font-size: 40px;
}

.categoriya2 {
    text-align: center;
    font-size: 40px;
}

.article-block {
    -webkit-padding-start: 0px;
    margin: auto;
    max-width: 980px;
}

.article-block:after {
    content: '.';
    clear: both;
    display: block;
}

.statia {
    float: left;
    margin-right: 15%;
    height: 550px;
}

.statia:nth-child(2n) {
    margin-right: 15px;
}

.statia:nth-child(2n+1) {
    margin-left: 15px;
}

.statia p {
    height: auto;
    width: 400px;
    font-size: 18px;
    -webkit-margin-before: 0;
}

.statia1 {
margin-left: 13%;
-webkit-padding-start: 0;
height: auto;
width: 30%;
float: left;
color: black;

font-size: 21px;
margin-top: 10px;
}

```

```

.statial:nth-child(1), .statial:nth-child(2) {
margin-left:13%;
-webkit-padding-start: 0;
height:auto;
width:30%;
float:left;
color:black;

font-size:21px;
margin-top:110px;
}
@media (max-width: 1000px) {
    .statia {
        margin-right: 5%;
    }
    .article-block { max-width: 843px; }
    .statia:nth-child(2n) { margin-right: 0; }
    .statia:nth-child(2n+1) { margin-left: 0; }
}
@media (max-width: 857px) {
    .statia {
        float: none;
        margin: 0;
        height: auto;
    }
    .article-block { max-width: 400px; }
}

.article_name{
    text-align: center;
    margin-bottom:20px;
    font-size:15pt;
    font-weight: normal;
}
/*                      SEARCH                      */

#input-search {
    position: absolute;
    outline: none;
    display: block;
    float: left;
    width: 130px;
    height: 20px;
    border: none;

    border-bottom: 1px solid rgb(80, 82, 80);
    background: transparent;
    color: rgb(80, 82, 80);
    font-weight: bolder
}

#button-search {
    padding: 0;
    display: block;
    float: left;

    width: 17px;
    height: 17px;

```

```

        background: url(../images/poisk.png);
        background-size: 100% 100%;
        border: none;
        margin-left: 130px;
        padding-left: 15px;
        cursor: pointer;
    }

/*          SLIDER          */

.slideshow-container {
    position: relative;
    width: 90%;
    margin: auto;
    margin-top: 109px;
}

.knopki {
    position: absolute;
    top: 39%;
    display: flex;
}

.prev,
.next {
    z-index: 999;
    display: inline-block;
    cursor: pointer;
    padding: 1%;
    color: white;
    font-weight: bold;
    font-size: 18px;
    transition: 0.6s ease;
    border-radius: 0 3px 3px 0;
}

.next {
    display: inline-block;
    margin-left: auto;
    right: 0;
}

.prev:hover,
.next:hover {
    background-color: rgba(0, 0, 0, 0.8);
}

.mySlides {
    text-align: center;
}

.text {
    color: #f2f2f2;
    font-size: 22px;
    padding: 8px 12px;
    position: relative;
    bottom: 90px;
}

```

```

}

.dot {
    position: relative;
    cursor: pointer;
    height: 13px;
    width: 13px;
    margin: 5px 2px 3px;
    background-color: #bbb;
    border-radius: 50%;
    display: inline-block;
    transition: background-color:0.6s ease;
}

.dot-container {
    margin-top: 5px;
    margin-left: 48%;
}

.activen,
.dot:hover {
    transition: 0.3s ease;
    background-color: #717171;
}

.fade {
    -webkit-animation-name: fade;
    -webkit-animation-duration: 1s;
    animation-name: fade;
    animation-duration: 1s;
}

/*          TOVARI          */

.salemenu {
    z-index: 1;
    position: relative;
    margin-top: 50px;
    width: 200px;
    background: #BFBFBF;
    height: 1300px;
    margin-bottom: -45px;
}

.salemenu ul {
    text-align: center;
    padding-top: 10%;
}

.salemenu ul li {
    list-style-type: none;
    margin-left: -25%;
    height: 37px;
    width: 218px;
    border-bottom: 0.3px inset;
    line-height: 37px;
    margin-left: -60px;
}

```

```

.salemenu ul a {
    color: black;
    text-decoration: none;
}

.salemenu ul: first-child {
    margin-bottom: 20px;
}

.salemenu ul li:hover {
    transition: 0.2s ease;
    background: #CBD2AC;
}

.menlogo {
    position: relative;
    float: left;
    background: url('../images/test4.jpg');
    background-size: 100% 100%;
    width: 100.22%;
    height: 500px;
    margin-top: -99.99%;
    margin-left: 11.7%;
}

.men-catalog {
    display: inline-block;
    width: 95%;
    height: content-height;
    margin-left: 5.0%;
    margin-top: -85%;
    position: relative;
    overflow: hidden;
    background: white;
    height: content;
    float: all;
}

.stranici {
    position: relative;
    width: auto;
    margin-left: 45%;
    margin-bottom: 5px;
}

.stranici a {
    color: black;
    text-decoration: none;
}

.o_nas {
    position: relative;
    margin-top: 20px;
    margin-left: 80px;
    font-size: 18px;
    width: content;
}

.zagolovok {
    width: content;
}

```

```

        height: 30px;
        margin-top: 160px;
        margin-left: 70%;
        font-size: 30px;
        font-weight: bolder;
    }

    .h {
        width: content;
        height: 50px;
        font-weight: bolder;
        color: black;
        font-size: 30px;
        margin-top: 50px;
        text-align: center;
    }

    .oplata,
    .zakaz,

```

B.11 Текст файла cart.php

```

<?php
    session_start();
    include("templates/db_connect.php");
    include("templates/header.php");
    include("include/auth_cookie.php");
    ?>
    <div class="body">

<?php
    $id = $_GET["id"];
    $action = $_GET["action"];
    $order_color = $_POST["order_color"];

    if ($_SESSION['auth'] == 'yes_auth')
    {

        switch ($action) {

            case 'clear':
                $clear = mysqli_query($link, "DELETE FROM cart WHERE
cart.login_id = '{$_SESSION['auth_login']}'");
                break;

            case 'delete':
                $delete = mysqli_query($link, "DELETE FROM cart WHERE
cart_id = '$id' AND cart.login_id = '{$_SESSION['auth_login']}'");
                break;

        }

    }

}

```

```

else {

    switch ($action) {

        case 'clear':
            $clear = mysqli_query($link, "DELETE FROM cart WHERE
cart_ip = '{$_SERVER['REMOTE_ADDR']}'");
            break;

        case 'delete':
            $delete = mysqli_query($link, "DELETE FROM cart WHERE
cart_id = '$id' AND cart_ip = '{$_SERVER['REMOTE_ADDR']}'");
            break;

    }

    if (isset($_POST["submitdata"]))
    {
        $_SESSION["order_delivery"] = $_POST["order_delivery"];
        $_SESSION["order_fio"] = $_POST["order_fio"];
        $_SESSION["order_email"] = $_POST["order_email"];
        $_SESSION["order_phone"] = $_POST["order_phone"];
        $_SESSION["order_address"] = $_POST["order_address"];
        $_SESSION["order_note"] = $_POST["order_note"];
        header("Location: cart.php?action=completion");
    }

    if ($_SESSION['auth'] == 'yes_auth')
    {

        $result = mysqli_query($link, "SELECT * FROM cart,
table_products WHERE cart.login_id = '{$_SESSION['auth_login']}' AND
table_products.products_id = cart.cart_id_products");
        If (mysqli_num_rows($result) > 0)
        {
            $row = mysqli_fetch_array($result);

            do
            {
                $int = $int + ($row["cart_price"] * $row["cart_count"]);
            }
            while ($row = mysqli_fetch_array($result));

            $itogpricecart = $int;
        }

    }
    else
    {

        $result = mysqli_query($link, "SELECT * FROM cart,
table_products WHERE cart.cart_ip = '{$_SERVER['REMOTE_ADDR']}' AND
table_products.products_id = cart.cart_id_products");
    }
}

```

```

If (mysqli_num_rows($result) > 0)
{
$row = mysqli_fetch_array($result);

do
{
$int = $int + ($row["cart_price"] * $row["cart_count"]);
}
while ($row = mysqli_fetch_array($result));

    $itogpricecart = $int;
}
}

switch ($action) {

    case 'oneclick':

        echo '
        <div id="block-step">
        <div id="name-step">
        <ul>
        <li><a class="active" >1. </a></li>
        <li><span>&rarr;</span></li>
        <li><a>2. Контактная информация</a></li>
        <li><span>&rarr;</span></li>
        <li><a>3. Завершение</a></li>
        </ul>
        </div>
        <p>шаг 1 из 3</p>
        <a href="cart&action=clear" >Очистить</a>
        </div>
        ';

        if ($_SESSION['auth'] == 'yes_auth')
        {
            $result = mysqli_query($link, "SELECT * FROM cart,
table_products, color WHERE cart.login_id =
'{$_SESSION['auth_login']}' AND table_products.products_id =
cart.cart_id_products AND cart.cart_color=color.color_name");
        }

        else {

            $result = mysqli_query($link, "SELECT * FROM cart,
table_products, color WHERE cart.cart_ip = '{$_SERVER['REMOTE_ADDR']}'
AND table_products.products_id = cart.cart_id_products AND
cart.cart_color=color.color_name");
            if (mysqli_num_rows($result)> 0)
            {
                $row = mysqli_fetch_array($result);

            echo '
            <div id="header-list-cart">
            <div id="head1">Изображение</div>
            <div id="head2">Наименование товара</div>

```

```

        <div id="head3">Цвет</div>
        <div id="head4">Размер</div>
        <div id="head5">Кол-во</div>
        <div id="head6">Цена</div>
    </div>
    <div class="hr"></div>
    ';

    do {
        $int = $row["cart_price"] * $row["cart_count"];
        $all_price = $all_price + $int;
        $cp = mysqli_query($link, "SELECT * FROM uploads_images WHERE
products_id='{ $row["products_id"] }' AND main=1 LIMIT 1 ");
        $colr=mysqli_fetch_array($cp);
        echo '

        <div class="block-list-cart">

            <div class="img-cart">
            </div>

            <p class="cart-title"><a
href="view_content.php?id=' . $row["products_id"] . '">' . $row["title"] . '</
a></p>

            <p class="cart-color"><label for="' . $row["color_id"] . '"
style="background-color:' . $row["color_icon"] . '; margin-left:0"
class="color_lable"></label></p>
            <p class="cart-size">' . $row["cart_size"] . '</p>

            <div class="count-cart">
            <ul class="input-count-style">

                <li>
                    <p align="center" iid="' . $row["cart_id"] . '" class="count-
minus">-</p>

                </li>

                <li>
                    <p align="center"><input id="input-id' . $row["cart_id"] . '"
iid="' . $row["cart_id"] . '" class="count-input" maxlength="3"
type="text" value="' . $row["cart_count"] . '"></p>
                </li>

                <li>
                    <p align="center" iid="' . $row["cart_id"] . '" class="count-
plus">+</p>
                </li>
            </ul>

            </div>

            <div id="tovar' . $row["cart_id"] . '" class="price-
product"><h5><span class="span-count" >' . $row["cart_count"] . '</span> x
<span>' . $row["cart_price"] . '</span></h5><p
price="' . $row["cart_price"] . '" >' . $int . ' грн</p></div>

```

```

        </div><div class="delete-cart"><a
href="cart.php?id='.$row["cart_id"].'&action=delete"></a></div>

';
    }
    while ($row = mysqli_fetch_array($result));
    echo '

        <h2 class="itog-price">Итого: '.$all_price.' грн </h2>
        <p class="button-next"><a href="cart&action=confirm"> Далее
</a></p>
    ';

}
else
{echo '<h3 id ="clear-cart" align="center"> </h3>';}

break;
    case 'confirm':

        echo '
        <div id="block-step">
        <div id="name-step">
        <ul>
        <li><a href="cart.php?action=oneclick" ></a></li>
        <li><span>&rarr;</span></li>
        <li><a class="active" >2. Контактная информация</a></li>
        <li><span>&rarr;</span></li>
        <li><a>3. Завершение</a></li>
        </ul>
        </div>
        <p>шаг 2 из 3</p>

        </div>

    ';

    if ($_SESSION['order_delivery'] == "Наложенный платёж") $chck1 =
"checked";
    if ($_SESSION['order_delivery'] == "Безналичный расчет") $chck2
= "checked";

    if ( $_SESSION['auth'] == 'yes_auth' )
    {

echo '

        <h3 class="title-h3-2" >Способы оплаты:</h3>
        <form method="post" action="cart&action=sucsess" id="form">
        <ul id="info-radio">
        <li>
        <input type="radio" name="order_delivery" class="order_delivery"
id="order_delivery1" required value="Наложенный платёж" '.$chck1.' />
        <label class="label_delivery" for="order_delivery1">Наложенный
платёж</label></li>

```

```

        <li><input type="radio" name="order_delivery"
class="order_delivery" id="order_delivery2" required
value="Безналичный расчет" '.$schck2.' />
        <label class="label_delivery" for="order_delivery2">Безналичный
расчет</label>
    </li>
</ul>
<h3 class="title-h3-2" >Информация для доставки:</h3>
<ul id="info-order">';
        $result = mysqli_query($link, "SELECT * FROM cart,
table_products WHERE cart.login_id = '{$_SESSION['auth_login']}' AND
table_products.products_id = cart.cart_id_products");
        if (mysqli_num_rows($result) > 0)
        {
            $row = mysqli_fetch_array($result);
            do{
                $int = $row["cart_price"] * $row["cart_count"];
                $all_price = $all_price + $int;
            echo '
                <input type="hidden" name="order_product"
value="'.$row["title"].'">
                <input type="hidden" name="order_count"
value="'.$row["cart_count"].'">
                <input type="hidden" name="order_color"
value="'.$row["cart_color"].'">
                <input type="hidden" name="order_size"
value="'.$row["cart_size"].'">
                <input type="hidden" name="order_price"
value="'.$row["cart_price"].'">';}
                while ($row = mysqli_fetch_array($result));
            }
            echo '
            <li><label for="order_fio"><span>*</span>ФИО</label><input
required type="text" name="order_fio" id="order_fio"
value="'.$_SESSION['auth_surname'].' '.$_SESSION['auth_name'].'
'.$_SESSION['auth_patronymic'].'" /><span class="order_span_style"
>Пример: Иванов Иван Иванович</span></li>
            <li><label for="order_email"><span>*</span>E-mail</label><input
required type="email" name="order_email" id="order_email"
value="'.$_SESSION["auth_email"].'" /><span class="order_span_style"
>Пример: ivanov@mail.ru</span></li>
            <li><label for="order_phone"><span>*</span>Телефон</label><input
required type="tel" name="order_phone" id="order_phone"
value="'.$_SESSION['auth_phone'].'" /><span class="order_span_style"
>Пример: +380501041000</span></li>
            <li><label class="order_label_style"
for="order_address"><span>*</span>Адрес<br /> доставки</label><input
required type="text" name="order_address" id="order_address"
value="'.$_SESSION["order_address"].'" /><span>Пример: г.
Запорожье,<br /> пр. Соборный д 158, кв 20</span></li>
            <li><label class="order_label_style"
for="order_note">Примечание</label><textarea name="order_note"
>'.$_SESSION["order_note"].'</textarea><span>Уточните информацию о
заказе.<br /> Например, удобное время для звонка<br /> нашего
менеджера</span></li>
            <input type="hidden" name="order_full_price"
value="'.$all_price.'">
        </ul>

```

```

        <button type="submit" class="button-pay">Подтвердить </button>
    </ul>
</form>

';}
else{

    echo '
    <h3 class="title-h3-2" >Способы оплаты:</h3>
    <form method="post" action="cart&action=sucsess" id="form">
    <ul id="info-radio">
    <li>
        <input type="radio" name="order_delivery" class="order_delivery"
id="order_delivery1" value="Наложенный платёж" '.$schck1.' />
        <label class="label_delivery" for="order_delivery1">Наложенный
платёж</label>
    </li>
    <li>
        <input type="radio" name="order_delivery" class="order_delivery"
id="order_delivery2" value="Безналичный расчёт" '.$schck2.' />
        <label class="label_delivery" for="order_delivery2">Безналичный
расчёт</label>
    </li>
    </ul>
    <h3 class="title-h3-2" >Информация для доставки:</h3>
    <ul id="info-order">

';

    $result = mysqli_query($link, "SELECT * FROM cart,
table_products WHERE cart.cart_ip = '{$_SERVER['REMOTE_ADDR']}' AND
table_products.products_id = cart.cart_id_products");
    if (mysqli_num_rows($result) > 0)
    {
        $row = mysqli_fetch_array($result);
        do{
            $int = $row["cart_price"] * $row["cart_count"];
            $all_price = $all_price + $int;
            echo '
            <input type="hidden" name="order_product"
value="'.$row["title"].'">
            <input type="hidden" name="order_color"
value="'.$row["cart_color"].'">
            <input type="hidden" name="order_size"
value="'.$row["cart_size"].'">
            <input type="hidden" name="order_price"
value="'.$row["cart_price"].'">';}
            while ($row = mysqli_fetch_array($result));
        }

        echo '
        <li><label for="order_fio"><span>*</span>ФИО</label><input
required type="text" name="order_fio" id="order_fio"
value="'.$_SESSION["order_fio"].'" /><span class="order_span_style"
>Пример: Иванов Иван Иванович</span></li>
        <li><label for="order_email"><span>*</span>E-mail</label><input
required type="email" name="order_email" id="order_email"

```

```

value="'.$_SESSION["order_email"].'" /><span class="order_span_style"
>Пример: ivanov@mail.ru</span></li>
    <li><label for="order_phone"><span>*</span>Телефон</label><input
required type="tel" name="order_phone" id="order_phone"
value="'.$_SESSION["order_phone"].'" /><span class="order_span_style"
>Пример: +380501041000</span></li>
    <li><label class="order_label_style"
for="order_address"><span>*</span>Адрес<br /> доставки</label><input
required type="text" name="order_address" id="order_address"
value="'.$_SESSION["order_address"].'" /><span>Пример: г.
Запорожье,<br /> пр. Соборный д 158, кв 20</span></li>
    <li><label class="order_label_style"
for="order_note">Примечание</label><textarea name="order_note"
>'.$_SESSION["order_note"].'</textarea><span>Уточните информацию о
заказе.<br /> Например, удобное время для звонка<br /> нашего
менеджера</span></li>
    <input type="hidden" name="order_full_price"
value="'.$_SESSION["order_full_price"].'" />
</ul>
    <button type="submit" class="button-pay">Подтвердить </button>
</form>';}

break;

case 'success':

    echo '
        <div id="block-step">
        <div id="name-step">
        <ul>
        <li><a>1. </a></li>
        <li><span>&rarr;</span></li>
        <li><a>2. </a></li>
        <li><span>&rarr;</span></li>
        <li><a class="active">3 </a></li>
        </ul>
        </div>
        <p>шаг 3 из 3</p>

        <h3 id="success-cart" align="center">Спасибо за!</h3>
        </div>';
    break;

default:
echo '
    <div id="block-step">
    <div id="name-step">
    <ul>
    <li><a class="active">1. </a></li>
    <li><span>&rarr;</span></li>
    <li><a>2. </a></li>
    <li><span>&rarr;</span></li>
    <li><a>3. </a></li>
    </ul>
    </div>
    <p>шаг 1 из 3</p>
    <a href="cart?action=clear" ></a>
    </div>

```

```

';

    if ($_SESSION['auth'] == 'yes_auth')
    {
        $result = mysqli_query($link, "SELECT * FROM cart,
table_products, color WHERE cart.login_id =
'{$_SESSION['auth_login']}' AND table_products.products_id =
cart.cart_id_products AND cart.cart_color=color.color_name");
    }

    else {

        $result = mysqli_query($link, "SELECT * FROM cart,
table_products, color WHERE cart.cart_ip = '{$_SERVER['REMOTE_ADDR']}'
AND table_products.products_id = cart.cart_id_products AND
cart.cart_color=color.color_name");}
        if (mysqli_num_rows($result)> 0)
        {
            $row = mysqli_fetch_array($result);

echo '
    <div id="header-list-cart">
    <div id="head1">Изображение</div>
    <div id="head2">Наименование товара</div>
    <div id="head3">Цвет</div>
    <div id="head4">Размер</div>
    <div id="head5">Кол-во</div>
    <div id="head6">Цена</div>
    </div>
    <div class="hr"></div>
    ';

        do {
            $int = $row["cart_price"] * $row["cart_count"];
            $all_price = $all_price + $int;
            $cp = mysqli_query($link, "SELECT * FROM uploads_images WHERE
products_id='{ $row["products_id"] }' AND main=1 LIMIT 1 ");
                                $colr=mysqli_fetch_array($cp);

            echo '

    <div class="block-list-cart">

        <div class="img-cart">
        </div>

        <p class="cart-title"><a
href="view_content.php?id=' . $row["products_id"] . '">' . $row["title"] . '</
a></p>

        <p class="cart-color"><label for="' . $row["color_id"] . '"
style="background-color:' . $row["color_icon"] . '"; margin-left:0"
class="color_lable"></label></p>
        <p class="cart-size">' . $row["cart_size"] . '</p>

        <div class="count-cart">
        <ul class="input-count-style">

```

```

        <li>
            <p align="center" iid="'.$row["cart_id"].'" class="count-
minus">-</p>

        </li>

        <li>
            <p align="center"><input id="input-id'.'.$row["cart_id"].'"
iid="'.$row["cart_id"].'" class="count-input" maxlength="3"
type="text" value="'.$row["cart_count"].'"></p>
        </li>

        <li>
            <p align="center" iid="'.$row["cart_id"].'" class="count-
plus">+</p>
        </li>
    </ul>

</div>

    <div id="tovar'.'.$row["cart_id"].'" class="price-
product"><h5><span class="span-count" >'.'.$row["cart_count"].'</span> x
<span>'.'.$row["cart_price"].'</span></h5><p
price="'.$row["cart_price"].'" >'.'.$int.' грн</p></div>

</div><div class="delete-cart"><a
href="cart.php?id='.'.$row["cart_id"].'&action=delete"></a></div>

';
    }
    while ($row = mysqli_fetch_array($result));
    echo '

        <h2 class="itog-price">Итого: '.'.$all_price.' грн </h2>
        <p class="button-next"><a href="cart&action=confirm"> Далее
</a></p>
    ';

}
else
{echo '<h3 id ="clear-cart" align="center">
</h3>';}

break;
}

?>
</div>

<?php
include("templates/footer.php");
?>

```

ДОДАТОК Д
Слайди презентації

Дослідження та програмна реалізація інформаційно- аналітичної системи для вибору закладу харчування за бажаними критеріями

Виконав: ст. гр. КНТ-219м Є. С. Брильов

Керівник: доцент, к.т.н. О.О.Степаненко

Рисунок Д.1 – Слайд № 1

Об'єктом дослідження є процес розробки програмних засобів.

Предмет дослідження — програмні засоби для вибору та замовлення продуктів харчування.

Мета роботи — дослідження та реалізація програмних засобів для вибору харчування за бажаними критеріями.

Рисунок Д.2 – Слайд № 2

Задачі роботи

- аналіз предметної області, зокрема вебсайтів для вибору харчування за бажаними критеріями;
- розробка технічного завдання для реалізації програми;
- вибір засобів розробки програмного забезпечення;
- розробка інформаційної моделі програмних засобів для вибору харчування за бажаними критеріями;
- конструювання програмного забезпечення для вибору харчування за бажаними критеріями;
- тестування розробленого програмного забезпечення.

3

Рисунок Д.3 – Слайд № 3

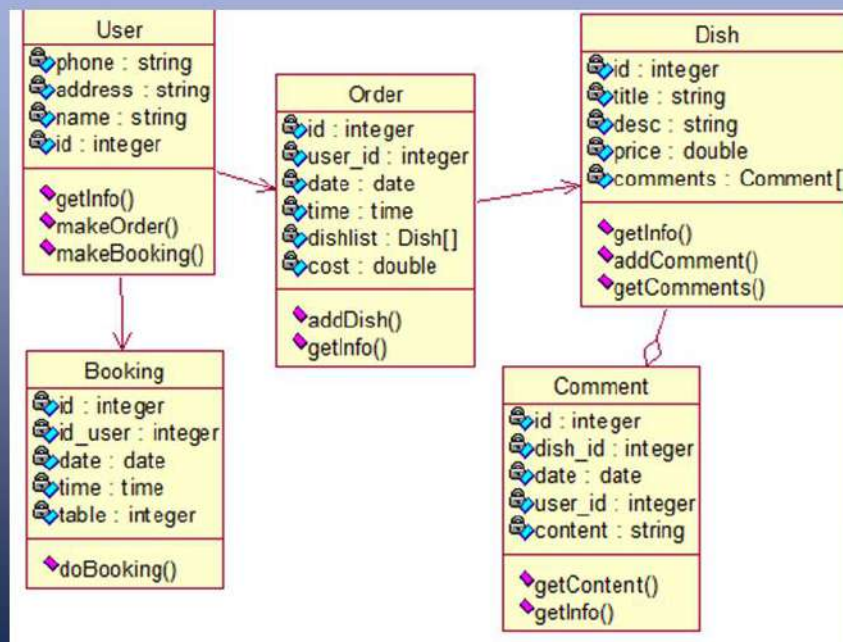
Аналіз програмних засобів для вибору та замовлення продуктів харчування

№ з/п	Критерій порівняння	Назва програми (сайту)			
		Tomato	Guide-cafe	HoReca	Booze
1	Реєстрація користувача	+	–	+	–
2	Наявність мапи для покращення навігації	–	–	–	–
3	Наявність мобільного додатку	+	–	–	+
4	Обширні можливості фільтрації	+	–	–	–
5	Якісне інформаційне оснащення	+	+	+	–
6	Можливість приймати участь в актуалізації контенту	–	+	+	–

4

Рисунок Д.4 – Слайд № 4

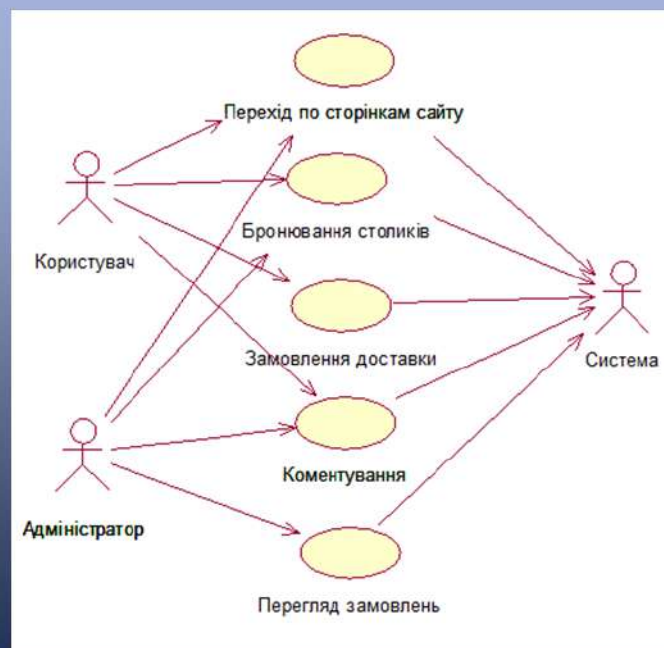
Модель системи у вигляді діаграми класів



5

Рисунок Д.5 – Слайд № 5

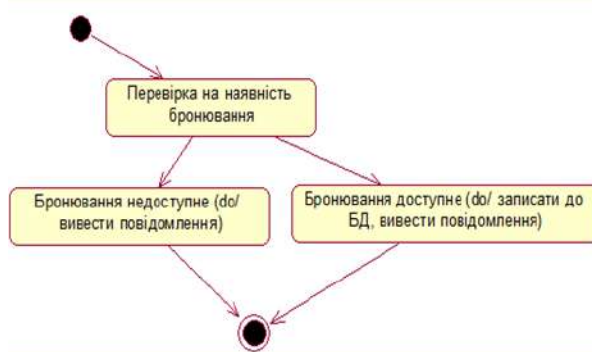
Моделі системи у вигляді діаграм прецедентів (варіантів використання)



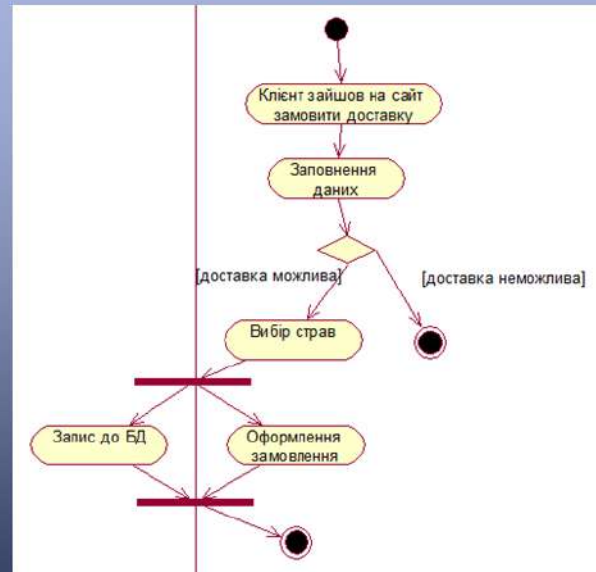
6

Рисунок Д.6 – Слайд № 6

Моделі системи у вигляді діаграми станів та діаграми діяльності



Діаграма станів

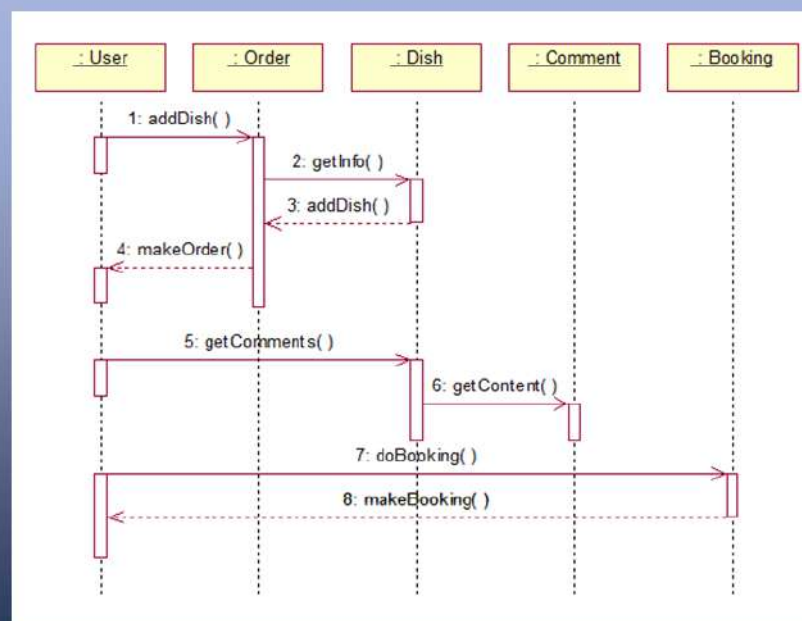


Діаграма діяльності

7

Рисунок Д.7 – Слайд № 7

Моделі системи у вигляді діаграми послідовності



8

Рисунок Д.8 – Слайд № 8

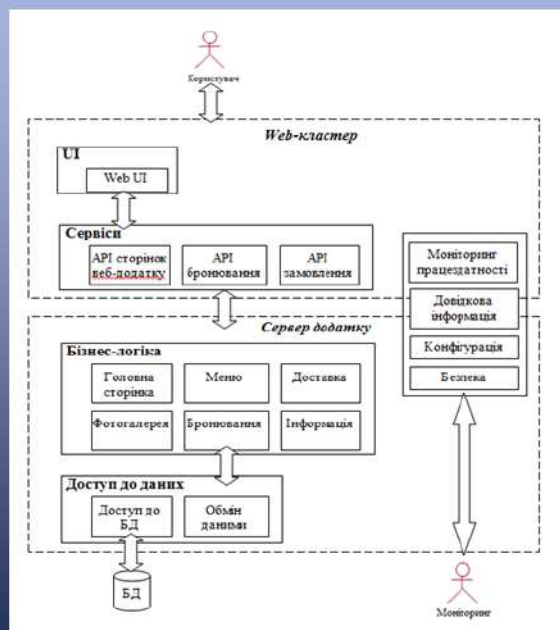
Вибір мови програмування

Критерій порівняння	C++	Java	PHP
Підтримка імперативної парадигми програмування	+	+	+
Зручність для створення вебсайтів	+ -	+	+
Наявність засобів для розробки клієнт-серверного програмного забезпечення	+ -	+	+
Зручність використання для створення webзастосунків	+ -	+	+
Зручність використання для створення клієнт-серверного програмного забезпечення	+ -	+ -	+
Кросплатформність	+	+	+

9

Рисунок Д.9 – Слайд № 9

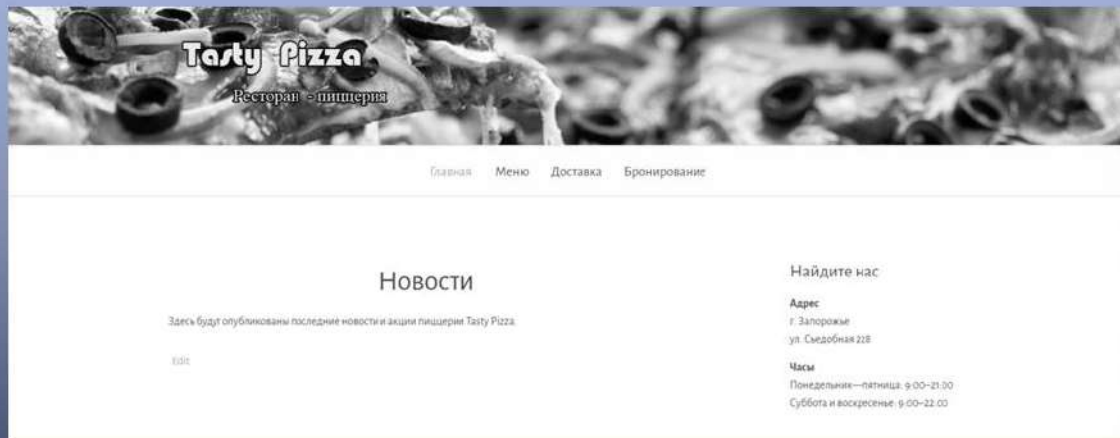
Архітектура програмного забезпечення



10

Рисунок Д.10 – Слайд № 10

Інтерфейс системи. Головна сторінка сайту



11

Рисунок Д.11 – Слайд № 11

Інтерфейс системи. Сторінка вибору продукції



12

Рисунок Д.12 – Слайд № 12

Інтерфейс системи. Сторінка бронювання

Бронирование

Забронируйте столики!

Заказать стол

Дата
декабря 5, 2020

Время

Критерии
1

Контактная информация

Имя

Адрес электронной почты

Телефон

[Добавить сообщение](#)

13

Рисунок Д.13 – Слайд № 13

Тестування програмного забезпечення

Тестування сумісності

Критерій оцінювання	Значення критерію	Відповідність значення критерію технічному завданню
Наявність меню	Так	Так
Можливість забронювати столики	Так	Так
Можливість користування багатьма Користувачами одночасно	Так	Так
Інтерфейс системи на заданій мові	Так	Так
Наявність графічного інтерфейсу	Так	Так

Протокол результатів тестування функціоналу програми

Критерій оцінювання	Значення критерію	Відповідність значення критерію технічному завданню
Наявність меню	Так	Так
Можливість забронювати столики	Так	Так
Можливість користування багатьма користувачами одночасно	Так	Так
Інтерфейс системи на заданій мові	Так	Так
Наявність графічного інтерфейсу	Так	Так

14

Рисунок Д.14 – Слайд № 14

Висновки

У ході виконання дипломної кваліфікаційної роботи магістра було проаналізовано предметну область. Визначено, що відомі сервіси з вибору та замовлення харчування мають певні недоліки, зокрема, вони часто є досить застарілими (мають застарілий і не зовсім зручний інтерфейс), як правило, є орієнтованими під конкретний заклад харчування та не можуть бути модифікованими для роботи з іншими закладами.

Наукова новизна роботи полягає у тому, що розроблено інформаційну модель програмних засобів для вибору харчування за бажаними критеріями. Розроблена інформаційна модель подана у вигляді набору UML-діаграм та дозволяє створювати та модифікувати програмне забезпечення для вибору харчування за бажаними критеріями.

Для розробки програмного забезпечення для вибору харчування за бажаними критеріями обрано мову PHP, оскільки така мова дозволяє створювати вебсайти, що забезпечують ефективну взаємодію з користувачем.

Виконано тестування розробленого програмного забезпечення. Результати тестування підтвердили працездатність розробленого програмного забезпечення для вибору харчування за бажаними критеріями, а також можливість його використання у різних веббраузерах. Виявлено, що функціонал програми, її ергономічність та функції безпеки відповідають технічному завданню.

15

Рисунок Д.15 – Слайд № 15