

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет комп'ютерних наук і технологій

(повне найменування факультету)

Кафедра програмних засобів

(повне найменування кафедри)

Пояснювальна записка

до дипломного проекту (роботи)

бакалавр

(ступінь вищої освіти)

на тему РОЗРОБКА ВЕБСАЙТУ «ПОМІЧНИК ДОМОГОСПОДАРКИ»
DEVELOPMENT OF THE HOUSEWIFE'S HELPER WEBSITE

Виконав(ла): студент(ка) 4 курсу, групи КНТ-129

Спеціальності 121 Інженерія програмного

(код і найменування спеціальності)

забезпечення

Освітня програма (спеціалізація)

Інженерія програмного забезпечення

ХИЖНЯК Д.В.

(ПРІЗВИЩЕ та ініціали)

Керівник СТЕПАНЕНКО О.О.

(ПРІЗВИЩЕ та ініціали)

Рецензент ГОЛУБ Т.В.

(ПРІЗВИЩЕ та ініціали)

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

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

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ, д.т.н, проф.
Сергій СУББОТІН
 “ ” 2023 року

З А В Д А Н Н Я

НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

ХИЖНЯКА Данило Валерійовича
(ПРІЗВИЩЕ, ім'я, по батькові)

1. Тема проєкту (роботи) Розробка вебсайту «Помічник домогосподарки».
Development of the Housewife`s Helper Website.
 керівник проєкту (роботи) к.т.н., доцент, СТЕПАНЕНКО Олександр
Олексійович,

(науковий ступінь, вчене звання, ПРІЗВИЩЕ, ім'я, по батькові)

затверджені наказом закладу вищої освіти від “04” квітня 2022 року № 87
 2. Строк подання студентом проєкту (роботи) 01 червня 2022 року
 3. Вихідні дані до проєкту (роботи) рекомендована література, технічне
завдання

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз проблеми та постановка завдань розробки.
2. Матеріали і методи розробки. 3. Опис вебсайту. 4. Оптимізація і тестування вебсайту.

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

6. Консультанти розділів проєкту (роботи)

Розділ	ПРИЗВИЩЕ, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-4 Основна частина	СТЕПАНЕНКО О.О., доцент		
Нормоконтроль	БЄЛОВА А.В., асистент		

7. Дата видачі завдання « 04 » квітня 2023 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Постановка завдання роботи.	1 тиждень	Завдання, ТЗ
2	Аналіз предметної області.	1 тиждень	Розділ 1
3	Вибір мови програмування та інших технологій розробки.	2 тиждень	Розділ 2
4	Розробка бази даних.	2 тиждень	Розділ 3
5	Розробка керівництва програміста та користувача.	3-4 тижні	Розділ 3, 4
6	Тестування та оптимізація вебсайту	5 тиждень	Розділ 4
7	Оформлення пояснювальної записки та документів до неї.	6 тиждень	Додатки
8	Нормоконтроль та рецензування.	7 тиждень	
9	Захист роботи.	8 тиждень	

Студент(ка)

_____ Данило ХИЖНЯК
(підпис) (Ім'я ПРИЗВИЩЕ)

Керівник проєкту (роботи)

_____ Олександр СТЕПАНЕНКО
(підпис) (Ім'я ПРИЗВИЩЕ)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи бакалавра: 64 с., 13 табл., 45 рис., 1 дод., 20 джерел.

ВЕБСАЙТ, БАЗА ДАНИХ, ПОМІЧНИК ДОМОГОСПОДАРКИ, РЕЦЕПТИ, СТРАВИ, ПОШУК, АНАЛІЗ, НОРМАЛІЗАЦІЯ, HTML, CSS, JAVA SCRIPT, MYSQL, PHP, BOOTSTRAP, ANGULAR.

Об'єкт дослідження – програмні засоби для домогосподарок.

Предмет дослідження – вебсайт помічник для домогосподарок.

Мета роботи – створення вебсайту помічника для домогосподарок із функціями публікації та пошуку рецептів.

Матеріали, методи та технічні засоби: структурне та об'єктно-орієнтоване програмування, мови програмування HTML, CSS, JAVA SCRIPT, MYSQL, PHP, BOOTSTRAP, ANGULAR, персональний комп'ютер з процесором Intel Core 2 Duo під управлінням операційної системи Microsoft Windows 10.

Результати. Створено вебсайт із книгою рецептів із функціями соціальної мережі. Домогосподарки можуть публікувати свої рецепти, знаходити та оцінювати вже збережені рецепти.

Висновки. Розроблений вебсайт допомагає і автоматизує функції збереження рецептів та поширення їх серед інших.

Галузь використання – помічник домогосподарок.

ABSTRACT

Explanatory note to the diploma qualifying work of the bachelor: 64 pages, 13 tables, 45 figures, 1 appendix, 20 sources.

WEBSITE, DATABASE, HOUSEHOLD ASSISTANT, RECIPES, DISHES, SEARCH, ANALYSIS, NORMALIZATION, HTML, CSS, JAVA SCRIPT, MYSQL, PHP, BOOTSTRAP, ANGULAR.

Object of study is a software for housewives.

Subject of study are a website for helpers for housewives.

The purpose of the work is to create an assistant website for housewives with the functions of publishing and searching for recipes.

Materials, methods and technical means: structural and object-oriented programming, programming languages HTML, CSS, JAVA SCRIPT, MYSQL, PHP, BOOTSTRAP, ANGULAR, a personal computer with an Intel Core 2 Duo processor running the Microsoft Windows 10 operating system .

Results. Created a recipe book website with social networking features. Housewives can publish their recipes, find and rate already saved recipes.

Conclusions The developed website helps and automates functions of saving recipes and sharing them with others.

The field of use is an assistant to housewives.

ЗМІСТ

	С.
Перелік скорочень та умовних познач.....	7
Вступ.....	8
1 Аналіз проблеми та постановка завдань розробки.....	10
1.1 Аналіз моделей розробки вебсайтів.....	10
1.2 Аналіз систем помічників для домогосподарок	12
1.3 Постановка завдань розробки.....	18
1.4 Висновки за розділом 1	21
2 Матеріали і методи розробки.....	23
2.1 Вибір методів розробки вебсайту.....	23
2.2 Вибір мов програмування вебсайту	27
2.3 Вибір бази даних	31
2.4 Висновки за розділом 2	35
3 Опис вебсайту.....	37
3.1 Керівництво програміста.....	37
3.2 Керівництво користувача	40
3.3 Висновки за розділом 3	46
4 Оптимізація і тестування вебсайту.....	48
4.1 Оптимізація вебсайту.....	48
4.2 Тестування вебсайту	49
4.3 Висновки за розділом 4	51
Висновки	52
Перелік джерел посилань	53
Додаток А Слайди презентації.....	55

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

API	– Application Programming Interface;
CSS	– Cascading Style Sheets;
HTML	– HyperText Markup Language;
IDE	– Integrated Development Environment;
IT	– Information technology;
JS	– Java Script;
MVC	– Model-View-Controller;
SEO	– Search Engine Optimization;
IoT	– Internet of Things;
ШІ	– Штучний інтелект.

ВСТУП

За останні роки штучний інтелект (ШІ) досяг значного прогресу, дозволяючи машинам виконувати складні завдання та розумно приймати рішення. Цю технологію можна використовувати для автоматизації повторюваних завдань, надання персоналізованих рекомендацій і покращення загальної взаємодії з користувачем [1].

Інтернет речей (IoT) стає все більш популярним, об'єднуючи фізичні пристрої з датчиками та програмним забезпеченням. Інтеграція пристроїв IoT на вебсайті може запропонувати домогосподаркам більшу зручність і автоматизацію в управлінні своїми будинками, наприклад, контролювати розумну побутову техніку та контролювати споживання енергії [2].

Хмарні обчислення революціонізували спосіб розробки та розміщення вебсайтів. Вони забезпечують доступ за вимогою до обчислювальних ресурсів, полегшуючи зберігання та доступ до даних, запуск програм і масштабування ресурсів за потреби. Використовуючи хмарні технології, вебсайт може бути дуже гнучким, масштабованим і економічно ефективним [3].

Мобільні технології стали невід'ємною частиною нашого повсякденного життя. Мобільні програми та адаптивний вебдизайн мають вирішальне значення для охоплення широкої бази користувачів і надання персоналізованого досвіду. Забезпечення зручності вебсайту для мобільних пристроїв дозволить користувачам мати доступ до нього в будь-який час і в будь-якому місці за допомогою своїх смартфонів або планшетів [4].

Актуальність цих IT-розробок полягає в тому, що вони дозволяють домогосподаркам підвищити свою продуктивність, спростити завдання та покращити загальну якість життя. Використовуючи штучний інтелект, Інтернет речей, хмарні обчислення та мобільні технології, вебсайт може забезпечувати автоматизацію, що економить час, доступ до цінної інформації та ресурсів [5].

Актуальність створення даної системи має ряд переваг. По-перше, вона пропонує цільову підтримку, адаптовану до конкретних потреб домогосподарок, надаючи їм індивідуальні рішення та ресурси. По-друге, це забезпечує зручність і доступність, дозволяючи домогосподаркам отримати доступ до вебсайту та його функцій у зручний для них час. По-третє, систему можна персоналізувати на основі індивідуальних уподобань і моделей використання, пропонуючи рекомендації та пропозиції, специфічні для кожного користувача. Нарешті, це дає домогосподаркам можливість контролювати свої обов'язки та надає можливості для особистого та професійного розвитку.

З точки зору подальшого розвитку, інтеграція системи з пристроями розумного будинку може покращити автоматизацію та безперебійний контроль над побутовими завданнями. Включення голосових помічників на основі обробки природної мови та технології розпізнавання голосу забезпечить роботу без рук і підвищить доступність. Крім того, використання методів аналізу даних може надати цінну інформацію про поведінку користувачів, дозволяючи безперервно вдосконалювати та вдосконалювати функції та функції системи.

Загалом, розвиток ІТ-технологій для створення вебсайтів пропонує величезний потенціал для підтримки домогосподарок і покращення їх повсякденного життя. Використовуючи ці досягнення, вебсайт може стати цінним ресурсом, що надає зручність, знання та розширення можливостей для домогосподарок у всьому світі [1].

1 АНАЛІЗ ПРОБЛЕМИ ТА ПОСТАНОВКА ЗАВДАНЬ РОЗРОБКИ

1.1 Аналіз моделей розробки вебсайтів

Моделі розробки описують процес і підхід до створення вебсайтів. Існують різні типи моделей розробки вебсайтів, кожна зі своїми характеристиками та компонентами. Ось деякі моделі, які часто використовуються [6].

Модель Waterfall використовує лінійний і послідовний підхід до розробки вебсайту [7]. Він складається з наступних основних компонентів:

- збір вимог: збір і документування вимог клієнтів і цілей проекту;
- дизайн: створення візуального та функціонального дизайну сайту;
- розробка: написання коду та розробка вебсайту на основі специфікації дизайну;
- тестування: проведення тестування та забезпечення якості для забезпечення належного функціонування вебсайту;
- розгортання: запуск вебсайту після остаточного тестування та затвердження клієнтом.
- технічне обслуговування: постійна підтримка, оновлення та виправлення помилок.

До переваг цього методі слід віднести: чітка структура, легка для розуміння та керування, підходить для невеликих проектів із визначеними вимогами. Недоліки: обмежена гнучкість для змін під час процесу розробки, відсутність залучення клієнта до наступних етапів.

Модель Agile — це ітеративний і гнучкий підхід, який зосереджується на співпраці та адаптивності [8]. Його основні компоненти включають:

- планування спринту: визначення цілей і завдань для кожної ітерації розробки (спринту);
- розробка: розбивка проекту на менші завдання та поступова розробка веб-сайту;

- тестування: безперервне тестування та відгуки протягом усього процесу розробки;
- перегляд і ітерація: регулярний огляд прогресу веб-сайту, внесення коригувань і вдосконалень на основі відгуків;
- розгортання: розгортання веб-сайту поступово або після кожної ітерації;
- постійне технічне обслуговування: постійне оновлення та вдосконалення веб-сайту на основі відгуків користувачів і мінливих вимог.

Переваги даного методу: більша гнучкість для змін і вдосконалень, раннє залучення клієнтів, ітеративна розробка з постійним вдосконаленням. Недоліки: потребує більш частого спілкування та координації, може призвести до довшого терміну розробки складних проектів .

Модель Waterfall є традиційним, лінійним підходом, тоді як модель Agile є більш адаптивною та ітеративною. В таблиці 1.1 наведена порівняльна характеристика двох методів.

Таблиця 1.1 – Порівняльна характеристика двох методів

Параметр	Порівняння
Залучення клієнтів	Waterfall включає відгуки клієнтів переважно на пізніх етапах, тоді як Agile заохочує постійну участь клієнтів і зворотний зв'язок протягом усього процесу розробки.
Швидкість розробки	Waterfall слідує за послідовним процесом, що потенційно може призвести до більш тривалих циклів розробки. Agile забезпечує швидшу розробку за допомогою ітераційних випусків.
Документація	Waterfall наголошує на вичерпній документації з самого початку, тоді як Agile зосереджується більше на робочому програмному забезпеченні, а не на розширеній документації.

Продовження таблиці 1.1

Параметр	Порівняння
Управління ризиками	Послідовний підхід Waterfall допомагає виявляти й усувати ризики на ранній стадії, тоді як Agile керує ризиками за допомогою постійних ітерацій і частого зворотного зв'язку
Складність	Waterfall підходить для проектів із чітко визначеними вимогами, тоді як Agile краще підходить для складних проектів із змінними вимогами.

Зрештою, вибір між моделями Waterfall і Agile залежить від конкретних вимог проекту, уподобань клієнта та рівня гнучкості та адаптивності, бажаного в процесі розробки.

1.2 Аналіз систем помічників для домогосподарок

Існує велика кількість вебсайтів, які можуть бути корисними домогосподаркам у різних аспектах їхнього життя. Наведемо кілька прикладів.

Вебсайти планування їжі та рецептів:

– rud.ua: надає велику колекцію рецептів з оцінками та відгуками користувачів (рис. 1.1);

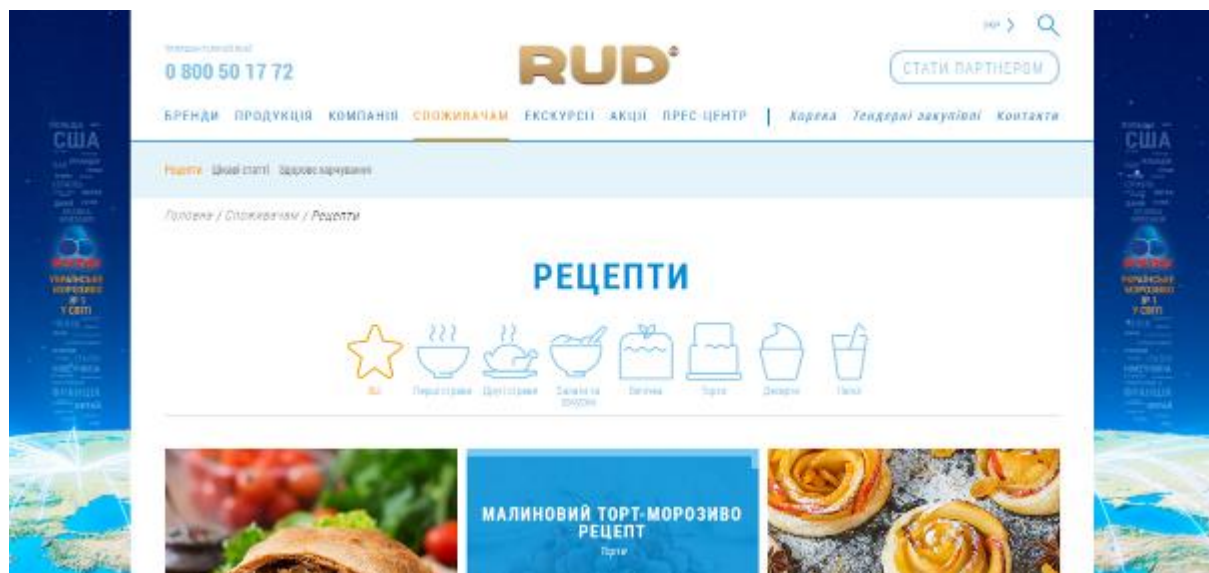


Рисунок 1.1 – Вебсайт рецептів [9]

– klorotenko.com пропонує рецепти, кулінарні поради та навчальні відео від популярних кухарів (рис. 1.2);

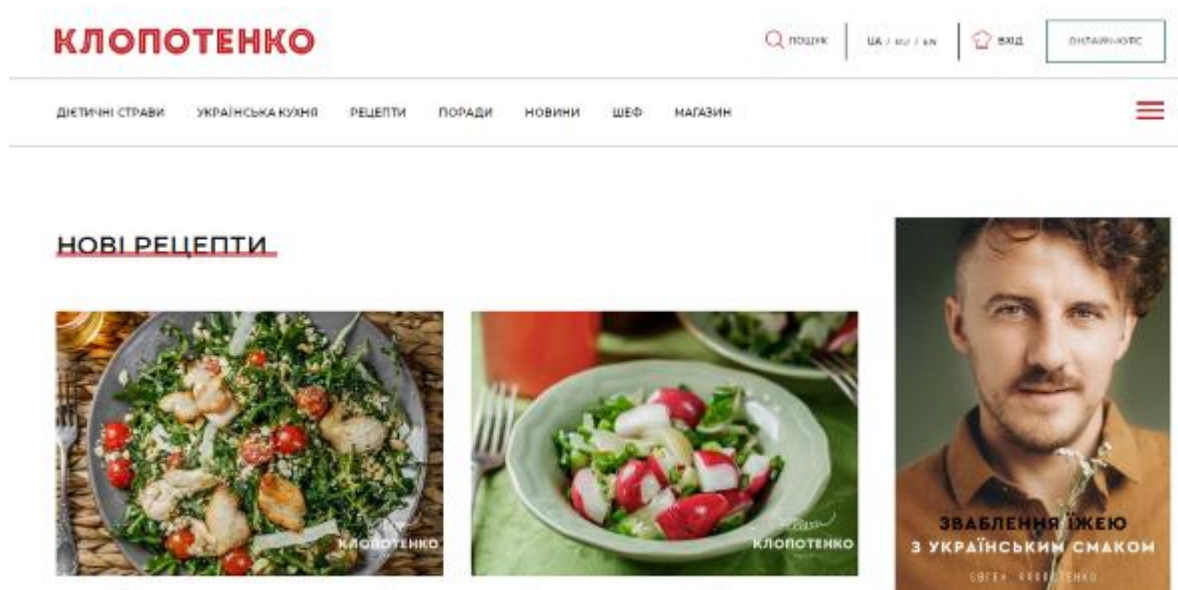


Рисунок 1.2 – Вебсайт klorotenko.com [10]

– shuba.life містить широкий вибір рецептів, інструментів для планування їжі та кулінарних посібників (рис. 1.3).

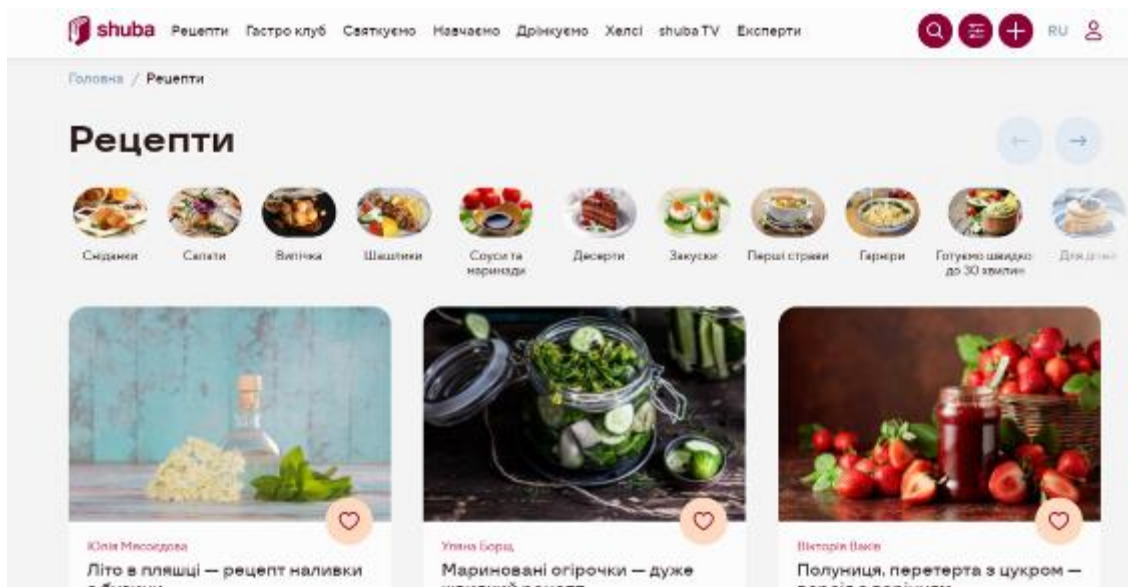


Рисунок 1.3 – Вебсайт shuba.life [11]

Сайти організації та прибирання дому:

– oprice.ua пропонує поради, інструменти та ресурси для організації дому, процедур прибирання та усунення сміття (рис.1.4);

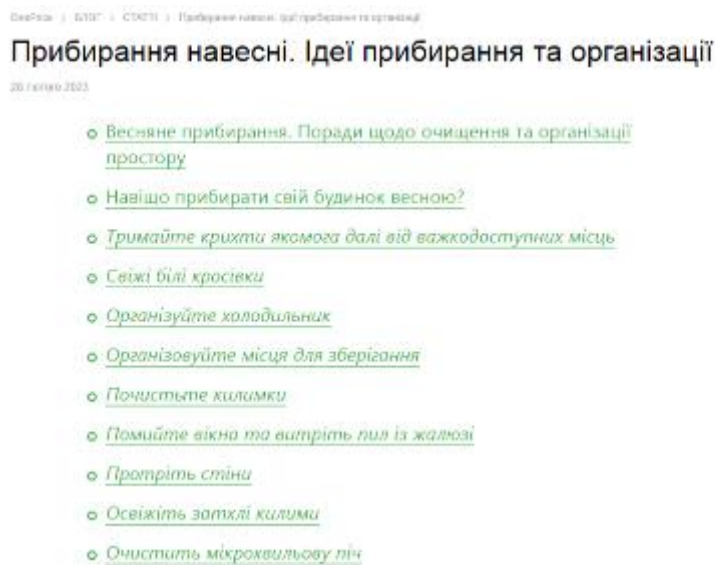


Рисунок 1.4 – Вебсайт operice.ua [12]

– the-village.com.ua надає посібники та поради щодо організації дому, прибирання та декору дому (рис.1.5);

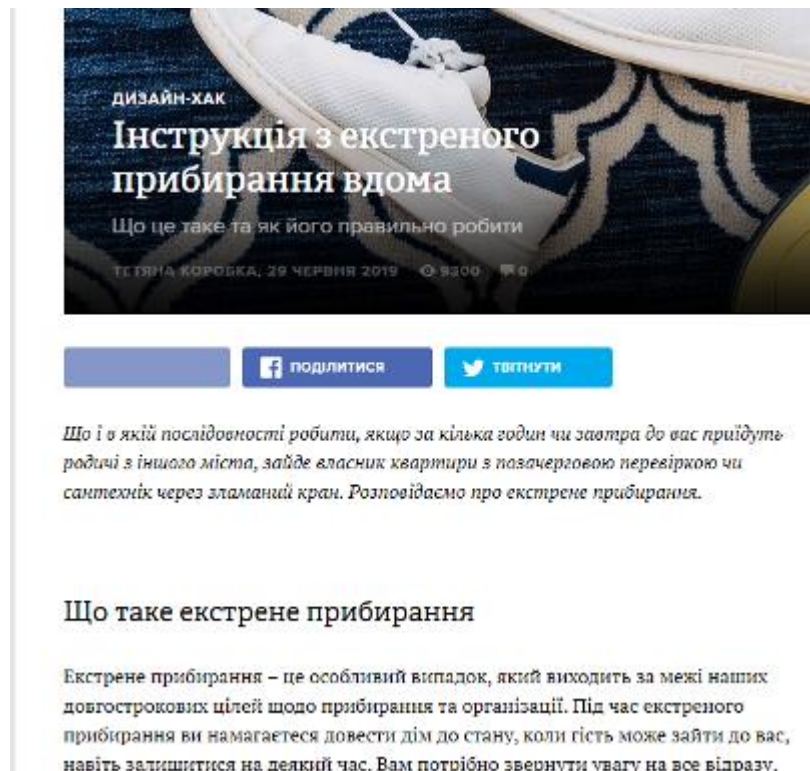


Рисунок 1.5 – Вебсайт the-village.com.ua [13]

– gospodarochka.in.ua пропонує практичні поради щодо прибирання будинку, організації та проектів (рис.1.6).



Рисунок 1.6 – Вебсайт gospodarochka.in.ua [14]

Веб-сайти бюджетування та фінансового менеджменту:

- homemoney допомагає складати бюджет, відстежувати витрати та ставити фінансові цілі (рис.1.7);

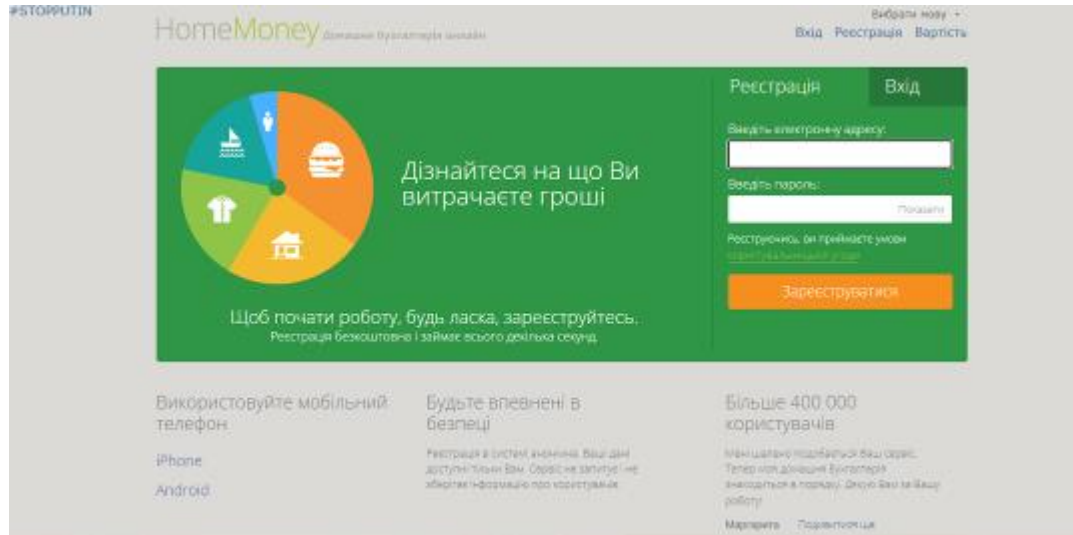


Рисунок 1.7 – Вебсайт homemoney [15]

- moneyhouse надає інструменти та ресурси для складання бюджету та управління особистими фінансами (рис.1.8).

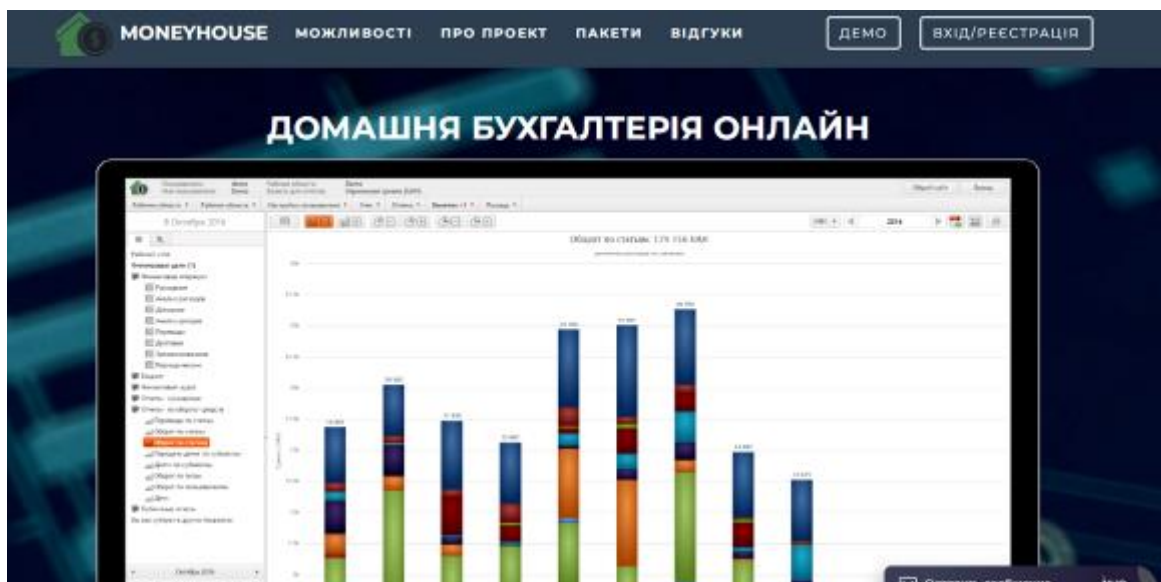


Рисунок 1.8 – Вебсайт moneyhouse [16]

Це лише кілька прикладів, і є багато інших вебсайтів, які задовольняють конкретні потреби та інтереси домогосподарок. Важливо досліджувати та знаходити вебсайти, які відповідають індивідуальним уподобанням і вимогам.

Враховуючи ці технічні параметри, домогосподарки можуть вибрати веб-помічників, які відповідають їхнім конкретним потребам і забезпечують оптимальний досвід користувача для керування нотатками, рецептами та коштами.

Порівняння розглянутих вебсайтів наведено в таблиці 1.2.

Таблиця 1.2 – Порівняння аналогів вебсистем помічника домогосподарки і власної розробки

Назва вебсайту	Призначення	Юзабіліті	Персоналізація	Актуальність	Відеоконтент	Відгуки	Пошук
rud.ua	Планування їжі та рецептів	+	-	+	+	-	+
klopotenko.com	Планування їжі та рецептів	+	+	+	+	+	+
shuba.life	Планування їжі та рецептів	+	+	+	-	-	+
oneprice.ua	Організації та прибирання дому	-	-	+	-	+	+
the-village.com.ua	Організації та прибирання дому	-	+	+	-	-	+
gospodarochka.in.ua	Організації та прибирання дому	+	-	+	-	-	+
homemoney	Бюджетування та фінансового менеджменту	+	+	+	-	-	+
moneyhouse	Бюджетування та фінансового менеджменту	+	+	+	-	-	-
Власна розробка homewife`s	Планування їжі та рецептів	+	+	+	-	+	+

1.3 Постановка завдань розробки

Розробка вебсайту помічника домогосподарки, зокрема функції книги рецептів, включає кілька ключових завдань, які буде виконано для забезпечення його успішного впровадження (табл. 1.3).

Таблиця 1.3 – Етапи розробки вебсайту

№	Назва етапу	Опис
1	Збір вимог	Це включає визначення здатності домогосподарок як записувати власні рецепти, так і шукати існуючі рецепти.
2	Дизайн	Фаза розробки інтерфейсу користувача зосереджена на створенні інтуїтивно зрозумілого та зручного інтерфейсу для розділу книги рецептів.
3	База даних	Буде розроблена та реалізована база даних для зберігання рецептів. Схема бази даних містить таблиці або колекції для зберігання деталей рецептів, таких як назва, інгредієнти, інструкції та інформація про автора.
4	Бекенд/Фронтенд	Розробка програмного коду інтерфейсу і функцій роботи із БД.
5	Тестування	Для тестування та гарантія якості буде проведено ретельне тестування, щоб переконатися в функціональності, зручності використання та ефективності функції книги рецептів.

Початковий крок включає збір вимог від потенційних користувачів, розуміння їхніх потреб і визначення бажаних функцій для розділу книги рецептів на вебсайті. Це включає визначення здатності домогосподарок як записувати власні рецепти, так і шукати існуючі рецепти.

Дизайн інтерфейсу користувача (UI). Фаза розробки інтерфейсу користувача зосереджена на створенні інтуїтивно зрозумілого та зручного інтерфейсу для розділу книги рецептів. Це включає розробку візуально привабливих карток або списків рецептів, впровадження параметрів пошуку та фільтрації та включення інтерактивних елементів для додавання та редагування рецептів.

Буде розроблена та реалізована база даних для зберігання рецептів. Схема бази даних містить таблиці або колекції для зберігання деталей рецептів, таких як назва, інгредієнти, інструкції та інформація про автора.

Розробка бекенда передбачає впровадження необхідних API та бізнес-логіки для виконання операцій, пов'язаних із рецептами. Це включає такі функції, як створення нового рецепту, отримання деталей рецепта, оновлення або видалення рецептів і керування профілями користувачів.

Розробка інтерфейсу зосереджена на створенні інтерфейсу користувача для розділу книги рецептів за допомогою таких веб-технологій, як HTML, CSS і JavaScript. Це передбачає створення адаптивних та інтерактивних сторінок із переліком рецептів, сторінок із детальною інформацією про рецепти та форм для додавання або редагування рецептів.

Для тестування та гарантія якості буде проведено ретельне тестування, щоб переконатися в функціональності, зручності використання та ефективності функції книги рецептів. Це включає в себе модульне тестування, інтеграційне тестування та тестування прийнятності користувачами для виявлення та усунення будь-яких проблем або помилок.

Розгортання та розміщення. Після завершення розробки та тестування вебсайт буде розгорнуто в середовищі хостингу, щоб зробити його доступним для користувачів через Інтернет. Це включає налаштування необхідної серверної інфраструктури, налаштування параметрів домену та DNS, а також забезпечення безпеки та масштабованості вебсайту.

Постійне технічне обслуговування та оновлення. Після початкової розробки виконувалося регулярне технічне обслуговування та оновлення для

вирішення будь-яких проблем, впровадження відгуків користувачів і впровадження нових функцій або вдосконалень у розділ книги рецептів.

Виконавши ці завдання, вебсайт помічника буде оснащено функцією книги рецептів, яка дозволяє господарям як записувати власні рецепти, так і легко знаходити рецепти, надані іншими, створюючи спільну та винахідливу платформу для кулінарного натхнення та обміну.

В таблиці 1.4 наведені загальні вимоги до розробляемого сайту.

Таблиця 1.4 – Загальні вимоги

Параметр	Опис
Інтерфейс користувача	Інтерфейс користувача має бути інтуїтивно зрозумілим і зручним, щоб домогосподаркам було легко орієнтуватися та ефективно керувати своїми завданнями. Чіткі та організовані макети з настоюваними параметрами є перевагою.
Доступність	Вебсайти мають бути доступні на різних пристроях і з різними розмірами екранів, щоб домогосподарки могли використовувати їх на своїх улюблених пристроях, таких як смартфони, планшети чи настільні комп'ютери.
Безпека даних	Враховуючи конфіденційний характер особистої інформації, заходи безпеки даних, як-от шифрування, надійна автентифікація та регулярне резервне копіювання, мають вирішальне значення для захисту даних користувачів
Інтеграція з іншими платформами	Інтеграція з популярними інструментами продуктивності, календарями та комунікаційними платформами може підвищити зручність використання вебсайту

Продовження таблиці 1.4

Параметр	Опис
Персоналізація	Можливість персоналізувати веб-сайт шляхом вибору тем, створення власних категорій або додавання власних полів може покращити взаємодію з користувачем і задовольнити індивідуальні вподобання.
Продуктивність і швидкість реагування	Вебсайти мають бути оптимізовані для швидкого завантаження та безперебійної роботи, забезпечуючи бездоганну взаємодію з користувачем

1.4 Висновки за розділом 1

Моделі розробки описують процес і підхід до створення вебсайтів. Існують різні типи моделей розробки вебсайтів, кожна зі своїми характеристиками та компонентами. Модель Waterfall є традиційним, лінійним підходом, тоді як модель Agile є більш адаптивною та ітеративною. Зрештою, вибір між моделями Waterfall і Agile залежить від конкретних вимог проекту, уподобань клієнта та рівня гнучкості та адаптивності, бажаного в процесі розробки.

Розробка вебсайту помічника домогосподарки, зокрема функції книги рецептів, включає кілька ключових завдань, які буде виконано для забезпечення його успішного впровадження. Початковий крок включає збір вимог від потенційних користувачів, розуміння їхніх потреб і визначення бажаних функцій для розділу книги рецептів на вебсайті. Фаза розробки інтерфейсу користувача зосереджена на створенні інтуїтивно зрозумілого та зручного інтерфейсу для розділу книги рецептів. Буде розроблена та реалізована база даних для зберігання рецептів. Розробка бекенда передбачає впровадження необхідних API та бізнес-логіки для виконання операцій, пов'язаних із рецептами. Для тестування та гарантія якості буде проведено ретельне тестування, щоб переконатися в функціональності, зручності

використання та ефективності функції книги рецептів. Після завершення розробки та тестування вебсайт буде розгорнуто в середовищі хостингу, щоб зробити його доступним для користувачів через Інтернет. Це включає налаштування необхідної серверної інфраструктури, налаштування параметрів домену та DNS, а також забезпечення безпеки та масштабованості вебсайту.

Виконавши ці завдання, вебсайт помічника буде оснащена функцією книги рецептів, яка дозволяє господарям як записувати власні рецепти, так і легко знаходити рецепти, надані іншими, створюючи спільну та винахідливу платформу для кулінарного натхнення та обміну.

2 МАТЕРІАЛИ І МЕТОДИ РОЗРОБКИ

2.1 Вибір методів розробки вебсайту

Система управління контентом (CMS) — це програмний додаток, який дозволяє користувачам створювати, керувати та змінювати цифровий вміст на вебсайті, не вимагаючи глибоких знань програмування [9]. Він забезпечує зручний інтерфейс і готові шаблони, теми та плагіни для спрощення розробки веб-сайту та керування вмістом.

Переваги CMS:

- зручність - платформи CMS зазвичай мають інтуїтивно зрозумілі інтерфейси, що полегшує керування вмістом для нетехнічних користувачів;
- швидка розробка - платформи CMS надають готові шаблони та компоненти, що дозволяє швидше розробляти вебсайт;
- управління вмістом - платформи CMS пропонують такі функції, як редактори WYSIWYG, керування медіафайлами та керування робочим процесом, що спрощує оновлення вмісту та співпрацю;
- екосистема плагінів - платформи CMS часто мають широкий спектр доступних плагінів і розширень, які розширюють функціональність без великого кодування;
- підтримка спільноти - популярні платформи CMS мають великі спільноти користувачів, які надають доступ до ресурсів, навчальних посібників і підтримки [16].

Недоліки CMS:

- обмежене налаштування: платформи CMS можуть мати обмеження щодо налаштування, особливо щодо складної функціональності або унікального дизайну [9];
- витрати на продуктивність - платформи CMS можуть вводити додатковий код і плагіни, що потенційно може вплинути на продуктивність веб-сайту;

- проблеми безпеки - платформи CMS можуть бути вразливими до системи безпеки, якщо їх регулярно не оновлювати та не захищати;
- крива навчання персоналізації - широке налаштування CMS може вимагати вивчення мов програмування [10].

Розробка вебсайтів має інший метод створення за допомогою мов програмування, фреймворків і бібліотек. Цей метод забезпечує повний контроль над дизайном, функціональністю та продуктивністю веб-сайту.

Переваги розробки сайту власноруч:

- гнучкість і налаштування - розробка на замовлення дозволяє створювати індивідуальні рішення, унікальний дизайн і складну функціональність [18];
- оптимізація продуктивності - спеціальні веб-сайти можна оптимізувати для підвищення швидкості та ефективності, що призведе до кращої продуктивності;
- масштабованість - спеціальна розробка дозволяє створювати масштабовані архітектури з урахуванням майбутнього зростання та змін;
- покращена безпека - застосовуючи надійні заходи безпеки, спеціальні веб-сайти можуть пом'якшити вразливі місця та захистити від атак [19];
- повний контроль: з індивідуальною розробкою немає жодних обмежень, які накладаються попередньо створеними шаблонами чи плагінами.

Недоліки розробки сайту власноруч:

- триваліший час розробки. Створення спеціального веб-сайту з нуля вимагає більше часу та ресурсів порівняно з використанням CMS;
- вищі витрати на розробку: розробка на замовлення часто тягне за собою вищі витрати через потребу в кваліфікованих розробниках і обширному кодуванні;

– технічне обслуговування та оновлення. Спеціальні вебсайти потребують постійного обслуговування, оновлень і потенційних виправлень помилок;

– технічна експертиза вимагає хорошого розуміння мов програмування та фреймворків [20].

В таблиці 2.1 наведено порівняння двох методів розробки вебсайтів.

Таблиця 2.1 – Порівняння двох методів розробки вебсайтів

	Переваги	Недоліки
Розробка за допомогою CMS	– зручність платформи CMS; – швидка розробка; – управління вмістом; – екосистема плагінів; – підтримка спільноти.	– обмежене налаштування: платформи; – витрати на продуктивність ; – проблеми безпеки; – крива навчання персоналізації.
Розробка власноруч	– гнучкість налаштування; – оптимізація продуктивності; – масштабованість; – покращена; – повний контроль: з індивідуальною розробкою.	– триваліший час розробки; – вищі витрати на розробку; – технічне обслуговування та оновлення; – технічна експертиза.

Підводячи підсумок, CMS пропонує простіший і швидший спосіб розробки вебсайтів, зосереджуючись на зручному для користувача управлінні вмістом. Він підходить для проектів зі стандартними вимогами та

обмеженими потребами налаштування. З іншого боку, розробка вебсайту власноруч пропонує повну гнучкість, масштабованість і контроль, але вимагає більше часу, ресурсів і технічних знань. Він підходить для складних проектів, які вимагають унікальної функціональності, дизайну та оптимізації продуктивності.

Для розробки вебсайту «Помічник домогосподарки» було обрано метод розробки програмного коду власноруч, тому що надалі досить важлива оптимізація сайту для просування в соціальних мережах.

На рисунку 2.1 показана структура вебсайту. Вона складається із декількох сторінок та блоків із категоріями рецептів.

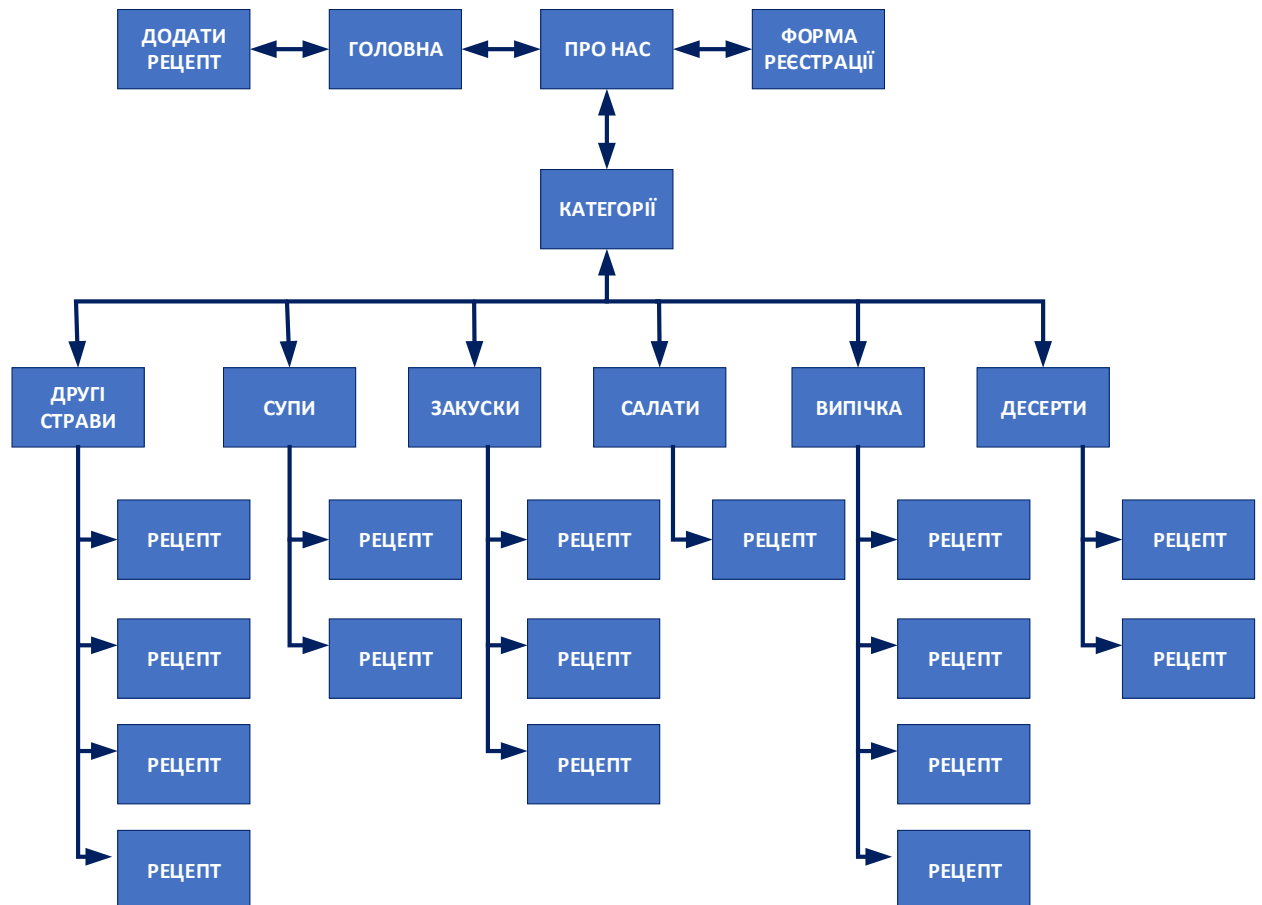


Рисунок 2.1 – Структура вебсайту «Помічник домогосподарки»

2.2 Вибір мов програмування вебсайту

Сучасні мови та фреймворки для розробки вебсайтів еволюціонували, щоб забезпечити ефективні, надійні та масштабовані рішення. Ось кілька популярних.

JavaScript — це універсальна мова програмування, яка використовується як для зовнішньої, так і для внутрішньої розробки. Вона забезпечує інтерактивність і динамічні функції на веб-сайтах [15]. До популярних фреймворків JavaScript належать:

- React широко поширений інтерфейсний фреймворк для створення інтерфейсів користувача та односторінкових програм [16];
- Angular комплексний зовнішній фреймворк, який підтримується Google і підходить для великомасштабних програм [17];
- Vue.js прогресивний фреймворк JavaScript із м'якою кривою навчання, ідеальний для створення інтерактивних інтерфейсів [18].

Python — це мова програмування високого рівня загального призначення, яка відома своєю зрозумілістю та простотою. Він широко використовується в веброботці. Серед популярних фреймворків Python:

- Django надійний і повнофункціональний фреймворк, який слідує архітектурному шаблону модель-подання-контролер (MVC) [19];
- Flask: легкий фреймворк, який дозволяє швидко розробляти веб-додатки малого та середнього розміру [20].

Ruby — це динамічна об'єктно-орієнтована мова програмування, відома своєю простотою та продуктивністю. Він часто використовується з фреймворком Ruby on Rails, який слідує шаблону MVC і наголошує на конвенціях над конфігурацією.

PHP — це широко використовувана серверна мова сценаріїв для веб-розробки. Він підтримує такі популярні системи керування вмістом (CMS), як WordPress, Drupal і Joomla. Фреймворк Laravel є популярним вибором для розробки PHP, що забезпечує елегантний і виразний синтаксис.

Java — це широко поширена мова програмування з сильною екосистемою для веброзробки. Фреймворки Java, які зазвичай використовуються для веброзробки, включають:

- Spring: комплексна структура, яка спрощує розробку програм корпоративного рівня;
- JavaServer Faces (JSF): вебфреймворк на основі компонентів, включений у Java Enterprise Edition (Java EE) для створення інтерфейсів користувача.

C# — потужна об'єктно-орієнтована мова програмування, розроблена Microsoft. Він зазвичай використовується для веб-розробки за допомогою структури ASP.NET. ASP.NET пропонує широкий спектр інструментів і бібліотек для створення надійних веб-додатків.

TypeScript — це наднабір JavaScript, який додає статичний тип та інші функції для покращення розробки JavaScript. Він часто використовується в поєднанні з такими фреймворками, як Angular, для створення великомасштабних програм.

Це лише кілька прикладів сучасних мов і фреймворків для розробки вебсайтів. Кожна мова та фреймворк мають свої сильні сторони та особливості, що задовольняють різні потреби та переваги розробки. Вибір мови та фреймворку залежить від таких факторів, як вимоги до проекту, масштабованість, продуктивність, досвід розробників і підтримка спільноти.

Кожна комбінація мови та фреймворку має власний набір переваг і недоліків. Вибір залежить від вимог проекту, досвіду розробника, потреб у масштабованості та підтримки екосистеми. Важливо враховувати такі фактори, як продуктивність, підтримка спільноти, крива навчання та конкретні потреби веб-сайту, що розробляється.

Всі ці мови можна поділити на розробку фроненду та бекенду. В таблиці 2.2 наведено аналіз мов програмування фронтенду.

Таблиця 2.2 – Аналіз мов програмування вебсайту фронтенд

Мова	Переваги	Недоліки
HTML (Hypertext Markup Language) — стандартна мова розмітки для створення структури та вмісту веб-сторінок. CSS (каскадні таблиці стилів) використовується для опису представлення та макета веб-сторінок.	прості у вивченні, широка підтримка, необхідні для веброзробки.	обмежена інтерактивність, потрібні додаткові мови програмування для динамічної функціональності. JavaScript:
JavaScript — це універсальна мова сценаріїв, яка додає веб-сторінкам інтерактивність і динамічність. Популярні фреймворки/бібліотеки JavaScript: React, Angular, Vue.js.	багата інтерактивність, велика екосистема бібліотек і фреймворків, працює у веб-браузерах.	проблеми із сумісністю веб-переглядача, потенційні проблеми з продуктивністю, можуть бути схильні до вразливості безпеки.
TypeScript — це типізований наднабір JavaScript, який покращує досвід розробки та забезпечує статичний тип. Популярні фреймворки TypeScript: Angular, React.	перевірка типів, розширені інструменти та навігація кодом, покращена продуктивність розробника.	крива навчання для розробників, які звикли до JavaScript, додатковий етап збірки для транспіляції на JavaScript.

В таблиці 2.3 наведено аналіз мов програмування бекенду.

Таблиця 2.3 – Аналіз мов програмування вебсайту бекенд

Мова	Переваги	Недоліки
Python — це універсальна та читабельна мова, відома своєю простотою та легкістю використання. Популярні фреймворки Python: Django, Flask.	Чіткий синтаксис, великі бібліотеки, сильна підтримка спільноти.	Продуктивність може бути нижчою порівняно зі скомпільованими мовами, можуть знадобитися додаткові конфігурації для високої масштабованості.
Ruby — це гнучка та виразна мова, яка зосереджена на простоті та задоволенні розробників. Популярний фреймворк Ruby: Ruby on Rails.	Стислий і зрозумілий синтаксис, конвенційний підхід над конфігурацією, вбудовані функції для швидкого розвитку.	Продуктивність може бути нижчою порівняно з іншими мовами, крива навчання для розробників, які тільки починають працювати з Ruby.
Java — це широко використовувана та незалежна від платформи мова, відома своєю стабільністю та масштабованістю. Популярні фреймворки Java: Spring, JavaServer Faces (JSF).	Потужна екосистема, великі бібліотеки, відмінна продуктивність, висока масштабованість.	Більш докладний синтаксис, триваліші цикли розробки порівняно з мовами з динамічною типізацією.
C# — це універсальна мова, розроблена корпорацією Майкрософт, яка підходить для створення надійних і масштабованих програм. Популярний фреймворк C#: ASP.NET.	Чітко типізована мова, чудова підтримка інструментів, повна інтеграція з технологіями Microsoft.	Обмежено екосистемою Windows, крива навчання для розробників, які не знайомі з C#.
PHP — популярна мова серверного програмування для веб-розробки.	Проста у вивченні та використанні, широка підтримка спільноти, сумісність між платформами, інтеграція з веб-серверами:	Продуктивність, слабка типізація, вразливість системи безпеки.

Проаналізувавши всі мови і задачі розробки проєкту. Для фронтенду вебсайту були обрані технології HTML, CSS, JavaScript (Angular). Розробка бекендної частини вебсайту буде проводитись на мові PHP.

2.3 Вибір бази даних

Сучасні типи баз даних для сайтів поділяються наступним чином:

- реляційні бази даних, такі як MySQL, PostgreSQL і Oracle, організовують дані в таблиці з попередньо визначеними зв'язками між ними. Вони використовують структуровану мову запитів (SQL) для керування та запиту даних. Реляційні бази даних широко використовуються і забезпечують надійну цілісність даних, масштабованість і підтримку складних запитів;

- бази даних NoSQL, включаючи MongoDB, CouchDB і Cassandra, є нереляційними базами даних, які зберігають дані в гнучких форматах без схем, таких як пари ключ-значення, документи або структури графів. Бази даних NoSQL відомі своєю високою масштабованістю, високою продуктивністю та гнучким моделюванням даних;

- бази даних NewSQL, такі як CockroachDB і TiDB, прагнуть поєднати найкращі функції як реляційних баз даних, так і баз даних NoSQL. Вони пропонують масштабованість і відмовостійкість баз даних NoSQL, зберігаючи властивості ACID (атомність, узгодженість, ізоляція, довговічність) і сумісність з SQL.

До переваг використання MySQL для вебсайтів відноситься:

- широке застосування. MySQL є однією з найпопулярніших і широко використовуваних реляційних баз даних з відкритим кодом;

- масштабованість і продуктивність. MySQL має високу масштабованість і може ефективно обробляти великі обсяги даних і високі навантаження трафіку. Вона пропонує різні методи оптимізації, параметри індексування та механізми кешування для підвищення продуктивності;

- відповідність ACID. MySQL забезпечує відповідність ACID, що гарантує надійне та безпечне виконання транзакцій. Це вкрай важливо для підтримки цілісності та узгодженості даних, особливо в програмах, які вимагають складних маніпуляцій з даними;

- кросплатформна сумісність. MySQL сумісний з різними операційними системами, включаючи Windows, Linux, macOS і різними веб-серверами. Ця гнучкість забезпечує легку інтеграцію в різноманітні середовища веб-розробки;

- тиражування та висока доступність. MySQL підтримує реплікацію, дозволяючи створювати кілька копій бази даних для резервування та балансування навантаження. Вона пропонує варіанти реплікації головний-підлеглий і головний-головний, забезпечуючи високу доступність і відмовостійкість;

- безпека даних. MySQL забезпечує надійні функції безпеки, включаючи автентифікацію користувача, контроль доступу та шифрування даних. Вона підтримує безпечні з'єднання через SSL/TLS, захищаючи передачу даних між програмою та базою даних.

Основні елементи бази даних для вебсайту «Помічник домогосподарки», який зберігає книгу рецептів описані далі.

На етап розробки було створено шість таблиць:

- users - таблиця, що зберігає користувачів (табл. 2.4);
- recipe – таблиця, що зберігає інформацію про рецепт (табл. 2.5);
- step– зберігає інформацію про кроки рецепту (табл. 2.6);
- categories – категорії рецептів (табл. 2.7.);
- products – продукти, необхідні для рецепту (табл. 2.8);
- rating – дані рейтингу рецепту (табл. 2.9);

Таблиця 2.4 – Таблиця користувачів

№ п/п	Назва атрибута	Тип даних
1	ID_USER	INTEGER
2	NAME_USER	VARCHAR(255)
3	EMAIL_USER	VARCHAR(255)
4	PASSWORD_USER	VARCHAR(255)
5	PHOTO_USER	INTEGER

Таблиця 2.5 – Таблиця рецептів

№ п/п	Назва атрибута	Тип даних
1	ID_RECIPE	INTEGER
2	NAME_RECIPE	VARCHAR(255)
3	ID_CATEGORY	INTEGER
4	DESCRIPTON	VARCHAR(255)
5	TIME	VARCHAR(255)
6	NUMBER	INTEGER
7	ID_PRODUCTS	INTEGER
8	AMOUNT	INTEGER
9	ID_USER	INTEGER

Таблиця 2.6 – Таблиця кроків приготування

№ п/п	Назва атрибута	Тип даних
1	ID_STEP	INTEGER
2	ID_RECIPE	INTEGER
3	DESCRIPTION	VARCHAR(100)
4	PHOTO	VARCHAR(100)

Таблиця 2.7 – Таблиця категорій

№ п/п	Назва атрибута	Тип даних
1.	ID	INTEGER
2.	TITLE	VARCHAR(255)
3.	BODY	TEXT

Таблиця 2.8 – Таблиця продукти

№ п/п	Назва атрибута	Тип даних
1.	ID	INTEGER
2.	NAME_PRODUCT	VARCHAR(255)

Таблиця 2.9 – Таблиця рейтингу

№ п/п	Назва атрибута	Тип даних
1	ID	INTEGER
2	ID_USER	INTEGER
3	ID_RECIPE	INTEGER
4	STARS	INTEGER

Концепція нормальної форми є важливою в моделюванні реляційної бази даних для забезпечення цілісності даних і мінімізації надмірності, яка може призвести до суперечливих або неправильних результатів. Нормальні форми визначають набір вимог, яким має відповідати зв'язок або таблиця.

У реляційній моделі відношення завжди знаходиться в першій нормальній формі за визначенням. Однак окремі таблиці можуть неправильно відображати зв'язки та не мати першої нормальної форми. Для досягнення вищих нормальних форм необхідно виконати певні критерії.

Вважається, що змінна відношення знаходиться у другій нормальній формі, якщо вона відповідає вимогам першої нормальної форми і якщо кожен неключовий атрибут (який не може бути отриманий з первинного ключа) залежить виключно від усього первинного ключа. Це забезпечує ефективну організацію даних і уникає надмірності.

Рухаючись далі, база даних перебуває в третій нормальній формі, коли вона приведена до другої нормальної форми, і кожен неключовий стовпець не залежить один від одного. Це означає, що кожен стовпець містить лише дані, які безпосередньо пов'язані з первинним ключем, і немає транзитивної залежності між неключовими стовпцями.

У нашому конкретному випадку база даних була розроблена для третьої нормальної форми. Це можна спостерігати на діаграмі бази даних, зображеній на рисунку 2.2, де видно, що кожен неключовий стовпець незалежний від інших, що забезпечує цілісність даних і мінімізує надмірність.

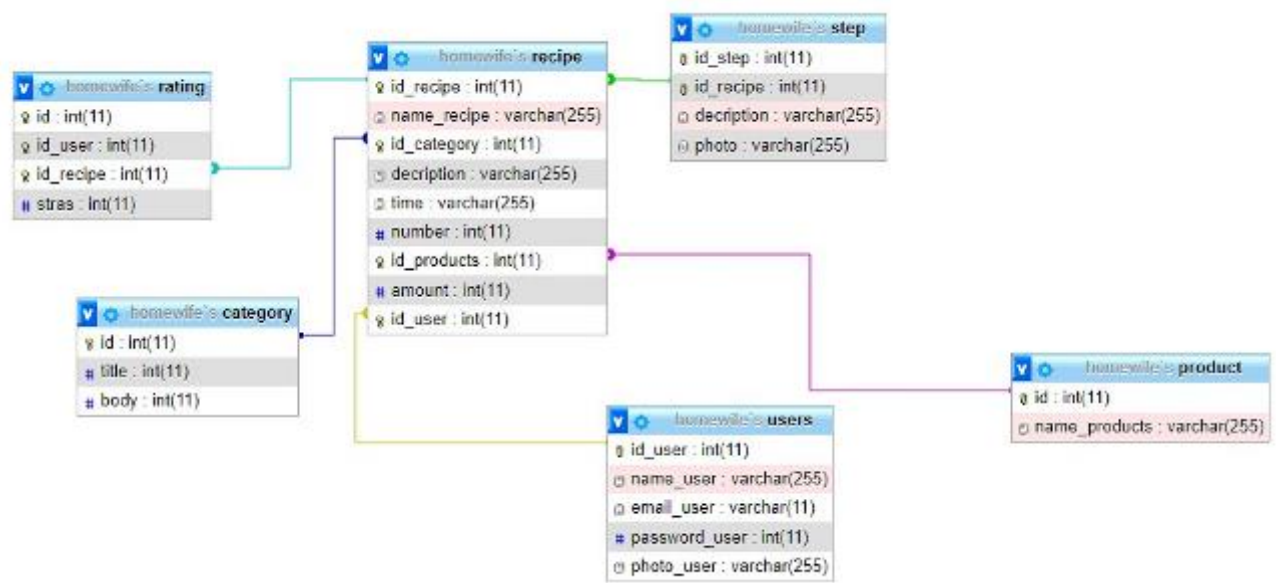


Рисунок 2.2 – Реляційна модель бази даних

2.4 Висновки за розділом 2

Підводячи підсумок, CMS пропонує простіший і швидший спосіб розробки вебсайтів, зосереджуючись на зручному для користувача управлінні вмістом. Він підходить для проектів зі стандартними вимогами та обмеженими потребами налаштування. З іншого боку, розробка вебсайту власноруч пропонує повну гнучкість, масштабованість і контроль, але вимагає більше часу, ресурсів і технічних знань. Він підходить для складних проектів, які вимагають унікальної функціональності, дизайну та оптимізації продуктивності.

Для розробки вебсайту «Помічник домогосподарки» було обрано метод розробки програмного коду власноруч, тому що надалі досить важлива

оптимізація сайту для просування в соціальних мережах. Була розроблена структура вебсайту, яка складається із декількох сторінок та блоків із категоріями рецептів.

Всі ці мови можна поділити на розробку фроненду та бекенду. Проаналізувавши всі мови і задачі розробки проєкту. Для фронтенду вебсайту були обрані технології HTML, CSS, JavaScript (Angular). Розробка бекендної частини вебсайту буде проводитись на мові PHP.

На етап розробки було створено шість таблиць:

- users - таблиця, що зберігає користувачів;
- recipe – таблиця, що зберігає інформацію про рецепт;
- step– зберігає інформацію про кроки рецепту;
- categories – категорії рецептів;
- products – продукти, необхідні для рецепту;
- rating – дані рейтингу рецепту.

У нашому конкретному випадку база даних була розроблена для третьої нормальної форми. Це можна спостерігати на діаграмі бази даних, зображеній на рисунку, де видно, що кожен неключовий стовпець незалежний від інших, що забезпечує цілісність даних і мінімізує надмірність.

3 ОПИС ВЕБСАЙТУ

3.1 Керівництво програміста

Структура файлової системи вебсайту складається з набору папок каталогу, виконуваних файлів (таких як документи HTML, CSS, PHP і JavaScript) і графічних ресурсів. Ці елементи разом створюють функціональні та візуальні компоненти рецептів.

Що стосується графіки, сучасні веб браузері підтримують різні формати зображень. Найбільш часто використовувані формати включають JPEG (jpg), Portable Network Graphics (png), Graphics Interchange Format (gif) і Flash. Крім того, браузері мають можливість відтворювати масштабовану векторну графіку (SVG) і використовувати елемент canvas HTML5 для створення векторної графіки.

На рисунку 3.1 представлено технологічний стек, використаний для розробки вебсайту. Він демонструє комбінацію різних технологій та інструментів, які використовуються для створення та роботи, забезпечуючи його функціональність і продуктивність.

Що стосується макета сторінки вебсайту, була реалізована структура з трьох частин, яка включає в себе розділ заголовка, блок основного вмісту та розділ нижнього колонтитула. Ці розділи були організовані за допомогою відповідних елементів HTML, таких як «заголовок», «рецепти» та «нижній колонтитул». Визнаючи їхнє загальне використання на кількох сторінках, розділи верхнього та нижнього колонтитулів було вилучено та збережено в окремих файлах.

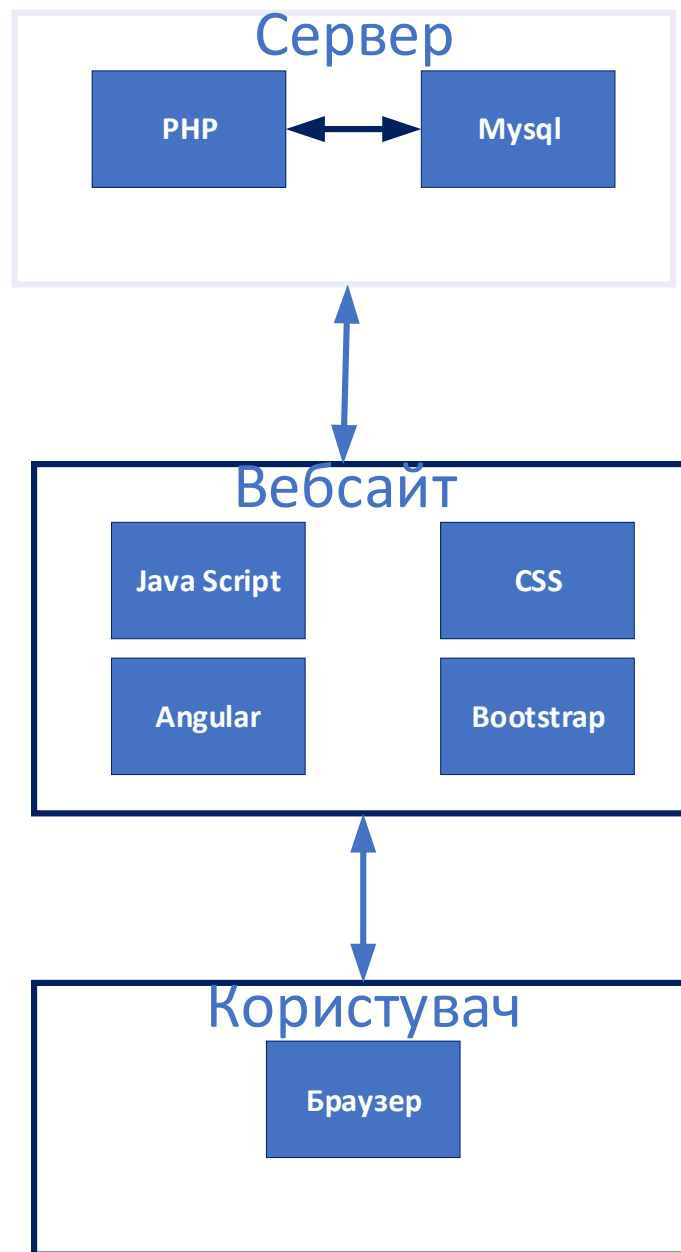


Рисунок 3.1 – Стек технологій вебсайту

Щоб створити привабливий візуальний дизайн для сторінок, стилі CSS були ретельно розроблені та об'єднані в окремий файл. Фрагмент коду, що демонструє реалізацію дизайну нижнього колонтитула, можна знайти в лістингу 3.1. Цей сегмент коду демонструє, як стилі CSS було застосовано до розділу нижнього колонтитула для досягнення бажаного візуального представлення.

Лістинг 3.1 – Заголовки вебсайту

```

:root {
  --color-default: #2b180d;
  --color-primary: #56b8e6;
  --color-secondary: #1b2f45;
}

/* Smooth scroll behavior */
:root {
  scroll-behavior: smooth;
}

/*-----
# General
-----*/
body {
  font-family: var(--font-default);
  color: var(--color-default);
}

a {
  color: var(--color-primary);
  text-decoration: none;
  transition: 0.3s;
}

```

Розділивши розділи верхнього та нижнього колонтитулів на окремі файли та використовуючи стилі CSS для естетичних покращень, сторінки веб-сайту структуровані та розроблені таким чином, щоб забезпечити візуально приємний та організований досвід користувача.

Використання функцій Java Script показані на лістингу 3.2.

Лістинг 3.2 – Функції Java Script

```

document.addEventListener('DOMContentLoaded', () => {
  "use strict";

  /**
   * Preloader
   */
  const preloader = document.querySelector('#preloader');
  if (preloader) {
    window.addEventListener('load', () => {
      preloader.remove();
    });
  }

  /**
   * Sticky header on scroll
   */
  const selectHeader = document.querySelector('#header');
  if (selectHeader) {
    document.addEventListener('scroll', () => {
      window.scrollY > 100 ? selectHeader.classList.add('sticked') : selectHeader.classList.remove('sticked');
    });
  }

  /**
   * Mobile nav toggle
   */
  const mobileNavShow = document.querySelector('.mobile-nav-show');
  const mobileNavHide = document.querySelector('.mobile-nav-hide');

  document.querySelectorAll('.mobile-nav-toggle').forEach(el => {
    el.addEventListener('click', function(event) {
      event.preventDefault();
      mobileNavToggle();
    })
  });

  function mobileNavToggle() {

```

3.2 Керівництво користувача

Рисунок 3.2 показує вигляд головної сторінки вебсайту. Вона складається із форми реєстрації або авторизації. Переліку категорій рецептів. Поле для пошуку рецепту на сайті.

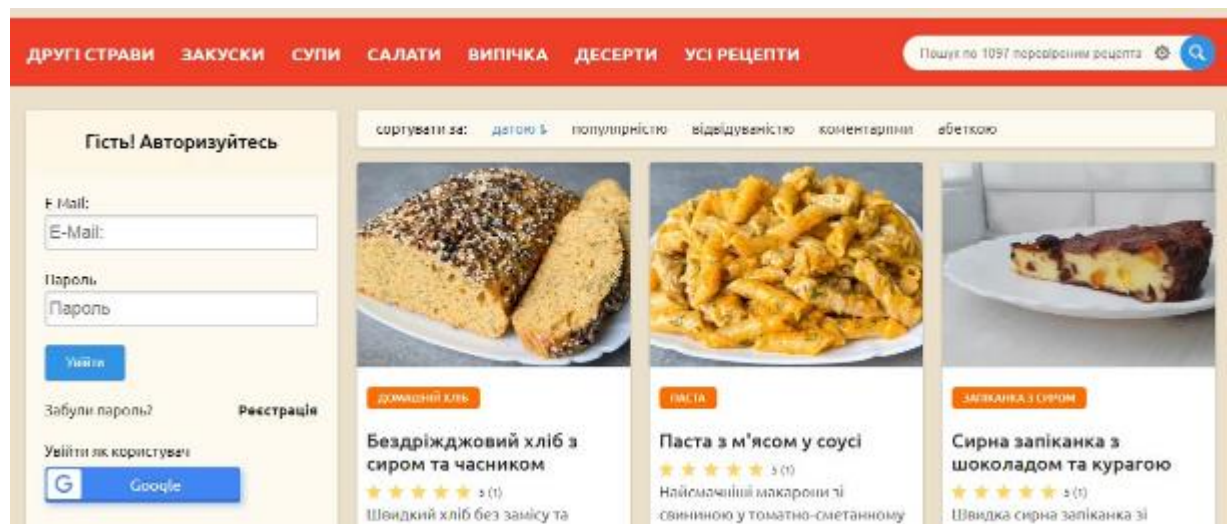


Рисунок 3.2 – Головна сторінка

Меню категорій складається із переліку: другі страви, закуски, супи, випічка, десерти, усі рецепти (рис. 3.3).



Рисунок 3.3 – Меню категорій

Пошук по вебсайту може бути за ключовим словами або використовуючи розширене меню пошуку (рис. 3.4).

Рисунок 3.4 – Розширене меню пошуку

На заголовку вебсайту розташована одна із головних кнопок системи – це кнопка «Додати рецепт» (рис. 3.5).

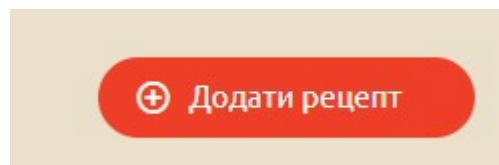


Рисунок 3.5 – Додати рецепт

Форма реєстрації/авторизації має поля для вводу пошти та паролю, кнопку «Увійти». Або можливий варіант використання облікового запису гугл акаунту (рис. 3.6).

Гість! Авторизуйтесь

E-Mail:

Пароль

Забули пароль? **Реєстрація**

Увійти як користувач



Рисунок 3.6 – Форма реєстрації/авторизації

Для того, щоб опублікувати рецепт треба заповнити необхідні поля вебсайту (рис. 3.7-3.9).

Новий рецепт

Оберіть категорію *

Назва рецепту *

Опис рецепту *

B
I
U
↺
↻
🔗
📷
📄
📑
🔍

Час приготування *
 Годин Хвилин

Рецепт на: *
 Осіб

☐ Потрібна підготовка

Рисунок 3.7 – Створення рецепту

Час приготування * Рецепт на: *

 Годин Хвилин  Порції

☐ Потрібна підготовка

Вкажіть інгредієнти * + масове додавання

Інгредієнт	Кількість	Міра ваги/обсягу
 Почніть вводити назву	Вага	Виберіть... 

 Додати інгредієнт  Додати підрозділ

Рисунок 3.8 – Додавання продуктів та часу приготування

1 КРОК

[Завантажте фотографію](#)

B I U          

Введіть опис кроку

Рисунок 3.9 – Опис кроків приготування

Перегляд вже створених рецептів має декілька змінних полів. На рисунку 3.10 показана головна фотографія.

Торт "Наполеон" із заварним кремом з листкового тіста

ЛИСКОВИЙ ТОРТ



Рисунок 3.10 – Опис рецепту

Наступна інформація показує опис рецепту, кількість порцій та години приготування (рис. 3.11).



Приготування
13 годин



Рецепт на:
9 порцій

Опис

Пропоную приготувати дуже смачний торт "Наполеон" із готового листкового тіста із заварним кремом. Це у рази швидше, ніж замішувати тісто самостійно. А ще коржі виходять смачними, добре просочуються ніжним заварним кремом. Простий рецепт "Наполеона". Запрошую всіх!

Рисунок 3.11 – Опис рецепту

Кількість відповідних продуктів можна змінювати залежно від розміру порцій готового продукту (рис. 3.12).

Інгредієнтів на порцій

☐ Тісто листкове бездрожжеве	1 кг
☐ Молоко	1 л
☐ Яйце куряче	3 шт
☐ Цукор	2 склян.
☐ Борошно пшеничне	5 ст. л.
☐ Ванільний цукор	1 пач.
☐ Олія рослинна	2 ст. л.

Рисунок 3.12 – Кількість порцій

Наприкінці рецепту є інформація про його рейтинг та пропонуються схожі рецепти із обраної категорії. (рис. 3.13, рис. 3.14).

Рейтинг: 4.7 (99)

★★★★★

Будь ласка, оцініть цей рецепт

Коментарі (0)

Поділитися рецептом із друзями

Рисунок 3.13 – Рейтинг рецепту



Рисунок 3.14 - Пропозиції

Важливий елемент вебсайту – це сортування рецептів (рис. 3.15).

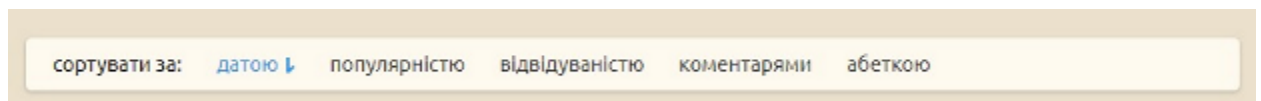


Рисунок 3.15 – Сортування рецептів

3.3 Висновки за розділом 3

Структура файлової системи вебсайту складається з набору папок каталогу, виконуваних файлів (таких як документи HTML, CSS, PHP і JavaScript) і графічних ресурсів. Ці елементи разом створюють функціональні та візуальні компоненти рецептів.

Що стосується макета сторінки вебсайту, була реалізована структура з трьох частин, яка включає в себе розділ заголовка, блок основного вмісту та розділ нижнього колонтитула. Ці розділи були організовані за допомогою відповідних елементів HTML, таких як «заголовок», «рецепти» та «нижній

колонтитул». Визнаючи їхнє загальне використання на кількох сторінках, розділи верхнього та нижнього колонтитулів було вилучено та збережено в окремих файлах.

Щоб створити привабливий візуальний дизайн для сторінок, стилі CSS були ретельно розроблені та об'єднані в окремий файл. Головна сторінка складається із форми реєстрації або авторизації. Переліку категорій рецептів. Поле для пошуку рецепту на сайті. Пошук по вебсайту може бути за ключовим словами або використовуючи розширене меню пошуку. Меню категорій складається із переліку: другі страви, закуски, супи, випічка, десерти, усі рецепти. Форма реєстрації/авторизації має поля для вводу пошти та паролю, кнопку «Увійти». Або можливий варіант використання облікового запису гугл акаунту. На заголовку вебсайту розташована одна із головних кнопок системи – це кнопка «Додати рецепт». Для того, щоб опублікувати рецепт треба заповнити необхідні поля вебсайту. Перегляд вже створених рецептів має декілька змінних полів: опис рецепту, кількість порцій та години приготування. Кількість відповідних продуктів можна змінювати залежно від розміру порцій готового продукту. Наприкінці рецепту є інформація про його рейтинг та пропонуються схожі рецепти із обраної категорії.

4 ОПТИМІЗАЦІЯ І ТЕСТУВАННЯ ВЕБСАЙТУ

4.1 Оптимізація вебсайту

Оптимізація коду розробленого вебсайту включає ряд етапів, спрямованих на підвищення продуктивності, скорочення часу завантаження та покращення загального досвіду користувача. Нижче наведено основні етапи оптимізації вебсайту.

Перший етап передбачає проведення ретельного аналізу існуючої кодової бази. Це включає визначення областей коду, які спричиняють вузькі місця продуктивності або неефективність. Інструменти профілювання та методи моніторингу ефективності можна використовувати для збору даних і визначення областей, які потребують оптимізації.

Кодову базу перевіряють, щоб виявити зайвий, неефективний або погано написаний код. Це включає усунення невикористаних змінних або функцій, видалення непотрібних залежностей і покращення структури коду та читабельності.

Перевірка валідності коду вебсайту була зроблена за допомогою онлайн сервісу validator.w3.org (рис. 4.1).

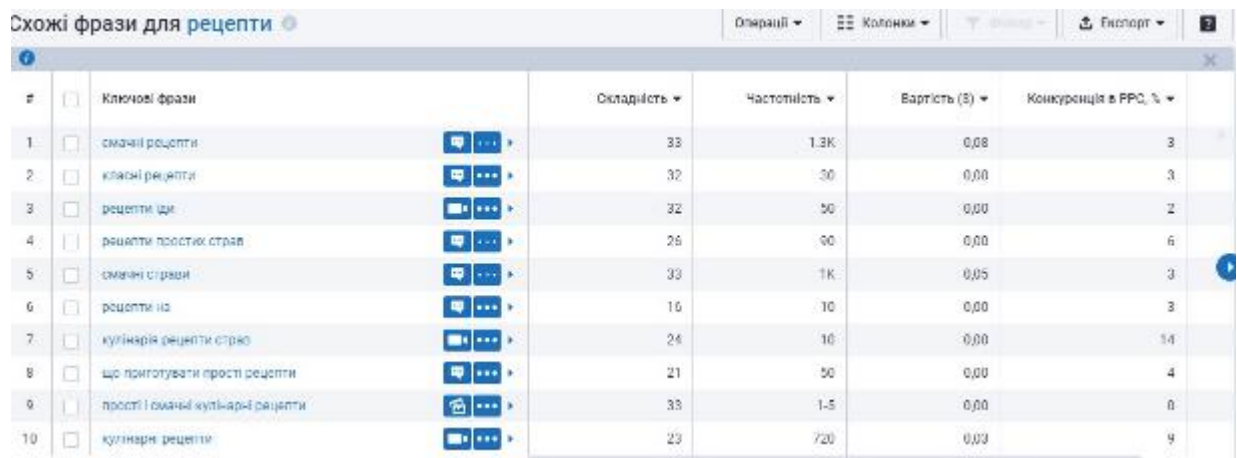


Рисунок 4.1 – Валідність коду вебсайту

В результаті перевірки було виявлено 161 помилку в коді, які потім були усунені.

Впровадження механізмів кешування, таких як кешування веб-переглядача та кешування на сервері, може значно покращити продуктивність вебсайту. Крім того, стиснення статичних файлів, таких як CSS і JavaScript, за допомогою таких методів, як мініфікація та стиснення gzip, зменшує розміри файлів і покращує час завантаження.

Аналіз ключових слів за даною темою в Інтернеті показано на рисунку 4.2.



#	Ключові фрази	Об'єм	Частотність	Вартість (\$)	Конкуренція в PPQ, %
1	смачні рецепти	33	1.3K	0.08	3
2	класні рецепти	32	50	0.00	3
3	рецепти їди	32	50	0.00	2
4	рецепти простих страв	26	50	0.00	6
5	смачні страви	33	1K	0.05	3
6	рецепти на	16	10	0.00	3
7	кулінарія рецепти страви	24	10	0.00	14
8	що приготувати прості рецепти	21	50	0.00	4
9	прості і смачні кулінарії рецепти	33	1-5	0.00	0
10	кулінарія рецепти	23	720	0.03	9

Рисунок 4.2 – Аналіз ключових слів

Дотримуючись цих етапів оптимізації, код вебсайту було оптимізовано, що призвело до покращення продуктивності, швидшого часу завантаження та покращення взаємодії з користувачем. Регулярний моніторинг і технічне обслуговування забезпечать оптимальну роботу вебсайту з часом.

4.2 Тестування вебсайту

Сумісність із різними браузерами та оптимізація для різних пристроїв вебсайтів є важливими факторами, щоб переконатися, що вебсайт

функціонує належним чином і виглядає привабливим у різних браузерах і пристроях.

Тестування проводилось у кількох браузерах, включаючи такі популярні, як Chrome, Firefox, Safari та Edge. Це допомагає виявити та вирішити будь-які проблеми візуалізації чи функціональності, характерні для кожного браузера.

Впровадження адаптивного вебдизайну дозволяє макету вебсайту адаптуватися та налаштовуватися відповідно до розміру екрана пристрою користувача. Це забезпечує оптимальний перегляд на різних пристроях, включаючи комп'ютери, планшети та мобільні телефони.

Тестування вебсайту на різних пристроях, включаючи смартфони, планшети та екрани різних розмірів, має важливе значення для забезпечення належної функціональності та візуального вигляду на різних пристроях, якими користуються відвідувачі веб-сайту.

Зосереджуючись на кросбраузерній сумісності та оптимізації для різних пристроїв, вебсайти може охопити ширшу аудиторію, забезпечити узгоджену взаємодію з користувачами та забезпечити оптимальну продуктивність на різних платформах і браузерах (рис. 4.3).

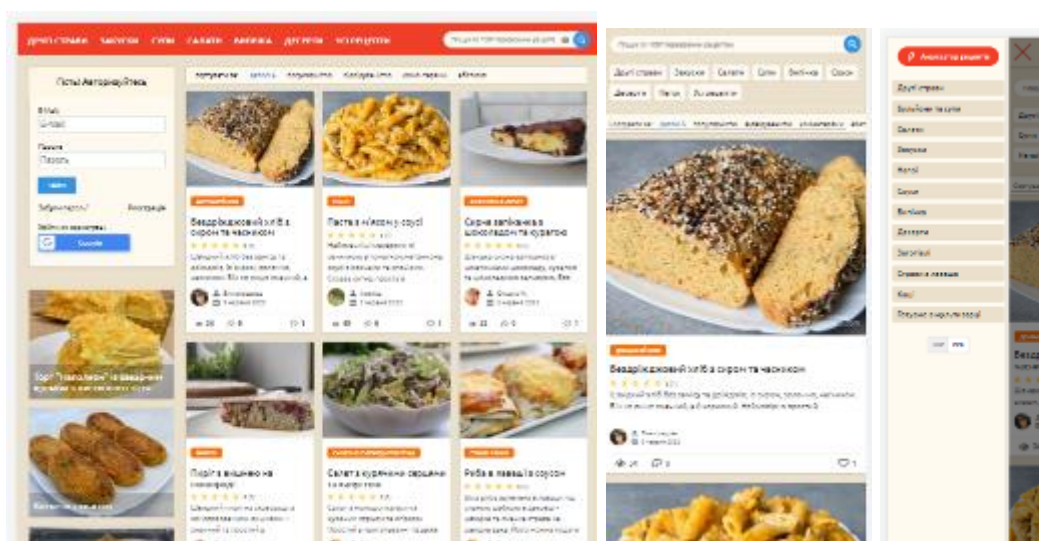


Рисунок 4.3 – Тестування кросбраузерності

4.3 Висновки за розділом 4

Дотримуючись етапів оптимізації, код вебсайту було оптимізовано, що призвело до покращення продуктивності, швидшого часу завантаження та покращення взаємодії з користувачем. Регулярний моніторинг і технічне обслуговування забезпечать оптимальну роботу вебсайту з часом.

Кросбраузерне тестування вебсайту на різних пристроях, включаючи смартфони, планшети та екрани різних розмірів, має важливе значення для забезпечення належної функціональності та візуального вигляду на різних пристроях, якими користуються відвідувачі веб-сайту. Зосереджуючись на кросбраузерній сумісності та оптимізації, вебсайти може охопити ширшу аудиторію, забезпечити узгоджену взаємодію з користувачами та забезпечити оптимальну продуктивність на різних платформах і браузерах

ВИСНОВКИ

В даній роботу був розроблений вебсайт «Помічник домогосподарки». Ця система служить платформою для створення та перегляду рецептів серед інших рецептів, створених різними користувачами. Додаток розрахований на широку аудиторію і знаходиться у вільному доступі для користувачів.

Розробка програми була зосереджена на виконанні конкретних функціональних вимог, таких як:

- поділ на основі ролей, що дозволяє керувати дозволами;
- система CRUD (Create, Read, Update, Delete) для рецептів, що дозволяє користувачам створювати, редагувати, видаляти та оновлювати власні рецепти;
- створення особистих профілів користувачів з можливістю оновлення персональних даних в профілі;
- функції керування користувачами, включаючи можливість адміністратора блокувати або видаляти користувача;
- впровадження Guards для безпечного контролю доступу.
- безпечні механізми авторизації та автентифікації.

У процесі розробки вебсайт розвивався та набував унікального стилю та функціональності. У роботі використовувалися фреймворки NestJs і Angular, використовуючи їхні новітні можливості та включаючи сучасні методології та концепції програмування.

Підводячи підсумок, додаток помічник домогосподарки було розроблено з використанням мов html, css, java script, php. Додаток успішно досяг усіх окреслених цілей і вимог, надаючи користувачам безпечну та зручну платформу для створення та перегляду рецептів.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Смержевський Н. В. Штучний інтелект: перспективи розвитку [Електронний ресурс] : thesis / Смержевський Н. В. – [Б. м.], 2019. – Режим доступу: <https://er.knutd.edu.ua/handle/123456789/14286>.
2. Костенко О. В. Ідентифікація IoT / О. В. Костенко // Знання європейського права. – 2021. – № 1. – С. 77–83.
3. Бабак О. Хмарні технології в освіті / Олександр Бабак, Людмила Ісак // Grail of Science. – 2023. – № 27. – С. 486–490.
4. Тур О. Інноваційні цифрові технології у сфері охорони здоров'я: мобільні застосунки / Оксана Тур, Вікторія Шабуніна // Grail of Science. – 2023. – № 27. – С. 336–340.
5. CAMPBELL M. A. Web Site / M. A. CAMPBELL. – [S. l.] : Independently Published, 2019. – 346 p.
6. Balasubramanian V. HTML / V. Balasubramanian, Helen Ashman // ACM SIGWEB Newsletter. – 1994. – Vol. 3, no. 3. – P. 27.
7. Dorman M. HTML / Michael Dorman // Introduction to Web Mapping. – [S. l.], 2020. – P. 3–32.
8. Marketing with Web 20. – [S. l.] : Cognella Academic Publishing, 2012. – 300 p.
9. Смачні рецепти: апетитні і різноманітні страви [Електронний ресурс] – Режим доступу: <https://rud.ua/consumer/recipe/>.
10. Кулінарні рецепти від Євгена Клопотенко [Електронний ресурс] – Режим доступу: <https://klopotenko.com/recipe/>.
11. Смачні страви швидко, смачно [Електронний ресурс] – Режим доступу: <https://shuba.life/>.
12. OnePrice: Інтернет магазин доступних товарів для дому [Електронний ресурс] – Режим доступу: <https://oneprice.ua/>.
13. Rubtsova N. V. WEBSITE OF INSTITUTIONS OF SECONDARY EDUCATION AS AN EFFECTIVE MEANS OF COMMUNICATION

[Electronic resource] / N. V. Rubtsova // AZIMUTH OF SCIENTIFIC RESEARCH: PEDAGOGY AND PSYCHOLOGY. – 2020. – Vol. 9, no. 31. – Mode of access: <https://doi.org/10.26140/anip-2020-0902-0050>.

14. Господарочка [Електроний ресурс] – Режим доступу: <https://gospodarochka.net/ua/>.

15. Sanjay S. K. Web Application Using HTML, CSS, Java script and Java / Kumar S Sanjay, M Giridharadhayalan // International Journal of Innovative Research in Engineering. – 2023. – P. 124–127.

16. Sharan K. Implementing a Script Engine] / Kishori Sharan // Scripting in Java. – Berkeley, CA, 2014. – P. 263–279.

17. Manning L. Angular / Lizzie Manning. – [S. l.] : Independently Published, 2018. – 120 p.

18. Bwi. Cool Web Site / Bwi. – [S. l.] : Pearson Education, Limited, 2000. – 402 p.

19. Smith D. Python Machine Learning: The Crash Course for Beginners to Programming and Deep Learning, Artificial Intelligence, Neural Networks and Data Science. Scikit Learn, Tensorflow, Pandas and Numpy. / Django Smith. – [S. l.] : Independently published, 2019. – 142 p.

20. Шепелєв Д. О. Інформаційна система оцінювання швидкості завантаження веб-додатків [Електронний ресурс] : master's thesis / Д. О. Шепелєв. – [Б. м.], 2021. – Режим доступу: <https://essuir.sumdu.edu.ua/handle/123456789/82567>.

ДОДАТОК А
Слайди презентації

Національний університет
«Запорізька політехніка»
Кафедра програмних засобів

РОЗРОБКА ВЕБСАЙТУ

«ПОМІЧНИК

ДОМОГОСПОДАРКИ»

Виконав: студент гр. КНТ-129

Данило ХИЖНЯК

Керівник:

Олександр СТЕПАНЕНКО

Рисунок А.1 – Слайд 1

Аналіз
моделей
розробки
вебсайтів

Параметр	Порівняння
Залучення клієнтів	Waterfall включає відгуки клієнтів переважно на пізніх етапах, тоді як Agile заохочує постійну участь клієнтів і зворотний зв'язок протягом усього процесу розробки.
Швидкість розробки	Waterfall слідує за послідовним процесом, що потенційно може призвести до більш тривалих циклів розробки. Agile забезпечує швидшу розробку за допомогою ітераційних випусків.
Документація	Waterfall наголошує на вичерпній документації з самого початку, тоді як Agile зосереджується більше на робочому програмному забезпеченні, а не на розширеній документації.
Управління ризиками	Послідовний підхід Waterfall допомагає виявляти й усувати ризики на ранній стадії, тоді як Agile керує ризиками за допомогою постійних ітерацій і частого зворотного зв'язку
Складність	Waterfall підходить для проектів із чітко визначеними вимогами, тоді як Agile краще підходить для складних проектів із змінними вимогами.

2/17

Рисунок А.2 – Слайд 2

Аналіз систем помічників для домогосподарок

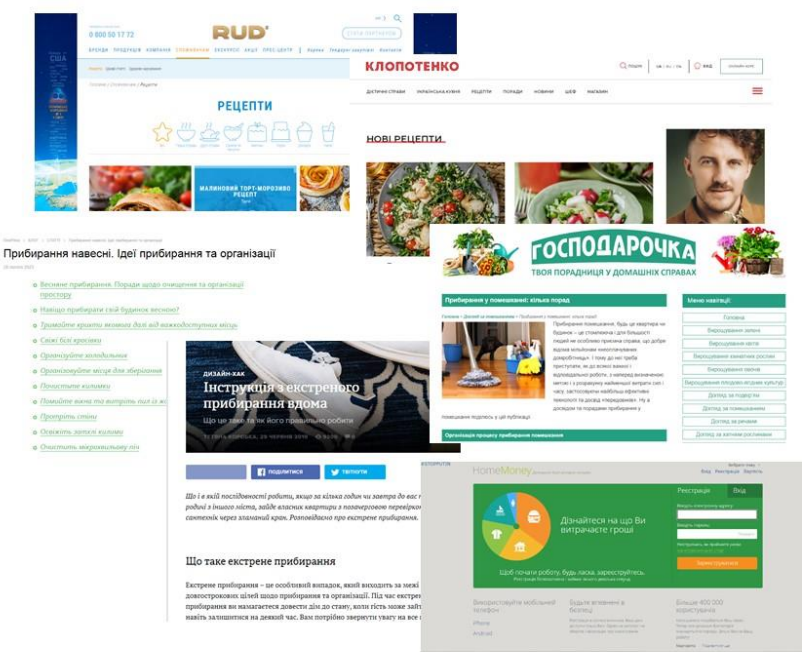


Рисунок А.3 – Слайд 3

Аналіз систем помічників для домогосподарок

Назва вебсайту	Призначення	Юзабіліті	Персоналізація	Актуальність	Відеоконтент	Відгуки/рейтинг	Пошук
rud.ua	Планування їжі та рецептів	+	-	+	+	-	+
klopotenko.com	Планування їжі та рецептів	+	+	+	+	+	+
shuba.life	Планування їжі та рецептів	+	+	+	-	-	+
oneprice.ua	Організації та прибирання дому	-	-	+	-	+	+
the-village.com.ua	Організації та прибирання дому	-	+	+	-	-	+
gospodarochka.in.ua	Організації та прибирання дому	+	-	+	-	-	+
homemoney	Бюджетування та фінансового менеджменту	+	+	+	-	-	+
moneyhouse	Бюджетування та фінансового менеджменту	+	+	+	-	-	-
Власна розробка homewife's	Планування їжі та рецептів	+	+	+	-	+	+

Рисунок А.4 – Слайд 4

Етапи розробки вебсайту

№	Назва етапу	Опис
1	Збір вимог	Це включає визначення здатності домогосподарок як записувати власні рецепти, так і шукати існуючі рецепти.
2	Дизайн	Фаза розробки інтерфейсу користувача зосереджена на створенні інтуїтивно зрозумілого та зручного інтерфейсу для розділу книги рецептів.
3	База даних	Буде розроблена та реалізована база даних для зберігання рецептів. Схема бази даних містить таблиці або колекції для зберігання деталей рецептів, таких як назва, інгредієнти, інструкції та інформація про автора.
4	Бекенд/Фронт енд	Розробка програмного коду інтерфейсу і функцій роботи із БД.
5	Тестування	Для тестування та гарантія якості буде проведено ретельне тестування, щоб переконатися в функціональності, зручності використання та ефективності функції книги рецептів.

5/17

Рисунок А.5 – Слайд 5

Загальні вимоги

Параметр	Опис
Інтерфейс користувача	Інтерфейс користувача має бути інтуїтивно зрозумілим і зручним, щоб домогосподаркам було легко орієнтуватися та ефективно керувати своїми завданнями. Чіткі та організовані макети з налаштованими параметрами є перевагою.
Доступність	Вебсайти мають бути доступні на різних пристроях і з різними розмірами екранів, щоб домогосподарки могли використовувати їх на своїх улюблених пристроях, таких як смартфони, планшети чи настільні комп'ютери.
Безпека даних	Враховуючи конфіденційний характер особистої інформації, заходи безпеки даних, як-от шифрування, надійна автентифікація та регулярне резервне копіювання, мають вирішальне значення для захисту даних користувачів
Інтеграція з іншими платформами	Інтеграція з популярними інструментами продуктивності, календарями та комунікаційними платформами може підвищити зручність використання вебсайту
Персоналізація	Можливість персоналізувати веб-сайт шляхом вибору тем, створення власних категорій або додавання власних полів може покращити взаємодію з користувачем і задовольнити індивідуальні вподобання.
Продуктивність і швидкість реагування	Вебсайти мають бути оптимізовані для швидкого завантаження та безперебійної роботи, забезпечуючи бездоганну взаємодію з користувачем

6/17

Рисунок А.6 – Слайд 6

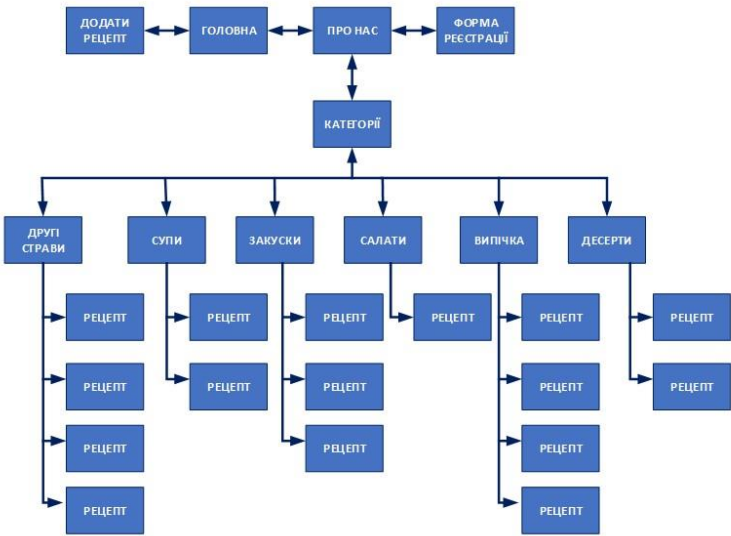
Вибір методів розробки вебсайту

	Переваги	Недоліки
Розробка за допомогою CMS	–зручність - платформи CMS; –швидка розробка; –управління вмістом; –екосистема плагінів; –підтримка спільноти.	–обмежене налаштування платформи; –витрати на продуктивність ; –проблеми безпеки; –крива навчання персоналізації.
Розробка власноруч	–гнучкість і налаштування; –оптимізація продуктивності; –масштабованість; –покращена; –повний контроль: з індивідуальною розробкою.	–триваліший час розробки; –вищі витрати на розробку; –технічне обслуговування та оновлення; –технічна експертиза.

7/17

Рисунок А.7 – Слайд 7

Структура вебсайту



8/17

Рисунок А.8 – Слайд 8

Аналіз мов програмування вебсайту фронтенд

Мова	Переваги	Недоліки
HTML (Hypertext Markup Language) — стандартна мова розмітки для створення структури та вмісту веб-сторінок.	прості у вивченні, широка підтримка, необхідні для веброзробки.	обмежена інтерактивність, потрібні додаткові мови програмування для динамічної функціональності.
CSS (каскадні таблиці стилів) використовується для опису представлення та макета веб-сторінок.		JavaScript:
JavaScript — це універсальна мова сценаріїв, яка додає веб-сторінкам інтерактивність і динамічність.	багата інтерактивність, велика екосистема переглядача, бібліотек і фреймворків, працює у веб-браузерах.	проблеми із сумісністю веб-переглядача, потенційні проблеми з продуктивністю, можуть бути схильні до вразливості безпеки.
Популярні фреймворки/бібліотеки JavaScript: React, Angular, Vue.js.		
TypeScript — це типізований наднабір JavaScript, який покращує досвід розробки та забезпечує статичний тип.	перевірка типів, розширені інструменти та навігація кодом, покращена продуктивність розробника.	крива навчання для розробників, які звикли до JavaScript, додатковий етап збірки для транспіляції на JavaScript.
Популярні фреймворки TypeScript: Angular, React.		

9/17

Рисунок А.9 – Слайд 9

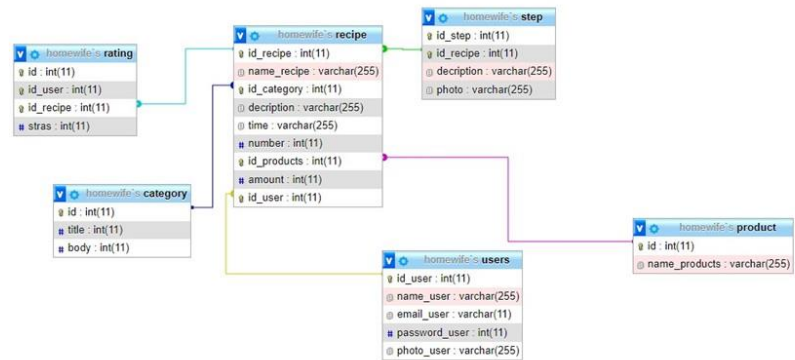
Аналіз мов програмування вебсайту бекенд

Мова	Переваги	Недоліки
Python — це універсальна та читабельна мова, відома своєю простотою та легкістю використання.	Чіткий синтаксис, великі бібліотеки, сильна підтримка спільноти.	Продуктивність може бути нижчою порівняно зі скомпільованими мовами, можуть знадобитися додаткові конфігурації для високої масштабованості.
Популярні фреймворки Python: Django, Flask.		
Ruby — це гнучка та виразна мова, яка зосереджена на простоті та задоволенні розробників.	Стислий і зрозумілий синтаксис, конвенційний підхід над порівняно з іншими мовами, крива конфігурацією, вбудовані функції навчання для розробників, які для швидкого розвитку.	Продуктивність може бути нижчою порівняно з іншими мовами, крива конфігурацією, вбудовані функції навчання для розробників, які тільки починають працювати з Ruby.
Популярний фреймворк Ruby: Ruby on Rails.		
Java — це широко використовувана та незалежна від платформи мова, відома своєю стабільністю та масштабованістю.	Потужна екосистема, великі бібліотеки, продуктивність, масштабованість.	Більш докладний синтаксис, відмінна триваліші цикли розробки висока порівняно з мовами з динамічною типізацією.
Популярні фреймворки Java: Spring, JavaServer Faces (JSF).		
C# — це універсальна мова, розроблена корпорацією Майкрософт, яка підходить для створення надійних і масштабованих програм.	Чітко типізована мова, чудова підтримка інструментів, інтеграція з технологіями Microsoft.	Обмежено екосистемою Windows, повна крива навчання для розробників, які не знайомі з C#.
Популярний фреймворк C#: ASP.NET.		
PHP — популярна мова серверного програмування для веб-розробки.	Проста у вивченні та використанні, широка підтримка спільноти, сумісність між платформами, інтеграція з веб-серверами.	Продуктивність, слабка типізація, вразливість системи безпеки.

10/17

Рисунок А.10 – Слайд 10

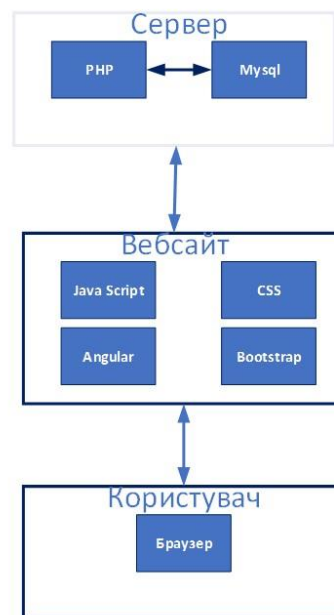
База даних



11/17

Рисунок А.11 – Слайд 11

Стек технологий



12/17

Рисунок А.12 – Слайд 12

Вигляд вебсайту

13/17

Рисунок А.13 – Слайд 13

Додати новий рецепт

14/17

Рисунок А.14 – Слайд 14

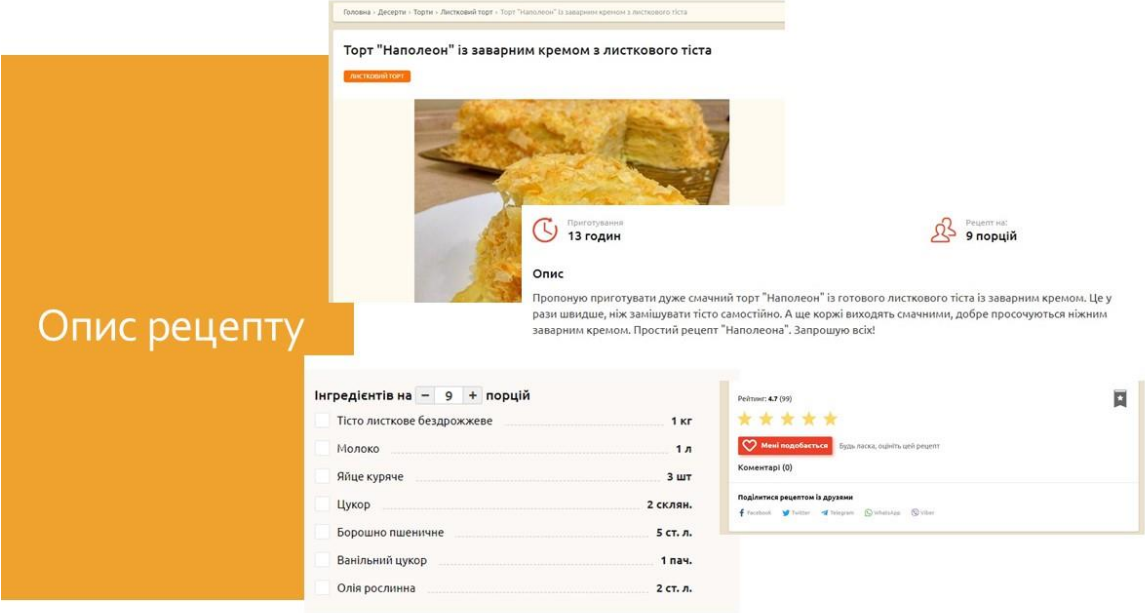


Рисунок А.15 – Слайд 15

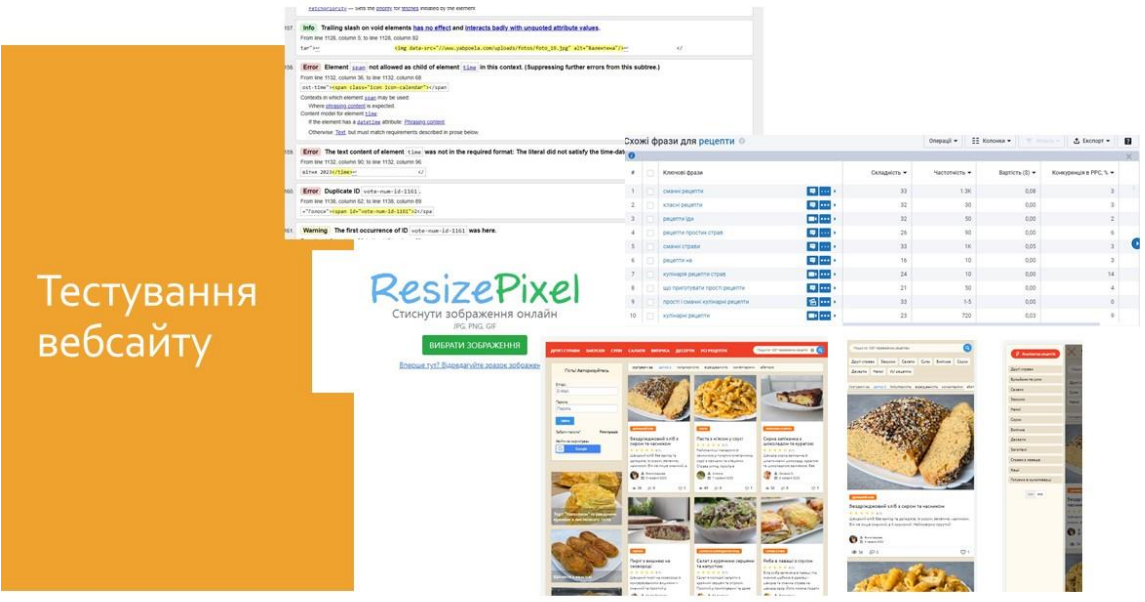


Рисунок А.16 – Слайд 16



Висновки

Задачі, які були вирішені в ході роботи:

- аналіз методів і моделей сучасних вебсайтів;
- аналіз технологій і мов програмування вебсайтів
- розробка бази даних;
- розробка вебсайту;
- оптимізація і тестування вебсайту.