

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет кібербезпеки, комп'ютерної та програмної інженерії (ЗФН)
Кафедра комп'ютерних інформаційних технологій

ДОПУСТИТИ ДО ЗАХИСТУ

Завідувач кафедри

Аліна САВЧЕНКО

“ _____ ” _____ 2021 р.

ДИПЛОМНА РОБОТА

(ПОЯСНЮВАЛЬНА ЗАПИСКА)

ВИПУСКНИЦІ ОСВІТНЬОГО СТУПЕНЯ

“МАГІСТРА”

ЗА ОСВІТНЬО-ПРОФЕСІЙНОЮ ПРОГРАМОЮ “ІНФОРМАЦІЙНІ
УПРАВЛЯЮЧІ СИСТЕМИ ТА ТЕХНОЛОГІЇ”

**Тема: “Програмний комплекс створення online-меню для закладів
громадського харчування”**

Виконавиця: студентка групи УС-201Мз Молодоженя Діана Сергіївна

Керівник: к.т.н., доцент кафедри КІТ Райчев Ігор Едуардович

Нормоконтролер: _____ Ігор РАЙЧЕВ

Київ - 2021

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет кібербезпеки, комп'ютерної та програмної інженерії (ЗФН)

Кафедра Комп'ютерних інформаційних технологій

Галузь знань, спеціальність, освітньо-професійна програма: 12 “Інформаційні технології”, 122 “Комп'ютерні науки”, “Інформаційні управляючі системи та технології”

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Аліна САВЧЕНКО

« ____ » _____ 2021р.

ЗАВДАННЯ

на виконання дипломної роботи студентки

_____ Молодожені Діани Сергіївни

(прізвище, ім'я, по батькові)

- 1. Тема роботи:** «Програмний комплекс створення online-меню для закладів громадського харчування»
затверджена наказом ректора від 12.10.2021 за № 2229/ст.
- 2. Термін виконання роботи:** з 12.10.2021 по 31.12.2021.
- 3. Вихідні дані до роботи:** мова графічного опису для об'єктного моделювання UML, мова гіпертекстової розмітки HTML, CSS, JavaScript, Nest.js.
- 4. Зміст пояснювальної записки:** вступ, загальна характеристика та принципи побудови web-сервісів для створення online-меню, аналіз методів та технологій розробки програмного комплексу, реалізація програмного комплексу для створення QR-меню.
- 5. Перелік обов'язкового ілюстративного матеріалу:** рисунки, діаграми, презентація.

6. Календарний план-графік

№ п/п	Завдання	Термін виконання	Підпис керівника
1.	Отримання завдання на дипломну роботу та побудова плану-графіку виконання робіт.	12.10.2021 – 14.10.2021	
2.	Загальний аналіз роботи сервісів для створення online-меню.	15.10.2021 – 20.10.2021	
3.	Опис функціоналу, інтерфейсу, переваг та недоліків протестованих веб-сервісів.	21.10.2021 – 24.10.2021	
4.	Формування вимог до розроблюваного програмного комплексу.	25.10.2021 – 28.10.2021	
5.	Аналіз методів та технологій для розробки сервісу.	29.10.2021 – 04.11.2021	
6.	Створення діаграми варіантів використання та діаграми класів, формування потреб для розробки.	05.11.2021 – 14.11.2021	
7.	Розробка програмного комплексу.	15.11.2021 – 01.12.2021	
8.	Опис функціоналу програмного комплексу для створення online-меню.	02.12.2021 – 05.12.2021	
9.	Оформлення пояснювальної записки, створення презентації та доповіді.	06.12.2021 – 10.12.2021	

7. Дата видачі завдання: 12.10.2021р.

Керівник дипломної роботи: _____ Ігор РАЙЧЕВ
(підпис керівника)

Завдання прийняла до виконання: _____ Діана МОЛОДОЖЕНЯ
(підпис випускниці)

РЕФЕРАТ

Пояснювальна записка до дипломної роботи “Програмний комплекс створення online-меню для закладів громадського харчування” складається із вступу, трьох розділів, висновків, списку бібліографічних посилань використаних джерел і містить 89 сторінок тексту, 56 рисунків.

Об’єкт розробки – програмний комплекс створення online-меню для закладів громадського харчування.

Мета роботи – розробка програмного комплексу online-меню, який оптимізує роботу закладів харчування, зменшить витрати на паперове меню, мінімізує ризики зараження вірусом у часи пандемії, відмовившись від фізичного меню.

Об’єктом дослідження – програма, формування вебсторінки на основі введених даних, генерація QR-коду.

Предмет дослідження – вебсервіс для формування online-меню з генерацію QR-посилання на дане меню.

Перелік ключових слів: ПРОГРАМНИЙ КОМПЛЕКС, QR-КОД, ONLINE-МЕНЮ, ВЕБСЕРВІС, ВЕБСТОРИНКА, МЕНЮ.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ	7
ВСТУП	8
РОЗДІЛ 1 ЗАГАЛЬНА ХАРАКТЕРИСТИКА ТА ПРИНЦИПИ ПОБУДОВИ WEB- СЕРВІСІВ ДЛЯ СТВОРЕННЯ ONLINE-МЕНЮ	13
1.1. Загальний аналіз роботи сервісів для створення онлайн меню	13
1.2. Методи надання доступу до онлайн меню	15
1.3. Принципи створення шаблонів для оформлення меню	16
1.4. Аналіз web-сервісу qrcafe	17
1.5. Аналіз web-сервісу MeMenu	21
1.6. Аналіз web-сервісу NewMenu	26
1.7. Формування вимог до web-сервісу	33
Висновок до розділу 1	34
РОЗДІЛ 2 АНАЛІЗ МЕТОДІВ ТА ТЕХНОЛОГІЙ РОЗРОБКИ ПРОГРАМНОГО КОМПЛЕКСУ	35
2.1. Діаграма варіантів використання	35
2.2. Формування потреб для розробки	38
2.2.1. Оцінка якості продукту	39
2.2.2. Функціональні та нефункціональні вимоги	41
2.3. Діаграма класів	43
2.4. Клієнт-серверна архітектура	45
2.5. Технології розробки серверної частини	47
2.6. Технології розробки клієнтської частини	54
2.7. Валідація даних	57
Висновки до розділу 2	59
РОЗДІЛ 3 РЕАЛІЗАЦІЯ ПРОГРАМНОГО КОМПЛЕКСУ ДЛЯ СТВОРЕННЯ ONLINE-МЕНЮ	61
3.1. Реалізація цільової сторінки продукту для заохочення клієнтів	61

3.3. Формування вебсторінки меню та генерація посилання та QR-коду	80
Висновки до розділу 3.....	84
ВИСНОВКИ	85
СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ.....	88

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ

CLI	–	Command-line interface
CSS	–	Cascading Style Sheets
HEX	–	Hexidematic (позначення шістнадцятирічної системи числення)
HSL	–	Hue, Saturation, Lightness
HTML	–	HyperText Markup Language
HTTP	–	HyperText Transfer Protocol
JPG	–	Joint Photographic Experts Group
JSON	–	JavaScript Object Notation
npm	–	node package manager
PDF	–	Portable Document Format
QR	–	Quick Response
RGB	–	Red, Green, Blue
Sass	–	Syntactically Awesome Style Sheets
SCSS	–	Sassy CSS
SEO	–	Search engine optimization
SQL	–	Structured query language
UCD	–	User-Centred Design
UML	–	Unified Modeling Language
URL	–	Uniform Resource Locator
WWW	–	World Wide Web
БД	–	база даних
ООП	–	Об'єктно-орієнтоване програмування
РБД	–	реляційна база даних
СКБД	–	система керування базами даних

ВСТУП

Вже давно різноманітні пристрої стали постійними супутниками життя кожної людини. Поступово усі галузі впроваджують використання нових технологій, що значно пришвидшують та полегшують виконання певних процесів. До прикладу, на зміну довжелезним чергам в магазинах приходять онлайн замовлення з кур'єрською доставкою чи можливість оплати комунальних платежів за допомогою мережі Інтернет, онлайн запис до лікаря на визначений час, відкриття банківських рахунків без потреби фізичної присутності у відділенні банку, купівля квитків на літак чи поїзд за допомогою смартфона чи комп'ютера та багато іншого. Тобто, абсолютно усі сфери життя людини, від медицини до туризму, так чи інакше пов'язані з новими технологіями. Коли лише декілька років тому, вважалось, що сучасні технології актуальні лиш для молоді, то наразі технологічний прогрес стирає усі ці вікові стереотипи та навіть можна сказати вимагає від більш старшого покоління освоєння певного рівня.

Однак, пандемія COVID-19 наприкінці 2019 року сколихнула увесь світ та внесла значні корективи у життя людства. Обмеження торкнулися практично кожної сфери життя, особливо постраждала сфера туризму, малий і середній бізнес та й в загалом економіка країн.

Головним способом комунікації з навколишнім світом стали саме гаджети. Таким чином, за даними спільного звіту креативного агентства We Are Social та платформи Hootsuite, котрі проаналізували дані статистичних компаній, кількість часу, який люди проводять в інтернеті – зросла за період пандемії.

Відповідно до дослідження, в середньому типовий користувач інтернету проводить в онлайні майже 7 годин на день, з яких близько 2,5 годин йде на використання соціальних мереж.

Враховуючи жорсткі карантинні обмеження увесь світ, ніби раптово перейшов в онлайн режим. Усі комунікації, прогулянки, зустрічі, звичні, на перший погляд, речі повсякденного життя стали неможливими і щонайстрашніше небезпечними для здоров'я та життя.

При можливості усі організації переводили своїх співробітників на дистанційну роботу, навчання в школах та університетах відбувалося в онлайн форматі. Торгово-розважальні центри, готелі, ресторани та усі магазини, окрім продуктових, взагалі не мали змоги працювати. При таких умовах, вони мали два рішення або закрити бізнес взагалі, або приймати даний виклик та швидко реагувати на ситуацію, яка склалася. Більшість знайшли вихід саме за допомогою інформаційних технологій, створюючи web-сайти або застосунки, які надають можливість своїм клієнтам робити замовлення онлайн та за допомогою адресної доставки або замовлень на винос, намагаються продовжувати свою діяльність у даному форматі. Бажаючи того чи ні, увесь світ почав пристосовуватись до нових, доволі жорстоких реалій життя.

З плином часу та в залежності від ситуації та кількості захворюваності, карантин все ж таки послаблюють. Але при цьому життя не стає повністю звичним. Для того, щоб певним організаціям хоч якось функціонувати в звичному режимі, вони повинні виконувати ряд певних правил, таких як носіння маски, забезпечення соціальної дистанції в 1,5 м, наявність антисептиків, заповненість не більше ніж одна особа на 10 кв. метрів та інші.

Якщо детально розглянути правила роботи закладів громадського харчування в період карантину, то Міністерство охорони здоров'я України вимагає розміщення меню на вході закладу для безконтактного ознайомлення, або воно повинно бути зроблено з такого матеріалу, який дозволить його продезінфікувати, або дозволяється видача одноразового меню з їх подальшою утилізацією.

Та дані вимоги накладають ряд недоліків в органічній роботі закладу. Наприклад, дані вимоги можуть бути незручними для відвідувачів, які мають можливість ознайомитись з меню лише на вході, що потенційно може створювати чергу. Або до прикладу, дезінфікування меню після кожного відвідувача займає певний час, що впливає на швидкість роботи працівників закладу при високій заповненості. В такому випадку варіант одноразового меню здається ідеальним, співробітники не витратять час на дезінфікування меню, клієнт протягом відвідування закладу завжди має змогу оглянути його та можливо дозамовити щось

ще. Та головний недолік такого рішення полягає в постійному поповненні кількості меню та, звісно, в фінансових затратах на його друк.

Тобто, враховуючи усе вище сказане, можна дійти висновку, що виникає проблема, яким саме способом надавати меню для клієнтів. І постає завдання знайти зручне рішення для обох сторін, як для відвідувачів закладу, так і для його співробітників. До того ж дане рішення має бути економічно вигідним та обов'язково задовольняти запроваджені правила Міністерство охорони здоров'я України.

Чудовим варіантом вирішення даної проблеми буде можливість надати кожному клієнту переглянути меню онлайн у власному смартфоні за допомогою QR-коду. Та якщо, до прикладу, мережевим ресторонам легше втілити дану задумку в життя, то що стосується звичайних кафе, пекарень чи столових, то для них послуги ІТ-фахівця та затрати на розробку можуть бути фінансово непосильні.

Тому є потреба у програмному комплексі створення онлайн меню для закладів громадського харчування. Даний сервіс надаватиме користувачу змогу у зручному інтерфейсі додавати інформацію щодо наявних позицій, категоризувати їх та обрати стиль результуючої сторінки. Після введення усіх параметрів та збереження даних, користувачу надаватиметься посилання на його меню та автоматично сформований QR-код (для друку наліпок на стіл). Тобто, будь-яка людина зможе створити меню, без особливого володіння комп'ютером, а тим паче без навичок програмування, дизайну. А дане завдання можна делегувати на будь-якого працівника, адже рівень складності – мінімальний.

Актуальність теми полягає у необхідності допомогти та полегшити роботу ресторанного бізнесу в зручному представленні меню клієнтам і тим самим зменшити ризики для їх здоров'я в період пандемії, зменшивши контакт з предметами багаторазового користування. Оскільки меню є одним із найбрудніших предметів у закладах харчування, то доцільно вважати, що його онлайн формат може дійсно мінімізувати можливість зараження вірусом. А клієнт в будь-який момент може просто навести свій смартфон на наліпку з QR-кодом та отримати доступ до меню.

Головна перевага даного сервісу – це швидкість та легкість. Адже, щоб створити QR-меню для ресторану не потрібно витратити багато часу ні на пошук

фахівців, ні на роботу з сервісом. Весь процес займає лічені хвилини, при якому більшість часу необхідна на додавання позицій та їх детальної інформації.

Для того, щоб забезпечити потреби закладу харчування при створенні повноцінного меню, сервіс повинен надавати такі можливості:

- вибір дизайну;
- вибір категорії (перші страви / закуски / десерти та ін.);
- введення позицій меню згідно з категорією;
- введення ціни відповідної страви та при необхідності її вагу та інгредієнти;
- створення окремого блоку для інформування про акції або страви дня (при необхідності);
- надання сформованого QR-коду;
- надання посилання на створене меню;
- збереження результатів.

Однією з важливих функцій даного сервісу є можливість зміни мови, тобто клієнт закладу лише в один клік зможе отримати переклад меню англійською мовою. Це є значною перевагою для іноземних відвідувачів. А враховуючи те, що у порівнянні з паперовим меню потребувалось би два екземпляри меню українською та англійською мовами, дана функція є зручною та економічно вигідною.

На даний час на ринку представлена чимала кількість схожих програмних продуктів. Але в більшості випадків такі сервіси є платними і рідко містять весь функціонал та здебільшого мають незрозумілий інтерфейс.

Найкращим місцем зберігання даних щодо меню закладу громадського харчування є база даних.

Мета і завдання виконання дипломного проекту. Мета дипломного проекту – розробити програмний комплекс для створення онлайн-меню для закладів громадського харчування.

Для досягнення даної мети в дипломному проекті були поставлені такі завдання:

- 1) Огляд та аналіз існуючих вебсервісів в обраній області.

2) Проектування і розробка власного сервісу.

3) Створення легкого та інтуїтивно зрозумілого користувацького інтерфейсу.

Об'єкт і предмет проектування. Об'єкт дипломного проектування – створення онлайн меню для закладів громадського харчування. Предмет дипломного проектування – розроблення вебсервісу для оформлення онлайн меню та надання доступу для його перегляду.

Практичне значення дипломного проекту полягає у наданні можливості створення онлайн меню, перегляд якого можливий за допомогою посилання або QR-коду, без особливих навичок у сфері інформаційних технологій.

РОЗДІЛ 1

ЗАГАЛЬНА ХАРАКТЕРИСТИКА ТА ПРИНЦИПИ ПОБУДОВИ WEB-СЕРВІСІВ ДЛЯ СТВОРЕННЯ ONLINE-МЕНЮ

1.1. Загальний аналіз роботи сервісів для створення онлайн меню

Метою роботи над цим розділом є аналіз функціоналу декількох популярних web-сервісів у сфері створення меню для закладів громадського харчування.

Цей розділ описує загальні принципи роботи обраних додатків.

Меню – це перелік розташованих у певному порядку різних холодних і гарячих закусок, перших і других страв, гарячих і холодних напоїв, десертів, які пропонує заклад із зазначенням ціни, переліком компонентів та вагою.

Меню є візитною карткою ресторану та способом реклами. Тому його оформлення є важливим компонентом в організації роботи закладу. Меню повинно бути продуманим, інформативним й естетично привабливим.

У такому випадку зрозуміло, що реалізація функціоналу візуального представлення меню відіграє досить важливу роль. Тому користувачеві потрібно запропонувати варіанти вибору оформлення меню. Адже своїм оформленням меню підкреслює стиль, концепцію, загальну атмосферу та рівень закладу. Для того аби задовольнити усіх користувачів додатку, необхідно надати можливість коригувати дизайн для його персоналізації, наприклад, відповідно до корпоративних кольорів чи логотипу закладу. Даний функціонал пришвидшить роботу над створенням меню, максимально намагаючись забезпечити усі потреби користувача. При цьому це значно економить час, ніж створення повноцінного дизайну з нуля.

Кафедра КІТ (47)				НАУ 21 09 94 000 ПЗ			
Виконала	Молодоженя Д.С.			ЗАГАЛЬНА ХАРАКТЕРИСТИКА ТА ПРИНЦИПИ ПОБУДОВИ WEB-СЕРВІСІВ ДЛЯ СТВОРЕННЯ ONLINE-МЕНЮ	Літ.	Арк.	Аркушів
Керівник	Райчев І.Е.					13	22
Консульт.					УС-201Мз 122		
Н. контр.	Райчев І.Е.						

Слід також мати на увазі, що підприємства ресторанного господарства можуть відвідувати клієнти з дітьми. Тому варто доповнити перелік шаблонів ілюстративним матеріалом клієнтоорієнтованим на маленьких гостей закладу, або внести у перелік категорій – дитяче меню.

При оформленні меню слід надати можливість обирати фон, шрифт, розмір шрифту, колір.

Сформувавши візуальну концепцію меню, користувач повинен мати змогу для його наповнення. У такому випадку, для зручності доцільно було б запропонувати користувачеві вводити назви страв, напоїв відповідно до категорій. Отже, сервіс має передбачити перелік усіх можливих категорій і вже відповідно до цього обираються необхідні та заповнюються переліком страв. Даний інтерфейс є важливим для зручної структуризації меню одразу на початковому етапі створення. Таке групування страв допоможе краще орієнтуватись потенційним клієнтам.

Також повинна бути передбачена можливість додавання додаткового блоку для акційних чи сезонних пропозицій, страв від шеф-кухаря, режиму роботи закладу чи іншої додаткової інформації.

Після того, як користувач обрав дизайн, заповнив меню та зберіг, він має отримати до нього доступ. Найкращим варіант – це посилання на нього та сформований QR-код. Посилання на меню зручно розміщувати у соціальних мережах, за допомогою якого клієнтам буде зручно переходити на дану сторінку для перегляду. Що стосується QR-коду, то його розміщення більш доцільне безпосередньо у закладі, у вигляді наліпок на столах чи барних стійках.

Підбиваючи підсумки зазначимо, що при створенні сервісу важливо забезпечити усі можливі потреби користувача та надати можливість персоналізації. Тим самим створити платформу з легким та зрозумілим інтерфейсом, надавши користувачеві простий та поетапний алгоритм створення меню, який оптимізує ряд задач та зменшить тривалість виконання процесу.

1.2. Методи надання доступу до онлайн меню

Одна з найважливіших функцій сервісу – це представлення доступу до розробленого меню. Звичні для нас гіперпосилання доцільно мають місце бути, адже їх вигідно розміщувати у соціальних мережах чи інших інтернет ресурсах. Але враховуючи ситуацію з пандемією та карантинними обмеженнями, потрібно надати доступ до меню безпосередньо кожному клієнту, наприклад, роздруковувати посилання та розмістити його у паперовому вигляді на столиках чи стенді є поганою ідеєю. Адже водити цей URL-адрес клієнтові буде некомфортно, і враховуючи, що кількість символів може бути чималою, то такий процес займатиме достатньо багато часу.

Тому в такому випадку QR-код є чудовою безконтактною альтернативою. На відміну від старого штрих-коду, який сканують тонким променем, QR-код визначається сенсором як двовимірне зображення. Основна перевага QR-коду — це легке розпізнавання сканувальним обладнанням, в тому числі і фотокамерою мобільного телефону. Це дає можливість його використання в логістиці, торгівлі та на виробництві. Переважно в Україні використовують QR-коди для зчитування інформації про квитки на залізничний транспорт, контактної інформації, реклами, для мультифакторної аутентифікації, тобто безпечного входу користувача в його акаунти (наприклад, Приват24) та багато іншого.

Готовий QR-код являє собою зображення в форматі JPG, яке можна роздрукувати на будь-якій поверхні (папір, тканина, чашка, пластик тощо), розмістити на своєму вебресурсі, або просто відкрити на своєму смартфоні та показати іншим.

Наявність смартфонів у більшості населення, легкість зчитування та кодування стали причиною популярності активного використання в повсякденному житті QR-кодів. Враховуючи усі переваги QR-кодування, сервіс повинен мати функцію представлення доступу до меню у даному вигляді.

1.3. Принципи створення шаблонів для оформлення меню

Як було зазначено раніше, естетична складова меню є досить важливою. Для того, щоб представлені шаблони відповідали потребам користувача, потрібно детально ознайомитись з правилами оформлення меню. Вона повинно задовольняти ряд критеріїв:

- читабельність;
- зручність;
- грамотність;
- приємна візуалізація.

Не варто використовувати яскраву кольорову гаму, краще надавати перевагу більш нейтральним та стриманим тонам. Чудово, коли меню підкреслює стиль закладу. Тому добре було б передбачити, що користувачі сервісу можуть розроблювати меню як для закладу, що спеціалізується на італійській, японській, європейській чи українській кухні і підготувати макети, що враховують дані особливості. Або інший варіант – намагатись створити підбірку універсальних шаблонів, не орієнтуюсь на конкретну концепцію.

При складанні меню існує певний стандартний порядок розташування страв, який варто дотримуватись. Врахувавши ці дані, сервіс має виділити та запропонувати ряд категорій. Отже, опрацювавши даний матеріал, сервіс має надавати такий перелік категорій.

- 1) Холодні закуски.
- 2) Салати.
- 3) Гарячі закуски.
- 4) Перші страви.
- 5) Основні страви.
- 6) Гарніри.
- 7) Десерти.
- 8) Напої.

До того ж було добре передбачити можливість додавання нової категорії. Наприклад, для акційних пропозицій чи фірмових страв, що являються родзинкою закладу і користуються попитом.

В створенні шаблонів повинно бути передбачено внесення при необхідності інформації щодо ціни, складу страви, маси готового продукту чи калорійності.

Шаблон повинен містити напрацювання структури, графічного контенту, бути продуманим та універсальним. Добре, коли шаблони достатньо різноманітні і враховуючи те, що його можна частково змінити під свої потреби, користувач в результаті має можливість створити повноцінно оформлене меню без навиків дизайну. Готові варіанти стилістичного оформлення та розташування інформації значно полегшують та оптимізують процес виконання.

1.4. Аналіз web-сервісу qrcafe

Платформа qrcafe пропонує забезпечення безконтактного сервісу завдяки онлайн меню через QR-код і з економію коштів на друкуванні паперового варіанту меню.

Web-сервіс надає можливість створити персоналізований дизайн, тобто використовувати власний фон та корпоративні кольори. Головна перевага сервісу – це надання послуг безкоштовно.

Для створення меню користувач переходить у адмін-панель та для початку вказує усю необхідну інформацію що стосується закладу та його роботи (Рис.1.1) та завантажити логотип ресторану, зображення обкладинки та обрати шаблон меню.

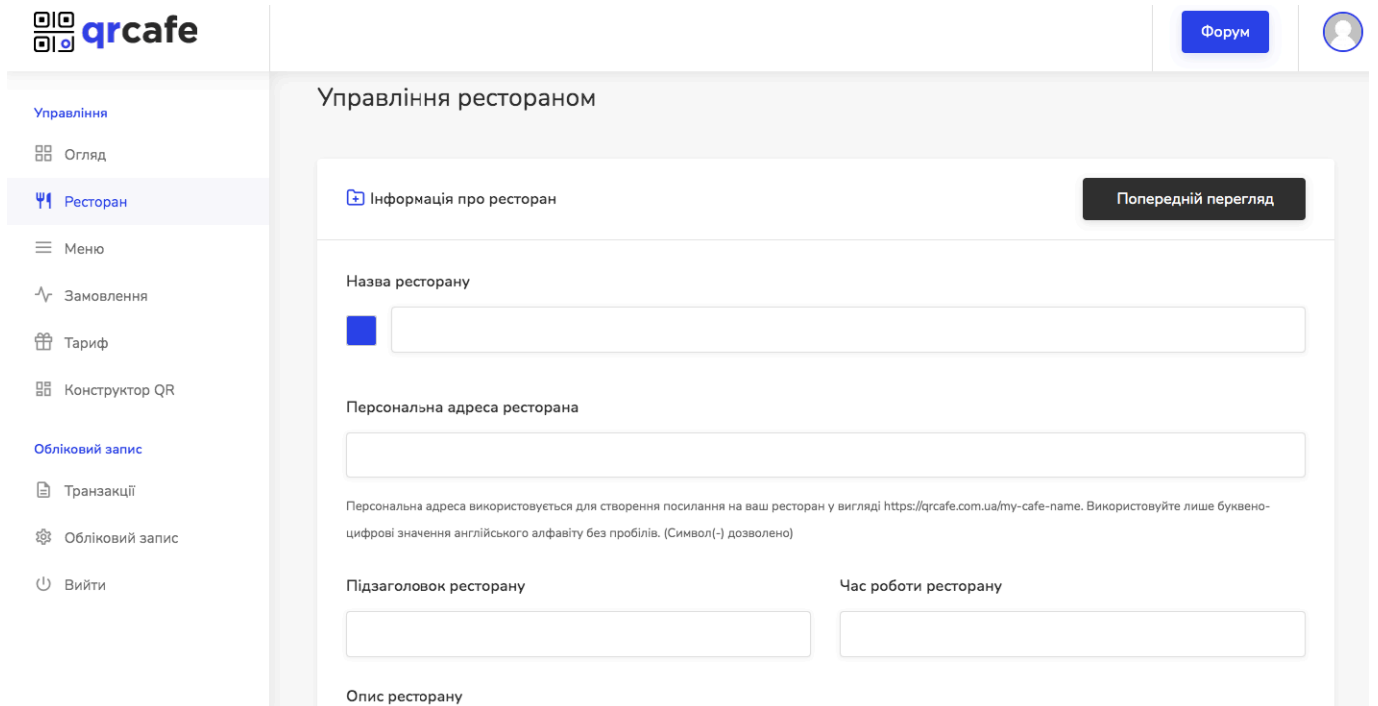


Рис. 1.1. Вкладка “Ресторан” адмін-панелі користувача

Далі на вкладці “Меню” (Рис.1.2) користувач має змогу додати перелік категорій, відповідно кожній категорії додати позиції меню за допомогою форми (Рис.1.3), де вказуються інформація щодо назви, опису, вартості страви та чи являється вона вегетаріанською і є можливість завантажити зображення.

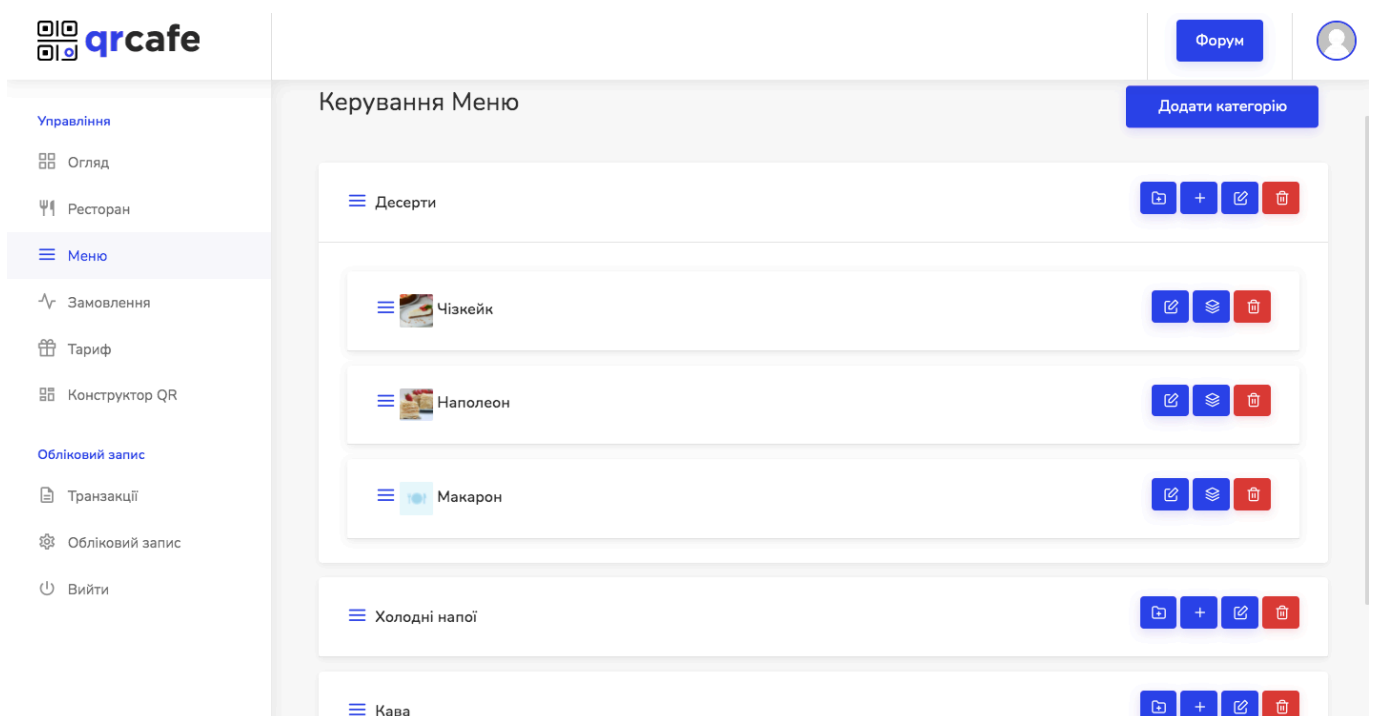


Рис. 1.2. Вкладка “Меню” адмін-панелі користувача

Додати страву ×


Назва страви

Опис страви

Вартість страви

Вегетаріанська

Зображення страви



[Завантажити зображення](#)

Рис. 1.3. Форма для додавання позиції меню

Після заповнення усіх даних, користувач переходить на вкладку “Конструктор QR” та має можливість простилізувати уже сгенерований сервісом QR-код (Рис.1.4).

Дана функція надає можливість зміни кольору коду або фону, вказати відступи від краю та радіус кута. На цьому етапі користувач має змогу завантажити QR-код для подальшого його використання у власному закладі.

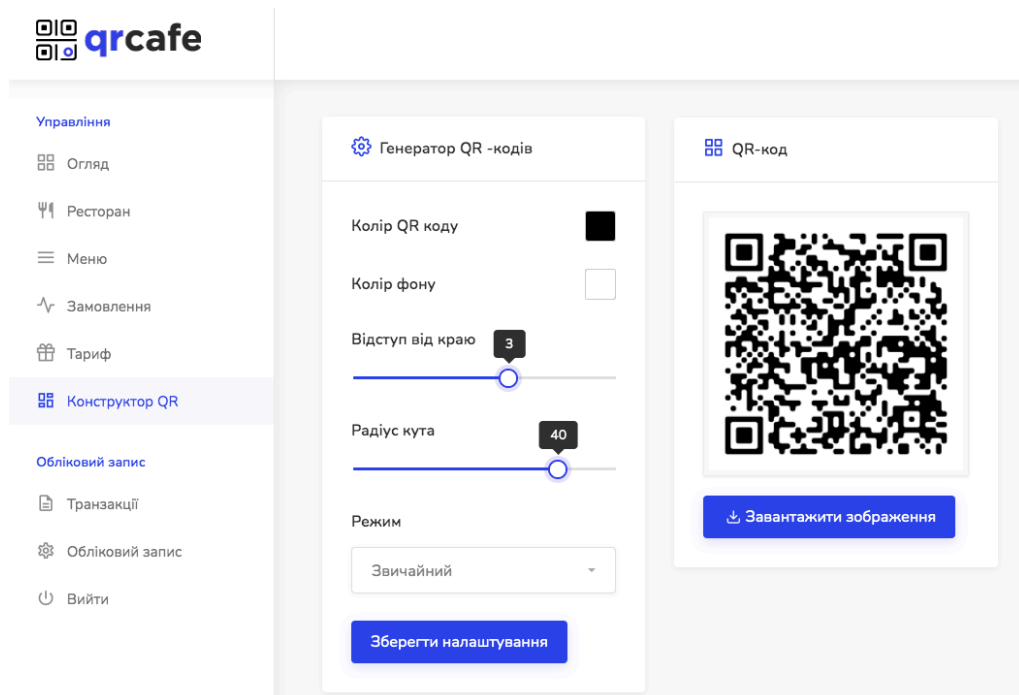


Рис. 1.4. Вкладка “Конструктор QR” адмін-панелі користувача

Протестувавши модуль створення меню та ввівши випадкові дані, відскануємо QR-код та перейдемо за посиланням для перегляду сформованого сервісом цифрового меню (Рис.1.5).

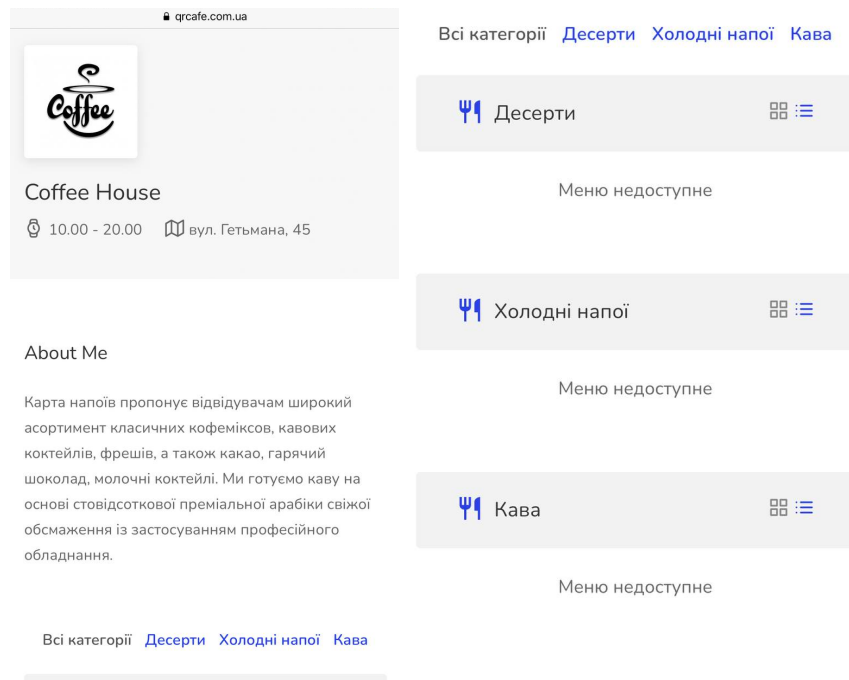


Рис.1.5. Web-сторінка з сформованим цифровим меню

Кінцевий результат меню викликає ряд питань, назва блоку “About Me” є не зовсім доречною, не відображаються позиції меню, список категорій не є клікабельним. Тобто, враховуючи значний технічний недолік, що сервіс не відображає вказані страви, то можна стверджувати, що дана платформа зі своєю головною задачею не справляється.

З переваг можна виділити клікабельне посилання адреси, що показує розташування в GoogleMaps, відображення логотипу, чітко виділені категорії, показ опису ресторану.

В підсумку, результат роботи сервісу має негативну оцінку і потребує значних доопрацювань та усунення неполадок. Що стосується інтерфейсу створення самого меню, додавання категорій та страв, то він є доволі зручним та інтуїтивно зрозумілим, єдине можливо варто внести в форму для додавання страв поле для вказування маси або ємності.

1.5. Аналіз web-сервісу MeMenu

Сервіс MeMenu пропонує доповнити меню додатковими опціями. Користувач завантажує власне повністю оформлене меню з фотографіями страв, цінами, описом в форматі PDF, і сервіс надає такі послуги, як:

- створений QR-код для його сканування клієнтом і завантаженням електронного меню на телефон;
- можливість додати посилання на акаунти у соціальних мережах Facebook та Instagram;
- можливість додати контактні дані Viber, Whatsapp, Telegram;
- підключення системи відгуків;
- відслідковування статистики відвідувань;
- можливість завантаження більше одного меню.

MeMenu позиціонує себе як швидкий та легкий у використанні і щонайголовніше безкоштовний сервіс. Та переходячи до безпосередньо користування та створення QR-меню, користувачеві пропонуються два тарифні плани. Дійсно, є безкоштовний тариф та тариф Оптима за 20\$ на місяць. На перший погляд тарифи досить схожі між собою і не мають суттєвої різниці. Обидва тарифи дають можливість створення QR-меню із PDF, перегляду відгуків, статистику та інше. Але ознайомившись детально, безкоштовний тариф має суттєвий недолік – вбудовану рекламу.

Розглянемо це збоку відвідувача закладу, він приходить до закладу, сканує QR-код, відкриває електронне меню та бажає ознайомитись з представленими стравами та зробити замовлення. Тобто, клієнт має конкретну ціль – обрати та замовити їжу і тут присутній відволікаючий фактор у вигляді реклами, будь то банер чи певний інтегрований текст, посилання чи щось інше це мимовільно привертає увагу клієнта.

Головна задача це сконцентрувати увагу клієнта на меню так, щоб зацікавити у максимальному виборі страв, що напряду впливає на доходи закладу. Наявність сторонньої реклами може стати відволікаючим елементом або дратуючим чинником.

Це може затримувати клієнта під час перегляду та вимагати додаткових дій від нього, наприклад закриття реклами.

Отже, враховуючи усі моменти можна зробити висновок, що безкоштовний тарифний план має суттєвий недолік, що може негативно вплинути у подальшому на загальне враження відвідувачів від закладу та на їх замовлення, наприклад зменшивши потенційну можливу кількість позицій у ньому.

Як вже було зазначено, сервіс на базі готового меню у форматі PDF формує QR-меню, тобто сайт не надає можливості створити та оформити його. Та за додаткові кошти пропонує послуги контент-менеджерів, які внесуть усі позиції та дані щодо меню. Важливо зазначити, що ця послуга оплачується додатково, незалежно від того, який тарифний план ви використовуєте. Тобто, уже на даному етапі, такі вказані переваги, як безкоштовний сервіс, економія фінансів користувачів, збільшення об'ємів продажів є доволі сумнівними.

Проаналізуємо безпосередньо функціонал сервісу. Для створення QR-меню необхідно заповнити форму (Рис.1.6), де вказати назву та тип компанії і завантажити меню в PDF форматі. Варто відразу звернути увагу, що вибір мови на сайті працює некоректно.

MEMENU

РЕСТОРАНИ QR МЕНЮ КОНТАКТИ ПІДКЛЮЧИТИ БІЗНЕС ПРОФІЛЬ UA

Название компании

Company type

Ресторани

Завантажте ваше меню в форматі PDF

Максимальний розмір файлу 30 Mb

Зберегти

Не виходить завантажити? Можливо, ваш файл іншого формату. Щоб конвертувати в PDF, перейдіть [сюди](#).

Рис. 1.6. Форма для заповнення даних

Після заповнення форми даними, користувач потрапляє в адмін панель (Рис.1.7), де має змогу завантажити логотип свого закладу, додати посилання на соціальні мережі та месенджери, ввести дані Wi-Fi мережі закладу та увімкнути систему відгуків.

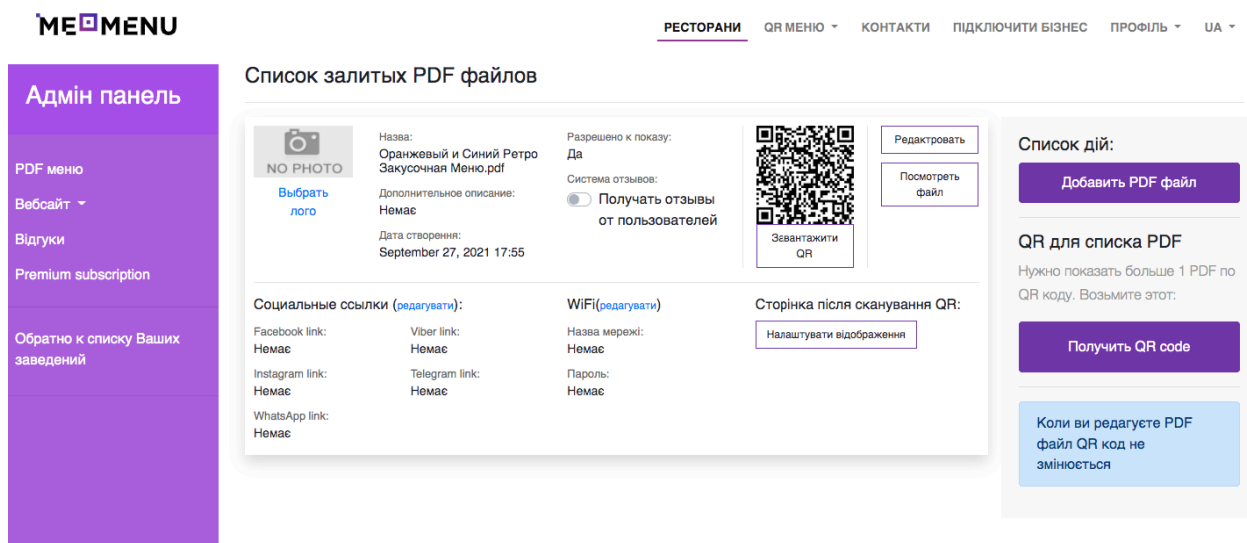


Рис. 1.7. Адмін-панель користувача

Після заповнення усіх даних, користувач має змогу завантажити QR-код для його подальшого використання. За допомогою кнопки “Добавить PDF файл” можна додавати інші меню.

Навівши камеру мобільного телефону на сформований сервісом QR-код, клієнт має змогу перейти за посиланням (Рис.1.8).



Рис. 1.8. Скріншот процесу сканування сформованого QR-коду

Відкривши посилання, клієнт не відразу має можливість переглянути меню. Він потрапляє на web-сторінку, де бачить рекламу і має змогу натиснути кнопку “Переглянути меню” (Рис.1.9), за допомогою якої клієнт вже безпосередньо матиме змогу ознайомитись з меню. Або якщо користувач підключив систему відгуків, то є можливість також натиснути на відповідну кнопку для того, щоб залишити відгук про кухню атмосферу чи сервіс закладу.

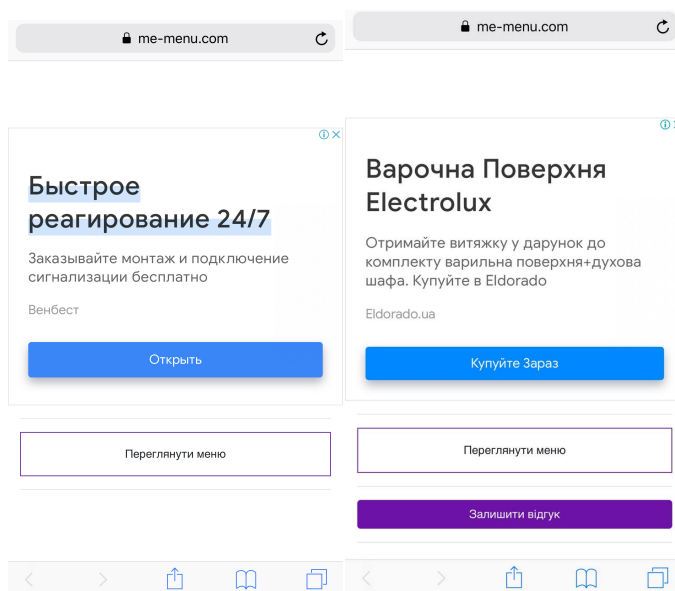


Рис. 1.9. Сторінка після переходу за посиланням з відключеною та підключеною системою відгуків

Для того, щоб залишити відгук про заклад, клієнт може заповнити та відправити форму (Рис.1.10), де має можливість поставити оцінку (кількість зірочок), вказати своє ім'я та написати більш розгорнутий відгук.

The screenshot shows a web browser window with the address bar displaying 'me-menu.com'. The website header includes the 'ME MENU' logo and a hamburger menu icon. The main heading is 'Залишити відгук' (Leave a review). Below the heading is a five-star rating system with all stars filled. There are two text input fields: 'Ваш отзыв' (Your review) and 'Ваше Имя' (Your name). At the bottom of the form is a purple button labeled 'Добавить отзыв' (Add review).

Рис. 1.10. Форма для залишення відгуку

Лише після того як клієнт залишив відгук, йому пропонується переглянути сторінку закладу (Рис.1.11).

The screenshot shows the same browser window as in Figure 1.10. The main heading is 'Спасибо за ваш отзыв. Ваше мнение очень важно для нас' (Thank you for your review. Your opinion is very important for us). Below this is a smaller heading: 'Предлагаем вам ознакомиться с рестораном поближе, на этой странице' (We offer you to get acquainted with the restaurant more closely on this page). At the bottom is a purple button labeled 'Страница заведения' (Establishment page).

Рис. 1.11. Посилання на сторінку закладу

На даній сторінці (Рис.1.12) можуть бути вказані посилання на соціальні мережі, режим роботи та інша інформація, що стосується роботи закладу. Залишивши три відгуки, тим самим протестувавши систему відгуків, було виявлено, що відгуки на сторінці закладу не з'явилися. А кількість вказаних вподобань і зірок закладу є сталою і формується незалежно від оцінювання клієнтів. Тобто даний модуль працює некоректно і потребує значного доопрацювання.

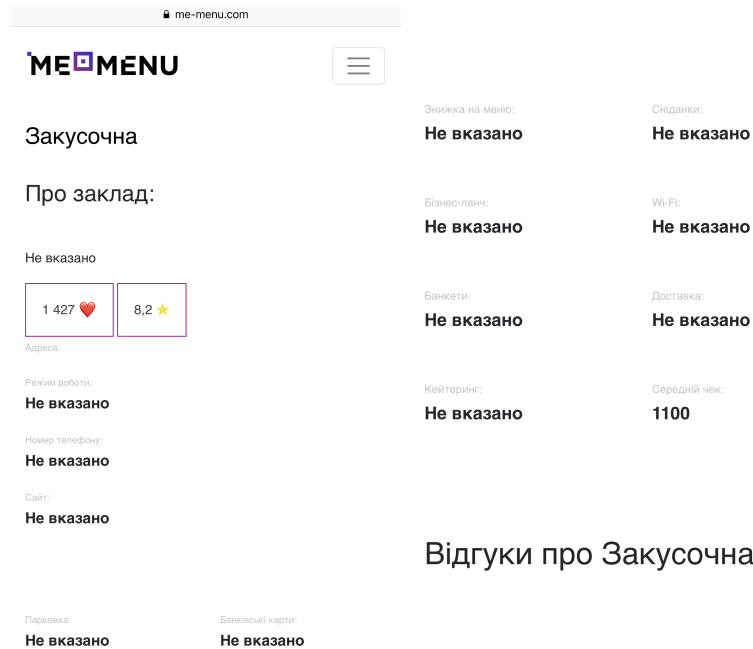


Рис. 1.12. Сторінка закладу

Узагальнюючи результати аналізу сервісу MeMenu, можна зробити наступні висновки, що перевагою розглянутого додатку є можливість завантаження більше одного меню, внесення додаткової інформації щодо роботи закладу та самий задум системи відгуків.

Але деякі з цих переваг потребують доопрацювання, адже додаткову інформацію можуть переглянути лише ті клієнти, які залишили відгук. А сама система відгуків працює некоректно, не зрозуміло за якими даними формуються оцінка і вподобання та не відображаються залишені відгуки.

Враховуючи усі недоліки, сервіс не відповідає заявленому функціоналу і потребує удосконалення бізнес-логіки та усунення технічних неполадок.

1.6. Аналіз web-сервісу NewMenu

Платформа NewMenu пропонує можливість створення меню та QR-коду. Користувачі сервісу можуть обрати для себе відповідний тарифний план: безкоштовний – сервіс рекомендує його для невеликих ресторанів чи кафе, або тариф Pro, що є платним та призначений для меню з великою кількістю позицій. Відмінності

даних тарифних планів полягають лише у ліміті позицій у меню та відсутності реклами у тарифі Pro.

Роботу з даним сервісом можна розділити на два етапи. Спочатку користувач створює цифрове меню, і для початку йому потрібно заповнити форму (Рис.1.13).

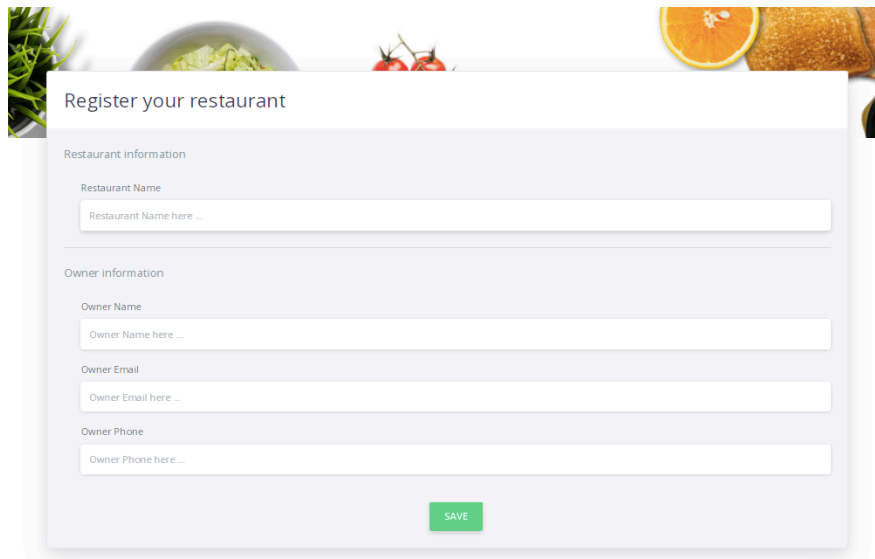
A registration form titled "Register your restaurant" is displayed over a background image of various food items. The form is divided into two sections: "Restaurant information" and "Owner information". Under "Restaurant information", there is a text input field for "Restaurant Name" with the placeholder text "Restaurant Name here ...". Under "Owner information", there are three text input fields for "Owner Name", "Owner Email", and "Owner Phone", each with its respective placeholder text. A green "SAVE" button is located at the bottom center of the form.

Рис. 1.13. Форма для заповнення інформації щодо ресторану та власника

Заповнивши реєстраційну форму, користувач переходить до інтерфейсу оформлення і заповнення меню (Рис.1.14). Платформа надає можливість вказати часи роботи закладу, його опис, адресу, валюту, завантажити логотип ресторану та обрати фонове зображення для меню.

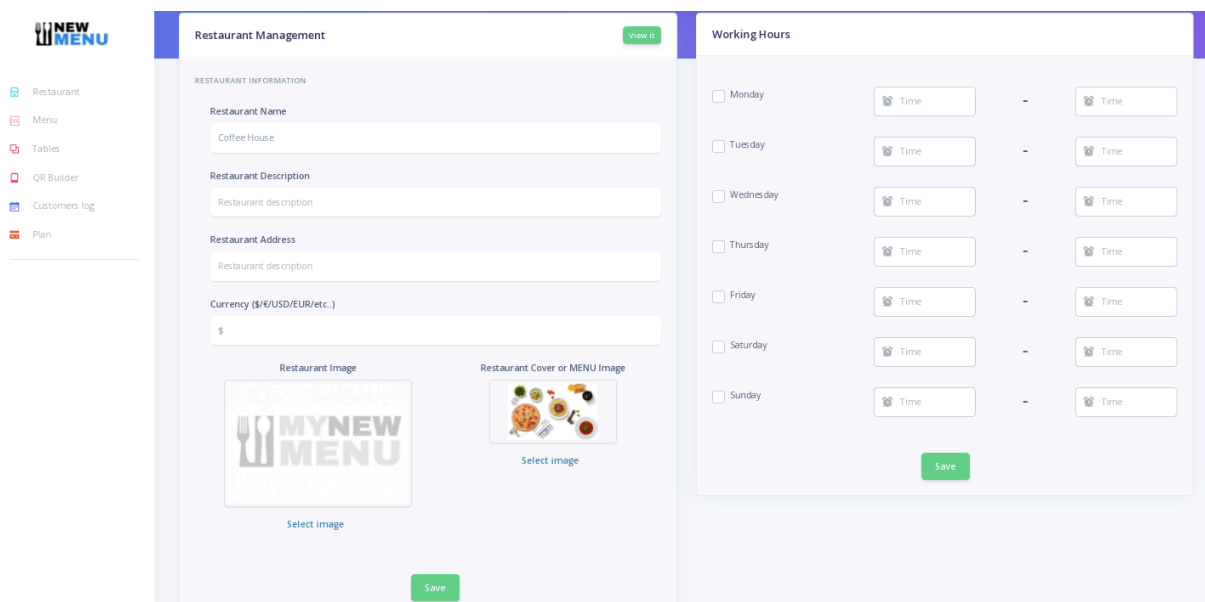
The user management panel is titled "NEW MENU" and is divided into two main sections: "Restaurant Management" and "Working Hours". The "Restaurant Management" section includes fields for "Restaurant Name" (filled with "Coffee House"), "Restaurant Description", "Restaurant Address", and "Currency (\$/€/USD/EUR/etc.)" (filled with "\$"). It also features two image upload areas: "Restaurant Image" and "Restaurant Cover or MENU Image", each with a "Select image" button. A green "Save" button is at the bottom. The "Working Hours" section has a list of days from Monday to Sunday, each with a checkbox and two time selection fields. A green "Save" button is at the bottom. A left sidebar contains navigation links for "Restaurant", "Menu", "Tables", "QR Builder", "Customers log", and "Plan".

Рис. 1.14. Адмін-панель користувача

Ввівши інформацію, що стосується закладу, користувач може перейти на вкладку створення безпосереднього самого меню (Рис.1.15). За допомогою даного інтерфейсу користувач має змогу створити перелік необхідним йому категорій.

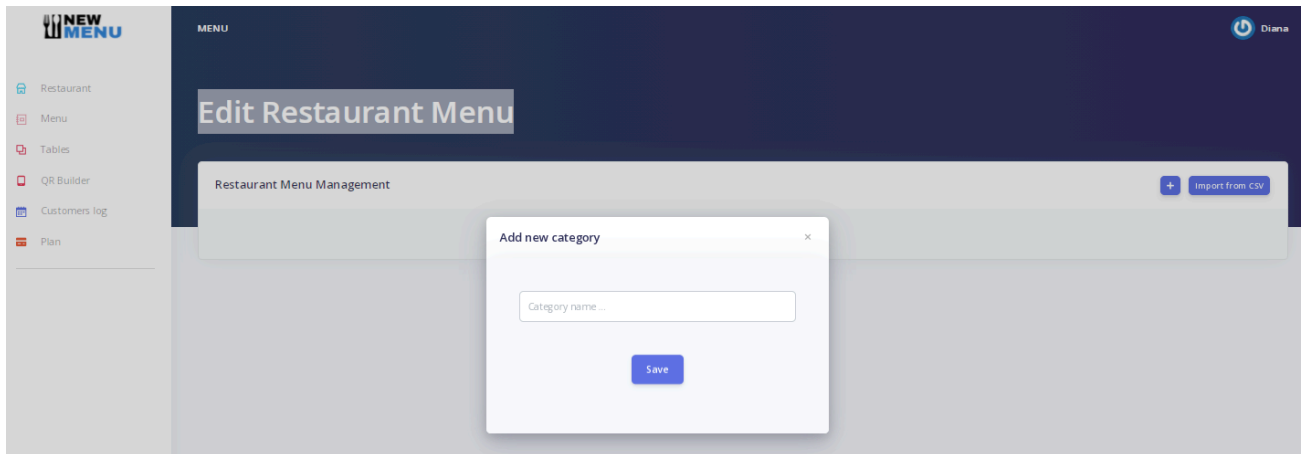


Рис. 1.15. Адмін-панель користувача створення категорій меню

Далі за допомогою форми, користувач має змогу внести позиції меню відповідно до обраної категорії (Рис.1.16). Є можливість вказати назву, опис, ціну та завантажити зображення страви.

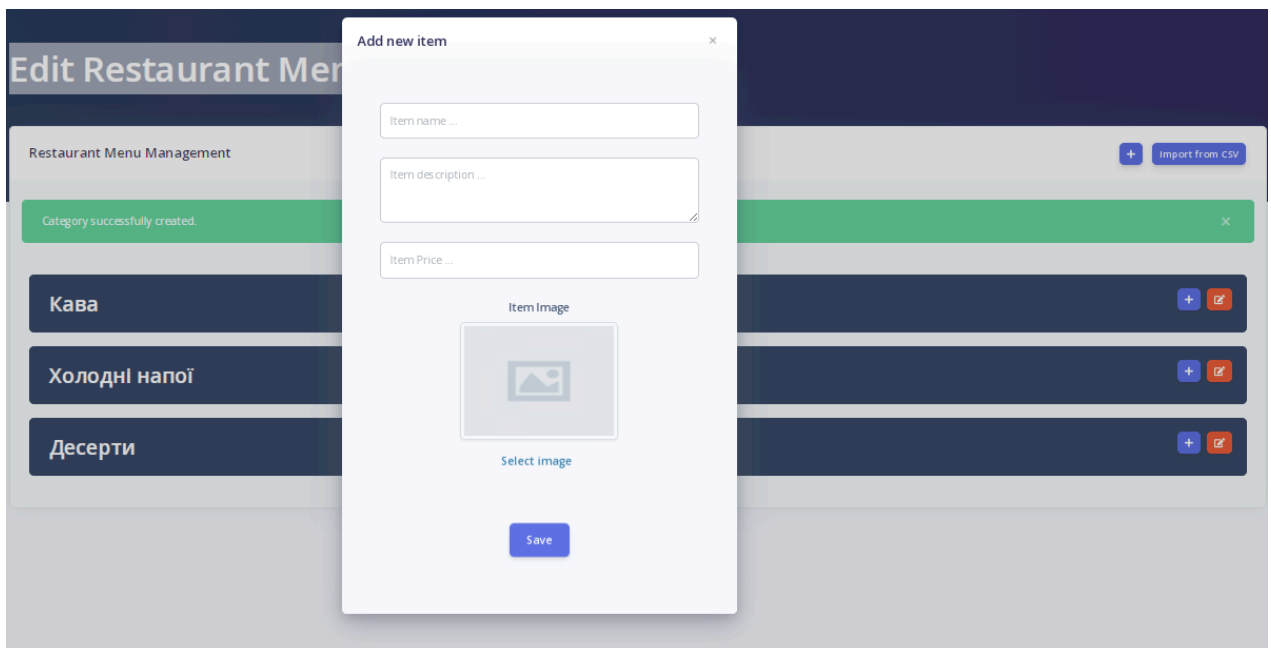


Рис. 1.16. Форма для внесення інформації щодо страви

На рисунку 1.17 вигляд інтерфейсу користувача із створеними категоріями та добавленими позиціями меню. При натисканні на будь-яку позицію, відкривається вікно для можливості редагування інформації. При заповненні даними, було вирішено в деяких випадках заповнювати не усі дані, тобто десь не вказувати опис чи не завантажувати зображення, щоб переглянути як це вплине на відображення кінцевого результату.

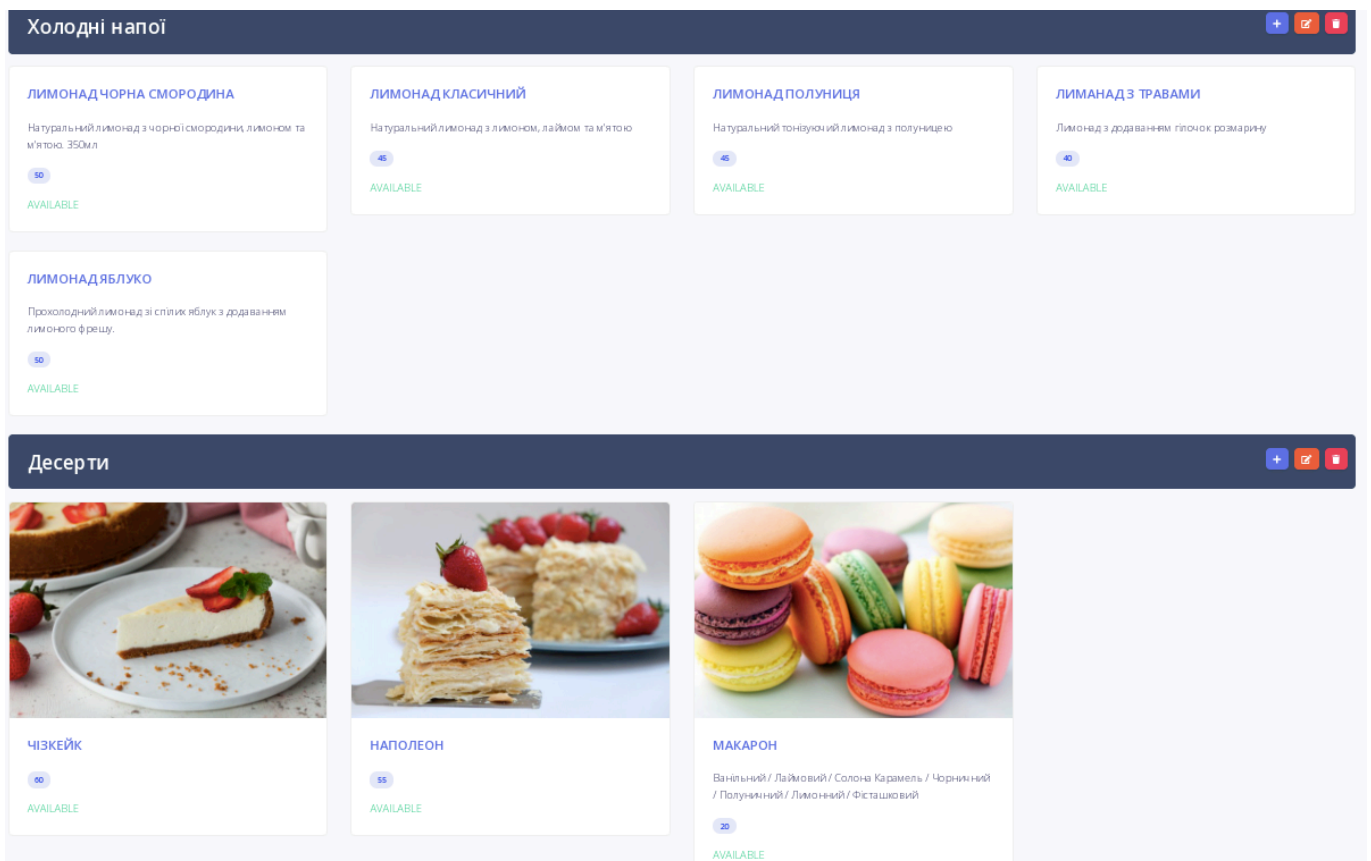


Рис. 1.17. Інтерфейс користувача із заповненими даними категорій та позицій меню

Після того як внесено усю необхідну інформацію для оформлення цифрового меню, користувач переходить на вкладку “QR Builder” (Рис.1.18). Дана опція дає можливість стилізувати уже автоматично сформований сервісом QR-код. Платформа NewMenu надає вибір з восьми видів стилізація коду та можливість зміни кольорової гами.

Дана функція стилізації QR-коду є доволі цікавою та дає можливість персоналізації, але вона не є обов’язковою в функціональному плані і несе більш естетичну функцію ніж практичну.

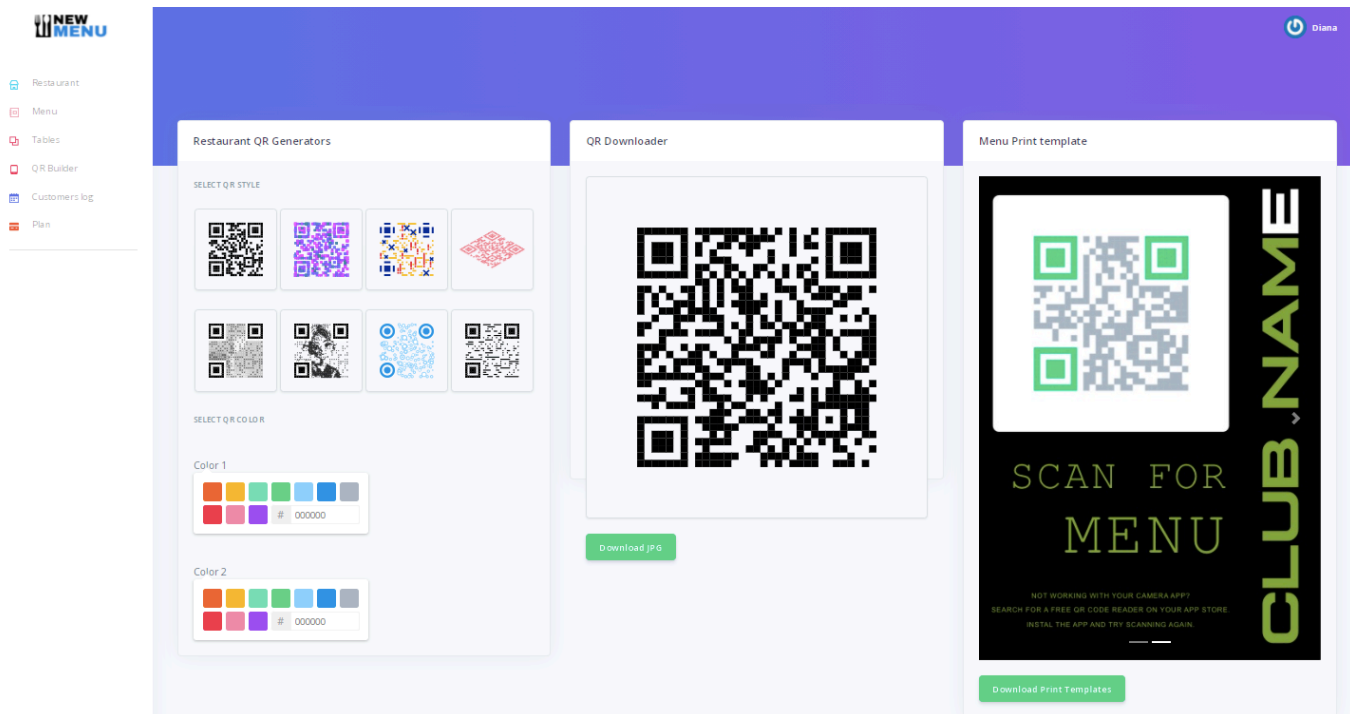


Рис. 1.18. Вкладка “QR Builder”

Отже, виконавши дії та заповнивши дані, користувач може відсканувати даний QR-код та переглянути кінцевий результат своєї роботи, який будуть бачити клієнти його закладу.

При переході за посиланням, клієнт бачить назву закладу, та його опис (Рис.1.19). До того ж опис займає досить багато місця, навіть не зважаючи на те, що складається з двох речень, можливо варто зменшити шрифт, або створити можливість згорнути і розгорнути текст, якщо він перевищує певну кількість символів.

Після опису йде перелік категорій. Даний елемент виглядає ніби неадаптивним для мобільних версій, хоча якщо категорій велика кількість є можливість скролу вправо чи вліво. Але за дизайном ця можливість не є інтуїтивно зрозумілою, тому варто розглянути інші варіанти відображення даної інформації.

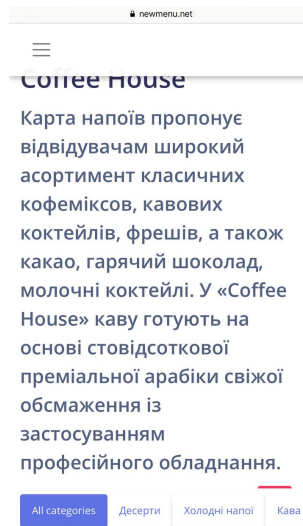


Рис. 1.19. Web-сторінка сформованого меню

Гортаючи вниз, клієнт може ознайомитись з переліком страв відповідних категорій (Рис.1.20). В цілому коректно відображена уся введена інформація відповідно до категорій, але щодо візуально фактору, то є певні зауваження. Стилістичне оформлення меню потребує змін збоку web-сервісу. У разі відсутності зображення страви, просто список виглядає ніби не довершеним і потребує кращої стилізації. Але є один недолік у тому, що завантажені логотип закладу та фон не відображаються. Тобто, змога персоналізувати меню надається, але технічно вона не працює.

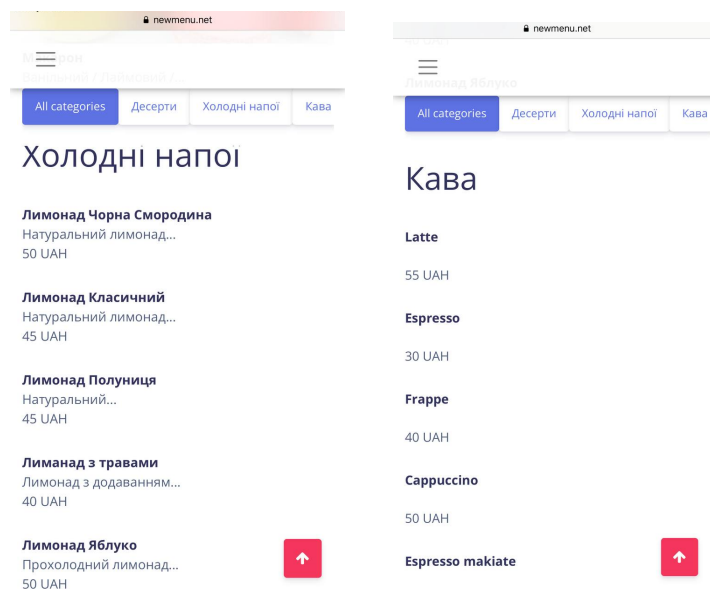


Рис. 1.20. Перелік позицій меню відповідно до категорій

Натиснувши на будь-яку позицію меню, відкривається вікно з розгорнутим описом страви (Рис.1.21). На рисунку наглядно видно, що у разі бажання зробити меню без додавання фото контенту, то показується дефолтне зображення з логотипом сервісу NewMenu. Тобто візуального оформлення у разі відсутності картинки не передбачено, що, звісно, є доволі значним недоліком. Адже в такому випадку меню здається недовершеним.

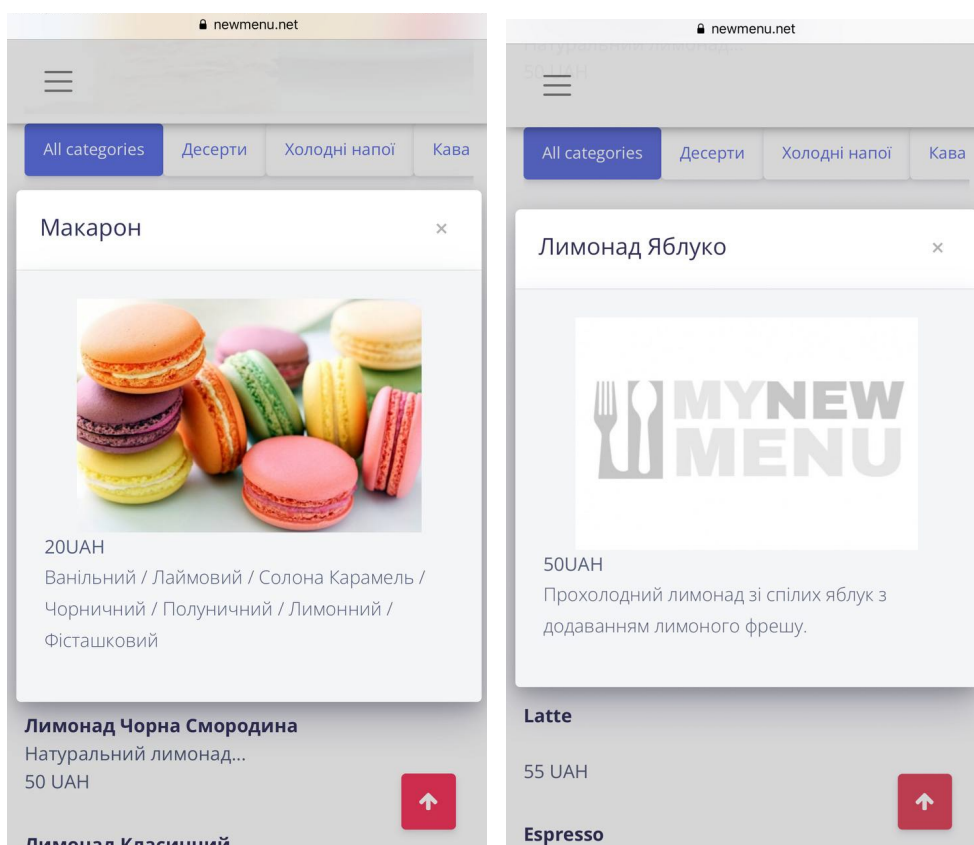


Рис. 1.21. Вікно з детальним описом позиції меню

Проаналізувавши даний web-сервіс можна виділити його основну перевагу, це не просто створення QR-коду, а можливість створити та оформити цифрове меню. При тому платформа має зручний, структурований інтерфейс створення цифрового меню в плані додавання, категорій, позицій та їх редагування. В цілому є усі необхідні елементи для вводу даних, але плюсом була б можливість додаткового введення маси або ємності страви.

Сервіс має ряд технічних недоліків, таких як не відображення логотипу закладу та фону меню, відсутність можливості створювати декілька меню, наприклад, під мережу закладів з різною концепцією. Але основним суттєвим недоліком є візуальне відображення сформованого меню, яке надається клієнтам закладу. Даний інтерфейс важко назвати інтуїтивно зрозумілим, відсутня будь-яка персоналізація закладу, оформлення web-сторінки не приносить естетичного задоволення. Відчувається ніби власники, хотіли з економити на дизайні. Тому сервісу NewMenu також варто звернути увагу на дизайн, позиціонування елементів, відступи між елементами, розмір шрифтів. Розглянути відображення позицій у випадку різного заповнення даними користувачами. В загалом реалізація продукту потребує значних коректив як в плані зручності користувацького інтерфейсу так і в плані технічної оптимізації.

1.7. Формування вимог до web-сервісу

Детально проаналізовані існуючі засоби для формування цифрового меню та QR-коду дали розуміння, який функціонал очікують користувачі та які основні недоліки наявні у існуючих web-сервісах.

З огляду на основний функціонал розглянутих засобів можна сформуванати базовий перелік розділів, котрі повинні бути наявні у розроблюваному web-сервісі. Зокрема повинен бути реалізований розділ формування меню та його персоналізації, створення категорій, додавання позицій. Тобто алгоритм оформлення меню має бути комбінованим: користувач повинен мати можливість обрати з поміж декількох варіантів стилізацію меню, або мати змогу завантажити власні зображення та використовувати корпоративні кольори, використовуючи зручний інтерфейс.

Продуманість інтерфейсу внесення інформації для створення меню відіграє важливу роль у зручності сервісу, адже саме він формує перше враження користувача.

Сервіс повинен мати функцію редагування, щоб користувач з легкістю міг внести поправки при виникненні такої потреби. Важливо також розробити модуль генерації QR-коду з можливістю його завантаження для користувача.

Web-сторінка з результатом готового меню повинно відображатись у зручній формі для клієнтів закладу, візуально приємно розміщувати усю внесену інформацію, коректно відображати ілюстрації. Перелік категорій та позицій меню має бути читабельним та гарно простилізованим. Дизайн має бути оформлений таким чином, щоб привертати увагу клієнта до потрібної інформації.

Сформований, на основі детального аналізу існуючих web-сервісів, перелік вимог дає чітке розуміння, які складові інтерфейсу мають потребу у детальному проектуванні та яким потребам користувачів вони повинні відповідати.

Висновок до розділу 1

В рамках роботи над першим розділом, були проаналізовані існуючі платформи для створення цифрового меню та генерування QR-коду. Зокрема, були розглянуті наступні web-сервіси: «qr-safe», «MeMenu», «NewMenu». Незважаючи на заявлені ними переваги та функціональні можливості, їх детальний аналіз дав можливість сформулювати перелік основних недоліків, та відповідно вимоги до інтерфейсу розроблюваного програмного комплексу.

Основним недоліком розглянутих web-сервісів є непродуманість та важкість у використанні інтерфейсу та некоректна реалізація з технічної сторони. У деяких сервісах відсутня можливість створення цифрового меню, вони формують QR-код лише на основі готового меню, яке завантажує користувач у PDF форматі. Кінцевий результат готового меню, який надається безпосередньо для перегляду клієнтам закладу, зазвичай має недоліки у візуальному відтворенні і працює некоректно.

Спираючись на виявлені недоліки та сильні сторони існуючих платформ, буде розроблено інтерфейс і функціонал проекту.

РОЗДІЛ 2

АНАЛІЗ МЕТОДІВ ТА ТЕХНОЛОГІЙ РОЗРОБКИ ПРОГРАМНОГО КОМПЛЕКСУ

2.1. Діаграма варіантів використання

Діаграма варіантів використання (Use Case Diagram) – це графічне представлення можливостей користувача при взаємодії з системою. Варіанти використання представлені еліпсами. Актори часто зображені у вигляді фігурок.

Розробка діаграми варіантів використання є важливим етапом розробки програмного продукту, яка дає розуміння функціональності і поведінки системи. Дана діаграма допомагає представити систему на більш високому рівні.

За допомогою діаграми варіантів використання визначаємо загальні межі і контекст модельованої предметної області на початкових етапах проектування системи, формуємо загальні вимоги до функціональної поведінки проектованої системи та розроблюємо вихідну концептуальну модель системи для її подальшої деталізації у формі логічних та фізичних моделей.

Використання програмного комплексу полягає у створенні цифрового меню та генерації QR-коду та формуванні посилання на нього. Сутність, яка взаємодіє з системою і використовує її функціональні можливості для досягнення певних цілей є актором. Можна виділити два актори, котрі можуть ініціювати події в даній предметній області:

1. Користувач – особа, яка використовує можливості сервісу задля отримання цифрового QR-меню.

Кафедра КІТ (47)				НАУ 21 09 94 000 ПЗ			
Виконала	Молодоженя Д.С.			АНАЛІЗ МЕТОДІВ ТА ТЕХНОЛОГІЙ РОЗРОБКИ ПРОГРАМНОГО КОМПЛЕКСУ	Літ.	Арк.	Аркушів
Керівник	Райчев І.Е.					35	26
Консульт.					УС-201Мз 122		
Н. контр.	Райчев І.Е.						

2. Інформаційна система – система, яка на основі представлених користувачем даних формує web-сторінку меню, посилання на неї та генерує QR-код.

Сценарій – це повний ланцюжок подій, ініційованих актором, і специфікація реакції на них. Сукупність сценаріїв визначає всі можливі шляхи використання системи, опишемо їх:

- 1) Створення меню – процес заповнення та завантаження необхідних даних для формування цифрового меню.
- 2) Внесення даних щодо закладу – заповнення представлених системою форм необхідної інформацією.
- 3) Заповнення форми “Базова інформація” – заповнення назви закладу, короткий опис закладу, телефон(и) та завантаження зображення закладу.
- 4) Заповнення адреси – внесення в поле форми адреси закладу або декілька адрес (у разі, якщо користувач представляє мережу закладів).
- 5) Заповнення посилань на соціальні мережі – внесення у відповідні поля форми посилань на сторінки соціальних мереж у разі їх існування.
- 6) Заповнення меню – заповнення усієї необхідної інформації для представлення переліку страв, що пропонуються закладом.
- 7) Вибір категорії меню – можливість вибору користувачем необхідної категорії меню із представленого переліку, у разі невідповідності, створення власної.
- 8) Створення позиції меню – внесення страви відповідно обраній категорії, заповнюючи форму, у якій користувач вводить назву страви, її масу, опис, калорійність та ціну.
- 9) Вибір оформлення меню – вибір одного варіанту з декількох представлених щодо оформлення стилю меню та розташування елементів чи секцій.
- 10) Вибір кольорової гами дизайну меню – вибір представлення назв категорій, а саме вибір кольору тексту та вибір фоновому кольору.

- 11) Вибір шрифту – вибір одного шрифту з представлених варіантів, який використовуватиметься на сформованій web-сторінці.
- 12) Вибір представлення секцій – користувач має змогу з певної кількості представлених шаблонів розміщення інформації про заклад та розміщення страв обрати актуальний для себе варіант.
- 13) Обробка та збереження даних – система оброблює та зберігає вказані користувачем дані.
- 14) Формування web-сторінки цифрового меню закладу – на основі вказаних користувачем даних, система формує web-сторінку з відповідним розташуванням елементів та секцій, враховуючи обрані користувачем стилі.
- 15) Формування посилання на меню – система формує посилання на створену web-сторінку.
- 16) Генерація QR-код – система генерує QR-код створений web-сторінки.

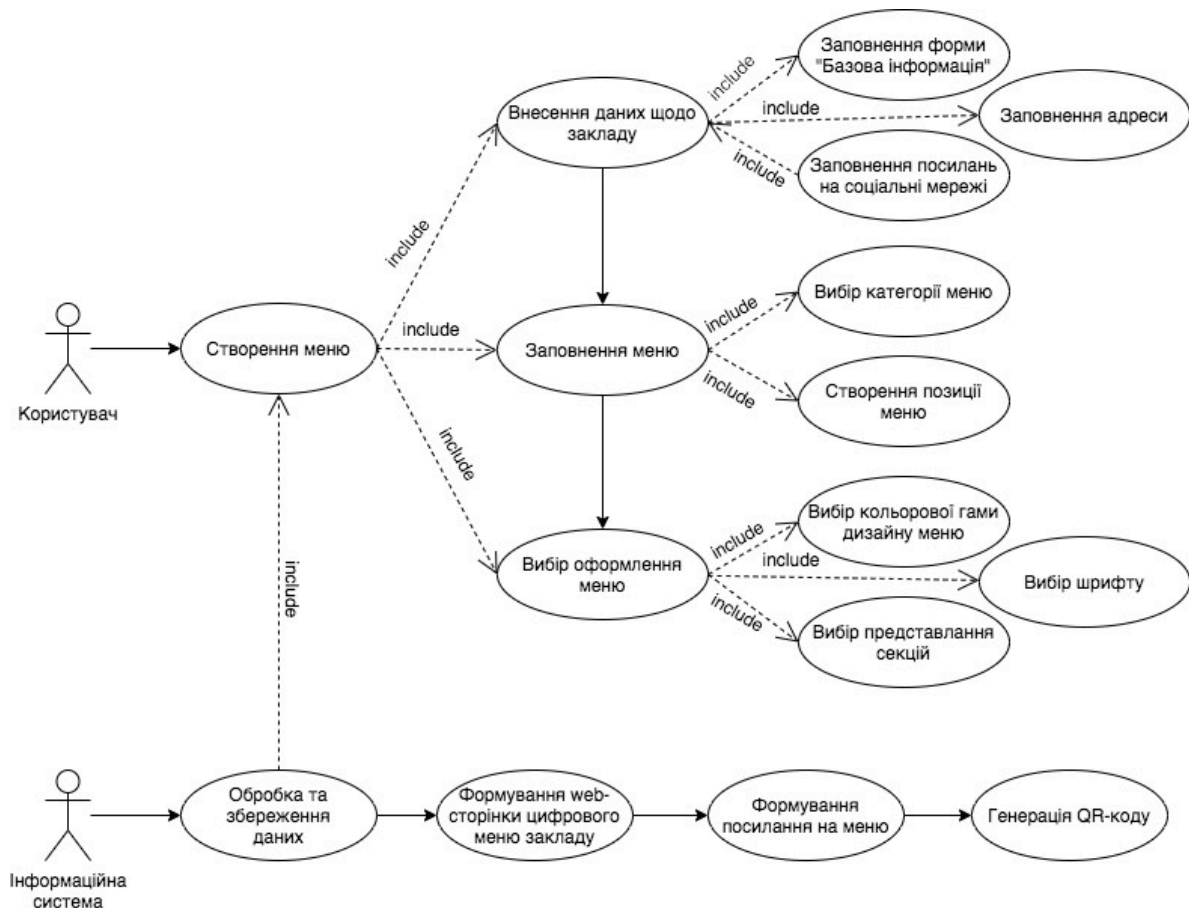


Рис. 2.1. Діаграма варіантів використання

2.2. Формування потреб для розробки

Формуючи потреби для розробки продукту потрібно чітко розуміння про користувача розробляемого продукту та цілі, які він переслідує. Оскільки враховуючи дану предметну область, в реальності неможливо орієнтуватися на всіх та кожного споживача, тому для управління продуктом потрібно сформувати розуміння цільової аудиторії. Тобто, розроблюючи даний програмний комплекс, доцільно орієнтуватись на конкретних споживачів. В реалізації такого підходу можлива, якщо дотримуватись принципів User-Centered Design.

User-centered design або UCD – це створення продукту, у якому перше місце займає користувач із його особливостями, завданнями, цілями і контекстом використання продукту. Такий підхід широко застосовується для вирішення різноманітних завдань – від створення сайтів та додатків до проектування вуличної навігації.

Використовуючи підхід UCD, потрібно зрозуміти та скласти збірний образ типових представників цільової аудиторії продукту, які об'єднані певними ознаками та описати задачі, які повинна виконати користувач в рамках продукту для того, щоб досягти своєї мети. Тоді орієнтуючись на правильно зіставлений «портрет» користувача, можна визначити, яким повинна бути взаємодія та зовнішній вигляд продукту та який функціонал та навігацію очікує побачити користувач.

Для кращого формування персони, візьмемо образ певної довіреною керівництвом закладом харчування особи, котра повинна створити цифрове меню. Зазвичай перед нею стоїть задача отримати максимально якісний продукт за короткий проміжок часу, не виходячи за рамки встановленого бюджету. Виходячи з того, що головна ціль користувача продукту є отримання цифрового QR-меню, потрібно сформувати задачі, котрі користувач зможе вирішувати за допомогою розробляемого продукту, а також враховувати загальні поведінкові фактори – фактори комфорту та дискомфорту. Отже, доцільно вважати, що сервіс повинен задовільнити такі потреби персони:

- структуровано ввести необхідні дані та інформацію щодо закладу та його меню;
- на основі введених даних сформувати стилістично оформлену web-сторінку;
- змінити або відкоригувати стилістику сторінки, враховуючи власні потреби або концепцію;
- отримати посилання на готову web-сторінку із меню закладу;
- отримати QR-код.

Розглянемо можливі фактори комфорту та дискомфорту для користувача. Враховуючи те, що користувачу все ж таки доведеться ввести чималий об'єм інформації, що стосується меню, опису страв, то потрібно створити максимально структурований та інтуїтивно зрозумілий інтерфейс. Важливо створити зручні форми для заповнення необхідної для користувача інформації та відобразити їх з логічною послідовністю. Адже чітко сформований алгоритм дій користувача при роботі з програмним комплексом зменшить витрати часу та оптимізує сам процес створення меню.

2.2.1. Оцінка якості продукту

Реалізація цілей і завдань покладається на високоякісне програмне забезпечення та системи. Високоякісні програмні продукти та комп'ютерні системи, що містять багато програмного забезпечення, необхідні для забезпечення цінності та уникнення потенційних негативних наслідків.

Оцінка якості програмного забезпечення є ключовим фактором в проектуванні та розробці. Цього можна досягти шляхом визначення необхідних і бажаних характеристик якості, пов'язаних із цілями та завданнями зацікавлених сторін для системи. Сюди входять якісні характеристики, пов'язані з програмною системою та даними, а також вплив системи на її зацікавлені сторони. Моделі якості в міжнародному стандарті можуть бути використані для визначення відповідних

характеристик якості, які можуть бути використані для встановлення вимог, їх критеріїв задоволення та відповідних заходів.

Оцінимо якість системи за допомогою моделі якості продукту за стандартом ISO/IEC 25010.

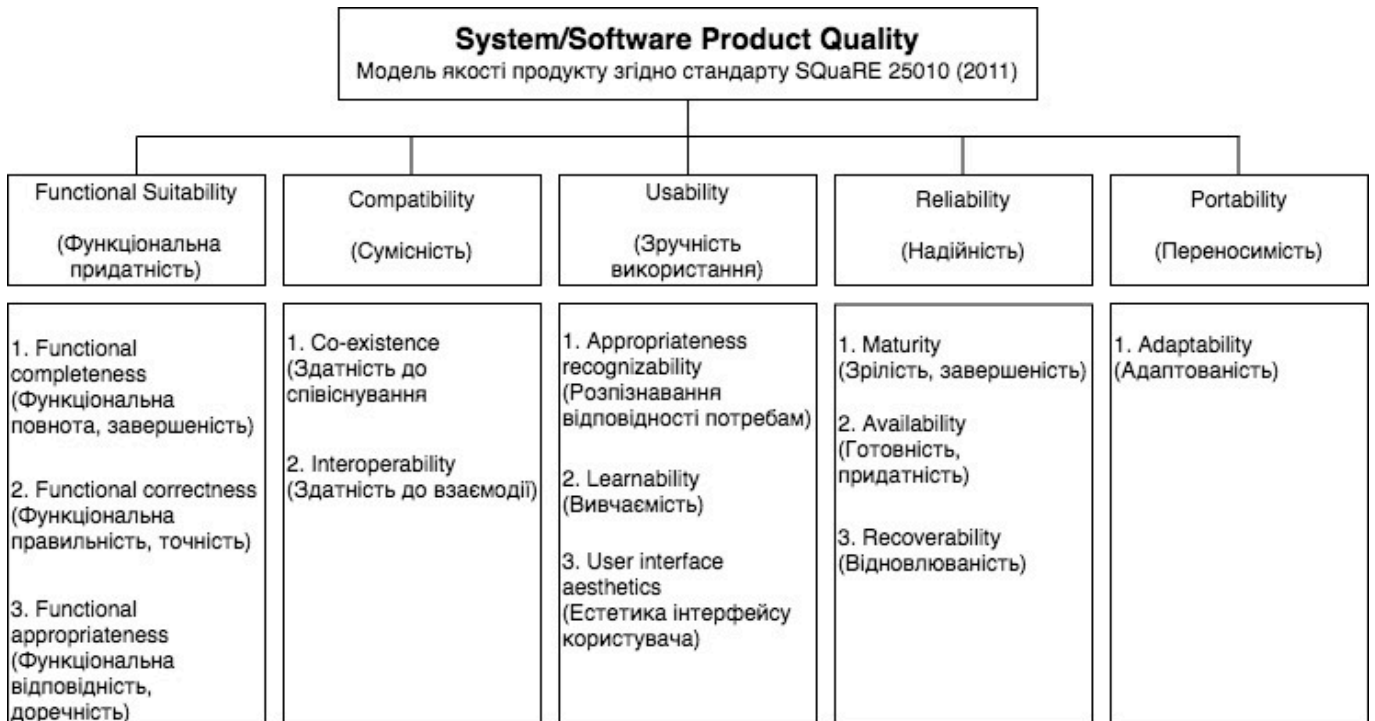


Рис. 2.2. Модель якості продукту

Функціональність включена до переліку характеристик через важливість сфери в котрій заплановано використання системи. Тому необхідно, щоб функції системи були реалізовані в повному обсязі, а її точність коректно відображувала інформацію.

Сумісність є важливою характеристикою у тому плані, що продукт, система чи компонент можуть обмінюватися інформацією з іншими продуктами, системами чи компонентами та/або виконувати свої необхідні функції, використовуючи те саме обладнання або програмне середовище.

Зручність використання важлива характеристика для зручного та інтуїтивно зрозуміло користування системою, для досягнення продуктивності та задоволеності під час використання.

Надійність включена до моделі з метою оцінки стійкості її до відмов та можливості забезпечення безперервної та стабільної роботи з інформацією. Дані вимоги ключові в процесі створення цифрового меню.

Переносимість системи необхідно оцінювати з метою встановлення здатності її до роботи в різноманітному середовищі, що включає велику кількість стороннього програмного забезпечення. Програмний комплекс призначений для роботи на персональних комп'ютерах користувачів, а відповідно повинен працювати в різноманітних зовнішніх умовах і адаптуватись до програмного середовища.

Тобто, відповідно до діаграми варіантів використання (Рис. 2.1) такі характеристики, як надійність, переносимість, сумісність, функціональна придатність є важливими в загальному для усієї системи. Що стосується зручності використання, то для створення меню є досить важлива вивчаємість та розпізнавання потребам. Естетика інтерфейсу користувача є важливою, як і при створенні меню, так і при формуванні web-сторінки цифрового меню закладу.

2.2.2. Функціональні та нефункціональні вимоги

Вимоги до програмних продуктів або інформаційних систем можна розділити на дві великі групи: функціональні вимоги та нефункціональні вимоги.

Функціональні вимоги визначають функцію, яку має виконувати система чи системний компонент.

Функціональне призначення програмного забезпечення визначає архітектуру програмно-апаратного комплексу, для його проектування та розробки програмного необхідно визначити:

- перелік розв'язуваних завдань;
- набір функцій, які мають бути реалізовані для вирішення кожного із завдань;
- архітектуру апаратного комплексу;
- архітектуру програмного комплексу;
- архітектуру та дизайн інтерфейсу програмного продукту;

– структуру та взаємозв'язок даних.

Нефункціональні вимоги фіксують умови, за яких рішення має залишатися ефективним, або якості, які рішення має мати. Ці вимоги визначають критерії, які можуть використовуватись для оцінки роботи системи, а не для конкретної поведінки.

Нефункціональне призначення програмного забезпечення визначає:

- вимоги до займаних ресурсів;
- обмеження на час вирішення завдання;
- швидкість доступу до даних;
- обсяг оброблюваних даних;
- безпека.

Нефункціональні вимоги можна розділити на такі якості виконання, як безпека та зручність використання і еволюційні якості, такі як тестованість, ремонтпридатність, розширюваність та масштабованість, які втілені у статичній структурі програмної системи. До нефункціональних вимог насамперед відносять атрибути якості продукту. Тобто, вимоги, що визначають якісні характеристики розробки. Сформуємо перелік вимог за допомогою моделі якості (Рис.2.1).

Функціональні вимоги.

Сумісність. Вимога описує, чи програмна система сумісна з різними системами.

Функціональна придатність. Точність визначає, що дані, що вводяться в систему, правильно обчислюються та використовуються системою, і що вихідні дані є правильними. Тобто дані, котрі вводить користувач повинні бути відображені точно та коректно та у тому вигляді, який обере користувач. Відповідність: Функціональні вимоги щодо відповідності підтверджують відповідність розробленої системи промисловим стандартам. Програмний продукт повинен відповідати інструкціям з кібербезпеки та Всесвітній павутині з точки зору доступності.

Нефункціональні вимоги.

Usability вимірює зручність програмної системи, що розробляється. Адже розроблений програмний продукт, на основі даних формує web-сторінку та

генерує QR-код. Так як це є зручно для користувача, то зручність використання програми буде високою.

Надійність – ще один аспект доступності. Цей атрибут якості підкреслює доступність системи за певних умов. Коли користувач переходить між кроками заповнення інформації, то система повинна збирати та зберігати останню актуальну введену інформацію.

Переносимість означає здатність програмної системи працювати в іншому середовищі, якщо основний залежний фреймворк залишається незмінним. Програмний продукт повинен бути кросбраузерним та коректно відобразитись в незалежності від типу пристрою.

2.3. Діаграма класів

Центральне місце в ООП займає розробка логічної моделі системи у вигляді діаграми класів. Діаграма класів служить для представлення статичної структури моделі системи. Клас в мові UML служить для позначення безлічі об'єктів, які мають однакову структуру, поведінку та відносини з об'єктами з інших класів. Сама форма класу складається з прямокутника з трьома рядками. Верхній рядок містить назву класу, середній рядок містить атрибути класу, а нижній розділ містить методи чи операції, які може використовувати клас.

Діаграма класів надає ряд переваг при розробці програмного продукту, адже ілюструє модель даних для інформаційної системи, надає краще розуміння загальної схеми програми, візуально показує конкретні потреби системи.

Згідно Мартіну Фаулеру існують три різні точки зору на побудову діаграми класів:

- 1) концептуальна точка зору;
- 2) точка зору специфікації;
- 3) точка зору реалізації.

У даному випадку краще будувати діаграму класів з точки зору реалізації, тобто класи діаграми відповідатимуть реальним класам програмної системи. У такому разі дана діаграма полегшить та значно пришвидшить програмну реалізацію продукту, адже з її допомогою матимемо наглядно ілюстровану логіку написання коду.

Отже, проаналізувавши предметну область виділимо 6 класів (Рис.2.3). MenuData – клас, основним методом якого є додавання нової категорії меню (addNewCategory).

CategoriesData – клас, атрибути якого містять тип категорію та перелік страв, що відповідають конкретній категорії. Даний клас має відношення композиції із класом MenuData. Це відношення служить для виділення спеціальної форми відносин "частина-ціле", при яких складові частини в деякому сенсі знаходяться всередині цілого. Специфіка взаємозв'язків між ними полягає в тому, що CategoriesData не може виступати в відриві від MenuData, тобто зі знищенням цілого знищуються і всі його складові частини.

Клас ProjectData міститиме у собі усю інформацію надану користувачем таку, як дані з форм та інформацію щодо категорій та страв. Тобто даний клас має відношення агрегації із класом CategoriesData, це означає, що клас ProjectData представляє собою сутність, що включає в себе в якості складових частин іншу сутність (CategoriesData).

Клас DishesData основними методами якого є отримання індексу категорії (getIndexCategory) та додаванням у неї страви (addDish). Враховуючи те, що об'єкт класу DishesData використовує об'єкт класу CategoriesData у своєму методі addDish, то такий вид відносин моделюється як залежність.

Клас CategoryData має відношення залежності з класом CategoriesData, оскільки може змінювати його за допомогою методу редагування назви категорій (editCategoryName). Також CategoryData має відношення залежності з класами DishesData та MenuData, тобто коли деяка зміна одного елемента моделі може вимагати зміни іншого залежного від нього елемента моделі.

CategoryChoose має відношення асоціації з класом CategoryData, що вказує на взаємодію між даними класами. Також клас CategoryChoose має відношення залежності із класом MenuData, тобто в результаті збереження обраної категорії (saveChooedCategory) вимагається зміна іншого класу і виклик його методу додавання нової категорії (addNewCategory).

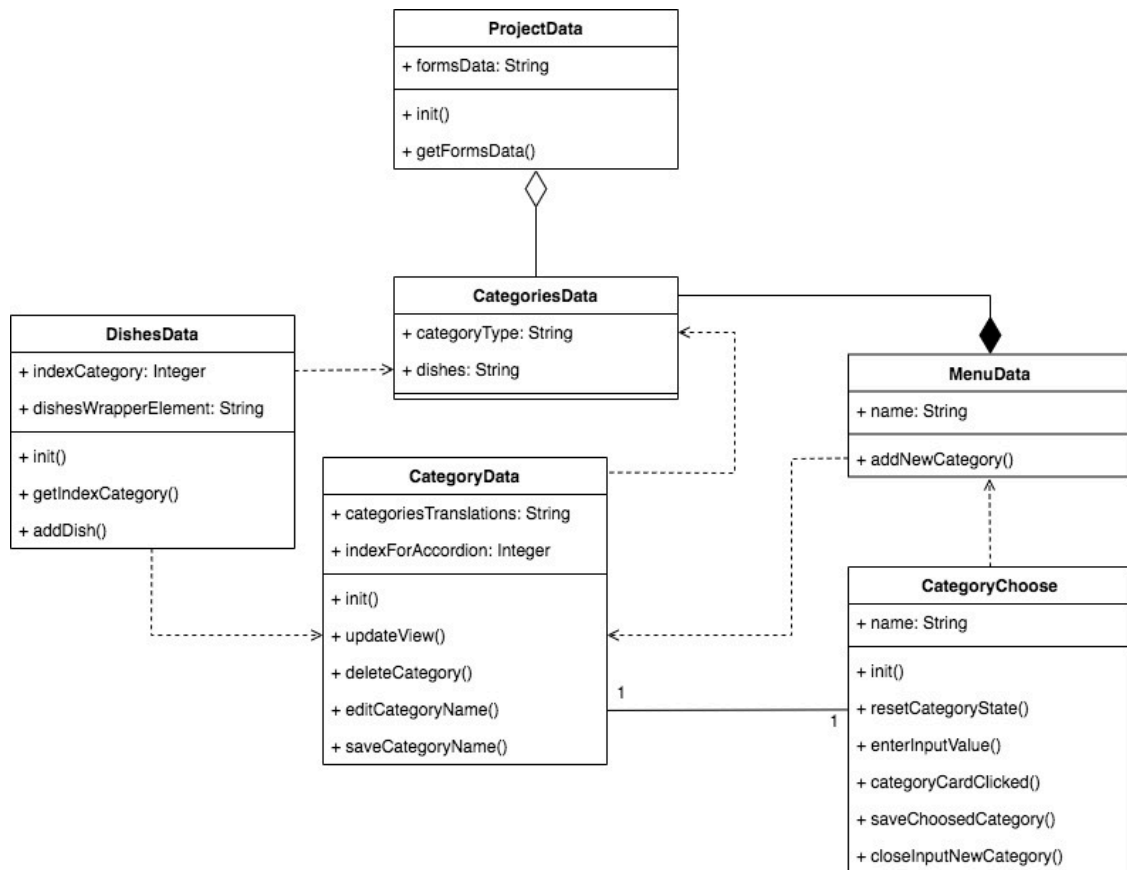


Рис. 2.3 – Діаграма класів

2.4. Клієнт-серверна архітектура

В загалом значна більшість вебресурсів мають складну архітектуру. Тобто вони не дають статичну відповідь на запити клієнта, а формують її динамічно, отримуючи дані з БД та формуючи їх за певними правилами.

В цілому можна сказати, що робота вебресурсів полягає в обміні запитами та відповідями між клієнтом та сервером. Зазвичай, коли йдеться про вебдодатки, то у ролі клієнта виступає браузер.

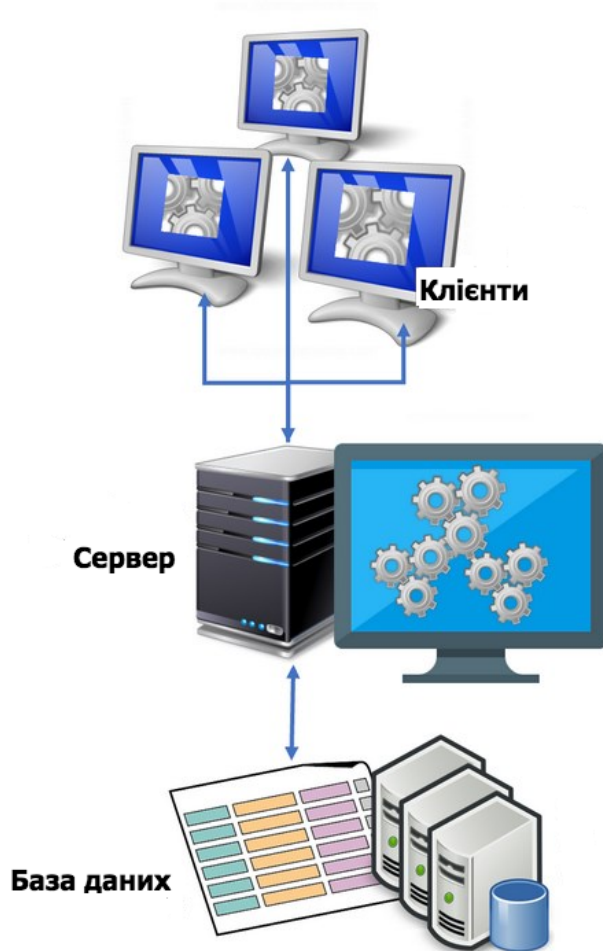


Рис. 2.4. Клієнт-серверна взаємодія

Браузер призначений для перегляду вебсторінок. З його допомогою інтернет користувач має можливість шукати інформацію, завантажувати файли, переглядати відео та фото, слухати музику в глобальній мережі Інтернет.

Для отримання доступу до інформації, що міститься на сайті, користувач вказує в адресному рядку браузера URL вебсторінки. Браузер надсилає запит на відповідну адресу. Якщо вказаний URL містить HTML-документ, сервер надсилає вебсторінку на IP-адресу комп'ютера.

Клієнт-серверна архітектура має свої переваги та недоліки. До переваг даної архітектури можна віднести:

- централізованість, оскільки всі дані та управління зосереджені на центральному сервері;

- інформаційна безпека, оскільки ресурси загального користування адмініструються централізовано;
- продуктивність використання виділеного сервера підвищує швидкість роботи ресурсів загального користування;
- масштабованість, кількість клієнтів та серверів можна збільшувати незалежно один від одного.

До недоліків слід віднести:

- перевантаження трафіку в мережі, що є головною проблемою в мережах клієнт-сервер. Коли велика кількість клієнтів одночасно запитують одну послугу на сервері, кількість запитів може створити перевантаження в мережі;
- наявність єдиної точки відмови у невеликих мережах із одним сервером. Якщо він відмовляє, то всі клієнти залишаються без обслуговування;
- перевищення меж ресурсів сервера, коли нові клієнти, які вимагають послугу, залишаються без обслуговування. У таких випадках потрібно термінове розширення ресурсів сервера;
- деколи клієнтські програми можуть не працювати на терміналах користувачів, якщо не встановлені відповідні драйвери. Наприклад, користувач надсилає запит на друк документа, а на сервері немає відповідного драйвера для друку цього формату документа на певному принтері.

2.5. Технології розробки серверної частини

Програмування серверної частини використовуються в більшості вебсайтів для того, щоб при необхідності динамічно відображати дані. Зазвичай дані беруться із бази даних, яка знаходиться на сервері і відображаються за допомогою HTML та JavaScript коду.

В сучасному світі веброзробки серверна частина розробки надає можливості кращої взаємодії з користувачем, зберігаючи його дані, формуючи контент конкретного під користувача.

Існують динамічні та статичні web-сайти. Їх різниця полягає у тому, що статичні повертає при запиті конкретного ресурсу один й той самий вміст з сервера кожного разу, а динамічні ті, де при необхідності частина вмісту відповіді генерується динамічно.

Тобто, динамічний сайт на основі інформації представленої користувачем може повертати різні дані. Тому для підтримки така вебсайту більша частина коду виконуватись на сервері.

Для кращого розуміння, детально розглянемо архітектуру динамічного сайту (Рис.). Браузери надсилають HTTP-запити на сервер, потім сервер обробляє запити та повертає відповідні HTTP-відповіді.

Запити статичних ресурсів (будь-які файли, які не змінюються, зазвичай це: CSS, JavaScript, зображення, попередньо створені PDF-файли та інше) обробляються так, що сервер отримує запитаний документ зі своєї файлової системи і повертає HTTP-відповідь. У тому випадку, якщо документ міститься, то повертається успішний статус, у разі не отримання документу повертається статус помилки.

Запити динамічних даних надсилаються у код серверної частини вебдодатку. Для цього сервер інтерпретує запит, читає необхідну інформацію з бази даних, комбінує вилучені дані з шаблонами HTML та повертає відповідь, що містить згенерований HTML.

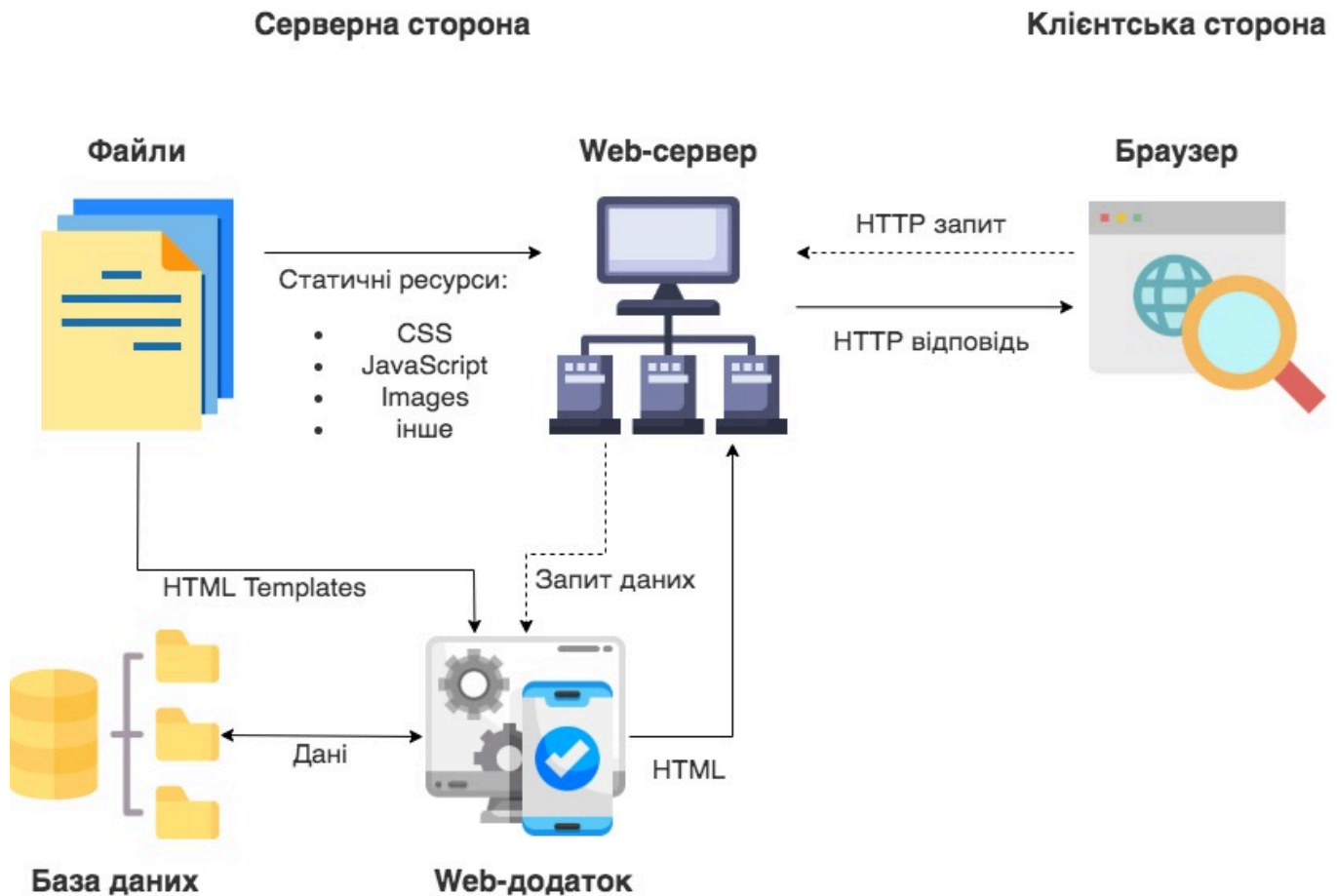


Рис. 2.5. Архітектура динамічного вебсайту

Вибір мов програмування для серверної частини є доволі різноманітним, серед яких PHP, Python, Java, Ruby on Rails, C#, Node.js. До того ж, кожна із технологій має велику кількість модулів та фреймворків, побудованих на її основі.

Вебфреймворки – це набори функцій, об'єктів, правил та інших конструкцій коду, призначені для прискорення розробки, вирішення певних проблем, та спрощення різних типів завдань.

Враховуючи завдання розроблюваного програмного комплексу, то серверна технологія Node.js має низку переваг, серед яких надійність, швидкість та простота використання. Також до переваг даної технології можна віднести вбудований менеджер додаткових модулів `node package manager (npm)`, котрий знайшов своє використання не тільки як менеджер пакетів не тільки Node.js застосунків, але й для клієнтських додатків.

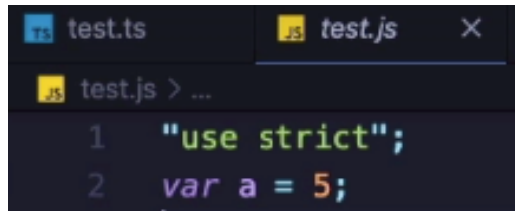
Існує чимала кількість фреймворків технології Node.js, які допомагають у розробці, тим самим, що пришвидшують її, надаючи базову структуру, та полегшують підтримку та тестування. Серед таких можна виділити: Nest.js, Express.js, Socket.io, Нарі.js та інші. При чому даний перелік постійно оновлюється.

Одним із найважливіших та найпопулярніших серед фреймворків Node.js є Nest.js. Nest.js – це прогресивний фреймворк з модульною архітектурою для створення ефективних, надійних і масштабованих додатків на стороні сервера. Він використовує сучасний JavaScript, створений за допомогою TypeScript (зберігає сумісність з чистим JavaScript) і поєднує елементи ООП, функціональне програмування і реактивне програмування. Nest на своїй основі використовує Express, але також забезпечує сумісність з широким спектром інших бібліотек. До прикладу, одна з таких – Fastify, що дозволяє легко використання інших доступних плагінів. Тобто будь-які розширення express легко впровадити у Nest.

Nest.js написаний на TypeScript. TypeScript – це типізована мова програмування, написана на основі JavaScript. Суворі типізація TypeScript зменшує кількість потенційних помилок. Дана мова програмування реалізує більшість концепцій, які властиві об'єктно-орієнтованим мовам (успадкування, інкапсуляція, поліморфізм та модифікатори доступу тощо). TypeScript є зворотно сумісним з JavaScript. Таким чином, будь-який код JavaScript є валідним для TypeScript. Компілятор TypeScript називається tsc, котрий написаний мовою TypeScript, та може бути скомпільований у стандартний JavaScript, а після цього запущений на будь-якому движку JavaScript (наприклад, у браузері). На рисунках 2.6 та 2.7 показано приклад компіляції TypeScript коду у JavaScript.

A screenshot of a code editor window. The window title bar shows two tabs: 'tsconfig.json' and 'test.ts'. The 'test.ts' tab is active. The editor content shows a single line of TypeScript code: '1 const a: number = 5;'. The text is color-coded: '1' is blue, 'const' is purple, 'a:' is light blue, 'number' is green, and '= 5;' is white. The cursor is positioned at the end of the line.

Рис. 2.6. Фрагмент TypeScript коду



```
test.ts test.js X
test.js > ...
1 use strict;
2 var a = 5;
```

Рис. 2.7. Фрагмент JavaScript коду

Головна задача TypeScript – це використання типів, у нетипізованій мові програмування JavaScript, що впливає на якість написання коду, який оснований на поведінці типів. Написання TypeScript коду можливо в більшості текстових редакторів та середовищах розробки, які мають підтримку даної мови програмування на рівні плагінів, серед них:

- Sublime,
- Visual Studio,
- WebStorm,
- Atom
- та інші.

Підсумовуючи, Node.js - це середовище виконання, яке базується на движку Chrome V8 JavaScript та має ряд переваг:

- велика кількість безкоштовних інструментів;
- можливість спільного та повторного використання;
- швидка обробка запитів;
- оптимальна швидкість роботи додатків;
- повна підтримка JSON;
- кросплатформність.

Для Nest.js створена велика кількість сторонніх бібліотек, які допоможуть у вирішенні практично будь-якого завдання. До того ж даний фреймворк:

- активно розвивається та підтримується;
- простий у використанні;
- має відкритий вихідний код;
- має зручну CLI для швидкого створення компонентів;

- добре документований;
- підтримує розробку будь-яких видів web-додатків.

Важливе місце в розробці програмного продукту займає місце вибору системи керування базою даних (СКБД). Не дивлячись на те, що завдання усіх СКБД полягає в створенні, редагуванні та отриманні доступу до інформації користувачем, яка зберігається в базі даних, то сам процес виконання задачі доволі різниться. Можливості та функції СКБД можуть мати суттєві відмінності.

База даних – певний набір даних, що зберігаються, дії над якими можливі відповідно до правил засобів моделювання даних.

Для того, щоб зробити правильний, потрібно розуміння, кому надати перевагу реляційним чи нереляційним БД. Отже, доцільно розглянути їх переваги та відмінності. РБД допомагає керувати великими об'ємами структурованих даних, які можуть представляти об'єкти з навколишнього світу. Ці дані обов'язково групуються у таблицях у заданому форматі. У стовпцях вказуються типи даних, а у рядках – записи. Що ж стосується нереляційних баз даних, то у них не використовується таблична схема рядків і стовпців. Їм притамана модель зберігання, оптимізована під конкретні вимоги типу даних, що зберігаються. Використання нереляційних баз даних широко застосовується у різноманітних додатках. Їх використання є доцільним, коли важлива не структурованість зрозумілих даних, а гнучка і легко масштабована БД, що характеризується до того ж високими параметрами продуктивності. Але у випадку, коли потрібна надійна обробка транзакцій, або ж обробка великої кількості складних запитів або рутинного аналізу даних, то варто надати перевагу реляційним базам даних.

Враховуючи відмінності між БД та потреби розроблюємого програмного комплексу, варто зупинити свій вибір на нереляційних базах даних.

На тепершній час популярними є такі СКБД:

- PostgreSQL;
- Microsoft SQL Server;
- MySQL;

- MongoDB;
- Oracle.

PostgreSQL є об'єктно-реляційною системою керування базами даних з відкритим сирцевим кодом.

Microsoft SQL Server розроблена корпорацією Microsoft система керування реляційними базами даних (РСКБД).

MySQL – реляційна система керування базами даних з відкритим вихідним кодом.

Oracle - об'єктно-реляційна система управління базами даних компанії Oracle.

Розглянемо детальніше нереляційну базу даних MongoDB, оскільки вона має ряд переваг:

- відсутність схеми;
- базується на колекціях різних документів;
- кількість полів, зміст та розмір цих документів може відрізнятися (різні сутності не повинні бути ідентичними за структурою);
- зрозуміла структура кожного об'єкта;
- легко масштабування;
- для зберігання даних, що використовуються в даний момент, використовується внутрішня пам'ять, що дозволяє отримувати швидший доступ;
- дані зберігаються у вигляді JSON документів;
- підтримка динамічних запитів документів (document-based query);
- відсутність складних JOIN запитів.

Тобто, можна сказати, що MongoDB реалізує новий підхід до побудови бази даних, де відсутні таблиці, схеми, SQL запити, зовнішні ключі та багато інших речей, які містять об'єктно-реляційні бази даних.

MongoDB — програма керування базами даних NoSQL з відкритим вихідним кодом. Бази даних NoSQL досить корисні для роботи з великими наборами розподілених даних. MongoDB — це інструмент, який може керувати документально-

орієнтованою інформацією, зберігати або отримувати інформацію. MongoDB використовується для збереження неструктурованих даних у форматі JSON. MongoDB не підтримує розширену аналітику та об'єднання, як підтримка баз даних SQL.

2.6. Технології розробки клієнтської частини

За допомогою клієнтської частини відбувається взаємодія користувача із додатком, що виконується у браузері або в іншому середовищі на стороні клієнта. Для побудови повноцінного клієнтського інтерфейсу використовуються такі базові технології:

- HTML;
- CSS;
- JavaScript.

HTML – це мова розмітки гіпертексту, що дозволяє створювати та структурувати розділи, параграфи, посилання, заголовки і блоки для вебсторінок. HTML має просту структуру коду, яка включає теги та атрибути. HTML не являється мовою програмування, тобто він не може створювати динамічні функції. Дана мова пройшла довгий шлях, починаючи з перших днів до останнього оновлення – версії HTML5. HTML5 розширила функціональні можливості мови, найбільш очікуваною особливістю стала підтримка вбудовування аудіо та відео без використання Flash-плеєра. Проте HTML має ряд переваг та недоліків. Серед переваг даної мови можна виділити:

- популярність мови з великою кількістю ресурсів;
- виконується у кожному браузері;
- послідовна розмітка;
- легко інтегрується з базовими мовами, такими як PHP та Node.js;
- офіційні вебстандарти підтримуються консорціумом World Wide Web (WWW).

Що стосується недоліків, то:

- не всі версії браузерів підтримують нові функції;
- не можливість реалізації логіки;
- для створення динамічної функціональності потребується використання JavaScript, або іншої серверної мови.

CSS служить для стилізації та оформлення зовнішнього вигляду документа. Тобто, це мова таблиці стилів.

CSS має свою структуру – селектор, властивість та значення властивості. Селектор – це ім'я HTML-елемента, для якого будуть застосовані стилі. Далі вказується властивість, яку бажаємо надати елементові, наприклад, колір, розмір шрифту і т.п. в залежності від обраного елемента. Далі після двокрапки записується значення властивості, яке вибирає одну з безлічі можливих ознак для даних властивостей. Сучасний підхід до розробки клієнтської частини неможливий без використання CSS. Дана технологія:

- забезпечує відносно просту та швидку розробку;
- підвищує гнучкість та зручність редагування – достатньо виконати виправлення в CSS-файлі, щоб зміни застосувались до усього;
- оптимізує код, зменшуючи кількість повторень;
- прискорює час завантаження, тому що CSS може кешуватися при першому відкритті, а в наступних зчитуються лише структура та дані;
- розширює можливості візуальних рішень для представлення контенту;
- забезпечує можливість легко застосовувати до одного документа різні стилі (адаптивна версія для мобільних пристроїв або спеціальні стилі для людей з вадами зору).

JavaScript (офіційно ECMAScript) – мультипарадигмальна мова програмування, яка забезпечує динамічну інтерактивність вебресурсів.

JavaScript має такі архітектурні риси:

- динамічна типізація;
- автоматичне керування пам'яттю;
- прототипне програмування;

- функції як об'єкти першого класу.

Окрім основних технологій, також широко використовуються й інші, котрі пришвидшують процес розробки.

Sass – використовується для покращення та спрощення структури CSS-коду. Sass має два синтаксиса:

Sass – у даному синтаксисі відсутні фігурні дужки, тобто вкладені елементи реалізовані за допомогою відступів.

SCSS– використовує фігурні дужки, як і сам CSS.



Рис. 2.8. Відмінності синтаксу sass, SCSS, CSS

Так як JavaScript користаються вликою популярність та швидко розвивається, він має велику кількість бібліотек та фреймворків.

Бібліотека – це багаторазова частина коду, котра пропонує певні функції.

Популярними бібліотеками JavaScript є:

- React;
- JQuery;
- D3.js.

Окрім цього в розробці активно використовуються фреймворки, такі як:

- Angular;
- Vue.js;
- Ember.js.

2.7. Валідація даних

Перш за все валідація – це перевірка даних на відповідність встановленим правилам та обмеженням.

Перед тим як надіслати дані на сервер важливо переконатись, що всі обов'язкові поля форми заповнені даними у коректному форматі. Це називається валідацією на стороні клієнта і допомагає переконатися, що дані, введені в кожен елемент форми, відповідають встановленим вимогам. Даний вид валідації покращує взаємодію користувача з інтерфейсом, адже виявивши некоректність даних, їх можна відразу виправити. У тому випадку якщо перевірка відбувається лише на сервері, процес заповнення може бути більш трудомістким, оскільки вимагає повторення тих самих дій відправлення даних на сервер для отримання зворотної відповіді з повідомленням про те, що потрібно виправити.

Враховуючи специфіку розроблюваного програмного комплексу, користувач має надати певний об'єм даних, а так як саме за допомогою форм зручно забезпечити їх структурований збір, то питання валідації даних є доволі актуальним.

Валідація даних можлива під час заповнення (Рис.2.9), що дозволяє оперативно повідомляти користувача про некоректність значення у певному полі.

Order Form

First Name: Linda

Last Name: Johnson

Email Address: linda.johnson2gmail.co,
Please enter a valid email address.

Credit Card Number: [Empty]

SUBMIT

Рис. 2.9. Валідація даних під час заповнення поля

Або ж під час збереження, тобто перевірка починається під спроби відправки форми (Рис.2.10). Це дозволяє перевіряти складні взаємозв'язки між полями об'єкта, оскільки перевірка відбувається одночасно.

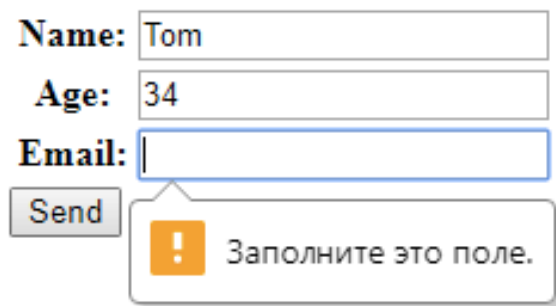
The image shows a web form with three input fields: 'Name' containing 'Tom', 'Age' containing '34', and 'Email' which is empty. A 'Send' button is located below the 'Email' field. A validation error message is displayed in a box below the 'Email' field, featuring an orange exclamation mark icon and the text 'Заполните это поле.' (Fill in this field.)

Рис. 2.10. Валідація при відправці даних

На етапах заповнення та відправки форми важливо перевіряти дані, щоб визначити, чи відповідають вони потрібному формату. Якщо формат коректний, програма дозволить відправити дані на сервер та зберегти базу даних; в іншому випадку потрібно вивести повідомлення з описом помилки, дозволяючи користувачеві знову ввести дані вже у коректному вигляді.

В процесі розробки необхідно приділити увагу валідація даних для того, що вебсервіс працював належним чином. Для коректного формування цифрового меню, користувач має надати певний обов'язковий об'єм даних. Тому потрібно відслідковувати, якщо дані від користувачів заповнені у неправильному форматі, некоректні самі по собі або зовсім пропущені, відсутність або некоректність яких вплине на кінцевий результат від чого напряму залежить задоволеність роботи клієнта з сервісом.

На стороні клієнта можливі два типи валідації форм:

- 1) вбудована валідація форм, котра використовує функціонал валідації HTML5;
- 2) JavaScript валідація, що кодується за допомогою мови програмування JavaScript.

HTML5-валідація зазвичай демонструє кращу продуктивність, проте JavaScript валідація краще налаштовується, однак вимагає програмування всієї логіки або використання бібліотеки.

Перевірка на валідність елементів форми HTML5 виконується за допомогою атрибутів валідації, до прикладу такі як:

- `required`: визначає, що поле обов'язково повинно бути заповнене;
- `minlength` та `maxlength`: задають мінімальну та максимальну довжину текстових даних (рядків);
- `min` та `max`: задають мінімальне та максимальне значення у тому разі, якщо поле розраховане на числовий тип даних;
- `type`: визначає тип даних, на який розраховано поле: число, email-адресу, пароль чи інше;
- `pattern`: за допомогою регулярного виразу, визначає шаблон, якому повинні відповідати дані, що вводяться.

Дані введені в поле форми, вважаються валідними у тому випадку, якщо відповідають правилам перерахованих вище атрибутів, в іншому випадку – не валідними.

Висновки до розділу 2

Детальний аналіз архітектури та методів розробки серверної та клієнтської частини дає можливість сформулювати основні принципи побудови програмного комплексу.

За допомогою серверної технології Node.js та фреймворку Nest.js можливо створити оригінальну архітектуру додатку, яка дозволить створювати добре тестовані, масштабовані та легко обслуговувані програми.

Керування, зберігання та отримання інформації здійснюватиметься завдяки MongoDB, що дозволить збереження неструктурованих даних у форматі JSON.

Для розробки клієнтської частини будуть застосовані такі засоби розробника, як:

- HTML;
- SCSS;
- JavaScript.

Задля коректності роботи програмного комплексу та покращення взаємодії користувача з вебдодатком особливу увагу варто приділити застосувати валідацію даних на всіх етапах.

РОЗДІЛ 3

РЕАЛІЗАЦІЯ ПРОГРАМНОГО КОМПЛЕКСУ ДЛЯ СТВОРЕННЯ ONLINE-МЕНЮ

3.1. Реалізація цільової сторінки продукту для заохочення клієнтів

Важливим етапом реалізації програмного продукту є заохочення та залучення клієнтів. Враховуючи те, що у даній сфері ринку існує певна конкуренція, необхідно попіклуватися про можливість розповісти та познаймити потенційного клієнта з продуктом, тим самим змотивувати його перейти безпосередньо до роботи з функціоналом програмного продукту. Для таких цілей доцільно використовувати цільову сторінку (Landing page).

Цільова сторінка – це окрема сторінка, створена в рекламних або маркетингових цілях для підвищення конверсій.

Конверсія – це співвідношення числа користувачів, які вчинили корисну дію, до кількості відвідувачів сайту за той же період часу, виражене у відсотках. Фактично даний показник вказує на ефективність та рентабельність ресурсу.

Landing page по суті закликає користувача зробити певну дію. Лендінги відіграють вирішальну роль у збільшенні впізнаваності бренду, підвищенні продажів, покращенні SEO показників та залученні нових клієнтів. Тобто, дана сторінка є ефективним способом розширити клієнтську базу. Отже, враховуючи те, що цільова сторінка можна сказати виконує конкретну стратегічну роль, вона має відповідати певним вимогам, таким як:

- 1) спонукати відвідувачів перейти до створення QR-меню;
- 2) переконати користувача у необхідності та актуальності продукту;

Кафедра КІТ (47)				НАУ 21 09 94 000 ПЗ			
Виконала	Молодоженя Д.С.			РЕАЛІЗАЦІЯ ПРОГРАМНОГО КОМПЛЕКСУ ДЛЯ СТВОРЕННЯ ONLINE-МЕНЮ	Літ.	Арк.	Аркушів
Керівник	Райчев І.Е.					61	24
Консульт.					УС-201Мз 122		
Н. контр.	Райчев І.Е.						

- 3) стисло та ясно описати продукт;
- 4) вказати переваги продукту.

Отже, дана цільова сторінка складатиметься із чотирьох основних секцій, які відповідатимуть описаним вище вимогам. Важливо, що інформація представлена на даній сторінці повинна бути стислою, максимально інформативною для того, щоб зацікавити користувача. Тобто, відповідно дослідженням є лише 6 секунд, щоб зацікавити користувача і він зупинився на даній сторінці, у іншому випадку користувач переходить до наступного ресурсу, продовжуючи свій пошук.

Лаконічний, візуально приємний дизайн стане перевагою у гонитві за пошуком клієнтів. Не варто використовувати занадто яскраві та активні відтінки чи агресивні кольори, акцент слід робити на тому, на які дії ви бажаєте стимулювати користувача: відправлення форми, перехід за посиланням, натиск кнопки, тощо.

Результатом є цільова сторінка (Рис.3.1), витримана у світлій кольоровій гамі з використанням акцентного більш яскравого блакитного відтінку для привернення уваги та виділення певних аспектів. У кожній із секцій чітко виділяється кнопка для переходу в адмін-панель користувача створення меню. Її повторення дає можливість перейти до основної дії при будь-якому переконливому аргументі, будь-то на етапі розповіді про продукт, його переваги чи актуальність для користувача. Так як ми є зацікавленими у натиску даної кнопки користувачем, то її повторення скоріш за все підвищить ефективність та ймовірність її натиску, при тому користувачу не доведеться гортати сторінку вгору чи вниз в пошуках даної кнопки.

Чітко описаний алгоритм роботи з сервісом дає розуміння чи співпадають очікування і бажання клієнта з можливостями продукту. Ілюстративний матеріал демонструє певні результати, які клієнт може отримати при роботі з сервісом. Стилістичний розподіл текстових блоків полегшує сприйняття інформації.

QR-меню для вашого закладу

Створіть власне цифрове меню абсолютно
БЕЗКОШТОВНО.

Створити QR-меню

Дізнатись більше >



QR-меню - це не просто спосіб виділитись, це необхідність!

В умовах сьогодні QR-меню - це перш за все зручність
та безпека для Ваших відвідувачів та персоналу.

Створіть легко та швидко цифрове меню для Вашого
закладу в декілька простих кроків. Для цього Вам не
потрібні особливі навички роботи з комп'ютером, а лише
бажання. Ми допоможемо зробити ваш заклад більш
сучасним та прогресивним!

Розпочати



Створіть цифрове меню просто та швидко

1 Заповніть необхідну інформацію

Просто введіть усю необхідну інформацію щодо
вашого закладу (адрес, соціальні мережі, короткий
опис), категорії меню та опис страв, їх ціни, склад і
т.п.

2 Оберіть візуальне оформлення

Оберіть шаблонний дизайн із запропонованих або
відредагуйте за власним побажанням, змінюючи
кольори та завантажуючи власні зображення

3 Отримайте сформований QR-код та посилання на меню

В результаті ви отримаєте посилання на готове
розроблене вами меню та QR-код з можливістю його
завантаження для подальшого друку.

Створити QR-меню



Наші переваги

- ✓ Зручний інтерфейс
- ✓ Безкоштовно
- ✓ Присмне візуальне оформлення
- ✓ Можливість створення двох і більше меню
- ✓ Оформлення відповідно до вашої концепції закладу

Створити QR-меню

Ми доведемо, що створити QR-меню легше ніж приготувати бутерброд

Зручний та інтуїтивно-зрозумілий
функціонал допоможе Вам
створити стильне цифрове меню з
можливістю його стилізації,
враховуючи власну концепцію чи
корпоративні кольори. В результаті
Ви отримаєте сформований QR-
код та посилання на Ваше меню.

Рис. 3.1. Landing Page

3.2. Реалізація інтерфейсу користувача

Продукт передбачає формування вебсторінки на основі даних введених користувачем. Тобто, потрібен зручний інтерфейс введення даних користувачем. Головний процес взаємодії користувача з програмним комплексом при створенні цифрового меню відбувається у три основні кроки.

- 1) “Базова інформація”.
- 2) “Формування меню”.
- 3) “Візуальне оформлення”.

Перший крок полягає у заповненні користувачем базової, основної інформації щодо закладу. Перед користувачем представлено три форми.

У першій формі (Рис.3.2) користувач має змогу завантажити фото закладу чи логотип, у наступних полях водиться назва закладу, номер телефону та короткий опис закладу. При чому передбачено можливість вводу одного і більше номерів телефону.

The screenshot shows a web form titled "Базова інформація" (Basic Information). At the top, there is a progress indicator with three steps: 1. Базова інформація (highlighted in blue), 2. Формування меню, and 3. Візуальне оформлення. The form itself has a light gray border and contains the following elements:

- Головне фото** (Main Photo): A gray square button with a camera icon, a blue "Завантажити" (Upload) button, and a white "Видалити" (Delete) button.
- Назва** (Name): A text input field with the placeholder text "Назва закладу" (Establishment name).
- Телефон** (Phone): A text input field with the placeholder text "+38(xxx)xx-xxx-xx". Below the field is a blue link with a plus icon and the text "Додати телефон" (Add phone).
- Опис** (Description): A large text area with the placeholder text "Розкажіть коротко про свій заклад" (Briefly describe your establishment).

Рис. 3.2. Форма для заповнення базової інформації

Наступна форма (Рис.3.3) призначена для введення адреси закладу, у випадку якщо це мережа закладів, то можливе додавання декількох адрес (Рис.3.4).

The image shows a web form titled "Адреса" (Addresses). It features a single text input field labeled "Адреса закладу" (Establishment address). Below the input field is a blue button with a plus icon and the text "Додати адресу" (Add address).

Рис. 3.3. Форма для введення адреси

The image shows a web form titled "Адреса" (Addresses). It features three stacked text input fields, each labeled "Адреса закладу" (Establishment address). The second and third input fields have a red 'X' icon on their right side, indicating an error. Below the input fields is a blue button with a plus icon and the text "Додати адресу" (Add address).

Рис. 3.4. Форма з декількома полями для введення адреси

Третя форма (Рис.3.5) дає змогу користувачеві додати посилання на соціальні мережі. Адже вони стали невід’ємною життя людства. На даний час соціальні мережі рахуються не просто засобом комунікації, а й потужним інструментом для бізнесу. Активне введення сторінок в соц. мережах допомагає в популяризації бізнесу, привертає уваги нових клієнтів та надає можливості користуватися бізнес-інструментами, які допомагають в плані маркетингу і дозволяють запускати рекламні компанії. Тому в основному заклади харчування активно введуть свої соціальні мережі для кращого інформування своїх постійних та потенційних клієнтів. Найпопулярнішими соціальними мережами на разі можна вважати:

- Instagram;
- Facebook;
- TikTok;
- Twitter.

Соціальні мережі

Instagram
www.instagram.com/

Facebook
www.facebook.com/

TikTok
www.tiktok.com/

Twitter
www.twitter.com/

Далі >

Рис. 3.5. Форма для введення соціальних мереж

Отже, ввівши усі необхідні дані, користувач натискає на кнопку “далі” та переходить до другого кроку – створення меню. Усі дані з форм відправляються на сервер та зберігаються у базі даних.

Першим етапом формування меню є вибір категорії (Рис.3.6). Користувач має змогу обрати категорію з можливих представлених або створити власну. Серед переліку шаблонних категорій представлені найбільш популярні, такі як:

- сніданки;
- закуски;
- гарніри;
- салати;
- гарячі страви;
- напої.

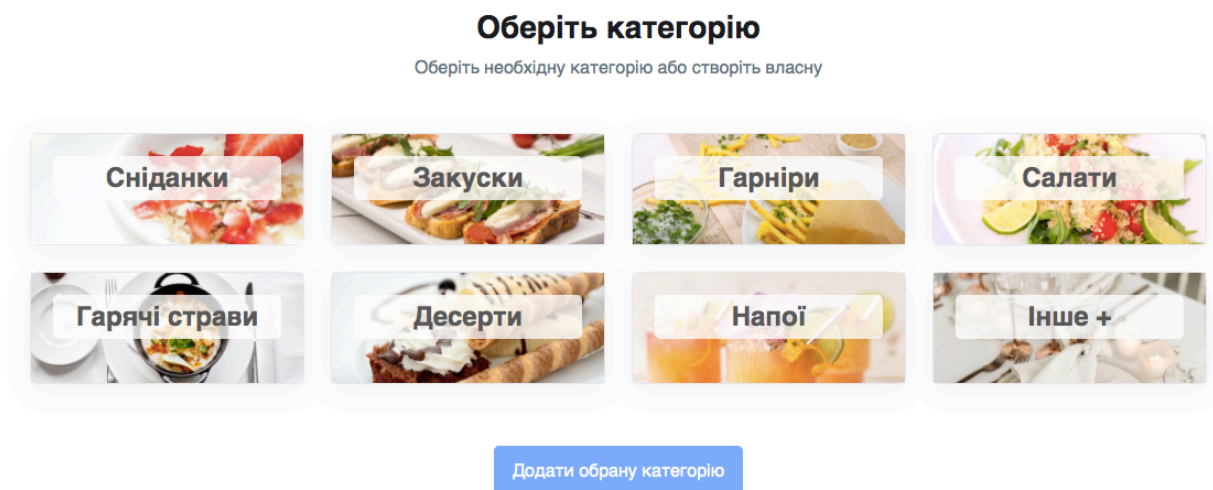


Рис. 3.6. Вибір категорію

При натисканні на одну з категорій кнопка “Додати обрану категорію” стає активною. У випадку, якщо представлені категорію не підходять користувачеві, то при натиску на блок “Інше +” з’являється поле для введення нової, необхідної для користувача категорії (Рис.3.7).

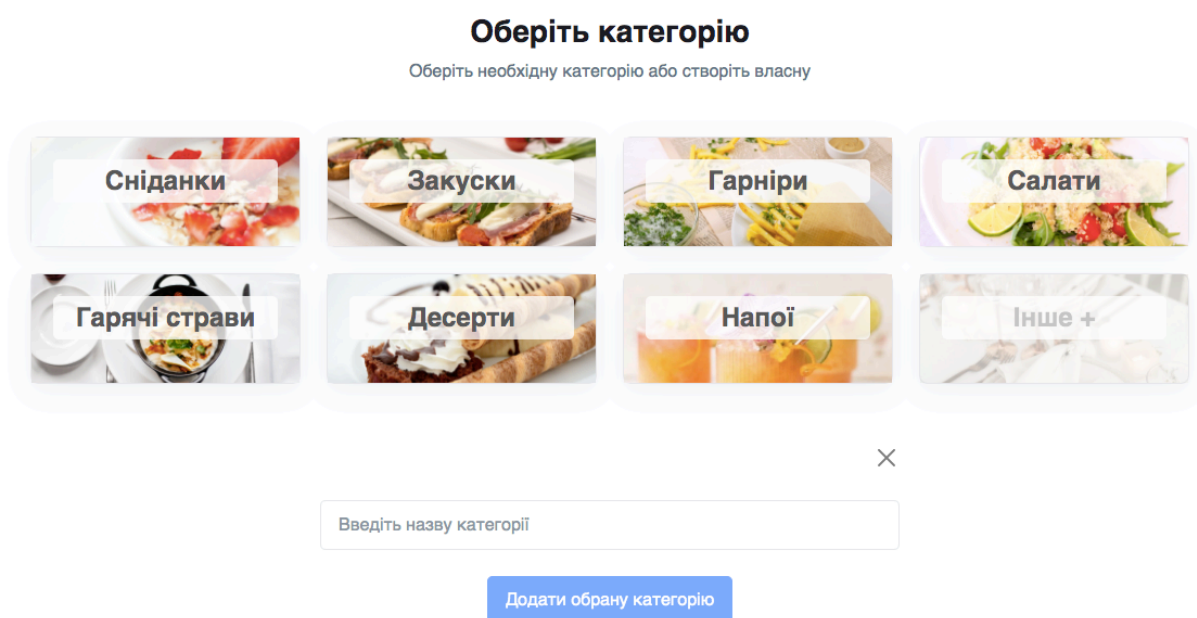


Рис. 3.7. Створення нової категорії

Задля уникнення помилок при заповненні поля, було прийнято рішення за допомогою JavaScript коду виконати перевірку. У випадку, якщо поле є пустим, кнопка “Додати обрану категорію” є не активною, але при введенні значення кнопка стає активною і з’являється можливість обрати введену категорію (Рис.3.8).

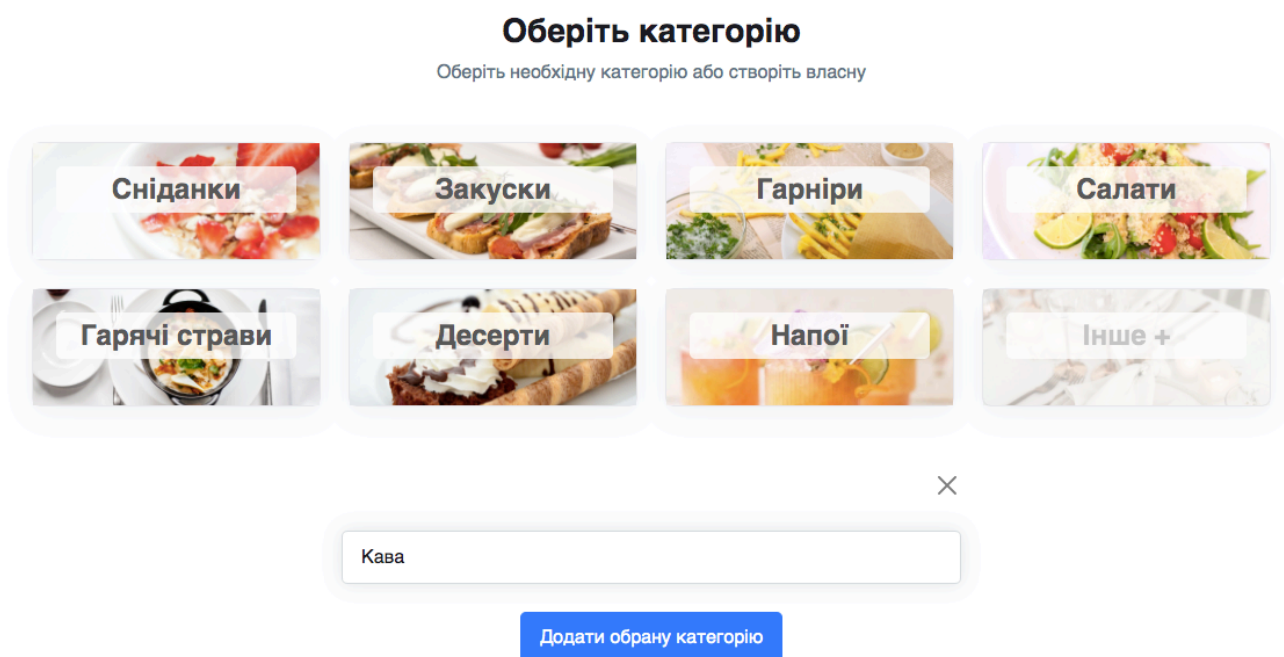


Рис. 3.8. Поведінка кнопки при введенні значення поля

Усі обрані та введені категорії автоматично генеруються у секції нижче (Рис.3.9), при кожному натиску кнопки “Додати обрану категорію”.











		СНІДАНКИ	+
		САЛАТИ	+
		ГАРЯЧІ СТРАВИ	+
		ДЕСЕРТИ	+
		НАПОЇ	+

Рис. 2.9. Список обраних категорій

Отже, на рисунку 3.9 представлений перелік категорій обраний у попередній секції. Зліва кожної категорії наявні іконки, при натисканні на які можливо видалити або редагувати назву категорії. При бажанні відредагувати, потрібно натиснути на іконку олівця, тоді поле ведення стане активним і з'явиться можливість редагувати текст. Після такого, як назву відредаговано потрібно натиснути на іконку прапорця в кружечку (Рис.3.10). І зміни будуть збережені.











 	СНІДАНКИ	+
 	САЛАТИ	+
 	ХОЛОДНІ СТРАВИ	+
 	ДЕСЕРТИ	+
 	НАПОЇ	+

Рис. 3.10. Процес редагування назви категорії

Справа знаходяться іконки зі знаком плюс, при натиску на них розгортається відповідна категорія (Рис.3.11). Це дає можливість для перегляду страв, які належать даній категорії та можливість додавання позицій (страв).

Оскільки на даному етапі ще не добавлено жодної страви, то їх перелік відсутній, наявний лише блок “Додати позицію”.

По мірі додавання нових позицій, вони буду з’являтися у переліку під обраною категорією. Дані дії потрібно буде виконати для кожної категорії з переліку обраних.

Наступним етапом формування меню буде власне створення та додавання позицій меню.

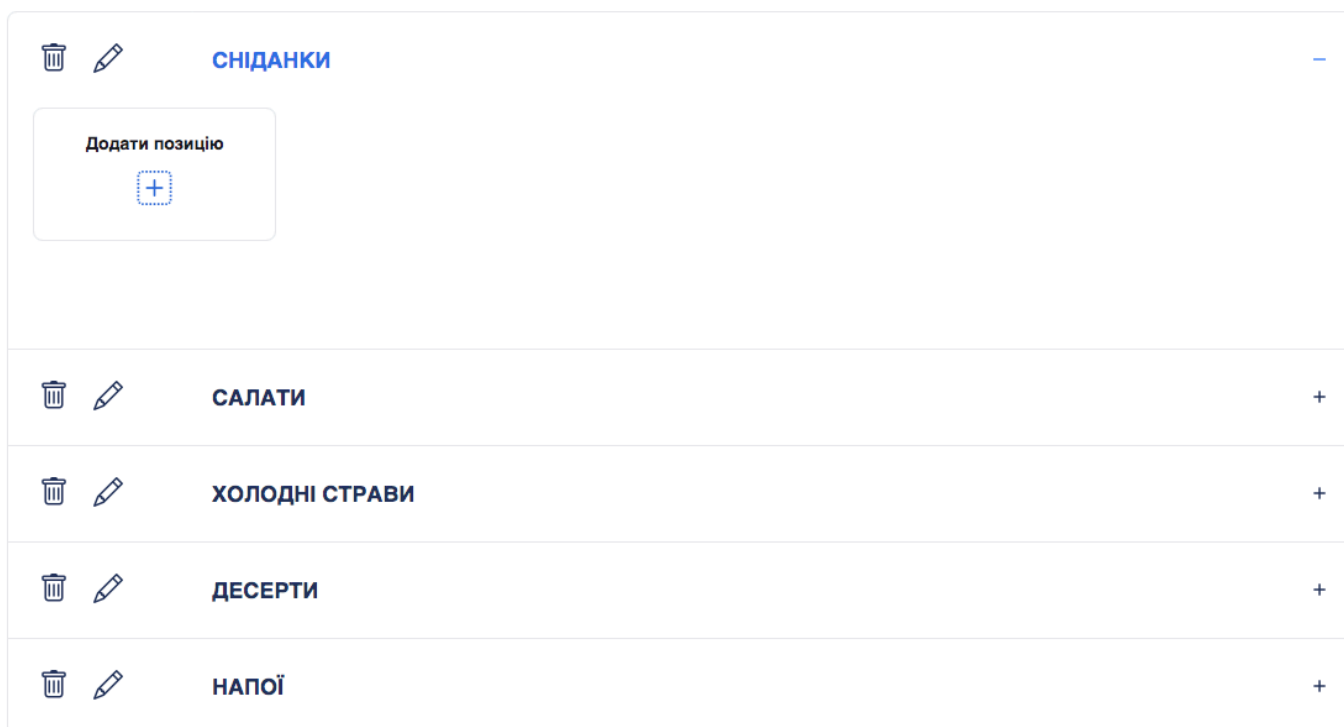


Рис. 3.11. Розгортка категорії “Сніданки”

При натисканні “Додати позицію” з’являється модальне вікно з формою для введення необхідної інформації щодо позиції меню (Рис.3.12).

Тобто, користувачеві попередньо варто мати, до прикладу, якийсь певний список страв та напоїв, які йому потрібно буде додати та визначитися, яку саме інформацію варто заповнити.

Усі поля форми не є обов’язковими для заповнення, окрім назви та ціни. Тому у цьому випадку, заповнення і вигляд майбутнього цифрового меню залежить повністю від бачення користувача та його вподобань. До прикладу, якщо користувач має якісний фото контент страв, він має можливість його завантажити за допомогою форми. Це зробить його меню більш яскравим, та швидше за все можливо підвищить продажі. У разі відсутності ілюстративного матеріалу, користувач може ввести лише текстовий опис страви. У обох варіантах програмний комплекс пропонує цікаве стилістичне оформлення, яке користувач зможе обрати на третьому кроці “Візуальне оформлення”.

Назва

Ціна

Маса / Об'єм

Склад

Зображення

Калорійність

Завантажте зображення страви
Максимальний розмір файлу 2MB

Закрити

Додати позицію

Рис. 3.12. Модальне вікно з формою для додавання позиції меню

Отже, користувач має змогу ввести:

- назву страви;
- ціну;
- маса або об'єм (у випадку, якщо позиція відноситься до супів або напоїв);
- склад (короткий опис страви, або перелік інгредієнтів, додатково можна вказати алергени наявні у страві);

- калорійність (для людей, котрі притримуються дієти або правильного харчування підрахунок калорійності страви буде доволі актуальним).

Після того, як користувач ввів усю необхідну інформацію, він натискає кнопку “Додати позицію” і вона автоматично з’являються у переліку страв в розгорті відповідної категорії (Рис.3.13).

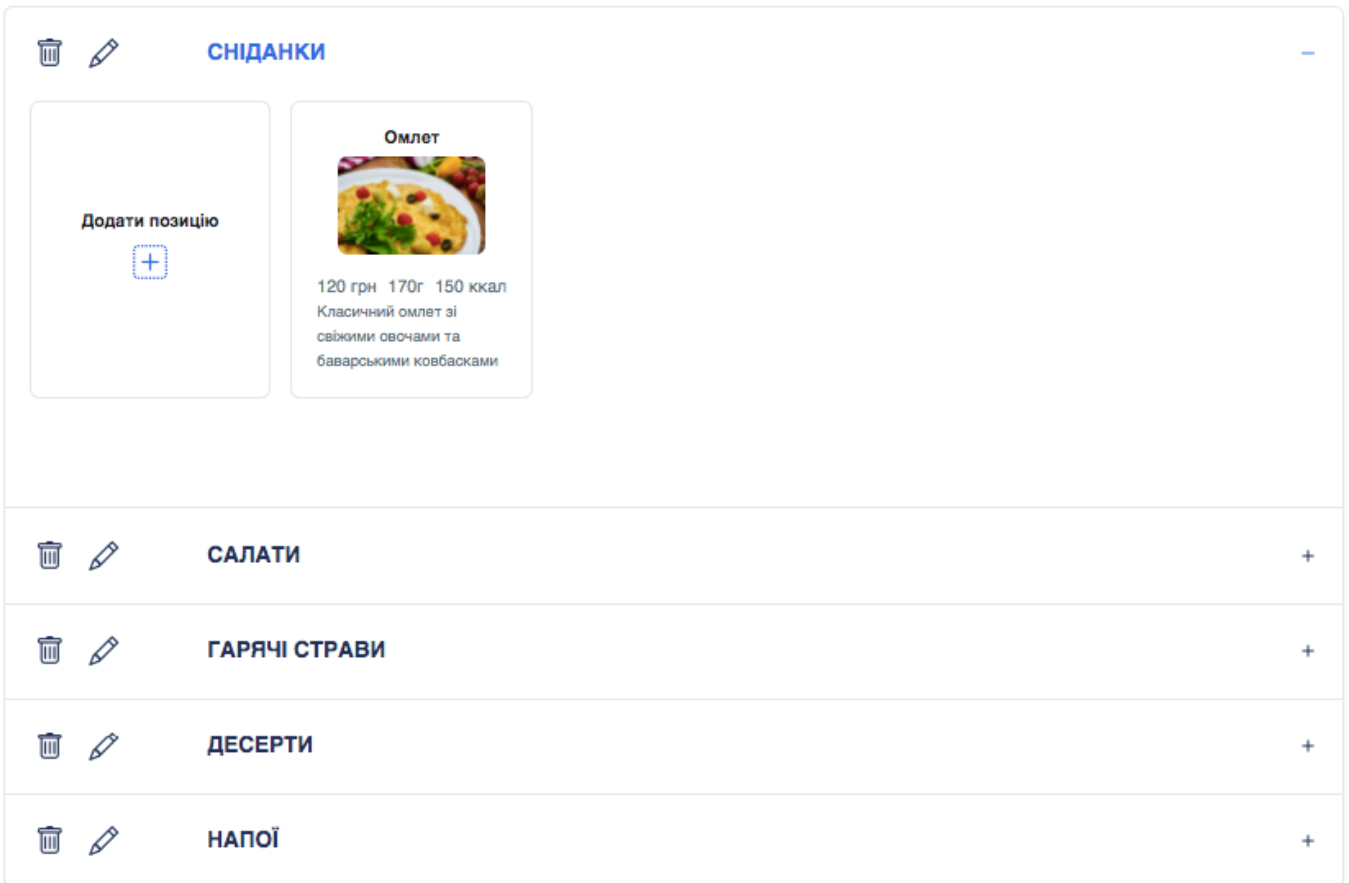


Рис. 3.13. Додавання позиції меню заповнену випадковими даними

Таким чином відбувається заповнення меню користувачем. Користувач за допомогою модального вікна з формою заповнює необхідну кількість страв для кожної категорії. Цей процес є найбільш часозатратним, але чіткий алгоритм дій та зручний інтерфейс полегшують та оптимізують його.

Проінформуйте своїх клієнтів щодо акційних пропозицій

Завантажте банери у форматі .jpg та розмірами 1100 x 280px з актуальними акційними пропозиціями, які проходять у вашому закладі



Перетягніть необхідну картинку у дане поле
або

Завантажте картинку

Приклад відображення банерів



< Назад

Далі >

Рис. 3.14. Секція завантаження акційних банерів

На рисунку 3.14 представлена наступна секція, за допомогою якої користувач має змогу завантажити зображення, які будуть представлені на вебсторінці цифрового меню. Дані зображення будуть знаходитись вверху сторінки і нестимуть інформативний характер. Тобто, користувач з їх допомогою має можливість презентувати нові акції, новинку меню, або ж, до прикладу, проінформувати про певні зміни відносно роботи закладу. Дана секція не є обов'язковою для заповнення, лише за бажанням користувача.

Для кращого розуміння, яким чином відбудеться візуальне оформлення даного блоку, нижче на рисунку 3.14 наведений приклад відображення банерів.

Важливо зауважити, що користувач має можливість завантажити не лише один, а й більшу кількість зображень. Тоді їх відображення матиме вигляд гарелеї, де за допомогою кнопок у вигляді стрілок, розташованих по боках банеру, клієнт зможе переглядати усі представлені зображення. Таке візуальне оформлення привертатиме увагу і не займатиме багато місця на вебсторінці. Враховуючи, його розміщення та стилізацію, для користувача вказано найбільш відповідні розміри банеру для його коректного відображення.

Внизу сторінки наявні дві кнопки, за допомогою яких користувач у тому разі, якщо заповнив усю необхідну інформацію, що стосується формування меню може перейти до наступного кроку, або ж у разі необхідності повернутись назад для того, щоб внести певні коригування у першому кроці “Базова інформація”.

Отже, наступним кроком створення меню являється вибір візуального оформлення. На даному етапі користувач має можливість стилізувати меню на власний розсуд, бажання, дотримуючись корпоративних кольорів чи т.п. Тобто, це дозволить зробити меню більш персоналізованим та унікальним.

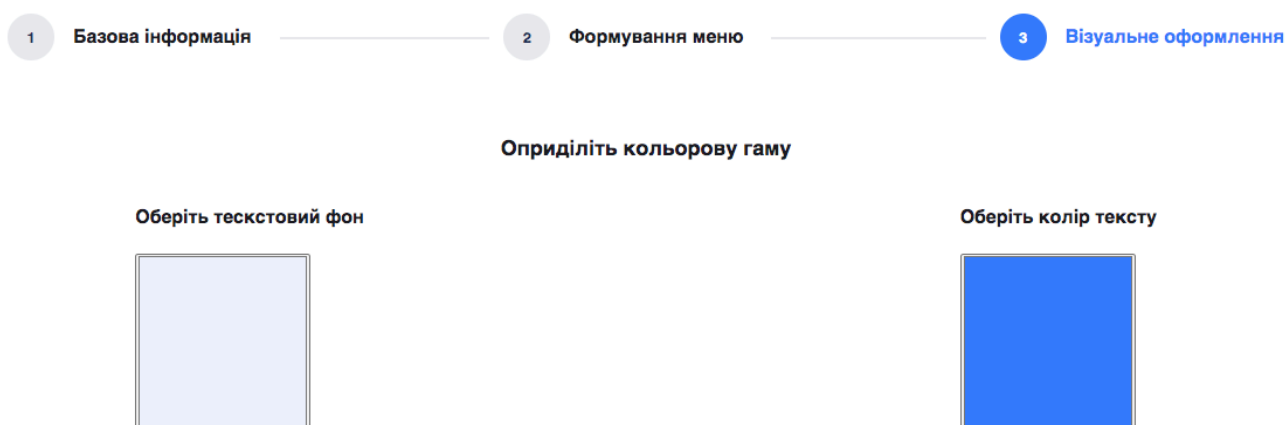


Рис. 3.15. Вибір кольорової гами

На рисунку 3.15 представлено форму вибору кольорів для стилізації вебсторінки цифрового меню.

Форма вибору кольору реалізована за допомогою елементу `input` з атрибутом `type= "color"`. У першому полі користувач може вибрати колір текстового фону, а у другому – колір тексту. Користувач має можливість вести колір у форматі RGB, HSL, HEX, або ж просто обрати за допомогою `color picker` (Рис. 3.16).



Рис. 3.16. Color Picker

Обрані кольори автоматично застосовуються до стилів заголовку вище (Рис.3.17), і таким чином відразу користувач зможе переглянути та оцінити, чи до вподоби йому таке поєднання кольорів та чи відповідає це його очікуванням. Це полегшить вибір кольорів та попередній огляд допоможе уникнути розчарування в кінцевому результаті.

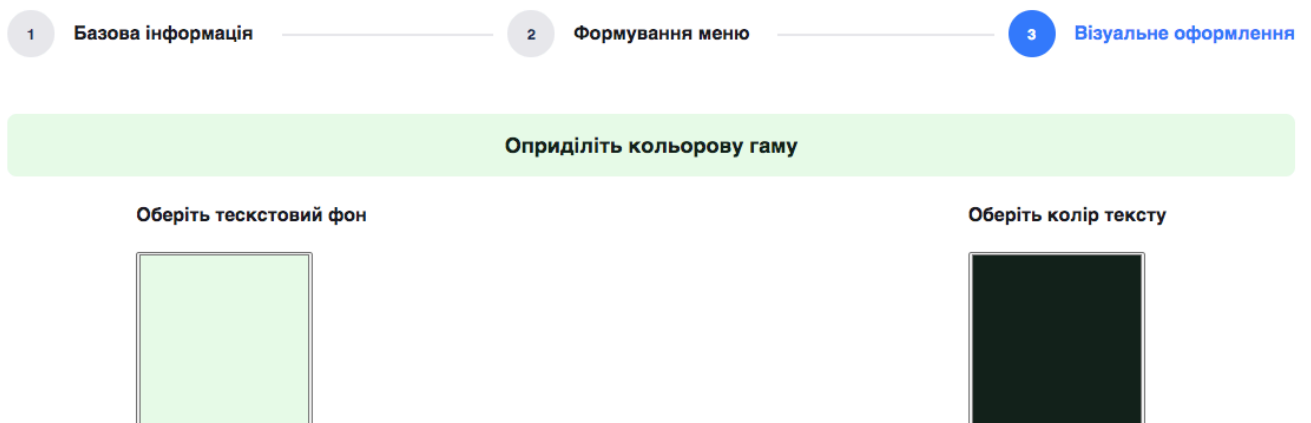


Рис. 3.17. Автоматичне застосування обраних кольорів

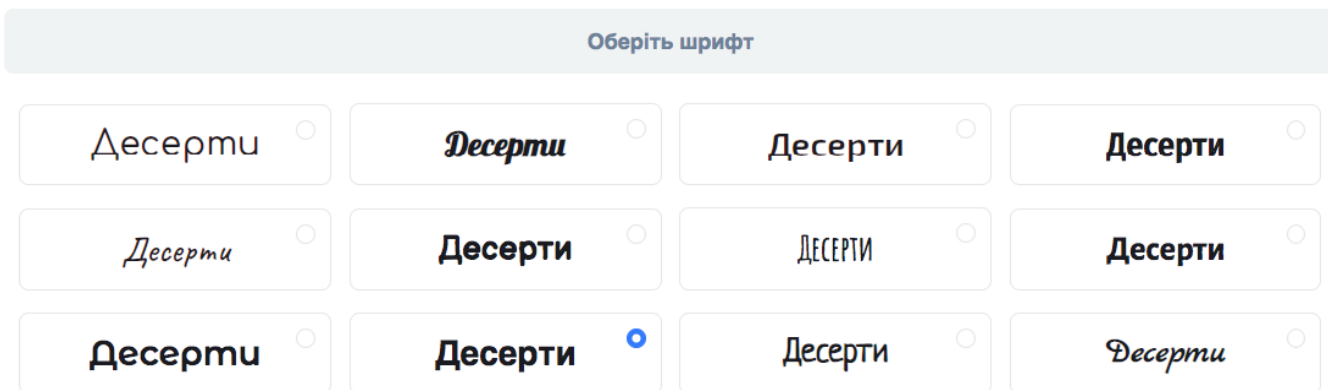


Рис. 3.18. Секція вибору шрифту

На рисунку 3.18 користувач має змогу обрати шрифт у який використовуватиметься у згенерованій вебсторінці. У даній секції представлено дванадцять варіантів вибору, серед яких наявні як і класичні види шрифтів, так і більш специфічні. На меті було охопити максимальний кількість варіантів, щоб задовільнити будь-якого клієнта з будь-якою концепцією закладу.

Наступний етап полягає у виборі представлення інформації що стосується закладу, так би мовити “шапка” вебсторінки цифрового меню. На рисунку 3.19 на вибір показано два шаблони представлення даного інформаційного блоку.

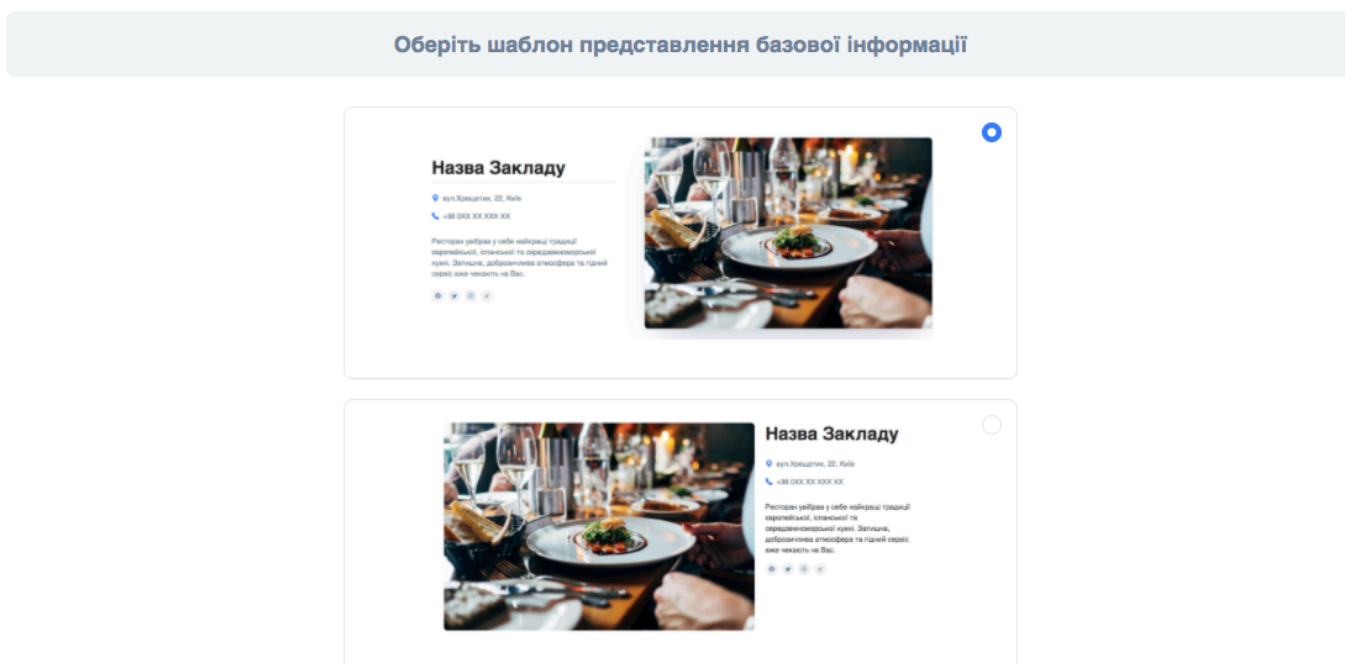


Рис. 3.19. Вибір шаблону представлення базової інформації

В цілому на шаблонах представлено місце розташування елементів, які користувач вказує у на першому кроці “Базова інформація”, серед них:

- головне фото закладу;
- адреса закладу;
- короткий опис закладу;
- посилання на соціальні мережі у вигляді відповідних іконок.

Варто зауважити, що даний шаблон є приблизним, він видозмінюватиметься відносно того скільки і які дані введе користувач. Та на його стилізацію впливатиме вибір шрифту та кольорової гами зроблений користувачем на третьому кроці – вибору візуального оформлення. Тобто, за допомогою даної форми користувач обирає лише розташування елементів – де знаходитиметься головне фото, текст справи чи зліва, чи матиме фото тінь і т.д.

Наступна форма (Рис. 3.20, 3.21, та 3.22) пропонує вибір з трьох шаблонів представлення самого меню закладу. Наявні шаблони пропонують варіанти меню з зображеннями страв та у разі їх відсутності.

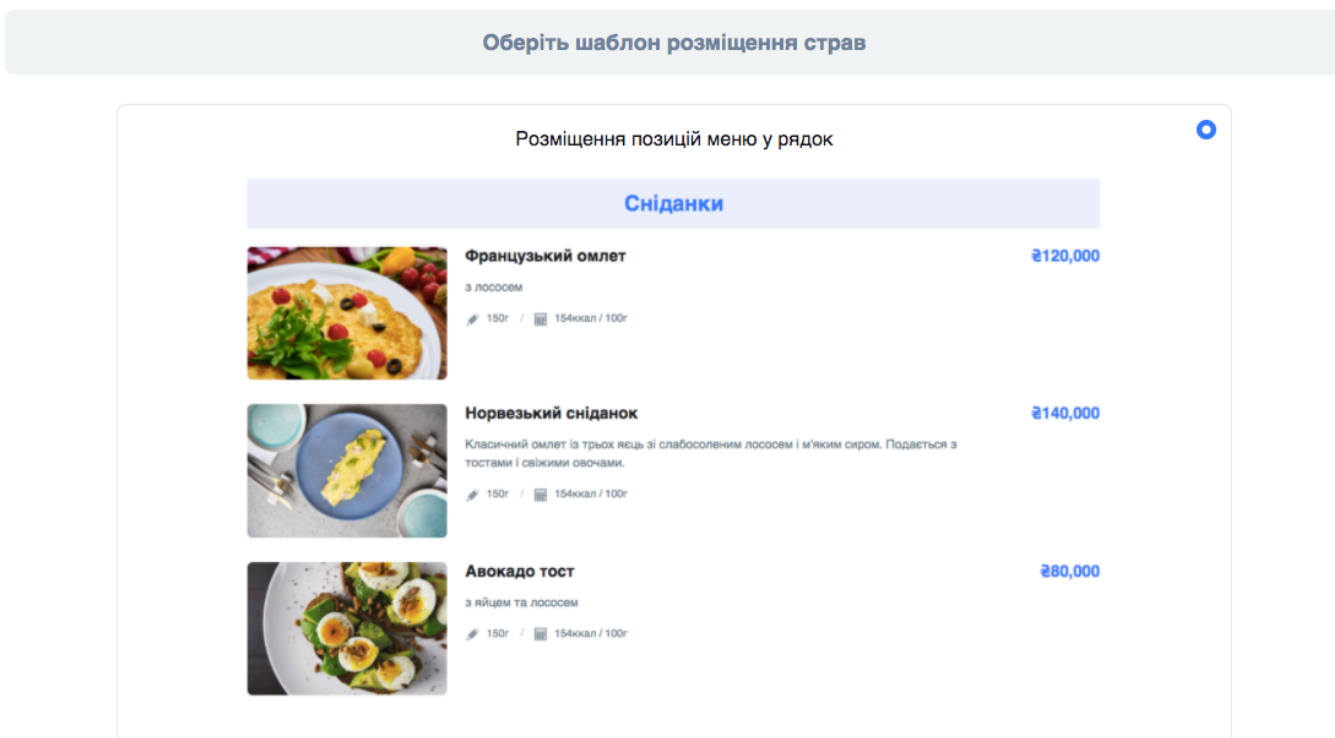


Рис. 3.20. Перший шаблон розміщення страв

Перший варіант (Рис.3.20) пропонує розміщення страв у один рядок. Тобто, у даному варіанті фото страв матимуть більший розмір, секції відділятимуться між собою за допомогою стилів. Кольори у яких представлена назва категорії та ціна будуть відповідати обраним користувачем.

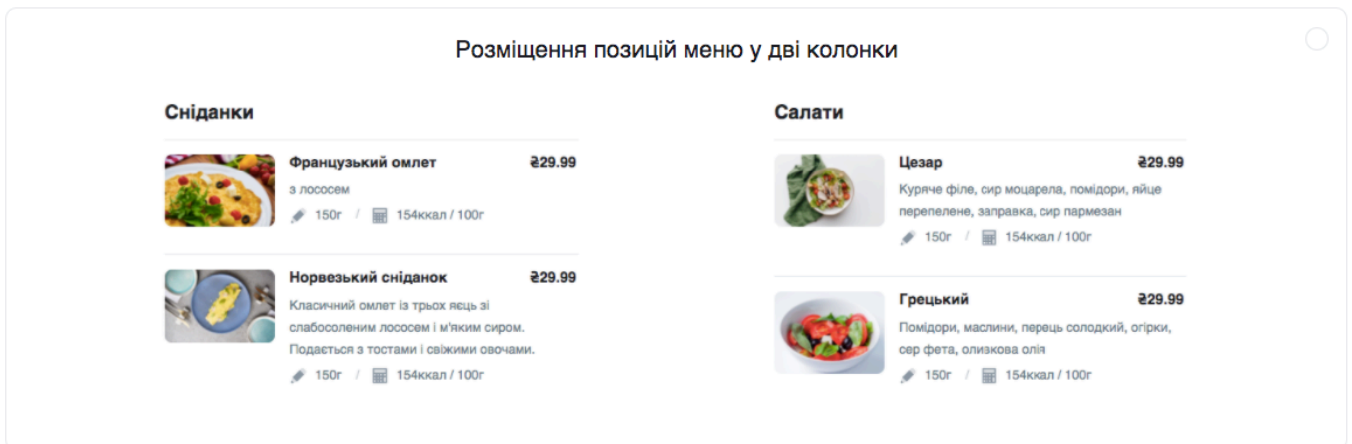


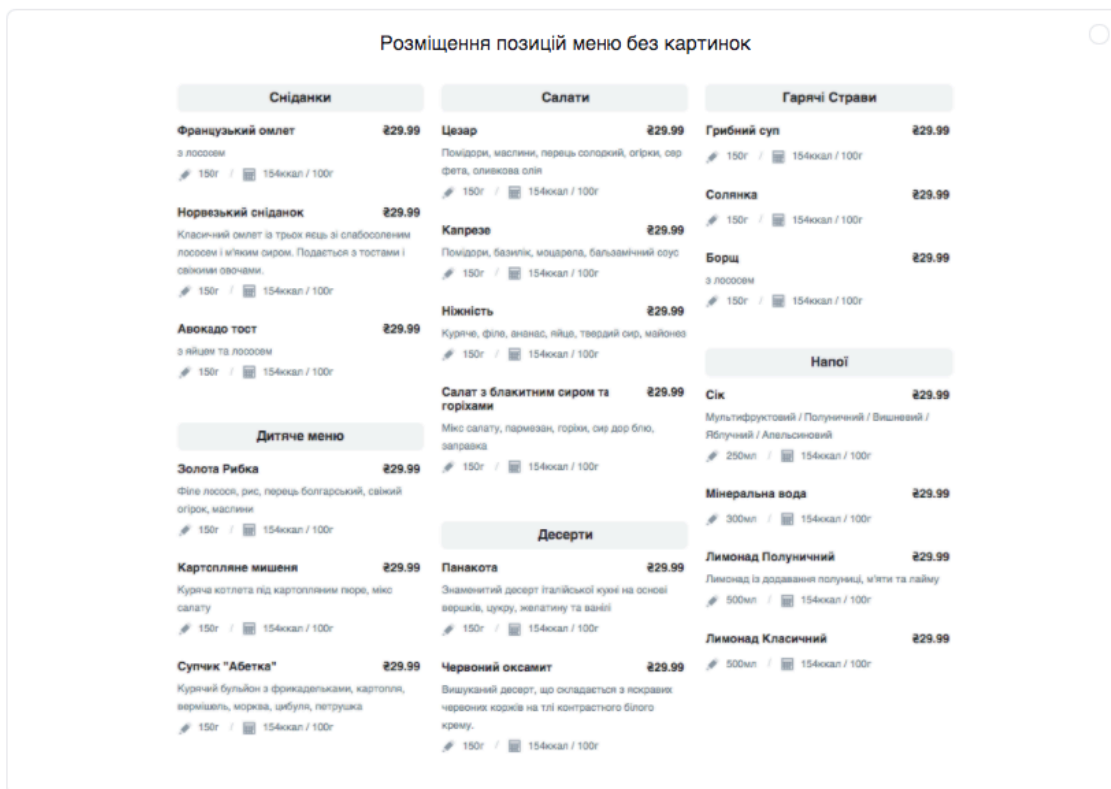
Рис. 3.21. Другий шаблон розміщення страв

Другий шаблон (Рис. 3.21) також призначений у тому випадку, якщо присутні зображення страв. У даному випадку розміщення позицій меню представлено у дві колонку.

Такий варіант розміщення є більш компактним і краще підійде у тому разі, якщо заклад має широкий асортимент страв, тоді список не матиме довжелезний вигляд, і клієнтам не доведеться витратити багато часу, щоб його гортати.

Третій шаблон (Рис. 3.22) має лаконічний дизайн і найкраще підійде у випадку відсутності фото-контенту. Категорії розподілені на три колонки.

Стилізація категорій залежатиме від вибору кольорової гами користувачем. Тобто, у разі вибору більш яскравих кольорів, меню матиме інший вигляд, маючи акцентні елементи.



[Назад](#)

[Створити меню](#)

Рис. 3.22. Третій шаблон розміщення страв

Варто зазначити, що кожен шаблон обдуманий та протилізований, головні елементи виділені жирним шрифтом, категорії розділені між собою.

Меню в кожному варіанті є зручним для перегляду та читабельним. Чітко виділена назва страви та її ціна, а також розподілення по категоріям дозволять клієнтам закладу легко орієнтуватися у меню. Уся необхідна інформація щодо страв є представленою.

Також розглянуті випадки розташування елементів у випадку, якщо деякі з них будуть відсутні. Тобто, якщо користувач програми, до прикладу, не вважатиме за потрібне вказувати калорійність чи опис страви, то це не матиме суттєвого впливу на дизайн меню.

Нижче на рисунку 3.22 представлені дві кнопки, за допомогою яких користувач має можливість або повернутись до попереднього кроку або створити меню. При натисканні кнопки “Створити меню” збираються дані з форм усіх трьох кроків.

На цьому етапі процес створення меню користувачем закінчується, і програма сама формує вебсторінку з меню на основі вказаних даних та генерує посилання на неї і QR-код.

3.3. Формування вебсторінки меню та генерація посилання та QR-коду

Усі дані вказані користувачем збережені у базі даних. На основі цих даних формується зміст та стилізація вебсторінки. Запит на отримання даних йде на сервер, який взаємодії з базою даних та в форматі JSON надає відповідь.

Після створення меню, користувач отримує посилання та QR-меню (Рис.3.23). За посиланням користувач має змогу перейти на створену вебсторінку з меню його закладу. Дане посилання він також може розмістити у соціальних мереж, щоб його клієнти могли ознайомитись із меню.

Користувач має змогу завантажити QR-код, натиснувши на відповідну кнопку і завантаження почнеться автоматично. Зображення QR-коду він може розмістити у своєму закладі, наприклад, на столах для зручного представлення своїм клієнтам.

За допомогою телефону з камерою кожен клієнт закладу може просканувати QR-код та потрапити на вебсторінку з меню даного закладу. Ознайомившись з меню, клієнт покличе офіціанта та зробить замовлення.

Таке рішення зменшує навантаження на офіціантів закладу і, звісно, в часи пандемії мінімізує ризики зараження вірусом.

До того ж клієнти, що планують візит до закладу, можуть ознайомитись з меню зарання, за допомогою посилання та вирішити чи до вподоби їм кухня закладу та чи прийнятна для них цінова політика.

Вітаємо Ваше меню готове!

Ви отримуєте посилання та QR-код

<http://localhost:3000/menu/614c8ad8d3e50cbd38c95e76>

Скопіювати



**Завантажте QR-код
з посиланням на
Ваше меню для
подальшого його
використання,
друку та ін.**

Завантажити QR-code



Рис. 3.23. Посилання та QR-код на вебсторінку створеного меню

Для прикладу пройшовши увесь момент створення меню, заповнивши усю необхідну інформацію випадковими даними, можемо оцінити результат роботи програмного комплексу (Рис.3.24).

Усі введені дані представлені коректно, посилання на соціальні мережі є активними. Акційні банери розташовані відразу під інформацією про заклад, щоб привернути увагу клієнтів. Розташування меню було обране у дві колонки. Відповідно до цього можна зробити висновок, що функціонал програмного продукту повністю відповідає поставленим задачам та вимогам.

Claude Monet

вул.Хрещатик, 22, Київ

+38 067 55 555 75

Ресторан увібрав у себе найкращі традиції вишуканої французької кухні. Затишна, доброзичлива атмосфера та гідний сервіс вже чекають на Вас.



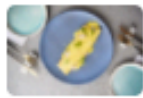
Сніданки



Французький омлет €29.99

з лососем

🍴 150г / 📊 154ккал / 100г



Норвезький сніданок €29.99

Класичний омлет із трьох яєць зі слабосолим лососем і м'яким сиром. Подається з тостами і свіжими овочами.

🍴 150г / 📊 154ккал / 100г

Салати



Цезар €29.99

Куряче філе, сир моцарела, помідори, яйце порепелене, заправка, сир пармезан

🍴 150г / 📊 154ккал / 100г



Грецький €29.99

Помідори, маслини, перець солодкий, огірки, сир фета, оливкова олія

🍴 150г / 📊 154ккал / 100г

Рис. 3.24. Фрагмент сформованої вебсторінки меню закладу

Важливо відміти, що в більшості випадків перегляд даного меню здійснюватимуть клієнти закладу за допомогою власних смартфонів, тому важливо коректно відображати вебсторінку на різних мобільних пристроях. Оскільки, при перегляді даної сторінки з телефону можуть виникати певні незручності та проблеми. Виходячи з цього втрачається читабельність та зручність, тому що аби побачити

певну інформацію користувачеві доведеться пролистувати, приближувати і т.п. Тому важливо було розробити мобільну версію. Створення мобільної версії було виконану за допомогою технології CSS – адаптивна верстка.

Адаптивна верстка – це верстка, при якій вебресурс коректно відображається на різних пристроях, залежності від поведінки користувача, платформи, розміру екрану та орієнтації девайса завдяки тому, що елементи динамічно підлаштовуються під різні розширення екрану. Використання медіа-запитів (@media), які дозволяють автоматично визначати роздільну здатність монітора, тип пристрою та підставляти задані значення в автоматичному режимі.

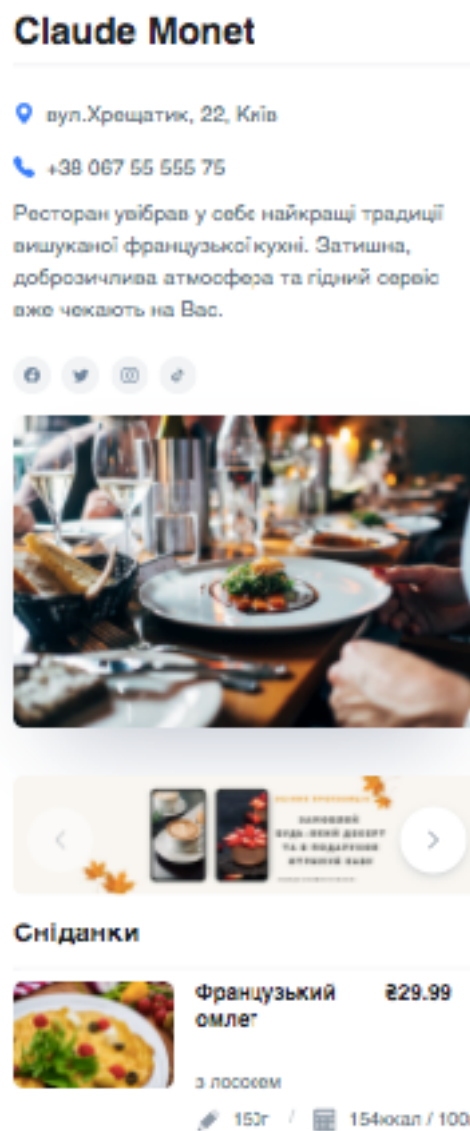


Рис. 3.25. Вигляд вебсторінки на мобільних пристроях

Для зручності перегляду вебсторінки з мобільних пристроїв більшість елементів розташовані в одну колонку. При адаптивній верстці зберігається читабельність, і користувачу не потрібно приближувати аби побачити ціну страви чи номер телефону, наприклад.

В результаті заклади харчування крокують поруч з розвитком сучасних технологій, оптимізують процес роботи, знижують затрати на паперове меню.

Висновки до розділу 3

За допомогою технологій розробки та засобів розробника створено програмний продукт, котрий на основі введених користувачем даних формує вебсторінку, посилення на неї та генерує QR-код.

За результатами тестування усі дані відображаються коректно, стилістичне оформлення виконується згідно обраних користувачем вимог у третьому кроці “Візуальне оформлення”. Функціонал програмного комплексу працює справно, неполадки відсутні.

Адаптивна верстка дає можливість переглядати вебсторінку з різних пристроїв, не порушуючи її цілісності, читабельності та зручності.

ВИСНОВКИ

Сучасний світ диктує нові правила. Так як пандемія COVID-19 внесли певні корективи в життя людей, то її вплив торкнувся кожної сфери. Сфера готельно-ресторанного бізнесу не стала винятком. Власники закладів харчування стикнулися з певними обмеженнями у роботі. Одне з таких обмежень стосувалося меню.

Зкладами харчування заборонили використовувати меню зроблене з матеріалу, який не підлягає дезінфекції. Оскільки, більшість ресторанів використовували паперове меню, то вони стикнулися з вибором або зміни матеріалу, або переходом на цифрове меню.

Виготовлення меню з якісного, цупкого матеріалу, який підлягає дезінфекції потребує вкладення чималих коштів, тоді як цифрове меню є менш затратним та легким в редагуванні. Однак, важливий момент полягає у зручному доступі до цифрового меню. Просто url-посилання не є зручним, введення посилання в адресний рядок займатиме багато часу та впливатиме на загальне враження клієнтів про заклад в цілому. Тобто, такий процес може викликати відчуття дратівливості. У тому випадку потрібно надати користувачеві швидкий і простий доступ до меню. Вирішенням цієї проблеми стане QR-код. Розміщення QR-кодів у закладі дозволить користувачам максимально швидко відобразити вебсторінку на власному смартфоні за допомогою камери.

Отже, враховуючи усе вище сказане, розробка програмного комплексу для створення online-меню є, як ніколи, актуальною. Дані обмеження не просто створили попит на цифрове меню, а зробили його необхідністю для кожного закладу.

Для кращого розуміння запитів та пропозицій ринку у даній ніші, варто ознайомитись з представленими програмними продуктами. На основі їх переваг та недоліків можна сформулювати перелік вимог, щоб створити конкурентно-спроможний та клієнтоорієнтований програмний комплекс.

Для того, щоб оцінити можливості та функціонал представлених на ринку продуктів було зроблено їх детальний аналіз. Більшість з протестованих вебсервісів мали певні недоліки, такі як:

- незручний інтерфейс;
- некоректне відображення введених даних;
- відсутність функціоналу створення меню користувачем самостійно;
- неадаптивний дизайн;
- візуальне оформлення готового цифрового меню іноді важко назвати приємним;
- відсутність деяких основних функцій;
- відсутність можливості персоналізації меню;
- коштовність послуг програмного продукту.

З огляду на перелік недоліків розглянутих вебсервісів, можна зробити висновок, що дані продукти потребують розвитку, розширення функціоналу, доопрацювання та усунення несправностей. На основі проведеного аналізу було сформовано базовий перелік розділів, котрі повинні бути наявні у розроблюваному web-сервісі. Зокрема повинен бути реалізований розділ формування меню та його персоналізації, створення категорій, додавання позицій. Тобто алгоритм оформлення меню має бути комбінованим: користувач повинен мати можливість обрати з поміж декількох варіантів стилізацію меню, або мати змогу завантажити власні зображення та використовувати корпоративні кольори, використовуючи зручний інтерфейс.

Продуманість інтерфейсу займає важливу роль у розробці програмного комплексу. Інтерфейс внесення інформації для створення меню повинен бути простим, інтуїтивно-зрозумілим та візуально приємним.

Особливу увагу важливо приділити дизайну результуючої сторінки з цифровим меню. Web-сторінка повинна відображатись у зручній формі для клієнтів закладу, візуально приємно розміщувати усю внесену інформацію, коректно відобразити ілюстрації. Перелік категорій та позицій меню має бути читабельним та гарно простилізованим. Дизайн має бути оформлений таким чином, щоб привертати увагу

клієнта до потрібної інформації. Вебсторінка повинна мати коректне відображення на усіх пристроях.

Також одним із обов'язкових є етап розробки модулю генерації QR-коду з можливістю його завантаження для користувача.

Після сформування переліку вимог і певного бачення продукту, було оглянуто технології розробки та принципи реалізації продукту.

Клієнтська частина програми була реалізована з допомогою мови гіпертекстової розмітки HTML, мови програмування JavaScript та стилізована з використання синтаксису SCSS.

Серверна частина виконана за допомогою технології Node.js та фреймворку Nest.js, які дозволить створити добре тестований, масштабований та легкий в обслуговуванні продукт.

Керування, зберігання та отримання інформації здійснюється завдяки базі даних MongoDB, яка дозволяє збереження неструктурованих даних у форматі JSON.

За результатами тестування розроблений програмний комплекс працює коректно, усі дані відображаються правильно, стилістичне оформлення виконується згідно обраних користувачем вимог у третьому кроці “Візуальне оформлення”. Функціонал програмного комплексу працює справно, неполадки відсутні. Усі висунуті вимоги виконані та реалізовані.

З використання медіа-запитів (@media), які дозволяють автоматично визначати роздільну здатність монітора, тип пристрою та підставляти задані значення в автоматичному режимі, було виконано адаптивну верстку. Тобто, відображення вебсторінки з різних пристроїв не порушують її цілісності, читабельності та зручності.

СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Шафер Стивен. HTML, XHTML и CSS: библия пользователя / Стивен Шафер. — М.: «Диалектика», 2011. — 656 с.
2. Макфарланд Дэвид Соьер. Новая большая книга CSS / Дэвид Соьер Макфарланд. — Санкт-Петербург: Питер, 2017. — 720 с.
3. Фаулер Мартин. Рефакторинг: улучшение проекта существующего кода / Мартин Фаулер, Кент Бек, Джон Брант, Уильям Опдайк, Дон Робертс. — СПб.: «Диалектика», 2019. — 448 с.
4. Фаулер Мартин. Рефакторинг кода на JavaScript: улучшение проекта существующего кода / Мартин Фаулер. — СПб.: «Диалектика», 2019. — 464 с.
5. Фаулер Мартин. UML. Основы. Краткое руководство по унифицированному языку моделирования / Мартин Фаулер, Кендалл Скотт. — 2-е. — М.: Символ-Плюс, 2002. — 192 с.
6. Буч Г. Язык UML. Руководство пользователя /. Г. Буч, Д. Рамбо, И. Якобсон. — 2-е. — М. : ДМК Пресс, 2006. — 496 с.
7. Дейли Брэд. Разработка веб-приложений с помощью Node.js, MongoDB и Angular: исчерпывающее руководство по использованию стека MEAN / Брэд Дейли, Брендан Дейли, Калев Дейли. — 2-е изд. — Санкт-Петербург: «Диалектика-Вильямс», 2020. — 656 с.
8. Янг А. Node.js в действии / А. Янг, Б. Мек, М. Кантелон. — 2-е изд.. — СПб.: «Питер», 2018. — С. 432.
9. Макфарланд Дэвид. JavaScript и jQuery: исчерпывающее руководство / Дэвид Макфарланд. — М.: «Диалектика», 2015. — 880 с.
10. Дакетт Джон. Javascript и jQuery. Интерактивная веб-разработка / Дакетт Джон. — М., 2017. — 640 с.
11. Никсон Р. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5. 4-е изд / Р. Никсон. — СПб.. — М.: «Диалектика», 2016. — 768 с.

12. Пауэрс Ш. Изучаем Node. Переходим на сторону сервера / Ш. Пауэрс. — СПб.: «Питер», 2017. — 304 с.

13. Флэваган Дэвид. JavaScript: карманный справочник, 3-е изд / Дэвид Флэваган. — М., 2013. — 320 с.