

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ВАСИЛЯ СТУСА

БЕВЗ ДМИТРО МИКОЛАЙОВИЧ

Допускається до захисту:

завідувач кафедри
інформаційних технологій,
д-р техн. наук, доцент

_____ Тетяна НЕСКОРОДСЬКА

«_____» _____ 2022р.

**РОЗРОБКА ВЕБ-СЕРВІСУ ДЛЯ ЗАМОВЛЕННЯ ТА НАДАННЯ БІЗНЕС
ЗАВДАНЬ**

Спеціальність 122 «Комп'ютерні науки»

Кваліфікаційна (бакалаврська) робота

Керівник:

Павло РИМАР, старший викладач
кафедри інформаційних технологій

Оцінка: _____ / _____ / _____
(бали за шкалою ЄКТС / за національною шкалою)

Голова ЕК: _____
(підпис)

АНОТАЦІЯ

Бевз Д.М. Розробка веб-сервісу для замовлення та надання бізнес завдань. Спеціальність 122 «Комп'ютерні науки», освітня програма «Сучасні інформаційні технології та програмування». Донецький національний університет імені Василя Стуса, Вінниця, 2022.

Кваліфікаційна (бакалаврська) робота присвячена розробці веб-сервісу, що об'єднує замовників, яким потрібно створити будь-яку роботу, і компетентних виконавців, які шукають підробіток або додатковий заробіток.

Користувач має змогу зареєструватись як замовник послуг або спеціаліст і в залежності від цього буде надаватись певний функціонал, що можна використовувати. Веб-додаток розроблено за допомогою двох мов програмування: JavaScript та C#.

Ключові слова: веб-додаток, JavaScript, C#, сервіс онлайн-послуг, комерція.
53 сторінки , 44 рис., 1 табл., 31 джерело.

ABSTRACT

Bevz D.M. Development of a web service for ordering and providing business tasks. Specialty 122 «Computer Science», Educational program «Modern information technologies and programming», Vasyl' Stus Donetsk National University, Vinnytsia, 2022.

Qualifying (bachelor's) work is dedicated to the development of a web service that brings together customers who need to create any job, and competent performers who are looking for part-time or additional earnings.

The user has the opportunity to register as a customer or specialist and depending on this he will be provided with certain functionality that he can use. The web application is developed using two programming languages: JavaScript and C#.

Keywords: web-application, JavaScript, C#, online service, commercial.

53 pages, 44 figure, 1 table, 31 sources.

ЗМІСТ

ВСТУП	4
РОЗДІЛ 1. ПОСТАНОВКА ЗАДАЧІ ТА ОГЛЯД ІСНУЮЧИХ АВТОМАТИЗОВАНИХ СИСТЕМ ДАНОГО НАПРЯМУ	6
1.1 Постановка задачі	6
1.2 Огляд існуючих аналогів	6
Висновок до розділу 1	11
РОЗДІЛ 2. ОГЛЯД ІНСТРУМЕНТІВ ДЛЯ РОЗРОБКИ	12
2.1 Мова програмування C#	12
2.2 .NET-платформа	14
2.3 База даних SQL	14
2.4 Мова програмування JavaScript	15
2.5 Додаткові інструменти для front end	20
2.7 Програмні інструменти (середовище розробки)	27
Висновок до розділу 2	29
РОЗДІЛ 3. РОЗРОБКА WEB – ДОДАТКУ	30
3.1 Процес авторизації / реєстрації.	30
3.2 Основний функціонал	31
Висновок до розділу 3	47
ВИСНОВКИ	48
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	49

ВСТУП

Особливе місце у структурі господарювання посідає підприємництво у сфері побутових послуг. У всьому світі, у т.ч. і в Україні, постійно розширюють обсяги побутового обслуговування населення – організованої діяльності суб'єктів бізнесу, пов'язаної з наданням побутових послуг. Основним товаром у сфері побутового обслуговування є побутова послуга [1].

Побутова послуга – вид діяльності суб'єктів підприємницької діяльності, пов'язаної із задоволенням конкретної побутової потреби індивідуального замовника[2].

Відповідно до затверджених постановою Кабінету Міністрів України № 974 від 4 червня 1999 року «Правил побутового обслуговування населення», усі громадяни володіють однаковими правами на задоволення своїх потреб у цій сфері. Надання переваг будь-якому із них, прямі або непрямі обмеження у правах є недопустимими.

З появою на Українському ринку аналога американського сервісу «*Taskrabbit*», усе більше людей виявляють інтерес до подібної моделі бізнесу. Фактично даний сервіс не є чимось радикально новим, а є майданчиком оновлюваних міні тендерів, за які «борються», зареєстровані раніше виконавці. Подальші види реалізації, монетизації та виконання можуть різнитись.

Відразу відкинемо тендерні майданчики, що створені для державних закупівель та великого бізнесу. Усі ці майданчики відносяться до окремої категорії та ніяк не зацікавляють людину, яка хоче вигуляти собаку або покласти у себе вдома плитку. Та й у цілому буде дивно виглядати тендер з доставки піци на сайті «*Прозорро*» поруч із замовленням на ремонт дороги міста.

Аналізуючи ринок подібних проектів, можна виявити десятки (якщо не сотню) спроб зробити подібні сайти, проте переважна більшість із них не увінчалися успіхом. При цьому їх творці чесно віддали багато сил на розробку та безрезультатне розкручування своїх проектів. Причина цьому вкрай проста: складність і працездатність виведення цих проектів на масовий ринок. Якщо

локально вони можуть успішно працювати, то на глобальному ринку втрачають домінацію. Такий успішний запуск стартапів під силу тільки великим інтернет холдингам, у яких уже є багатомільйонна аудиторія.

Мета бакалаврської роботи: розробка веб-сервісу для надання населенню побутових послуг. Вивчення методики проектування веб-додатків на мовах програмування JavaScript та C#. За допомогою веб-сервісу ми зможемо працевлаштувати велику кількість людей та допомогти тим, хто цього потребує.

Задачами дослідження є:

- Огляд існуючих сервісів пошуку виконавців для побутових та бізнес задач за даною тематикою, виявлення їх переваг і недоліків.
- Ознайомлення та вивчення спеціалізованого програмного забезпечення.
- Розробка онлайн-сервісу, що допоможе вирішити проблеми людей щодо пошуку виконавців для завдань, що вони поставлять.

Об'єкт дослідження: онлайн-сервіс (майданчик) пошуку виконавців для разових побутових та бізнес-завдань.

Предмет дослідження: процес розробки веб-додатків.

Структура роботи: кваліфікаційна робота складається зі вступу, трьох розділів, висновків, списку використаних джерел.

Робота містить 53 сторінки, 44 рисунки, 1 таблицю та список літератури з 31 джерела.

РОЗДІЛ 1

ПОСТАНОВКА ЗАДАЧІ ТА ОГЛЯД ІСНУЮЧИХ АВТОМАТИЗОВАНИХ СИСТЕМ ДАНОГО НАПРЯМУ

У даному розділі наведена постановка задачі, а також проведено огляд існуючих програм за даною тематикою.

1.1 Постановка задачі

Розробити web-додаток для замовлення та надання бізнес завдань. Буде розроблений наступний функціонал: сторінка авторизації / реєстрації (аутентифікації), головна сторінка з можливими категоріями для замовлення, сторінка для створення замовлення, сторінка із відкритими замовленнями, сторінка перегляду окремого замовлення, сторінка з загальними замовленнями, можливість двофакторної авторизації, чат для спілкування адміністратора з відвідувачем сайту.

1.2 Огляд існуючих аналогів

На сьогоднішній день на ринку є сервіси, що надають подібні послуги. Ринок сервісів для замовлення побутових послуг в Україні в період 2017-2019 років розвивався зі щорічною середньою динамікою близько 19% [15]. У другому кварталі 2020 року зростання ємності ринку припинилося і намітилася зворотна тенденція через вплив введених карантинних обмежень, погіршення загальної економічної і суспільно-політичної ситуації в країні. У 2021 році знову почався ріст попиту, а це означає, що збільшилась і пропозиція.



Рисунок 1.1 – Динаміка обсягів реалізованих послуг підприємствами сфери послуг різним групам споживачів у 2016-2020 рр. в Україні, млрд. грн.

Моніторинг ринку сервісів для замовлення побутових послуг в Україні показує, що найбільшим попитом в українських споживачів користуються послуги:

- кур'єрської доставки;
- домашніх майстрів: сантехніків, електриків, складальників меблів та ін.;
- прибирання житла і офісів;
- дрібного ремонту та оздоблення приміщень;
- ремонту побутової техніки і електроніки.

Найбільшу частку в структурі ринку сервісів для замовлення побутових послуг в Україні займають послуги з доставки, які стали ще популярнішими в період карантину, особливо в великих містах.

Аналіз ринку сервісів для замовлення побутових послуг в Україні свідчить про те, що подальший розвиток галузі, що досліджується відбувається під впливом наступних факторів:

1. Трансформація форм організації роботи виконавців послуг. Усе більше фахівців на ринку сервісів для замовлення побутових послуг в Україні вибирають самозайнятість, яка дає їм великий контроль над робочим часом і рівнем доходів.

2. Відмова корпоративних споживачів від штатних фахівців з надання побутових послуг на користь разових замовлень або використання аутсорсингу в міру необхідності;
3. Зниження купівельної спроможності жителів країни, яке призводить до зменшення споживання послуг, що не відносяться до категорії першої необхідності;
4. Розширення асортименту ринку сервісів для замовлення побутових послуг і варіантів їх надання за рахунок науково-технологічного прогресу і підвищення рівня технічної грамотності населення;
5. Реакція державних структур і суспільства на пандемію COVID-19.

Послуги, які раніше люди отримували поза домом: перукарні, громадське харчування, офлайн магазини, тепер все більше надаються за місцем проживання клієнтів або дистанційно з наступною доставкою.

Kabanchik.ua – онлайн сервіс пошуку приватних фахівців для вирішення побутових та бізнес задач. Майданчик об'єднує замовників послуг, яким необхідно виконати будь-яку роботу, та компетентних фахівців, які шукають підробіток або додатковий заробіток.



КАБАНЧИК

Рисунок 1.2 – Логотип онлайн сервісу «Kabanchik»

Сервіс спеціалізується не на конкретній ніші, а займає більш роздрібну позицію. Наприклад, сервіс надає такі послуги:

- кур'єрська доставка (покупка, доставка, отримання товарів і документів);
- домашній майстер (всі роботи, пов'язані зі збірки та монтажу меблів, підключенням і ремонтом побутової техніки, сантехніки);
- логістика (перевезення вантажів, пасажирів, оренда складу, вивезення сміття та снігу);
- меблі (все, що пов'язано з виготовленням, частковим ремонтом та реставрацією);
- технічне обслуговування (прибирання приватної та комерційної нерухомості, виробничих об'єктів, прилеглих територій, видалення комах-паразитів, миття фасадів, вікон) та інше.

Rabotniki.ua – його не можна назвати повним аналогом, тому що він охоплює виключно тематику будівництва, ремонту та прибирання. Також модель монетизації та пошуку виконавця дещо відрізняється від Кабанчика. Кабанчик зайняв нішу дрібніших замовлень (ремонт розетки, монтаж люстри), тоді як Rabotniki.UA публікують більше великих приватних будівельних замовлень на великі обсяги робіт. У цілому всі замовлення відкриті для перегляду і все можна оцінити самому.



Рисунок 1.3 – Логотип сервісу «Rabotniki»

Фріланс-сервіс **Freelancehunt.com** – це сервіс для пошуку та взаємодії фрілансерів та замовників [16].

Слово «фріланс» походить від англійського freelance (вільний спис). Звичайно, вільних списоносцями-найманців зараз вже немає, і фрилансерами

називають людей, які вважають вільну форму організації трудового процесу та відносин привабливою і займаються роботою в той час, коли їм зручно, роблячи ту роботу, яка їм подобається [17].

Фрілансер (англ. freelancer) – вільний працівник. Термін фрілансер вперше вживається Вальтером Скоттом у романі «Айвенго» для опису «середньовічного найманного воїна». Будь-який добрий фахівець, який має прикладні знання і навички, може бути запрошений виконувати позаштатні завдання, тобто стати фрілансером. Серед фрілансерів зустрічаються люди різних професій – від фізиків, інженерів, технологів, журналістів, до копірайтерів, дизайнерів, програмістів. Мати саме IT-спеціалізацію необов'язково – можна виконувати нескладні види робіт: розміщувати оголошення, писати коментарі на форумах або заповнювати картки товарів у інтернет-магазинах [18].

Як для замовників, так і для виконавців на сервісі пропонуються такі послуги:

1. Сейф – безпечна угода, яка гарантує оплату виконавцям у разі якісного виконання замовлення та повернення коштів замовнику, якщо робота не буде виконана або виконана неякісно. Комісія послуги становить лише 9-10%, залежно від того, як вона буде розподілена. Учасники можуть узгодити між собою, хто оплачуватиме 9% - замовник чи виконавець. Також є можливість розділити комісію навпіл – по 5%. У разі спірних питань сторони можуть звернутися до арбітражу.
2. Персональний проект – замовники можуть запрошувати до роботи виконавця, що сподобався, обравши його в каталозі фрілансерів. Також цей сервіс можна використовувати для доплат, якщо робота виконавця дуже сподобалася або, наприклад, початковий обсяг та бюджет був неправильно розрахований.
3. Бізнес Сейф – безпечна угода між фрілансерами та юридичними особами з наданням документів, що закривають. Під час роботи через Бізнес Сейф ви можете платити (як замовник) або отримувати оплату (як виконавець) за

послуги за допомогою розрахункового рахунку. У разі спірних питань учасникам надається арбітраж.



Рисунок 1.4 – Логотип сервісу «Freelancehunt»

Висновок до розділу 1

У цьому розділі було описано постановку задачі бакалаврської роботи, проведений огляд існуючих сервісів за даною тематикою, а також були окреслені їх особливості. Наступний розділі буде присвячений вибору інструментів для розробки.

РОЗДІЛ 2

ОГЛЯД ІНСТРУМЕНТІВ ДЛЯ РОЗРОБКИ

Даний розділ присвячений опису використаного стеку технологій, які використовувалися під час проектування web-додатку.

2.1 Мова програмування C#

C# – об'єктно-орієнтована, орієнтована на компоненти мова програмування. З моменту створення мова C# збагатилася функціями для підтримки нових робочих навантажень та сучасними рекомендаціями щодо розробки ПЗ [3].

C# є сучасною та об'єктно-орієнтованою мовою програмування. Ця мова дозволяє розробникам створювати різні типи та захищених, надійних та безпечних у використанні та розробці програм, що використовують .NET. C# відноситься до найвідомішого сімейства мов - C. Ця мова буде добре знайома будь-кому, хто працював з C, C++, Java або JavaScript.



Рисунок 2.1 – Логотип мови програмування C#

Переваги C#

- Мова C# використовує об'єктно-орієнтований підхід до програмування у всьому, потрібно буде описувати абстрактні конструкції предметної

області, а потім реалізовувати взаємодію між цими конструкціями. Такий підхід користується широкою популярністю тому, що дозволяє не запам'ятовувати всю інформацію, а працювати за принципом чорної скриньки: **подав вхідні дані => МАГІЯ => PROFIT.**

- У мові присутня велика кількість синтаксичного цукру, який робить важке життя програміста трохи солодшим. Замість того, щоб писати велику кількість рядків коду, можна використовувати готову конструкцію, а компілятор зробить всю «брудну» роботу.
- Платформа NET Framework [4]. Написаний програмістом код мовою C#, транслюється в проміжну мову (IL), який у свою чергу вже перетворюється на машинний код на комп'ютері прямо під час виконання програми (JIT) [26]. В цьому випадку можна двом програмістам писати програмний код одного й того самого проекту різними мовами і жодному з них не потрібно буде перевчатись.
- Легкість використання. Завантажується потрібне рішення із nuget і можна його використовувати. Здебільшого вони безкоштовні. Сюди ж можна віднести велику кількість навчального та довідкового матеріалу.
- Строга типізація, яка дозволяє встановити захист від дурня, і кросплатформенність у .NET Core [27].

Недоліки C#

- C# дуже легко дизасемблюється. Це означає, що з великою ймовірністю код буде отриманий і вивчений конкурентами. Звичайно, є спеціальні інструменти, які можуть ускладнити цей процес, але на 100% захиститися від цього неможливо.
- .NET
- C# не є поширеною мовою. Більшість програмістів зосереджено в комерційній Enterprise сфері, що накладає вкрай серйозні обмеження на пошук роботи в невеликих містах

2.2 .NET-платформа

.NET – це платформа від Microsoft, яка дозволяє створювати програмні програми [6].

Прийнято вважати, що .NET Framework створили саме, як альтернативу Java (платформа від компанії Sun). Найголовніша відмінність у тому, що .NET розрахований працювати саме з операційними системами сімейства Microsoft Windows, офіційно розрахований. З того часу він пройшов нелегкий та довгий шлях від 1.0 до 4.8 (18 квітня 2019) версії. На сьогодні, незважаючи на появу платформи нового покоління (.NET Core), це як і раніше, досить популярна річ. Існує безліч програмних продуктів, бібліотек та фреймворків, що написані під .NET Framework і розвиваються далі.

Модульна платформа .NET Core – платформа, що випущена у 2016 році, що сумісна з різними операційними системами (кросплатформова). Кросплатформенність .NET Core відкрила вкрай велику кількість нових сценаріїв та можливостей для застосування. Саме така новинка відіграла велику роль у просуванні .NET серед розробників та представників бізнесу.

Більшість вважають, що мова C# і платформа .NET це одне і те ж, проте ні. Так, вони безсумнівно розвиваються з оглядом один на одного, але не мають суворої взаємозалежності, як такої. Наприклад, окрім офіційно підтримуваних реалізацій .NET, існують альтернативні варіанти, такі як Mono, .NET Compact Framework, .NET Micro Framework та інші. На всіх цих платформах можна використовувати мову програмування C#, але лише певною мірою. З іншого боку, з .NET сумісний не тільки C#, а й інші мови: F#, VB.NET і навіть C++.

2.3 База даних SQL

SQL – (SQL, Structured Query Language) декларативна мова програмування для взаємодії користувача з базами даних, що застосовується для формування запитів, оновлення і керування реляційними БД, створення схеми бази даних та її модифікації, системи контролю за доступом до бази даних [7].

Якщо говорити простими словами, то SQL – мова структурованих запитів (мова програмування), яка використовується як ефективний спосіб збереження даних, пошуку їх частин, оновлення даних їх видалення та вилучення з бази.

Microsoft SQL Server – система управління базами даних, яка розробляється корпорацією Microsoft. Як сервер даних виконує головну функцію по збереженню та наданню даних у відповідь на запити інших застосунків, які можуть виконуватися як на тому ж самому сервері, так і у мережі [24].

SQL – мова вкрай проста і легка у вивченні, у галузі вільного програмного забезпечення, сьогодні активно застосовується:

- розробниками баз даних (забезпечують функціональність додатків),
- тестувальниками (в ручному та автоматичному режимі),
- адміністраторами (виконують підтримку працездатності середовища).

Це є універсальна мова, що чітко має визначену структуру за встановленими стандартами. Взаємодія з базами даних відбувається швидко навіть у ситуаціях, коли обсяги цих даних великі. Окрім того, ефективне управління можливе навіть без особливих знань коду.

2.4 Мова програмування JavaScript

Це найбільш використовувана мова програмування. У результаті проведеного на «Stack Overflow» у 2021 опитування 69,7% з 47 184 опитаних професійних розробників віддали перевагу JavaScript [9].

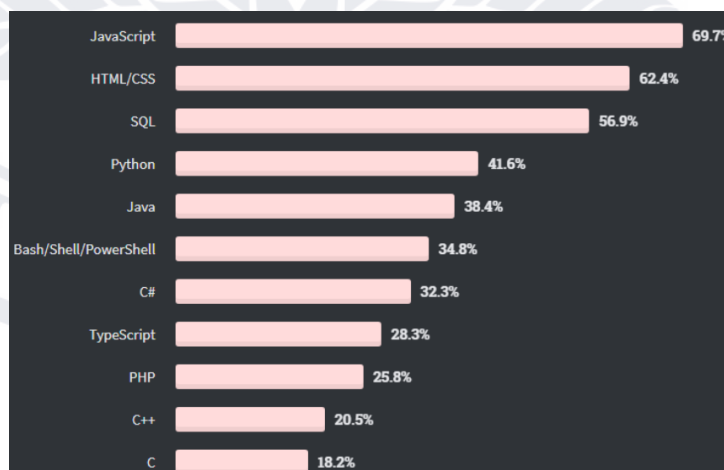


Рисунок 2.2 – Результат опитування

Зрозуміло, що JavaScript не є ідеальним. І все ж таки не можна не відзначити його багату екосистему з великою кількістю фреймворків, бібліотек та інших корисних інструментів, а також величезне співтовариство розробників JS.

TypeScript часто називають покращеною версією JavaScript, і на те є всі підстави. TypeScript використовує всі сильні сторони JavaScript (адже він компілюється в JS) і поєднує їх із власними потужними функціональними можливостями, такими як статична типізація, підтримка модулів та інтерфейсів тощо. TypeScript – це строго типізована мова програмування, що базується на JavaScript, що надає вам найкращі інструменти в будь-якому масштабі [25].

Згідно з останнім звітом від GitHub, JavaScript зараз теж лідирує за популярністю, але при цьому звіт свідчить про швидке зростання затребуваності TypeScript в останні роки.



Рисунок 2.3 – Звіт із сайту «GitHub»

JavaScript – мова програмування, що дозволяє реалізувати ряд складних рішень у web-документах. Вона допомагає зробити сторінки сайту більш інтерактивними [28].

Як це було б не дивно, проте JavaScript є об'єктно-орієнтованою мовою (клієнтською), яка підтримується додатками, що працюють з дизайном сайту. JavaScript став ще більш популярним серед розробників, коли з'явилася технологія AJAX, цей етап став новим у розробці сайтів.

Ось лише кілька окремих прикладів застосування технології:

- відображення контенту, який періодично оновлюється (інтерактивні карти);
- створення якісної анімації і графічних об'єктів у форматі 2D/3D;
- опція прокрутки відеозапису в медіа програвачі.



Рисунок 2.4 – Логотип мови програмування JavaScript

Знання основ HTML і CSS, дозволить користувачу швидше розібратись із роботою мови програмування JavaScript.

AJAX – це Асинхронний JavaScript та XML. Це набір методів веб-розробки, які дозволяють веб-застосункам працювати асинхронно – обробляти будь-які запити до сервера у фоновому режимі.

Принцип роботи технології AJAX [8]. Дана технологія не є мовою програмування. Як уже говорилося раніше, AJAX – це набір методів веб-розробки. Система зазвичай складається з:

- HTML/XHTML для основної мови та CSS для презентації.
- Об'єктна модель документа (DOM) для динамічного відображення даних та їхньої взаємодії.
- XML для обміну даними та XSLT для їхнього керування. Багато розробників почали замінювати JSON, тому що він ближче формою до JavaScript.
- Об'єкт XMLHttpRequest для асинхронної взаємодії.

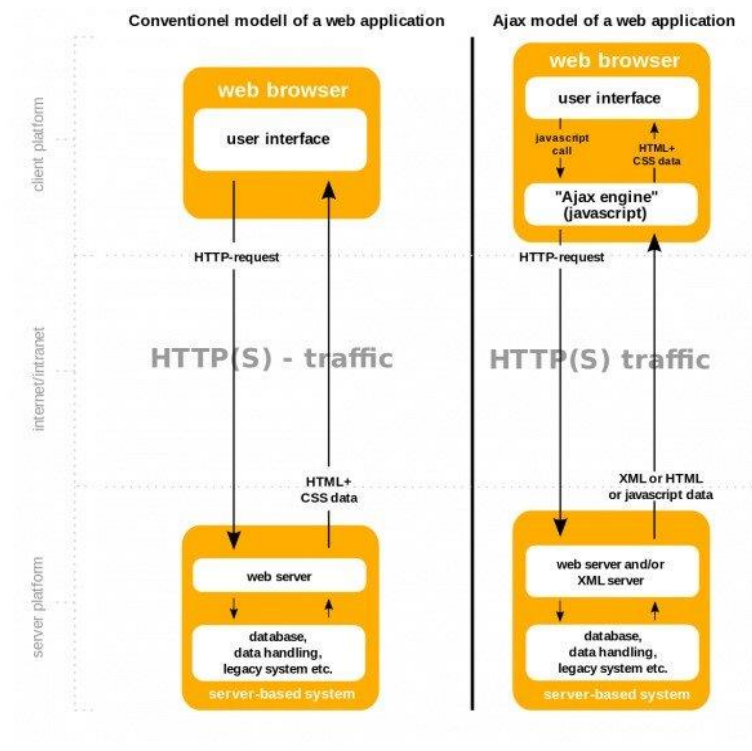


Рисунок 2.5 – Діаграма

Таблиця 2.1 – Порівняння моделей

Звичайна модель	AJAX - модель
HTTP-запит надсилається з веб-браузера на сервер.	Браузер створює виклик JavaScript, який активує XMLHttpRequest.
Сервер отримує та згодом витягує дані.	У фоновому режимі веб-браузер створює запит HTTP до сервера.
Сервер надсилає запитані дані до веб-браузера.	Сервер отримує, витягує та відправляє дані назад у веб-браузер.
Веб-браузер отримує дані та перезавантажує сторінку для відображення даних.	Веб-браузер отримує запитані дані, які безпосередньо відображаються на сторінці. Перезавантаження не потрібне.

Переваги JavaScript

- Швидкість. JavaScript часто запускається відразу в браузері клієнта.
- Простота - синтаксис JavaScript був натхненний мовою Java. Вважають, що його легко вивчити порівняно з іншими популярними мовами, наприклад C++.
- Сумісність. JavaScript можна вставити на будь-яку веб-сторінку, на відміну від PHP. Також JS можна використовувати в багатьох програмах різного виду, завдяки підтримці інших мов, таких як Perl та PHP.
- Навантаження сервера. JavaScript працює на стороні клієнта, тому загалом це зменшує навантаження на сервери, а простим програмам сервер у цілому може взагалі не знадобитися.
- Розширені інтерфейси. JS можна використовувати для створення функцій перетягування чи інших компонентів, наприклад повзунків. Це все значно покращує користувацький інтерфейс та роботу на сайті.
- Універсальність. Є багато способів використовувати JavaScript через сервери Node.js.
- Оновлення. За появи ECMAScript 5 (специфікація сценаріїв, на яку покладається JavaScript), ECMA International займається щорічним оновленням JS.

Недоліки JavaScript

- Захист на стороні клієнта. Оскільки виконання коду JavaScript зазвичай виконується на стороні клієнта, помилки, неточності та недогляд іноді можуть бути використані для злих цілей. Через це деякі люди вирішують повністю вимикати JS.
- Підтримка браузера. Хоч і сценарії на стороні сервера завжди дають однакові результати, різні браузери іноді інтерпретують код JavaScript по-різному. На сьогодні відмінності є мінімальними, і не слід перейматись.

2.5 Додаткові інструменти для front end

Розробники вже звикли бачити серед провідних JavaScript-фреймворків React (бібліотека), Angular та Vue.js.

Angular. Веб-фреймворк, що дозволяє JavaScript інтегруватися з HTML та CSS. За його допомогою збудовано понад 400 тисяч сайтів по всьому світу. Із ним можна розробляти нативні та веб-програми для ПК та мобільних пристроїв. Також підходить для корпоративного програмного забезпечення. Його використовують Google, Microsoft та YouTube.

Декілька плюсів:

- допомагає створювати прогресивні програми (PWA);
- зручно маніпулювати DOM-елементами;
- підтримка Google та потужна екосистема.

Backbone.js. Часто використовують для розробки SPA та клієнтського софту. У бібліотеку входять понад 100 розширень – зручно виконувати складні функції з меншою кількістю коду. Вона дозволяє розробникам вибирати інструменти для проекту, виходячи з потреб і вимог. Backbone.js використовують Airbnb, Drupal, Trello.

Декілька плюсів:

- чистий та легкий в обслуговуванні код;
- автоматичне оновлення HTML-коду;
- гнучкість і великий простір прийняття рішень.

Ember.js. JavaScript-фреймворк, який використовується для розробки масштабованих односторінкових веб-програм. Спочатку був відомий, як SproutCore 2.0 framework. Він поставляється з механізмом рендерингу Glimmer, з яким можна рендерувати DOM-елементи. Інтерфейс командного рядка Ember.js дозволяє виконувати тести всередині браузера, автоматичне перезавантаження компонентів, швидку перебудову тощо. Ним користуються Apple, Microsoft, LinkedIn, Netflix.

Декілька плюсів:

- висока продуктивність;
- адаптивність та гнучкість;
- дозволяє писати розширені HTML-теги.

Express.js. Фонова платформа веб-дизайну для Node.js. Використовується для створення веб-застосунків та API. Фреймворк досить швидкий і надає компоненти маршрутизації, підтримує проміжне програмне забезпечення, шаблони та інші функції, які роблять зручніше розробку. А ще він сумісний із базами даних, наприклад MongoDB та MySQL. Його використовують PayPal, Uber, IBM.

Декілька плюсів:

- швидкість;
- просте налагодження;
- дозволяє здійснювати динамічну візуалізацію HTML-сторінок.

Next.js. Дозволяє розробляти Jamstack та серверні програми. Поряд з іншими технологіями він підтримує CSS та стилізований JSX. Крім того, у ньому можна динамічно імпортувати модулі JavaScript та компоненти React. А також експортувати повністю статичний сайт із програми. Його використовують Netflix, Github, Avocode.

Декілька плюсів:

- автоматичний поділ коду та маршрутизація;
- підтримка SEO;
- серверний рендеринг;
- динамічні компоненти.

React. JavaScript-бібліотека з відкритим вихідним кодом для створення інтерфейсів користувача. Вона дозволяє створювати компоненти інтерфейсу для мобільного та ПК-софту. Ще його рекомендують використовувати для розробки SPA та корпоративних додатків. На GitHub React – другий за популярністю фреймворк. Його використовують Facebook, Instagram, WhatsApp.

Декілька плюсів:

- легке об'єднання JavaScript та HTML;
- просто розробляти динамічні веб-програми;
- проста відладка;
- підтримка спільноти.

Vue.js. Прогресивний фреймворк, який можна інтегрувати з уже готовими проектами та бібліотеками JS. У 2020 році Vue став найзірковішим фреймворком на GitHub, обійшовши Angular та React. До того ж, за останні 3-4 роки інтерес до фреймворку зріс на 18-20%. Його використовують Stack Overflow, GitLab, Adobe.

Декілька плюсів:

- високий ступінь настрюваності;
- легко вчити;
- підтримка CSS переходів та анімації;
- гнучкість та модульність.

Для розробки даного проекту був обраний саме React. Ця бібліотека є лише рівнем представлення.



Рисунок 2.6 – Лого бібліотеки React

React бібліотеку часто ставлять в один ряд з іншими JavaScript фреймворками, але суперечки «React vs Angular» не мають сенсу, тому що їх не можна порівнювати.

Angular – повноцінний фреймворк (що включає також рівень представлення), а React – ні. React має можливість дати шаблони та деякі callback-функції для відображення HTML коду. Результатом роботи React є

HTML. Зв'язок HTML/JavaScript, називають компонентами. Воони займаються тим, що зберігають свій внутрішній стан у пам'яті (наприклад: обрана користувачем закладка в браузері), але в результаті випльовується лише HTML.

Головні плюси:

1. Завжди можна сказати, як компонент буде відображено, дивлячись на вихідний код. Хоч це нічим не відрізняється від Angular шаблонів, проте можна вважати хорошою перевагою. Коли розробляється складна програма це дуже важливо, а особливо коли продукт розробляє команда.
2. Зв'язування JavaScript та HTML у JSX, робить компоненти простими для розуміння. Оце поєднання HTML/JavaScript може збентежити. Зазвичай навчають так, щоб не вставляти JavaScript у DOM (наприклад: обробники OnClick).
3. Можна рендерити React на сервері. Якщо розробник створює публічний сайт або програму, і рендерить усе на клієнті, то він обрав неправильний шлях.

Мінуси:

1. Не буде отримано: систему подій (відмінну від нативних DOM подій), роботу з AJAX, будь-який шар даних; Promises, фреймворк попри всі випадки життя.
2. Погана та незрозуміла документація (так каже статистика).
3. React є досить великим, враховуючи те, як мало від нього отримують, включаючи погану кросбраузерну підтримку.

Для візуалізації було обрано нестандартну гіпертекстову розмітку та каскадну таблицю стилів, а спеціальну бібліотеку «MaterialUI». MUI пропонує повний набір інструментів інтерфейсу користувача, які допоможуть швидше передавати нові функції [11].



Рисунок 2.7 – Лого бібліотеки «MaterialUI»

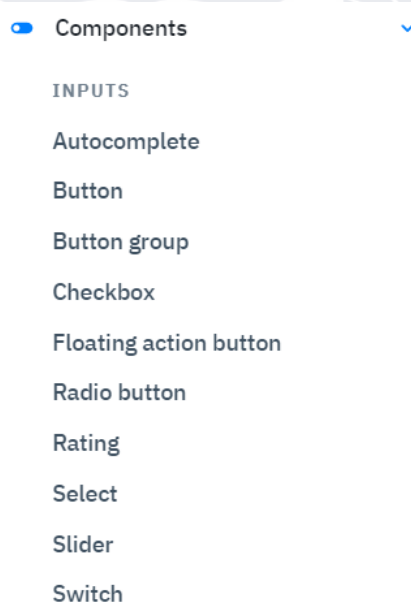


Рисунок 2.8 – Декілька варіантів компонент із бібліотеки

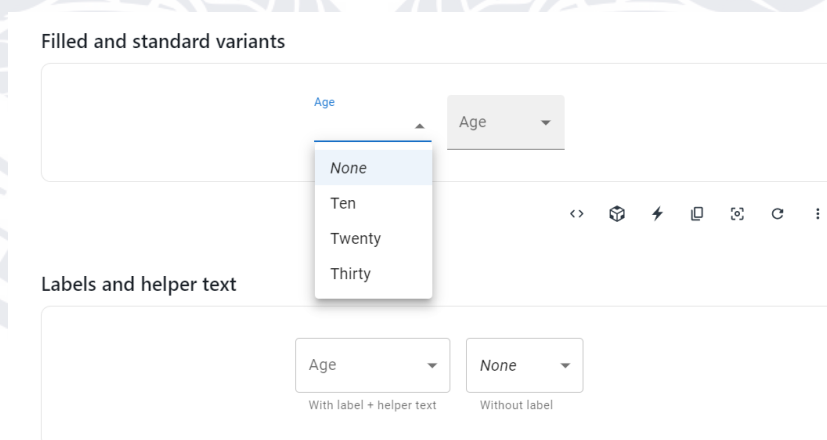


Рисунок 2.9 – Приклад компоненти «select»

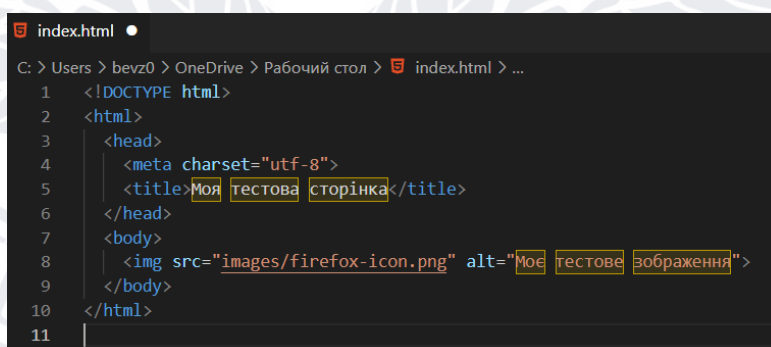
HTML – стандартизована мова розмітки документів для перегляду веб-сторінок у браузері. Веб-браузери отримують HTML документ від сервера за протоколами HTTP/HTTPS або відкривають з локального диска, далі інтерпретують код в інтерфейс, який відображатиметься на екрані монітора [19].

Це код, який використовують для відображення та структурування веб-сторінки з її контентом. Прикладом цього може слугувати контент, що може бути структурований всередині різних тегів: параграф, маркований список або з використанням зображень і таблиць даних.



Рисунок 2.10 – Логотип гіпертекстової розмітки (HTML)

HTML – це мова розмітки, і використовується, щоб повідомити браузер, як саме потрібно відображати веб-сторінки. HTML складається з ряду елементів, які використовують для того, щоб вкладати або обертати різні частини для відображення контенту або його дії певним чином.

A screenshot of a code editor showing the structure of an HTML document. The file is named "index.html". The code is as follows:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="utf-8">
5     <title>Моя тестова сторінка</title>
6   </head>
7   <body>
8     
9   </body>
10 </html>
11
```

Рисунок 2.11 – Структура HTML документу

CSS. Щоб взнати значення, знову ж таки потрібно посилатись на вікіпедію. Отже, це спеціальна мова стилю сторінок, що використовується для опису їхнього зовнішнього вигляду [20].



Рисунок 2.12 – Логотип мови стилів сторінок CSS

Це так названий код, який розробники використовують для стилізації веб-сторінок. Основи CSS допоможуть зрозуміти, як зробити текст червоним або можливо червоним кольором, як зробити так, щоб контент з'являвся чи розташовувався на певному місці екрану, як прикрасити веб-сторінку за допомогою фонових зображень та кольорів (CSS є мовою таблиці стилів). Наприклад, щоб вибрати всі елементи абзацу на сторінці HTML і змінити текст всередині них з чорного на червоний, потрібно написати такий CSS код:

```
p {  
  color: red;  
}
```

Рисунок 2.13 – Приклад коду стилів

CSS препроцесор. Це програма, що підпорядковується власному синтаксису, але з нього може згенерувати CSS код [29]. Існує безліч препроцесорів і більшість із них розширює можливості чистого CSS, додаючи такі опції як: домішки, вкладені правила, селектори успадкування та інших. Ці особливості полегшують роботу з CSS: спрощують читання коду та його подальшу підтримку.

Sass – найстабільніша і найпотужніша мова розширень CSS професійного рівня у світі [21].



Рисунок 2.14 – Лого препроцесору Sass (scss)

2.7 Програмні інструменти (середовище розробки)

«WebStorm» – інтегроване середовище розробки для JavaScript, HTML та CSS від компанії JetBrains [12]. Вважають, що найбільшою перевагою інтегрованого середовища для розробки є робота з проектами (у тому числі, рефакторинг JS коду, що міститься в різних теках і файлах проекту).

«Visual Studio Code» – засіб для створення, редагування та налагодження сучасних веб-застосунків і програм для хмарних систем [13]. Цей продукт був представлений компанією Microsoft у квітні 2015. Головною перевагою є те, що це безкоштовний продукт



Рисунок 2.15 – Логотип програмного продукту

Особливості «Visual Studio Code»

- «VS Code» дозволяє розробляти консольні додатки, а також додатки з графічним інтерфейсом, веб-сайти та веб-додатки.

- У редакторі розміщуються інструменти для роботи з Git відразу як то кажуть з коробки, а також засоби для рефакторинга, навігації по коду, автодоповнення типових конструкцій і контекстної підказки.
- «VS Code» підтримує розробку для платформ ASP.NET і Node.js.
- Вагомою перевагою редактора є підтримка досить великої кількості мов, таких як C++, C#, Python, PHP, JavaScript та ін.

Переваги «Visual Studio Code»

- Уся програма та інтерфейс має безліч налаштувань.
- Присутня бібліотека доповнень і готових рішень (розширювана).
- Мультифункціональність. Редактор підтримує майже всі мови програмування, які використовуються для створення додатків;
- Гнучкість та простота.

Microsoft Visual Studio – лінійка продуктів компанії Microsoft, що включає інтегровану середу розробки програмного забезпечення та ряд інших інструментів [14]. Це дійсно хороша IDE, до того ж є повнофункціональна безкоштовна версія Community.



Рисунок 2.16 – Логотип програмного продукту

Особливості «Visual Studio»

- «VS» має можливість для розробки консольних додатків, додатків із графічним інтерфейсом, а також веб-сайтів та веб-додатків.

- У редакторі розміщуються інструменти для роботи з Git відразу як то кажуть з коробки, а також засоби для рефакторинга, навігації по коду, автодоповнення типових конструкцій і контекстної підказки.
- «VS» підтримує розробку на платформі ASP.NET, і вважається досить хорошим рішенням.

Переваги «Visual Studio»

- Вбудований web-сервер. ASP.NET потребує web-сервер, який чекатиме на web-запити й обробляти відповідні сторінки для обслуговування web-додатка.
- Для створення більшості програм потрібна досить велика кількість стандартного стереотипного коду, ASP.NET тому не виняток.
- Висока швидкість розробки. Багато можливостей Visual Studio спрямовані на те, щоб допомагати розробнику робити свою роботу вкрай швидко.
- Запропоновані Visual Studio інструменти налагодження є найкращим засобом для відстеження дивних і загадкових помилок та діагностування незрозумілої поведінки.

Висновок до розділу 2

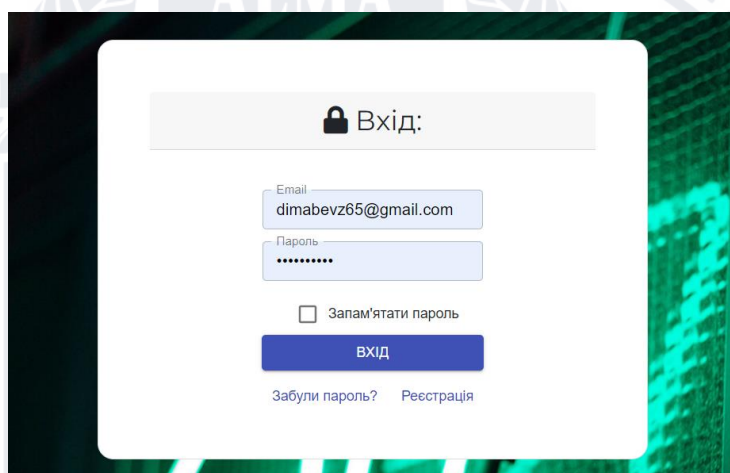
У даному розділі був описаний стек технологій, які використовувались для розробки web-додатку. Наступний розділ буде присвячений опису розробленого продукту.

РОЗДІЛ 3

РОЗРОБКА WEB – ДОДАТКУ

Даний розділ присвячений опису розробки веб-додатку для замовлення та надання бізнес-задач.

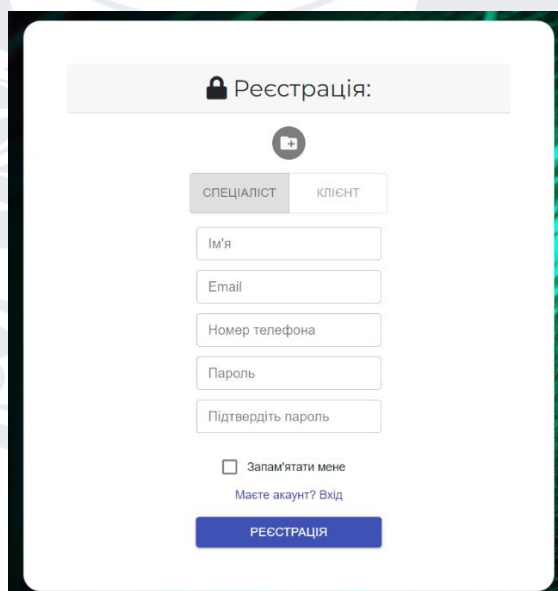
3.1 Процес авторизації / реєстрації.



The screenshot shows a login form titled "Вхід:" (Login). It contains two input fields: "Email" with the value "dimabevz65@gmail.com" and "Пароль" (Password) with masked characters. Below the password field is a checkbox labeled "Запам'ятати пароль" (Remember password). At the bottom, there is a blue button labeled "ВХІД" (Login), and two links: "Забули пароль?" (Forgot password?) and "Реєстрація" (Registration).

Рисунок 3.1 – Форма авторизації

На рис. 3.1 можна побачити, що при переході на сторінку входу, з'являється вікно авторизації зі стандартними полями. Після виконання потрібного алгоритму дій можна увійти на сайт.



The screenshot shows a registration form titled "Реєстрація:" (Registration). It features a toggle switch between "СПЕЦІАЛІСТ" (Specialist) and "КЛІЄНТ" (Client). Below are five input fields: "Ім'я" (Name), "Email", "Номер телефона" (Phone number), "Пароль" (Password), and "Підтвердіть пароль" (Confirm password). At the bottom, there is a checkbox labeled "Запам'ятати мене" (Remember me), a link "Маєте акаунт? Вхід" (Have an account? Login), and a blue button labeled "РЕЄСТРАЦІЯ" (Registration).

Рисунок 3.2 – Форма реєстрації

На рис. 3.2 можна побачити, що при переході на сторінку реєстрації, з'являється вікно з формою. Після виконання потрібного алгоритму дій можна провести реєстрацію на сайті. Користувачу дається можливість обрати роль: спеціаліст чи клієнт. У залежності від обраної ним ролі буде надаватись певний функціонал з яким можна працювати.

3.2 Основний функціонал

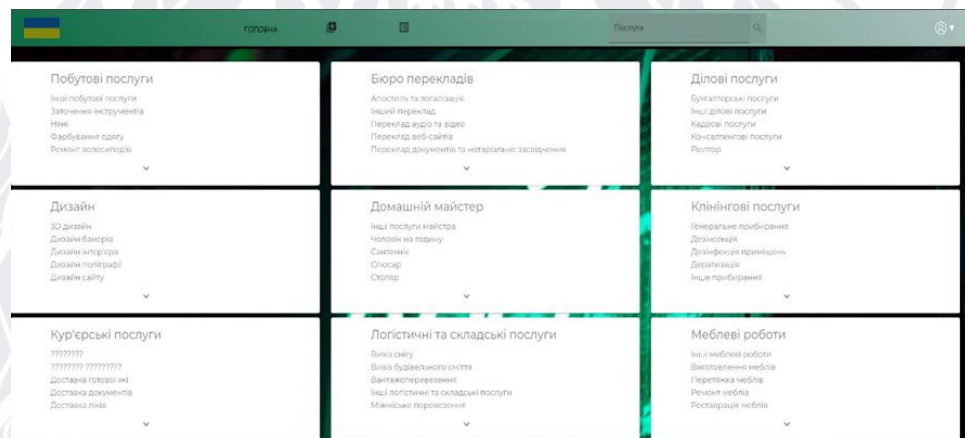


Рисунок 3.3 – Головна сторінка

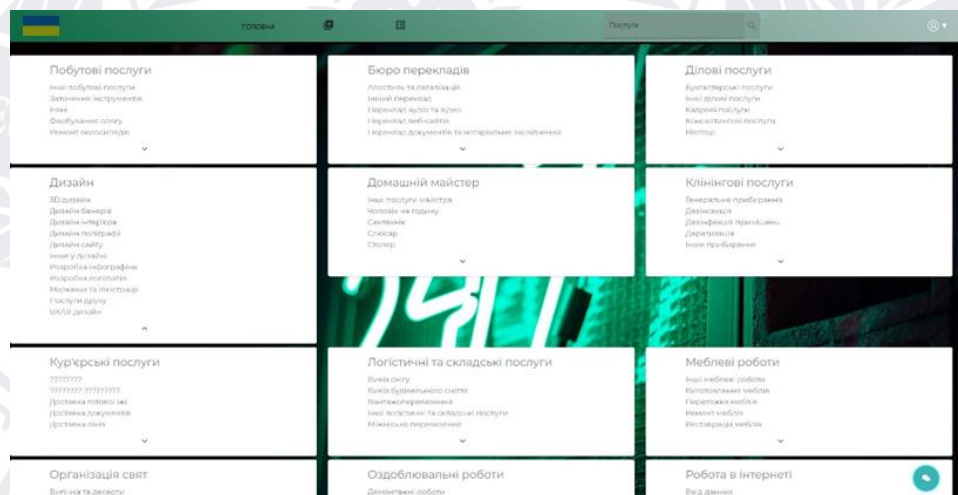


Рисунок 3.4 – Категорії

Після виконання входу, користувача перенесе на головну сторінку, рис.3.3. На рис. 3.4 можна побачити категорії з підкатегоріями, які є на сайті для

інформування користувача. Цим показано, що види послуг надаються в залежності від даних категорій / підкатегорій.

У хедері можна побачити, що є навігація для зручного користування роутами.

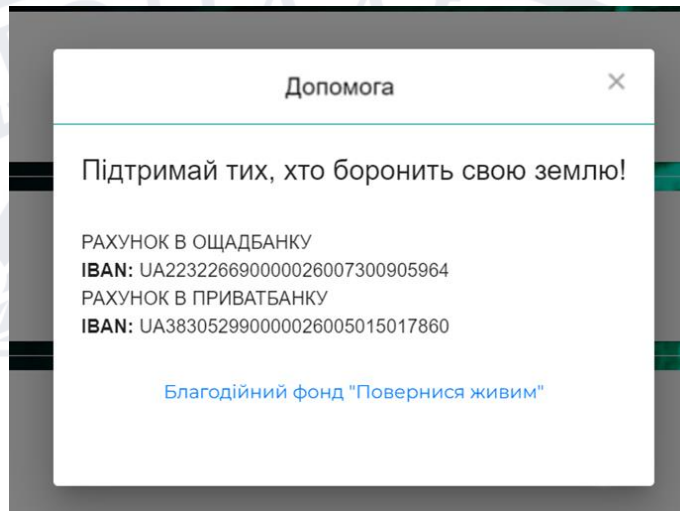


Рисунок 3.5 – Вікно для підтримки

При натисканні на кнопку, що має вигляд прапора України (рис.3.5), користувач може побачити модальне вікно, де є інформація з рахунками для підтримки української армії та лінк, що веде на сайт фонду «Повернись живим».

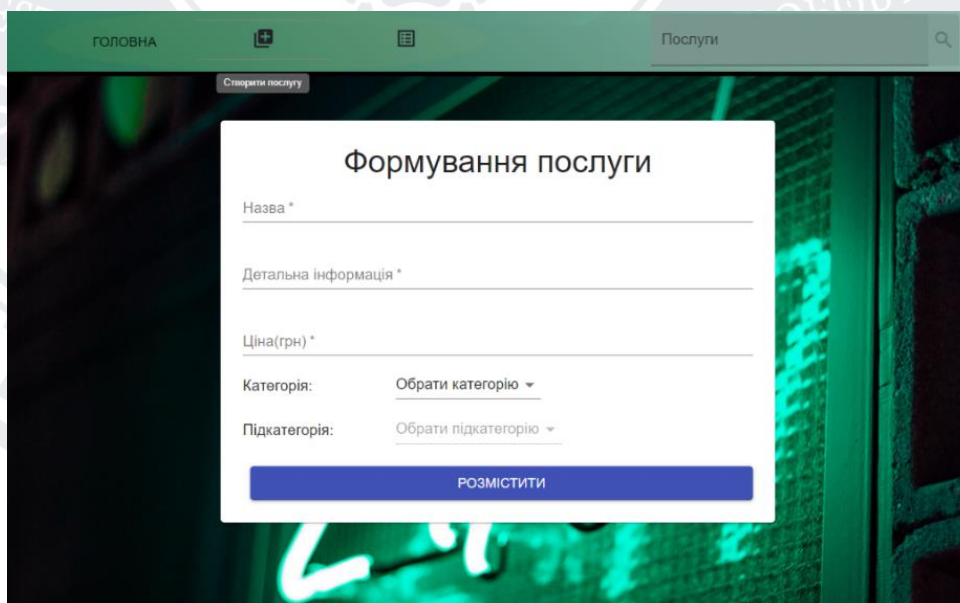


Рис. 3.6 – Сторінка для формування послуги

На рис. 3.4 можна побачити сторінку для формування послуги, де є селекти з вибором категорії та підкатегорії. Якщо одне з полів для вводу буде пустим, то запит на розміщення замовлення не буде відправлено тому, що прописано валідацію полів за допомогою бібліотеки «react-material-ui-form-validator».

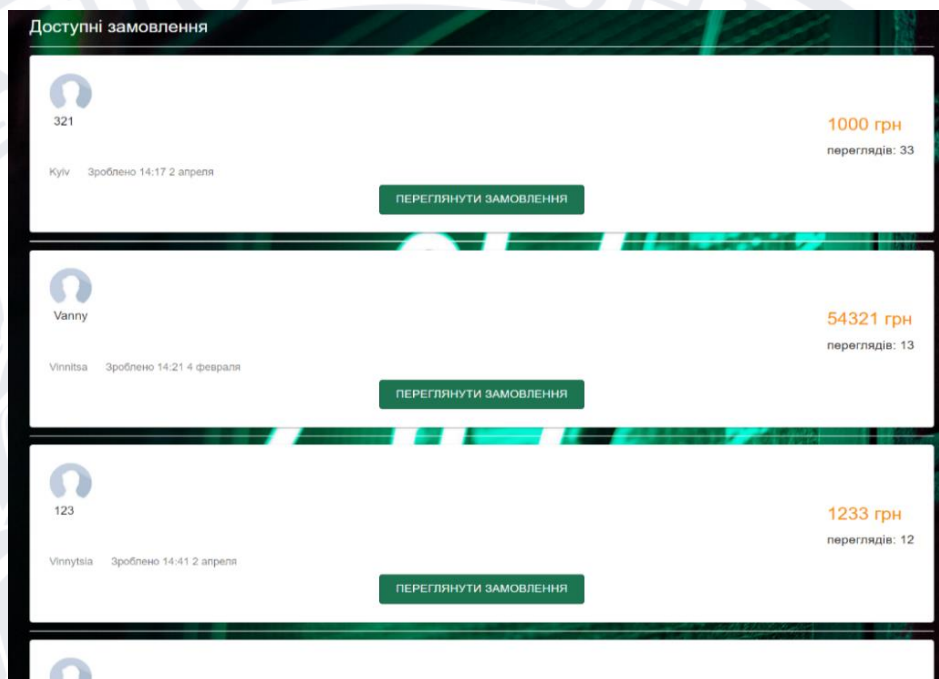


Рис. 3.7 – Сторінка доступних замовлень

На рис. 3.5 зображено сторінку доступних замовлень, будь-хто з користувачів може зайти на неї, проте функціонал буде відрізнятись. Різницю можна побачити на рисунках 3.8 та 3.9, де відрізняється кнопка. Для спеціаліста зображено кнопку з написом «прийняти замовлення», що означає він береться за виконання даного замовлення. А для клієнта зображено кнопку з написом «створити схоже замовлення», після натискання на кнопку його переведе на сторінку зі створенням схожого замовлення.

Рис. 3.8 демонструє, що спеціаліст може взяти замовлення та виконати його. Коли спеціаліст бере замовлення до виконання, при натиску на кнопку відбувається PUT запит, де записується ід спеціаліста в базу замовлення. Тобто до замовлення додається ід спеціаліста, щоб потім була можливість вивести усі послуги (що вже виконані чи перебувають у роботі) у профілі спеціаліста.

Також це реалізовано для того, щоб реалізувати в майбутньому сповіщення у дзвіночок для сповіщень. Виходить так, що коли спеціаліст підтверджує замовлення, то клієнту приходить сповіщення до дзвіночка та сповіщає його про те, що його замовлення взяв один із спеціалістів або коли спеціаліст завершив роботу.

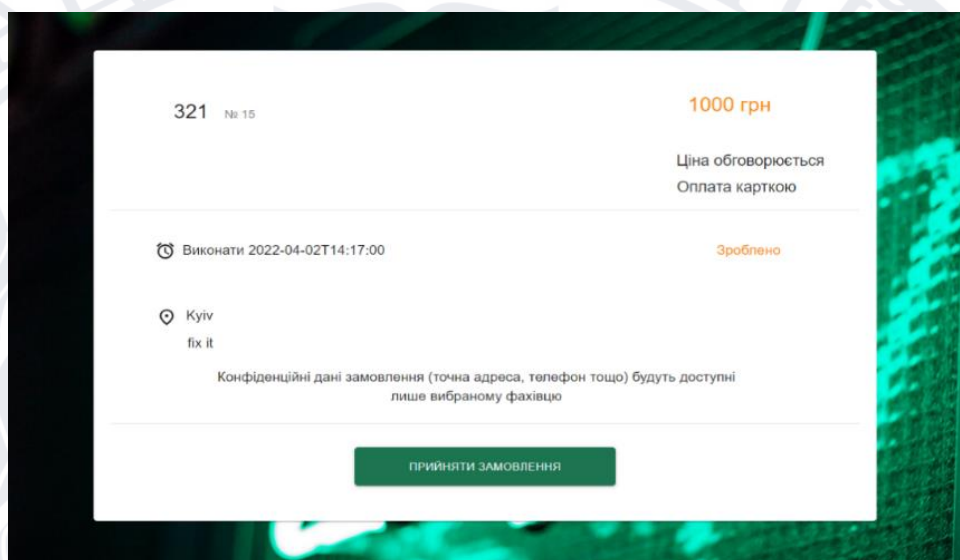


Рисунок 3.8 – Форма замовлення (вигляд зі сторони спеціаліста)

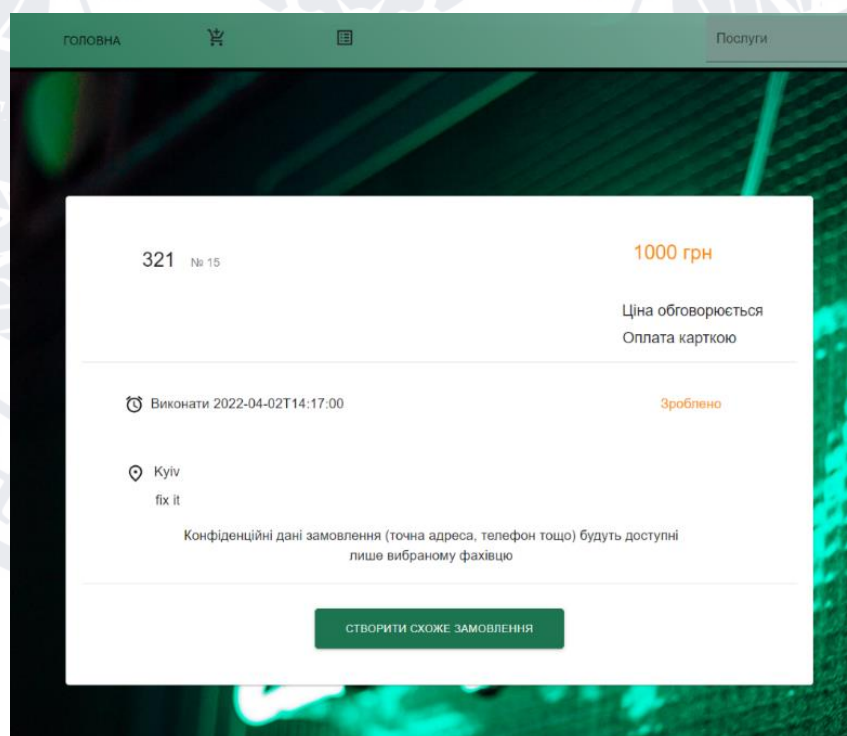


Рисунок 3.9 – Форма замовлення (вигляд зі сторони клієнта)



Рисунок 3.10 – Результат пошуку послуг

На даному рисунку 3.10 можна побачити сторінку з результатом пошуку послуг. У хедері користувач знайде поле вводу, де можна вписати назву категорії і autocomplete підбере потрібний результат, що можна використовувати.

Функціонал реалізований за допомогою JavaScript. Алгоритм зчитує дані з поля і починає фільтрувати підказки лише після вводу трьох літер у поле.

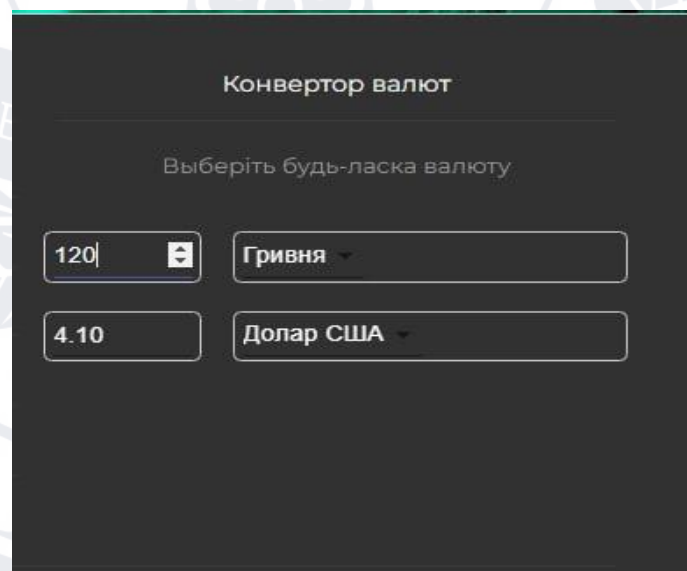


Рисунок 3.11 – Конвертор валют

На рис. 3.9 можна побачити реалізований конвертор валют для користувачів, щоб визначити результат обміну однієї валюти на іншу. Ціни на сайті можуть бути вказані не лише в українській національній валюті, саме тому виникла ідея реалізації конвертора валют. Він створений аби користувач міг не використовувати сторонні ресурси для конвертації різного роду валют у потрібну для нього.

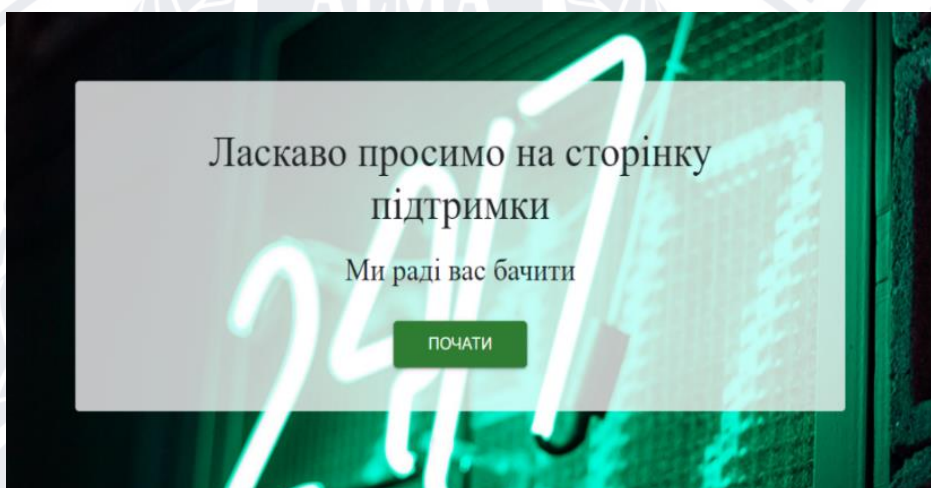


Рисунок 3.12 – Сторінка входу до чата (сторона адміністратора)

Рис. 3.10 демонструє головну вітальну сторінку, що з'являється у адміністратора сайту після натискання на кнопку «чат підтримки». Кнопка буде розташована у модальному вікні «налаштування акаунта» після його відкриття. Ніхто з інших користувачів не може скористатись цим режимом, окрім адміністратора. Після натискання на кнопку «закрити чат» зі сторони адміністратора, його перекине на цю сторінку, що тим самим дасть можливість знову спілкуватись із іншим користувачем.

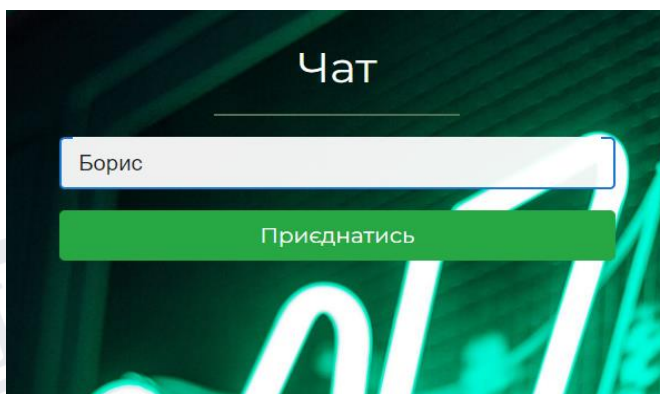


Рисунок 3.13 – Чат підтримки (сторона адміністратора)

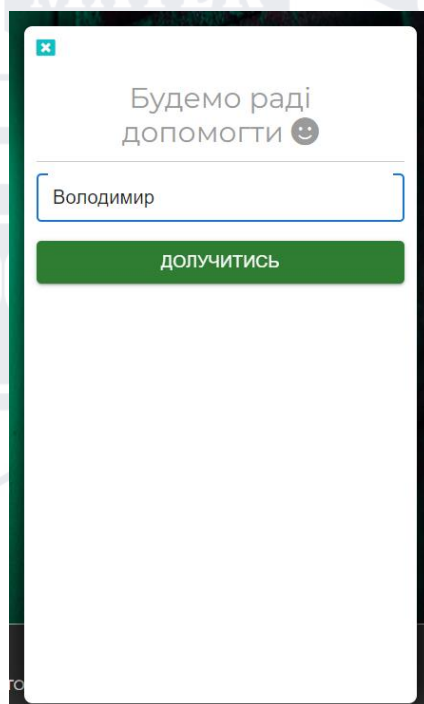


Рис. 3.13 – Чат підтримки (сторона користувача)

На рисунках 3.11 та 3.12 можна побачити чат підтримки, який може використати користувач. При натисканні на «бульбашку» відкривається чат зі сторони користувача рис. 3.12, де є можливість вписати своє ім'я для спілкування. Поки ім'я не буде вписано, кнопка «долучитись» не буде активована, так само і зі сторони адміністратора рис. 3.11.



Рисунок 3.14 – Список доступних кімнат для спілкування

На рис. 3.13 можна побачити список доступних кімнат (для адміністратора) з активними користувачами, що увійшли до чату. Адміністратор може обрати будь-яку кімнату та приєднатись до чату з користувачем рис. 3.14.

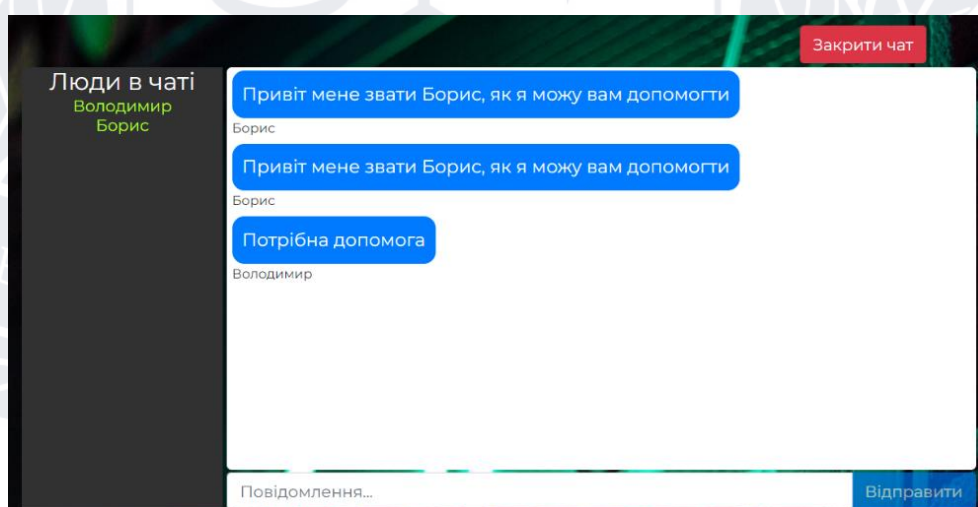


Рисунок 3.15 – Вигляд чату на стороні адміністратора

Реалізація чату відбувається так: є два хаба, один для клієнта і один для адміністратора, хаб з'єднується на фронтенд та бекенд частинах. На бекенді прописано методи, які потім посилаються на фронтенд (це такі методи, як приєднатись до кімнати, вийти з неї, відправити дані користувачів, написати повідомлення).

На фронтенд частині реалізовано інтерфейс, для клієнта та адміністратора. Сторона адміністратора: є форма в якій при зверненні до підтримки сайту, клієнт вписує своє ім'я та натискає «додатись», після чого він під'єднується до хаба за допомогою методу «JoinRoom», а далі під'єднується до кімнати, назва якої співпадає з його id.

Адміністратор робить те ж саме у своїй формі, після чого він долучається до кімнати без назви, завдяки чому він доєднується до хабу і отримує актуальну інформацію про всіх користувачів, що хочуть отримати підтримку. Є кнопка «рефреш», яка реконектить його до хабу. Також є кнопка, щоб вийти з хабу, яка відключає від хабу. Потім при виборі користувача адміністратором, він під'єднується до кімнати користувача, signalr формує між ними зв'язок по веб-сокету і вони можуть обмінюватись повідомленнями. При виході кожної із сторін, протилежна сторона інформується в чаті.

ASP.NET signalr - це бібліотека для ASP.NET розробників, яка спрощує процес додавання веб-функцій у режимі реального часу до програм. Ця бібліотека дає можливість у режимі реального часу миттєво відправити вміст для підключених клієнтах у міру доступності сервера, а не чекати, доки клієнт запросить нові дані [31].

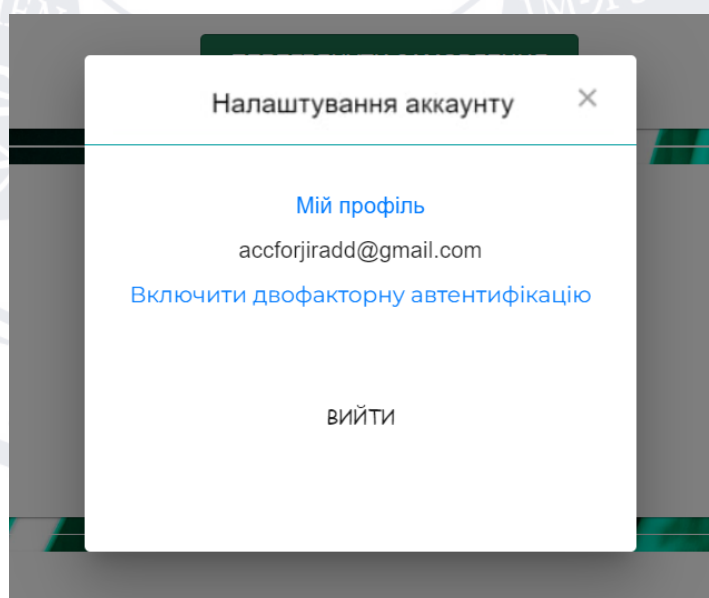


Рисунок 3.16 – Модальне вікно з користувача

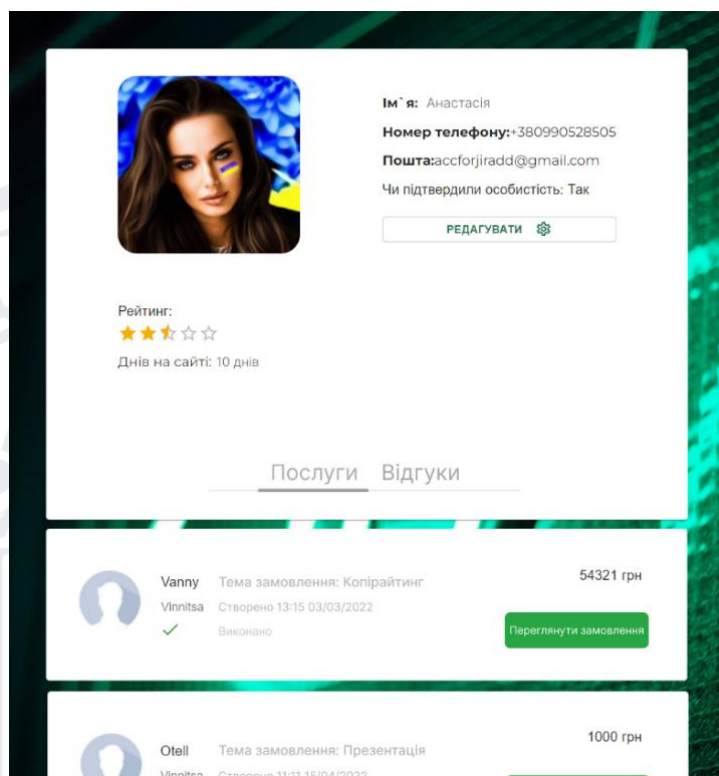


Рисунок 3.17 – Сторінка профілю спеціаліста

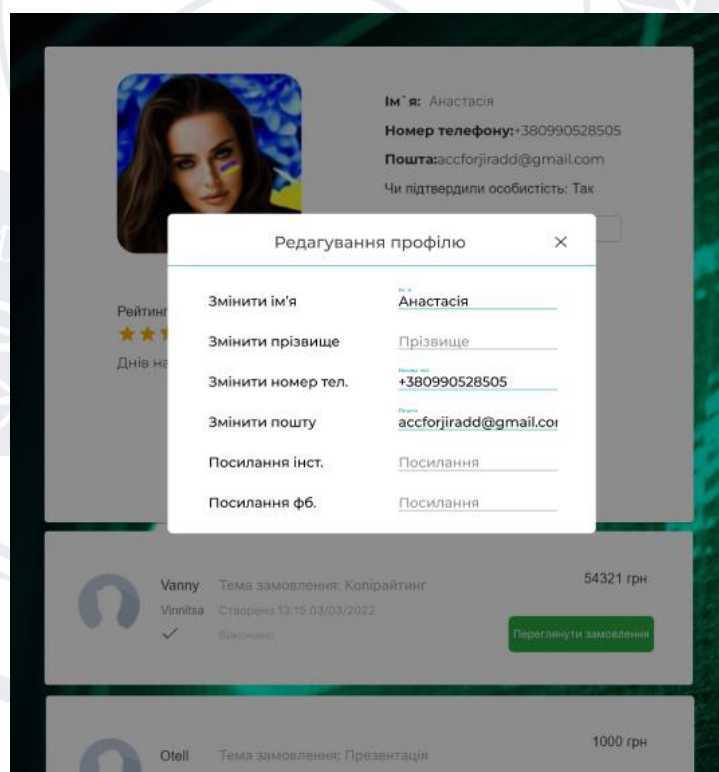


Рисунок 3.18 – Вікно редагування профілю

На рис. 3.15 зображено модальне вікно, за допомогою якого може перейти на сторінку профілю, що зображено на рис. 3.16. Є можливість редагування профілю, перегляду послуг, що виконані або перебувають у процесі роботи. Також клієнт може виставити рейтинг спеціалісту, лише обравши відповідну оцінку.

Рейтинг створений для того, щоб клієнт міг виставити від 0 до 5 балів спеціалісту. Рейтинг працює таким чином, що клієнт натискає на зірочку, далі з фронтенд частини відразу приймається 3 параметри: клік клієнта, кількість поставлених балів та сам рейтинг. Виходить так, що після кліку обробляються параметри і відправляються на бекенд, де через модельки та запит відправляються та зберігаються в SQL Server - базу даних. Через сторедж процедуру ділиться загальна кількість балів (від 0 до 5) на кількість кліків клієнтів, що дає можливість розрахувати правильно кількість рейтингу. Формула виглядає так: $\text{TotalPoin} / \text{TotalClick} = \text{Rating}$.

Далі за допомогою GET запита дані витягуються на фронтенд частині та сетаються у потрібні місця верстки.

Можна побачити дві вкладки, на одній з них є відгуки, що може залишити клієнт.

У клієнта схожа сторінка профілю, проте єдине, що відрізняється так це вкладки. Немає можливості лишити відгук на сторінці клієнта, а також виставити йому оцінку. На вкладці «послуги» у клієнта відображається список створених лише ним послуг.

При натисканні на кнопку «редагувати», користувач має можливість побачити функціонал, що реалізований відповідно у модальному вікні рис. 3.17. На даному рисунку можна побачити, що користувачу надається можливість змінити ім'я, номер телефону та пошту, а також є можливість додати прізвище та посилання на соціальні мережі.

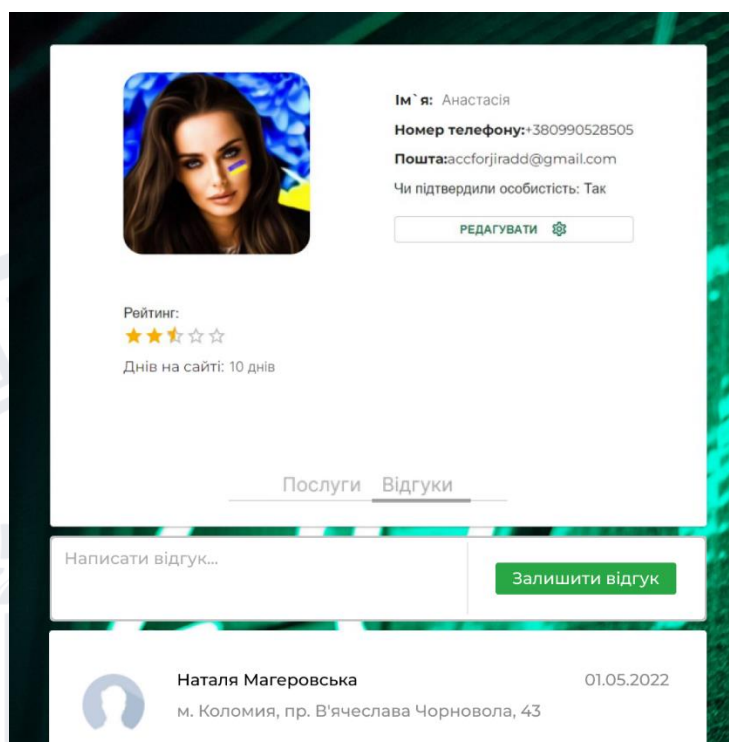


Рисунок 3.19 – Вкладка з відгуками (сторінка спеціаліста)

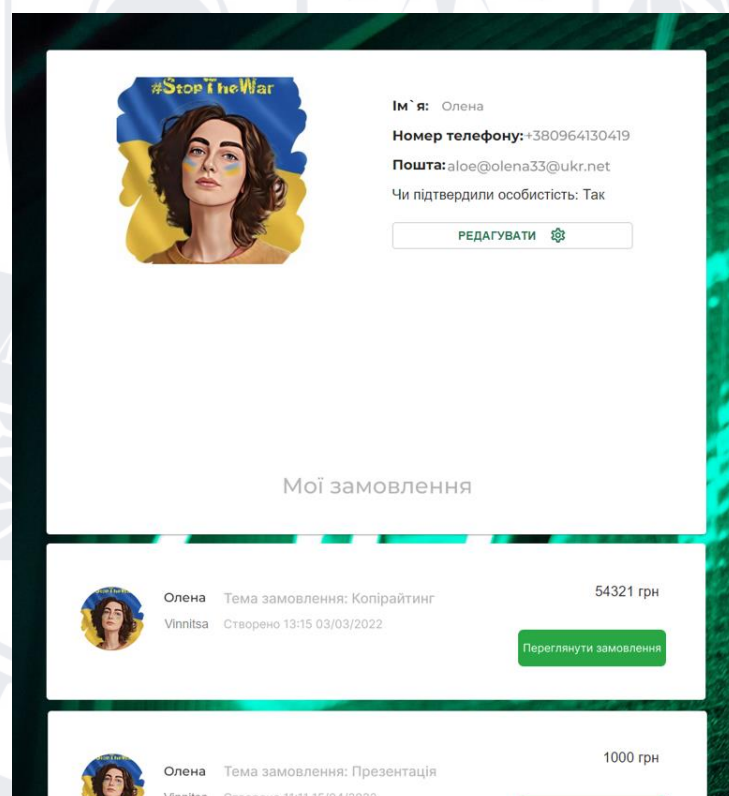


Рисунок 3.20 – Вікно редагування профілю

Рис. 3.20 демонструє сторінку профілю клієнта. Так як він не є спеціалістом чи адміністратором, то у нього немає вкладок з послугами / відгуками чи можливості перейти у чат підтримки від сторони адміністратора. Натомість у клієнта є можливість переглянути список замовлень, що він створив. Коли клієнт створює замовлення, воно автоматично додається у даний список. Надається така можливість за допомогою порівнювання id. Якщо у замовленні id по ролі клієнта ідентичний тому, що має сторінка, тоді виводиться відповідне замовлення у список. За подібним принципом працює вивід послуг на сторінці профілю спеціаліста.

На рис 3.19 зображено вкладку з відгуками, що можуть залишати клієнти на сторінці спеціаліста. Можна писати відгук будь-якому спеціалісту, який допоміг вирішити проблему клієнту.

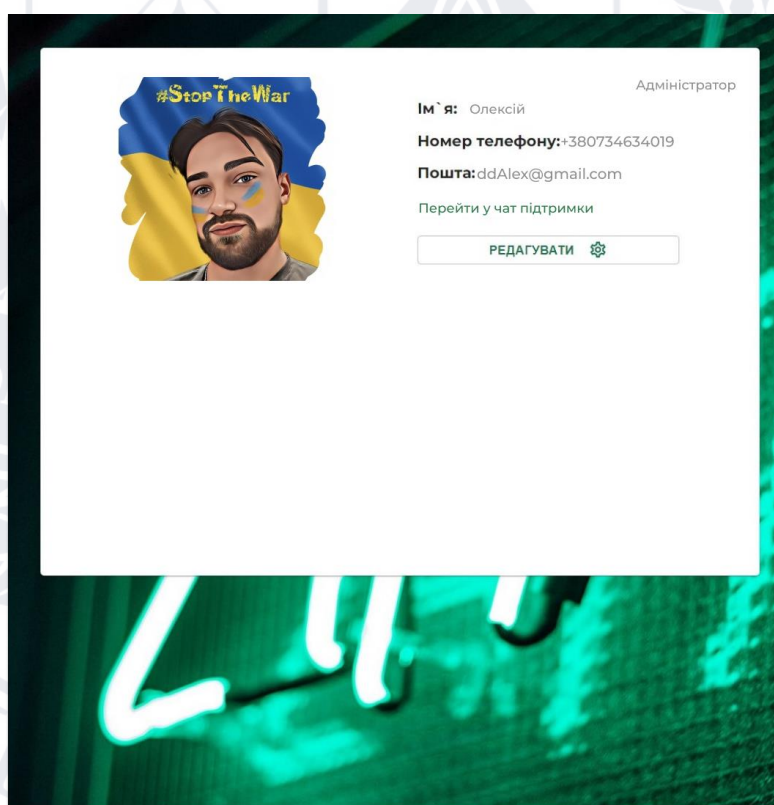


Рисунок 3.21 – Вікно редагування профілю

Рис. 3.21 демонструє сторінку профілю адміністратора. Так як він не є спеціалістом чи клієнтом, то у нього немає вкладок з послугами / відгуками чи

можливості переглянути список замовлень, що він створив. Натомість у адміністратора є можливість перейти у чат підтримки, де він надасть допомогу користувачам, що потребують її. Також на сторінці профілю відображається роль (правий верхній кут), що дозволяє ідентифікувати користувача.

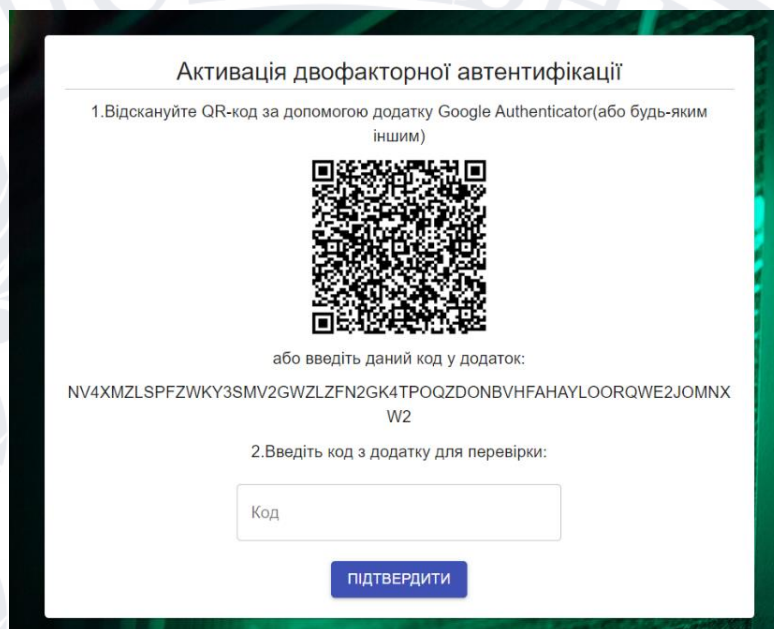


Рисунок 3.22 – Сторінка активації двофакторної автентифікації

На рис. 3.22 продемонстровано сторінку з увімкнення двофакторної автентифікації. Будь-хто з користувачів має можливість увімкнути режим двофакторної автентифікації, проте тільки після авторизації. Лінк для переходу на сторінку режиму для ввімкнення / вимкнення двофакторної автентифікації розміщується у модальному вікні профілю.

Виходячи з того, що у світі багато хакерів, що крадуть інформацію людей, було вирішено додати таку можливість. Перший скрін демонструє увімкнення режиму за допомогою сканування QR-коду або введення коду в додаток Google Authenticator, що дає змогу інтеграції продукту в будь-який додаток. Потрібно лише його завантажити для платформи Android чи IOS на смартфон.

Google Authenticator – додаток для двохетапної аутентифікації за допомогою Time-based One-time Password Algorithm (TOTP) і HMAC-based One-

time Password Algorithm (HOTP) від Google. Сервіс реалізує алгоритми зазначені в RFC 6238 і RFC 4226 [22].

Як правило, користувач повинен встановити програму на свій мобільний пристрій перед початком її використання. Для того, щоб увійти на сайт або скористатися послугами сервісу, потрібно обрати відповідне поле для вводу, ввести ім'я та пароль користувача. Після чого запустити додаток і ввести згенерований одноразовий пароль у спеціально відведене поле. Сайт надає загальний секретний ключ користувачеві, який повинен бути збережений до Google Authenticator додатку. Цей таємний ключ буде використовуватися для всіх подальших майбутніх входів на сайт.

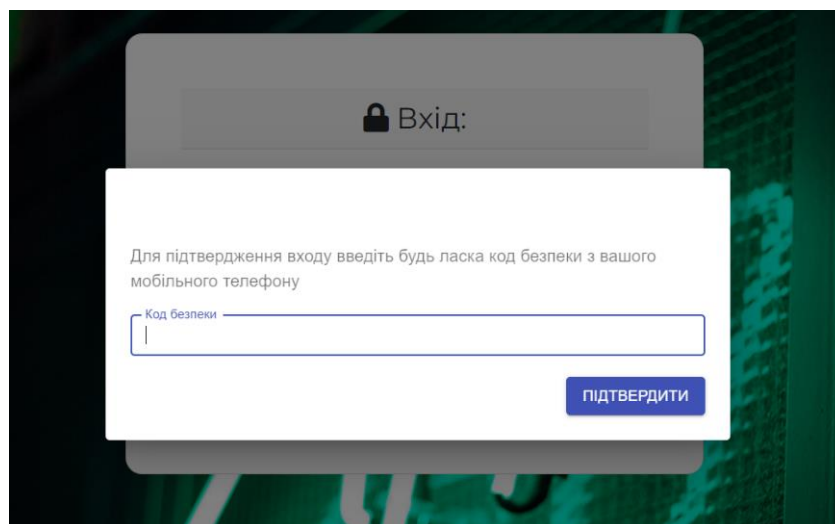


Рисунок 3.23 – Режим входу з двофакторною автентифікацією

На рис. 3.23 зображено режим входу з двофакторною автентифікацією. Будь-хто з користувачів, хто увімкнув її зможе увійти тільки після введення коду безпеки (приклад коду дивитись на рис. 3.24). Тобто користувач заходить на сайт і бачить сторінку входу, де вводить свої дані: логін та пароль або використовує кукі для того, щоб браузер запам'ятав дані для входу. Після того, як дані було введено, користувач натискає кнопку «вхід» і йому буде продемонстровано вікно для введення секретного коду. Якщо його ввели, то авторизація на сайт буде виконана, інакше ні.

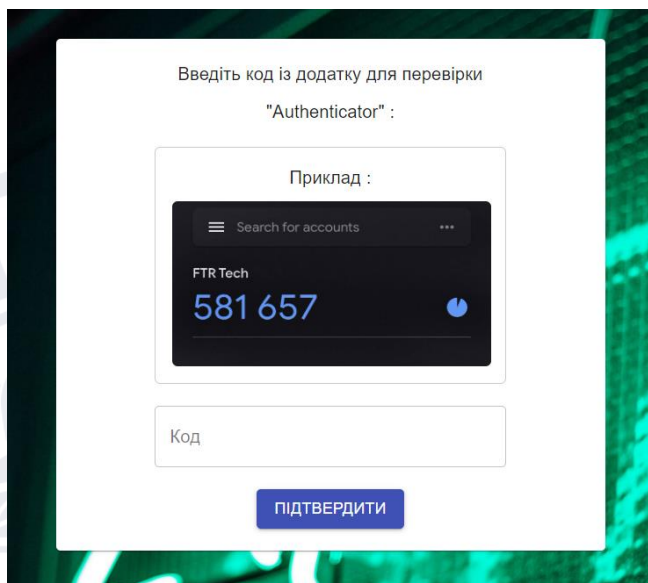


Рисунок 3.24 – Режим входу з двофакторною автентифікацією

На рис. 3.24 продемонстровано сторінку з режимом для вимкнення двофакторної автентифікації. Якщо в користувача з'явилося бажання, щоб вимкнути режим двофакторної автентифікації, тоді потрібно зайти на сторінку, лінк на яку розміщується так само в модальному вікні профілю.

Користувачу так само потрібно зайти в додаток Google Authenticator, де у нього буде список із доступних сайтів, у які інтегрований даний продукт. У кожного з елементів списку є назва сайту, код та логін. Після введення секретного коду до відповідного поля та натискання кнопки «підтвердити» режим буде вимкнено.

Дизайн сайту було розроблено за допомогою сайту Figma, що полегшує малювання дизайну для людей, які жодного разу не мали змоги користування PhotoShop.

Figma – векторний онлайн-сервіс розробки інтерфейсів та прототипування з можливістю організації спільної роботи, що розробляється однойменною компанією [23]. Сервіс працює у відповідних двох форматах: у браузері та як клієнтський додаток на десктопі в користувача. Він автоматично зберігає онлайн-версії файлів, з якими працював користувач.

Цей сервіс легше для засвоєння людям, які нічого не знають програму PhotoShop. Найголовніша фіча даного сервісу – усі файли не зберігаються на комп’ютері локально, а в хмарі. Саме тому людям не потрібно перейматись за збереження і кожного разу шукати чорновики макету з різними незрозумілими назвами, усі зміни зберігаються автоматично. Також можна ділитись лінком на макет, як готовим мокапом так і лінком для спільної розробки.

Висновок до розділу 3

У даному розділі було описано розроблений web-додаток для замовлення та надання бізнес завдань. Також описано всі екрани та принципи роботи.

ВИСНОВКИ

Розробка веб-сервісу для замовлення та надання бізнес завдань є доволі актуальною темою для роботи. Актуальність даної роботи полягає в тому, що станом на сьогодні люди більше часу онлайн, особливо під час війни немає необхідності зайвий раз виходити з дому, тому було вирішено розробити даний сервіс.

Було проведено порівняльний аналіз із схожими системами. Для порівняльного аналізу було обрано 3 web-додатки. У роботі було описано їх індивідуальні особливості.

Для досягнення мети дослідження було вивчено методику проектування web-додатків з використанням мови програмування C# та JavaScript. Постановка задачі полягала в розробці самого web-додатку для замовлення та надання бізнес завдань. Задля її розв'язання було створено рішення розбиття великої задачі на підзадачі та окреслено такі з них: огляд існуючих аналогів з виявленням їх переваг та недоліків; вивчення спеціалізованого середовища розробки Visual Studio Code та Visual Studio. У розробці було використано наступні інструменти: мови програмування C# та JavaScript, фреймворк .NET та бібліотека React, база даних Sql, середовище розробки Visual Studio Code та Visual Studio, бібліотека для UI – MaterialUI, HTML, фреймворк для CSS - SaaS.

Логіка web-додатку побудована на основі сервісу Kabanchik. Це дозволило оцінити обсяг роботи, структуру сервісу та процес розробки. Також був доданий режим поділу послуг, що виконує спеціаліст: «у роботі» та «виконано». Система дозволяє клієнту, що створив замовлення, переглядати його процес виконання. У свою чергу, це дозволяє уникати простроченого терміну виконання, а також у залежності від того який рейтинг у спеціаліста, дозволяє зекономити гроші клієнта.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Апопій В. В., Олексин І. І., Шутовська Н. О., Футало Т. В. Організація і технологія надання послуг. - К.: Академія, 2006. - 311 с.
2. https://uk.wikipedia.org/wiki/%D0%9F%D0%BE%D0%B1%D1%83%D1%82%D0%BE%D0%B2%D1%96_%D0%BF%D0%BE%D1%81%D0%BB%D1%83%D0%B3%D0%B8
3. С# - об'єктно-орієнтована мова [Електронний ресурс] – Режим доступу до ресурсу: <https://docs.microsoft.com/ru-ru/dotnet/csharp/tour-of-csharp/>
4. Платформа NET Framework [Електронний ресурс] – Режим доступу до ресурсу: <https://shwanoff.ru/dotnet-framework/>
5. Практично на будь-яке питання можна знайти відповідь на стековерфлоу [Електронний ресурс] – Режим доступу до ресурсу: <https://stackoverflow.com/>
6. Існує безліч програмних продуктів, бібліотек і фреймворків, які написані та розвиваються під .NET Framework [Електронний ресурс] – Режим доступу до ресурсу: <https://training.epam.ua/#!/News/301?lang=ru>
7. SQL [Електронний ресурс] – Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/SQL>
8. Принцип роботи технології AJAX [Електронний ресурс] – Режим доступу до ресурсу: <https://www.hostinger.com.ua/rukovodstva/chto-takoe-ajax/>
9. За опитуванням на «Stack Overflow» [Електронний ресурс] – Режим доступу до ресурсу: <https://stackoverflow.com/questions/javascript>
10. Звіт про стан фронтенду «State of Frontend 2020» не відкрив нічого нового у перевагах більш ніж 4500 професійних фронтенд-розробників, які брали участь в опитуванні [Електронний ресурс] – Режим доступу до ресурсу: <https://tproger.ru/articles/10-javascript-frejmworkov-kotorye-stoit-vyuchit-v-2021-godu/>
11. Бібліотека «MaterialUI» [Електронний ресурс] – Режим доступу до ресурсу: <https://mui.com/>

12. WebStorm – інтегроване середовище розробки для JavaScript, HTML та CSS від компанії JetBrains [Електронний ресурс] / wikipedia.org – Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/WebStorm>

13. «Visual Studio Code» - засіб для створення, редагування та зневадження сучасних веб-застосунків і програм для хмарних систем [Електронний ресурс] – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Visual_Studio_Code

14. Microsoft Visual Studio – лінійка продуктів компанії Microsoft, що включає інтегровану середу розробки програмного забезпечення та ряд інших інструментів [Електронний ресурс] – Режим доступу до ресурсу: https://ru.wikipedia.org/wiki/Microsoft_Visual_Studio

15. Ринок сервісів для замовлення побутових послуг в Україні в період 2017-2019 років розвивався зі щорічною середньою динамікою близько 19% [Електронний ресурс] – Режим доступу до ресурсу: <https://pro-consulting.ua/ua/pressroom/rynok-servisov-dlya-zakaza-bytovyih-uslug-v-ukraine-sblizhenie-s-potrebitelem-v-usloviyah-karantina>

16. Фріланс-сервіс Freelancehunt.com - це сервіс для пошуку та взаємодії фрілансерів та замовників <https://freelancehunt.com/blog/freelancehunt-com-frilans-siervis-dlia-poiska-i-vzaimodieistviia-frilansierov-i-zakazchikov/>

17. Слово «фріланс» походить від англійського freelance (вільний спис) [Електронний ресурс] – Режим доступу до ресурсу: <http://cashgo.pp.ua/publ/19-1-0-62>

18. Фрілансер (англ. freelancer) - вільний працівник [Електронний ресурс] – Режим доступу до ресурсу: <https://freelancehunt.com/>

19. HTML – це стандартизована мова розмітки документів для перегляду веб-сторінок у браузері [Електронний ресурс] – Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/HTML>

20. CSS. Щоб взнати значення, знову ж таки потрібно посилатись на вікіпедію. Отже, це спеціальна мова стилю сторінок, що використовується для

опису їхнього зовнішнього вигляду [Електронний ресурс] – Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/CSS>

21. Sass – найстабільніша і найпотужніша мова розширень CSS професійного рівня у світі [Електронний ресурс] – Режим доступу до ресурсу: <https://sass-lang.com/>

22. Google Authenticator [Електронний ресурс] – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Google_Authenticator

23. Figma — векторний онлайн-сервіс розробки інтерфейсів та прототипування [Електронний ресурс] – Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/Figma>

24. Microsoft SQL Server [Електронний ресурс] – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Microsoft_SQL_Server

25. TypeScript – це JavaScript із синтаксисом для типів [Електронний ресурс] – Режим доступу до ресурсу: <https://www.typescriptlang.org/>

26. JIT-компілятор (Just in time) [Електронний ресурс] – Режим доступу до ресурсу: https://nekliukov.gitbooks.io/konspekt-po-kinge-dzheffri-rihtera-clr-via-c/content/CLR_Basics/ispolnenie-koda/jit-kompilyator.html

27. .NET Core [Електронний ресурс] – Режим доступу до ресурсу: <https://docs.microsoft.com/ru-ru/dotnet/core/introduction>

28. JavaScript (JS) - динамічна, об'єктно-орієнтована^[4] прототипна мова програмування [Електронний ресурс] – Режим доступу до ресурсу: <https://uk.wikipedia.org/wiki/JavaScript>

29. CSS препроцесор. Це програма, що підпорядковується власному синтаксису, але з нього може згенерувати CSS код [Електронний ресурс] – Режим доступу до ресурсу: https://developer.mozilla.org/ru/docs/Glossary/CSS_preprocessor

30. Оскільки HTML це набір тегів то розробники виділяють найважливіші з них [Електронний ресурс] – Режим доступу до ресурсу: <https://dma-development.net/html/osnovni-tehy-html/>

31. ASP.NET signalr - це бібліотека для ASP.NET розробників, яка спрощує процес додавання веб-функцій у режимі реального часу до програм [Електронний ресурс] – Режим доступу до ресурсу: <https://docs.microsoft.com/ru-ru/aspnet/signalr/overview/getting-started/introduction-to-signalr>

