

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ЕЛЕКТРОНІКИ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

на тему: «Інформаційна система підтримки діяльності агенції по організації виїздів за кордон»

за спеціальністю 122 «Комп'ютерні науки»,
освітньо-професійна програма «Інформаційні технології проектування»

Виконавець роботи: студентка групи ІТ.м-01 Пономаренко Вікторія Сергіївна

Кваліфікаційну роботу
захищено на засіданні ЕК
з оцінкою

«___» грудня 2021 р.

Науковий керівник

(підпис)

к.т.н., доц., Федотова Н. А.

Голова комісії

(підпис)

Шифрін Д.М.

Засвідчую, що у цій дипломній роботі немає
запозичень з праць інших авторів
без відповідних посилань.

Студент _____
(підпис)

Сумський державний університет
Факультет електроніки та інформаційних технологій
Кафедра інформаційних технологій
Спеціальність 122 «Комп'ютерні науки»
Освітньо-професійна програма «Інформаційні технології проектування»

ЗАТВЕРДЖУЮ

Зав. секцією ІТП

_____ В. В. Шендрик
«___» _____ 2021 р.

ЗАВДАННЯ

на кваліфікаційну роботу магістра студентіві

Пономаренко Вікторія Сергіївна

(прізвище, ім'я, по батькові)

1 Тема проекту Інформаційна система підтримки діяльності агенції по організації виїздів за кордон

затверджена наказом по університету від «29» жовтня 2021 р. № 0787-VI

2 Термін здачі студентом закінченого проекту «10» грудня 2021 р.

3 Вхідні дані до проекту технічне завдання для розробки інформаційної системи, запит на розробку інформаційної системи

4 Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити) аналіз обраної предметної області, постановка задачі й методи дослідження, проектування та моделювання, практична реалізація інформаційної системи

5 Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень) актуальність, постановка задачі, дослідження аналогів, порівняльна характеристика аналогів, контекстна діаграма процесу розробки, діаграма декомпозиції процесу розробки, діаграма варіантів використання користувача, діаграма варіантів використання адміністратора, реалізація проекту, база даних, архітектура додатку, реалізація інформаційної системи, посилання на інформаційну систему, публікації проекту, висновки.

6. Консультанти випускної роботи із зазначенням розділів, що їх стосуються:

Розділ	Консультант	Підпис, дата	
		Завдання видав	Завдання прийняв
Аналіз предметної області	Федотова Н. А.		
Постановка задачі та вибір методів розробки	Федотова Н. А.		
Практична реалізація інформаційної системи	Федотова Н. А.		

Дата видачі завдання _____.

Керівник _____
(підпис)

Завдання прийняв до виконання _____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів випускної проекту	Термін виконання етапів проекту	Примітка
1	Ознайомлення з предметної областю	01.09.21-07.09.21	
2	Аналіз конкурентів	08.09.21-17.09.21	
3	Постановка задачі	20.09.21-11.10.21	
4	Розробка дизайну	29.09.21-05.11.21	
5	Проектування бази даних	08.11.21-03.12.21	
6	Розробка каркасу	01.11.21-10.12.21	
7	Розробка функціональних блоків	08.11.21-10.12.21	
8	Тестування	22.11.21-17.12.21	
9	Наповнення контентом	22.11.21-17.12.21	
10	Впровадження інформаційної системи до загального доступу	06.12.21-17.12.21	

Магістрант _____

Пономаренко В.С.

Керівник роботи _____

к.т.н., доц. Федотова Н.А.

РЕФЕРАТ

Тема кваліфікаційної роботи є «Інформаційна система підтримки діяльності агенції по організації виїздів за кордон».

Практична значимість даного дипломного проекту полягає у розробці інформаційної системи по організації виїздів за кордон, що буде відповідати усім встановленим параметрам. Користувачі отримають можливість взаємодіяти із агентствами та отримати гарантію в отриманні якісних послуг.

В першому розділі – проведено детальний аналіз предметної області. У даному обраному розділі виконано: огляд актуальних досліджень та публікацій, аналіз аналогів, постановку задачі.

В другому розділі – сформовано мету та задачі дослідження за темою розробки інформаційної системи. Даний розділ також включає в себе вибір засобів реалізації системи.

В третьому розділі – описано проектування інформаційної системи. Було спроектовано та розроблено діаграму IDEF0 та декомпозиція діаграми. Також було реалізовано діаграму варіантів використання відповідно до функціоналу інформаційної системи. Спроектовано база даних.

В четвертому розділі – виконано реалізацію та детальний опис використання розробленої інформаційної система підтримки діяльності агенції по організації виїздів за кордон зі сторони користувача-замовника, представника агентства та адміністратора.

Результатом проведеної роботи є виконана інформаційна система підтримки діяльності агенції по організації виїздів за кордон.

Кваліфікаційна робота містить 104 сторінок, 12 таблиць, 58 рисунків, список літератури 32 найменувань, 2 додатки.

Ключові слова: ІНФОРМАЦІЙНА СИСТЕМА, НАВЧАННЯ ЗА КОРДОНОМ, СЕРВІС, ПЛАТІЖНА СИСТЕМА, ПРЕДСТАВНИК АГЕНСТВА, КОРИСТУВАЧ-ЗАМОВНИК, ПРОПОЗИЦІЯ, ДОГОВІР, ВІДГУК, ОСВІТА, ТУРИЗМ, НАВЧАННЯ, РОБОТА ЗА КОРДОНОМ, ПРОЕКТ.

ЗМІСТ

ВСТУП.....	6
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	8
1.1 Огляд останніх досліджень і публікацій.....	8
1.2 Аналіз програмних продуктів	9
2 ПОСТАНОВКА ЗАДАЧІ ТА МЕТОДИ ДОСЛІДЖЕННЯ.....	18
2.1. Мета та задачі дослідження.....	18
2.2. Розробка візуальної частини інформаційної системи	19
2.3. Розробка функціональної частини інформаційної системи.....	21
3 МОДЕЛЮВАННЯ ТА ПРОЕКТУВАННЯ.....	24
3.1 Структура інформаційної системи	24
3.2 Структурно-функціональне моделювання процесу.....	27
3.3 Моделювання діаграми варіантів використання.....	29
3.4. Моделювання діаграм діяльності	34
3.5. Проектування бази даних	36
4 ПРАКТИЧНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ	44
4.1 Програмна реалізація	44
4.2 Використання інформаційної системи зі сторони представника агентства	45
4.3 Використання інформаційної системи зі сторони користувача-замовника	56
4.4 Адміністрування інформаційної системи	63
ВИСНОВКИ	68
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	69
ДОДАТОК А. ПЛАНУВАННЯ РОБІТ	73
ДОДАТОК Б. КОД РЕАЛІЗАЦІЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ	83

ВСТУП

Управління технологіями — це справді багатопрофільна галузь, яка об'єднує навички управління персоналом і проектами з надійними знаннями технологічних систем і операцій. Її мета надзвичайно важлива: забезпечити конкурентну перевагу для організацій шляхом ефективного впровадження та розвитку технологій. Інженерні або технологічні підприємства, такі як компанії, що займаються програмним забезпеченням або апаратним забезпеченням, покладаються на добре підготовлених спеціалістів.

Менеджери з технологій можуть зіткнутися зі складним спектром посадових обов'язків, включаючи нагляд за великою кількістю технічного персоналу, проектування та розгортання комп'ютерних та інших технологічних систем, управління проектами, виробництвом та операційним управлінням, оцінкою цінності технології, а також контролем безпеки та якості. Управління технологіями може навіть охоплювати інновації та підприємництво, оскільки це стосується технологічних підприємств.

Крім того, з підвищенням соціально-матеріального рівня життя швидко розвивається внутрішній і виїзний туризм. Для підвищення якості туристичного обслуговування та своєчасного, точного та швидкого охоплення бізнес-інформації необхідно побудувати інформаційну систему управління туризмом.

Зберігання та обробка туристичної інформаційної системи – це інформація, пов'язана з туристичним бізнесом, яка є своєрідною системою підтримки бізнес-рішень. З розвитком інформаційних технологій та оновленням електронної комерції, з появою ряду трьохсторонніх постачальників безпечних платежів, таких як «Alipay» та «Paypal», зараз туризм все більше потребує інформації та онлайн-послуг. Це прискорить побудову внутрішньої інформаційної системи управління туризмом. Інформаційна система управління туризмом буде широко використовуватися в туристичній галузі.

Отже, було сформовано тему роботи, тему, об'єкт, предмет дослідження та інше. Розглянемо кожен із пунктів детальніше.

Тема: «Інформаційна система підтримки діяльності агенції по організації виїздів за кордон».

Головна мета: розробити інформаційну систему у вигляді як відкрита платформа для налаштування взаємодії користувачів із компаніями за декількома напрямками, а саме освіта, робота та туризм.

Об'єкт дослідження: процес взаємодії із користувачами за допомогою інформаційної системи.

Предмет дослідження: методи взаємодії агентства та користувача за для зменшення ризику шахрайства при організації виїздів за кордон.

Гіпотеза: використання розробленої інформаційної системи шляхом встановлення повноцінного контролю над оплатою послуг дозволить надавати гарантії якісних послуг.

Наукова новизна: Вперше використання модулю для повноцінного контролю над оплатою послуг та зниження до мінімуму можливості шахрайства при організації виїздів за кордон.

Практичне значення: Практична значимість даного дипломного проекту полягає у розробці інформаційної системи по організації виїздів за кордон, що буде відповідати усім встановленим параметрам. Користувачі отримають можливість взаємодіяти із агентствами та отримати гарантію в отриманні якісних послуг.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Огляд останніх досліджень і публікацій

На сьогоднішній час безліч фахівців у різних сферах припускають, що вища освіта у сфері подорожей й туризму стикається зі значними руйнівними змінами в багатьох сферах. Адже останнім часом все видозмінюється через зовнішні проблеми, такі як примусове дистанційне чи онлайн-навчання, COVID-19 [1-2].

Студенти хочуть мати можливість навчатися в будь-якому місці, в будь-який час й у будь-якому зручному форматі. Наслідком цього є й зміни у сфері навчання, багато вищих навчальних закладів досліджують нові способи співпраці та обміну ресурсами, щоб задовольнити як потреби студентів, так й промисловості. Особливо в рамках програм по туризму, навчанні за кордоном та працевлаштуванням. Розглянемо кожен із пунктів детальніше.

Готовий доступ до подорожей і розширених технологій соціальних мереж змінив природу навчання за кордоном до такої міри, що сьогоднішній досвід принципово відрізняється від досвіду попередніх епох. Гарним прикладом є США, де все більше студентів їдуть за кордон на короткострокові програми. У той же час протилежна сторона представляє більш різноманітні академічні галузі, де збільшується кількість студентів із природничих, суспільних, бізнес-сфер та інших [3]. Тим не менш, навчання за кордоном залишається очікуваним досвідом для студентів. На мою думку, технології можуть відігравати позитивну роль, особливо якщо студентам надають відповідну підтримку.

Перейдемо також до дослідження визначає поняття процесу групового розвитку як сукупності явищ, що відбуваються в певних групах та робочих колективах. Даний процес безпосередньо представлений у всіх формальних та неформальних соціальних одиницях. При виконанні досліджень та аналізу інформації, було представлено та охарактеризовано різні етапи розвитку команди. Зокрема, до даних етапів відноситься: залежність, диференціація, криза відповідальності, напад на лідера, креативність та реальні можливості команди.

Література з управління проектами описує детальний і загальноприйнятий процес управління ризиками [4]. Дані ризики складаються в основному з чотирьох ітераційних етапів: ідентифікація ризику, оцінка ризику, планування та виконання реагування на ризики, часто включається управління процесом управління ризиками. Розглянемо принцип роботи управління проекту на прикладі із будівництвом. Планування будівельного проекту є важливим елементом в управлінні та виконанні будівельних проектів, який включає визначення робочих завдань та їх взаємодію, а також оцінку необхідних ресурсів та очікуваної тривалості діяльності.

Описану систему ризиків потрібно враховувати при роботі на розробки інформаційної системи. Дотримуючись правил роботи в команді можна досягти значних успіхів у будь-якій сфері й безпосередньо в розробці інформаційної системи [5].

Отже, проаналізувавши всю зібрану інформацію, можна зробити висновок, що обрана тема, а саме розробка інформаційної системи діяльності агенції по організації виїздів за кордон є актуальною. Дана тематика включає в себе як організація менеджменту, так й вирішення проблем виїзду за кордон.

1.2 Аналіз програмних продуктів

Розглянемо декілька компаній підготовки до виїзду за кордон для працевлаштування, туризм чи навчання. Було обрано такі агентства як «Поїхали з нами», «Work Department» та «InfoStudy».

1.2.1 Туристична мережа «Поїхали з нами» [7]

Туристична мережа «Поїхали з нами» — це найбільша мережа туристичних агенцій в Україні, що налічує більше ніж 300 офісів у 79 містах України. До того ж, обрана мережа вийшла у лідери в 2015 році за кількістю відправлених туристів. Дана мережа щодня допомагають більше ніж 800 туристів обрати країни та різноманітні тури. За відкритими даними 96% із користувачів туристичної мережі залишаються задоволені своїми поїздками.

На головній сторінці представлена актуальна інформація, блоки із загальною інформацією та форма для пошуку (рис.1.1). Інформація чудово оформлена та структурована, що є вагомим перевагою.

Окремо також виділю частину з пошуком. Даний функціонал зручний у використанні та зручний у використанні для звичайних користувачів (рис.1.2).

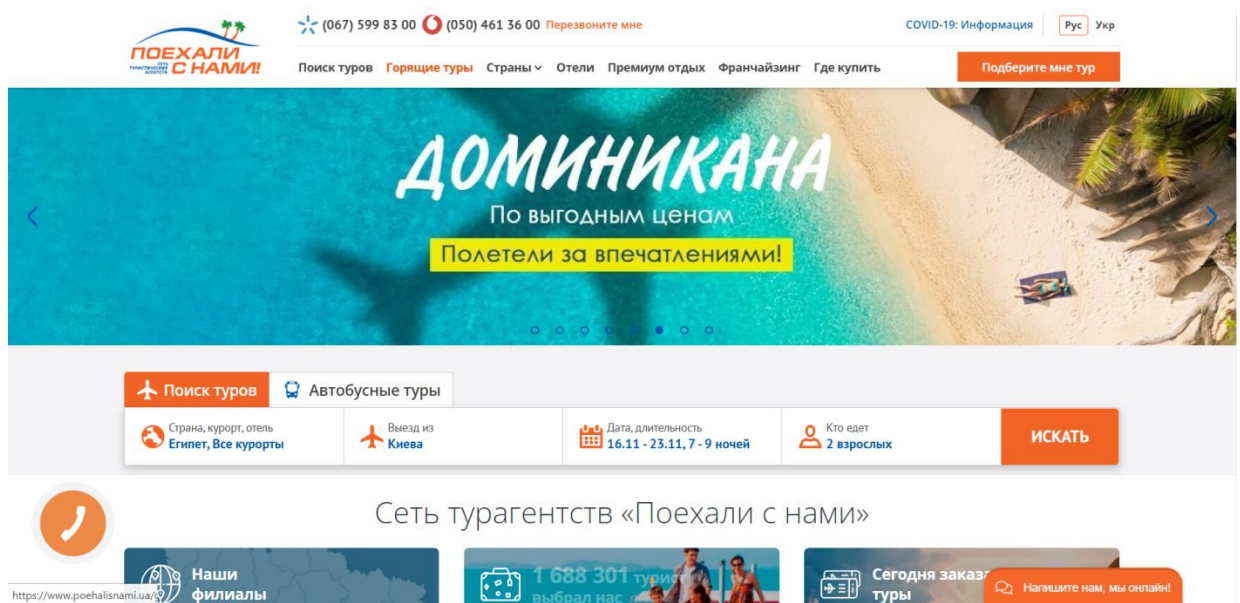


Рисунок 1.1 – Головна сторінка туристичної мережі «Поїхали з нами»

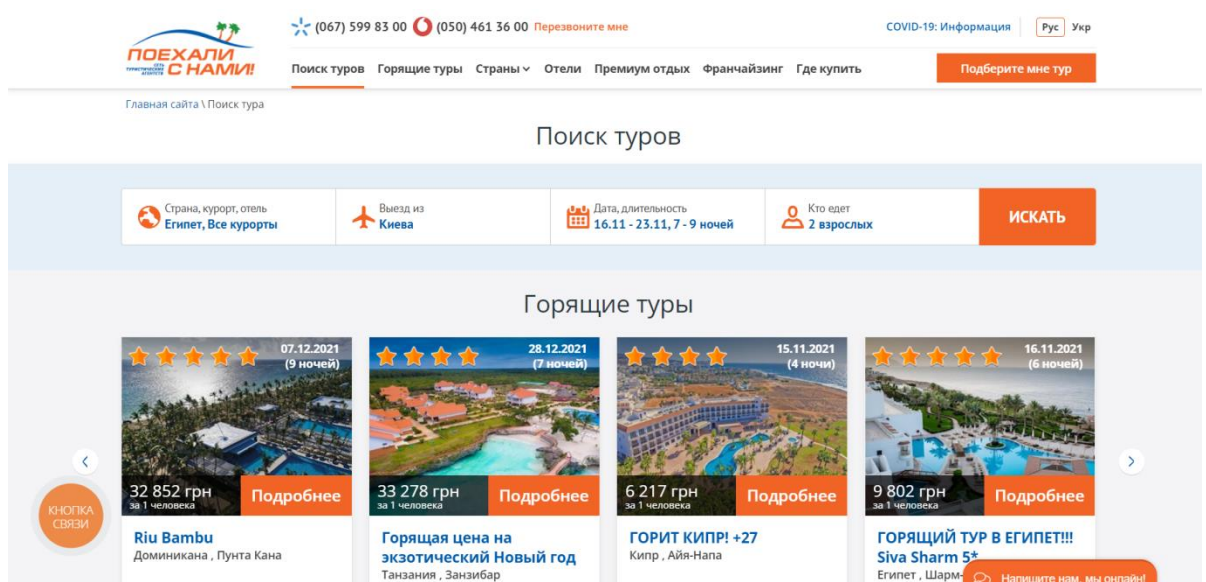


Рисунок 1.2 – Приклад пошуку за обраними параметрами

На рисунку 1.3 представлено додатковий розподіл за популярними темами: відпочинок на морі, пляжні тури, відпочинок з дітьми та інше.

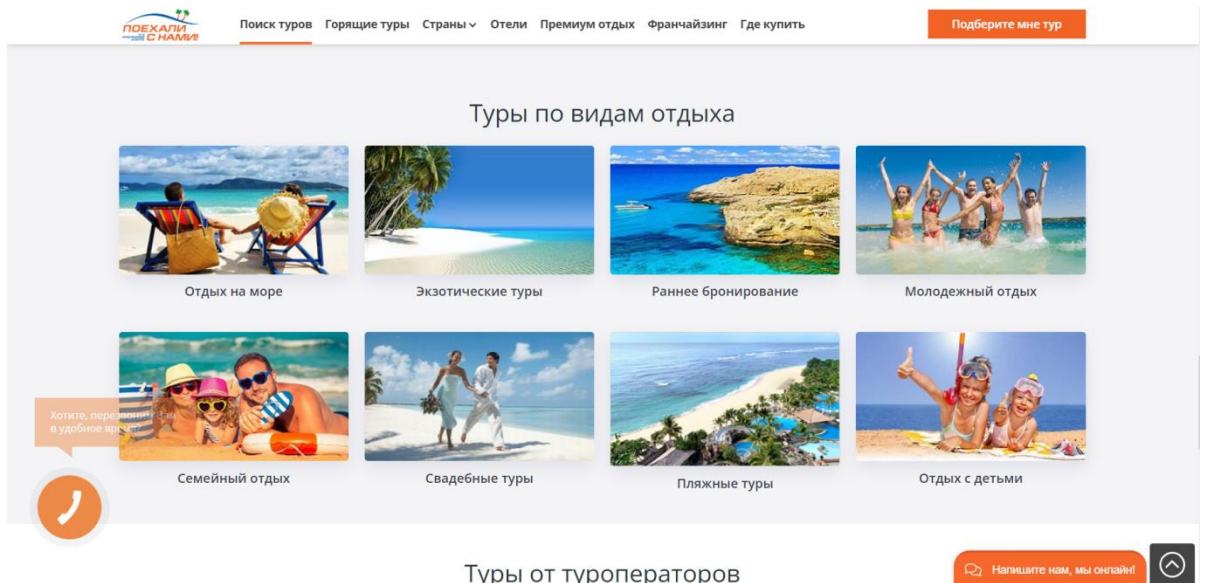


Рисунок 1.3 – Розподіл за видами відпочинку

До недоліків обраної туристичної мережі можна віднести надокучливе спливаюче вікно (рис.1.4). Дана система заохочення є застарілою та не ефективною на сьогоднішній час. Зазвичай у даному блоці відповіді на запитання приходять загального характеру.

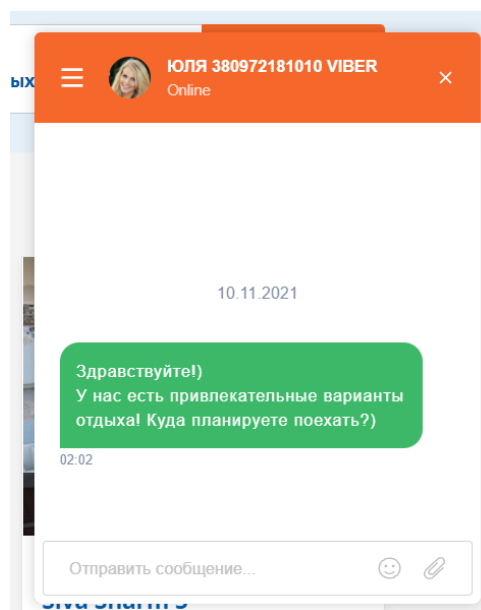
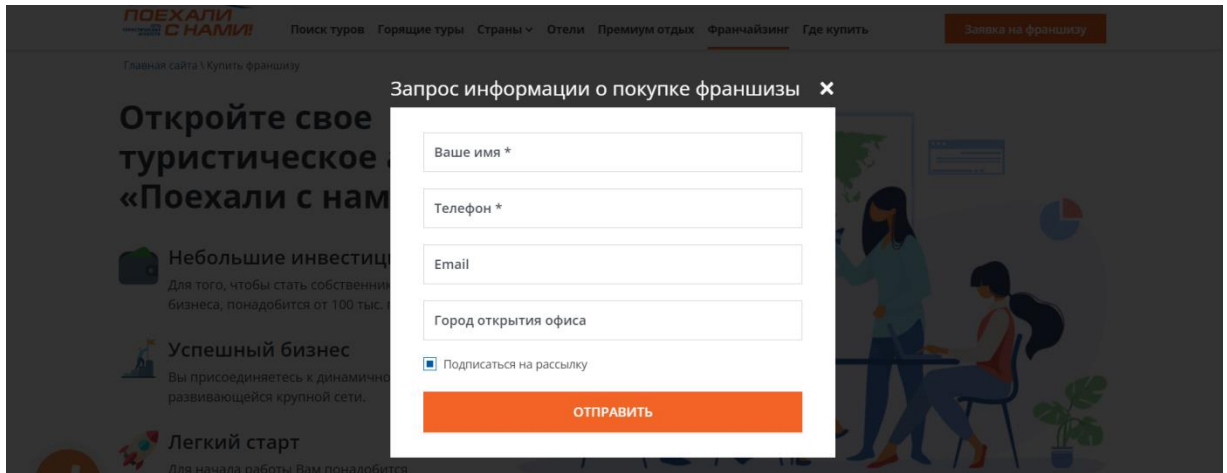


Рисунок 1.4 – Модульне вікно для запитань

Також до недоліків можна віднести відсутність можливості комунікації. В інформаційній сторінці можна лише відправити запит на певне відділення у місці замовника (рис.1.5).



Поехали с нами! Поиск туров Горячие туры Страны Отели Премиум отдых Франчайзинг Где купить Запрос на франшизу

Главная сайта Купить франшизу

Откройте свое туристическое «Поехали с нами»

Небольшие инвестиции
Для того, чтобы стать собственником бизнеса, понадобится от 100 тыс.

Успешный бизнес
Вы присоединяетесь к динамично развивающейся крупной сети.

Легкий старт
Для начала работы Вам понадобится

Запрос информации о покупке франшизы ✕

Ваше имя *

Телефон *

Email

Город открытия офиса

Подписаться на рассылку

ОТПРАВИТЬ

Рисунок 1.5 – Вікно для відправлення запиту

1.2.2 Агентство по трудоустройству «Work department» [8]

Обране агентство по трудоустройству представляє інформацією на декількох мовах, що є досить зручним рішенням. На головній сторінці (рис.1.6) знаходиться навігаційне меню, рекламний банер та кнопка для заповнення запиту.

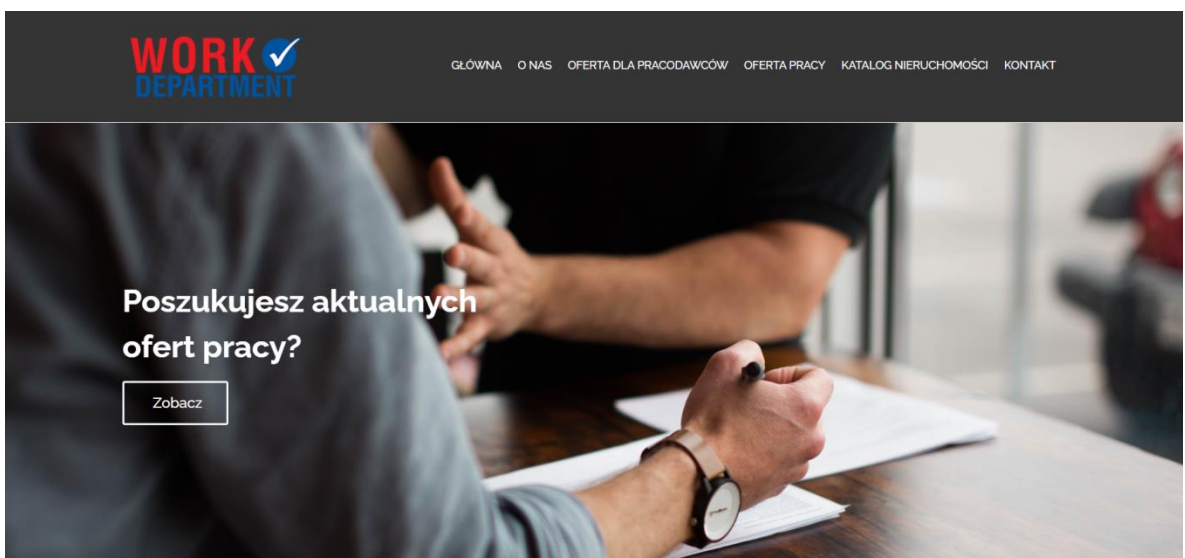


Рисунок 1.6 – Головна сторінка агентства по трудоустройству

До недоліків можна віднести неякісний дизайн. Це відображено у реалізації актуальних вакансій (рис.1.7-1.8). Заголовки та додатковий текст мають недостатню міжрядкову відстань. Крім того, заголовок, текст та кнопка одного кольору, що також є помилкою.

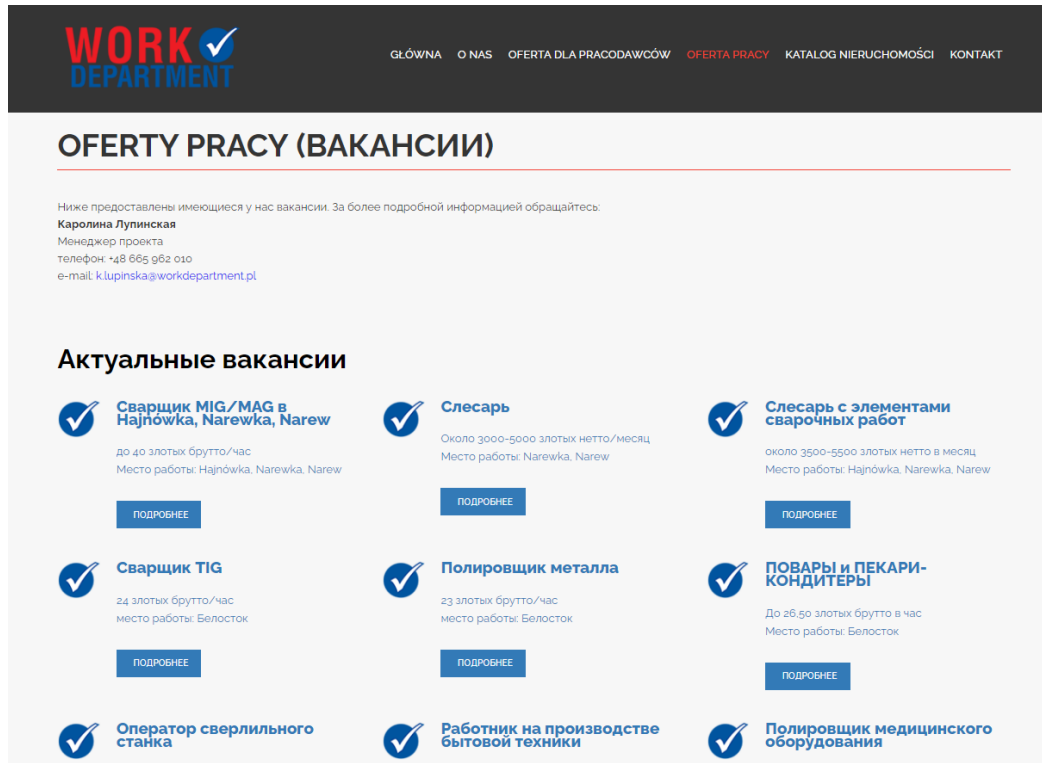


Рисунок 1.7 – Перелік вакансій

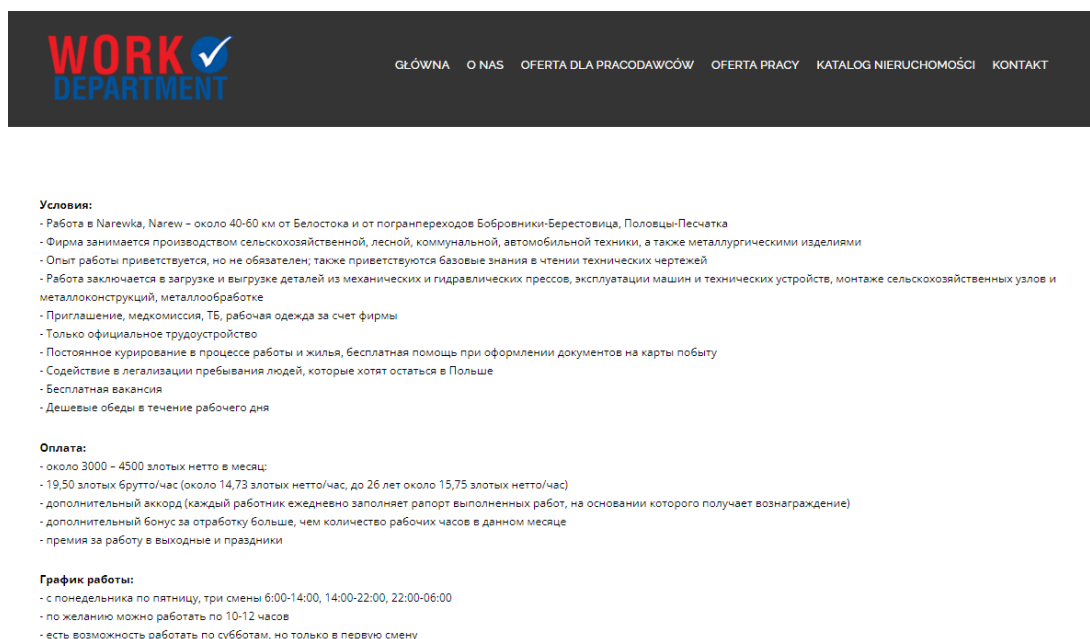


Рисунок 1.8 – Інформація про обрану вакансію

1.2.3 Агентство «InfoStudy» [9]

На головній сторінці інформаційної системи агентства «InfoStudy» також представлено банер та навігаційні кнопки для перегляду можливих для виїзду країн та запис на консультацію (рис.1.9).

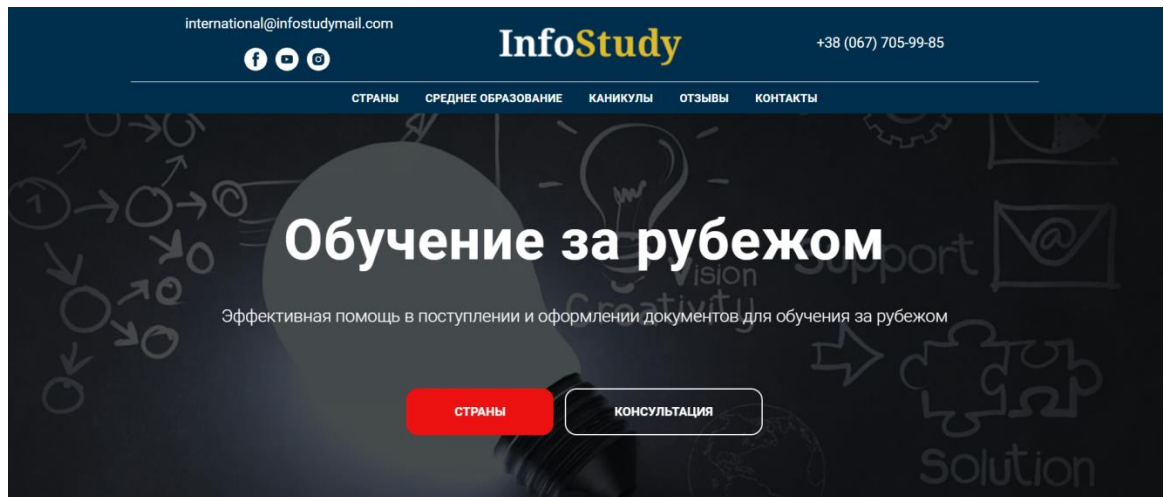


Рисунок 1.9 – Головна сторінка «InfoStudy»

Перейдемо на сторінку із переліком країн (рис.1.10). До переваг інформаційної системи агентства можна віднести якісний дизайн та рішення для подачі інформації. Адже користувач має можливість витратити менше часу на пошук [10].

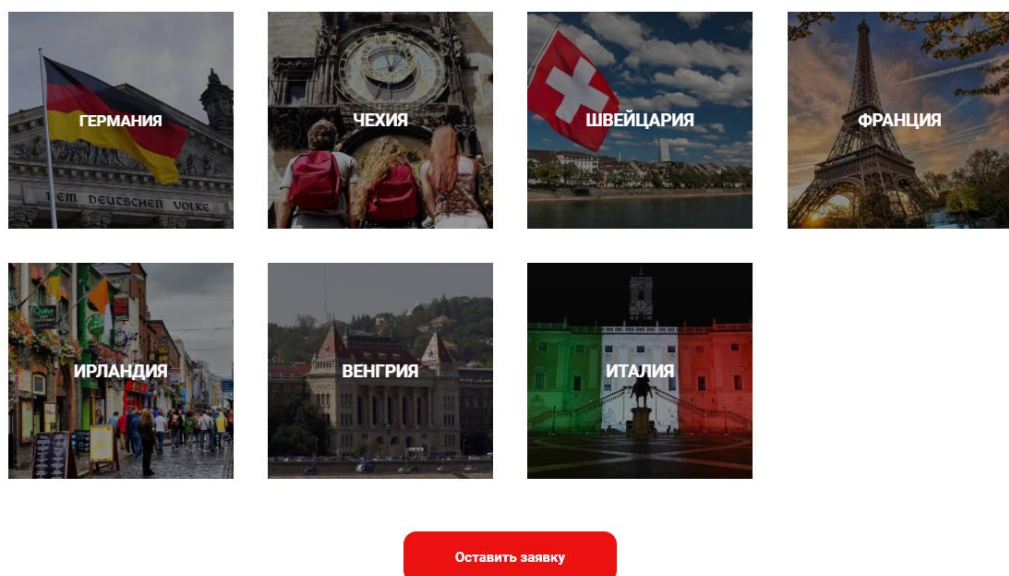


Рисунок 1.10 – Блоки відсортовані за країнами

Виконаємо перехід блок «Великобританія». На даній сторінці знаходиться оновлена інформація про навчання в країні за декількома напрямками (рис.1.11).

Программы обучения в Великобритании



Университеты
Великобритании

Подробнее



Бакалавриат в
Великобритании

Подробнее



Магистратура в
Великобритании

Подробнее



Среднее
образование в
Великобритании

Подробнее

Подать заявку

Учебные заведения Великобритании

Мы рекомендуем

Рисунок 1.11 – Сторінка із інформацією про навчання у Великобританії

Подавши заявку у «InfoStudy» користувач перенаправляється на сторінку із формою для заповнення особистих даних (рис.1.12).

Оставьте заявку на консультацию

Мы свяжемся с вами ближайшее время и ответим на все волнующие вас вопросы

Уровень знания английского

Вы можете оставить дополнительный комментарий:

ОТПРАВИТЬ ЗАЯВКУ

Рисунок 1.12 – Сторінка для оформлення заявки

1.2.4 Порівняльна характеристика аналогів

Виконаємо порівняльну характеристику аналогів інформаційної системи підтримки діяльності агенції по організації виїздів за кордон (табл.1.1). Було проаналізовані такі системи як: «Поїхали з нами», «Work Department» та «InfoStudy».

Таблиця 1.1 – Порівняльна характеристика аналогів

Критерії	«Поїхали з нами»	«Work Department»	«InfoStudy»	Інформаційна система «Silinez»
Зручність використання	+	–	+	+
Якісний дизайн	+	–	+	+
Багатомовності	–	+	–	+
Модуль для подання запиту	+	–	+	+
Наявність особистого кабінету користувача	–	–	–	+
Інформація про замовлення	+	+	+	+
Модуль для написання повідомлення агентству	–	–	–	+
Можливість обрати послуги	+	+	+	+
Групування інформації	+	–	+	+
Адаптивність під мобільні пристрої	+	–	–	+

Отже, при проведенні порівняльної характеристики було виявлено аспекти на які потрібно бути звернути увагу перш за все. До даних категорій можна віднести:

реалізація багатомовності, особистого кабінету користувача та адаптивність під мобільні пристрої.

Крім того, буде реалізований модуль для листування із представником агентства за обраною тематикою.

2 ПОСТАНОВКА ЗАДАЧІ ТА МЕТОДИ ДОСЛІДЖЕННЯ

2.1. Мета та задачі дослідження

На сьогоднішній день, проблема виїзду громадян за кордон є досить актуальною. Звичайно, існує безліч причин здійснення такого кроку. Кожен замислюється над тим, де провести відпустку чи про наявність отримати освіта за кордоном. Крім того, приводом виїзду можуть полягати в економічному чи соціальному становищі. Наприклад безробіття, відсутність вакансії за фахом, проблеми з отримання відповідних навичок та інше.

Віднайти компанію, що оформляє документи, підбирає та знаходять вакансію у час новітніх технологій зазвичай не проблема. Головним фактором недовіри, зі сторони клієнтів – відсутність гарантії надання якісних послуг.

Отже, головна мета інформаційної системи для виїзду за кордон полягає в налаштуванні та урегулюванні роботи замовника та компанії по підборі місць чи підготовці до виїзду за кордон. Розроблена система зменшить ризики шахрайства як з боку фірми по наданням послуг, так й зі сторони користувача.

Інформаційна система використовуватиметься як платформа для взаємодії користувача із різними агентствами. Для досягнення сформованої мети магістерської роботи буде реалізовано наступний перелік наступних задач та функцій:

- Реєстрація та авторизація користувачів;
- Розміщення на сайті можливих пропозицій відповідно до типу компанії;
- Перегляд сформованих можливостей зареєстрованими користувачами;
- Створення модулю для пошуку та фільтрації пропозицій в інформаційній системі;
- Формування договорів;
- Формування рейтингу та історії виконаних проектів;
- Можливість формування коментарю після оформлення та виконання договору;

- Можливість спілкування між представником агентства та користувачем за допомогою чату;

- Блокування користувачі зі сторони адміністрації інформаційної системи;

Крім того, даний сервіс полегшить пошук відповідних пропозицій для користувача, та підвищить кількість замовлень для компанії виконавця. Інформаційна система підтримки діяльності агенції по організації виїздів за кордон використовуватиметься як відкрита платформа для налаштування взаємодії користувачів із компаніями за декількома напрямками, а саме освіта, робота та туризм. Найважливіша відмінність – повноцінний контроль над оплатою послуг та зниження до мінімуму можливості шахрайства.

2.2. Розробка візуальної частини інформаційної системи

Розробку інформаційної системи підтримки діяльності агенції по організації виїздів за кордон включає в себе розробку візуальної частини. Для виконання візуальної частини використовувався HTML, CSS та Javascript.

HTML є будівельними блоками кожної сторінки інформаційної системи. Наприклад HTML використовується як текст-маркування, а CSS повідомляє веб-серверу стиль та структуру документа. CSS також включаючи макет, кольори, шрифти, відступи тощо. Разом з HTML, CSS та Javascript, складає основу того, як працює візуальна частина інформаційної системи підтримки діяльності агенції по організації виїздів за кордон.

Було також проведено порівняльний аналіз переваг та недоліків декількох найпопулярніших фреймворків мови програмування Javascript. Порівнювалися такі фреймворки як React.js, Angular та Vue.js. Детальну інформацію по кожному із представлених фреймворків можна переглянути у таблиці 2.1.

Таблиця 2.1 – Порівняльна характеристика фреймворків Javascript

React.js			
№	Переваги	№	Недоліки
1	Сильна спільнота, адже на сьогоднішній час React.js є другим найпопулярнішим фреймворком.	1	Не працює без інших бібліотек і служб, таких як Redux, маршрутизацій тощо.
2	Гнучкість React – модульність бібліотеки	2	Крута крива навчання [20].
3	Бібліотека React, та компоненти, допомагають розробникам підвищити продуктивність [19].	3	
Angular			
№	Переваги	№	Недоліки
1	Використання при розробці двосторонньої прив'язки даних.	1	Фреймворк схожий на дерево компонентів, тому потрібно гарно розуміти, як працюють всі елементи.
2	Швидше прототипування додатків.	2	Обмежена доступна документація може надалі впливати на процес навчання [22-23].
3	Покращена продуктивність роботи сервера [22].	3	
Vue.js			
№	Переваги	№	Недоліки
1	Зручний для розробників фреймворк. Підходить для тих, хто не мають достатньо попереднього досвіду використання фреймворків у своєму процесі створення [21].	1	Невелика кількість кваліфікованих працівників, що не відповідає потребі ринку.
2	Елегантний синтаксис мови.	2	Брак досвідчених розробників.

Продовження таблиці 2.1

Vue.js			
№	Переваги	№	Недоліки
3	Добре структурований код.	3	Велика кількість документації на китайській мові.
4	Видатна продуктивність.		
5	Точна та якісна користувальницька документація.		

У результаті було виконано порівняльний аналіз переваг та недоліків таких фреймворків як React.js, Angular та Vue.js. Для розробки візуальної частини було обрано фреймворк Vue.js, адже він досить легкий у використанні новачкам та ідеально відходить для розробки інформаційної системи підтримки діяльності агенції по організації виїздів за кордон.

2.3. Розробка функціональної частини інформаційної системи

Розробку інформаційної системи підтримки діяльності агенції по організації виїздів за кордон включає в себе розробку функціональної частини. Було також проведено порівняльний аналіз переваг та недоліків декількох найпопулярніших мов програмування, що використовується для розробки веб-додатків.

Порівнювалися такі мови програмування як PHP, Java, C# та Ruby. Детальну інформацію по кожній із мов програмування представлено у таблиці 2.2.

Таблиця 2.2 – Порівняльна характеристика мов програмування

PHP			
№	Переваги	№	Недоліки
1	Потужна система шаблонів.	1	Існують простіші мови програмування для веб-розробки.

Продовження таблиці 2.2

PHP			
№	Переваги	№	Недоліки
2	Хороша документація.	2	Відсутність спеціалізованих бібліотек для сучасних потреб
3	Простий і швидкий механізм маршрутизації [25].	3	Присутні проблеми із безпекою продуктів на основі PHP.
4	Покращена продуктивність роботи сервера.	4	
Java			
№	Переваги	№	Недоліки
1	Використання та створення окремих програм на основі Spring.	1	Дана мова потребує значні знання в програмуванні на Java.
2	Створений для великомасштабних систем, які використовують хмарні технології.	2	Випадків використовується для великих систем та мікросервісів.
3	Чудова документація.	3	
C#			
№	Переваги	№	Недоліки
1	Мова високого рівня з можливостями доступу до пам'яті	1	Низька продуктивність роботи розроблених додатків [26].
2	Являється частиною платформи .NET	2	Залежність від платформи .NET
3	Точна та якісна користувальницька документація.	3	
Ruby			
№	Переваги	№	Недоліки
1	Стійкі до вразливостей за допомогою вбудованих механізмів [27].	1	Відсутність інтересу до цієї системи.

Продовження таблиці 2.2

Ruby			
№	Переваги	№	Недоліки
2	Досить інтуїтивно зрозумілий, компактний і легкий для розуміння	2	Відсутність гнучкості
3	значну кількість безкоштовних бібліотек з відкритим кодом		

У результаті було виконано порівняльний аналіз переваг та недоліків таких мов програмування як PHP, Java, C# та Ruby. Для розробки функціональної частини було обрано мову програмування PHP.

3 МОДЕЛЮВАННЯ ТА ПРОЕКТУВАННЯ

3.1 Структура інформаційної системи

Інформаційна система для виїзду за кордон повинна відповідати переліку вимог, що розподілені на три категорії: візуальне наповнення, контент та структура. Детальніша інформація представлена у табл.3.1.

Таблиця 3.1 – Вимоги до інформаційної системи

№	Категорія	Назва	Опис
1	Структура	Колонтитул	На кожній сторінці у верхній частині повинен знаходитися колонтитул із навігаційним меню та логотипом інформаційної системи.
2		Навігаційне меню	Навігаційне меню представляє з себе перелік гіперпосилань на основні сторінки інформаційної системи.
3		Основний вміст	Блок на сторінці, що займає більшу частину сторінки та має інформаційне наповнення відповідно до тематики.
4		Нижній колонтитул	Блок в нижній частині сторінки, куди виносять корисну, але не першорядну інформацію. Як приклади дані про копірайт, карта сайту, дублювання основних пунктів навігації та інше.
5	Візуальне наповнення	Гарна читабельність шрифтів	Для зручності використання інформаційної системи потрібно використовувати шрифти зі зручною читаністю. Адже непридатні для

№	Категорія	Назва	Опис
			читання шрифти ускладнюють зручність перегляду.
6		Палітра кольорів	При виборі палітри кольорів потрібно переконатися, що вони мають високий рівень доступності, буде швидко зчитуватиметься всіма користувачами.
7		Адаптивна верстка	При реалізації адаптивної верстки сайту будь-який користувач матиме можливість використовувати інформаційну систему на телефоні.
8		Єдиний стиль	Для зручного використання також важливо підтримувати єдиний стиль для оформлення блоків з інформацією, картинок, іконок та іншого.
9	Контент	Вдалість використання	Текст повинен бути вдалим, потрібно підкріплювати його ілюстраціями, або навіть замінювати фотографіями.
10		Оформлення тексту	Потрібно приділити увагу оформленню тексту, адже це впливає на те, як звичайний користувач буде сприймати ту чи іншу інформацію.

Розглянемо детальніше частина щодо структури інформаційної системи. Загальну структуру інформаційної системи підтримки агентства по виїзду за кордон можна розділити на дві частин: до та після авторизації. Головна відмінність у даному розподілі – доступи до певних реалізованих функціональних модулів та сторінок.

До авторизації у системі користувачам відкриті модулів та сторінок, що представлені у таблиці 3.2.

Таблиця 3.2 – Доступні сторінка до авторизації в інформаційній системі

№	Сторінка	Опис
1	Головна сторінка	Дана сторінка представляє з себе маркетингову рекламу та інформативну сторінку, що має основну інформацію про інформаційну систему.
2	Сторінка авторизація	Дана сторінка використовується для авторизації вже зареєстрованих у системі користувачів.
3	Сторінка реєстрація	Дана сторінка використовується для реєстрації нових користувачів у системі.

Після авторизації чи реєстрації користувач отримує перелік додаткових можливостей в залежності від типу акаунту. Розглянемо детальніше, які модулі доступні для кожного типу акаунту в таблиці 3.3.

Таблиця 3.3 – Таблиця рівнів доступу до сторінок інформаційної системи

№	Сторінки	Представник агентства	Користувач-замовник
1	«Список можливостей»	+	+
2	«Мої замовлення»	+	–
3	«Мої запити»	+	+
4	«Повідомлення»	+	+
5	«Агентства»	+	+
6	«Користувачі»	+	+
7	«Новини тематики»	+	+
8	«Найчастіші запитання»	+	+
9	«Сторінка редагування особистої інформації»	+	+

Інформаційна система підтримки агентства по виїзду за кордон якісно спроектована для продуктивної взаємодії всіх типів користувачів. Даний тезис продемонстрований у розробленій структурі.

3.2 Структурно-функціональне моделювання процесу

Розглянемо розроблену IDEF0. Кожне поле діаграми представляє з себе окремий процес, але вони мають свою особливість. IDEF0 відрізняється розміщенням та використанням стрілок [11-13, 17]. Також, окрім звичайних входів/виходів, представлені ще два типи стрілок. Дані стрілки представляють із себе елементи управління.

Розглянемо IDEF0 інформаційної системи підтримки агентства по виїзду за кордон (рис.3.1).



Рисунок 3.1 – IDEF0 інформаційної системи

Як показано на рис.3.1, входами моделі є запит, що пов'язаний із виїздом за кордон (пошук роботи, навчання чи туризм), інформація про користувача системи та інформація про пропозицію [14].

Виходи: виконаний договір представлений представником-агенства, звіт про виконання проекту користувачами та підвищення рейтингу.

Наступним етапом є проведення детальної декомпозицію контекстної діаграми на функціональні блоки:

- Реєстрація /Авторизація зареєстрованого користувача у системі;
- Створення посту за обраною тематикою, що до можливості виїзду за кордон;
- Перегляд отриманих запитів від користувачів;
- Обговорення та прийняття умов співпраці представника агентства та користувача;
- Виконання оплати відповідно до пропозиції;
- Переведення зарезервованих коштів представнику агентства.

Розглянемо IDEF1 [15] інформаційної системи підтримки агентства по виїзду за кордон (рис.3.2).

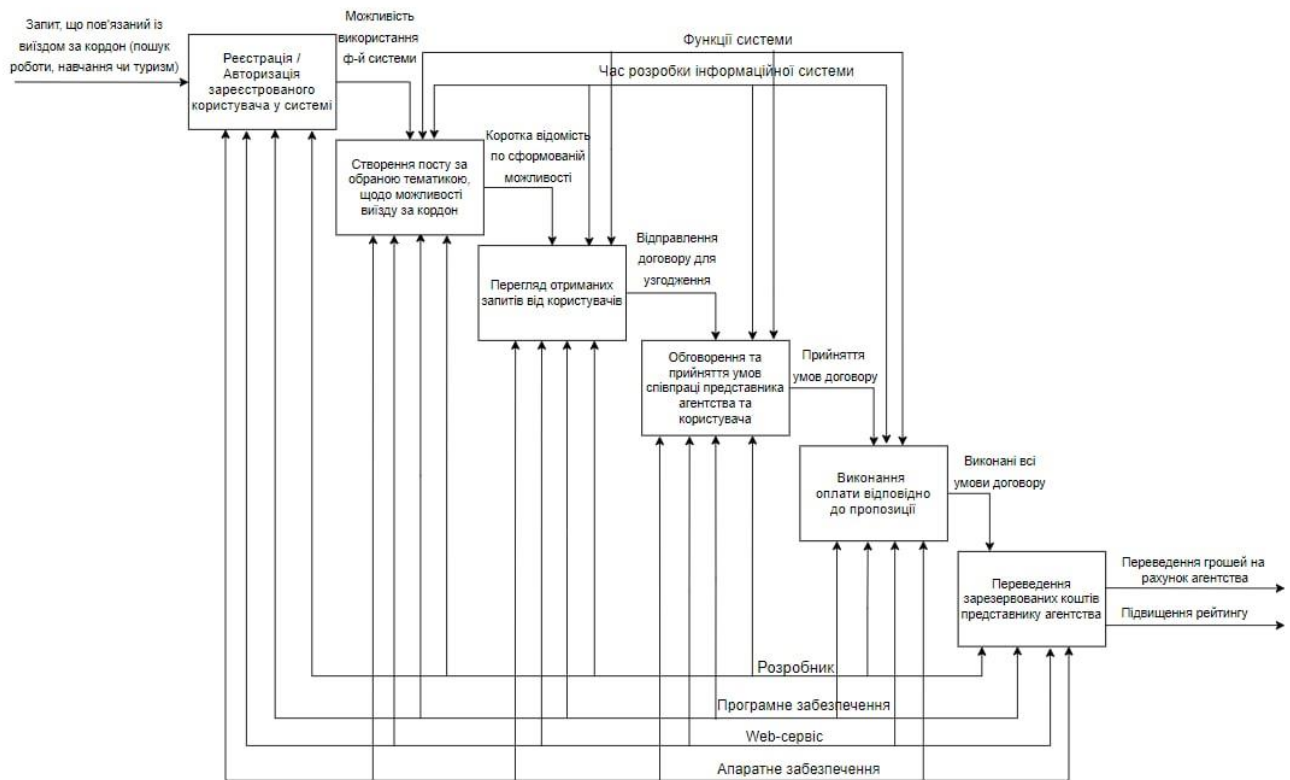


Рисунок 3.2 – IDEF1 проекту

3.3 Моделювання діаграми варіантів використання

Для представлення повноцінного функціоналу інформаційної системи підтримки діяльності агенції по організації виїздів за кордон було реалізовано діаграму варіантів використання [16, 18].

Розглянемо детальніше акторів та варіанти використання інформаційної системи підтримки діяльності агенції по організації виїздів за кордон в табл.3.4 та табл.3.5.

Таблиця 3.4. - Опис акторів

№	Іменування актора	Детальний опис
1	Представник агентства	Даний тип користувача представляє з себе представника агентства. Представник агентства має можливість розміщувати пости та формувати пропозиції для оформлення договорів.
2	Користувач-замовник	Даний тип користувача має можливість відправляти запити на сформований пост представником агентства для заключення договору за однією із тематики, а саме: робота, туризм та навчання.
3	Адміністратор	Даний тип користувача має найбільший спектр можливостей на сайті. Адміністратор має можливість редагувати та видаляти дані на сайті, відповідати на запитання.

Таблиця 3.5. - Опис варіантів використання

№	Варіант використання	Детальний опис	Іменування актора
1	Авторизація	Даний модуль дозволяє авторизуватися зареєстрованому користувачеві на сайті.	Всі користувачі
2	Реєстрація	Даний модуль дозволяє зареєструватися новому користувачеві.	Всі користувачі
3	Редагування інформації на сайті	Даний модуль дозволяє адміністратору редагувати та змінювати загальну інформацію на сторінках інформаційної системи.	Адміністратор
4	Перегляд пропозицій	Даний модуль дозволяє переглядати інформацію про сформовані пропозиції. Переглядати пропозиції можуть лише зареєстровані користувачі інформаційної системи.	Всі користувачі
5	Перегляд інформації про зареєстрованих користувачів (представників агентств та користувачів-замовників)	Даний модуль дозволяє зареєстрованим користувачам переглядати інформацію (контактна інформація, відгуки, інформація про місце роботи) про інших користувачів, хоч представників агентства, хоч користувачів-замовників.	Всі користувачі
6	Створення нової пропозиції	Даний модуль дозволяє представнику агентства створювати нові пропозиції за напрямком роботи компанії.	Представник агентства

Продовження таблиці 3.5.

№	Варіант використання	Детальний опис	Іменування актора
7	Редагування створеної пропозиції	Даний модуль дозволяє представнику агентства редагувати власні пропозиції за напрямком роботи компанії.	Представник агентства
8	Відповідь на запит від користувача-замовника	Даний модуль дозволяє представнику агентства відповідати на запит користувача до певної пропозиції. Відповідаючи на запит представник агентства формує запит для обговорення умов договору для подальшої співпраці із користувачем-замовником.	Представник агентства
9	Запит на створену пропозицію від представника агентства	Даний модуль дозволяє користувачу-замовнику відправляти запит на створену пропозицію від представника агентства.	Користувач-замовник
10	Формування відгуку	Даний модуль дозволяє залишити відгук про співпрацю із користувачем після оговореної дати виконання роботи згідно договору.	Представник агентства, користувач-замовник
11	Чат користувачі системи	Даний модуль дозволяє спілкуватися через вбудований чат представнику агентства та користувачу-замовнику. При конфліктних ситуаціях адміністрація матиме можливість перевірити достовірність інформації [30-32].	Представник агентства, користувач-замовник

Продовження таблиці 3.5.

№	Варіант використання	Детальний опис	Іменування актора
12	Пошук та фільтрації пропозицій	Даний модуль дозволяє виконувати фільтрацію пропозицій тематики за тегами, ціною та країною. Також присутній окремий пошук за назвою.	Всі користувачі
13	Блокування користувачів	Даний модуль дозволяє адміністратору заблокувати користувачу.	Адміністратор
14	Редагування особистих даних	Даний модуль дозволяє редагувати особисту інформацію в інформаційній системі, наприклад: змінювати контактну інформацію, оновлювати фото, редагувати інформацію про агентство.	Всі користувачі
15	Формування запитання адміністрації	Даний модуль дозволяє користувачеві інформаційної системи підтримки діяльності по виїзду за кордон задавати та відправляти адміністраторам запитання.	Представник агентства, користувач-замовник
16	Публікація новин та статей	Даний модуль дозволяє формувати статті чи новини. Даний функціонал можливий лише з панелі адміністратора.	Адміністратор

Сформувавши представлення про функціонал інформаційної системи підтримки діяльності агенції по організації виїздів за кордон було сформовано діаграма варіантів використання на рисунку 3.3-4.

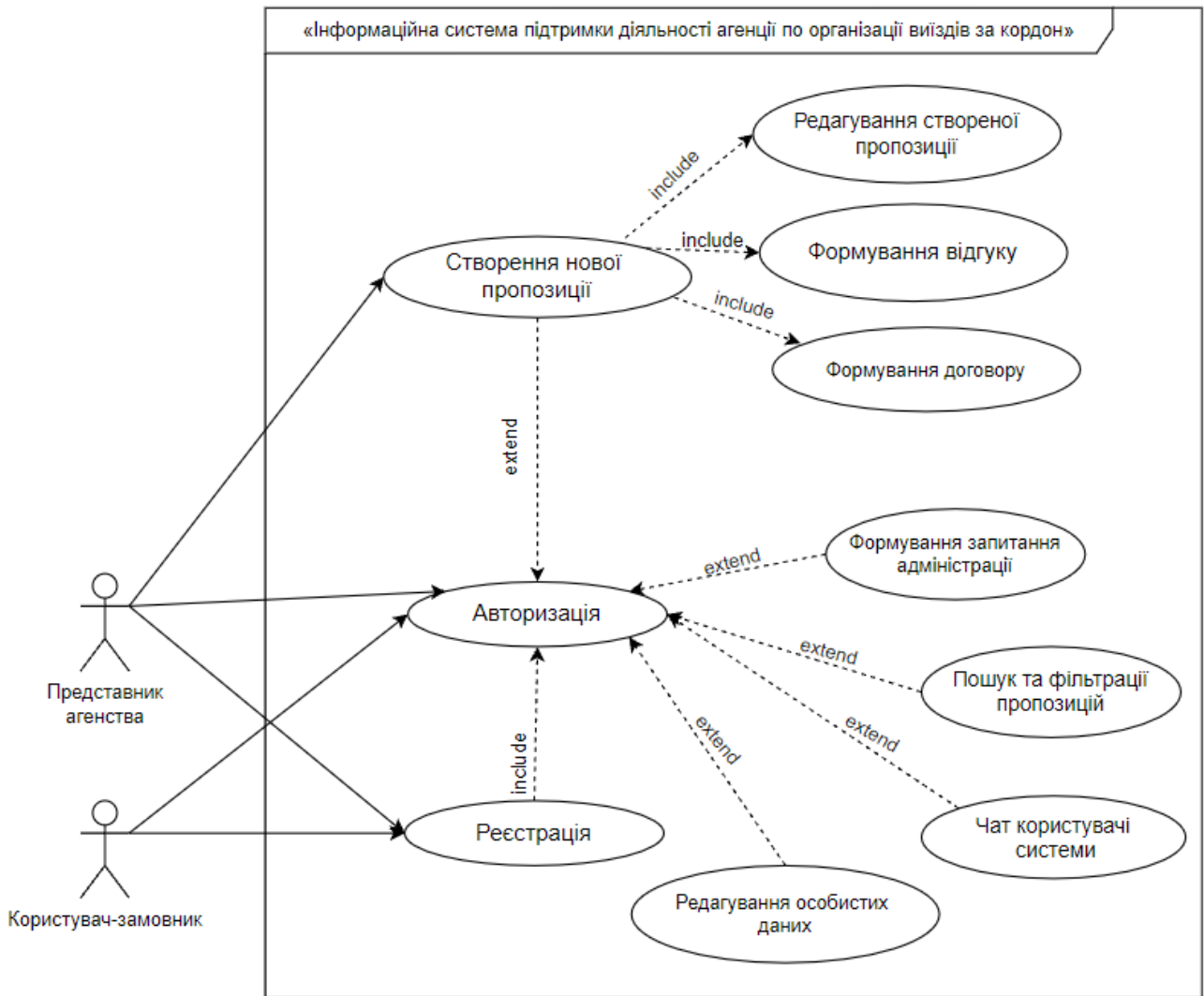


Рисунок 3.3 – Діаграма варіантів використання для користувачів системи



Рисунок 3.4 – Діаграма варіантів використання для адміністратора

3.4. Моделювання діаграм діяльності

Моделювання діаграми діяльності представляє з себе реалізацію та опис логіки процедури, бізнес-процесу чи потоків роботи. На рисунку 3.5-9 зображено діаграми діяльності модулів інформаційної системи підтримки діяльності агенції по організації виїздів за кордон.

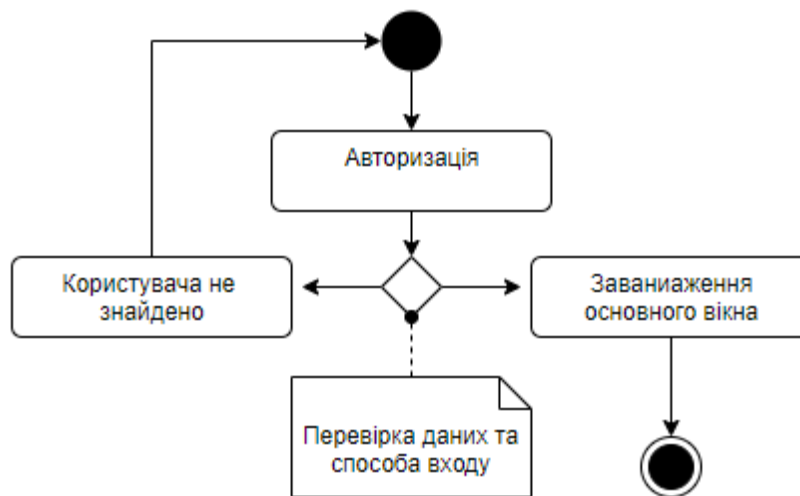


Рисунок 3.5 – Діаграма діяльності модулю авторизації

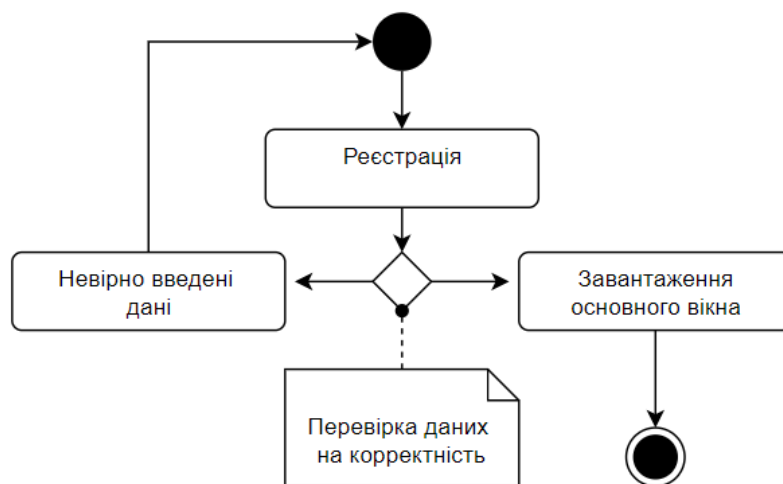


Рисунок 3.6 – Діаграма діяльності модулю реєстрації

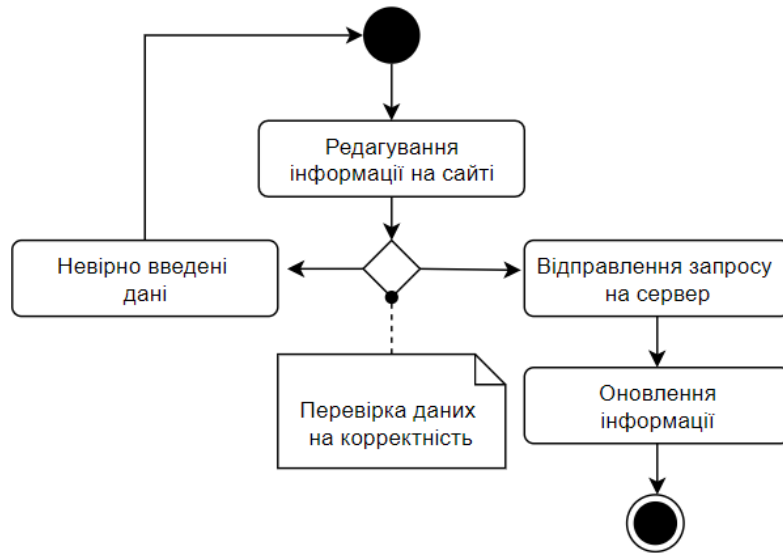


Рисунок 3.7 – Діаграма діяльності модулю редагування інформації



Рисунок 3.8 – Діаграма діяльності модулю перегляду інформації

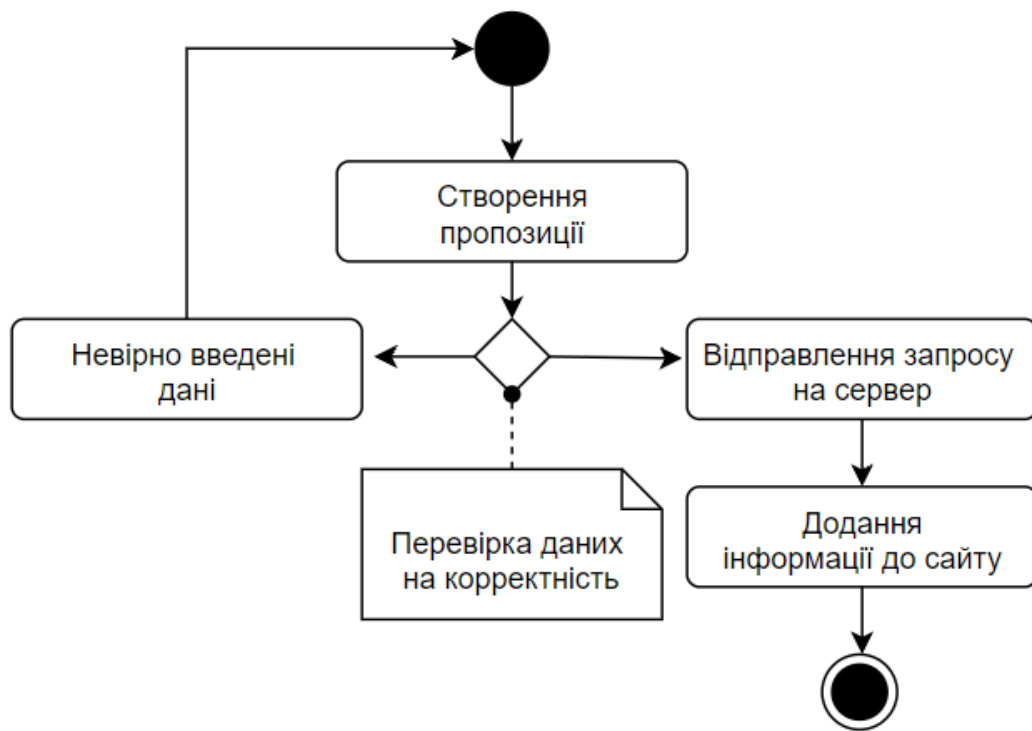


Рисунок 3.9 – Діаграма діяльності модулю створення пропозиції

3.5. Проектування бази даних

Основні користувачі системи – представник агентства та користувач-замовник. Віднайти компанію, що оформляє документи, підбирає та знаходять вакансію у час новітніх технологій зазвичай не проблема. Головним фактором недовіри, зі сторони клієнтів – відсутність гарантії надання якісних послуг [28-29].

Отже, інформаційна система використовуватиметься як відкрита платформа для налаштування взаємодії користувачів із компаніями за декількома напрямками, а саме освіта, робота та туризм.

Отже, було створено такі сутності як:

- user: загальна інформація про користувача, така як ім'я, прізвище, вік, стать та інше.
- type: напрямок роботи (робота/навчання/туризм);
- user_role: роль користувача у системи (тип акаунту);
- country: переліком країн;
- media: додаткова контактна інформація;

- message: повідомлення користувача;
- chat: створений чат;
- message_has_user: чат між двома користувачами;
- order: створена пропозиція з описом, ціною за часом виконання
- order_status: статус оформлення чи виконання пропозиції;
- user_has_order: оформлення пропозиції між представником агентства та користувачем-замовником;
- feedback: відгук по виконанню роботи;
- new: тематичні новини в інформаційній системі;
- question: запитання від користувачів;
- info: інформація на сайті.

Зобразимо відносини в ER-діаграмі (рис.3.10). Проаналізувавши сутність, використовувані в моделі інформаційної системи, перейдемо до реалізації структури БД. Для цього представимо імена необхідних таблиць, атрибутів, типів, їх призначення та обмеження в табл.3.6.

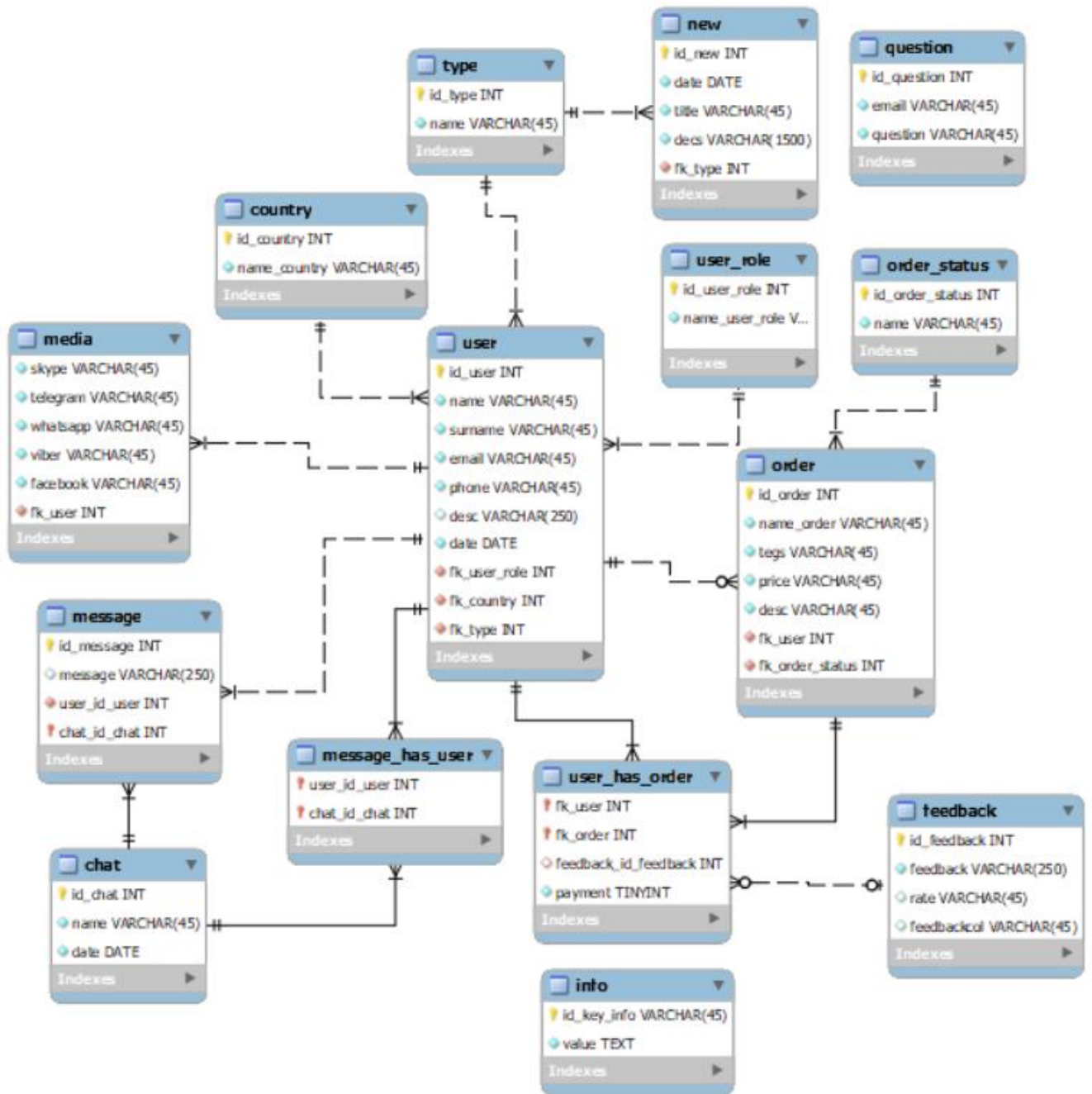


Рисунок 3.10 – ER-діаграма

Таблиця 3.6 – Інформація за таблицями ER-діаграми інформаційної системи підтримки роботи агентства

№	Назва таблиці	Поле	Зміст	Тип поля	Ключі	Обмеження
1	user	id_user	Ідентифікатор користувача	INTEGER	PK	Не порожній
		name	Ім'я користувача	VARCHAR(45)		Не порожній
		surname	Прізвище користувача	VARCHAR(45)		Не порожній
		phone	Номер телефону користувача	VARCHAR(45)		Не порожній
		desc	Короткий опис про користувача	TEXT		
		email	Електронна пошта користувача	VARCHAR(45)		Не порожній
		date	Дата реєстрації	DATE		Не порожній
		fk_type	Ідентифікатор типу тематики	INTEGER	FK	Не порожній
		fk_country	Ідентифікатор назви країни	INTEGER	FK	Не порожній
2	type	id_type	Ідентифікатор типу тематики	INTEGER	PK	Не порожній
		name	Назва типу тематики	VARCHAR(45)		Не порожній
3	new	id_new	Ідентифікатор новини	INTEGER	PK	Не порожній
		date	Дата створення новини	DATE		Не порожній
		title	Назва/заголовок новини	VARCHAR(100)		Не порожній
		desc	Основна інформація новини	TEXT		Не порожній

Продовження таблиці 3.6

№	Назва таблиці	Поле	Зміст	Тип поля	Ключі	Обмеження
3	new	fk_type	Ідентифікатор типу тематики	INTEGER	FK	Не порожній
4	country	id_country	Ідентифікатор країни	INTEGER	PK	Не порожній
		name	Назва країни	VARCHAR(45)		Не порожній
5	media	fk_user	Ідентифікатор користувача	INTEGER	FK	Не порожній
		skype	Посилання на «Skype»	VARCHAR(100)		
		telegram	Посилання на «telegram»	VARCHAR(100)		
		whatsapp	Посилання на «WhatsApp»	VARCHAR(100)		
		viber	Посилання на «Viber»	VARCHAR(100)		
		facebook	Посилання на «Facebook»	VARCHAR(100)		
6	user_role	id_user_role	Ідентифікатор ролі користувача в інформаційній системі	INTEGER	PK	Не порожній
		name	Назва ролі користувача	VARCHAR(45)		Не порожній
7	question	id_question	Ідентифікатор запитання	INTEGER	PK	Не порожній
		email	Електронна пошта для зв'язку	VARCHAR(45)		Не порожній
		question	Запитання користувача	VARCHAR(500)		Не порожній

Продовження таблиці 3.6

№	Назва таблиці	Поле	Зміст	Тип поля	Ключі	Обмеження
8	message	id_message	Ідентифікатор повідомлення	INTEGER	PK	Не порожній
		message	Текст повідомлення	TEXT		Не порожній
		fk_user	Ідентифікатор користувача	INTEGER	FK	Не порожній
		fk_chat	Ідентифікатор чату	INTEGER	FK	Не порожній
9	chat	id_chat	Ідентифікатор чату	INTEGER	PK	Не порожній
		name	Назва чату	VARCHAR(250)		Не порожній
		date	Дата створення чату	DATE		Не порожній
10	message_has_user	fk_user	Ідентифікатор користувача	INTEGER	FK	Не порожній
		fk_message	Ідентифікатор повідомлення	INTEGER	FK	Не порожній
11	order	id_order	Ідентифікатор створеної пропозиції	INTEGER	PK	Не порожній
		name	Назва пропозиції	VARCHAR(250)		Не порожній
		tags	Теги для пошуку пропозиції	ENUM		Не порожній
		price	Ціна співпраці за даною пропозицією	INTEGER		Не порожній
		desc	Опис пропозиції	TEXT		Не порожній
		fk_user	Ідентифікатор користувача	INTEGER	FK	Не порожній

Продовження таблиці 3.6

№	Назва таблиці	Поле	Зміст	Тип поля	Ключі	Обмеження
11	order	fk_order_state	Ідентифікатор стану пропозиції	INTEGER	FK	Не порожній
12	order_state	id_order_state	Ідентифікатор стану пропозиції	INTEGER	PK	Не порожній
		name	Назва стану пропозиції	VARCHAR(45)		Не порожній
13	user_has_order	fk_user	Ідентифікатор користувача	INTEGER	PK	Не порожній
		fk_order	Ідентифікатор створеної пропозиції	INTEGER	FK	Не порожній
		fk_feedback	Ідентифікатор відгуку після виконання умов пропозиції	INTEGER	FK	Не порожній
		payment	Статус оплати угоди виконання пропозиції	BOOLEAN		Не порожній
14	feedback	id_feedback	Ідентифікатор відгуку після виконання умов пропозиції	INTEGER	PK	Не порожній
		feedback	Текст відгуку про співпрацю	VARCHAR(500)		Не порожній
		rate	Оцінювання за декількома категоріями	ENUM		Не порожній

Продовження таблиці 3.6

№	Назва таблиці	Поле	Зміст	Тип поля	Ключі	Обмеження
15	info	Id_key_info	Ключ запису інформації	VARCHAR(100)	PK	Не порожній
		value	Інформація для запису	TEXT		Не порожній

4 ПРАКТИЧНА РЕАЛІЗАЦІЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

4.1 Програмна реалізація

Програмна реалізація інформаційної системи підтримки діяльності агенції по організації виїздів за кордон була виконана за допомогою:

- Vue.js, що використовується для реалізації візуальної частини;
- PHP, а саме фреймворк Laravel, для реалізацію функціональної частини.

Розглянемо структуру на рисунку 4.1 та детальний опис кожного із пакетів в таблиці 4.1

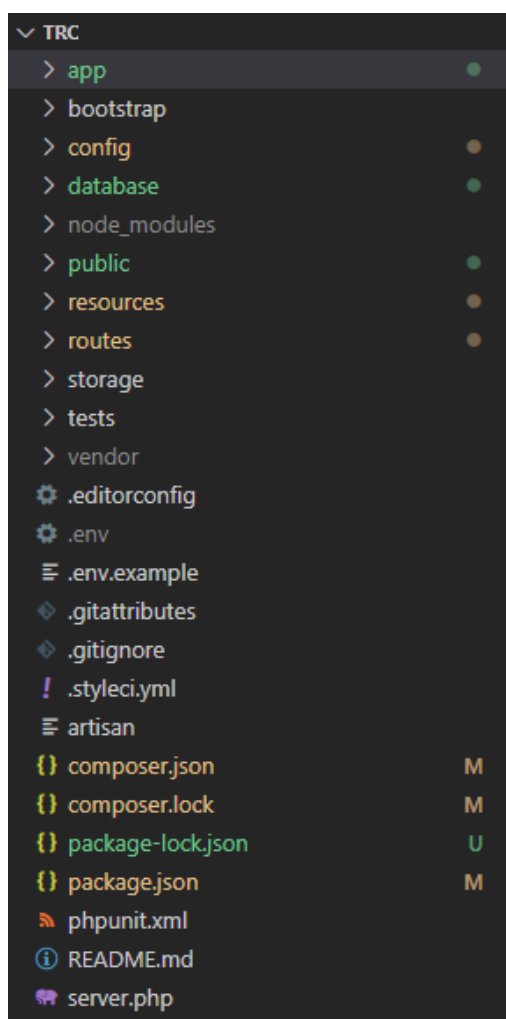


Рисунок 4.1 – Структура проекту

Таблиця 4.1 – Опис структури проекту

№	Назва	Опис
1	«app»	Код ядра Laravel-додатку.
2	«bootstrap»	Файли, які завантажують розширюючі можливості Vue.js, а саме VueBootstrap.
3	«config»	Конфігураційні файли додатку.
4	«database»	Міграції моделей бази даних та класи для наповнення БД даними.
5	«public»	Вхідною точкою для всіх запитів та містить всі ресурси.
6	«resources»	Некомпільовані ресурси, наприклад SASS, LESS, JavaScript.
7	«routes»	Налаштовані маршрути додатку.
8	«storage»	Містить скомпільовані шаблони, файл-кешу, файл-сесії та інші файли, що були створені Laravel.

Отже, результатом є підготовлена структура проекту із всіма потрібними шаблонами, файлами та налаштуваннями.

4.2 Використання інформаційної системи зі сторони представника агентства

Для використання інформаційної системи підтримки діяльності агенції по організації виїздів за кордон потрібно перейти на сторінку travel-work.zzz.com.ua. Перша сторінка на яку потрапляє користувач (рис.4.2) є сторінкою із рекламою та короткою інформацією. Також вона є головною сторінкою для незареєстрованих користувачів.

На даній сторінці також знаходиться також переадресування на сторінки реєстрації та авторизації у шапці сторінці.

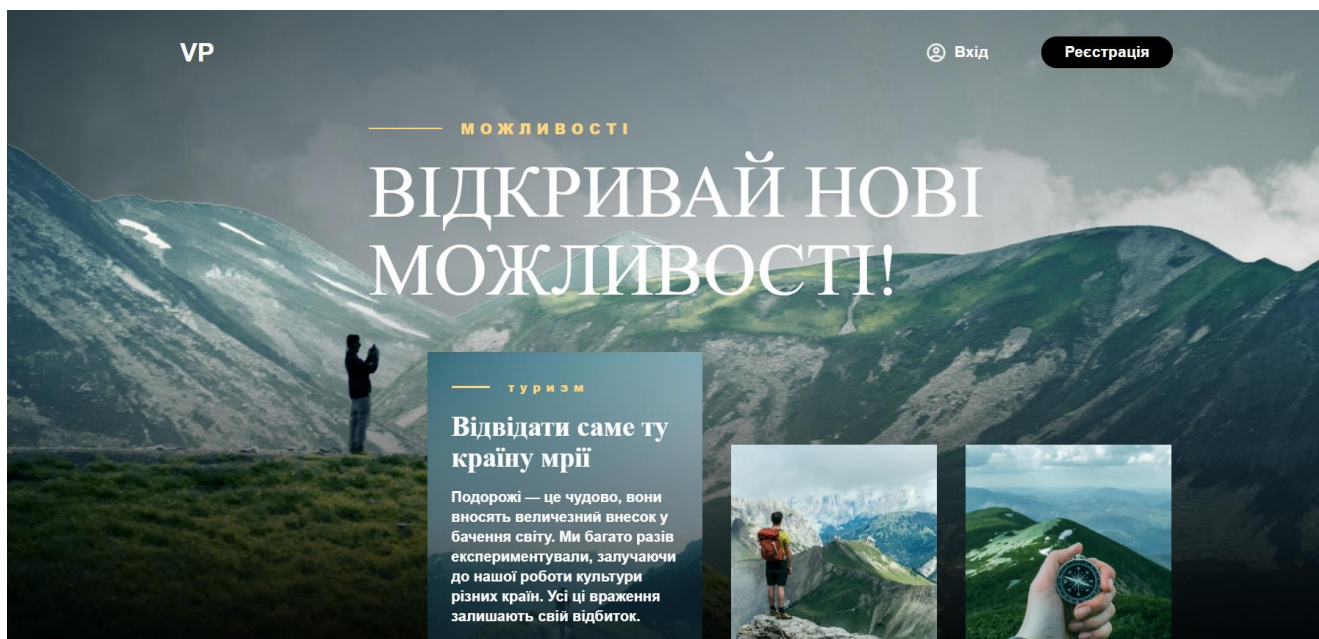


Рисунок 4.2 – Головна сторінка для незареєстрованих користувачів

На сторінці реєстрації користувач повинен заповнити такі поля як:

- Ім'я;
- Прізвище;
- Дата народження;
- Стать;
- Тип акаунту;
- Електронна пошта;
- Пароль.

Всі поля є обов'язковими. Після заповнення всіх форм користувач повинен натиснути кнопку «Зареєструватися».

Після реєстрації користувач перенаправляється на сторінки авторизації (рис.4.4). На сторінці авторизації користувач повинен ввести електронну пошту та пароль, що були введені при реєстрації.

ВІТАЄМО!Вже зареєстрований? [Вхід](#)

Ім'я *

Прізвище *

Дата народження *

Стать *

 Чоловіча Жіноча

Тип акаунту *

 Представник гоства Користувач

Пошта

Пароль

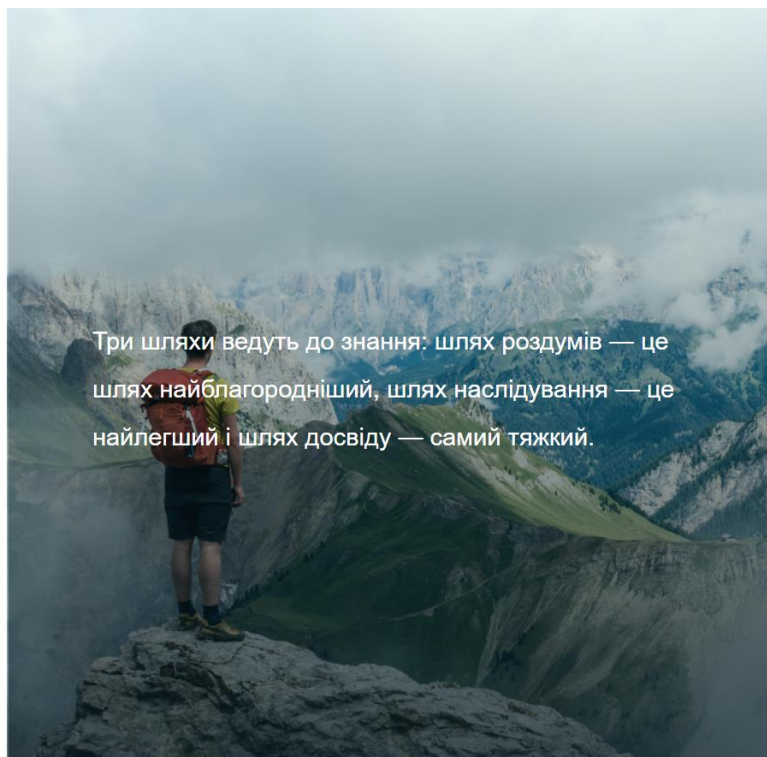
Зареєструватися!

Рисунок 4.3 – Сторінка реєстрації

ВІТАЄМО!Ще не зареєстрований? [Зареєструватися](#)

Пошта

Пароль

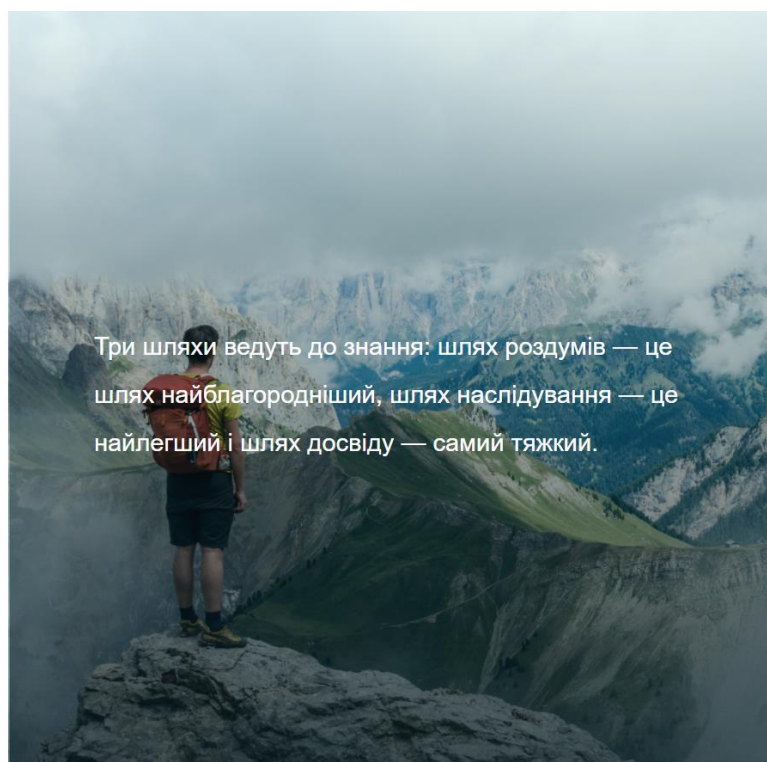
Вхід

Рисунок 4.4 – Сторінка авторизації

Після реєстрації та авторизації користувач перенаправляється на сторінку для редагування загальної інформації (рис.4.5). Користувач не матиме можливість використовувати інші функції поки не пройде верифікацію.

Користувач повинен заповнити такі обов'язкові поля:

- Фото користувача;
- Номер телефону;
- Країна;
- Місто;
- Коротка інформація.

Також для користувачів типу «представник агентства» є обов'язковими поля:

- Назва компанії;
- Кількість працівників в компанії;

Додатковими полями є посилання на соціальні мережі, такі як Skype, Viber, Telegram, WhatsApp та Facebook. Для користувачів типу «представник агентства» є додатковим полем – посилання на інформаційну сторінку агентства.

The screenshot shows the 'Редагувати профіль' (Edit Profile) page. At the top, there is a navigation bar with 'VP' and links for 'Туризм', 'Працевлаштування', and 'Навчання'. The user's name 'Катерина Шевченко' is displayed in the top right. Below the navigation bar, there are links for 'Список можливостей', 'Мої замовлення', 'Мої запити', 'Повідомлення', 'Агенства', 'Користувачі', 'Новини тематики', and 'Задати запитання'. The language is set to 'UA'.

The main content area is titled 'Редагувати профіль'. It features a profile picture placeholder with a 'Завантажити фото' (Upload photo) button. To the right of the photo are input fields for 'Ім'я*' (Name), 'Прізвище*' (Surname), and 'Електронна пошта*' (Email). Below these are fields for 'Номер телефону*' (Phone number), 'Дата народження*' (Date of birth), 'Стать*' (Gender), 'Країна*' (Country), and 'Місто*' (City).

Field	Value
Ім'я*	Катерина
Прізвище*	Шевченко
Електронна пошта*	kate@gmail.com
Номер телефону*	+380508395772
Дата народження*	05.12.1993
Стать*	Жіноча
Країна*	Україна
Місто*	Київ

Рисунок 4.5 – Сторінка редагування інформації користувача

Крім того, при завантаженні фотографії користувач має можливість виконати редагування фото та обрати область, що буде відображатися (рис.4.6).

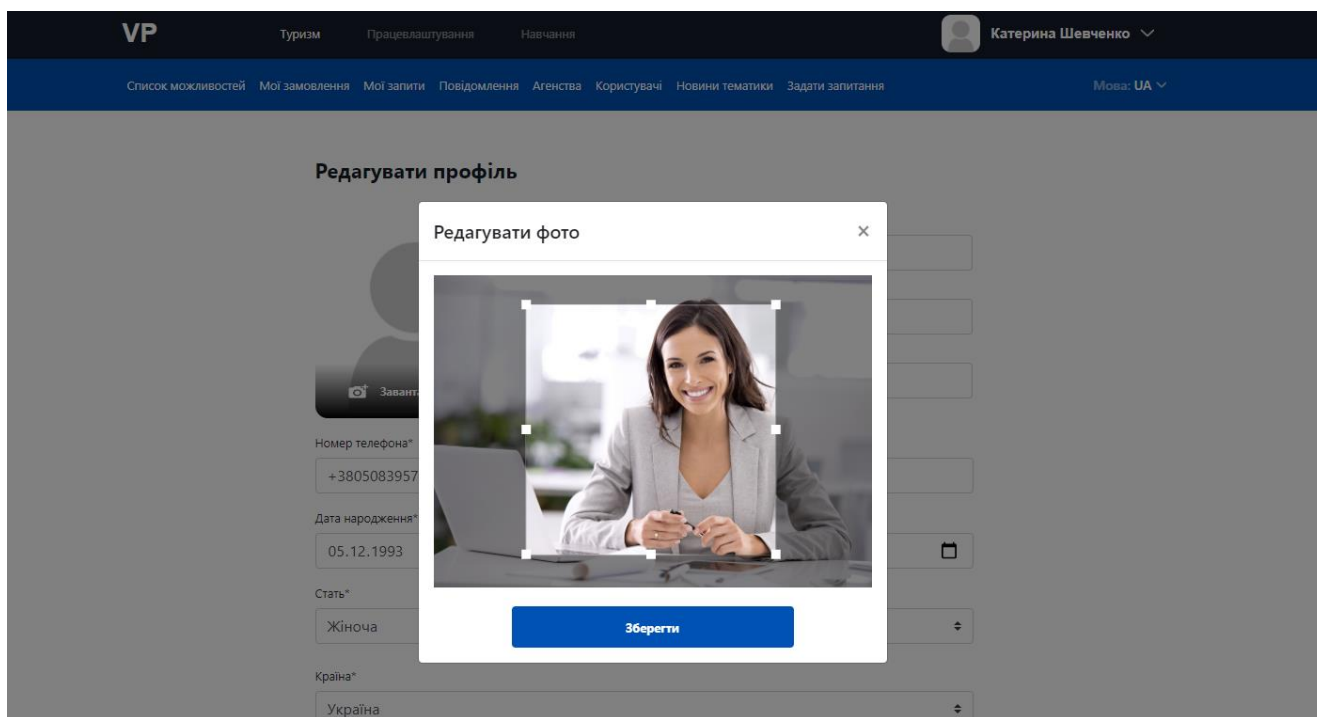


Рисунок 4.6 – Редагування фото

Після редагування фото та збереження доданої інформації на сайт сторінка перезавантажується та на хедері відображається завантажене фото користувача (рис.4.7).

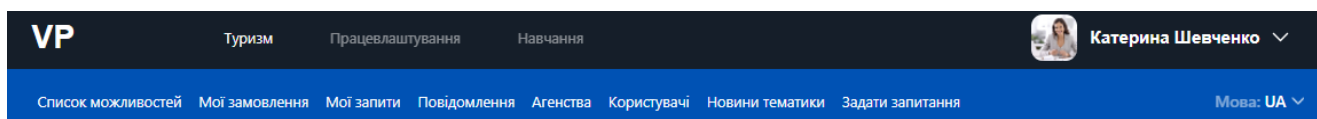


Рисунок 4.7 – Хедер інформаційної сторінки

Відтепер користувачеві відкриваються певні можливості. По-перше, представник агентства має можливість створювати замовлення за напрямком своєї компанії.

Для перегляду всіх пропозицій представнику агентства потрібно перейти на сторінку «Мої замовлення». На даній сторінці знаходиться таблиця із усіма створеними замовленнями. Для додавання нової можливості потрібно натиснути на кнопку «Додати» (рис.4.8).

The screenshot displays the 'Мої замовлення' (My orders) page. At the top, there is a navigation bar with the VP logo and menu items: Туризм, Працевлаштування, Навчання. The user's name, Катерина Шевченко, is shown in the top right. Below the navigation bar, there is a secondary menu with options: Список можливостей, Мої замовлення, Мої запити, Повідомлення, Агенства, Користувачі, Новини тематики, Задати запитання. The main content area features a table with the following columns: Заголовок, Теги, Країна, Ціна, and Опис. A green '+ Додати' button is located in the top right corner of the table. The table contains one entry for 'Международная школа STUDY Academy' with tags 'Освіта', 'Навчання', and 'Канада'. The description of this entry details a study program in Canada and the USA, mentioning the STUDY Academy and its focus on providing quality education and career preparation for Ukrainian students.

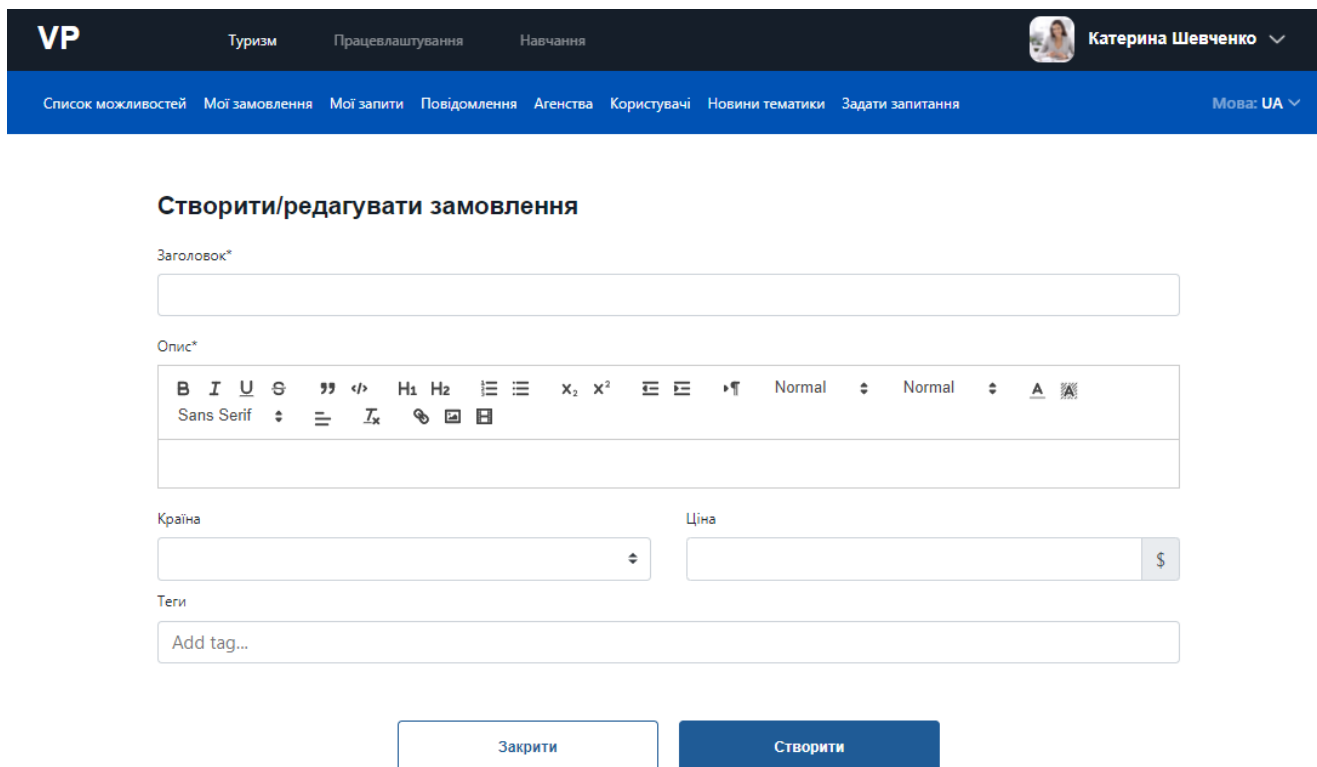
Рисунок 4.8 – Сторінка «Мої замовлення» представника агентства

При створенні нового замовлення представник агентства повинен заповнити такі поля:

- Заголовок пропозиції;
- Опис пропозиції;
- Країна;
- Ціна;
- Теги.

При створенні опису пропозиції представник агентства має можливість редагувати інформацію як у програмах для роботи з текстом. Крім того є можливість додавання фото до опису.

Додаткові теги можна використовувати для швидкого пошуку пропозицій за тими чи іншими напрямками (рис.4.9).



The screenshot shows a web interface for creating or editing an order. At the top is a dark blue header with the 'VP' logo on the left and the user's name 'Катерина Шевченко' with a dropdown arrow on the right. Below the header is a navigation bar with links: 'Список можливостей', 'Мої замовлення', 'Мої запити', 'Повідомлення', 'Агенства', 'Користувачі', 'Новини тематики', and 'Задати запитання'. On the far right of this bar is 'Мова: UA' with a dropdown arrow.

The main content area is titled 'Створити/редагувати замовлення'. It contains the following fields and controls:

- Заголовок***: A text input field.
- Опис***: A rich text editor with a toolbar containing icons for bold (B), italic (I), underline (U), strikethrough (ABC), quote, code, heading 1 (H1), heading 2 (H2), list, indent, subscript (x₂), superscript (x²), link, unlink, and image. Below the toolbar is a text input field.
- Країна**: A dropdown menu.
- Ціна**: A text input field with a currency symbol (\$) on the right.
- Теги**: A text input field with the placeholder 'Add tag...'.

At the bottom of the form are two buttons: 'Закрити' (Close) and 'Створити' (Create).

Рисунок 4.9 – Сторінка для створення замовлення

Приклад заповнення представлений на рисунку 4.10-11. При заповненні та збереженні інформації представник агентства отримує повідомлення про успішне оновлення даних (рис.4.12).

Створити/редагувати замовлення

Заголовок*

Міжнародная школа STUDY Academy

Опис*

B I U **H1 H2** $x_2 x^2$ Normal Normal

Міжнародна школа STUDY Academy відкриває систему канадської та американської освіти для українських школярів. Ми об'єднали найважливіші напрямки у навчанні та особистісному розвитку, щоб українські школярі отримали підготовку на рівні акредитованих шкіл Канади та США.

Бажаєте здобути престижну міжнародну освіту? Бажаєте визначитися з майбутньою професією та продовжити навчання за кордоном? Ласкаво просимо до Міжнародної школи STUDY Academy!

Переваги навчання у Міжнародній школі STUDY ACADEMY:

Навчання за програмами шкіл Канади та США;
Отримання атестату про середню освіту від школи Канади або США паралельно з українським атестатом;
Підготовка до навчання у школах-пансіонах та університетах за кордоном;
Профорієнтація: усвідомлений вибір майбутньої професії;
Викладачі з України, Канади та США;
Школа поряд з вами: навчайтеся в одному з 11 навчальних центрів у Києві або онлайн;
Гарантоване зарахування до 330 університетів та 250 шкіл за кордоном.





Рисунок 4.10 – Приклад створення замовлення



Країна Ціна \$

Теги

Рисунок 4.11 – Приклад створення замовлення(2)

міжнародная школа STUDY Academy

Опис*

B I U **H1 H2** $x_2 x^2$ Normal Normal

Міжнародна школа STUDY Academy відкриває систему канадської та американської освіти для українських школярів. Ми об'єднали найважливіші напрямки у навчанні та особистісному розвитку, щоб українські школярі отримали підготовку на рівні акредитованих шкіл Канади та США.

Все ок!

Інформацію оновлено.

Рисунок 4.12 – Нотифікація про оновлення інформації

Перейдемо на сторінку «Мої запити». На даній сторінці знаходяться запити, що отримує представник агентства що до створених замовлень (рис.4.13).

Користувач	Замовлення	Країна	Ціна		
Victoria Ponomarenko	Тур на 3 дні до Бельгії	Бельгія	300	✉	Залишити відгук
test test	Тур на 3 дні до Бельгії	Бельгія	300	✉	Новий запит
Антон Кузьма	Тур на 3 дні до Бельгії	Бельгія	300	✉	Новий запит
Анна Прокопенко	Тур на 3 дні до Бельгії	Бельгія	300	✉	Оформлення договору
Софія Зотова	Тур на 3 дні до Бельгії	Бельгія	300	✉	Новий запит
Олег Антонов	Тур на 3 дні до Бельгії	Бельгія	300	✉	Новий запит

« < 1 > »

Рисунок 4.13 – Сторінка «Мої запити»

На сторінці отриманих запитів знаходиться таблиця в якій знаходиться:

- Користувач;
- Замовлення;
- Країна;
- Ціна;
- Іконка для написання повідомлення;
- Статус запиту у вигляді кнопки;
- Кнопка для видалення.

До етапів запити відноситься:

- Новий запит;
- Оформлення договорів;
- Виконання договору
- Залишити відгук

- Виконаний запит;
- Відхилений запит.

Візуальне відображення кожного із етапів представлений на рисунку 4.14.

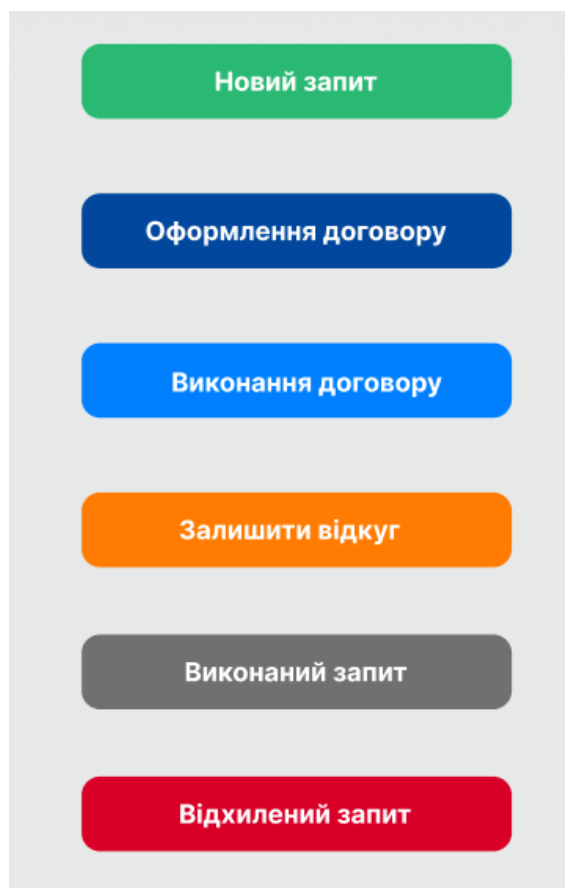


Рисунок 4.14 – Етапи запита

Розглянемо декілька прикладів умов переходу від одного етапу до іншого. Отримавши новий запит представник агентства повинен надіслати користувачеві додаткову інформацію (рис.4.15-16). Представник агентства має можливість додати приклад договору до запиту. Також потрібно вказати дедлайн виконання умов договору.

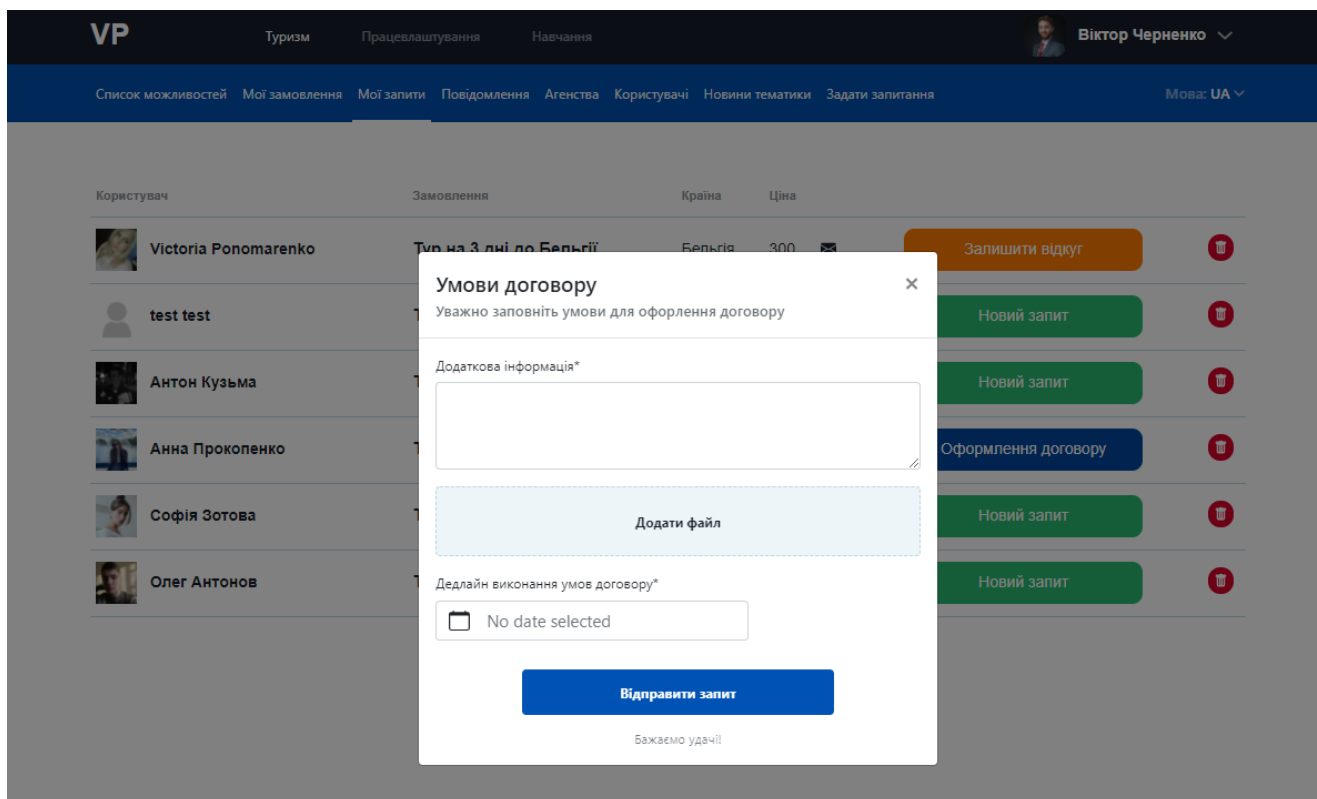


Рисунок 4.15 – Оформлення договору

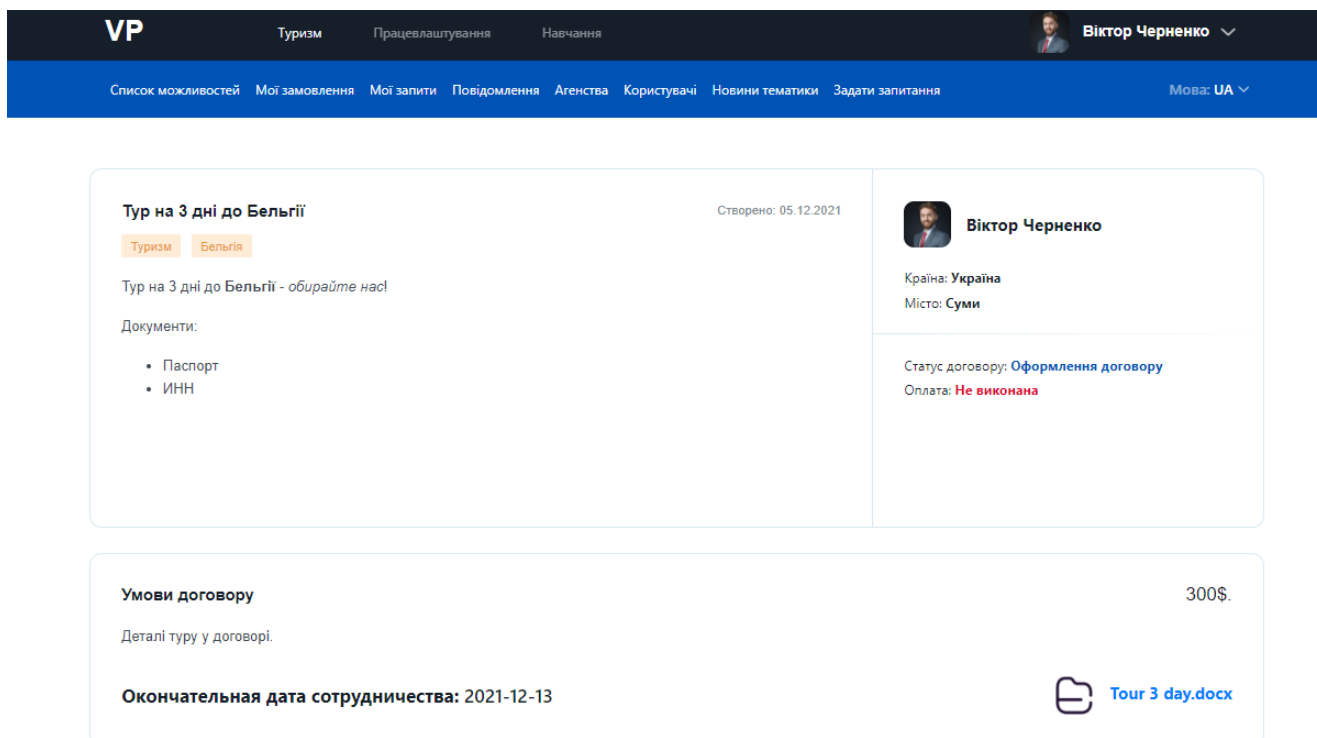


Рисунок 4.16 – Оформлення договору після надсилання умов

4.3 Використання інформаційної системи зі сторони користувача-замовника

Розглянемо використання інформаційної системи підтримки діяльності агенції по організації виїздів за кордон із сторони користувача замовника. На рисунку 4.17 представлена сторінка «Список можливостей». На даній сторінці будь-який зареєстрований та перевірений користувач має можливість виконувати пошук можливості.

Пошук можна виконувати:

- За назвою;
- Використовуючи фільтр;
- Окреме сортування;

Виконувати фільтрацію можна за тегами, країною та діапазоном ціни.

Рисунок 4.17 – Сторінка «Список можливостей»

Розглянемо декілька прикладів виконаної фільтрації на рисунку 4.18-19.

The screenshot shows the VP website interface. At the top, there is a navigation bar with the VP logo, menu items (Туризм, Працевлаштування, Навчання), a user profile (Victoria Ponomarenko), and a language selector (Мова: UA). Below the navigation bar is a green header with a search bar and a 'Пошук' button. The main content area is divided into a left sidebar with filters and a right main area with search results.

Filters (Left Sidebar):

- Теги:**
 - Бельгія (1)
 - Туризм (1)
 - Канада (2)
 - Навчання (3)
 - Освіта (3)
 - Англія (1)
- Країна:** (Dropdown menu)
- Ціна договору:**
 - Від: \$ 500 \$

Search Results (Right Main Area):

- Тур на 3 дні до Бельгії** (Створено: 05.12.2021)
- Країна: Бельгія
- Ціна: 300 \$
- Теги: Бельгія, Туризм
- Опис: Тур на 3 дні до Бельгії - обирайте нас! Документи: Паспорт/ІНН

Рисунок 4.18 – Сортування за датою та ціною

The screenshot shows the VP website interface, similar to the previous one, but with the search results sorted by price. The search bar and filters remain the same.

Filters (Left Sidebar):

- Теги:**
 - Бельгія (1)
 - Туризм (1)
 - Канада (2)
 - Навчання (3)
 - Освіта (3)
 - Англія (1)
- Країна:** (Dropdown menu)
- Ціна договору:**
 - Від: \$ До: \$

Search Results (Right Main Area):

- Вісшее образование в Великобритании** (Створено: 11.12.2021)
- Країна: Велика Британія
- Ціна: 1800 \$
- Теги: Англія, Навчання, Освіта
- Опис: Переваги здобуття вищої освіти у Великій Британії! Прямий вступ - економія 2-х років підготовки з новою програмою STUDY.UA. Випускники британських ВНЗ отримують дозвіл на працевлаштування протягом 2-х років після закінчення навчання. Британські університети співпрацюють з бізнес-компаніями та організаціями, що дає студентам можливість отримувати практичні навички у робочому середовищі. Британські студенти мають можливість офіційно працювати до 20 годин на тиждень під час навчання та повний робочий день під час канікул. Можливість навчання за програмою Sandwich: 1 рік стажування за фахом. Під час стажування студент не платить за навчання! STUDY.UA – офіційний представник в Україні. Гарантоване зарахування до британських ВУЗів для студентів підготовчих програм від STUDY.UA

- Международная школа STUDY Academy** (Створено: 11.12.2021)
- Країна: Канада
- Ціна: 1500 \$
- Теги: Канада, Навчання, Освіта
- Опис: Міжнародна школа STUDY Academy відкриває систему канадської та американської освіти для

Рисунок 4.19 – Сортування за підвищенням ціни

Користувач на сторінці «Мої запити» отримує інформацію що до зміни статусів запитів. Наприклад, на рисунку 4.20 представлено зміну статусу однієї із заяв із «Новий запит» на «Оформлення договору».

Агент	Замовлення	Країна	Ціна		
Віктор Черненко	Тур на 3 дні до Бельгії	Бельгія	300	✉	Оформлення договору
Катерина Шевченко	Міжнародна школа STUDY Academy	Канада	1500	✉	Новий запит

Рисунок 4.20 – Сторінка «Мої запити»

Відкривши запит ми також бачимо детальну інформацію що до статусу договору, оплати, інформації, тощо (рис.4.21).

Створено: 05.12.2021	Віктор Черненко Країна: Україна Місто: Суми
	Статус договору: Оформлення договору Оплата: Не виконана

Рисунок 4.21 – Статус запити

На даній сторінці користувач-замовник має можливість прийняти чи не прийняти умови отриманого договору після ознайомлення із повною інформацією. Після прийняття умов користувачеві стає доступною додаткове вікно із оплатою договору (рис.4.22).



Тур на 3 дні до Бельгії Створено: 05.12.2021 Туризм Бельгія Тур на 3 дні до Бельгії - <i>обирайте нас!</i> Документи: <ul style="list-style-type: none">ПаспортИНН	 Віктор Черненко Країна: Україна Місто: Суми Статус договору: Виконання договору Оплата: Не виконана
Умови договору Деталі туру у договорі. Окончателная дата сотрудничества: 2021-12-13	300\$.  Tour 3 day.docx
Прийняття умов договору: <input type="text" value="Прийнято"/> <input type="button" value="Прийняти"/>	Виконати переказ ціну договору: Кошти: <input type="text"/> <input type="button" value="Оплатити"/>

Рисунок 4.22 – Деталі оформлення договору

Після прийняття умов договору та переказ оплати за договір ці модулі стають заблоковані для змін для користувача (рис.4.23).

Тур на 3 дні до Бельгії


Створено: 05.12.2021

Туризм Бельгія

Тур на 3 дні до Бельгії - *обирайте нас!*

Документи:

- Паспорт
- ИНН

 **Віктор Черненко**


Країна: Україна
Місто: Суми

Статус договору: Виконання договору
Оплата: Виконана

Умови договору 300\$.

Деталі туру у договорі.

Окончателная дата сотрудничества: 2021-12-13

 [Tour 3 day.docx](#)

Прийняття умов договору:

Виконати переказ ціну договору:

Кошти:

Рисунок 4.23 – Результат прийняття умов та виконання оплати

У той же час статус договору та статус оплати замовлення змінюється на «Виконання договору» та «Виконана» відповідно (рис.4.24).

Тур на 3 дні до Бельгії


Створено: 05.12.2021


Бельгія Туризм

Тур на 3 дні до Бельгії - *обирайте нас!*

Документи:

- Паспорт
- ИНН



 **Віктор Черненко**

Країна: Україна
Місто: Суми

Статус договору: Залишити відкуг
Оплата: Виконана

Рисунок 4.24 – Змінення статусів

Кожна успішна угода закінчується відгуком. Представник агентства та користувач-замовник залишають відгуки про співпрацю (рис.4.25).

Умови договору 300\$
додаткова інформація про договір

Окончательная дата сотрудничества: 2021-12-06 [Ponomarenko Victoria_TZ \(in progress\).docx](#)

Прийняття умов договору:

Виконати переказ ціну договору: Кошти:

Залишити відгук про співпрацю

Своєчасність роботи	★★★★★	<p>Коментар*</p> <p>Все чудово! Рекомендую до співпраці</p> <p><input type="button" value="Відправити"/></p> <p><small>Заповнюйте відгук відповідно співпраці. Коментар є обов'язковим для заповнення. Перевірте заповнену інформацію, адже подальше редагування буде неможливим.</small></p>
Відповідність опису до отриманих послуг	★★★★☆	
Своєчасність роботи	★★★★★	
Відповідність опису до отриманих послуг	★★★★☆	




Рисунок 4.24 – Залишення відгуку

Кожен із залишених відгуків відображається на сторінці відповідного користувача. Приклад відгуку представлений на сторінці представника агентства на рисунку 4.25.

Крім того, із додаткових можливостей, будь-який користувач має можливість задати запитання адміністрації надіславши його на окремій сторінці «Задати запитання» (рис.4.26). Також користувач має можливість переглянути відповідь на найпопулярніші запитання.

VP Туризм Працевлаштування Навчання Віктор Черненко

Список можливостей Мої замовлення Мої запити Повідомлення Агенства Користувачі Новини тематики Задати запитання Мова: UA



Віктор Черненко

Країна Україна, Суми

Агенство ТопПросто

Розмір агенства 25


Предствник агенства

Номер телефону 380506858662

Електронна пошта victor.chern@gmail.com

Заголовок пропозиції	Теги	Країна	Ціна договору	Опис
Тур на 3 дні до Бельгії		Бельгія	300	Тур на 3 дні до Бельгії - обирайте нас!Документи: ПаспортИНН

Тур на 3 дні до Бельгії



Victoria Ponomarenko

Своєчасність роботи ★★★★★ Все чудово! Рекомендую до співпраці

Відповідність опису до отриманих послуг ★★★★☆

Своєчасність роботи ★★★★★

Відповідність опису до отриманих послуг ★★★★☆

Рисунок 4.25 – Відображення відгуку на сторінці

VP Туризм Працевлаштування Навчання Віктор Черненко

Список можливостей Мої замовлення Мої запити Повідомлення Агенства Користувачі Новини тематики Задати запитання Мова: UA

Запитання

Відправити

1 **Чи можна виконати оплату по за системою?** +

Катерина Шевченко: test1@gmail.com

Рисунок 4.26 – Сторінка «Задати запитання»

Крім того, в інформаційній системі реалізована функція перекладу на англійську мову (рис.4.27).

The screenshot shows a web application interface. At the top, there is a dark blue header with the logo 'VP' and navigation links: 'Tourism', 'Employment', and 'Teaching'. On the right side of the header, the user's name 'Victor Chernenko' is displayed with a dropdown arrow. Below the header is a blue navigation bar with links: 'List of possibilities', 'My orders', 'My requests', 'Message', 'Agencies', 'Users', 'News topics', and 'Ask questions'. On the far right of this bar, the language is set to 'EN'. The main content area features a white box with a 'Question' label and a large text input field. Below the input field is a blue 'Send' button. Below this, a light blue notification box contains a question: '1 Is it possible to make a payment under the system?' with a close button (X) on the right. The notification also shows the sender's name 'Katerina Shevchenko : test1@gmail.com' and the answer: 'Answer: No, payment must be made in the system.'

Рисунок 4.27 – Приклад запитання на англійській мові

4.4 Адміністрування інформаційної системи

Після авторизації адміністратор потрапляє на сторінку із редагування інформації (рис.4.28).

The screenshot shows the administrator's profile editing page. At the top, there is a dark blue header with the logo 'VP' and the user's name 'Admin Admin' with a dropdown arrow. Below the header is a red navigation bar with links: 'Замовлення', 'Користувачі', 'Новини', and 'Запитання'. On the far right of this bar, the language is set to 'UA'. The main content area is titled 'Редагувати профіль'. It features a profile picture placeholder with a camera icon and the text 'Завантажити фото'. To the right of the placeholder are three input fields: 'Ім'я*' (containing 'Admin'), 'Прізвище*' (containing 'Admin'), and 'Електронна пошта*' (containing 'admin@gmail.com'). Below these fields is a blue 'Зберегти зміни' button.

Рисунок 4.28 – Авторизація адміністратора

Адміністратор має такий перелік сторінок:

- Заовлення;
- Користувачі;
- Новини;
- Запитання;

На сторінці із заовленнями (рис.4.29) адміністратор має можливість виконувати фільтрацію за типом заовлення та сортувати за датою створення чи ціною.

The screenshot shows the VP admin interface. At the top, there is a navigation bar with the VP logo, a user profile for 'Admin Admin', and a language selector set to 'UA'. Below the navigation bar, there are menu items: 'Замовлення', 'Користувачі', 'Новини', and 'Запитання'. A search bar is present with a magnifying glass icon and a red 'Пошук' button. Below the search bar, there are two dropdown menus: 'Тип заовлення:' set to 'Всі' and 'Сортувати:' set to 'За датою'. The main content area displays a table of orders with the following columns: 'Користувач', 'Замовлення', 'Країна', and 'Ціна'. Each row includes a small profile picture, the user's name, the order details, the country, the price, and a red trash icon for deletion.

Користувач	Замовлення	Країна	Ціна
Віктор Черненко	Тур на 3 дні до Бельгії	Бельгія	300
Катерина Шевченко	Международная школа STUDY Academy	Канада	1500
Катерина Шевченко	Высшее образование в Великобритании	Велика Британія	1800
Катерина Шевченко	Міжнародна школа STUDY Academy	Канада	1700

At the bottom of the table, there is a pagination control showing a single page '1' with navigation arrows.

Рисунок 4.29 – Сторінка із заовленнями

На сторінці користувачів адміністратор має можливість переглядати основну інформацію та виконувати фільтрацію за типом акаунту (рис.4.30). На рисунку 4.31 представлено приклад виконаної фільтрації.

VP Admin Admin

Замовлення Користувачі Новини Запитання Мова: UA

Пошук

Тип акаунту: Всі

Прізвище та ім'я	Країна	Тип акаунту	Верифікація
Agency Agency		Представник агентства	Ні
test test	Алжир	Користувач	Так
Victoria Ponomarenko	Україна	Користувач	Так
Віктор Черненко	Україна	Представник агентства	Так
Катерина Шевченко	Україна	Представник агентства	Так

Рисунок 4.30 – Сторінка «Користувачі»

VP Admin Admin

Замовлення Користувачі Новини Запитання Мова: UA

Пошук

Тип акаунту: Агентство

Прізвище та ім'я	Країна	Тип акаунту	Верифікація
Agency Agency		Представник агентства	Ні
Віктор Черненко	Україна	Представник агентства	Так
Катерина Шевченко	Україна	Представник агентства	Так

« < 1 > »

Рисунок 4.31 – Приклад фільтрації на сторінці «Користувачі»

Наступною сторінкою є «Новини». На даній сторінці адміністратор має можливість додавати нові новини та редагувати уже створені за кожною із тем (рис.4.32).

На сторінці також реалізована функція фільтрація новин за тематикою та сортування за датою.

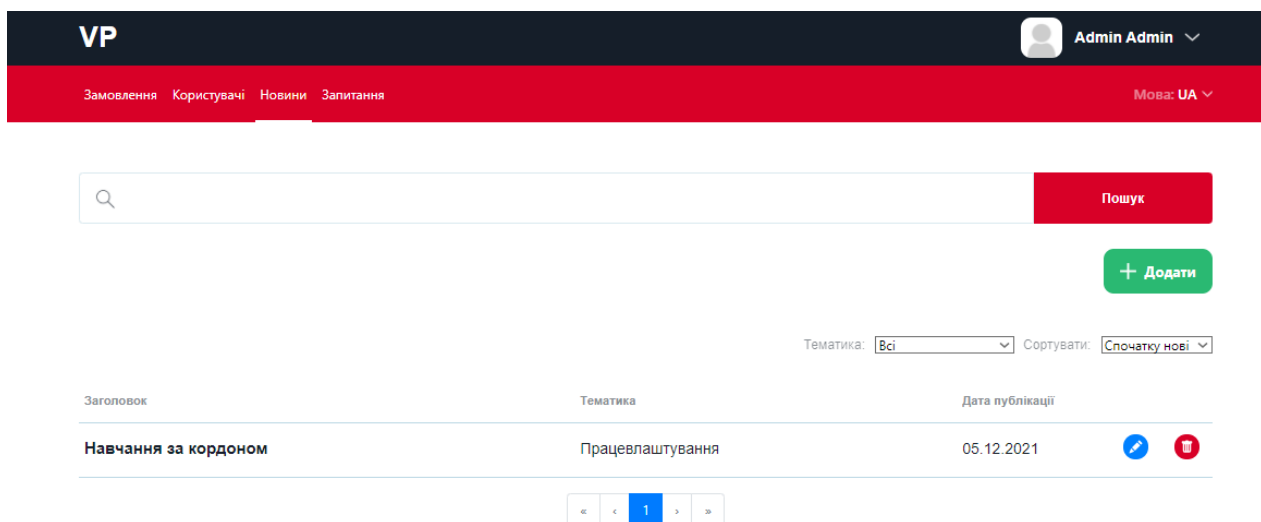


Рисунок 4.30 – Сторінка «Новини»

На рисунку 4.31 приставлені блоки, що є обов’язковими при створенні нової новини.

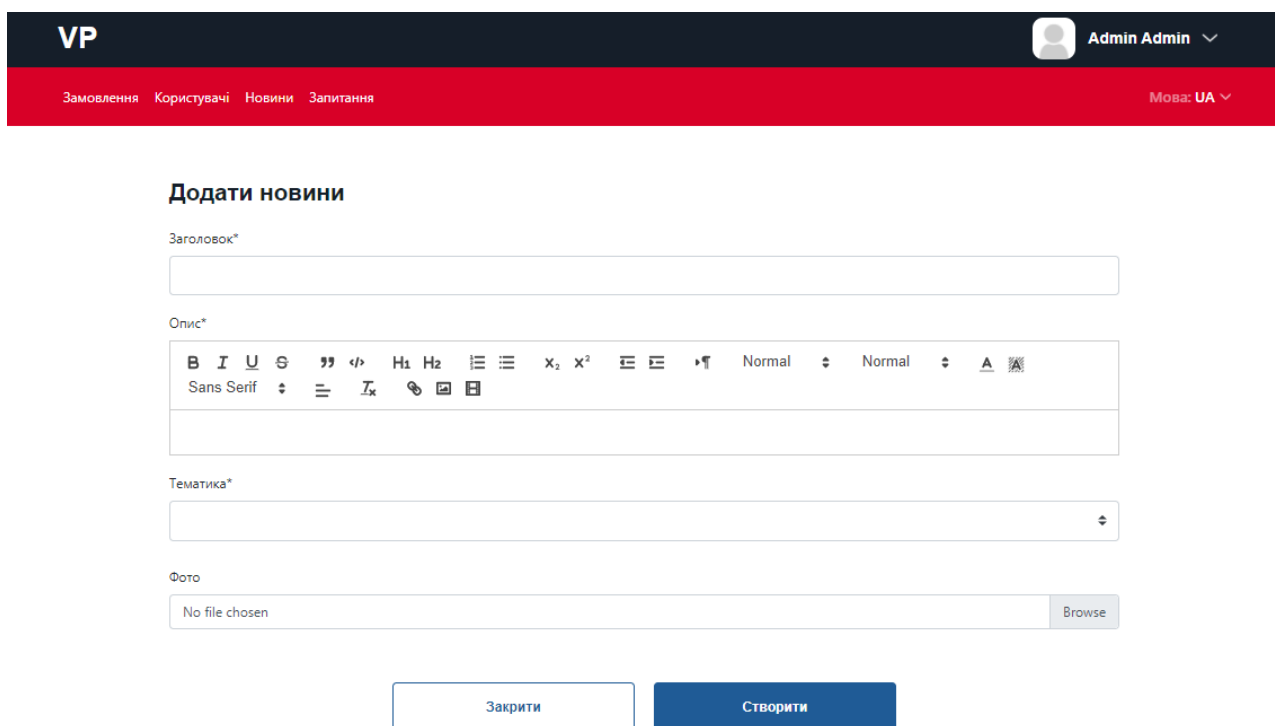


Рисунок 4.31 – Додавання новини

Останньою сторінкою для адміністрування є сторінка із запитаннями (рис.4.32). На дані сторінці адміністратор має можливість відповісти на запитання користувачів. Відповіді на запитання будуть відображатися на сторінці інформаційної системи. Приклад відображення на сайті представлено на рисунку 4.33.

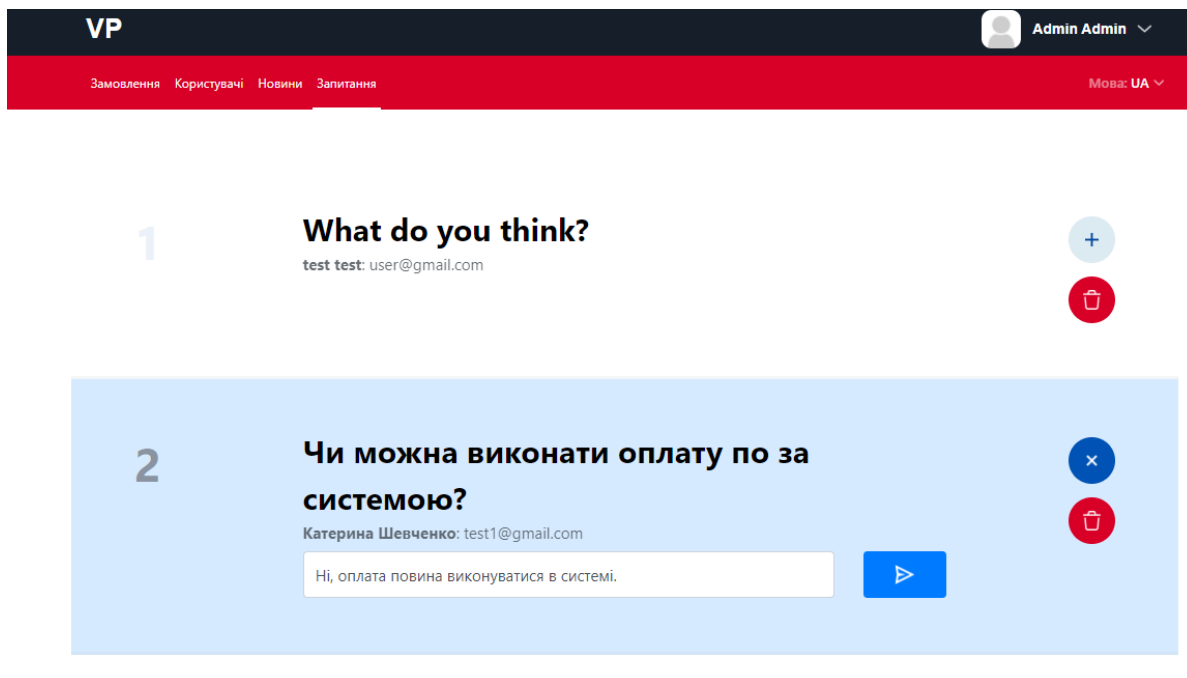


Рисунок 4.32 – Відповідь на запитання

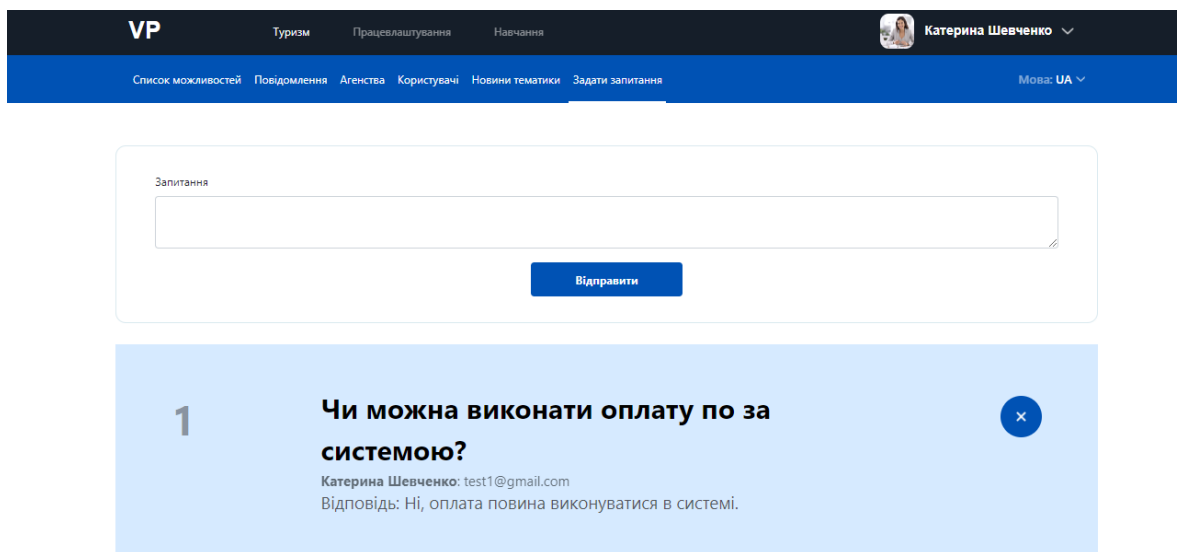


Рисунок 4.33 – Відображення відповіді на сторінці «Задати запитання»

ВИСНОВКИ

При виконанні дипломної роботи на тему «Інформаційна система підтримки діяльності агенції по організації виїздів за кордон» було проведено декілька етапів роботи.

На першому етапі було проведено детальний аналіз використання інформаційних систем при виїзді за кордон. Було також виконано аналіз сервісів із схожими функціональними можливостями.

На другому етапі було сформовано мету та задачі дослідження за темою розробки інформаційної системи. Даний розділ також включає в себе вибір засобів реалізації системи.

На третьому етапі було описано проектування інформаційної системи. Було спроектовано та розроблено діаграму IDEF0 та декомпозиція діаграми. Також було реалізовано діаграму варіантів використання відповідно до функціоналу інформаційної системи. Спроектовано база даних.

На останньому четвертому етапі було виконано реалізацію та розроблений детальний опис використання розробленої інформаційної система підтримки діяльності агенції по організації виїздів за кордон. Даний опис було виконано зі сторони усіх можливих типів користувачів, а саме користувача-замовника, представника агентства та адміністратора.

Результатом проведеної роботи є виконана інформаційна система підтримки діяльності агенції по організації виїздів за кордон.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. P. Yang, in *Lecture Notes on Data Engineering and Communications Technologies* (Springer Science and Business Media Deutschland GmbH, 2022), vol. 84, pp. 731–736.

2. TECHNOLOGY MANAGEMENT DEGREES AND CAREERS [Електронний ресурс] – Режим доступу до ресурсу: <https://www.allengineeringschools.com/engineering-careers/career/technology-management/> (дата звернення: 15.09.2021).

3. R. Godwin-Jones, Integrating technology into study abroad. *Language Learning and Technology*. 20, 1–20 (2016).

4. A. B. De Castro, N. Dyba, E. D. Cortez, G. G. Pe Benito, Collaborative Online International Learning to Prepare Students for Multicultural Work Environments. *Nurse Educator*. 44, E1–E5 (2019).

5. N. O'Connor, Using active learning strategies on travel and tourism higher education programmes in Ireland. *Journal of Hospitality, Leisure, Sport and Tourism Education*. 29 (2021), doi:10.1016/j.jhlste.2021.100326.

6. K. Jayasudha, B. Vidivelli, Analysis of major risks in construction projects. *ARNP Journal of Engineering and Applied Sciences*. 11, 6943–6950 (2016).

7. Мережа «Поїхали з нами» [Електронний ресурс] – Режим доступу до ресурсу: <https://www.pochalisnami.ua/tour> (дата звернення: 18.10.2021).

8. Агентство «Work Department» [Електронний ресурс] – Режим доступу до ресурсу: <https://workdepartment.pl/#kontakt> (дата звернення: 18.20.2021).

9. Агентство «InfoStudy» [Електронний ресурс] – Режим доступу до ресурсу: <https://infostudy-uk.com> (дата звернення: 18.10.2021).

10. Why is data validation important? [Електронний ресурс] – Режим доступу до ресурсу: <https://iterative.ly/blog/why-is-data-validation-important> (дата звернення: 28.10.2021).

11. INTEGRATION DEFINITION FUNCTION MODELING (IDEF0) . Draft Federal Information Processing Standards Publication 184 ,1993 Decembers 21

12. Sheer, A. ARIS-Business Process Modelling / Sheer A. – Springer-Verlag, Berlin,1998.

13. Booch, G. The Unified Modeling Language User Guide / Booch, G., Rumbaugh, J., Jacobson, I. – Second Edition, AddisonWesley, 2005.

14. Переваги ієрархічної Структури Робіт (WBS) для менеджерів ІТ проектів [Електронний ресурс] – Режим доступу до ресурсу: <https://habr.com/ru/post/327872/> (дата звернення: 18.11.2021).

15. INTEGRATION DEFINITION INFORMATION MODELING (IDEF1X), Draft Federal Information Processing Standards Publication 185 1993 Decembers 21.

16. Organization Breakdown Structure (OBS) [Електронний ресурс] – Режим доступу до ресурсу: <https://uplandsoftware.com/psa/resources/glossary/organization-breakdown-structure-obs/> (дата звернення: 04.10.2020).

17. Що таке діаграма Ганта? [Електронний ресурс] – Режим доступу до ресурсу: <https://www.atlassian.com/ru/agile/project-management/gantt-chart> (дата звернення: 18.11.2021).

18. План дій при управлінні ризиками проекту [Електронний ресурс] – Режим доступу до ресурсу: <http://projectimo.ru/planirovanie-proekta/setevoj-grafik.html> (дата звернення: 18.11.2021).

19. React: Making faster, smoother UIs for data-driven Web app [Електронний ресурс] – Режим доступу до ресурсу: <https://www.infoworld.com/article/2608181/react--making-faster--smoother-uis-for-data-driven-web-apps.html> (дата звернення: 28.11.2021).

20. DOM benchmark comparison of the front-end. JavaScript frameworks React, Angular, Vue, and Svelte [Електронний ресурс] – Режим доступу до ресурсу:https://www.doria.fi/bitstream/handle/10024/177433/levlin_mattias.pdf?sequence=2&isAllowed=y (дата звернення: 01.12.2021).

21. What is Vue.js? [Електронний ресурс] – Режим доступу до ресурсу: https://www.w3schools.com/whatis/whatis_vue.asp (дата звернення: 01.12.2021).

22. Angular [Електронний ресурс] – Режим доступу до ресурсу: <https://dev.to/siddharthshyniben/angular-pros-and-cons-m91> (дата звернення: 01.12.2021).

23. The Pros and Cons of Angular Development: All You Need to Know [Електронний ресурс] – Режим доступу до ресурсу: <https://www.brainvire.com/blog/pros-and-cons-of-angular-development/> (дата звернення: 01.12.2021).

24. Angular Development [Електронний ресурс] – Режим доступу до ресурсу: <https://exoft.net/angular-pros-and-cons/> (дата звернення: 01.12.2021).

25. WHY USE PHP? [Електронний ресурс] – Режим доступу до ресурсу: <https://light-it.net/blog/why-use-php-main-advantages-and-disadvantages/> (дата звернення: 01.12.2021).

26. The Good and the Bad of C# Programming [Електронний ресурс] – Режим доступу до ресурсу: <https://www.altexsoft.com/blog/c-sharp-pros-and-cons/> (дата звернення: 01.12.2021).

27. Insights into Ruby on Rails [Електронний ресурс] – Режим доступу до ресурсу: <https://www.codica.com/blog/pros-and-cons-of-ruby-on-rails-for-web-development/> (дата звернення: 01.12.2021).

28. Система управління базами даних. Посібник «Організація баз даних та баз знань»/ Укладач В.О. Нелюбов. – Київ: Редакційно-видавничій відділ, 2015.

29. Boonch, G. The Unified Modeling Language Inn User Guide / Bonch, G., Rumbaugh, J.– Second Edition, 2005.

30. Райс Ерік. Інформаційний та архітектурний підхід для створення успішних веб-сторінок. [Текст] / Ерік Райс. — Addison Wesley, 2001.

31. Круг Стів. Дизайн: або не змушуй мене думати [Текст] / Стів Круг. — Символ, 2001.

32. Методичні вказівки з курсу «Організація баз даних». Укладач В.О. Нелюбов, 2010.

ДОДАТОК А. ПЛАНУВАННЯ РОБІТ

1 ІДЕНТИФІКАЦІЯ МЕТИ ІТ-ПРОЕКТУ

На сьогоднішній день, проблема виїзду громадян за кордон є досить актуальною. Звичайно, існує безліч причин здійснення такого кроку. Кожен замислюється над тим, де провести відпустку чи про наявність отримати освіта за кордоном. Крім того, приводом виїзду можуть полягати в економічному чи соціальному становищі. Наприклад безробіття, відсутність вакансії за фахом, проблеми з отримання відповідних навичок та інше.

Віднайти компанію, що оформляє документи, підбирає та знаходять вакансію у час новітніх технологій зазвичай не проблема. Головним фактором недовіри, зі сторони клієнтів – відсутність гарантії надання якісних послуг.

Отже, метою проекту є інформаційна система яка створена як відкрита платформа для налаштування взаємодії користувачів із компаніями за декількома напрямками, а саме освіта, робота та туризм. Найважливіша відмінність – повноцінний контроль над оплатою послуг та зниження до мінімуму можливості шахрайства.

Розглянемо також методологію SMART. Дана техніка постановки цілей дозволяє досягти успіху в продажах, формує модель поведінки, виробляє стійкі правила життєвої позиції.

Розглянемо детальніше інформацію за кожним з пунктів у таблиці А.1.

Таблиця А.1 – Таблиця опису проекту за SMART методом

Specific (конкретна)	Розробка інформаційної системи, за допомогою якої можна буде виконувати поставлені цілі та задачі, знаходити виконавців.
Measurable (вимірювана)	Прогрес та якість буде вимірюватися в кількості користувачах, замовленнях та кількості позитивних відгуків.
Achievable (досяжна)	Данна інформаційна система буде розроблятися з використанням HTML, CSS, JavaScript, Vue.js та PHP, MySQL, Laravel.
Relevant (реалістична)	Для виконання проекту є всі необхідні засоби та знання.
Time-framed (обмежена у часі)	Для проекту визначений дедлайн та побудована Діаграма Ганта для відстеження часу виконання.

2 ПЛАНУВАННЯ ЗМІСТУ СТРУКТУРИ РОБІТ ІНФОРМАЦІЙНОЇ СИСТЕМИ

Для виконання планування змісту структури робіт інформаційної системи підтримки діяльності агенції по організації виїздів за кордон виконаємо проектування WBS діаграми. Метод розбивки робіт при розробці проекту представляє з себе декомпозиції основних завдань проекту на менші підзадачі. Обраний спосіб розділяти великі проекти допоможе виконати розробку інформаційної системи набагато швидше та ефективніше. Адже головна мета WBS діаграми є зробити процес реалізації більш керованим. Розбиття його на менші частини означає, що роботу можуть одночасно виконувати різні члени команди. Дане розбиття призводить до підвищення продуктивності команди. Загальна WBS діаграма представлена на рисунку А.1.

Також для детального представлення роботи над інформаційної системи було розроблено структуру розподілу організації. Дану структуру розподілу також називають OBS. Вона представляє з себе ієрархічна модель, що описує встановлені організаційні рамки для планування проектів, управління ресурсами, відстеження часу та витрат, розподілу витрат, звітування та управління роботою.

Загальна OBS структура представлена на рисунку А.2.

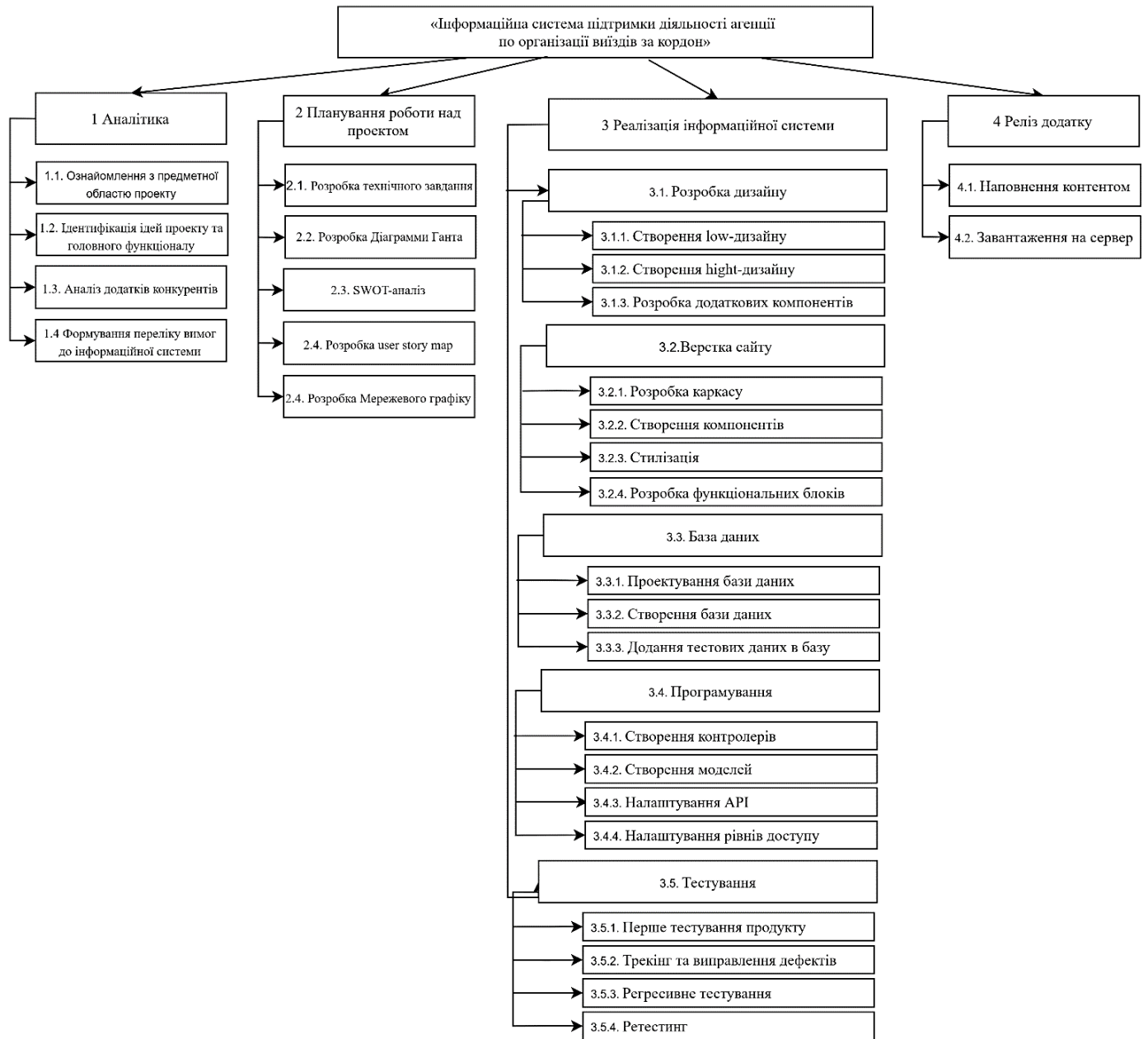


Рисунок А.1 – WBS структура інформаційної системи

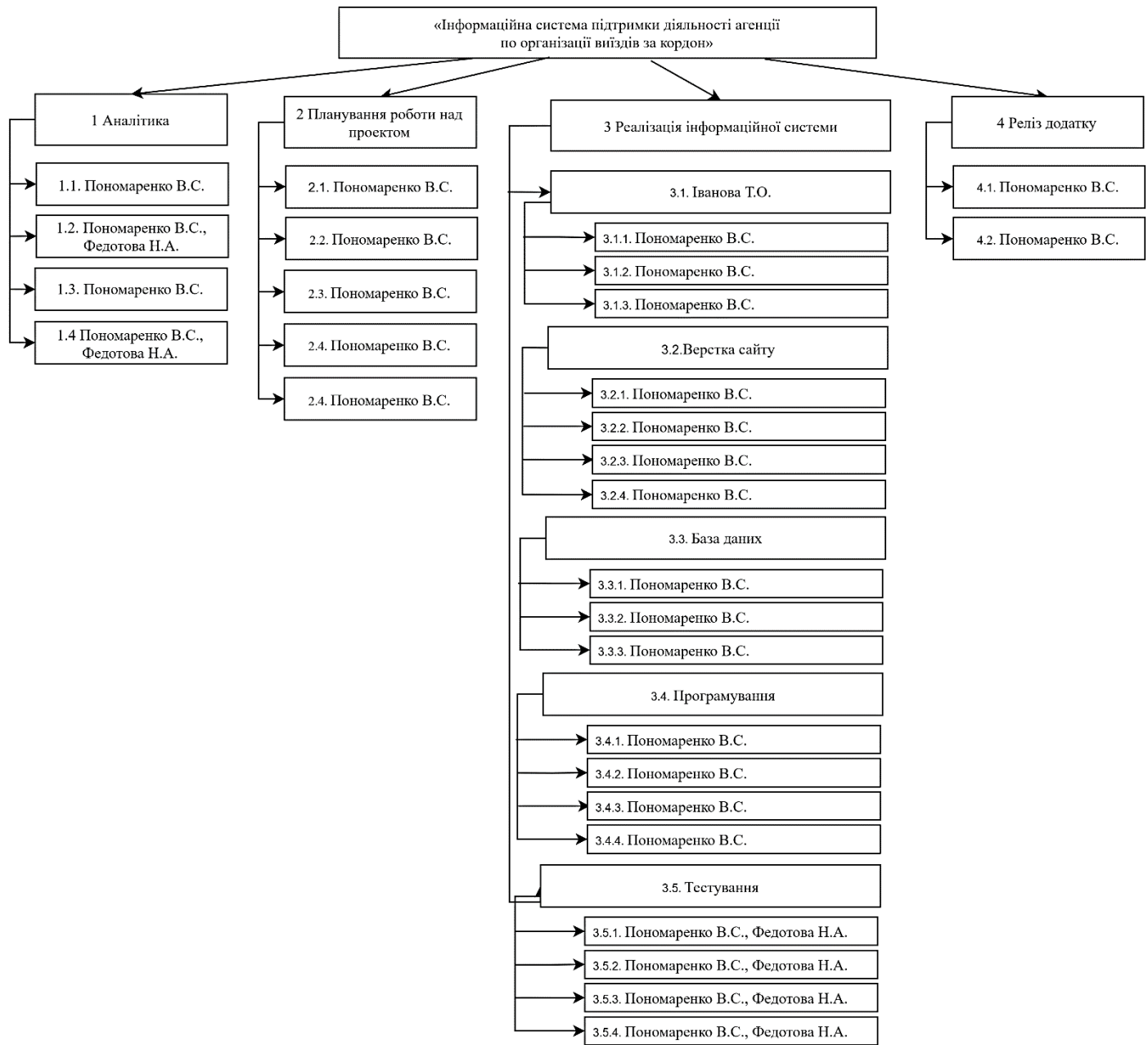


Рисунок А.2 – OBS структура інформаційної системи

3 ПОБУДОВА КАЛЕНДАРНОГО ГРАФІКУ ВИКОНАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

За допомогою діаграм Ганта керівники проектів і менеджери по продукту розбивають проекти на робочі завдання для зручності управління, підтримують порядок в роботі і роблять залежності між завданнями наочними.

Діаграми Ганта дозволяють спростити складові проекти. За допомогою цього засобу можна досить наочно і зручно для узагальнення представити велику кількість даних. Завдяки цій гістограмі велика кількість зацікавлених осіб, команд або їх учасників не стане проблемою для запису завдань, як і часті зміни обсягу роботи.

Ще одна перевага використання діаграми Ганта полягає в тому, що вона дає загальне уявлення про проект в цілому, в тому числі про всі контрольних точках і терміни виконання. Діаграму Ганта можна уявити як ефективний засіб раннього попередження. Розглянемо створену діаграму Ганта до інформаційної системи (рис. А.3-4).

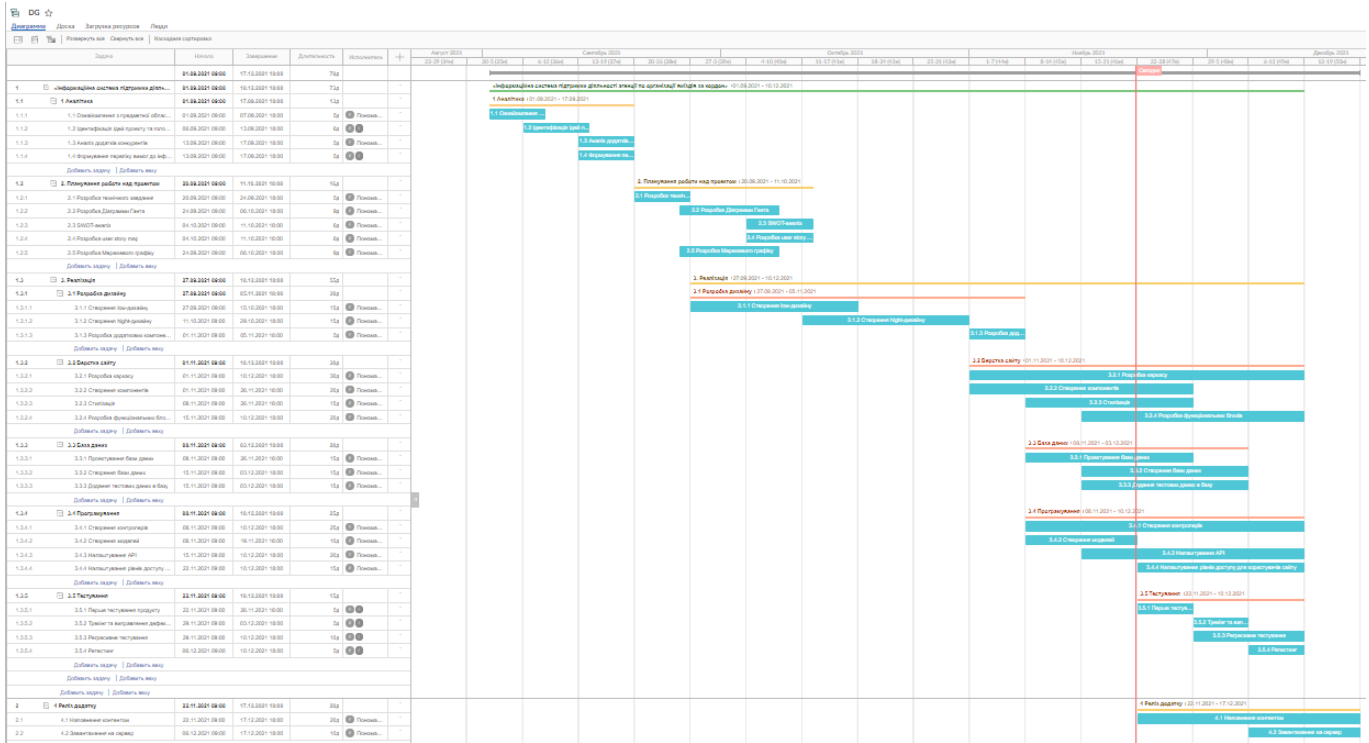


Рисунок А.3 – Діаграма Ганта (у тижнях)

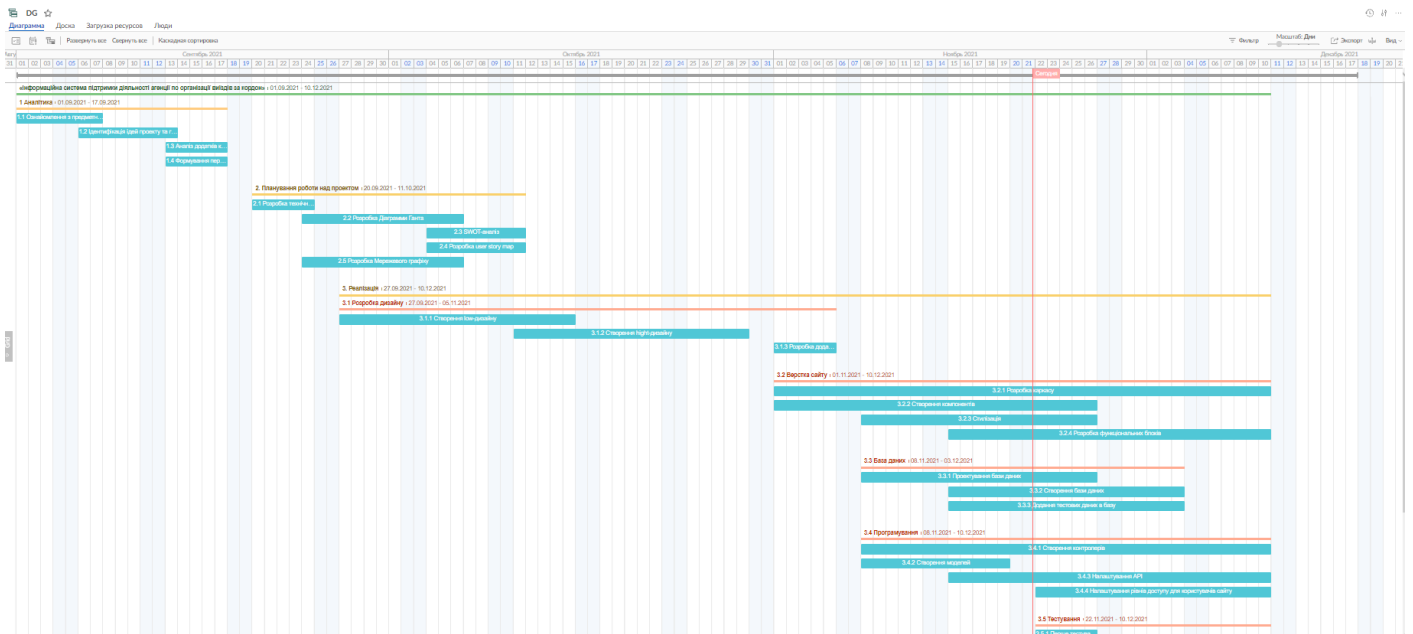


Рисунок А.4 – Діаграма Ганта (у днях)

4 ПЛАНУВАННЯ РИЗИКІВ ПРОЕКТУ

Ризиком є ймовірна подія, яка у випадку її виникнення може як негативно, так і позитивно вплинути на конкретний проект. Управління ризиком – це процес зміни ризиків та реагування на події під час виконання проекту. При реалізації проекту важливою частиною є моніторинг ризиків.

Отже, в даному випадку можна виділити деякі ризики. Виходячи цих оцінок можливо знайти ранг ризиків $R = P * L$, де:

- R – ранг ризику;
- P – ймовірність виникнення;
- L – ступінь впливу.

Шкала оцінки ризику може відповідати емпіричній шкалі оцінки ризику:

- 5 балів - критичний ризик (0,81 - 1);
- 4 бали - максимальний ризик (0,61 - 0,8);
- 3 бали - високий ризик (0,41 - 0,6);
- 2 бали - нормальний ризик (0,31 - 0,4);
- 1 бал - малий ризик (0 - 0,3).

Оцінки ризиків проекту наведено в табл. А.2 - 3.

Таблиця А.2 – Таблиця ризиків

№	Об'єкт ризику	Ризик	P	L	R
1	Час	Зміна пріоритету проекту чи вплив сторонніх факторів.	0,5	0,3	0,15
2	Якість	Системи, що не відповідають задачам , грубі помилки в алгоритмах процесів, критичні помилки в інформаційній системі.	0,1	0,4	0,04

№	Об'єкт ризику	Ризик	P	L	R
3	Бюджет	Поява додаткових незапланованих робіт по проекту.	0,7	0,2	0,14
4	Трудові ресурси та їх кваліфікація	Неможливість участі в запланованих роботах по проекту необхідних співробітників зі сторони замовника та виконавця у зв'язку з відпусткою, відрядженням та ін.	0,1	0,1	0,01
5	Інтеграція	Не вірна інформація, щодо зовнішніх систем, з яких передбачена взаємодія в межах проекту	0,5	0,4	0,2
6	Ринок	Розширення функціональних характеристик програмних продуктів, які використовувалися замовником в межах мети проекту	0,1	0,05	0,005
7	Бюджет	Зриви планів робіт; неправильна стратегія; персонал без потрібної кваліфікації; переоплати за роботу/матеріали; непогоджені частини проекту; невірний кошторис.	0,5	0,2	0,1

Таблиця А.3 – Матриця ризиків

№	Опис ризику	Вплив	Ймовірність	RV	Пом'якшення наслідків
1	Нестабільність Програмного забезпечення	5	3	С	Використовувати перевірене та якісне програмне забезпечення.

№	Опис ризику	Вплив	Ймовірність	RV	Пом'якшення наслідків
2	Некоректний функціонал інформаційної системи	4	2	М	Тестування інформаційної системи.
3	Неактуальність інформаційної системи	3	2	М	Використовувати найновіші тренди та сучасний підхід.
4	Додаткові вимоги	4	3	М	Обговорити всі вимоги з замовником перед початком проекту.
5	Затримка фінансування	3	1	L	Обговорити план фінансування проекту.

- Зелений колір: прийнятні ризики.
- Жовтий колір: виправданні ризики.
- Червоний колір: недопустимі ризики.

ДОДАТОК Б. КОД РЕАЛІЗАЦІЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ

create_users_table.php

Міграція створення таблиці аккаунта.

```
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreateUsersTable extends Migration
{
    public function up()
    {
        Schema::create('users', function (Blueprint $table) {
            $table->id();
            $table->string('name');
            $table->string('surname');
            $table->string('date_bth');
            $table->boolean('sex');
            $table->string('phone')->nullable();
            $table->foreignId('country_id')->nullable();
            $table->string('city')->nullable();
            $table->text('description')->nullable();
            $table->string('skype')->nullable();
            $table->string('telegram')->nullable();
            $table->string('whatsapp')->nullable();
            $table->string('viber')->nullable();
            $table->string('facebook')->nullable();
            $table->string('photo')->default('/img/no-avatar.png');
            $table->string('photo_small')->default('/img/no-avatar.png');
            $table->string('email')->unique();
            $table->string('password');
            $table->foreignId('role_id');
            $table->foreignId('agency_type_id')->nullable();
            $table->string('title_company')->nullable();
            $table->string('link_site')->nullable();
            $table->string('size_company')->nullable();
            $table->boolean('verification')->default(0);
            $table->rememberToken();
            $table->timestamps();
        });
        Schema::table('users', function (Blueprint $table) {
            $table->index('role_id');
            $table->foreign('role_id')->references('id')->on('roles');
            $table->index('agency_type_id');
            $table->foreign('agency_type_id')->references('id')->on('agency_types');
            $table->index('country_id');
            $table->foreign('country_id')->references('id')->on('country');
        });
    }
    public function down()
    {
        Schema::dropIfExists('users');
    }
}
```

create_orders_table.php

Міграція створення таблиці замовлень.

```
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreateOrdersTable extends Migration
{
    public function up()
    {
        Schema::create('orders', function (Blueprint $table) {
            $table->id();
            $table->string('title');
            $table->text('description');
            $table->integer('price');
            $table->foreignId('category_id');
            $table->foreignId('country_id');
            $table->foreignId('user_id');
            $table->timestamps();
        });
        Schema::table('orders', function (Blueprint $table) {
            $table->index('category_id');
            $table->foreign('category_id')->references('id')->on('agency_types');
            $table->index('country_id');
            $table->foreign('country_id')->references('id')->on('country');
            $table->index('user_id');
            $table->foreign('user_id')->references('id')->on('users')->onDelete('cascade');
        });
    }
    public function down()
    {
        Schema::dropIfExists('orders');
    }
}
```

create_requests_table.php

Міграція створення таблиці запитів.

```
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreateRequestsTable extends Migration
{
    public function up()
    {
        Schema::create('requests', function (Blueprint $table) {
            $table->id();
            $table->foreignId('order_id');
            $table->foreignId('user_id');
            $table->text('description')->nullable();
            $table->string('file')->nullable();
        });
    }
}
```

```

        $table->string('filename')->nullable();
        $table->string('deadline')->nullable();
        $table->foreignId('status_id');
        $table->boolean('status_pay')->default(0);
        $table->timestamps();
    });
    Schema::table('requests', function (Blueprint $table) {
        $table->index('user_id');
        $table->foreign('user_id')->references('id')->on('users')->onDelete('cascade');
        $table->index('order_id');
        $table->foreign('order_id')->references('id')->on('orders')->onDelete('cascade');
        $table->index('status_id');
        $table->foreign('status_id')->references('id')->on('request_status');
    });
}
public function down()
{
    Schema::dropIfExists('requests');
}
}

```

create_response_table.php

Міграція створення таблиці відгуків.

<?php

```

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreateResponseTable extends Migration
{
    public function up()
    {
        Schema::create('response', function (Blueprint $table) {
            $table->id();
            $table->foreignId('user_id');
            $table->foreignId('request_id');
            $table->text('description');
            $table->integer('star_1')->default(1);
            $table->integer('star_2')->default(1);
            $table->integer('star_3')->default(1);
            $table->integer('star_4')->default(1);
            $table->timestamps();
        });
        Schema::table('response', function (Blueprint $table) {
            $table->index('user_id');
            $table->foreign('user_id')->references('id')->on('users')->onDelete('cascade');
            $table->index('request_id');
            $table->foreign('request_id')->references('id')->on('requests')->onDelete('cascade');
        });
    }
    public function down()
    {
        Schema::dropIfExists('response');
    }
}

```

create_chats_table.php

Міграція створення таблиці чатів.

```
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreateChatsTable extends Migration
{
    public function up()
    {
        Schema::create('chats', function (Blueprint $table) {
            $table->id();
            $table->foreignId('user_id_1');
            $table->foreignId('user_id_2');
            $table->timestamps();
        });

        Schema::table('chats', function (Blueprint $table) {
            $table->index('user_id_1');
            $table->foreign('user_id_1')->references('id')->on('users')->onDelete('cascade');
            $table->index('user_id_2');
            $table->foreign('user_id_2')->references('id')->on('users')->onDelete('cascade');
        });
    }
    public function down()
    {
        Schema::dropIfExists('chats');
    }
}
```

create_messages_table.php

Міграція створення таблиці листувань.

```
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

class CreateMessagesTable extends Migration
{
    public function up()
    {
        Schema::create('messages', function (Blueprint $table) {
            $table->id();
            $table->text('text');
            $table->foreignId('user_id');
            $table->foreignId('chat_id');
            $table->timestamps();
        });

        Schema::table('messages', function (Blueprint $table) {
            $table->index('user_id');
            $table->foreign('user_id')->references('id')->on('users')->onDelete('cascade');
            $table->index('chat_id');
            $table->foreign('chat_id')->references('id')->on('chats')->onDelete('cascade');
        });
    }
}
```

```

public function down()
{
    Schema::dropIfExists('messages');
}
}

```

User.php

Модель таблиці користувачів.

```

<?php

namespace App\Models;

use Illuminate\Foundation\Auth\User as Authenticatable;
use Illuminate\Notifications\Notifiable;
use Laravel\Passport\HasApiTokens;

class User extends Authenticatable
{
    use Notifiable, HasApiTokens;

    protected $fillable = [
        'name',
        'surname',
        'date_bth',
        'sex',
        'phone',
        'country_id',
        'city',
        'description',
        'skype',
        'telegram',
        'whatsapp',
        'viber',
        'facebook',
        'photo',
        'photo_small',
        'verification',
        'email',
        'password',
        'role_id',
        'agency_type_id',
        'title_company',
        'link_site',
        'size_company'
    ];

    protected $hidden = [
        'password'
    ];

    function country() {
        return $this->belongsTo('App\Models\Country', 'country_id');
    }

    function orders() {

```

```

    return $this->hasMany('App\Models\Order', 'user_id');
}

function requests() {
    return $this->hasMany('App\Models\Requests', 'user_id');
}

function response() {
    return $this->hasMany('App\Models\Response', 'user_id');
}

function role() {
    return $this->belongsTo('App\Models\Role', 'role_id');
}
}

```

Order.php

Модель таблиці замовлень.

```

<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Order extends Model
{
    use HasFactory;

    protected $table = 'orders';

    protected $fillable = [
        'title',
        'description',
        'price',
        'country_id',
        'user_id',
        'category_id'
    ];

    protected $casts = [
        'created_at' => 'datetime:d.m.Y',
        'updated_at' => 'datetime:m.d.Y',
    ];

    function tags() {
        return $this->hasMany('App\Models\OrderTags', 'order_id');
    }

    function country() {
        return $this->belongsTo('App\Models\Country', 'country_id');
    }

    function user() {
        return $this->belongsTo('App\Models\User', 'user_id');
    }

    function request() {
        return $this->hasMany('App\Models\Requests', 'order_id');
    }
}

```



```
}
}
```

Requests.php

Модель таблиці запитів.

```
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Requests extends Model
{
    use HasFactory;

    protected $table = 'requests';

    protected $fillable = [
        'order_id',
        'user_id',
        'description',
        'file',
        'filename',
        'deadline',
        'status_id',
        'status_pay'
    ];

    function user() {
        return $this->belongsTo('App\Models\User', 'user_id');
    }

    function order() {
        return $this->belongsTo('App\Models\Order', 'order_id');
    }

    function status() {
        return $this->belongsTo('App\Models\RequestStatus', 'status_id');
    }

    function response() {
        return $this->hasOne('App\Models\Response', 'request_id');
    }
}
```

Response.php

Модель таблиці відгуків.

```
<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Factories\HasFactory;
use Illuminate\Database\Eloquent\Model;

class Response extends Model
{
    use HasFactory;
```

```

protected $table = 'response';

protected $fillable = [
    'user_id',
    'request_id',
    'description',
    'star_1',
    'star_2',
    'star_3',
    'star_4'
];

function user() {
    return $this->belongsTo('App\Models\User', 'user_id');
}
}

```

UserController.php

Контролер для роботи з користувачами.

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;
use App\Models\User;
use App\Models\Requests;
use App\Models\Order;
use Illuminate\Support\Facades\Hash;

use Image;

class UserController extends Controller
{
    protected $fileStorage = "/uploads/";
    protected $defaultPhoto = "/img/no-avatar.png";

    function profile() {
        $model = User::find(Auth::id());
        return response()->json($model);
    }

    function update(Request $request) {
        $model = User::find(Auth::id());
        $data = $request->all();

        if($request->new_photo != 'null') {
            $name = $this->fileStorage . uniqid() . '.' . $request['new_photo']->getClientOriginalExtension();
            $nameSmall = $this->fileStorage . uniqid() . '_small.' . $request['new_photo']->getClientOriginalExtension();
            $data['photo'] = $name;
            $data['photo_small'] = $nameSmall;
        }

        if($request->new_photo != 'null') {
            $this->cropImg(
                $request->file('new_photo'),
                $name,
                $nameSmall,

```

```

        $request->cropWidth,
        $request->cropHeight,
        $request->cropLeft,
        $request->cropTop
    );
}

$data['verification'] = true;

$model->update($data);
return response('ok', 200);

}

function getUser(Request $request, $id) {
    $data = User::with('country')->find($id);
    if($data['role_id'] == 2) {
        $data['response'] = Order::where('category_id', $request->category_id)
            ->where('user_id', $id)
            ->with('request.response.user', function($q) use ($id) {
                return $q->with('user')->where('user_id', '!=', $id)->get();
            })
            ->get();
        $data['orders'] = Order::with('country')->where('category_id', $request->category_id)->where('user_id', $id)->get();
    }
    if($data['role_id'] == 3) {
        $data['response'] = Requests::with('order', 'user')->where('user_id', $id)
            ->whereHas('order', function ($query) use ($request) {
                $query->where('category_id', $request->category_id);
            })->with('response.user', function($q) use ($id) {
                return $q->with('user')->where('user_id', '!=', $id)->get();
            })->get();
    }
    return response()->json($data);
}

function getUsers(Request $request) {
    $model = User::with('country', 'role')->withCount('response', 'orders')->where('role_id', '!=', 1);
    if($request->verification) {
        $model->where('verification', 1);
    }
    if($request->role_id) {
        $model->where('role_id', $request->role_id);
    }
    if($request->category_id) {
        $model->where('agency_type_id', $request->category_id);
    }
    if($request->title) {
        $model->where('name', 'like', "%".$request->title."%")->orWhere('surname', 'like', "%".$request->title."%");
    }
    if(isset($request->sorted)) {
        switch ($request->sorted) {
            case '1':

                break;
            case '2':
                $model->orderBy('size_company', 'asc');
                break;
            case '3':
                $model->orderBy('name', 'asc');

```

```

        break;
    }
}
$data = $model->get();
return response()->json($data);
}

function deleteUser($id) {
    User::find($id)->delete();
    return response('ok', 200);
}

function cropImg($file, $name, $nameSmall, $width, $height, $left, $top) {
    $image = Image::make($file->getRealPath());
    $imageSmall = Image::make($file->getRealPath());
    $image->resize(800, 1200, function ($constraint) {
        $constraint->aspectRatio();
    }->save(public_path() . '/' . $name);
    $imageSmall->crop($width, $height, $left, $top)->resize(300, 300, function ($constraint) {
        $constraint->aspectRatio();
    }->save(public_path() . '/' . $nameSmall);
}}

```

OrderController.php

Контролер для роботи з замовленнями.

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;
use App\Models\Order;
use App\Models\Tags;
use App\Models\OrderTags;

class OrderController extends Controller
{
    function get(Request $request) {
        $model = Order::with('tags.tag', 'country', 'user', 'request');
        if($request->user_id) {
            $model->whereHas('user', function($query) use ($request) {
                $query->where('id', $request->user_id);
            });
        }
        if(isset($request->category_id)) {
            $model->where('category_id', $request->category_id);
        }
        if($request->title != "") {
            $model->where('title', 'like', "%".$request->title."%");
        }
        if(isset($request->country)) {
            $model->where('country_id', $request->country);
        }
        if(isset($request->priceFrom)) {
            $model->where('price', '>=', $request->priceFrom);
        }
        if(isset($request->priceTo)) {
            $model->where('price', '<=', $request->priceTo);
        }
    }
}

```

```

if(isset($request->tags)) {
    $model->whereHas('tags', function($query) use($request) {
        $query->whereIn('tag_id', $request->tags);
    });
}
if(isset($request->sorted)) {
    switch ($request->sorted) {
        case '1':
            $model->orderBy('created_at', 'asc');
            break;
        case '2':
            $model->orderBy('price', 'asc');
            break;
        case '3':
            $model->orderBy('price', 'desc');
            break;
    }
}
$data = $model->get();
return response()->json($data);
}
function getId($id) {
    $data = Order::with('tags.tag', 'country', 'user', 'request')->find($id);
    return response()->json($data);
}
function post(Request $request) {
    $order = new Order();
    $data = $request->all();
    $data['user_id'] = Auth::id();
    $orderResponse = $order->create($data);
    foreach ($data['tags'] as $key => $value) {
        if(!Tags::where('title', $value)->exists()) {
            $tag = new Tags();
            $tagOrder = new OrderTags();
            $tagResponse = $tag->create([
                'title' => $value
            ]);
            $tagOrder->create([
                'order_id' => $orderResponse->id,
                'tag_id' => $tagResponse->id
            ]);
        } else {
            $tagOrder = new OrderTags();
            $tagOrder->create([
                'order_id' => $orderResponse->id,
                'tag_id' => Tags::where('title', $value)->first()->id
            ]);
        }
    }
    return response()->json([
        'id' => $orderResponse->id
    ]);
}
function put(Request $request, $id) {
    $order = Order::find($id);
    $data = $request->all();
    $order->update($data);
    OrderTags::where('order_id', $id)->delete();
    foreach ($data['tags'] as $key => $value) {
        if(!Tags::where('title', $value)->exists()) {

```

```

    $tag = new Tags();
    $tagOrder = new OrderTags();
    $tagResponse = $tag->create([
        'title' => $value
    ]);
    $tagOrder->create([
        'order_id' => $id,
        'tag_id' => $tagResponse->id
    ]);
} else {
    $tagOrder = new OrderTags();
    $tagOrder->create([
        'order_id' => $id,
        'tag_id' => Tags::where('title', $value)->first()->id
    ]);
}
}
return response('ok', 200);
}
function del($id) {
    Order::find($id)->delete();
    return response('ok', 200);
}

function getTags(Request $request) {
    $model = Tags::with('tags.order')->withCount('tags');
    if(isset($request->category_id)) {
        $model->whereHas('tags.order', function($query) use ($request) {
            $query->where('category_id', $request->category_id);
        });
    }
    $data = $model->get();
    return response()->json($data);
}
}

```

RequestController.php

Контролер для роботи з запитами.

```

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;
use App\Models\Requests;

class RequestController extends Controller
{
    protected $fileStorage = "/userfiles/";

    function get(Request $request) {
        $model = Requests::with('user', 'order.country', 'order.user');
        $user = Auth::user();
        if($user->role_id == 2) {
            $model->whereHas('order', function($query) use ($user) {
                $query->where('user_id', $user->id);
            });
        }
        if($user->role_id == 3) {

```

```

    $model->where('user_id', $user->id);
}
if(isset($request->category_id)) {
    $model->whereHas('order', function($query) use ($request) {
        $query->where('category_id', $request->category_id);
    });
}
$data = $model->get();
return response()->json($data);
}

function getId($id) {
    $data = Requests::with(
        'order.country',
        'order.tags.tag',
        'order.user.country',
        'user',
        'status',
        'response'
    )->find($id);
    return response()->json($data);
}

function post(Request $request) {
    $model = new Requests();
    $data = $request->all();
    $model->create([
        'order_id' => $data['order_id'],
        'user_id' => Auth::id(),
        'status_id' => 1
    ]);
    return response('ok', 200);
}

function put(Request $request, $id) {
    $data = $request->all();
    if(isset($data['pay'])) {
        $data['status_pay'] = true;
    }
    $model = Requests::find($id);
    if(isset($request['file']) && $request['file'] != "null") {
        $path = $this->fileStorage;
        $name = $path."/".uniqid().'.'.$request['file']->getClientOriginalExtension();
        $request['file']->move(public_path().$path, $name);
        $data['file'] = $name;
    }
    $model->update($data);
    return response('ok', 200);
}

function delete($id) {
    Requests::find($id)->delete();
    return response('ok', 200);
}

public function updateRequesrs()
{
    $data = Requests::get();
    foreach ($data as $key => $value) {
        $date = strtotime($value['deadline']);
        if($value['status_id'] == 4) {
            Requests::find($value['id'])->update([

```

```

        'status_id' => 3
    ]);
}
}
return response('ok', 200);
}
}

```

ChatController.php

Контролер для роботи чату.

```
<?php
```

```
namespace App\Http\Controllers;
```

```
use App\Events\MessageSentEvent;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;
use Carbon\Carbon;
use App\Models\Chats;
use App\Models\Messages;
```

```
class ChatController extends Controller
```

```
{
    function postChat(Request $request) {
        $model1 = Chats::where("user_id_1", $request->user_id)->where("user_id_2", Auth::id());
        $model2 = Chats::where("user_id_1", Auth::id())->where("user_id_2", $request->user_id);
        if($model1->exists()) {
            $message = new Messages();
            $id1 = $model1->first()['id'];
            $message->create([
                "text" => $request->message,
                "user_id" => Auth::id(),
                "chat_id" => $id1
            ]);
        } else if($model2->exists()) {
            $message = new Messages();
            $id2 = $model2->first()['id'];
            $message->create([
                "text" => $request->message,
                "user_id" => Auth::id(),
                "chat_id" => $id2
            ]);
        } else {
            $chat = new Chats();
            $response = $chat->create([
                "user_id_1" => Auth::id(),
                "user_id_2" => $request->user_id,
            ]);
            $message = new Messages();
            $message->create([
                "text" => $request->message,
                "user_id" => Auth::id(),
                "chat_id" => $response['id']
            ]);
        }
    }
}

```

```
function getChats() {
```

```
    $chats = Chats::with('messages.user')->where("user_id_1", Auth::id())->orWhere("user_id_2", Auth::id())->get();
```



```

    foreach ($chats as $key => $chat) {
        $chat->date = Carbon::parse($chat->created_at)->format('d.m.Y H:i');
    }
    return response()->json($chats);
}

function postMessages(Request $request, $chat_id) {
    $user = Auth::user();

    $message = new Messages();
    $response = $message->create([
        "text" => $request->message,
        "user_id" => Auth::id(),
        "chat_id" => $chat_id
    ]);

    broadcast(new MessageSentEvent($response, $user, $chat_id))->toOthers();

    return response()->json([
        'message' => $response,
        'user' => $user
    ]);
}

function getMessages($chat_id) {
    $result = [];
    $data = Messages::with('user')->where('chat_id', $chat_id)->get();
    foreach ($data as $key => $chat) {
        array_push($result, [
            'message' => $chat,
            'user' => $chat['user']
        ]);
    }
    return response()->json($result);
}
}
}

```

ListPossibilities.vue

КОМПОНЕНТ ЗАМОВЛЕНЬ.

```

<template>
<div>
<Header></Header>
<b-container class="main">
<b-row>
<b-col md="4" lg="4" xl="4" xs="12">
<div class="filter">
<b-form-group
class="mb-0"
label="Теги"
v-slot="{ ariaDescribedby }"
>
<b-form-checkbox
v-for="tag in tags"
:key="tag.id"
:value="tag.id"
:aria-describedby="ariaDescribedby"
v-model="filter.tags"
>
{{ tag.title }} <span>{{ tag.tags_count }}</span>

```

```

    </b-form-checkbox>
  </b-form-group>

  <div class="col-form-label">
    Країна
  </div>
  <b-form-select
    :options="country"
    v-model="filter.country"
    class="mb-2"
    value-field="id"
    text-field="title"
  ></b-form-select>

  <div class="col-form-label">
    Ціна договору
  </div>
  <b-row>
    <b-col>
      <b-input-group append="$" class="mb-2 mr-sm-2 mb-sm-0">
        <b-form-input
          v-model="filter.priceFrom"
          placeholder="Від"
        ></b-form-input>
      </b-input-group>
    </b-col>
    <b-col>
      <b-input-group append="$" class="mb-2 mr-sm-2 mb-sm-0">
        <b-form-input
          placeholder="До"
          v-model="filter.priceTo"
        ></b-form-input>
      </b-input-group>
    </b-col>
  </b-row>
  <button :style="background: '+$store.getters.color'" @click="getData()">Примінити</button>
</div>
</b-col>
<b-col md="8" lg="8" xl="8" xs="12" class="main-block">
  <Search :filter="filter" @getData="getData"></Search>
  <div class="sort">
    <label for="sort">Сортувати: </label>
    <select id="sort" v-model="filter.sorted" @change="getData()">
      <option value="1">За датою</option>
      <option value="2">Спочатку дороги</option>
      <option value="3">Спочатку дешеві</option>
    </select>
  </div>
  <div v-for="item in paginatedList" :key="item.id">
    <router-link :to="'/'+$route.params.category+'/possibility/'+item.id">
      <Item :data="item"></Item>
    </router-link>
  </div>
  <b-pagination
    v-model="currentPage"
    align="center"
    :total-rows="data.length"
    :per-page="prePage"
  ></b-pagination>
</b-col>

```

```

    </b-row>
  </b-container>
</Footer></Footer>
</div>
</template>
<script>
import country from '../mixins/country.vue';
import Header from '../components/Header.vue'
import Footer from '../components/Footer.vue'
import Item from '../components/Item.vue'
import Search from '../components/Search.vue'
export default {
  mixins: [country],
  components: {
    Header,
    Footer,
    Item,
    Search
  },
  data() {
    return {
      currentPage: 1,
      prePage: 6,
      data: [],
      tags: [],
      filter: {
        title: "",
        sorted: "1",
        country: null,
        priceFrom: "",
        priceTo: "",
        tags: [],
        category_id: null
      }
    }
  },
  created() {
    this.getData();
    this.getTags();
  },
  computed: {
    paginatedList() {
      let start = (this.currentPage - 1) * this.prePage;
      let end = start + this.prePage;
      return this.data.slice(start, end);
    }
  },
  methods: {
    getData() {
      this.filter.category_id = this.$route.params.category;
      axios.get('/api/orders/', {
        params: this.filter
      }).then(response => {
        this.data = response.data.map(item => {
          item.description = item.description.replace(/<.*?>/g, "")
          return item;
        });
      });
    },
    getTags() {

```

```

    axios.get('/api/tags', {
      params: {
        category_id: this.$route.params.category
      }).then(response => {
        this.tags = response.data;
      })
    }
  },
  watch: {
    $route(to, from) {
      this.getData();
      this.getTags();
    }
  },
}
</script>
<style lang="css" scoped>
.filter {
  background: #FFFFFF;
  border: 1px solid #DBEAF0;
  box-sizing: border-box;
  border-radius: 12px;
  padding: 20px;
}
.filter button {
  width: 100%;
  color: #fff;
  padding: 14px;
  border-radius: 4px;
  border: 0;
  margin-top: 25px;
}
a:hover {
  text-decoration: none;
}
@media screen and (max-width: 600px) {
  .main-block {
    margin-top: 20px;
  }
}
</style>

```

MessageSentEvent.php

Контролер чатів. Працює з діалогами користувачів. Зберігає та створює нові чати між користувачами та зберігає повідомлення надіслані користувачами у відповідні чати.

Даний код надає можливість підключення до чату тільки авторизованим користувачам.

```
<?php
```

```
use Illuminate\Support\Facades\Broadcast;
```

```
Broadcast::channel('chat-{id}', function () {
    return \Illuminate\Support\Facades\Auth::check();
});
```

Наступний код визначає канал на якому ведеться діалог між користувачами.

```
class MessageSentEvent implements ShouldBroadcast
{
    use Dispatchable, InteractsWithSockets, SerializesModels;

    public $message;
    public $account;
    public $chat_id;

    public function __construct(Messages $message, User $user, $chat_id)
    {
        $this->message = $message;
        $this->account = $user;
        $this->chat_id = $chat_id;
    }

    public function broadcastOn()
    {
        return new Channel('chat-' . $this->chat_id);
    }
}
```

```
<?php
```

```
namespace App\Http\Controllers;

use App\Events\MessageSentEvent;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Auth;
use Carbon\Carbon;
use App\Models\Chats;
use App\Models\Messages;

class ChatController extends Controller
{
    function postChat(Request $request) {
        $model1 = Chats::where("user_id_1", $request->user_id)->where("user_id_2", Auth::id());
        $model2 = Chats::where("user_id_1", Auth::id())->where("user_id_2", $request->user_id);
        if($model1->exists()) {
            $message = new Messages();
            $id1 = $model1->first()['id'];
            $message->create([
                "text" => $request->message,
                "user_id" => Auth::id(),
                "chat_id" => $id1
            ]);
        } else if($model2->exists()) {
            $message = new Messages();
            $id2 = $model2->first()['id'];
            $message->create([
                "text" => $request->message,
                "user_id" => Auth::id(),
                "chat_id" => $id2
            ]);
        } else {
            $chat = new Chats();
            $response = $chat->create([
```

```

        "user_id_1" => Auth::id(),
        "user_id_2" => $request->user_id,
    ]);
    $message = new Messages();
    $message->create([
        "text" => $request->message,
        "user_id" => Auth::id(),
        "chat_id" => $response['id']
    ]);
    }
}

function getChats() {
    $chats = Chats::with('messages.user')->where("user_id_1", Auth::id())->orWhere("user_id_2", Auth::id())->get();
    foreach ($chats as $key => $chat) {
        $chat->date = Carbon::parse($chat->created_at)->format('d.m.Y H:i');
    }
    return response()->json($chats);
}

function postMessages(Request $request, $chat_id) {
    $user = Auth::user();

    $message = new Messages();
    $response = $message->create([
        "text" => $request->message,
        "user_id" => Auth::id(),
        "chat_id" => $chat_id
    ]);

    broadcast(new MessageSentEvent($response, $user, $chat_id))->toOthers();

    return response()->json([
        'message' => $response,
        'user' => $user
    ]);
}

function getMessages($chat_id) {
    $result = [];
    $data = Messages::with('user')->where('chat_id', $chat_id)->get();
    foreach ($data as $key => $chat) {
        array_push($result, [
            'message' => $chat,
            'user' => $chat['user']
        ]);
    }
    return response()->json($result);
}
}

```

AuthController.php

Контроллер авторизації користувачів сайту. Містить методи авторизації, реєстрації, завершення сесії та перевірки статусу авторизованого користувача. У

випадку якщо сесія закінчилась відбувається переадресація на сторінку авторизації.

```
<?php
```

```
namespace App\Http\Controllers;
```

```
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Hash;
```

```
use App\Models\User;
use Auth;
use Cookie;
```

```
class AuthController extends Controller
{
```

```
function checkUser() {
    $model = User::find(Auth::id());
    return response()->json($model);
}
```

```
function login(Request $request) {
    $request->validate([
        'email' => 'required|string|email',
        'password' => 'required|string',
    ]);
    $credentials = request(['email', 'password']);
    if (Auth::attempt($credentials)) {
        $user = Auth::user();
        $token = $user->createToken('Personal Access Token')->accessToken;
        $cookie = $this->getCookieDetails($token);
        return response()->json([
            'user' => $user,
            'token' => $token,
        ], 200)
        ->cookie($cookie['name'], $cookie['value'], $cookie['minutes'], $cookie['path'], $cookie['domain'], $cookie['secure'],
        Cookie::httpOnly(), $cookie['samesite']);
    } else {
        return response()->json([
            'error' => 'Помилка авторизації. Некоректний email, або пароль.'
        ], 422);
    }
}
```

```
// register
```

```
function register(Request $request) {
    $request->validate([
        'email' => 'required|string|email|unique:users',
        'password' => 'required|string'
    ]);
    $user = new User();
    $data = $request->all();
    $data["password"] = Hash::make($request->password);
    $user->create($data);
    $credentials = request(['email', 'password']);
    if(!Auth::attempt($credentials)) {
        return response()->json(['message' => 'Unauthorized'], 401);
    }
}
```

```
$authUser = Auth::user();
$tokenResult = $authUser->createToken('Personal Access Token');
$token = $tokenResult->token;
$token->save();
return response()->json([
    'access_token' => 'Bearer '.$tokenResult->accessToken,
    'user' => $authUser
]);
}

private function getCookieDetails($token) {
    return [
        'name' => '_token',
        'value' => $token,
        'minutes' => 1440,
        'path' => null,
        'domain' => null,
        'secure' => null,
        'httponly' => true,
        'samesite' => true,
    ];
}

function logout(Request $request) {
    $request->user()->token()->revoke();
    $cookie = Cookie::forget('_token');
    return response('ok', 200)->withCookie($cookie);
}
}
```