

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК ТА ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРИЗОВАНИХ СИСТЕМ УПРАВЛІННЯ

ДОПУСТИТИ ДО ЗАХИСТУ

Завідувач кафедри

Олександр ЛИТВИНЕНКО

“ ” _____ 2023 р.

КВАЛІФІКАЦІЙНА РОБОТА

(ПОЯСНЮВАЛЬНА ЗАПИСКА)

ЗДОБУВАЧА ВИЩОЇ ОСВІТИ

ОСВІТНЬОГО СТУПЕНЯ «МАГІСТР»

Тема: Програмний модуль управління взаємодією з клієнтами для авіакомпанії

Виконавець: Назарій КОРОЛЬ

Керівник: Надія МАРЧЕНКО

Нормоконтролер: Євгеній ТУПОТА

Київ 2023

НАЦІОНАЛЬНИЙ АВІАЦІЙНИЙ УНІВЕРСИТЕТ

Факультет комп'ютерних наук та технологій

Кафедра комп'ютеризованих систем управління

Спеціальність 123 «Комп'ютерна інженерія»

Освітньо-професійна програма «Системне програмування»

Форма навчання денна

ЗАТВЕРДЖУЮ

Завідувач кафедри

Олександр ЛИТВИНЕНКО

« _____ » _____ 2023 р.

ЗАВДАННЯ

на виконання кваліфікаційної роботи

Короля Назарія Володимировича

1. Тема кваліфікаційної роботи Програмний модуль управління взаємодією з клієнтами для авіакомпанії

затверджена наказом ректора від «28» 08 2023 р. № 1494/ст

2. Термін виконання роботи (проєкту): з 02.10.2023 р. по 31.12.2023 р.

3. Вихідні дані до роботи (проєкту): середовище розробки Visual Code, редактор діаграм Microsoft Visio, текстовий процесор Microsoft Word.

4. Зміст пояснювальної записки: вступ, аналіз предметної області, вимоги до програмного модуля, програмна реалізація модуля, методика роботи користувача з модулем, висновки.

5. Перелік обов'язкового графічного (ілюстративного) матеріалу:

1) система збору і обробки інформації;

2) архітектура веб-додатку;

3) базовий набір веб-технологій;

4) схема алгоритму обробки запитів клієнтів;

5) взаємодія веб-серверу та серверу додатку на окремих машинах.

6. Календарний план

| № п/п | Етапи виконання кваліфікаційної роботи | Термін виконання етапів | Примітка |
|-------|---|-------------------------|----------|
| 1 | Пошук та аналіз джерел для аналізу предметної області за темою кваліфікаційної роботи | 02.10-21.10 | |
| 2 | Розробка плану кваліфікаційної роботи | 22.10-02.10 | |
| 3 | Розробка розділу 1: Аналіз предметної області | 03.11-13.11 | |
| 4 | Розробка розділу 2: Вимоги до програмного модуля | 14.11-24.11 | |
| 5 | Розробка розділу 3: Програмна реалізація модуля | 25.11-02.12 | |
| 6 | Розробка розділу 4: Методика роботи користувача з модулем та підготовка до попереднього захисту | 03.12-10.12 | |
| 7 | Оформлення пояснювальної записки та написання висновків | 11.12-18.12 | |
| 8 | Розробка презентації для захисту роботи | 19.12-24.12 | |
| 9 | Підготовка до захисту | 25.12-31.12 | |

7. Дата отримання завдання: «02» 10. 2023 р.

Керівник кваліфікаційної роботи _____

Надія МАРЧЕНКО

Завдання прийняв до виконання _____

Назарій КОРОЛЬ

РЕФЕРАТ

Кваліфікаційна робота на тему «Програмний модуль управління взаємодією з клієнтами для авіакомпанії». Записка до кваліфікаційної роботи містить: 79 с., 17 рис., 1 табл., 33 літературних джерела, 1 додаток.

Ключові слова: АВІАКОМПАНІЯ, СИСТЕМА УПРАВЛІННЯ, МОДУЛЬ, CRM, QR-КОД.

Об'єктом дослідження є надання послуг авіакомпанією.

Предметом дослідження є програмний модуль управління взаємодією з клієнтами для авіакомпанії.

Метою кваліфікаційної роботи є розробити програмний модуль управління взаємодією з клієнтами для авіакомпанії.

Технічними та програмними засобами що використовувалися при розробці кваліфікаційної роботи є персональний комп'ютер з операційною системою *Windows 10*.

Для створення програмного модулю використовувався процедурний метод програмування, алгоритми побудовані відповідно міжнародним і державним стандартам. Основними характеристиками та показниками в роботі є швидкодія, універсальність та інноваційність розробленого програмного забезпечення.

В свою чергу науковою новизною – перевірка медичних норм пасажирів з подальшим друком QR-коду.

Значущість виконаної роботи полягає у детальному аналізі програмних засобів керування взаємодією з клієнтами авіакомпанії, розробці власного технічно правильного та зручного для користувача веб-додатку.

Головною рекомендацією використання результатів можна назвати впровадження сучасних технологій та програмних рішень в аеропортах та авіакомпаніях.

ЗМІСТ

| | |
|--|----|
| ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ | 6 |
| ВСТУП..... | 7 |
| РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ | 12 |
| 1.1. Аналіз систем управління взаємодією з клієнтами для авіакомпанії..... | 12 |
| 1.2. Варіанти управління інформаційною діяльністю авіакомпанії..... | 18 |
| 1.3. Постановка завдання проектування | 22 |
| 1.4. Висновки до розділу | 24 |
| РОЗДІЛ 2 ВИМОГИ ДО ПРОГРАМНОГО МОДУЛЯ..... | 25 |
| 2.1. Функціональні вимоги програмного модуля | 25 |
| 2.2. Структура програмного модуля..... | 25 |
| 2.3. Архітектура програмного забезпечення | 34 |
| 2.4. Висновки до розділу | 42 |
| РОЗДІЛ 3 ПРОГРАМНА РЕАЛІЗАЦІЯ МОДУЛЯ | 45 |
| 3.1. Системні вимоги програмного модуля | 45 |
| 3.2. Алгоритми реалізації модулів..... | 51 |
| 3.3. Тестування роботи програмного модуля | 54 |
| 3.4 Висновки до розділу | 57 |
| РОЗДІЛ 4 МЕТОДИКА РОБОТИ КОРИСТУВАЧА З МОДУЛЕМ | 59 |
| 4.1. Системні вимоги й інсталяція модуля | 59 |
| 4.2. Робота користувача з додатком | 61 |
| 4.3. Робота адміністратора з додатком..... | 65 |
| 4.4. Висновки до розділу | 70 |
| ВИСНОВКИ | 73 |
| СПИСОК БІБЛОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ..... | 77 |
| Додаток А..... | 80 |

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ, ТЕРМІНІВ

API – прикладний програмний інтерфейс.

IoT – мережа фізичних об'єктів, які мають вбудовані технології, що дозволяють здійснювати взаємодію з зовнішнім середовищем, передавати відомості про свій стан і приймати дані ззовні.

Блокчейн – збудований за певними правилами безперервний послідовний ланцюжок блоків (зв'язковий список), що містять інформацію.

QR-код – матричний код (двовимірний штрих-код), розроблений і представлений японською компанією «*Denso-Wave*» в 1994 році. Основна перевага *QR*-коду – це легке розпізнавання сканувальним обладнанням (в тому числі й фотокамерою мобільного телефона), що дає можливість використання в торгівлі, на виробництві, в логістиці.

Методологія програмування – сукупність методів, застосовуваних на різних стадіях життєвого циклу розробки програмного забезпечення, що мають спільний філософський підхід та, відповідно до цього підходу, дозволяють забезпечити найкращу ефективність процесів розробки.

ERP - система планування ресурсів підприємства.

SCM – управління потоком товарів, даних та фінансів, пов'язаних із продуктом або послугою, від закупівлі сировини до доставки продукту до кінцевого пункту призначення.

SQL ін'єкції – це метод отримання несанкціонованого доступу (різновид злому) до бази даних, при якому шкідливий код виконується прямо з поля вводу звичайної форми.

UI (User Interface) – користувацький інтерфейс. Цим терміном описують те, як продукт виглядає: кольори, дизайн, анімація, контент, форми (кнопки, мітки, поля для вводу) тощо.

UX (User Experience) – це користувацький досвід.

ВСТУП

Авіаційна галузь виступає ключовою ланкою в глобальній економічній системі, слугуючи мостом між країнами, культурами та ринками. Ця галузь підтримує міжнародну торгівлю, туризм і бізнес, сприяючи створенню робочих місць та росту ВВП. Економічно, авіаційна індустрія створює мільйони робочих місць, прямо або опосередковано, і сприяє зростанню інших галузей, таких як туризм та гостинність. Через великі інвестиції в авіатехніку та інфраструктуру, галузь стимулює технологічний прогрес і інноваційність.

Соціально, авіація забезпечує зв'язок між різними частинами світу, сприяючи культурному обміну та розширенню горизонтів особистісного розвитку. Вона також сприяє швидкому реагуванню на надзвичайні ситуації, такі як природні катастрофи або медичні екстрени.

Незважаючи на її важливість, авіаційна галузь часто зіткнулася з викликами, які варіюються від політичних і економічних криз до пандемій та кліматичних змін. Її адаптивність та здатність до інновацій визначають стабільність галузі та її вплив на глобальний контекст. Майбутнє авіаційної галузі визначатиметься технологічними нововведеннями, стратегіями сталого розвитку та здатністю адаптуватися до змінюваних умов ринку. В цілому, авіаційна галузь відіграє життєво важливу роль у сучасному світі, і її стабільність та процвітання мають безпосередній вплив на економічне та соціальне благополуччя країн та їх населення.

Сучасний стан авіаційної галузі України на фоні глобалізації є ключовим для економічної і соціальної інтеграції країни у світові системи.

В період 2014-2015 років авіаційна галузь України зіткнулася з великими викликами через військово-політичні події, що призвели до складної ситуації в державі. 2014 рік став особливо складним через анексію Криму, рекомендації щодо безпеки польотів від міжнародних організацій та інші фактори, які вплинули на скорочення авіаперевезень.

2016 рік приніс з собою певний оптимізм, як показник відновлення української авіаційної галузі. З'явилися позитивні тенденції в показниках роботи авіакомпаній, а пасажиропотік почав показувати стабільність.

У 2015-2016 роках угорська авіакомпанія "*WizzAir*", ірландський "*Ryanair*" та інші лоукостери почали свою діяльність в Україні, відкриваючи нові маршрути. Це призвело до підвищення конкуренції на ринку та більш доступних цін для споживачів.

Однак, незважаючи на певне зростання, іноземні лоукостери до 2022 року були обережними щодо розширення своєї діяльності в Україні через економічні та політичні виклики.

2017 рік став важливим для авіаційної галузі України завдяки запровадженню безвізового режиму з ЄС. Це сприяло зростанню пасажиропотоків і стимулювало розвиток внутрішніх і зовнішніх авіамаршрутів.

Однак, починаючи з 24 лютого 2022 року, коли в Україні почався військовий конфлікт, авіаційна галузь знову зазнала значних втрат. Міжнародні авіалінії призупинили рейси до України, внутрішні авіаперевезення також зазнали великого удару.

В сучасних умовах, коли технологічні інновації та цифрова трансформація змінюють ландшафт бізнесу, інформаційне управління набуває особливої актуальності. Успішність будь-якої організації в значній мірі залежить від її здатності ефективно збирати, обробляти та аналізувати інформацію. Це не тільки дозволяє компаніям залишатися конкурентоспроможними, але й прогнозувати зміни ринку, швидко адаптуючись до них.

Інформаційне управління сприяє оптимізації робочих процесів, покращенню комунікацій між відділами і збільшенню продуктивності праці.

Крім того, воно дозволяє компаніям краще розуміти потреби своїх клієнтів та пропонувати їм продукти та послуги відповідно до їх очікувань.

Однак, важливо розуміти, що інформаційне управління не обмежується тільки технологіями. Це також включає культуру організації, навички співробітників та стратегічне планування. В ідеалі, інформаційне управління повинно бути вбудоване

в основу корпоративної культури, стаючи ключовим елементом прийняття рішень на всіх рівнях.

В умовах швидкої зміни технологічного оточення та зростання обсягів даних, інформаційне управління стає не просто важливим, але й критично необхідним компонентом успіху сучасних компаній. Тільки ті організації, які розуміють значущість та ефективно впроваджують принципи інформаційного управління, зможуть гарантувати свій стабільний розвиток і лідерство на ринку.

В цьому контексті основний об'єкт нашого дослідження є надання послуг авіакомпанією, оскільки авіаційна галузь є однією з тих, яка вимагає особливої уваги до інформаційного управління через її динамічність, високий рівень конкуренції та специфіку роботи. Предметом же дослідження є програмний модуль управління взаємодією з клієнтами для авіакомпанії. Розробка та оптимізація такої системи дозволить авіакомпанії ефективно використовувати наявні інформаційні ресурси, реагувати на зміни в зовнішньому середовищі вчасно та адаптуватися до нових ринкових умов.

Головною метою кваліфікаційної роботи є розробити програмний модуль управління взаємодією з клієнтами для авіакомпанії. Така система буде спрямована на вдосконалення збору, обробки та аналізу інформації, що, у свою чергу, дозволить компанії швидко реагувати на зміни ринкових умов та відповідати очікуванням клієнтів.

Для досягнення вказаної мети необхідно вирішити наступні конкретні завдання:

- 1) провести аналіз існуючої системи управління інформаційною діяльністю авіакомпанії, виявивши її сильні сторони та слабкі місця;
- 2) дослідити найкращі практики інформаційного управління в авіаційній галузі на світовому рівні;
- 3) визначити ключові показники ефективності (*KPI*) для системи управління інформаційною діяльністю;
- 4) визначити вимоги до програмного модуля;
- 5) описати його структуру;

- б) виконати програмну реалізацію модуля;
- 7) протестувати розроблений варіант;
- 8) описати процес роботи з модулем як користувача так і працівника авіакомпанії, який буде виконувати роль адміністратора.

Вирішення цих завдань допоможе створити ефективну, гнучку та адаптивну систему управління інформаційною діяльністю, яка сприятиме досягненню стратегічних цілей авіакомпанії та підвищенню її конкурентоспроможності на ринку.

Для збору і аналізу даних планується використовувати комплексний підхід, що поєднує як кількісні, так і якісні методи дослідження. Зокрема, буде використовуватися метод спостереження для вивчення реальних бізнес-процесів та діяльності співробітників у реальному робочому середовищі.

Анкетування та опитування співробітників авіакомпанії допоможуть отримати додаткову інформацію про їх взаємодію з інформаційною системою та виявити можливі проблеми. Для глибокого аналізу конкретних аспектів системи планується використовувати методи експертних оцінок та *SWOT*-аналізу. Кількісний аналіз даних буде проведено за допомогою статистичних методів та інструментів, таких як кореляційний аналіз, регресійний аналіз та інше. Дані планується візуалізувати за допомогою графіків, діаграм та інфографіки, що сприятиме кращому розумінню отриманих результатів та їх інтерпретації.

Актуальність полягає в тому, що авіаперевізники постійно конкурують, і що для задоволення клієнтів необхідно вдосконалювати системи управління відносинами з клієнтами. Зосереджуючись на універсальності та швидкості відповідно до міжнародних і національних стандартів програмування, цей дослідницький підхід до розробки програмного забезпечення є технічно правильним і інноваційним. Перевірка медичних стандартів пасажирів і використання *QR*-кодів є важливими компонентами, які можуть покращити безпеку та ефективність обслуговування.

Наукова новизна дослідження полягає в удосконаленні моделі системи управління інформаційною діяльністю авіакомпанії, що дозволить здійснити перехід на якісно новий рівень використання інформаційних ресурсів та інформаційного

потенціалу із застосуванням сучасних методів оброблення та аналізу інформаційних потоків. Зокрема, застосування *API* погоди, такого як *OpenWeatherMap API*, дозволить отримувати актуальні дані про погодні умови та використовувати їх для планування рейсів, зменшуючи ризик затримок або скасування, а також впровадження генерації *QR*-коду, в результаті позитивної відповіді при аналізі наданих користувачем медичних параметрів.

Кваліфікаційна робота складається з наступних основних розділів:

1) аналіз предметної області:

- розглядається специфіка управління взаємодією з клієнтами для авіакомпанії;
- вивчення існуючих рішень щодо управління інформаційною діяльністю авіакомпанії;
- основне завдання та цілі проектування системи.

2) вимоги до програмного модуля:

- визначення функціональних вимог до програмного модуля;
- опис структури програмного модуля;
- розробка архітектури програмного забезпечення.

3) програмна реалізація модуля:

- встановлення системних вимог до програмного модуля;
- розробка алгоритмів реалізації модулів;
- тестування роботи програмного модуля.

4) методика роботи користувача з модулем:

- визначення системних вимог і інструкції щодо інсталяції комплексу;
- опис роботи користувача з додатком для бронювання квитків;
- методика роботи адміністратора з додатком.

Кожен розділ завершується висновками, що підбивають основні результати та визначені напрямки для подальших досліджень.

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1. Аналіз систем управління взаємодією з клієнтами для авіакомпаній

1.1.1. Огляд предметної області та основні виклики та ризики

Система управління взаємодією з клієнтами є ключовим інструментом для авіакомпаній. Щоб залишатися конкурентоспроможними, авіакомпанії повинні не тільки забезпечувати високий рівень обслуговування, але й використовувати дані клієнтів для підвищення ефективності та прибутковості.

Авіакомпанії в теперішньому конкурентному середовищі завжди прагнуть підвищувати задоволеність своїх клієнтів. Одним з ключових інструментів досягнення цього є система управління взаємодією з клієнтами (*CRM*). Така система допомагає авіакомпаніям у процесах бронювання та продажу квитків, обслуговуванні клієнтів на борту, постобслуговуванні, а також в реалізації програм лояльності та маркетингових кампаній. Збір особистої інформації, історії покупок, відгуків та реакцій на маркетингові акції є важливим для забезпечення ефективної роботи *CRM*. Щоб система була ефективною, авіакомпаніям слід використовувати сучасні бази даних, інструменти для аналізу даних, а також взаємодіяти з клієнтами через різні інтерфейси, такі як веб-сайти та мобільні додатки. Однак є і виклики: забезпечення безпеки даних, швидка адаптація до змін на ринку і вимог клієнтів, а також інтеграція з іншими системами. Рекомендується регулярно оновлювати систему, інвестувати в технології, зосереджуватися на безпеці даних та дотриманні законодавства. В умовах сучасного ринку авіакомпаніям необхідно активно користуватися *CRM* для підвищення ефективності та прибутковості своєї діяльності [18].

В галузі авіації, де конкуренція є великою, системи управління відносинами з клієнтами (*CRM*) грають ключову роль. Однак існує ряд викликів та ризиків, які потребують уважного розгляду.

Забезпечення безпеки даних клієнтів є одним з головних пріоритетів, особливо в епоху цифровізації та онлайн-взаємодій. Кожного дня авіакомпанії збирають величезний об'єм інформації про своїх клієнтів, і це відкриває потенційні можливості для порушень конфіденційності, що може призвести до втрати довіри клієнтів та репутаційних ризиків.

Також у теперішньому динамічному ринку авіакомпаніям доводиться швидко адаптуватися до змін. Тренди, вимоги клієнтів та технологічні нововведення постійно змінюються, що вимагає від авіакомпаній негайної реакції, щоб не залишитися позаду конкурентів.

Забезпечення якісного обслуговування клієнтів на всіх етапах взаємодії також є ключовим викликом. Від моменту бронювання квитка до післяпольотного обслуговування, клієнти очікують високого рівня сервісу, що вимагає від авіакомпаній постійного моніторингу та удосконалення своїх процесів.

Останнім, але не менш важливим, є виклик інтеграції з іншими системами і технологіями. Авіакомпанії співпрацюють з різноманітними партнерами, від систем бронювання до аеропортових служб. Ця інтеграція потребує чіткої координації, аби забезпечити плавність робочих процесів і забезпечити максимальну вигоду для клієнтів.

1.1.2. Основні процеси взаємодії з клієнтами

Для успішної роботи авіакомпанії, основне завдання полягає в ефективній взаємодії з клієнтами. Відсутність систематичного підходу до кожного з етапів спілкування може призвести до втрати довіри та, в кінцевому рахунку, до фінансових збитків.

Бронювання та продаж квитків – це початковий етап взаємодії з пасажиром. З точки зору наукового аналізу, ключовим є створення інтуїтивно зрозумілого та швидкого інтерфейсу для бронювання, оптимізація алгоритмів вибору рейсів, а також інтеграція з платіжними системами. Інформаційні системи мають забезпечувати

безперебійність, захищеність даних користувачів та гнучкість при змінах умов бронювання.

Обслуговування клієнтів на борту літака вимагає високого рівня організації. Аналіз потреб пасажирів, таких як комфорт, безпека, харчування та розваги, визначає стратегію служби борту. Науковий підхід включає в себе дослідження оптимальних методів навчання екіпажу, аналіз зворотного зв'язку від клієнтів та вдосконалення процесів на борту (рис. 1.1).



Рис. 1.1. Ілюстрація процесів взаємодії з клієнтами в аеропорту та на борту літака

Постобслуговування зосереджується на аналізі зворотного зв'язку після подорожі. Це включає в себе вирішення скарг, компенсації за затримки та інші проблеми. Науковий підхід вимагає систематизації отриманої інформації, її аналізу з метою вдосконалення послуг та розробки методик реагування на виниклі ситуації [24].

Програми лояльності спрямовані на збереження та зміцнення відносин з постійними клієнтами. Науковий підхід включає в себе дослідження мотивів і потреб

клієнтів, розробку стимулюючих програм і відстеження їх ефективності за допомогою аналітичних інструментів.

Маркетингові кампанії вимагають глибокого аналізу ринку, сегментів цільової аудиторії та ефективних комунікаційних стратегій. Науковий підхід забезпечує базу для розробки кампаній, які враховують актуальні тренди, потреби клієнтів та можливості технологічних інновацій [20].

В усіх вищезазначених процесах важливим є постійний моніторинг, аналіз результатів та гнучкість у вдосконаленні системи взаємодії з клієнтом.

1.1.3. Необхідні дані та технології

В світі інформаційних технологій збір та аналіз даних стає важливим інструментом у роботі багатьох компаній, особливо у сфері послуг. Це дозволяє компаніям краще розуміти своїх клієнтів, прогнозувати їхні потреби та надавати якісніші послуги.

Особиста інформація – це базові дані про клієнта. Ім'я, контактні дані та дата народження допомагають формувати індивідуальний підхід до кожного пасажира. Інформаційні технології забезпечують зберігання, захист та швидкий доступ до цих даних. Сучасні системи баз даних забезпечують захищеність особистої інформації від несанкціонованого доступу та зловживань [21].

Історія покупок та польотів є ключовою для аналізу преференцій споживача. Це допомагає авіакомпаніям розробляти індивідуальні пропозиції, визначати популярні напрямки та оптимізувати графік рейсів. Інформаційні технології, такі як системи управління відносинами з клієнтами (*CRM*), дозволяють зберігати та аналізувати ці дані в реальному часі [21].

Відгуки та зворотний зв'язок – важливий інструмент удосконалення послуг. Інформаційні технології допомагають автоматизувати процес збору відгуків, класифікувати їх та визначати ключові області для вдосконалення [21].

Інформація про участь в програмах лояльності допомагає відстежувати активність користувачів, їхній рівень лояльності та відданість бренду. ІТ-системи дозволяють автоматизувати акціонерні програми, надавати користувачам переваги на основі їхньої активності та взаємодії з брендом (рис. 1.2).

Збір і аналіз даних є життєво важливими для покращення сервісу авіакомпаній. Реакція клієнтів на послуги може бути важливим показником якості, окрім особистої інформації та історій покупок. За допомогою інформаційних технологій відгуки та зворотній зв'язок дозволяють швидко та точно визначити сильні та слабкі сторони обслуговування.

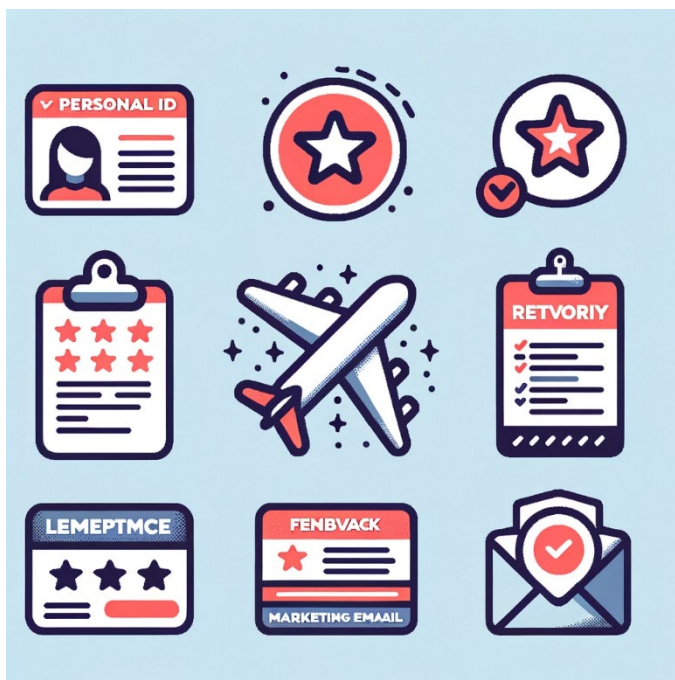


Рис. 1.2. Векторна графіка, яка відображає різні точки збору даних, пов'язаних з клієнтами авіакомпаній

Відповіді на маркетингові акції відображають ефективність рекламних кампаній. Інформаційні технології допомагають аналізувати ці відповіді в реальному часі, адаптувати маркетингові стратегії та зосереджувати увагу на найбільш результативних каналах комунікації (рис. 1.3).



Рис. 1.3. Система збору і обробки інформації

Інформаційні технології відіграють ключову роль у зборі, обробці та аналізі даних. Сучасні ІТ-рішення забезпечують гнучкість, швидкість та точність обробки великих обсягів інформації, що дає компаніям конкурентні переваги на ринку.

Для авіакомпаній сучасні технології стають основою ефективної роботи та конкурентноспроможності на ринку. Для використання у даній сфері можна розглянути наступні технології.

Існують такі бази даних для зберігання інформації про клієнтів:

- 1) *SQL*-базовані системи (наприклад, *MySQL*, *PostgreSQL*, *MS SQL Server*) – універсальні, надійні і широко використовувані в багатьох галузях;
- 2) *NoSQL* бази даних (*MongoDB*, *Cassandra*) – дозволяють гнучко реагувати на зміни в структурі даних, що може бути корисним для зберігання великої кількості динамічної інформації [19].

Переваги: гарантована консистентність даних, висока швидкість обробки запитів, можливість масштабування.

Інструменти для аналізу даних:

- 1) *AI* та машинне навчання: *TensorFlow*, *PyTorch*, *Scikit-learn*. За їх допомогою можна створювати моделі для прогнозування поведінки клієнтів, визначення попиту на рейси або персоналізації рекламних кампаній.

Переваги: автоматизація рутинних задач, високий рівень адаптації до даних, покращення якості сервісу.

Інтерфейси для взаємодії з клієнтами:

- 1) веб-сайти на основі сучасних фреймворків (*React, Angular, Vue.js*) для зручного бронювання та взаємодії з компанією;
- 2) мобільні додатки (*iOS, Android*) – забезпечують клієнтів інструментами для управління своїми рейсами, бронюванням та отримання повідомлень на ходу.

Переваги: більше точок контакту з клієнтом, підвищення лояльності, можливість збору додаткової інформації для аналітики.

Інтеграція з іншими системами:

- 1) *API (Application Programming Interface)* – дозволяє інтегрувати систему управління авіакомпанії з зовнішніми системами бронювання, партнерами, системами управління аеропортами [19].

Переваги: Автоматизація процесів, швидкість обміну даними між системами, збільшення обсягу продажів через зовнішні канали.

Всі ці технології та інструменти разом формують інтегровану систему управління для авіакомпанії, де кожен компонент виконує свою роль у створенні конкурентноспроможного та ефективного сервісу.

1.2. Варіанти управління інформаційною діяльністю авіакомпанії

1.2.1. Інформаційні системи в авіаційній галузі: огляд ключових технологій та їх застосування

Ключову роль у веденні бізнесу, особливо в такій високотехнологічній галузі як авіація відіграють інформаційні системи. Управління інформаційною діяльністю авіакомпанії включає в себе численні аспекти, від оптимізації роботи авіакомпанії до забезпечення безпеки польотів.

Системи резервування квитків (*GDS*). Більшість авіакомпаній використовують глобальні системи резервування (*GDS*), такі як *Amadeus, Sabre* або *Galileo*. Ці системи

дозволяють авіакомпаніям узгоджувати резервування квитків з агентствами, іншими авіакомпаніями та партнерами;

Системи управління польотами (*FMS*). Ці системи допомагають пілотам у навігації, плануванні маршрутів та оптимізації споживання пального;

Системи управління обслуговуванням пасажирів (*PSS*). Це комплекс програм, який дозволяє авіакомпаніям управляти резервуваннями, чекінами, посадкою та іншими процесами обслуговування пасажирів;

Системи безпеки даних. Охорона конфіденційності даних пасажирів та забезпечення інформаційної безпеки стали ключовими пріоритетами для авіакомпаній. Сучасні технології шифрування, брандмауери та антивірусні програми допомагають забезпечити безпеку інформації;

Системи аналізу даних. Збір та аналіз даних дозволяє авіакомпаніям оптимізувати свої рейси, ціноутворення та маркетингові стратегії [6].

Авіакомпанії використовують різноманітні інформаційні системи для оптимізації своєї роботи, підвищення ефективності та забезпечення безпеки. Щоб бути конкурентоспроможними на ринку, авіакомпаніям необхідно постійно оновлювати свої системи та впроваджувати нові технології.

1.2.2. Цифрова трансформація в авіаційній галузі: від технологій *IoT* до екологічної сталості

Згідно з Міжнародною Асоціацією Повітряного Транспорту, 2020 рік став найгіршим у історії авіаційної промисловості. Хоча багато учасників цієї галузі досі виходять з хаосу, завданого *COVID*, ця криза пришвидшила та вимагала технологічних змін, які вирішують ключові проблеми, з якими галузь зіштовхується у світі після *COVID* [29].

В рамках аналізу та спостережень за ІТ-тенденціями у ключових галузях, розглянемо одну з галузей, яка найбільше постраждала від пандемії – авіацію.

Давайте розглянемо тенденції в області інформаційних технологій цього року, на прикладі США, які пропонують новий підхід до польотів.

Розумні аеропорти завдяки цифровій трансформації. Авіаційна галузь була змушена пристосуватися до «нової норми». Що стосується аеропортів, це призвело до максимальної автоматизації та цифрової трансформації всіх процесів.

Наприклад, численні аеропорти вже модернізували процес реєстрації для мінімізації безпосереднього людського контакту. На сьогодні в багатьох аеропортах автоматизовані кіоски замінили традиційні стійки реєстрації, а цифрові індикатори спрямовують пасажирів під час посадки.

Інші приклади поглибленої цифровізації включають системи безпеки, передові технології сканування предметів та тіла, без-паперові посадкові талони та автоматизовані процеси посадки. Всі ці системи пов'язані за допомогою протоколів IoT та аналізуються за допомогою алгоритмів штучного інтелекту та машинного навчання, про що буде йти мова [29].

Технології *IoT* та хмарних обчислень. Авіаційна галузь, як і багато інших, активно переходить до хмарних технологій. Це не лише сприяє зменшенню вуглецевого сліду, який часто критикують, але й привносить численні практичні переваги.

По-перше, хмарні мережі слугують основою для безлічі пристроїв, що підтримують *IoT*, які наразі встановлюються в аеропортах. Можливість автономного відслідковування пасажирів протягом усього процесу реєстрації та посадки надає ряд переваг, таких як підвищена безпека.

По-друге, самі аеропорти є досить коштовними у веденні. Навіть невелике підвищення ефективності завдяки датчикам *IoT* у системах освітлення на основі *PoE* може призвести до значного зниження витрат [29].

Штучний інтелект та машинне навчання. Продовжуючи тему, згадані вище підключені пристрої (разом з даними, які вони передають до хмари) також створюють основу для алгоритмів штучного інтелекту, які аналізують трильйони терабайт даних, щоб видобути корисні висновки для операційної діяльності.

Тоді як лікарні можуть використовувати алгоритми на основі машинного навчання для поліпшення догляду за пацієнтами, аеропорти повинні застосовувати цю технологію для аналізу розходів у роздрібних магазинах, безпеки, маршрутів

подорожей, санітарних зон, стоянок, звичок при реєстрації та багато іншого. Такий всеосяжний аналіз може бути проведений одночасно лише за допомогою програм на основі штучного інтелекту та алгоритмів машинного навчання [29].

Однак це стосується не лише аеропортів. Виробництво авіаційної техніки також отримує переваги від висновків, отриманих за допомогою машинного навчання. Наприклад, *GE* використовує машинне навчання та аналіз даних для виявлення дефектів у двигунах, збільшуючи термін служби компонентів та зменшуючи витрати на обслуговування. *Boeing*, у свою чергу, успішно розробив алгоритми машинного навчання для проектування літаків та автоматизації заводських операцій.

Блокчейн. Більшість галузей активно досліджує можливості блокчейну, і авіаційна галузь не є винятком. Наразі найбільше використання блокчейну спостерігається у виробництві літаків [29].

Після скандалів, пов'язаних з *737 MAX*, *Boeing* активно впроваджує технологію блокчейну для зберігання незмінних і доступних для спільного використання записів про деталі літаків та системи під час виробництва та обслуговування. В результаті компанія може більш точно прогнозувати події, пов'язані з обслуговуванням, оптимізувати виробничі операції та продовжувати термін служби окремих деталей.

Airbus, натомість, впроваджує рішення на базі блокчейну у свої департаменти відстеження поставок та за купівель, сподіваючись використовувати цю технологію для прискорення та забезпечення транзакцій, а також підвищення ефективності на всіх етапах ланцюга поставок.

Сталість. Помітне поліпшення багатьох екологічних показників, що стало результатом того, що більшість світу зупинилася на більшу частину 2020 року, посилило тиск на авіаційну галузь з метою активнішої участі у боротьбі проти зміни клімату.

Інформаційні технології будуть на передовій цих змін і можуть мати такий же великий вплив, як і пошук більш екологічних біо-палив. Багато авіакомпаній, таких як *Southwest*, вже перейшли на без паперові квитки та продовжують вносити подібні поліпшення у свої оперативні дії та системи резервування, щоб зменшити споживання енергії та викиди вуглецю [30].

Це також стосується виробництва та експлуатації. Багато технологій, згаданих вище, поєднуються, щоб допомогти значно підвищити ефективність використання пального літаками, усунути відходи та декарбонізувати галузь.

1.3. Постановка завдання проектування

Для створення ефективної моделі системи управління інформаційною діяльністю авіакомпанії необхідно визначити ключові компоненти та параметри цієї системи. Перш за все, система повинна включати інтеграцію всіх джерел даних, від резервування квитків до служби підтримки пасажирів, для забезпечення безперебійного обміну інформацією. Додатково, система має бути гнучкою, щоб легко адаптуватися до змінних умов ринку та технологічних нововведень.

На основі аналізу існуючих рішень необхідно врахувати залучення сучасних технологій, таких як штучний інтелект, машинне навчання та блокчейн, для оптимізації процесів, підвищення ефективності роботи та безпеки даних.

Крім того, система повинна бути спроектована так, щоб підтримувати сталість та екологічні ініціативи компанії, спрямовані на зменшення вуглецевого сліду та оптимізацію використання ресурсів. В кінцевому підсумку, головною метою проектування є створення інтегрованої, надійної та адаптивної системи управління інформаційною діяльністю, яка допоможе авіакомпанії досягти стратегічних цілей і забезпечити високий рівень обслуговування пасажирів (рис. 1.4) [18].

Для створення інтегрованої системи управління інформаційною діяльністю авіакомпанії було обрано мову програмування *PHP*, доповнивши її використанням технологій *HTML* та *CSS*. Дана *CRM*-система буде базуватися на даних *MySQL*, представляючи веб-додаток, доступний для клієнтів (пасажирів) 24/7. Цей веб-додаток буде мати *Front-end* (користувацький інтерфейс), *Back-end* (адміністративну панель) та спеціальний розділ для реєстрації менеджерів авіакомпанії [18].

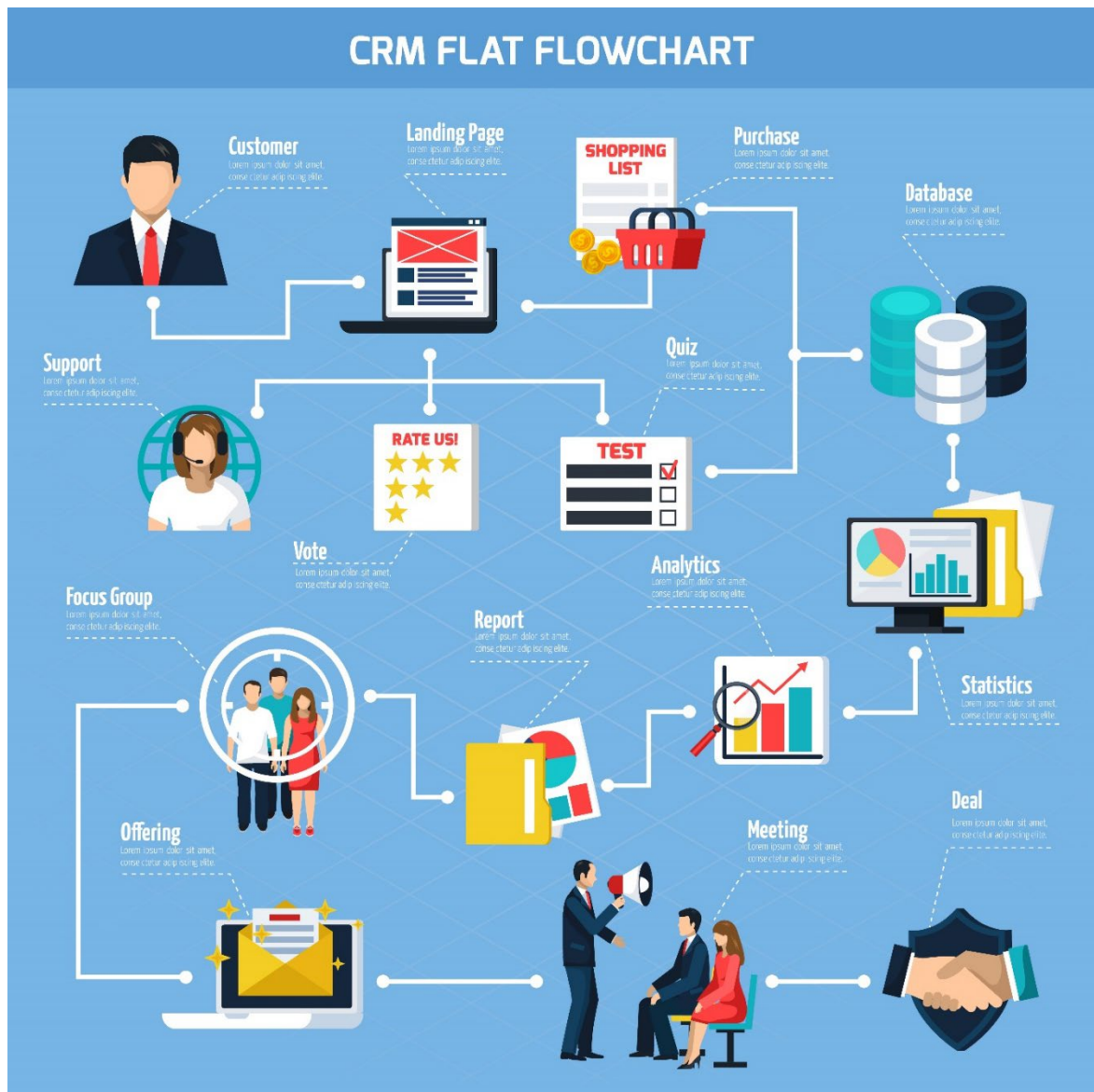


Рис. 1.4. Процеси CRM: від звернення клієнта до успішної угоди

Основна інновація системи полягає у введенні функції, яка дозволяє адміністратору додатку перевіряти стан здоров'я пасажирів перед вильотом. Моніторинг основних показників, таких як температура тіла та артеріальний тиск, дозволить пасажирів отримати QR-код, який може бути використаний для подальшого проходження паспортного контролю, якщо всі показники знаходяться у нормі. Такий підхід не лише оптимізує процес реєстрації та перевірки, але й сприяє забезпеченню безпеки пасажирів та екіпажу.

1.4. Висновки до розділу

В умовах сучасного ринку авіаперевезень авіакомпанії стикаються з численними викликами та можливостями. Збереження безпеки даних клієнтів є відсутнім пунктом, особливо в світлі зростаючих кіберзагроз і жорстких законодавчих вимог. Щоб залишатися конкурентоспроможними, авіакомпаніям потрібно швидко адаптуватися до змін у вимогах ринку, надаючи при цьому якісне обслуговування на всіх етапах взаємодії з клієнтом.

Інтеграція з іншими системами і технологіями також відіграє ключову роль, дозволяючи авіакомпаніям надавати послуги ефективніше та гнучкіше. Однак успіх у цій сфері залежить не тільки від технологій. Щоб досягти бажаних результатів, авіакомпаніям слід регулярно оновлювати свої системи, інвестувати в аналіз даних та зосереджувати свої зусилля на забезпеченні задоволення клієнтів.

З урахуванням цього, сучасні авіакомпанії повинні дотримуватися комплексного підходу до управління відносинами з клієнтами, інтегруючи передові технології, стратегії обслуговування та безпеки даних. Такий підхід дозволить підвищити лояльність клієнтів, забезпечити їхню задоволеність та забезпечити стабільний розвиток компанії в майбутньому.

Доповнивши вище сказане, можна зробити такі висновки:

- 1) авіаційна галузь активно рухається в напрямку цифрової трансформації, інтегруючи передові технології для оптимізації своєї діяльності та підвищення рівня обслуговування клієнтів;
- 2) вибір мови програмування *PHP* та технологій *HTML* та *CSS* для розробки *CRM*-системи, що базується на *MySQL*, дозволяє створити надійний веб-додаток, доступний пасажиром та персоналу авіакомпанії 24/7;
- 3) впровадження інноваційних функцій, таких як перевірка стану здоров'я пасажирів перед вильотом за допомогою *QR*-коду, підвищує безпеку польотів та спрощує процедуру перевірки пасажирів.

РОЗДІЛ 2

ВИМОГИ ДО ПРОГРАМНОГО МОДУЛЯ

2.1. Функціональні вимоги програмного модуля

2.1.1. Визначення основних функцій програмного модуля

Програмний модуль, що є основою кваліфікаційної роботи це фактично *CRM* для авіакомпанії. Тому в цьому розрізі варто його розглядати.

Основним завданням впровадження *CRM*-системи є ефективний менеджмент відносин з клієнтами та оптимізація бізнес-процесів. Цей інструмент набуває особливої важливості для підприємств, що спеціалізуються у сфері комерції чи наданні послуг і мають безпосередній контакт зі споживачами: електронні комерційні платформи, онлайн-освітні заклади, агентства з соціальних медіа маркетингу, туроператори, готелі та авіакомпанії [21].

CRM-система буде вам незамінною, якщо користувач стикається із:

- 1) збором та зберіганням великої кількості даних клієнтів;
- 2) необхідністю оптимізації бізнес-процесів та автоматизації продажу;
- 3) організацією та контролем діяльності численного персоналу;
- 4) потребою в докладній статистичній інформації щодо продажів [21].

CRM-системи використовуються не лише у великому та середньому бізнесі, а й у малому. Більшість таких систем є платними, проте існують і безкоштовні варіанти. Існують готові рішення, але зазвичай системи розробляються на замовлення.

Основні характеристики *CRM*-системи:

- 1) організована база даних клієнтів: всі відомості про клієнтів відображені у вигляді зрозумілих карток. Для кожного клієнта фіксуються історія звернень, суть переговорів, майбутні зустрічі, дзвінки та завдання. Базу даних у *CRM* можна сегментувати за різними критеріями. Автоматичний збір інформації зменшує ручне введення даних;

- 2) автоматизація: робота з *CRM* зменшує часові витрати та дозволяє автоматизувати повсякденні завдання. Поширені функції автоматизації включають масові розсилки та генерацію листів за шаблоном;
- 3) менеджмент роботи персоналу: в *CRM* можна встановлювати завдання, розподіляти їх між виконавцями, слідкувати за процесом виконання та визначати пріоритетність завдань;
- 4) менеджмент продажів та аналітика: можливість відстеження етапів продажів є однією з ключових переваг *CRM*-систем. Більшість *CRM* також мають можливість створення звітів та індивідуальних аналітичних функцій [21].

Основними ж функціями даного програмного модуля або *CRM* є додавання рейсів адміністратором, реєстрація пасажирів на рейс та новизна у вигляді перевірки артеріального тиску і температури тіла. Останні у нормальному стані дають пасажирові можливість отримати *QR*-код для подальшого проходження паспортного контролю.

2.1.2. Опис кожної функції, її призначення та основні характеристики

Відкрита частина веб-додатку:

1) вхід та реєстрація для адміністратора:

- призначення: дозволяє адміністратору отримати доступ до закритої частини веб-додатку;
- основні характеристики: ця функція має показовий характер, тому реальний доступ до реєстрації адміністратора повинен бути обмежений і прихований від звичайних користувачів.

2) доступні рейси:

- призначення: надає користувачам інформацію про наявні рейси;
- основні характеристики: відображення міста відправлення та призначення, а також вартості рейсу.

3) дані про здоров'я або перевірка основних норм пасажирів:

- призначення: забезпечує безпеку польоту шляхом збору і перевірки медичних даних пасажирів;
- основні характеристики: користувач може ввести необхідні медичні дані, після чого система генерує *QR*-код для швидкої ідентифікації.

4) модуль «Зареєструватись» для пасажирів літака:

- призначення: дозволяє пасажирам бронювати місця на обраний рейс;
- основні характеристики: користувач може обрати потрібний рейс, ввести свої особисті дані (ім'я, прізвище), контактну інформацію та залишити коментар щодо свого бронювання [21].

Закрита частина веб-додатку (для адміністраторів та менеджерів авіакомпанії):

1) список зареєстрованих клієнтів (пасажирів):

- призначення: надає менеджерам та адміністраторам повний список зареєстрованих пасажирів для кожного рейсу;
- основні характеристики: можливість перегляду деталізованої інформації про кожного пасажирів, його бронювання та коментарі.

2) можливість додавати рейс адміністратору або менеджеру авіакомпанії:

- призначення: дає можливість адміністрації компанії додавати нові рейси до системи;
- основні характеристики: інтерфейс для введення даних про рейс, включаючи місто відправлення та призначення, дату та час, вартість та інші деталі рейсу [20].

2.1.3. Взаємозв'язок між функціями та їх інтеграція в загальну систему

Взаємодія між модулями: функції «Доступні рейси» та «Модуль «Зареєструватись» для пасажирів літака» взаємодіють безпосередньо. Пасажир, переглядаючи доступні рейси, вибирає підходящий для себе і переходить до модулю реєстрації, де вводить необхідну інформацію;

Інтеграція даних про здоров'я: після реєстрації на рейс, дані про здоров'я пасажирів, зокрема *QR*-код, асоціюються з його бронюванням, що дозволяє адміністрації літака швидко отримати доступ до цієї інформації;

Інтеграція з закритою частиною: коли пасажир реєструється на рейс, його дані автоматично зберігаються в «Списку зареєстрованих клієнтів», до якого мають доступ адміністратори та менеджери авіакомпанії через свої акаунти;

Додавання рейсів: кожен рейс, доданий адміністратором або менеджером авіакомпанії, відображається у відкритій частині системи в розділі «Доступні рейси», що дозволяє потенційним пасажирів бачити наявність нових рейсів та реєструватися на них;

Узгодженість даних: всі дані, що вводяться або зберігаються в системі, синхронізуються в реальному часі. Це забезпечує одночасний доступ до актуальної інформації для всіх користувачів системи [6].

Таким чином, всі функції *CRM*-системи тісно взаємодіють між собою, створюючи єдиний механізм управління відносинами з клієнтами для авіакомпанії. Інтеграція різних модулів та функцій дозволяє забезпечити ефективний менеджмент, автоматизацію процесів та високий рівень обслуговування пасажирів.

2.1.4. Аналіз пасажирських авіаперевезень, вимоги до безпеки та надійності функцій, основні показники пасажиропотоку

Даний модуль розроблений з особливим акцентом на безпеку та надійність. Використовуючи передові захищені протоколи, ми гарантуємо конфіденційність та цілісність переданих даних. База даних, яка є основою для зберігання користувацької інформації, відмінно захищена від таких загроз, як *SQL*-ін'єкції, завдяки використанню сучасних механізмів та засобів захисту. Функціонал веб-додатку розроблений таким чином, що кожна його частина має вбудовані заходи безпеки, що забезпечують стабільність та надійність у роботі. Додатково, наша *CRM*-система сертифікована відповідно до високих стандартів безпеки, що підтверджує її

захищеність і надійність. Всі користувацькі дані обробляються в строгій відповідності до принципів безпеки, гарантуючи їхню конфіденційність.

Для розробки веб-додатку слід використовувати методологію системного проектування. Ця методологія призначена для створення та застосування інтелектуальних сфер, необхідних для швидкого вирішення поставлених завдань. Основна мета в цих сферах полягає в дослідженні та проектуванні складних об'єктів різного класу з підтримкою комп'ютерних пристроїв та за участю авіаційного експерта.

Завдання управління пасажирськими перевезеннями в авіаційному транспорті включають забезпечення необхідного рівня комфортності та рентабельності підприємства, що надає ці послуги. Таким чином, ефективне управління фінансовими та інформаційними потоками – створення оптимальних маршрутів – дозволяє пасажиром з комфортом дістатися потрібних їм аеропортових комплексів за короткий час і з мінімальними витратами.

Авіаційний транспорт має ряд переваг порівняно з іншими видами транспорту, такими як:

- 1) частота пасажирських перевезень, незалежно від сезону;
- 2) комфорт, що надається авіапасажиром;
- 3) можливість швидко подолати великі відстані – послугу, яку не може надати інший вид транспорту;
- 4) відносно низька собівартість перельоту [29].

Ці переваги зобов'язують до постійного вдосконалення пасажирського авіаційного транспорту та його розвитку. З вітчизняного та світового досвіду, авіа пасажирські перевезення в певні періоди року є економічно не вигідними і потребують дотацій з державного бюджету.

Оцінка пасажиропотоку повинна здійснюватися за такими параметрами, як напрямок, інтенсивність пасажиропотоку та аналіз його рівномірності.

Пасажиропотоки не є сталою величиною і часто бувають нерівномірними. За допомогою коефіцієнта нерівномірності можна оцінити варіативність

пасажиропотоку. Попит на авіап перевезення варіюється залежно від дня тижня та сезону.

Обсяг посадки та висадки в аеропортах дає узагальнену кількість авіапасажирів, що дозволяє передбачити завантаженість літака пасажирами та покращити управління пасажиропотоком в аеропорту.

Робота з авіап перевезенням пасажирів характеризується ,як загальна кількість відправлених $A_{\text{відпр}}$ та прибулих $A_{\text{приб}}$ авіапасажирів [6]:

$$A_{\text{заг}} = A_{\text{відпр}} + A_{\text{приб}}$$

Статистика щодо кількості відправлених та прибулих авіапасажирів використовується при розробці розкладу рейсів та для визначення необхідної площі терміналів аеропортів. Розрахунок кількості пасажирів, які вирушили в польоти, базується на даних про продаж квитків. Для оцінки кількості прибулих пасажирів зазвичай використовується припущення про те, що їх число еквівалентне кількості тих, хто відправився.

Чисельність пасажирів в аеровокзалі $A_{\text{аер}}$ утворюються шляхом додавання відправлених $A_{\text{відпр}}$, пасажирів, $A_{\text{приб}}$ прибулих та пасажирів, які проїхали транзитом $A_{\text{тр}}$ [6].

$$A_{\text{аер}} = A_{\text{відпр}} + A_{\text{приб}} + A_{\text{тр}}$$

Аби визначити пасажирообіг, потрібно знати загальний добуток чисельності авіапасажирів за різними курсами та знати відстань яку потрібно подолати. Відстань літаків по відокремлених курсах можливо визначити, як загальний добуток відстані пройденої даними літаками (N_1, N_2, \dots, N_n) які перебувають на деяких відрізках продовж L_1, L_2, \dots, L_n [6]:

$$\sum NL_{\text{пас}} = N_1L_1 + N_2L_2 + \dots + N_nL_n$$

Частка пасажиропотоку, яку обслуговують літаки, залежить від їхнього розміру; великі потоки, що не вимагають пересадки, обслуговуються великими, швидкими літаками. У літню пору пасажиропотік росте, тому рух літаків регулюється відповідно до сезону. Регулярність авіарейсів – ключове завдання авіатранспорту, при

цьому літаки слідують за розкладом без затримок. З розвитком онлайн-продажів авіакомпанії створили системи та додатки для зручності покупки квитків.

2.2. Структура програмного модуля

2.2.1. Визначення основних компонентів програмного модуля

Основними компонентами програмного модулю для взаємодії авіакомпанії з клієнтами є:

- 1) інтерфейс користувача: видима частина веб-додатку, яка забезпечує інтуїтивний доступ до усіх основних функцій системи для кінцевого користувача;
- 2) модуль реєстрації та аутентифікації: відповідає за створення акаунтів користувачів, а також їх перевірку під час входу до системи;
- 3) модуль управління рейсами: надає інструменти для додавання, редагування та видалення інформації про рейси;
- 4) модуль управління клієнтами: забезпечує зберігання та обробку даних про клієнтів, їхні бронювання та історію взаємодії з компанією;
- 5) база даних: містить три основні таблиці, *bookings* – дані про бронювання квитків, *clients* – дані про клієнтів, *flights* – дані про рейс (рис. 2.1);
- 6) модуль безпеки: забезпечує захист від зовнішніх та внутрішніх загроз, включаючи захист даних, шифрування і механізми протидії зловмисним діям.

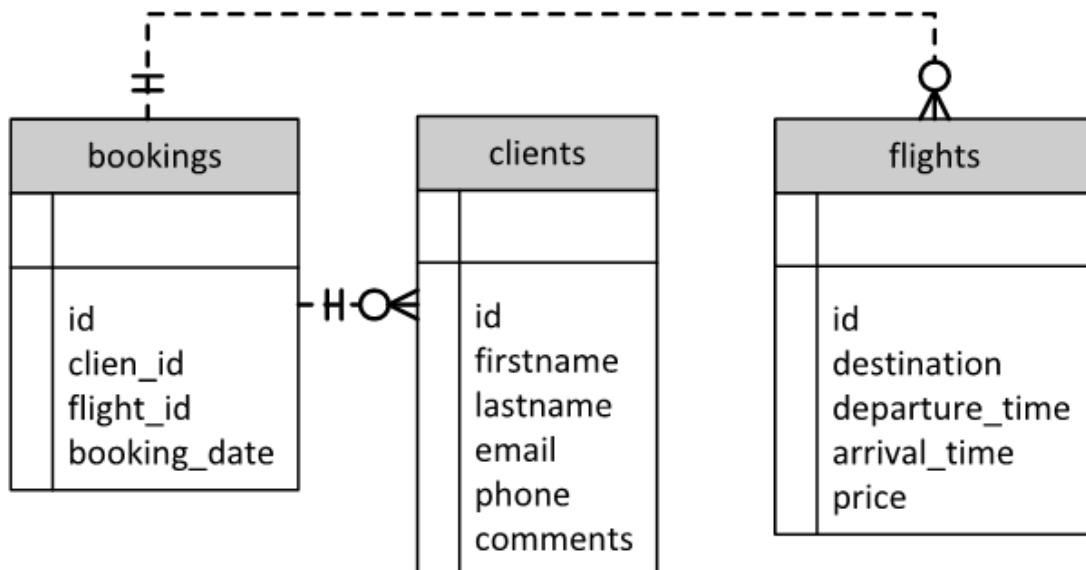


Рис. 2.1. Модель бази даних

Кожен з цих компонентів відіграє ключову роль у функціонуванні CRM-системи (рис. 2.2).



Рис. 2.2. Основні компоненти CRM авіакомпанії

2.2.2. Опис взаємодії між компонентами

Компоненти програмного модулю управління взаємодією з клієнтами авіакомпанії взаємодіють у сплетінні тісних зв'язків. Інтерфейс користувача, що є інтуїтивно зрозумілим, надає користувачу можливість легко ознайомлюватися та взаємодіяти з системою. Адаптивність шаблону забезпечує зручний доступ з

мобільних пристроїв і планшетів, дозволяючи користувачам спілкуватися з додатком з будь-якого місця і в будь-який час. Модуль реєстрації клієнтів тісно співпрацює з базою даних та модулем управління клієнтами, призначеним для адміністраторів. Модуль управління рейсами також взаємодіє з цими компонентами, створюючи гармонійний механізм управління. Втім, деякі елементи, такі як «Погодні умови», «Модуль автентифікації адміністраторів» та «Модуль перевірки медичних» показів, діють автономно від інших компонентів.

2.2.3. Визначення ієрархії та залежностей компонентів

Інтерфейс користувача - верхній рівень ієрархії, оскільки це перше, з чим зіштовхується користувач. Він надає доступ до всіх інших функцій та модулів системи. Інтерфейс адаптований до різних типів пристроїв, що забезпечує гнучкість взаємодії;

Модуль реєстрації та аутентифікації - основний засіб входу в систему, що взаємодіє з інтерфейсом користувача;

База даних - центральний компонент зберігання інформації, який має два основних розділи (рис. 2.2):

- дані адміністраторів: зберігає інформацію про адміністраторів та їх права доступу;
- дані про клієнтів та їх медичні показники: інформація, яка потрібна для управління рейсами і взаємодії з пасажиром.

Модуль управління рейсами: взаємодіє безпосередньо з базою даних та інтерфейсом користувача, надаючи інформацію про доступні рейси та дозволяючи адміністрації управляти цією інформацією;

Модуль управління клієнтами: також взаємодіє з базою даних, дозволяючи адміністрації переглядати та редагувати інформацію про клієнтів;

Модуль безпеки: працює на всіх рівнях системи, забезпечуючи надійний захист даних та функціоналу від зовнішніх та внутрішніх загроз.

Загалом, всі компоненти тісно пов'язані та взаємодіють один з одним, щоб забезпечити ефективну та безпечну роботу системи управління взаємодією з клієнтами авіакомпанії.

2.3. Архітектура програмного забезпечення

2.3.1. Опис основних принципів та підходів до побудови архітектури

Архітектура веб-додатку відіграє ключову роль у визначенні структури та взаємодії всіх його компонентів. Вона не лише описує розташування кожного з компонентів, але й акцентує увагу на способах взаємодії між ними, системами проміжного програмного забезпечення від сторонніх виробників, веб-службами та базами даних. Завдяки такому підходу можливе ефективне представлення взаємодії між численними застосунками, які спільно працюють над наданням послуг кінцевим користувачам.

Високо рівневі компоненти системи та їх взаємодії формують складові архітектури програмного забезпечення. Вони слугують основою для розробки та оптимізації додатку, гарантуючи його стабільність, продуктивність та масштабованість.

У свою чергу, дизайн програмного забезпечення зосереджується на більш детальному рівні – рівні коду. Його мета полягає в розподілі бізнес-логіки додатку між різними модулями, кожен з яких має свої специфічні завдання та цілі. Такий підхід не лише спрощує розробку та налагодження, але й забезпечує гнучкість управління бізнес-логікою додатку (рис. 2.3).

Також, варто зазначити, що архітектура веб-додатків є важливою для забезпечення масштабованості, безпеки та ефективності системи. Розподіл бізнес-логіки на окремі частини та їх взаємодію, щоб зосередитися на модульності, дозволяє підтримувати гнучкість і легкість розвитку додатку в майбутньому.

Вибір правильної моделі розгортання є важливою частиною успішної архітектури веб-додатку. Цей вибір залежить від масштабованості, доступності та

продуктивності системи. Створення надійної та ефективної системи можливо завдяки впровадженню різних технологій, включаючи веб-сервери та сучасні архітектури мікросервісів.

Контроль якості програмного забезпечення також є важливим компонентом. Тестування протягом різних етапів розробки дозволяє виявити помилки та вдосконалювати систему, що гарантує високу якість продукту. Зважаючи на те, що ефективна архітектура впливає не лише на структуру системи, але й на її надійність і ефективність у реальному використанні.

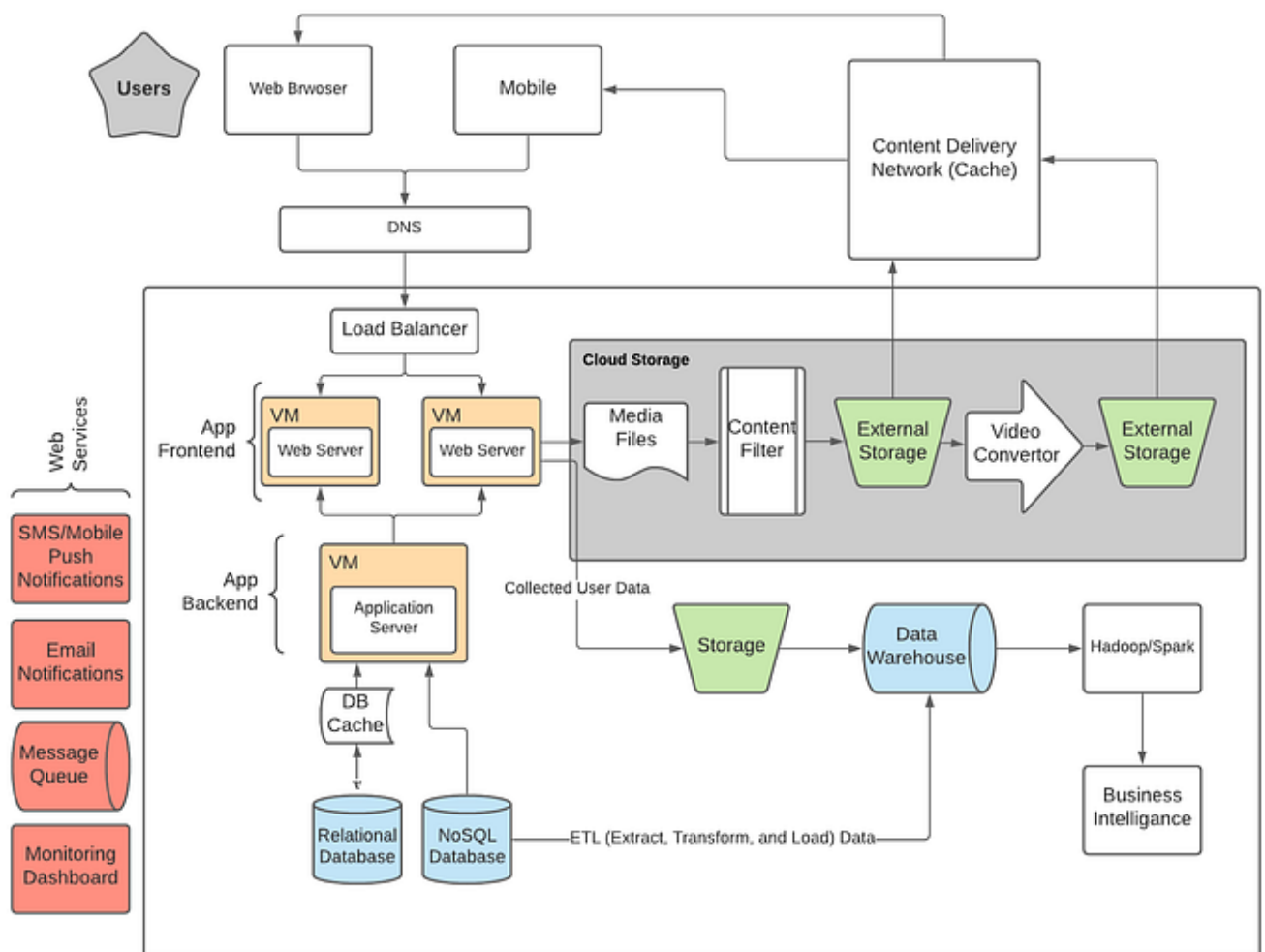


Рис. 2.3. Архітектура веб-додатку

Основні принципи та підходи до побудови архітектури програмного забезпечення відіграють ключову роль у розробці високоякісних програмних продуктів. При створенні архітектури було використано принцип процедурного або функціонального програмування, який дозволяє створювати чіткі алгоритми та

забезпечує легкість у підтримці та розширенні коду. Додатково, було прийнято рішення про розподілення даних між двома базами, що забезпечує оптимальну роботу з даними та високу продуктивність системи.

Хоча об'єктно-орієнтоване програмування є популярним підходом у розробці, для цього додатку воно не вважалося оптимальним, так як масштаб проекту не вимагав його застосування. Це вказує на те, що при розробці прийнято обдумане рішення, спрямоване на найкраще використання ресурсів та досягнення поставлених цілей.

Всі модулі та компоненти веб-додатку були створені з нуля, що гарантує унікальність коду та відсутність залежностей від сторонніх ресурсів. Це забезпечує стабільність роботи додатку та контроль над усіма його аспектами.

Архітектура базується на мові *PHP 7.2*, яка дозволяє створювати гнучкі, швидкі та надійні веб-додатки. *PHP 7.2* пропонує підвищену продуктивність, нові функціональні можливості та забезпечує сумісність з попередніми версіями. Система управління базами даних *MySQL 5.7* була обрана через її стабільність, високу продуктивність та широкі можливості оптимізації. *MySQL 5.7* також є однією з найбільш розповсюджених систем управління базами даних, що гарантує легкість підтримки та розширення.

2.3.2. Визначення ключових технологій та інструментів

Ключові технології (рис. 2.4):

- 1) *PHP 7.2* – сучасна серверна мова програмування, яка використовується для розробки веб-додатків. *PHP 7.2* пропонує підвищену продуктивність, новітні функції та оптимізацію для сучасних веб-застосувань;
- 2) *MySQL 5.7* – надійна і дуже розповсюджена система управління базами даних, яка використовується для зберігання, обробки та управління даними веб-додатку;

- 3) *HTML5* – стандарт мови розмітки гіпертексту, який забезпечує структуру веб-сторінок. *HTML5* пропонує новітні елементи, атрибути та поведінку, а також більшу підтримку для мультимедіа;
- 4) *CSS3* – мова стилів, яка використовується для оформлення веб-сторінок і забезпечення їх візуального представлення.

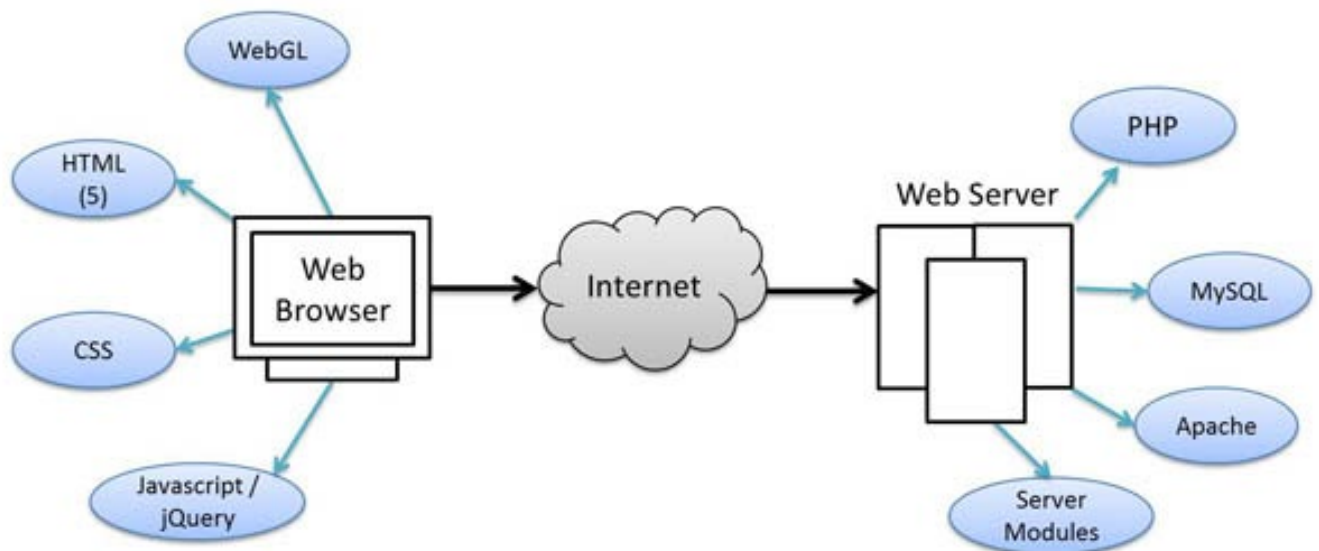


Рис. 2.4. Базовий набір веб-технологій

Інструменти:

- 1) *Notepad++* – простий у використанні текстовий редактор з підтримкою різних мов програмування. Він відомий своєю легкістю, швидкістю та гнучкістю;
- 2) *Visual Studio Code* – потужний відкритий редактор коду від *Microsoft*, який має розширений набір функцій, підтримку розширень та інтеграцію з різними системами контролю версій;
- 3) *OpenServer* – платформа для розробки та тестування веб-додатків на локальній машині. Вона включає в себе сервер, *PHP*, *MySQL* та інші компоненти, необхідні для роботи веб-сервера.

Операційна система:

- 1) *Windows 10* – надійна операційна система від *Microsoft*, яка широко використовується розробниками для створення, тестування та запуску програмного забезпечення.

Браузер:

1) *Google Chrome* – один з найпопулярніших веб-браузерів на ринку. Він відомий своєю високою продуктивністю, безпекою та масою розширень. *Google Chrome* часто використовується розробниками завдяки його девелоперським інструментам та сумісності з сучасними веб-стандартами.

Під час розробки веб-додатку було використано декілька додаткових ключових технологій та інструментів, які допомогли спростити процес створення та додали функціональність:

1) *Bootstrap.min.css* та *Bootstrap JS*:

- *Bootstrap* є найпопулярнішим фреймворком для розробки відгукуючих веб-сайтів. Він надає готові стилі та компоненти, які спрощують процес верстки, роблячи його швидшим та ефективнішим;
- *Bootstrap JS* додає динамічність до стандартних компонентів, таких як модальні вікна, вкладки та випадаючі меню. Це дозволяє розробникам легко інтегрувати анімацію та інтерактивність без необхідності писати велику кількість коду.

2) *Popper.js* є бібліотекою для управління спливаючими вікнами, спливаючими підказками та випадаючими меню. Він забезпечує динамічне позиціонування елементів відносно джерела або контейнера. Це особливо корисно для веб-додатків з відгукуючим дизайном;

3) *Google Fonts* – це безкоштовна служба, яка надає велику колекцію відкритих шрифтів. Використовуючи *Google Fonts*, розробники можуть легко додавати красиві та унікальні шрифти до своїх веб-додатків, покращуючи візуальний досвід користувача;

4) *OpenWeatherMap API* дозволяє розробникам інтегрувати інформацію про погоду в їхні веб-додатки. Цей *API* надає дані про поточний стан погоди, прогнози, історичні дані та інші погодні параметри для будь-якого місцезнаходження у світі. Його легко інтегрувати та налаштовувати згідно з потребами додатку [7].

2.3.3. Графічне представлення інтерфейсу веб-додатку

У візуальному інтерфейсі моєї *CRM*-системи розміщені добре структуровані модулі, які адаптивно та логічно розташовані на сторінці додатку, забезпечуючи інтуїтивно зрозуміле користувацьке середовище (рис. 2.5).

У верхньому лівому куті розташована кнопка «Вхід для адміністраторів», яка забезпечує швидкий доступ до панелі управління системою. Нижче від неї розташована шапка додатку, де можна побачити логотип та назву *CRM*.

Основна частина інтерфейсу поділена на дві секції: зліва розміщено модуль «Доступні рейси», де користувачі можуть переглядати актуальну інформацію про польоти, а справа – розділ «Дані про здоров'я», де ми збираємо інформацію про медичний стан пасажирів.

В нижній частині інтерфейсу розташована форма «Реєстрація для пасажирів», що дозволяє новим користувачам швидко зареєструватися в системі та отримати доступ до всіх її функцій.

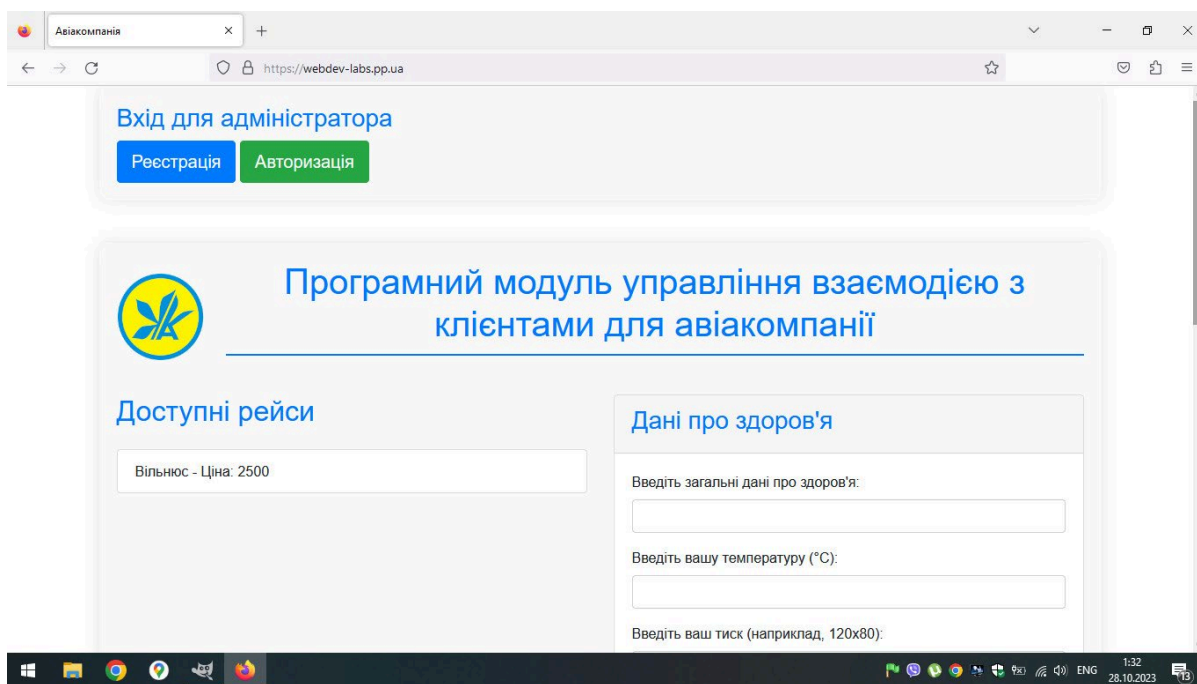


Рис. 2.5. *Front-end* або видима частина веб-додатку (*CRM*)

Для забезпечення безпеки та ефективної організації роботи в системі управління, ми надаємо можливість окремої реєстрації для адміністраторів та

менеджерів. Це дозволяє контролювати доступ до ключових функцій системи та розподіляти обов'язки між різними рівнями управління.

Форма реєстрації розроблена так, щоб бути максимально зручною та інтуїтивно зрозумілою (рис. 2.6). Вона включає в себе наступні поля для заповнення:

- 1) ім'я - повне ім'я користувача, що допомагає ідентифікувати його в системі;
- 2) логін - унікальний ідентифікатор, який буде використовуватися для входу в систему;
- 3) *e-mail* - електронна адреса, яка може використовуватися для спілкування, відновлення доступу та отримання повідомлень від системи;
- 4) пароль - надійний пароль для захисту акаунта від несанкціонованого доступу;
- 5) аватар - можливість завантажити зображення розміром 100×100 px, яке буде використовуватися як профільне фото користувача в системі.

Особливу увагу слід звернути на адаптивність форми. Вона автоматично налаштовує свій розмір та оформлення в залежності від розміру екрану пристрою, на якому вона відкривається, забезпечуючи зручну можливість користування як на настільних комп'ютерах, так і на мобільних пристроях.

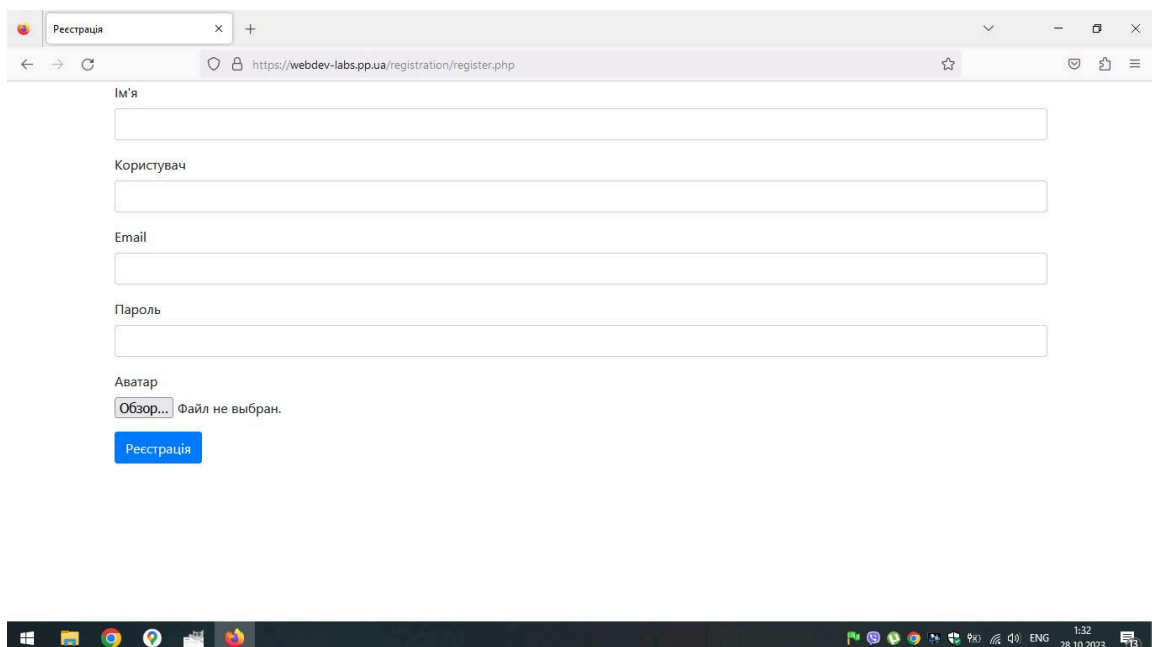


Рис. 2.6. Реєстрація адміністраторів та менеджерів системи управління

Адміністративна панель веб-додатку, яка служить центром управління та моніторингу, представляє собою комплексний інтерфейс із такими основними розділами (рис. 2.7):

- 1) дані про погодні умови: ця секція надає актуальну інформацію про погодні умови, що може бути корисною для планування рейсів та інформування клієнтів. Відображаються дані про температуру, вітряність, опади та інші погодні параметри;
- 2) список клієнтів: у цьому розділі адміністратор може переглядати повний список зареєстрованих клієнтів, їх контактні дані, історію бронювань та іншу інформацію, яка важлива для взаємодії з користувачами;
- 3) можливість додати рейс і ціну: тут адміністратори та менеджери можуть легко додавати нові рейси, вказуючи деталі такі як місце відправлення, призначення, дату та час вильоту. Додатково, вони можуть встановлювати або змінювати ціни на квитки, враховуючи різні фактори, такі як попит, сезонність та акційні пропозиції.

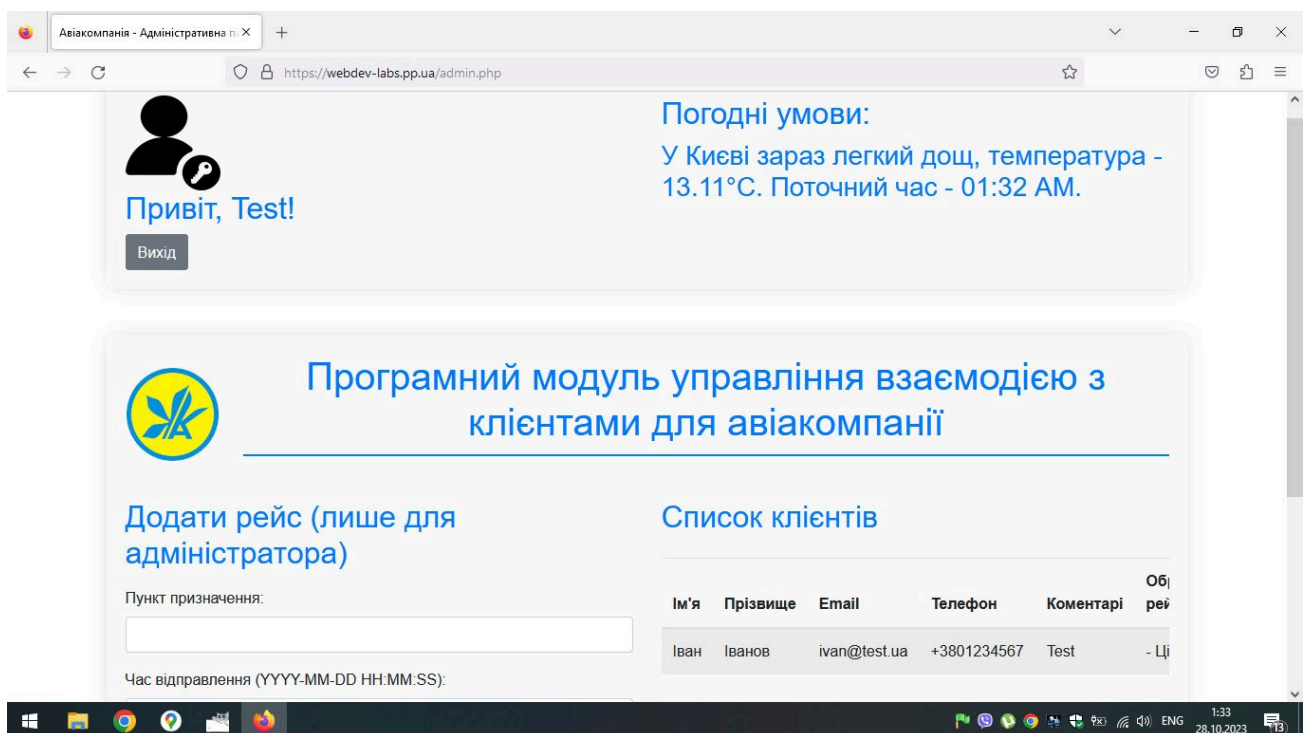


Рис. 2.7. Адміністративна панель веб-додатку

2.3.4. Оцінка ефективності та гнучкості вибраної архітектури

При розробці програмного забезпечення вибір відповідної архітектури є критично важливим, адже від цього залежить не тільки швидкість та стабільність роботи додатку, але й його адаптивність до змін у майбутньому. Наша архітектура продемонструвала високу продуктивність при обробці великих обсягів даних та одночасних запитів користувачів. Це було досягнуто завдяки оптимальному вибору технологій та методологій програмування, а також завдяки правильному розподілу ресурсів між модулями.

Архітектура додатку була розроблена з урахуванням потенційних змін та розширень, що забезпечує її високу гнучкість. При необхідності додати новий модуль або функціональність, система може легко адаптуватися без потреби глобальних змін в коді.

Завдяки розподіленій архітектурі та використанню сучасних технологій, наш додаток може легко масштабуватися, відповідаючи зростаючим потребам користувачів та збільшенню обсягу даних. Додатково, розроблена архітектура забезпечує відмінну підтримку відновлення після збоїв та відмов, завдяки методам резервного копіювання, балансування навантаження та іншим.

Оскільки технологічний ландшафт постійно змінюється, важливо, щоб архітектура програмного забезпечення могла адаптуватися до нових інструментів, фреймворків та стандартів. Наша архітектура була розроблена з цією метою, що дозволяє нам залишатися актуальними та конкурентоспроможними на ринку.

2.4. Висновки до розділу

В даному розділі аналіз та виконання були зосереджені на визначенні основних функцій програмного модуля *CRM*-системи. Система стане незамінним інструментом для тих, хто стикається з великою кількістю даних клієнтів, потребою оптимізації бізнес-процесів, а також організацією роботи численного персоналу.

Також докладно було описано кожен з функцій, її призначення та ключові характеристики. Зокрема, було висвітлено модулі відкритої частини веб-додатку, такі як реєстрація для адміністратора, перегляд доступних рейсів, дані про здоров'я та реєстрація для пасажирів. Також була розглянута закрита частина веб-додатку, призначена для адміністраторів та менеджерів авіакомпанії.

У контексті архітектури програмного забезпечення, визначено основні принципи та підходи до її побудови, а також описані ключові технології та інструменти, які були використані під час розробки.

Звернувши увагу на взаємозв'язок та інтеграцію функцій у загальну систему, забезпечено високий рівень зручності та інтуїтивності інтерфейсу користувача. Компоненти програмного модулю взаємодіють у сплетінні тісних зв'язків, створюючи гармонійний механізм управління.

В кінці розділу проведено оцінку ефективності та гнучкості вибраної архітектури, підтвердивши її відповідність сучасним вимогам та викликам ринку.

Крім того в цьому розділі успішно визначено основні функції програмного модуля для *CRM*-системи. Ця система виявилася незамінною для оптимізації роботи з великою кількістю даних клієнтів, автоматизації продажів, організації роботи численного персоналу та збору докладної статистичної інформації щодо продажів.

Надано докладний опис кожної функції веб-додатку, її призначення та основні характеристики. Зокрема, відкрита частина веб-додатку включає в себе модулі для входу та реєстрації адміністратора, перегляд доступних рейсів, збір даних про здоров'я пасажирів та їх реєстрацію. Тоді як закрита частина додатку, призначена для адміністраторів та менеджерів авіакомпанії, включає в себе список зареєстрованих клієнтів, можливість додавання рейсів та інші адміністративні функції.

Особливу увагу приділено структурі програмного модуля та його архітектурі. Визначено основні компоненти програмного модуля та їх взаємодію, а також розглянуто основні принципи та підходи до побудови архітектури, які були використані під час розробки *CRM*-системи.

Незважаючи на успішне виконання завдань, поставлених у цьому розділі, було виявлено деякі проблеми та виклики, які вимагали адаптивного підходу та гнучкості

в рішеннях. Однак завдяки науковому підходу та використанню сучасних технологій досягнуто високої ефективності та надійності системи управління.

На основі проведеної роботи можна дати наступні рекомендації та визначити напрямки для подальших досліджень.

Оптимізація бази даних є одним з ключових аспектів, який допоможе забезпечити більш швидкий доступ до інформації та підвищити продуктивність системи. Рекомендується розглянути можливість інтеграції *CRM*-системи з іншими бізнес-системами, такими як *ERP*, *SCM* та інші, для розширення її можливостей.

Оскільки технології постійно розвиваються, важливо регулярно оновлювати *CRM*-систему, додавати нові функції та вдосконалювати існуючі. Особлива увага має бути приділена безпеці даних, зокрема за допомогою шифрування, двоетапної автентифікації та інших методів захисту.

Враховуючи поширеність мобільних пристроїв, рекомендується розробити мобільний застосунок або оптимізувати веб-додаток для користувачів мобільних пристроїв. Щоб зрозуміти потреби користувачів та вдосконалити систему, рекомендується провести дослідження користувацького досвіду та отримати зворотний зв'язок від клієнтів.

Компоненти програмного модулю управління взаємодією з клієнтами авіакомпанії взаємодіють у сплетінні тісних зв'язків, і для їх оптимізації рекомендується використовувати інструменти аналітики для збору та аналізу даних про користувачів.

В заключення, попри успіхи в розробці *CRM*-системи, завжди є можливість для її вдосконалення та розширення. Рекомендується постійно слідкувати за новими технологіями та тенденціями в галузі, щоб забезпечити конкурентоспроможність та ефективність системи.

РОЗДІЛ 3

ПРОГРАМНА РЕАЛІЗАЦІЯ МОДУЛЯ

3.1. Системні вимоги програмного модуля

3.1.1. Апаратні вимоги

Вибір сервера для веб-додатку залежить від низки факторів, включаючи очікувану кількість користувачів, які працюють одночасно, особливості проекту та дані, отримані під час навантажувального тестування. Для модуля управління взаємодією з клієнтами авіакомпанії після проведення навантажувальних тестів були розроблені певні конфігурації сервера, які забезпечують його стабільну роботу. Під час тестування використовувалися сервери з операційною системою *Windows Server 2012 Standard*. Важливо відзначити, що максимальне споживання пам'яті залежить від конкретного додатку і може варіюватися в залежності від його специфіки. Таким чином, для кожного проекту необхідний індивідуальний підхід при виборі сервера та налаштуванні його параметрів (табл. 3.1).

Таблиця 3.1

Характеристика обладнання

| Назва обладнання | Характеристики |
|------------------------|-------------------------------|
| Кількість користувачів | 250 |
| Процесор | 4 ядра, 2.9 ГГц та вище |
| Платформа | 32-х або 64-х розрядна |
| Оперативна пам'ять | 10 Гб та вище |
| Жорсткий диск | 300 Мб вільного об'єму і вище |

3.1.2. Програмні вимоги

Необхідні програмні вимоги :

1) веб-сервер: приймає *HTTP*-запити від клієнтів (зазвичай, веб-браузерів) і відповідає їм *HTTP*-відповідями, часто разом з *HTML*-сторінкою, зображенням, файлом, медіа-потокком або іншими даними (рис. 3.1). Термін «веб-сервер» може відноситися як до програмного забезпечення, так і до комп'ютера, на якому воно встановлено.

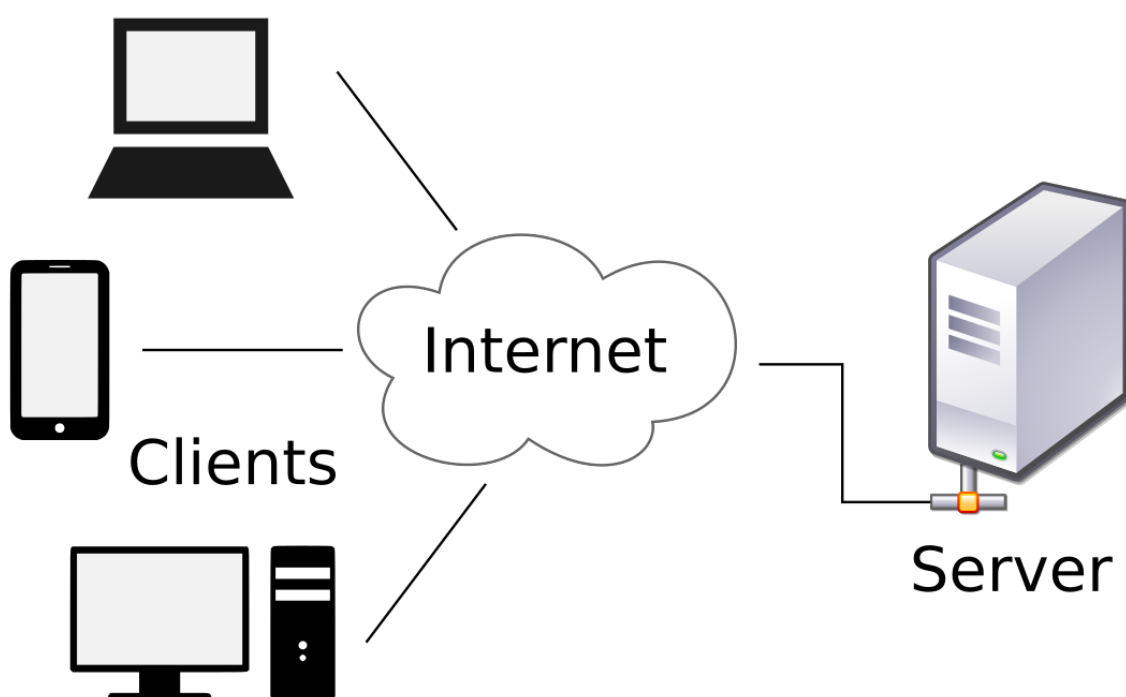


Рис. 3.1. Клієнт-серверна архітектура

2) програмне забезпечення веб-сервера (рис. 3.2):

- *Apache*. На 2022 рік це найбільш популярний веб-сервер, який займає більше 22% ринку. Він є вільним і найчастіше використовується в *UNIX*-подібних операційних системах;
- *Nginx*. Популярний веб-сервер, який часто використовується як зворотній проксі та для роздачі статичних файлів;
- *Lighttpd*. Вільний веб-сервер, розроблений з урахуванням швидкості, безпеки, відповідності стандартам і малого розміру. Важче

налаштовувати в порівнянні з іншими. Часто використовується в вбудованих системах, наприклад, в маршрутизаторах:

- *LiteSpeed Web Server*;
- *IIS*: Веб-сервер від *Microsoft* для *Windows*;
- *HTTP File Server*: Вільний файловий веб-сервер з графічним інтерфейсом.

3) операційні системи. Підтримуються такі версії операційних систем сімейства *Windows*:

- *Windows 7 Service Pack 1*;
- *Windows 8*;
- *Windows 8.1*;
- *Windows 10*;
- *Windows Server 2008 R2 Service Pack 1*;
- *Windows Server 2008 Service Pack 2*;
- *Windows Server 2012*;
- *Windows Server 2012 R2*;
- *Windows Server 2016*.

Підтримуються наступні версії операційних систем сімейства *Linux*:

- *Astra Linux*;
- *ALT Linux*;
- *CentOS*;
- *Debian*;
- *Red Hat Enterprise Linux*;
- *Ubuntu*.

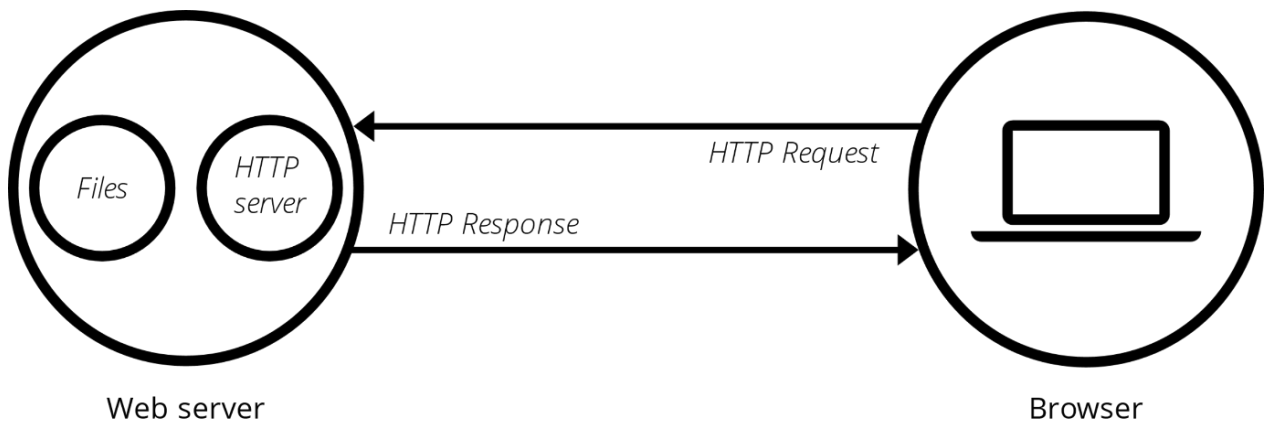


Рис. 3.2. Представлення роботи веб-сервера

Програмний модуль сумісний з переліченими операційними системами та веб-серверами, щоб забезпечує його правильну роботу і взаємодію з клієнтами.

3.1.3. Мережеві вимоги

Мережеві вимоги до модуля управління взаємодією з клієнтами авіакомпанії базуються на підтримці протоколів передачі даних: *HTTP* та *HTTPS*.

Вимоги до каналу зв'язку залежать від апаратних і програмних характеристик та передбачуваної кількості одночасно працюючих користувачів, і визначаються на підставі результатів навантажувального тестування проекту.

При одночасній роботі з типовим проектом 100 користувачів, необхідна пропускна здатність каналу сервера веб-додатка становить:

- 1) мінімальна: 15 Мбіт/сек;
- 2) рекомендована: 20 Мбіт/сек.

Планування масштабування системи також є важливою частиною. Попереднє передбачення потенційного збільшення кількості користувачів і обсягу даних дозволить підготувати інфраструктуру для можливого росту без значного зниження продуктивності. Це стосується архітектурних рішень, які сприяють легкому масштабуванню, таких як використання контейнеризації та мікросервісної архітектури. Такий метод дозволить ефективно реагувати на зростання навантаження та гарантувати, що система продовжить працювати без значних втрат продуктивності.

Залежність пропускної здатності каналу від кількості користувачів є лінійною (рис. 3.3).

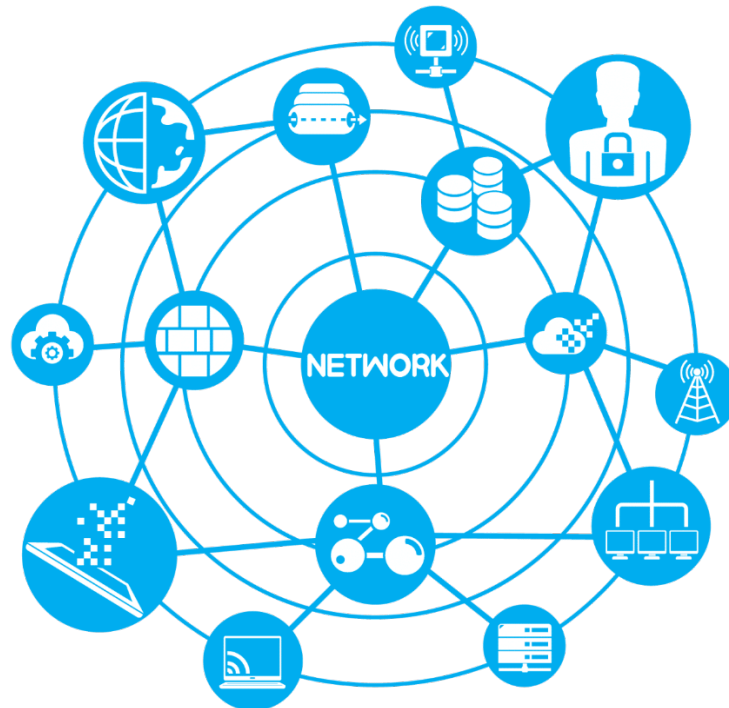


Рис. 3.3. Мережеві можливості доступу до CRM-системи

Для забезпечення оптимальної продуктивності та надійності системи, рекомендується також розглянути наступні аспекти:

- 1) використання балансувальника навантаження для розподілу трафіку між декількома серверами веб-додатка, що дозволить збільшити загальну пропускну здатність системи та забезпечити її стійкість до відмов;
- 2) оптимізація розміру даних, що передаються, зокрема зменшення розміру зображень, стилів і скриптів, які завантажуються на сторінку. Це можна зробити за допомогою інструментів компресії та мініфікації;
- 3) використання кешування на рівні сервера та клієнта для зменшення кількості запитів до сервера та прискорення завантаження веб-сторінок;
- 4) моніторинг стану мережі та серверів в реальному часі для швидкого виявлення та усунення можливих проблем.

Врахування цих аспектів допоможе забезпечити стабільну та ефективну роботу веб-додатка, навіть при великій кількості одночасних користувачів.

3.1.4. Вимоги до безпеки

Можемо виділити такі вимоги до безпеки:

- 1) шифрування даних: повинно забезпечуватися надійне шифрування всіх даних, що передаються і зберігаються. Важливо використовувати сучасні алгоритми шифрування, такі як *AES-256*, для забезпечення високого рівня захисту інформації. Всі ключі шифрування повинні зберігатися в безпечному сховищі;
- 2) аутентифікація користувачів: повинно вимагатися від користувачів пройти процес аутентифікації, перш ніж надавати доступ до будь-яких функцій управління взаємодією з клієнтами. Система аутентифікації повинна підтримувати багатофакторну аутентифікацію, включаючи, наприклад, паролі, *OTP* (одноразові паролі), і біометричні дані;
- 3) захист від зовнішніх атак: необхідно мати можливість виявляти і блокувати спроби зовнішніх атак, таких як *DDoS* (розподілена атака типу «відмова в обслуговуванні»), *SQL* ін'єкції, кросс-сайтовий скриптинг та інші. Для цього можна використовувати фаєрволи, системи виявлення та запобігання вторгненням (*IDS/IPS*), а також інші інструменти безпеки. Також важливо регулярно оновлювати всі компоненти системи, щоб вони включали останні патчі безпеки.

Зважаючи на наявні вимоги до безпеки, важливо також подумати про питання контролю доступу. У системі повинні бути чітко визначені права кожного рівня користувача на доступ до різних функцій і даних. Це включає обов'язкову перевірку доступу до критичних операцій і обмеження доступу до конфіденційної інформації лише для авторизованих осіб.

Для підвищення безпеки системи також необхідні механізми реагування на події безпеки. Це включає спостереження та реагування на підозрілу активність, виявлення аномальних патернів у поведінці користувачів і автоматичні заходи безпеки в разі виявлення загроз.

Важливо пам'ятати, що безпека системи — це процес, який триває постійно. Регулярні аудити безпеки, оновлення процедур і аналіз останніх загроз є важливими для ефективного функціонування та захисту даних.

3.2. Алгоритми реалізації модулів

3.2.1. Опис алгоритмів обробки запитів клієнтів

Алгоритм обробки запитів клієнтів у системі бронювання квитків працює таким чином:

- 1) вибір рейсу : клієнт вводить свої переваги, такі як пункт відправлення, пункт призначення, дати та інші можливі критерії. Система шукає доступні рейси, щоб знайти відповідний;
- 2) бронювання квитків: якщо підходящий рейс знайдений, клієнт може приступити до резервування квитків. Це включає вибір конкретного рейсу, введення даних пасажирів, вибір місць та здійснення платежу;
- 3) скасування бронювання: якщо клієнт бажає скасувати бронювання, система витягує деталі бронювання клієнта, підтверджує критерії скасування (як-от строки та штрафи) та обробляє скасування, якщо це можливо;
- 4) запит інформації: для клієнтів, що запитують інформацію про рейси без бронювання, система надає деталі про розклад рейсів, наявність місць, ціни та іншу відповідну інформацію [32].

Якщо рейс не доступний або не відповідає критеріям клієнта, система буде:

- 1) інформувати клієнта про недоступність маршруту, якщо він не існує;
- 2) повідомляти клієнта про недостатню ціну (можливо, пропонуючи альтернативи), якщо маршрут існує, але не може бути заброньований в межах бюджетних обмежень клієнта [18].

Цей алгоритм забезпечує систематичний підхід до обробки взаємодій з клієнтами у системі бронювання (рис. 3.4).

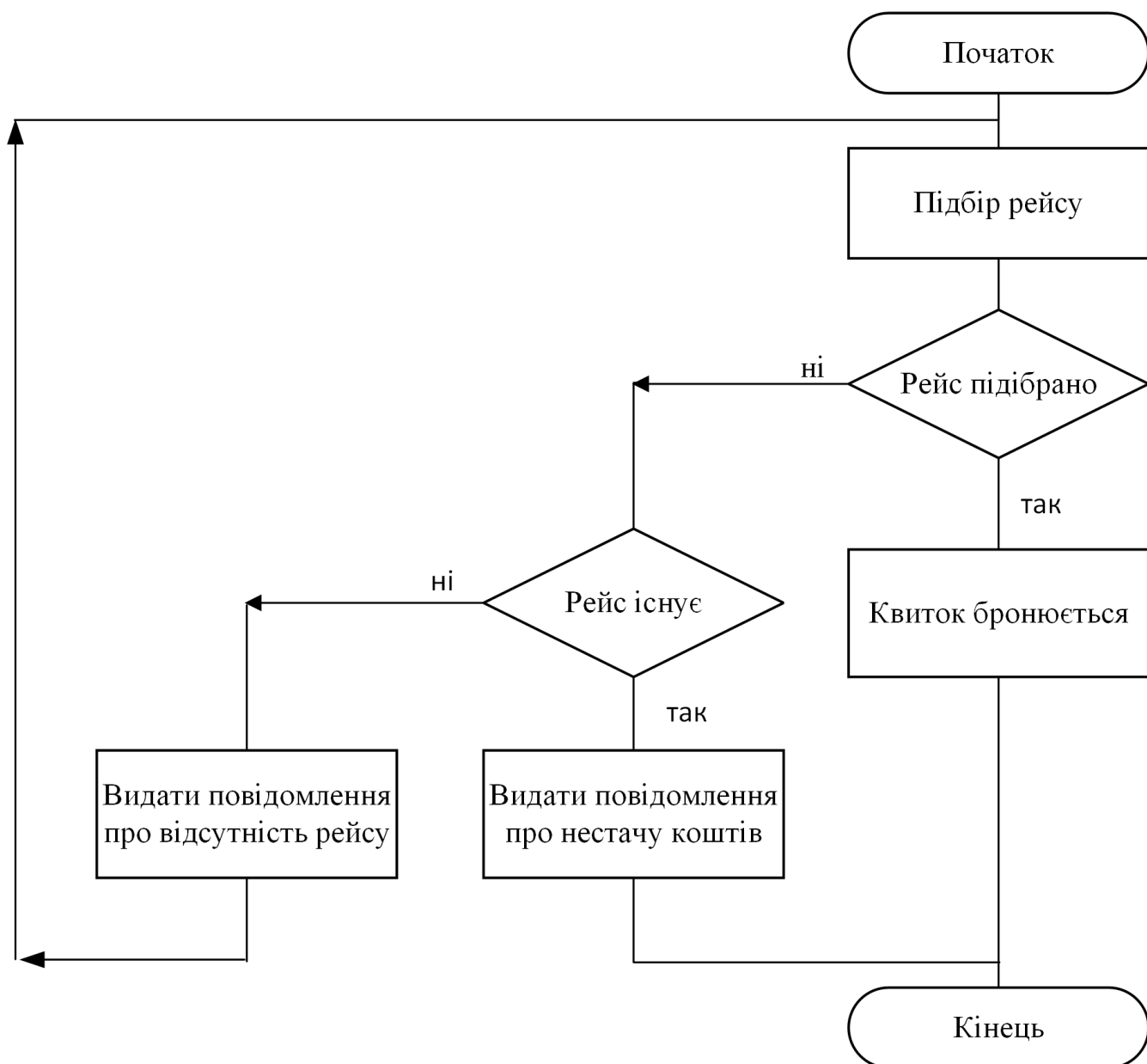


Рис. 3.4. Схема алгоритму обробки запитів клієнтів

3.2.2. Ключові алгоритми

Реалізація модулів системи бронювання квитків включає кілька ключових алгоритмів:

1) пошук оптимальних рейсів:

- вхідні дані: переваги користувача щодо подорожі (дати, пункти призначення тощо);

- процес: система оцінює доступні рейси, використовуючи алгоритми, що можуть включати параметри: ціна, тривалість, пересадки та переваги до авіакомпаній;
- результат: список рейсів, які найкраще відповідають критеріям користувача, відсортований згідно з параметрами оптимізації.

2) розрахунок вартості квитків:

- вхідні дані: деталі обраного рейсу, клас обслуговування та будь-які додаткові послуги;
- процес: система розраховує загальну вартість, враховуючи базовий тариф, податки, доплати, знижки та додаткові послуги (як-от додатковий багаж або страхування);
- результат: загальна ціна, яку повинен сплатити клієнт.

3) взаємодія з платіжними системами:

- вхідні дані: інформація про оплату від клієнта;
- процес: система безпечно передає дані про оплату до платіжного шлюзу. Потім вона чекає на підтвердження або відмову у транзакції, при цьому граційно обробляючи можливі помилки або відхилення;
- результат: підтвердження бронювання з видачою квитка або повідомлення про невдачу з поясненням причини [6].

Ці алгоритми працюють у координації для забезпечення безшовного досвіду бронювання для клієнта, одночасно гарантуючи ефективність і точність транзакції.

3.2.3. Опис використаних бібліотек та фреймворків для реалізації функцій модуля

Для реалізації функціональності модуля в системі бронювання квитків використовуються такі бібліотеки та фреймворки, як *Composer*, *Bootstrap*, *jQuery* і *Popper.js*.

Composer слугує менеджером залежностей для *PHP*, який дозволяє легко встановлювати та оновлювати бібліотеки, що є необхідними для проекту.

Файл *bootstrap.min.css* є мінімізованою версією CSS фреймворку *Bootstrap*, який забезпечує швидку та легку стилізацію веб-сторінок із готовими компонентами інтерфейсу.

jQuery.min.js - це мінімізована версія бібліотеки *jQuery*, яка спрощує маніпуляцію *DOM*, обробку подій, анімацію та *Ajax* взаємодії для швидкої розробки веб-програм.

Popper.min.js – це бібліотека для управління впливаючими вікнами, яка використовується в *Bootstrap* для забезпечення позиціонування випадаючих меню, вспливаючих підказок та інших плаваючих елементів [11].

Крім того, *jQuery* та *Popper.js* сприяють швидкості веб-додатків. Що є важливим для створення інтерактивних веб-сторінок, *jQuery* полегшує обробку подій і полегшує управління *DOM*. *Popper.js*, який працює разом з *Bootstrap*, дозволяє правильно позиціонувати різні елементи інтерфейсу, що робить роботу з впливаючими вікнами та підказками зручною. Такий набір інструментів полегшує розробку та допомагає створити ефективну систему бронювання квитків.

3.3. Тестування роботи програмного модуля

3.3.1. Вибір методології тестування

Під час розробки програмного забезпечення модуля управління взаємодією з клієнтами авіакомпанії застосовувалася методологія модульного тестування, що дозволила перевіряти кожен окремий компонент системи на коректність виконання передбачених функцій. Модульне тестування полягає (рис. 3.5) у створенні тестових сценаріїв, які перевіряють поведінку окремих частин програми на відповідність специфікаціям та вимогам. Для перевірки роботи реєстрації та авторизації адміністратора розроблялись тест-кейси, які імітували вхід в систему з використанням правильних та неправильних даних для виявлення помилок у механізмах аутентифікації. Тестування перевірки стану здоров'я включало аналіз реакції системи на введення різних показників температури та тиску, а також їх

коректне відображення і обробку. Для перевірки модуля виводу погодних умов розроблялися тести, які перевіряли точність та актуальність відображення інформації про погоду. Окремою частиною тестування було формування *qr*-кодів, де основною задачею було переконатися, що *qr*-коди генеруються правильно та містять відповідну інформацію, яка може бути зчитана спеціалізованими пристроями або додатками.

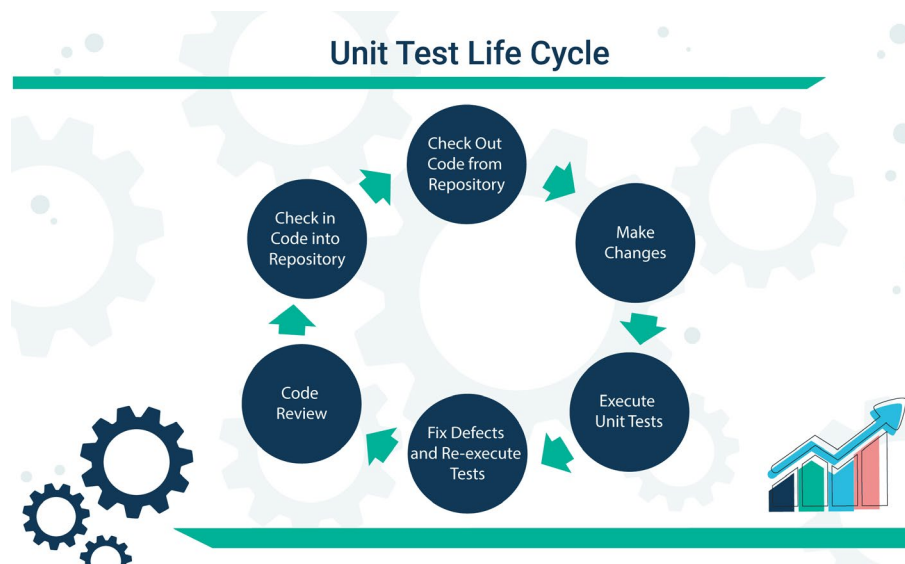


Рис. 3.5. Життєвий цикл модульного тестування

За допомогою цього методу можна перевірити окремі компоненти системи на відповідність специфікаціям і вимогам. Важливим етапом є реєстрація та авторизаційне тестування адміністраторів. Це включає в себе розробку тестових прикладів для тестування реакції системи на різні типи вхідних даних з метою виявлення та виправлення помилок у механізмі автентифікації. Крім того, тестування включає перевірку правильності обробки та відображення даних про здоров'я, таких як температура та тиск.

Для модуля виводу погоди були проведені тести, щоб перевірити точність і релевантність відображеної інформації про погоду. Окремим етапом тестування є генерація QR-кодів, головна мета якого – переконатися, що вони створені правильно та містять інформацію, необхідну для подальшого зчитування спеціальними пристроями чи додатками. Цей метод тестування дозволяє виявляти та виправляти помилки на ранніх етапах розробки, забезпечуючи високу якість та надійність програмного продукту.

3.3.2. Сценарії тестування

Для забезпечення надійності та ефективності програмного модуля управління взаємодією з клієнтами, було розроблено ряд тестових сценаріїв. Кожен тестовий випадок фокусується на перевірці певної функції модуля:

- 1) тестовий випадок для функції бронювання квитків:
 - перевірка правильності бронювання на доступні рейси;
 - перевірка відмови у бронюванні при відсутності місць;
 - перевірка збереження даних клієнта при бронюванні.
- 2) тестовий випадок для функції відміни бронювання:
 - перевірка процесу відміни бронювання квитка;
 - перевірка повернення коштів клієнту після відміни бронювання;
 - перевірка оновлення статусу місця на рейсі після відміни бронювання.
- 3) тестовий випадок для функції отримання інформації про рейси:
 - перевірка точності та актуальності інформації про рейси;
 - перевірка функціоналу пошуку рейсів з заданими параметрами;
 - перевірка відображення змін у розкладі рейсів.

Кожен із цих тестових випадків містить декілька тестових сценаріїв, які включають введення валідних та невалідних даних для виявлення помилок та перевірки межових випадків. Виконання цих сценаріїв дозволяє гарантувати, що програмний модуль працює стабільно і згідно з вимогами.

3.3.3. Результати тестування

Під час тестування програмного модуля було виявлено декілька помилок, які були усунуті для підвищення ефективності та надійності роботи системи:

- 1) помилка в алгоритмі бронювання квитків: при бронюванні квитків на майже повністю зайняті рейси виникала помилка переповнення. Виправлення полягало у вдосконаленні логіки перевірки наявності вільних місць перед підтвердженням бронювання;

- 2) помилка в системі відміни бронювання: виявлено, що при відміні бронювання сума повернення коштів не завжди коректно розраховувалась. Проблема була усунута оновленням формули розрахунку вартості;
- 3) неточності в інформації про рейси: тестування виявило, що іноді інформація про час вильоту була недостовірною. Оптимізовано систему оновлення даних рейсів, що забезпечує їх актуальність.

Після виправлення помилок проведено повторне тестування, яке підтвердило ефективність внесених змін. Окрім цього, було проведено оптимізацію бази даних та запитів до неї, що дозволило зменшити час відгуку системи та підвищити швидкість обробки запитів користувачів. Результати тестування та оптимізації показали значне покращення працездатності модуля.

3.4. Висновки до розділу

Висновки до розділу підсумовують основні аспекти розробки та перевірки модуля, з акцентом на досягнення, виклики та уроки, отримані під час процесу:

- 1) системні вимоги: ретельне визначення апаратних, програмних, мережевих та безпекових вимог стало основою для ефективної реалізації модуля. Встановлення чітких параметрів дозволило забезпечити його сумісність та високу продуктивність;
- 2) алгоритми реалізації модулів: розроблені ключові алгоритми, включаючи обробку запитів клієнтів, пошук оптимальних рейсів, розрахунок вартості квитків, взаємодію з платіжними системами, забезпечили модуль потрібним функціоналом і зробили його використання інтуїтивно зрозумілим та зручним;
- 3) використання бібліотек та фреймворків: застосування надійних і перевірених бібліотек, таких як *Composer*, *Bootstrap*, *jQuery*, *Popper.js*, дозволило створити міцну основу для функціональності модуля, забезпечити його стабільність та зручність у використанні;

4) тестування програмного модуля: використання модульного тестування, складання детальних сценаріїв тестування та аналіз виявлених помилок із подальшим їх виправленням дозволило значно підвищити якість та надійність програмного забезпечення. Процес тестування забезпечив відповідність продукту встановленим стандартам і вимогам;

Узагальнюючи, можна сказати, що розробка та тестування програмного модуля відбулися успішно, з урахуванням всіх поставлених вимог і забезпеченням високого рівня якості кінцевого продукту. Здійснені заходи не тільки сприяли створенню надійного та ефективного рішення, але й заклали основу для подальшого розвитку та масштабування системи.

Протягом реалізації програмного модуля ми зіткнулися з рядом проблем, які були успішно вирішені. Технічні труднощі, пов'язані з сумісністю бібліотек та залежностями версій, були подолані за допомогою контейнеризації, що забезпечило стабільність системи. Програмні помилки, виявлені на етапі тестування, були ефективно усунені завдяки ретельній ревізії коду. Проектування користувацького інтерфейсу було оптимізовано через застосування принципів *UX/UI* дизайну, що підвищило інтуїтивність використання програми.

Відповідність вимогам безпеки була досягнута шляхом імплементації сучасних методів шифрування та аутентифікації. Інтеграція з існуючими системами була здійснена через створення адаптивних *API*-шлюзів, що дозволило плавно включити новий модуль в існуючу інфраструктуру. Також було вирішено питання продуктивності системи, яке виникло внаслідок обробки великої кількості даних, через оптимізацію запитів до баз даних та кешування.

Ці заходи не лише усунули поточні проблеми але й забезпечили зміцнення позицій програмного модуля з точки зору стабільності та безпеки, підвищивши його цінність та надійність. Здобутий у ході роботи досвід став важливим активом для подальшого розвитку та впровадження інноваційних рішень у майбутніх проектах.

РОЗДІЛ 4

МЕТОДИКА РОБОТИ КОРИСТУВАЧА З МОДУЛЕМ

4.1. Системні вимоги й інсталяція модуля

4.1.1. Системні вимоги для інсталяції програмного модулю

Програмний модуль для управління інформаційною діяльністю авіакомпанії не є десктопною програмою, а є веб-додатком чи *CRM*-системою. Він здатен функціонувати, використовуючи сервер як в глобальній мережі (Інтернет), так і локально, наприклад, у мережі аеропорту. Глобальна версія, звичайно, дозволяє здійснювати реєстрацію та бронювання пасажирів з будь-якої точки світу.

Мінімальні характеристики чи вимоги до програмного забезпечення - це такі, що дозволяють вільно користуватися веб-браузерами (*Google Chrome, Mozilla Firefox, Microsoft Edge, Opera, Safari*). Як відомо, браузери постійно оновлюються і потребують значних програмних та апаратних ресурсів. Бажані операційні системи - це *Windows 10, 11, Linux (Ubuntu, Mint, Debian, OpenSuse)* та інші, які підтримують останні версії браузерів, а також *Mac OS*. Апаратне забезпечення повинне мати принаймні двоядерний, а краще чотири ядерний процесор, 4 *ГБ* ОЗУ, *SSD* на 80-120 *ГБ*, відеоадаптер на 512 *МБ* – 1 *ГБ*, монітор із роздільною здатністю 1280×1024 (15 дюймів) та аксесуари, такі як клавіатура та миша. З цими параметрами можна вільно користуватися додатком та використовувати його потенціал [9].

Також необхідно мати смартфон, здатний сканувати *QR*-код, який видається пасажирові після успішної перевірки стану здоров'я.

Встановлення програми на веб-сервер здійснює спеціалізована особа - системний адміністратор, який обізнаний з тонкощами налаштування сервера та розміщення файлів програми та бази даних. Адміністратор авіакомпанії, в свою чергу, виконує завдання з додавання рейсів та перевірки реєстрації пасажирів. Він також стежить за погодними умовами та видачою квитків пасажиром.

4.1.2. Покрокова інструкція з інсталяції та налаштування системи

Файли веб-додатку упаковані в архівний файл. Дві бази даних зберігаються окремо. Для встановлення програми потрібно приблизно 100 Мб вільного місця на жорсткому диску. Самі файли програмного модуля займають близько 20 Мб, а також потрібно додаткове місце для зберігання аватарів адміністраторів і менеджерів додатку. Починаємо з розпакування архіву, потім встановлюємо бази даних, після чого налаштовуємо підключення до баз даних. Останнім етапом є налаштування сертифіката безпеки. Після цього можемо у веб-браузері перевірити роботу веб-додатку (рис. 4.1).

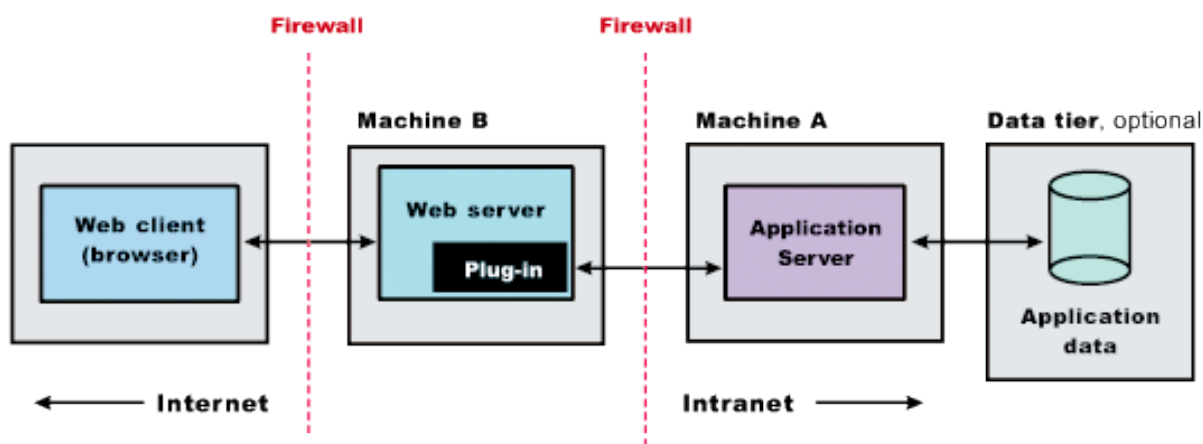


Рис. 4.1. Взаємодія веб-серверу та серверу додатка на окремих машинах

4.1.3. Перевірка правильності інсталяції та усунення можливих помилок

Після завершення інсталяції я успішно відкрив веб-додаток у браузері та переконався, що головна сторінка завантажується без помилок. Всі графічні елементи відображаються коректно, і тому отримав доступ до всіх запланованих функцій додатку.

Було виконано вхід в систему як адміністратор та звичайний користувач, що дозволило перевірити правильність роботи системи аутентифікації. Функції створення, редагування та видалення записів, таких як рейси або профілі користувачів, були протестовані та працювали без збоїв.

Також перевірів з'єднання з базами даних, виконавши ряд запитів, і переконався, що вони обробляються коректно. Функціональність експорту та імпорту даних була перевірена та працювала згідно з очікуваннями.

Перевірка відправки та отримання електронних повідомлень через додаток показала, що ця система функціонує бездоганно. Було забезпечено тестування додатку на різних пристроях та у різних веб-браузерах, щоб підтвердити його сумісність.

Було підтверджено, що сертифікати безпеки встановлені правильно, і з'єднання з сайтом захищене. Всі помилки, які були виявлені під час тестування, були записані та усунені згідно з документацією та за допомогою технічної підтримки.

Логи сервера та додатку були ретельно перевірені, що дозволило ідентифікувати та усунути причини можливих проблем. Це, за потреби, включало внесення змін до конфігураційних файлів та оновлення програмного забезпечення.

4.2. Робота користувача з додатком

4.2.1. Реєстрація користувачів у системі

Процес реєстрації користувачів у системі:

- 1) користувач відкриває форму реєстрації на головній сторінці додатку;
- 2) вводить необхідні персональні дані, такі як ім'я, прізвище, електронна пошта, та створює унікальний пароль;
- 3) підтверджує, що ознайомлений з умовами користування та політикою конфіденційності;
- 4) натискає кнопку «Зареєструватися»;
- 5) проходить процес верифікації електронної пошти через посилання, що надійшло в листі;
- 6) після підтвердження електронної пошти, користувачеві надається доступ до свого особистого кабінету.

Процес авторизації користувачів в системі (рис. 4.2):

- 1) користувач вибирає опцію «Увійти» на головній сторінці додатку;
- 2) вводить свою електронну пошту та пароль, який був створений під час реєстрації;
- 3) натискає кнопку «Увійти в систему»;
- 4) якщо введені дані коректні, користувач отримує доступ до свого особистого кабінету;
- 5) у разі виникнення проблем з входом, система надасть опцію скидання пароля через електронну пошту;
- 6) після скидання пароля користувач може створити новий пароль та знову увійти в систему.

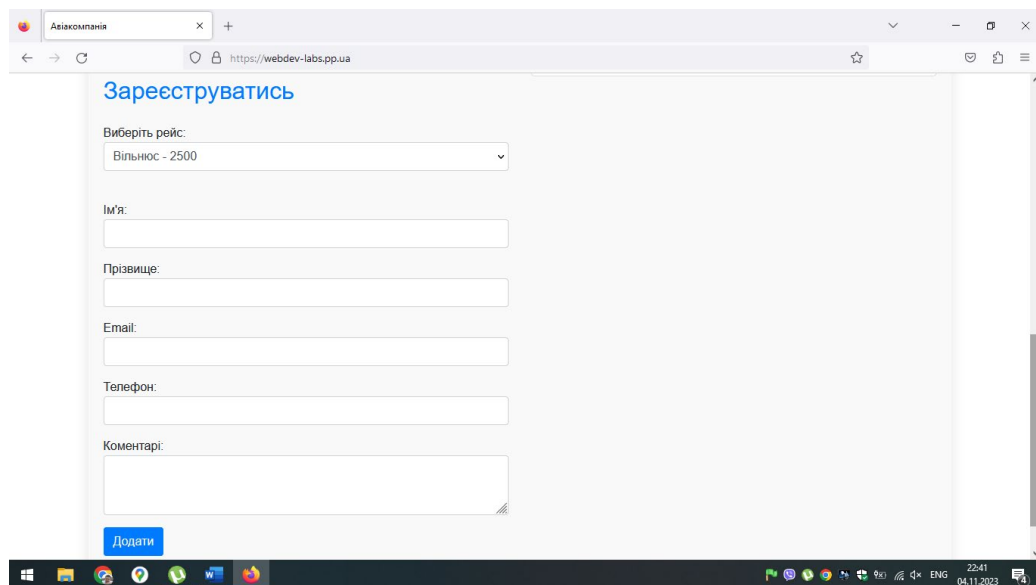


Рис. 4.2. Форма реєстрації

4.2.2. Інтерфейс користувача та основні функції для бронювання квитків

1) пошук рейсів:

- користувач запускає функцію пошуку на головній панелі, вводячи пункт відправлення та призначення;
- вказує дати подорожі та кількість пасажирів;

- вибирає клас обслуговування та інші додаткові параметри, якщо необхідно;
- натискає кнопку «Знайти рейси» для відображення доступних варіантів.

2) вибір рейсу:

- переглядає знайдені варіанти, порівнює ціни та час вильоту/прильоту;
- обирає найбільш підходящий рейс, натискаючи на нього;
- деталі рейсу відображаються для ознайомлення та підтвердження вибору.

3) бронювання квитків:

- після вибору рейсу переходить до процедури бронювання;
- вносить персональні дані пасажирів, включаючи ім'я та паспортні дані;
- вибирає додаткові опції, як-от багаж та спеціальне харчування, якщо це доступно;
- переходить до оплати, де вводить дані платіжної картки та підтверджує транзакцію.

4) підтвердження бронювання:

- отримує електронне підтвердження бронювання з усіма деталями рейсу та інформацією про квитки;
- має можливість роздрукувати квитки або зберегти їх у мобільному застосунку.

5) управління бронюванням:

- в особистому кабінеті може переглядати свої бронювання, змінювати дані або скасовувати квитки;
- використовує функції швидкого доступу для управління майбутніми подорожами.

Цей інтерфейс та функції забезпечують користувачу зручний та інтуїтивно зрозумілий процес бронювання квитків, з можливістю адаптації під особисті потреби та переваги.

4.2.3. Процес пошуку та вибору рейсів

Процес пошуку та вибору рейсів починається з того, що користувач входить у вкладку пошуку на головній сторінці веб-додатку або мобільного додатку і вводить необхідні дані для пошуку, такі як місце відправлення, пункт призначення, дати поїздки та кількість осіб, що мандрують. Від користувача також потрібно визначити, чи потрібен односторонній або зворотний квиток і встановити пріоритети пошуку, які можуть включати найменшу ціну, найкоротший час у дорозі, а також вибір між рейсами без пересадок або з мінімальною кількістю пересадок. Крім того, можна вибрати клас кабіни та вказати інші спеціальні вимоги.

Після встановлення всіх параметрів, користувач натискає кнопку "Пошук", що дозволяє системі відобразити список рейсів, які відповідають вказаним критеріям. Список можна сортувати за різними параметрами для зручності. Користувач може переглянути результати пошуку, аналізувати варіанти, звертаючи увагу на час вильоту, тривалість польоту, ціну та авіакомпанію. Щоб отримати більше інформації про будь-який рейс, достатньо клікнути на нього і переглянути детальну інформацію, включаючи тип літака, наявність пересадок та умови бронювання [27].

Ознайомившись з усіма деталями та переконавшись, що обраний рейс відповідає всім потребам, користувач натискає кнопку «Бронювати» або «Вибрати» для того, щоб перейти до процесу бронювання квитків. Цей інтуїтивний процес забезпечений для максимальної зручності та ефективності вибору рейсів у додатку.

Процес пошуку та вибору рейсів в інформаційній системі авіакомпанії є інтуїтивним та зручним для користувачів. Він включає в себе використання спеціалізованих фільтрів, щоб користувачі могли визначити найбільш відповідні рейси за власними критеріями. Користувачі спочатку вводять необхідні дати подорожі, пункти відправлення та прибуття, а також інші параметри пошуку, такі як клас обслуговування та кількість пасажирів. Система потім обробляє цю інформацію та відображає список можливих варіантів рейсів.

4.2.4. Оформлення бронювання та умови зміни/відміни бронювання

Оформлення бронювання в системі авіакомпанії є простим і швидким процесом. Після вибору рейсу користувачі переходять до введення своїх особистих даних, включаючи ім'я, прізвище, контактну інформацію, та інформацію про документ, що засвідчує особу. Наступним кроком є вибір способу оплати та завершення транзакції, після чого бронювання буде підтверджено, а всі деталі рейсу та бронювання будуть надіслані на електронну пошту користувача.

Умови зміни або відміни бронювання варіюються в залежності від тарифних правил, які були обрані під час бронювання. Для деяких гнучких тарифів може бути дозволено змінювати дати подорожі, імена пасажирів або маршрут без додаткових зборів, або за мінімальною платою. Тим часом, інші більш строгі тарифи можуть не дозволяти зміни або відміну без суттєвих фінансових штрафів.

Користувачам слід уважно ознайомитись з тарифними правилами перед завершенням бронювання, а також після оформлення зберегти всі отримані документи, які містять інформацію про умови та можливості зміни або відміни. У випадку необхідності зміни або відміни бронювання, користувачам рекомендується звернутись до служби підтримки авіакомпанії або здійснити це через особистий кабінет у веб-додатку, де часто надається можливість самостійно управляти своїм бронюванням.

4.3. Робота адміністратора з додатком

4.3.1. Роль адміністратора в системі бронювання

Робота адміністратора в системі бронювання є ключовою для забезпечення її ефективного функціонування та управління. Адміністратор несе відповідальність за цілий спектр завдань, починаючи від контролю доступу та закінчуючи аналітикою даних користувачів.

Перш за все, адміністратор встановлює параметри доступу, визначаючи, хто може використовувати систему та яким чином. Він створює та керує обліковими записами, призначаючи ролі користувачів, як-от менеджерів, операторів та інших адміністраторів, забезпечуючи, щоб кожен мав відповідні права та доступ до необхідних інструментів.

Адміністратор також відповідає за налаштування та оновлення системи, включаючи встановлення патчів безпеки, оновлення функціоналу та інтеграцію з іншими системами або сервісами. Він моніторить роботу системи, швидко реагуючи на будь-які технічні проблеми, які можуть виникнути, і координує роботу технічної підтримки для їх усунення.

Адміністратор також грає важливу роль у забезпеченні безпеки даних. Він впроваджує політики безпеки, регулярно проводить аудити та тести на проникнення, щоб забезпечити захист інформації від несанкціонованого доступу або зловмисних атак.

Окрім технічних аспектів, адміністратори також аналізують дані користувачів та трафік, щоб виявляти тенденції та робити обґрунтовані рішення щодо поліпшення системи. Вони готують звіти та рекомендації для керівництва, які допомагають формувати майбутнє розвитку системи бронювання.

Таким чином, адміністратор стає важливою ланкою, яка забезпечує стабільність, безпеку та розвиток системи бронювання, що безпосередньо впливає на якість обслуговування клієнтів та ефективність бізнес-процесів компанії.

4.3.2. Налаштування системи та управління користувацькими акаунтами

Налаштування системи та управління користувацькими акаунтами – це одне з основних завдань адміністратора в системі бронювання, яке включає ряд важливих кроків та процедур.

Спочатку адміністратор конфігурує параметри системи, забезпечуючи, щоб всі компоненти працювали належним чином і відповідали вимогам безпеки. Це може охоплювати встановлення і налаштування серверного програмного забезпечення,

налаштування мережі та забезпечення інтеграції з іншими сервісами, такими як платіжні системи або системи управління клієнтськими базами.

Після цього адміністратор переходить до управління користувацькими акаунтами. Він створює нові акаунти, призначає ролі користувачам, таким як менеджери, оператори чи інші адміністратори. Він також встановлює рівні доступу та привілеї для кожного користувача, виходячи з їхньої ролі та обов'язків в системі.

Крім того, адміністратор відповідає за моніторинг активності користувачів, забезпечення виконання політик безпеки та проведення регулярних оновлень акаунтів, щоб вони відповідали найсучаснішим вимогам. Він відстежує зміни в акаунтах, включаючи входи та виходи з системи, та реагує на будь-які підозрілі дії.

Адміністратор також відповідає за навчання користувачів, навчаючи, як користуватися системою та розумінню політики безпеки. Він може створювати інструкції та проводити тренінги, щоб усі користувачі були добре інформовані та мали необхідні навички для ефективної роботи.

Управління користувацькими акаунтами також включає вирішення будь-яких проблем, які можуть виникнути, наприклад, втрату пароля або необхідність зміни даних користувача. Адміністратор виступає як перша лінія підтримки для користувачів, які можуть мати запитання або потребують допомоги зі своїми акаунтами.

Завдяки цим діям адміністратор забезпечує високий рівень контролю та ефективності в управлінні користувацькими акаунтами, що є важливим для забезпечення безперебійної роботи системи бронювання та задоволення потреб бізнесу.

4.3.3. Моніторинг стану бронювання та управління рейсами

Адміністратор системи бронювання відіграє ключову роль у моніторингу стану бронювання та управлінні рейсами. Цей процес включає відстеження поточних та майбутніх бронювань, перевірку наявності місць, та аналіз тенденцій бронювання, що дозволяє прогнозувати попит та оптимізувати заповнюваність рейсів.

Моніторинг стану бронювань здійснюється через інтерфейс адміністратора, який надає інформацію в реальному часі про всі активні бронювання. Адміністратор може переглядати деталі кожного бронювання, включаючи дату поїздки, статус платежу, інформацію про пасажирів та їхні вимоги до подорожі. Він також може виявляти і вирішувати конфлікти або проблеми з бронюванням, такі як пере бронювання або помилки у внесенні даних.

Управління рейсами включає регулювання розкладу рейсів, встановлення та зміну цін на квитки в залежності від різних факторів, таких як сезонність, спеціальні події, або зміни в операційних витратах. Адміністратор може вносити зміни до існуючих рейсів або додавати нові, враховуючи попит і рекомендації з аналізу даних.

Окрім того, адміністратор відповідає за перевірку безпеки та надійності рейсів, співпрацюючи з авіаційними компаніями та іншими партнерами для забезпечення відповідності всіх рейсів нормам і стандартам безпеки. Він також має відстежувати та реагувати на екстрені ситуації або затримки, оперативно інформуючи пасажирів та реорганізовуючи розклади.

Моніторинг стану бронювань та управління рейсами дозволяє адміністратору забезпечити ефективне використання ресурсів, підвищити задоволеність пасажирів і збільшити прибутковість системи бронювання (рис. 4.4).

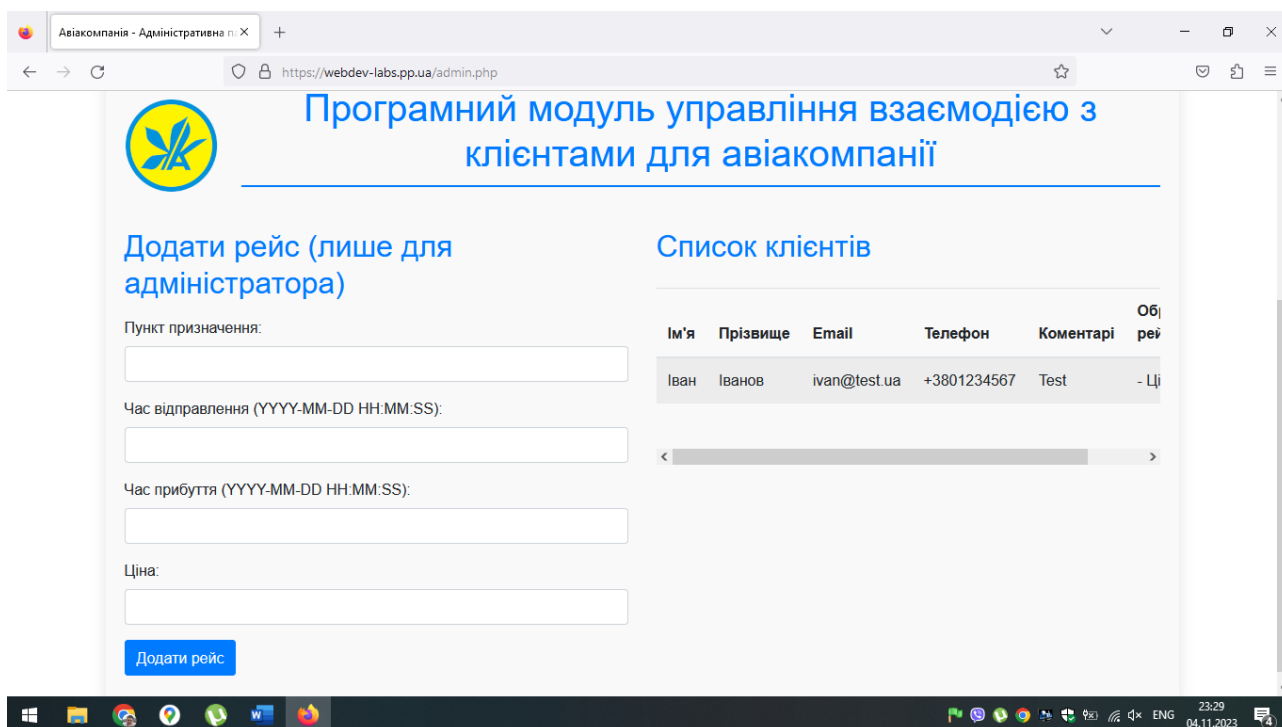


Рис. 4.4. Форма «Список клієнтів» (справа) для моніторингу бронювання авіаквитків

4.3.4. Звітність та аналітика використання системи

Звітність та аналітика в системі бронювання є фундаментальними інструментами для оцінки ефективності та вдосконалення сервісів. Вони забезпечують детальний огляд даних по користувачах, рейсам, доходах та багатьом іншим критеріям.

Адміністратор використовує інструменти звітності для генерування звітів про використання системи, включно з кількістю здійснених бронювань, загальним доходом, середньою вартістю бронювання, піками активності користувачів, і частотою скасувань. Ці звіти можуть бути персоналізовані для показу даних за певні періоди часу або категорії даних, що дає можливість виявити тенденції та закономірності у поведінці користувачів.

Аналітика використання системи спрямована на розуміння того, як користувачі взаємодіють з системою, виявлення найбільш популярних рейсів, а також оцінку задоволеності користувачів через опитування та відгуки. Використання аналітичних інструментів допомагає ідентифікувати слабкі місця в роботі системи та виявити можливості для покращення, наприклад, оптимізація процесу бронювання або поліпшення інтерфейсу користувача.

Додатково, інструменти аналітики можуть бути використані для проведення сегментованого аналізу, який дозволяє адміністратору розглянути віддільні сегменти ринку, такі як *VIP*-користувачі, часті пасажери, або групові бронювання. Це забезпечує можливість створення цільових маркетингових кампаній та пропозицій, що можуть підвищити лояльність користувачів і залучити нових клієнтів.

Аналіз звітності та аналітики бронювання також допомагає зрозуміти користувацьку перевагу та поведінку. Ця інформація має вирішальне значення для вдосконалення пропозицій, які задовольняють потреби клієнтів. Наприклад, пошук популярних маршрутів може допомогти визначити, які місця є найбільш привабливими для подорожей у певний період року чи для певних цільових груп. Це дозволяє створювати унікальні пропозиції та рекламні кампанії, щоб підвищити інтерес до певних послуг або привернути нових клієнтів.

Дослідження рівня задоволеності користувачів за допомогою опитувань і відгуків допомагає визначити сфери, які потребують удосконалення. Наприклад, якщо опитування виявить проблеми з процесом бронювання квитків, аналітики можуть визначити конкретні кроки, які викликають ці проблеми, щоб розробити рішення та покращити процес.

Для адміністратора системи також важлива деталізована звітність, оскільки вона дозволяє відстежувати рівень використання сервісів і їх ефективність.

В цілому, звітність та аналітика є життєво важливими для забезпечення стабільного розвитку та досягнення стратегічних бізнес-цілей системи бронювання, дозволяючи приймати обґрунтовані рішення на основі точних даних.

4.4. Висновки до розділу

Висновки до розділу щодо системи бронювання свідчать про її ефективність і відповідність заданим критеріям користувачів та адміністраторів. Система була розроблена з урахуванням їх потреб і вимог, що дозволило створити інтуїтивно зрозумілий і зручний інтерфейс, спрощуючи процеси реєстрації, пошуку та бронювання рейсів.

Користувачі системи отримали можливість швидко знаходити варіанти рейсів, порівнювати ціни і вибирати найкращі пропозиції, що в свою чергу підвищує їх задоволеність від сервісу. Для адміністраторів система надає потужні інструменти управління та моніторингу, які дозволяють оперативно реагувати на зміни в резерваціях, а також вдосконалювати систему, базуючись на детальній аналітиці і звітності.

Також система бронювання показала гарну масштабованість та гнучкість, оскільки може бути адаптована під змінні потреби користувачів та розвиток технологій. Проте, важливо продовжувати збір зворотного зв'язку від користувачів і регулярно проводити оновлення, щоб забезпечити актуальність сервісу та його конкурентоспроможність.

У підсумку, оцінка показала, що система бронювання відповідає основним вимогам користувачів та адміністраторів, проте для її поліпшення необхідно постійне вдосконалення та ітераційний розвиток.

Висновки з приводу ефективності системи бронювання підкреслюють, що вона ефективно вирішує поставлені перед нею завдання. Ефективність системи можна оцінити через її здатність задовольняти потреби користувачів у швидкому пошуку і бронюванні рейсів, забезпечуючи гнучкість і комфорт управління для адміністраторів.

Ключові показники ефективності включають зниження часу на пошук та бронювання, зменшення помилок у резерваціях та підвищення задоволеності клієнтів завдяки інтуїтивно зрозумілому інтерфейсу і простоті використання. За допомогою аналітичних інструментів адміністратори можуть відслідковувати тенденції бронювань та оптимізувати робочі процеси, що також сприяє загальній ефективності системи.

Додатково, система демонструє високу стабільність та надійність, що є критично важливим для забезпечення безперервної роботи сервісу бронювання. Можливість масштабування дозволяє впроваджувати необхідні оновлення та вдосконалення без збоїв у роботі існуючої системи.

Звісно, існують аспекти, які потребують додаткової уваги та розвитку, включаючи захист даних користувачів та забезпечення їх конфіденційності. Незважаючи на це, загальний аналіз підтверджує, що система є ефективним інструментом, який забезпечує високий рівень сервісу та задоволення потреб користувачів та адміністраторів.

На основі аналізу ефективності системи бронювання, можна внести наступні рекомендації для її подальшої оптимізації та розвитку:

- 1) покращення інтерфейсу користувача (UI): регулярно оновлювати дизайн користувацького інтерфейсу, зробити його більш інтуїтивним та відповідним до сучасних трендів UX дизайну, щоб поліпшити загальний досвід користувачів;

- 2) модуль штучного інтелекту (*AI*) для персоналізації: інтегрувати *AI* для аналізу поведінки користувачів та надання персоналізованих рекомендацій, що може збільшити конверсію бронювань та задоволеність користувачів;
- 3) оптимізація мобільного додатку: забезпечити, щоб мобільна версія додатку була оптимізована і зручна для використання, враховуючи зростаючу кількість користувачів мобільних пристроїв;
- 4) поліпшення модуля звітності: розробити більш детальні аналітичні інструменти для адміністраторів, щоб забезпечити глибший аналіз даних і легше виявлення тенденцій та можливостей для росту;
- 5) захист даних та конфіденційність: запровадити передові рішення для захисту даних користувачів, включаючи шифрування, щоб зміцнити довіру користувачів;
- 6) модуль прогнозування попиту: використовувати історичні дані для створення прогнозів попиту на рейси, що допоможе оптимізувати наявність та ціноутворення;
- 7) розширення можливостей *API*: розвинути *API* для інтеграції з іншими системами та сервісами, що дозволить розширити можливості бронювання та забезпечити більшу синергію з партнерами;
- 8) система лояльності: розробити систему лояльності з персоналізованими бонусами та пропозиціями, щоб збільшити відданість користувачів;
- 9) підтримка багатомовності: забезпечити підтримку додаткових мов у системі, щоб розширити ринок збуту та поліпшити досвід міжнародних користувачів.

Реалізація цих рекомендацій може допомогти забезпечити, щоб система бронювання залишалася конкурентоспроможною, відповідала потребам користувачів та технологічному прогресу.

ВИСНОВКИ

У сучасних умовах ринку авіаперевезень, авіакомпанії опиняються перед численними викликами, які вимагають швидкої адаптації та впровадження новітніх технологій для збереження конкурентоспроможності. Важливим аспектом є забезпечення безпеки даних клієнтів, особливо в контексті зростаючих кібер-загроз та посилення законодавчих вимог. Ефективна інтеграція систем управління взаємодією з клієнтами з іншими інформаційними системами і технологіями дозволяє не лише покращити якість обслуговування, але й робить послуги більш ефективними та гнучкими.

Враховуючи це, авіакомпанії повинні застосовувати комплексний підхід до управління взаємодіями з клієнтами, що охоплює впровадження передових технологій, оновлення стратегій обслуговування та безпеки даних. Такий підхід сприятиме підвищенню лояльності клієнтів, забезпеченню їхньої задоволеності та стабільному розвитку компанії в майбутньому.

Запропонована кваліфікаційна робота зосереджена на глибокому аналізі, детальному дослідженні та розробці системи управління взаємодіями з клієнтами для авіакомпанії. У рамках роботи було здійснено систематичне порівняння існуючих аналогічних систем, ідентифікацію ключових бізнес-процесів, та визначено необхідні дані для збору з метою подальшої їх обробки. Також було виконано аналіз інструментів та технологій, що можуть бути застосовані при створенні системи, включаючи виклики та потенційні ризики, з якими я зіткнувся, і які було успішно вирішено. Особливу увагу було приділено ролі цифрової трансформації в авіаційній сфері, зокрема аналізу таких технологій, як інтернет речей, штучний інтелект, машинне навчання та блокчейн.

У другому розділі основна увага була приділена визначенню ключових функцій програмного модуля *CRM*-системи. Ця система виявилася незамінною для роботи з об'ємними даними клієнтів, оптимізації бізнес-процесів та організації роботи

чисельного персоналу. Кожна функція була детально описана, висвітлюючи її призначення та основні характеристики.

Що стосується архітектури програмного забезпечення, у розділі були визначені основні принципи та підходи до її створення, а також описані ключові технології та інструменти, які застосовувалися під час розробки.

Особлива увага була приділена взаємодії та інтеграції функцій у загальну систему, що забезпечило високий рівень зручності та інтуїтивності користувацького інтерфейсу. Компоненти програмного модуля взаємодіють у складній мережі зв'язків, формуючи злагоджений механізм управління.

У кінці розділу було проведено оцінку ефективності та гнучкості обраної архітектури, що підтвердило її відповідність новітнім вимогам та викликам ринку.

Крім того, в цьому розділі були успішно визначені основні функції програмного модуля для *CRM*-системи. Система стала незамінною для оптимізації роботи з великою кількістю даних клієнтів, автоматизації продажів, організації роботи численного персоналу та збору детальної статистичної інформації про продажі. Було надано детальний опис кожної функції веб-додатку, її призначення та основних характеристик.

У третьому розділі можна підбити підсумки, що стосуються створення та перевірки модуля, з особливим фокусом на здобутки, виклики та важливі знання, здобуті протягом цього процесу. Визначення системних вимог з точністю було ключем до успішного впровадження модуля, забезпечуючи його сумісність і високу ефективність. Розроблені алгоритми дозволили модулю ефективно виконувати свої функції, роблячи його простим і зручним у використанні. Впровадження перевірених бібліотек і фреймворків створило міцну основу для модуля, забезпечуючи його стабільну роботу. Ретельне тестування сприяло покращенню якості та надійності програми, гарантуючи її відповідність стандартам. Вдосконалення модуля через виправлення помилок і оптимізацію процесів підвищило його продуктивність і покращило враження користувачів від системи. В цілому, розробка і тестування модуля були виконані успішно, відповідаючи всім вимогам і забезпечуючи високу якість кінцевого продукту. Здійснені кроки не лише забезпечили створення надійного

і продуктивного рішення, але й стали основою для його майбутнього розвитку і масштабування.

Протягом реалізації програмного модуля виник ряд проблем, які були успішно вирішені. Технічні труднощі, пов'язані з сумісністю бібліотек та залежностями версій, були подолані за допомогою контейнеризації, що забезпечило стабільність системи. Програмні помилки, виявлені на етапі тестування, були ефективно усунені завдяки ретельній ревізії коду. Проектування користувацького інтерфейсу було оптимізовано через застосування принципів *UX/UI* дизайну, що підвищило інтуїтивність використання програми.

Відповідність вимогам безпеки була досягнута шляхом імплементації сучасних методів шифрування та аутентифікації. Інтеграція з існуючими системами була здійснена через створення адаптивних *API*-шлюзів, що дозволило плавно включити новий модуль в існуючу інфраструктуру. Також було вирішено питання продуктивності системи, яке виникло внаслідок обробки великої кількості даних, через оптимізацію запитів до баз даних та кешування.

Ці заходи не лише усунули поточні проблеми але й забезпечили зміцнення позицій програмного модуля з точки зору стабільності та безпеки, підвищивши його цінність та надійність. Здобутий у ході роботи досвід став важливим активом для подальшого розвитку та впровадження інноваційних рішень у майбутніх проектах.

Завершений аналіз про систему бронювання підтверджує її високу продуктивність та відповідність встановленим стандартам, які були розроблені на основі вимог користувачів та адміністраторів. Передові практики *UX/UI* дизайну були застосовані для створення інтуїтивно зрозумілого та легкого в користуванні інтерфейсу, що спрощує процес бронювання і пошуку рейсів.

Клієнти системи мають доступ до ефективних інструментів для швидкого відбору рейсів, порівняння тарифів і вибору оптимальних пропозицій, що забезпечує високий рівень задоволеності сервісом. Адміністратори отримали в розпорядження потужний набір інструментів для управління та моніторингу, дозволяючи швидко адаптуватися до змін ринкових умов та вносити необхідні корективи у систему на основі аналітики та звітності.

Система продемонструвала відмінну масштабованість та адаптивність, що дозволяє їй продуктивно працювати з різноманітними і зростаючими запитами користувачів та бути актуальною з огляду на швидкий розвиток технологій. Незважаючи на це, критично важливим залишається збір відгуків від користувачів та проведення регулярних оновлень для забезпечення неперервної актуальності та конкурентоспроможності сервісу.

У кінцевому рахунку, система бронювання виявилася відповідною до ключових вимог користувачів та адміністраторів, але її подальше поліпшення потребує неперервної оптимізації та ітеративних удосконалень.

Аналіз системи бронювання вказує на необхідність оновлення інтерфейсу для кращого користувацького досвіду, інтеграції *AI* для персоналізованих рекомендацій, створення та оптимізації мобільного додатку з урахуванням збільшення мобільних користувачів. Важливим є також розширення аналітичних інструментів для адміністраторів, підвищення захисту даних користувачів через шифрування для зміцнення довіри, розробка прогнозів попиту на рейси для кращого управління наявністю, розширення *API* для інтеграції з іншими системами, створення системи лояльності, навчальних програм для користувачів та адміністраторів, а також підтримка додаткових мов для міжнародного розширення. Ці заходи допоможуть системі залишатися конкурентоспроможною та відповідати потребам користувачів і технологічному прогресу.

Завдання, визначене в рамках проекту, було виконано ефективно та досягнуто його закінчення. Розроблений веб-додаток відповідає усім встановленим нормам і стандартам. Проведене тестування підтвердило його функціональну відповідність та надійність. Введення інноваційної функції – створення *QR*-коду, що відображає стан здоров'я пасажира, демонструє наукову новизну та підкреслює потенціал для подальшого технічного та технологічного розвитку в сфері наук і технологій.

СПИСОК БІБЛІОГРАФІЧНИХ ПОСИЛАНЬ ВИКОРИСТАНИХ ДЖЕРЕЛ

1. ПОЛОЖЕННЯ ПРО ДИПЛОМНІ РОБОТИ (ПРОЕКТИ) ВИПУСКНИКІВ НАЦІОНАЛЬНОГО АВІАЦІЙНОГО УНІВЕРСИТЕТУ. Київ: НАУ, 2017.
2. ДОКУМЕНТАЦІЯ. ЗВІТИ У СФЕРІ НАУКИ І ТЕХНІКИ. Структура і правила оформлення. ДСТУ 3008-95. Київ.
3. Аносов А. А. Вибір СУБД для побудови інформаційних систем. Київ: Знання, 2012. 278 с.
4. Афанасьєв В. А. Основи зовнішньоекономічної та комерційної діяльності в системі повітряного транспорту: посібник. Львів: Інфо-світ, 2012. 283 с.
5. Безгін К. С. Управління якістю бізнес-процесів на підприємстві: дис. канд. екон. наук: 08.00.04. Приазовський держ. технічний ун-т. Маріуполь, 2009. 196 с.
6. Берегова Я. В. Оперативна взаємодія з клієнтами як стратегічний фактор ефективності авіакомпанії. Інноваційні транспортні технології та транспортні системи: збірник матеріалів міжнародної науково-технічної конференції. Київ: НАУ, 2022. 270 с.
7. Стегер Дж., Снайдер М. Програмування *Microsoft Dynamics CRM 4.0*. Пер. Дудченко А. Київ: Наш Формат, 2017. 624 с.
8. Гарнаєв І. К. Інформаційні технології в соціально-культурному сервісі та туризмі. Київ: Оргтехніка, 2012. 236 с.
9. Гриценко А. А. Інформаційні технології в управлінні підприємством. Київ: КНЕУ, 2015. 403 с.
10. Ільїна Е. Н. Туроперейтинг: організація діяльності. Київ, Фінанси і статистика 2013. 338 с.
11. Іванов А. І., Петров Б. В. Основи систем бронювання, Київ. 2019. 371 с.
12. Карпенко С. С. Інформаційні технології в туризмі. Київський політехнік. 2021. № 3. С. 112-123.
13. Керівництво по бронюванню та продажу повітряних перевезень для агентств. Експертус. [Електронний ресурс].- режим доступу: <https://med.expertus.com.ua/laws/34358> (дата звернення: 28.10.2023).

14. Керівництво по бронюванню: інструкції для агентів APC СІРІН. Версія 10.3: у 6 т. 2013. 218 с.
15. Косолапов А. Б. Менеджмент в туристичній фірмі: навчальний посібник. Київ: КНОРУС, 2007. 229 с.
16. Котиков В. І. Електронна бібліотека інформотека: структура побудови, аналіз і синтез її основних елементів. Київ: Новий друк, 2012. 329 с.
17. Кузьмін О. Є., Мельник Л. М. CRM-системи як інструмент стратегії управління взаємовідносинами з клієнтами. Харків: ХНУ. 2017. 320 с.
18. Пушников А. Ю. Введення в системи управління базами даних: навч. посібник. Київ: Просвіта, 2013. 85 с.
19. Сергєєв А. К. Системи комплексного обслуговування: аспект бізнес-процесів. Київ: Смолоскип, 2001. 202 с.
20. Папенко Л. М. Підходи до визначення поняття «клієнтоорієнтованість» у контексті управління сервісним підприємством. Ч. 3. Науковий вісник Херсонського державного університету. 2015. Вип. 15. С. 67–70.
21. Родіонов П. Ю. Управління інформаційною діяльністю авіакомпанії: дисертація на здобуття наукового ступеня кандидата економічних наук. Київ: Національний авіаційний університет, 2015. 239 с.
22. Мельник Л. Т. Сучасні тенденції розвитку туристичного бізнесу в Україні. Одеса: Астропринт, 2018. 352 с.
23. Нормативи щодо обробки даних в системах бронювання. Міністерство цифрової трансформації України, 2020. 73 с.
24. Офіційний веб-сайт *ACI* (Міжнародна Рада Аеропортів). [Електронний ресурс].- режим доступу: <https://www.aci.aero/> (дата звернення 25.10.2023). - Назва з екрана.
25. Офіційний веб-сайт *SAPA* (Центр авіації Азії та Тихого океану). [Електронний ресурс].- режим доступу: <https://centreforaviation.com/> (дата звернення 25.10.2023). - Назва з екрана.

26. Офіційний веб-сайт *IATA* (Міжнародна Асоціація Авіаційного Транспорту). [Електронний ресурс].- режим доступу: <https://www.iata.org/> (дата звернення 25.10.2023). - Назва з екрана.
27. Сидорова Г. О. Аналітика в системах бронювання: сучасні підходи. Журнал інформаційних систем. 2020. № 4. С. 15-29.
28. Офіційний веб-сайт Державної служби статистики України. [Електронний ресурс].- режим доступу: <http://www.ukrstat.gov.ua/> (дата звернення 25.10.2023). - Назва з екрана.
29. *9 IT Trends Shaping the Aviation Industry in 2023*. NCS. [Електронний ресурс].- режим доступу: <https://www.networkcablingservices.com/9-it-trends-shaping-the-aviation-industry/> (дата звернення 25.10.2023). - Назва з екрана.
30. *Baggage Tracking*. *IATA*. [Електронний ресурс].- режим доступу: <http://www.iata.org/whatwedo/ops-infra/baggage/> (дата звернення 25.10.2023). - Назва з екрана.
31. *CRM*: цілі впровадження, огляд можливостей, варіанти розгортання та багато іншого. [Електронний ресурс].- режим доступу: <https://cutt.ly/SwwTGO7f> (дата звернення 25.10.2023). - Назва з екрана.
32. Офіційний сайт ДП МА «Бориспіль». [Електронний ресурс].- режим доступу: <https://kbp.aero/> (дата звернення 25.10.2023). - Назва з екрана.
33. Про захист персональних даних: Закон України від 1 червня 2010 р. № 2297-VI. Верховна Рада України. [Електронний ресурс].- режим доступу: <https://zakon.rada.gov.ua/laws/show/2297-17#Text> (дата звернення 25.10.2023). - Назва з екрана.

Лістинг коду візуальної частини

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Авіакомпанія</title>
  <!-- Підключення Bootstrap CSS -->
  <link href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
rel="stylesheet">
  <link rel="stylesheet" href="css/style.css">
</head>

<body>
  <?php
    $clientId = $_SESSION['client_id'];
    $clientInfo = getClient($clientId);

    if ($_SERVER["REQUEST_METHOD"] == "POST" && isset($_POST['add_client']))
    {
      $firstname = $_POST['firstname'];
      $lastname = $_POST['lastname'];
      $email = $_POST['email'];
      $phone = $_POST['phone'];
      $comments = $_POST['comments'];
```



```
    addClient($firstname, $lastname, $email, $phone, $comments);
}
?>
```

```
<?php
    if (isset($_GET['book_flight'])) {
        $flightId = $_GET['book_flight'];

        // Тут можна додати логіку для бронювання рейсу. Наприклад, зберегти
        інформацію про рейс в сесії і перенаправити користувача на сторінку оплати.
    }
?>
```

```
<?php
// Припустимо, що ідентифікатор клієнта і рейсу передаються через GET-запит
(через URL)
if (isset($_GET['clientId']) && isset($_GET['flightId'])) {
    $clientId = $_GET['clientId'];
    $flightId = $_GET['flightId'];

    // Викликати функцію bookFlight та вивести результат
    echo bookFlight($clientId, $flightId);
}
?>
```

```
<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $flightId = $_POST['flightId'];
    $firstname = $_POST['firstname'];
    $lastname = $_POST['lastname'];
```

```

$email = $_POST['email'];
$phone = $_POST['phone'];
$comments = $_POST['comments'];

// Додаємо клієнта
$clientId = addClient($firstname, $lastname, $email, $phone, $comments);
if ($clientId) {
    // Здійснюємо бронювання
    bookFlight($clientId, $flightId);
} else {
    echo "Не вдалося додати клієнта та здійснити бронювання.";
}
}
?>

```

```

<div class="container mt-6">
    <h3>Вхід для адміністратора</h3>
    <a class="btn btn-primary btn-lg" href="/registration/register.php"
role="button">Реєстрація</a>
    <a class="btn btn-success btn-lg" href="/registration/login.php"
role="button">Авторизація</a>
</div>
<div class="container mt-5">
    <!-- Header -->
    <div class="d-flex align-items-center mb-4">
        
        <h1 class="ml-4 text-center flex-grow-1">Програмний модуль управління
взаємодією з клієнтами для авіакомпанії</h1>
    </div>

```

```

<div class="row">
  <!-- Flights column -->
  <div class="col-md-6">
    <h2 class="mb-4">Доступні рейси</h2>
    <ul class="list-group mb-5">
      <?php
        $flights = getFlights();
        while ($flight = $flights->fetch_assoc()) {
          echo "<li class='list-group-item'>" . $flight['destination'] . " - Ціна: " .
            $flight['price'] . "</li>";
        }
      ?>
    </ul>
  </div>
  <!-- Перевірка пасажирів по стану здоров'я -->
  <div class="col-md-6">
    <div class="card">
      <div class="card-header">
        <h3>Дані про здоров'я</h3>
      </div>
      <div class="card-body">
        <form action="backend.php" method="post">
          <div class="form-group">
            <label for="healthData">Введіть загальні дані про здоров'я:</label>
            <input type="text" class="form-control" name="healthData"
              id="healthData">
          </div>
        </form>
      </div>
    </div>
  </div>

```

```

    <div class="form-group">
        <label for="temperature">Введіть вашу температуру (°C):</label>
        <input type="text" class="form-control" name="temperature"
id="temperature">
    </div>

    <div class="form-group">
        <label for="pressure">Введіть ваш тиск (наприклад,
120x80):</label>
        <input type="text" class="form-control" name="pressure"
id="pressure">
    </div>

    <div class="form-group">
        <input type="submit" class="btn btn-primary" value="Згенерувати QR
Код">
    </div>
</form>
</div>
</div>
</div>
<!-- Реєстрація пасажирів -->
<div class="col-md-6">
    <h2 class="mb-4">Зареєструватись</h2>
    <form action="index.php" method="post" class="mb-5">
        <div class="form-group">
            <div class="form-label">Виберіть рейс:</div>
            <select name="flightId" class="form-control">
                <?php
                    $flights = getFlights();

```

```

        while ($flight = $flights->fetch_assoc()) {
            echo "<option value=\"" . $flight['id'] . "\">" . $flight['destination'] . " -
" . $flight['price'] . "</option>";
        }
    ?>
</select>
</div>
<br>
<div class="form-group">
    <div class="form-label">Ім'я:</div>
    <input type="text" name="firstname" class="form-control" required>
</div>
<div class="form-group">
    <div class="form-label">Прізвище:</div>
    <input type="text" name="lastname" class="form-control" required>
</div>
<div class="form-group">
    <div class="form-label">Email:</div>
    <input type="email" name="email" class="form-control">
</div>
<div class="form-group">
    <div class="form-label">Телефон:</div>
    <input type="tel" name="phone" class="form-control">
</div>
<div class="form-group">
    <div class="form-label">Коментарі:</div>
    <textarea name="comments" class="form-control"></textarea>
</div>
<div class="form-group">

```

```
        <input type="submit" name="add_client" value="Додати" class="btn btn-
primary">
    </div>
</form>
</div>
</div>
</div>
<!-- Підключення Bootstrap JS та Popper.js для додаткової функціональності -->
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
<script
src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script
>
<script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
</body>

</html>
```