

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних систем та мереж
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

Магістр

(назва освітнього ступеня)

на тему: Методи та засоби оптимізації комп'ютеризованих CRM-систем
для логістики

Виконав: студент(ка) 6 курсу, групи СІМ-61
спеціальності 123 «Комп'ютерна інженерія»

(шифр і назва спеціальності)

	<u>Бражніков В.А.</u> (підпис) (прізвище та ініціали)
Керівник	<u>Осухівська Г.М.</u> (підпис) (прізвище та ініціали)
Нормоконтроль	<u>Тиш Є.В.</u> (підпис) (прізвище та ініціали)
Завідувач кафедри	<u>Осухівська Г.М.</u> (підпис) (прізвище та ініціали)
Рецензент	<u>Гашин Н.Б.</u> (підпис) (прізвище та ініціали)

Тернопіль
2024

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра комп'ютерних систем та мереж
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
Осухівська Г.М.
(підпис) (прізвище та ініціали)

«11» листопада 2024 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня магістр
(назва освітнього ступеня)

за спеціальністю 123 «Комп'ютерна інженерія»
(шифр і назва спеціальності)

студенту Бражнікову Владиславу Андрійовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Методи та засоби оптимізації комп'ютеризованих CRM-систем для логістики

Керівник роботи Осухівська Галина Михайлівна, к.т.н., доцент
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від « 29 » листопада 2024 року № 4/7-1144

2. Термін подання студентом завершеної роботи 20.12.2024

3. Вихідні дані до роботи

4. Зміст роботи (перелік питань, які потрібно розробити)

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

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
<i>Охорона праці</i>	<i>Пилипець М.І., проф.каф. МТ</i>		
<i>Безпека в надзвичайних ситуаціях</i>	<i>Стадник І.Я., проф.каф. ОХ</i>		

7. Дата видачі завдання 11.11.2024

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка

Студент

(підпис)*Бражніков В.А.*

(прізвище та ініціали)

Керівник роботи

(підпис)*Осухівська Г.М.*

(прізвище та ініціали)

АНОТАЦІЯ

Бражніков В.А. Методи та засоби оптимізації комп'ютеризованих CRM-систем для логістики, робота на здобуття кваліфікаційного ступеня магістра: спец. 123 — комп'ютерна інженерія / наук.кер. Г.М. Осухівська Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя, 2024. — 71 с.

Ключові слова: комп'ютеризована система, логістичні операції, оптимізація, php, javascript, vue, MySQL, Lavarel.

Кваліфікаційна робота присвячена дослідженню методів та засобів оптимізації CRM-систем у логістичних операціях. Робота включає в себе аналіз існуючих досліджень у сфері оптимізації логістичних операцій та акцентує увагу на можливостях CRM-систем у логістичних операціях. Для створення системи використовувалося програмне забезпечення написане на мовах Javascript, PHP та база даних у вигляді MySQL.

Графічний інтерфейс розроблений з використанням фреймворку Vue та bootstrap. Платформа забезпечує безпечну автентифікацію користувачів, ефективне управління вантажами та взаємодію між клієнтами. Результати поданого дослідження демонструють практичне застосування сучасних інструментів та методів побудови CRM-систем, що відповідають функціональним і продуктивним вимогам платформи для оптимізації логістичних операцій.

ANNOTATION

Brazhnikov V.A. Methods and tools for optimizing computerized CRM systems for logistics. Master's Graduation Thesis: speciality 123 - Computer engineering/supervisor Osukhivska H.M.. Ternopil: Ternopil Ivan Puluj National Technical University, 2024. — 71 p.

Keywords: computerized system, logistics operations, optimization, php, javascript, vue, MySQL, Lavarel.

The qualification work is devoted to the study of methods and means of optimizing CRM systems in logistics operations. The work includes an analysis of existing research in the field of optimizing logistics operations and focuses on the capabilities of CRM systems in logistics operations. To create the system, software written in Javascript, PHP and a database in the form of MySQL were used.

The graphical interface was developed using the Vue and bootstrap frameworks. The platform provides secure user authentication, effective cargo management and interaction between clients. The results of the presented research demonstrate the practical application of modern tools and methods for building CRM systems that meet the functional and productive requirements of the platform for optimizing logistics operations.

ЗМІСТ

ВСТУП

РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА МЕТОДІВ РОЗРОБКИ 11

1.1. Аналіз предметної області 11

1.2. Аналіз сучасних CRM-систем у логістиці..... 12

1.3. Проблеми та виклики у роботі CRM-систем..... 17

1.4. Огляд методів оптимізації..... 19

1.5. Аналіз ефективності інтеграції нових технологій у логістиці 20

1.6. Оцінка ринку та приклади впровадження 23

1.7. Висновки 25

РОЗДІЛ 2 ОБҐРУНТУВАННЯ ВИБОРУ ТЕХНОЛОГІЙ ТА АРХІТЕКТУРИ CRM-СИСТЕМИ 27

2.1. Встановлення архітектури 27

2.2. Використання БД MySQL 30

2.3. Використані фреймворки 33

2.4. Формування основних вимог..... 36

2.5. Визначення варіантів використання 38

2.6. Висновки до розділу 39

РОЗДІЛ 3 ПРАКТИЧНА РЕАЛІЗАЦІЯ CRM-СИСТЕМИ ДЛЯ ЛОГІСТИКИ..... 40

3.1. Реалізація серверної частини 40

3.2. Клієнтська частина..... 47

3.3. Перевірка роботи системи..... 49

3.4. Висновки до розділу 54

РОЗДІЛ 4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ 55

4.1. Охорона праці.....	55
4.2. Створення комфортних умов праці на логістичному підприємстві	58
4.3. Висновки до розділу	59
ВИСНОВКИ.....	60
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	61
Додаток А. Тези конференцій	64
Додаток Б. Скрипт запуску.....	67

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

CRM – customer relationship management (система

TMS – transport management system (система управління транспортом)

GIS – geographic information system (геоінформаційна система)

FMS – fleet management system (система управління автогосподарством)

WMS – warehouse management system (системи управління складом)

CTMS – Container Terminal Management (системи управління терміналом)

SCM – supply chain management (система управління ланцюгами поставок)

GTS (Global Trade Services) - системи забезпечення ЗЕД та торгівлі

REST API - Representational State Transfer Application Programming Interface

CRUD – create, read, update, delete

ВСТУП

Актуальність роботи. Сьогодні логістика є ключовою складовою глобальної економіки, де ефективне управління транспортними, складськими і постачальницькими процесами визначає конкурентоспроможність компаній. CRM-системи відіграють важливу роль у забезпеченні якісної взаємодії з клієнтами, оптимізації бізнес-процесів і підвищенні оперативності роботи.

Швидкий розвиток цифрових технологій створює нові можливості для вдосконалення CRM-систем. Це дозволяє обробляти великі обсяги даних, автоматизувати рутинні операції та інтегрувати різні підсистеми для кращої координації процесів. У логістиці, де кожен етап операцій є критичним, такі інновації сприяють зниженню витрат, прискоренню виконання завдань і покращенню аналітики для стратегічного планування.

Окрім того, сучасні CRM-системи мають адаптуватися до зростаючої складності управління даними, такими як інформація про маршрути, статуси замовлень та історію взаємодії з клієнтами. Їхня оптимізація дозволяє більш ефективно управляти ланцюгами постачання, прогнозувати потреби клієнтів і забезпечувати високий рівень обслуговування. Це є важливим аспектом у боротьбі за клієнта на фоні високої конкуренції в галузі.

Також важливим є внесок оптимізованих CRM-систем у стійкість та інноваційний розвиток логістичних компаній. Завдяки їхньому впровадженню бізнес може швидше реагувати на зовнішні зміни, такі як економічні кризи чи глобальні виклики, та впровадження нових стратегій для досягнення довгострокових цілей.

Таким чином, дослідження методів і засобів оптимізації CRM-систем для логістики є актуальним як з наукової, так і з практичної точки зору. Воно спрямоване на підвищення ефективності взаємодії з клієнтами, покращення бізнес-процесів і забезпечення конкурентних переваг для логістичних компаній у сучасному цифровому світі.

Метою кваліфікаційної роботи є підвищення ефективності логістичних процесів завдяки оптимізації CRM-систем.

Завдання кваліфікаційної роботи:

- проаналізувати сучасні методи оптимізації логістичних процесів;
- обрати модель для подальшої реалізації;
- розробити алгоритм підвищення ефективності;
- реалізувати програму- з урахуванням розробленого алгоритму;

Відповідно до мети та завдань кваліфікаційної роботи визначено її об'єкт та предмет.

Об'єкт дослідження: комп'ютеризовані CRM-системи для підвищення ефективності логістичних процесів

Предмет дослідження: CRM-системи для логістичних процесів.

Методи дослідження: аналіз літератури та огляд сучасних технологій, системний аналіз, методи оптимізації, експериментальне моделювання та тестування.

Наукова новизна дослідження

Вперше запропоновано використання фреймворків Laravel, Vue.js та Vuex, що дозволило оптимізувати CRM-системи у логістичних процесах.

Запропоновано підходи, які включають застосування евристичних методів для підвищення ефективності управління ресурсами та автоматизацію аналізу взаємодії з клієнтами для логістики.

Вперше обґрунтовано використання гібридних технологій, що поєднують аналітику великих даних та математичне моделювання для підвищення продуктивності CRM-систем у логістиці, що дозволяє зменшити витрати часу та ресурсів та підвищити рівень обслуговування клієнтів у логістичних компаніях.

Практичне значення результатів кваліфікаційної роботи: розроблене програмне забезпечення можна використовувати у логістичних компаніях різних рівнів для оптимізації роботи з замовленнями та клієнтами.

Публікації. Результати дослідження представлено у вигляді 2 тез конференцій.

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА МЕТОДІВ РОЗРОБКИ

1.1. Аналіз предметної області

Сьогодні автоматизація в транспортній галузі стала невід'ємною частиною бізнесу. Великі транспортні компанії активно використовують програмне забезпечення для оформлення замовлень, відстеження перевезень, складського та бухгалтерського обліку. Замовлення можна легко оформити через сайт чи мобільний застосунок, що значно спрощує роботу і прискорює процеси.

Однак, поки великі компанії вже давно впровадили електронний документообіг та автоматизували багато рутинних процесів, середні та малі підприємства часто відстають в цьому напрямку. Вони все ще використовують традиційні методи спілкування з клієнтами та обробки інформації, що може негативно впливати на ефективність їх роботи.

Автоматизація процесів у середніх та малих транспортних компаніях надає можливість досягти суттєвих переваг, серед яких збільшення ефективності роботи, оптимізація витрат, підвищення якості обслуговування клієнтів та зміцнення позицій на ринку.

У світі транспортної логістики існує безліч інформаційних систем, кожна з яких призначена для вирішення певних завдань. Їх можна класифікувати за різними критеріями, наприклад, за функціональним призначенням.

До систем управління транспортом відносяться TMS (Transportation Management System), які допомагають планувати маршрути, контролювати перевезення та оптимізувати логістичні процеси. GIS (Geographic Information System), що дозволяють візуалізувати та аналізувати дані на картах, та FMS (Fleet Management Systems), які допомагають контролювати та управляти автопарком.

Для управління складськими операціями використовуються WMS (Warehouse Management System), які оптимізують процеси прийому, зберігання та відвантаження

товарів. CTMS (Container Terminal Management) призначені для управління контейнерними терміналами.

Управління ланцюгами поставок здійснюється за допомогою SCM (Supply Chain Management), які координують всі етапи руху товарів від виробника до споживача. GTS (Global Trade Services), включаючи митне оформлення, допомагають компаніям управляти міжнародною торгівлею.

Існують також системи для управління фінансами (FRP - Finance Requirements Planning), планування продажів та операційної діяльності (S & OP - Sales & Operation Planning), управління документообігом (CMS - Content Management System), персоналом (HRM - Human Resources Management) та взаємовідносинами з постачальниками (SRM - Supplier Relationship Management) та замовниками (CRM - Customer Relationship Management).

Окремо варто виділити ITS (Intelligent Transportation System), які використовують сучасні технології для управління транспортними потоками та підвищення безпеки дорожнього руху.

Всі ці системи відіграють важливу роль в сучасній логістиці, допомагаючи компаніям оптимізувати свої операції, знижувати витрати та підвищувати ефективність бізнесу.

1.2. Аналіз сучасних CRM-систем у логістиці

Сучасні CRM-системи є основою роботи у сучасних логістичних компаніях, вони забезпечують ефективну взаємодію з клієнтами, моніторинг процесів доставки та аналіз ключових показників ефективності. Завдяки їхньому впровадженню

логістичні оператори можуть оптимізувати операційні процеси, знижувати витрати та підвищувати якість обслуговування клієнтів.

На рисунку 1.1 представлено графік підвищення тенденції використання CRM-систем логістичними компаніями.

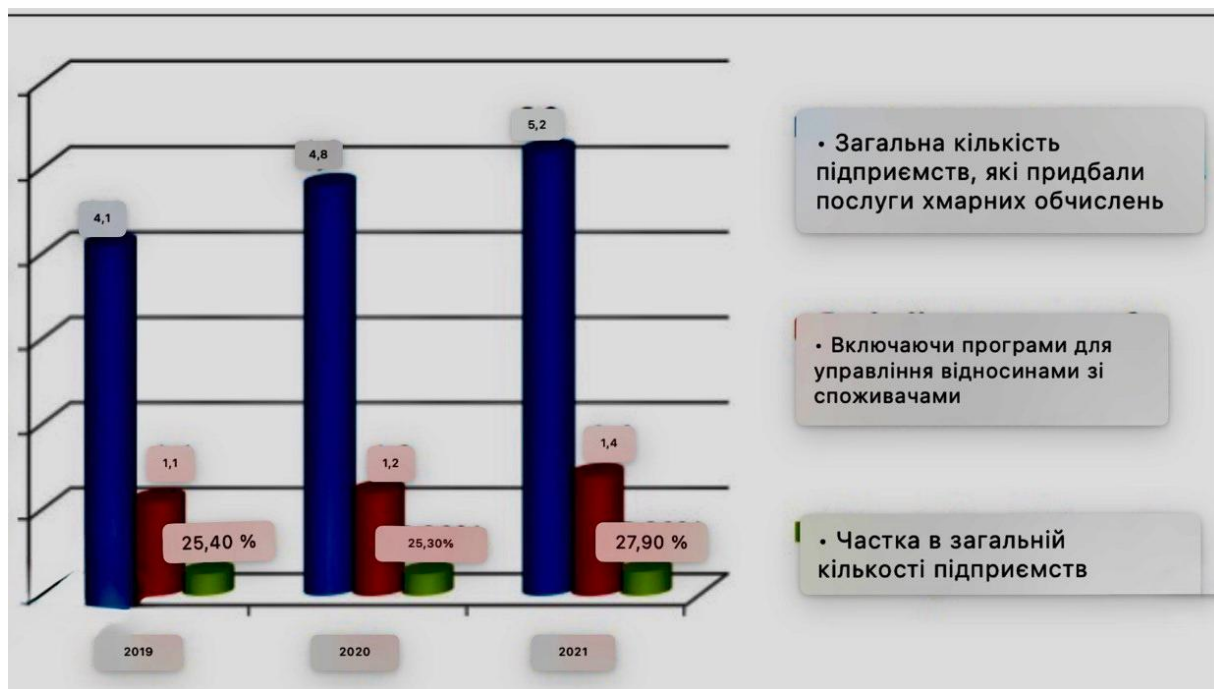


Рис. 1.1. Збільшення тенденції використання CRM-систем

За статистичними даними, середній щорічний приріст світового ринку CRM-систем становить 10.68%, і станом на 2023 рік оцінюється в 79.4 млрд доларів (рис. 1.2), до 2029 очікується що ця цифра виросте до 131.9 млрд доларів, це вказує на те, що все більше світових компаній будуть рухатися у бік впровадження CRM-технологій.

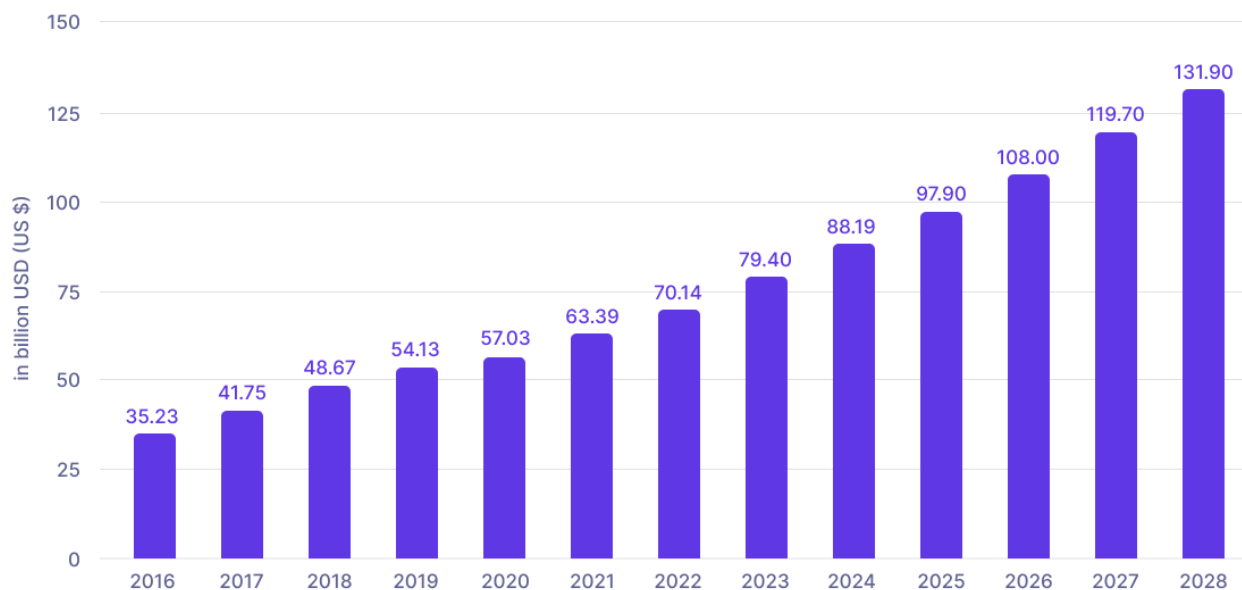


Рис. 1.2. Графік впровадження CRM-технологій

Аналіз методів оптимізації вказує на ефективність використання алгоритмів маршрутизації та математичного моделювання. Інтеграція сучасних технологій, таких як IoT та Big Data, дозволяє значно підвищити ефективність CRM у логістиці, знижуючи витрати та покращуючи якість обслуговування.

Однією з основних характеристик сучасних CRM-рішень є спрощення взаємодії з клієнтами. Це включає збір, зберігання та обробку інформації про клієнтів через різні канали, зокрема електронну пошту, соціальні мережі та чат-боти. Ще однією важливою функцією є управління замовленнями, що охоплює повний життєвий цикл замовлення від моменту його створення до доставки, з інтеграцією ERP- та SCM-систем для забезпечення ефективної координації.[14]

CRM-системи також надають можливість моніторингу доставки у реальному часі, що дозволяє як клієнтам, так і компаніям отримувати актуальну інформацію про статус вантажів. Інструменти аналітики та прогнозування, вбудовані в такі системи,

допомагають аналізувати великі дані для формування прогнозів щодо попиту, виявляти проблеми в логістичних процесах та пропонувати оптимальні рішення.

Популярні CRM-системи, такі як Salesforce CRM, Zoho CRM, SAP Customer Experience та Microsoft Dynamics 365, пропонують широкий спектр функцій для логістики, включаючи автоматизацію процесів, інтеграцію з іншими системами, аналітичні модулі та можливості прогнозування. Однак, незважаючи на їх переваги, компанії стикаються з викликами, зокрема високою вартістю впровадження, недостатньою адаптованістю до специфіки логістичних процесів, проблемами масштабованості та необхідністю навчання персоналу.

Одним з прикладів сучасних CRM-систем є Super Dispatch (рис. 1.3), він поєднує у собі всі потрібні функції для проведення логістичних операцій.

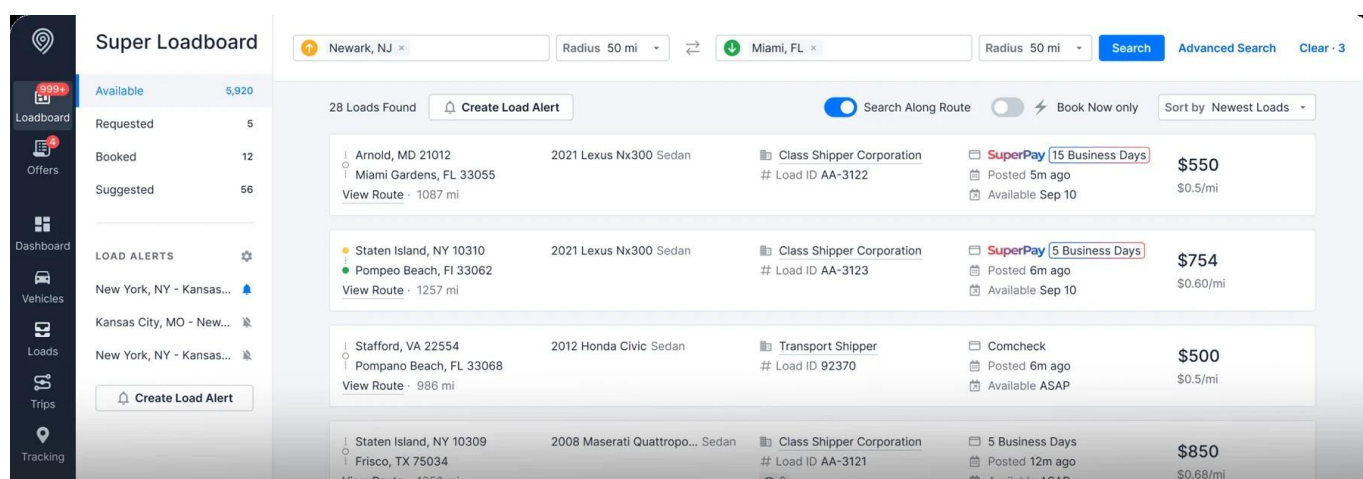


Рис. 1.3. Зовнішній вигляд CRM-системи Super Dispatch

У сфері логістики впровадження CRM-систем є ключовим фактором для покращення взаємодії з клієнтами та підвищення ефективності логістичних операцій. Існує багато факторів, які CRM-системи покращують після свого впровадження, вони представлені на рисунку 1.4.

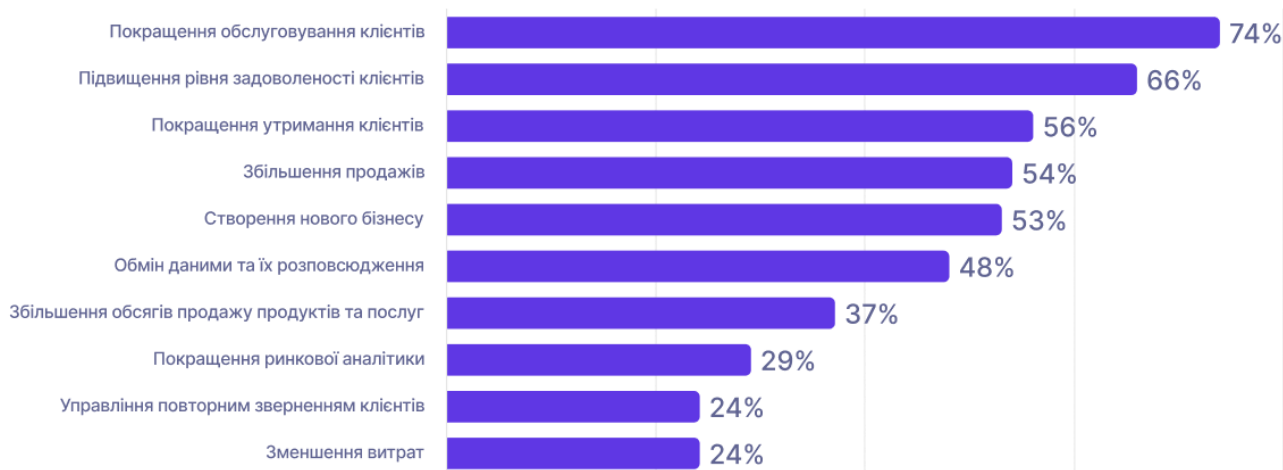


Рис. 1.4. Ключові переваги впровадження CRM-систем

Розглянемо функціональні можливості та результати впровадження трьох популярних рішень: Salesforce, Microsoft Dynamics 365 та Super Dispatch.

Salesforce пропонує широкий спектр інструментів для управління продажами, обслуговування клієнтів та маркетингу, гнучкість її архітектури дозволяє адаптувати систему під особливі потреби логістичних компаній, забезпечуючи інтеграцію з різними платформами та додатками. Згідно з даними Salesforce, клієнти, які перейшли з інших CRM-систем, відзначають 34% зростання ROI, 30% підвищення продуктивності завдяки використанню штучного інтелекту та 32% збільшення доходів від продажів.

Microsoft Dynamics 365 є комплексним рішенням, що поєднує CRM та ERP-функціонал. Для логістичних компаній це означає можливість управління ланцюгами постачання, фінансами та взаємодією з клієнтами в єдиній системі. Інтеграція з іншими продуктами Microsoft, такими як Office 365 та Azure, забезпечує безшовну роботу та підвищує продуктивність. Деякі компанії, що перейшли з Salesforce на Dynamics 365, відзначають покращення в управлінні даними та зниження витрат на впровадження.

Super Dispatch спеціалізується на автоматизації процесів у сфері автомобільних перевезень. Її функціонал включає управління замовленнями, відстеження вантажів у реальному часі та електронний документообіг. Це рішення орієнтоване на

перевізників та брокерів, забезпечуючи спрощення операцій та зменшення паперової роботи. Користувачі відзначають підвищення ефективності диспетчеризації та зниження кількості помилок при обробці замовлень.

Вибір між цими CRM-системами залежить від специфіки бізнесу та його потреб. Salesforce підходить для компаній, що шукають гнучке та масштабоване рішення з потужними аналітичними можливостями. Microsoft Dynamics 365 буде корисним для тих, хто прагне інтегрувати CRM з існуючими бізнес-процесами та продуктами Microsoft. Super Dispatch є оптимальним вибором для компаній, що спеціалізуються на автомобільних перевезеннях і потребують спеціалізованих інструментів для цієї галузі.

Підсумовуючи, сучасні CRM-системи мають великий потенціал для оптимізації логістичних процесів, проте їх ефективність можна значно підвищити шляхом впровадження спеціалізованих інструментів, таких як алгоритми маршрутизації, прогнозування попиту та інтеграція з IoT-пристроями. Ці напрямки вдосконалення стануть основою подальшого розвитку CRM-рішень у логістиці.

1.3. Проблеми та виклики у роботі CRM-систем

Впровадження CRM-системи може значно покращити роботу логістичної компанії, але важливо враховувати потенційні труднощі. Одна з них - це інтеграція CRM з вже існуючими системами, такими як WMS та TMS, а також з різноманітними джерелами даних, що використовуються в логістиці (GPS-трекери, мобільні додатки тощо). Це може бути складним та затратним процесом.



Рис. 1.5. Графік впровадження CRM-систем в Україні

Графік, що ілюструє особливості впровадження CRM-систем в Україні (рис. 1.5) протягом 2022-2023 років, демонструє кілька ключових тенденцій. По-перше, спостерігається загальне зниження інтересу до CRM-рішень, про що свідчить зменшення кількості запитів у Google. Ймовірно, це пов'язано з впливом війни, економічною нестабільністю та можливим перенасиченням ринку. Це може вказувати на те, що впровадження CRM-системи є трудомістким процесом і не кожна логістична компанія навіть середнього розмаху може собі це дозволити в складних економічних умовах.

Крім того, CRM-система повинна бути здатною ефективно обробляти величезні обсяги інформації про клієнтів, замовлення, маршрути та транспорт, а також надавати інструменти для аналізу цих даних. Це дозволить оптимізувати маршрути, управляти запасами та приймати обґрунтовані рішення.

Необхідно також враховувати специфічні потреби логістики. Система повинна забезпечувати відстеження вантажів в режимі реального часу, планування оптимальних маршрутів з урахуванням різних факторів (час, дорожні умови, обмеження ваги) та облік особливостей різних типів вантажів.

Важливим аспектом є мобільність. Водії та кур'єри повинні мати доступ до CRM через мобільні пристрої, щоб оновлювати статус доставки, отримувати завдання

та спілкуватися з диспетчерами, навіть перебуваючи в зонах з обмеженим доступом до інтернету.

Не можна забувати і про безпеку даних. CRM-система містить конфіденційну інформацію, яку необхідно захищати від несанкціонованого доступу. Система також повинна відповідати вимогам законодавства щодо захисту персональних даних.

Врешті-решт, впровадження та обслуговування CRM-системи може бути дорогим, а навчання персоналу потребує часу та ресурсів.

Обдуманий підхід до вибору та впровадження CRM з урахуванням всіх цих факторів допоможе логістичній компанії оптимізувати свої бізнес-процеси та підвищити ефективність роботи.

1.4. Огляд методів оптимізації

Оптимізація в логістиці є ключовим фактором успіху, а CRM-системи відіграють важливу роль у цьому процесі. Для вирішення завдань оптимізації в CRM застосовуються різноманітні методи, кожен з яких має свої плюси і мінуси.

Алгоритми маршрутизації є основою для планування оптимальних маршрутів доставки. Вони враховують різні фактори, такі як відстань, час, дорожні умови, обмеження ваги та часові вікна доставки. Існують різні види алгоритмів, від класичних (алгоритм Дейкстри, алгоритм найближчого сусіда) до більш складних метаевристичних алгоритмів (генетичні алгоритми, алгоритми мурашиних колоній). Вибір алгоритму залежить від особливості поставленої задачі та вимог до точності та швидкості розрахунку.

Машинне навчання дозволяє CRM-системам "вчитися" на історичних даних та прогнозувати майбутні події. Наприклад, можна прогнозувати час доставки з урахуванням трафіку, погодних умов та інших факторів, або прогнозувати попит на послуги в різних регіонах. Це спрощує розподіл ресурсів та підвищує ефективність роботи.

Математичне моделювання дозволяє створювати математичні моделі логістичних процесів та знаходити оптимальні рішення за допомогою математичних

методів. Наприклад, можна моделювати роботу складу, оптимізувати розміщення товарів та мінімізувати час комплектації замовлень.

Придатність методів для CRM в логістиці:

1) Алгоритми маршрутизації є необхідним елементом будь-якої CRM-системи в логістиці, оскільки дозволяють планувати оптимальні маршрути та знижувати транспортні витрати.

2) Машинне навчання дозволяє підвищити точність прогнозування та автоматизувати прийняття рішень, що особливо важливо в умовах великих обсягів даних та мінливого середовища.

3) Математичне моделювання дозволяє аналізувати складні логістичні процеси та оптимізовувати їх вирішення для підвищення ефективності роботи.

В цілому, поєднання різних методів оптимізації в CRM-системах дозволяє логістичним компаніям значно підвищити ефективність своєї роботи, зменшити видатки та забезпечити кращу підтримку клієнтів.

1.5. Аналіз ефективності інтеграції нових технологій у логістиці

CRM-системи значно спрощують обмін інформацією та співпрацю між співробітниками, що позитивно впливає на ефективність роботи транспортного підприємства в цілому. Завдяки швидкому доступу до даних, підвищується продуктивність та покращується взаємодія в команді.

CRM впроваджується з метою формування централізованого інструменту, який дозволяє ефективно працювати як з потенційними, так і з лояльними клієнтами. CRM дозволяє ефективно керувати взаємовідносинами з клієнтами, перетворюючи випадкових покупців на лояльних клієнтів та бізнес-партнерів. Завдяки аналізу даних про клієнтські запити та преференції, компанія може підвищити рівень задоволеності клієнтів та оптимізувати свою тарифну політику. Автоматизація процесів дозволяє мінімізувати ручну працю та оперативно реагувати на потреби клієнтів, а також вчасно виявляти потенційні ризики. [16]

Важливою перевагою CRM-систем є їх універсальність. Вони можуть бути корисними для різних відділів компанії, від продажів та обслуговування клієнтів до маркетингу та розвитку бізнесу. Крім того, CRM підходить для транспортних компаній будь-якого розміру, допомагаючи їм ефективно управляти взаємовідносинами з клієнтами та оптимізувати бізнес-процеси.

CRM надає широкі можливості для зберігання інформації про клієнтів, обліку проблем та запитів, виявлення потенційних продажів та управління процесом перевезення. Це особливо актуально для компаній, які стикаються з проблемами в організації роботи з клієнтами, такими як відсутність централізованої бази даних, втрата важливої інформації, неефективний контроль продажів та недостатня увага до клієнтських запитів.

Хоч інтеграція CRM-системи має потенціал для значного покращення бізнес-процесів, необхідно ретельно проаналізувати фінансову вигоду від таких вкладень. Ключовим інструментом для цього є розрахунок рентабельності інвестицій (ROI), який визначається шляхом ділення прибутку, отриманого від впровадження CRM, на загальну вартість проекту, з подальшим множенням результату на 100%.

До витрат на впровадження CRM належать придбання ліцензії або розробка системи, інтеграція з існуючими системами та адаптація до специфіки компанії, навчання персоналу для ефективного використання інструментів, а також постійна технічна підтримка та оновлення. Основні вигоди включають зростання продажів завдяки персоналізованому підходу, зменшення операційних витрат через автоматизацію, покращення обслуговування клієнтів завдяки швидкому доступу до їхньої інформації, а також підвищення ефективності маркетингових кампаній через краще таргетування та аналіз.

Приклади підтверджують доцільність впровадження. Згідно з даними компанії Wezom, у пострадянських країнах CRM може окупитися за три місяці, скорочуючи вартість залучення клієнтів і зменшуючи відтік користувачів. Інший приклад наводить компанія KeerpinCRM, де впровадження системи збільшило продажі на 20% і скоротило витрати на 10%, забезпечивши позитивний ROI протягом першого року використання.

Для підвищення економічної ефективності важливо ретельно вибрати систему, яка найбільше відповідає потребам бізнесу, і розробити детальний план інтеграції. Значну увагу слід приділити навчанню персоналу, що дозволить максимально використовувати можливості CRM, а також регулярно моніторити ефективність і вносити корективи для досягнення оптимальних результатів. Враховуючи всі аспекти, впровадження CRM-системи може бути економічно вигідним рішенням за умов правильного планування і оцінки.

Опрацювання великих масивів інформації за допомогою Big Data аналітики надає можливість виявляти важливі закономірності та отримувати корисні знання з потоку даних, що виникають в процесі логістичної діяльності. В CRM-системах Big Data може використовуватися для виявлення тенденцій та закономірностей. Аналіз даних про клієнтів, замовлення, маршрути та транспорт може допомогти виявити тенденції та закономірності, що впливають на ефективність логістичних операцій. Також Big Data дозволяє оптимізувати маршрути та розклади доставки, аналізуючи дані про трафік, дорожні умови та інші фактори, що знижує витрати на транспорт та покращує своєчасність доставки. Крім того, аналіз даних про задоволеність клієнтів, скарги та відгуки може допомогти виявити проблемні місця та покращити якість обслуговування.

Інтернет речей надає можливість збирати дані з різних датчиків та пристроїв, таких як GPS-трекери, датчики температури, вологості та інші. В CRM-системах IoT може використовуватися для відстеження вантажів у режимі реального часу. Дані з GPS-трекерів дозволяють відстежувати місцезнаходження вантажів у режимі реального часу, що в свою чергу підвищує прозорість та контроль над процесом доставки. Також IoT дозволяє моніторити стан вантажів під час транспортування, що особливо важливо для чутливих до умов зберігання товарів. Аналіз даних з IoT-пристроїв може допомогти прогнозувати та запобігати проблемам, таким як затримки доставки, пошкодження вантажів та інші.

Автоматизація дозволяє автоматизувати рутинні завдання та процеси в логістиці. В CRM-системах автоматизація може використовуватися для автоматизації обробки замовлень. Автоматичне створення замовлень, підготовка документів,

планування маршрутів та інші завдання можуть бути автоматизовані, що скорочує час обробки та знижує ризик помилок. Також автоматизація може використовуватися для автоматизації комунікації з клієнтами. Для покращення взаємодії з клієнтами та підвищення їх лояльності, CRM-системи дозволяють налаштувати автоматичне інформування про стан замовлень, нагадування про терміни доставки та інші важливі повідомлення.

Функціонал CRM-систем дає можливість автоматично формувати звіти та аналітичні дані, що сприяє ефективному моніторингу ключових показників та прийняттю зважених управлінських рішень. Впровадження інноваційних технологій в CRM-системи відкриває перед логістичними компаніями широкі можливості для оптимізації бізнес-процесів, скорочення витрат, підвищення якості обслуговування та зміцнення своїх позицій на ринку.

1.6. Оцінка ринку та приклади впровадження

Сьогодні на ринку існує багато CRM-систем для контролю логістичних операцій. Серед прикладів можна навести Oracle Siebel (рис. 1.6), Super Dispatch, Salesforce sales Cloud, SAP CRM, Microsoft Dynamics 365 та інші. Усі вони містять у собі хмарні технології, системи взаємодії з клієнтами та контролю за замовленнями та доставкою.

До прикладу використання CRM систем серед великих світових компаній можна навести DHL, який, зіткнувшись з неефективним управлінням взаємовідносинами з клієнтами та обробкою замовлень, впровадив Salesforce Sales Cloud (рис. 1.7). Це дозволило централізувати дані про клієнтів, автоматизувати процеси та покращити комунікацію, що призвело до зростання задоволеності клієнтів, скорочення часу обробки замовлень та підвищення ефективності продажів.

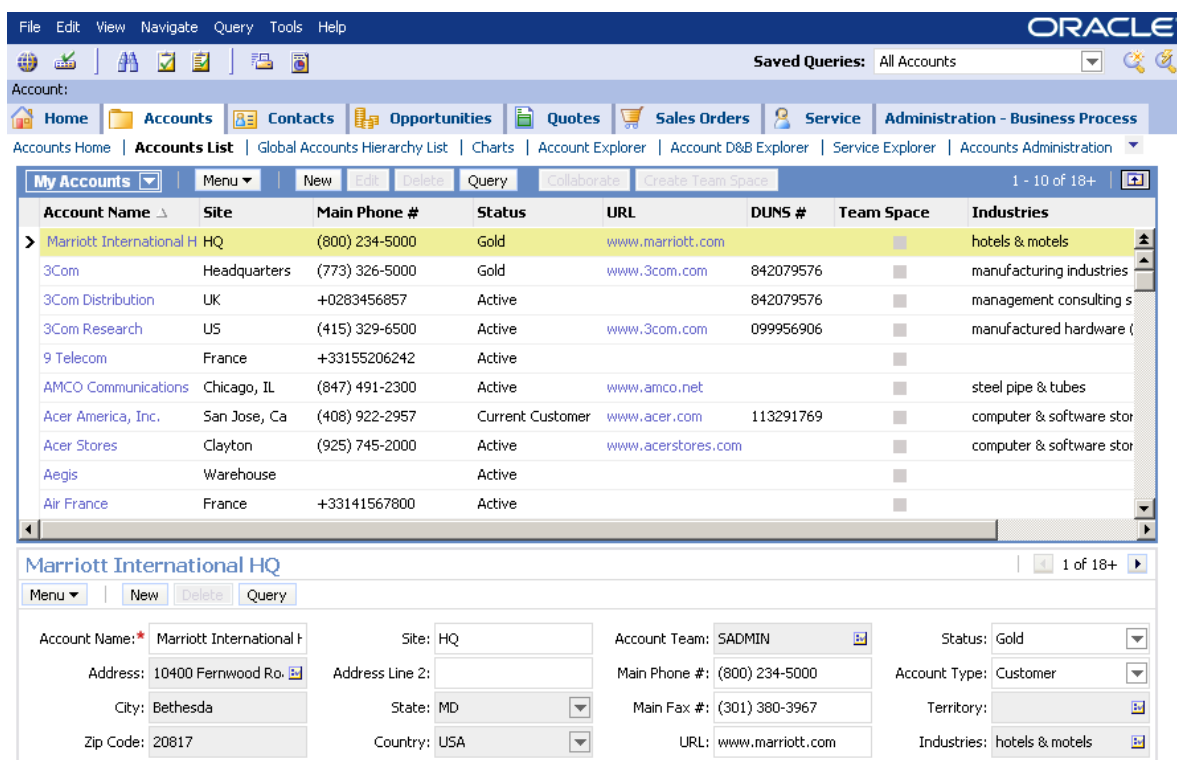


Рис. 1.6. Зовнішній вигляд Oracle Siebel

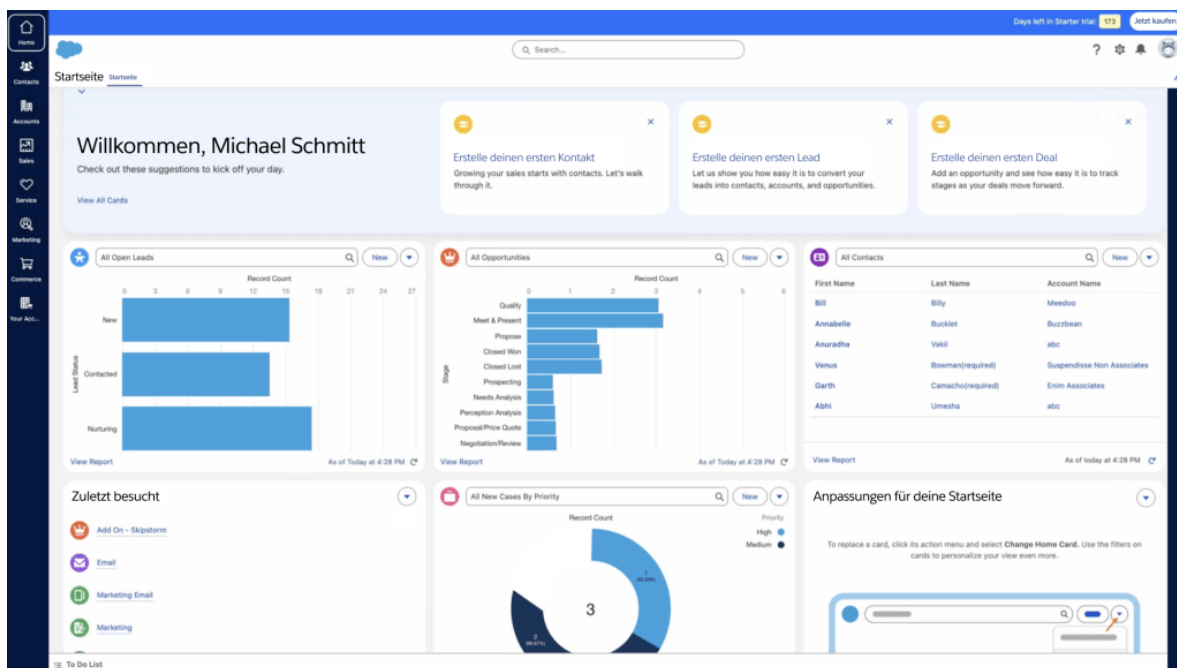


Рис. 1.7. Зовнішній вигляд Salesforce Sales Cloud

З метою підвищення ефективності логістичних операцій та зменшення витрат, компанія Kuehne + Nagel інтегрувала CRM-систему з розширеним функціоналом для планування маршрутів та аналізу даних. Це дозволило оптимізувати логістичні

процеси, зокрема, скоротити витрати на пальне та забезпечити більш пунктуальну доставку, що сприяло покращенню як фінансових показників, так і екологічної сталості компанії.

Maersk, прагнучи покращити прозорість та контроль над процесом доставки, впровадив CRM-систему з інтеграцією IoT-пристроїв. Це дозволило відстежувати вантажі в режимі реального часу, моніторити їх стан та надавати клієнтам актуальну інформацію, що зменшило кількість затримок та втрат, а також підвищило задоволеність клієнтів.

В цілому, CRM-системи в логістиці сприяють підвищенню продуктивності за рахунок автоматизації процесів, оптимізації маршрутів та покращення комунікації. Вони також допомагають знизити витрати, оптимізуючи використання ресурсів та зменшуючи кількість помилок. Крім того, CRM-системи сприяють підвищенню задоволеності клієнтів завдяки персоналізованому підходу, своєчасній доставці та прозорості процесу. В результаті, компанії збільшують доходи, розширюють клієнтську базу та покращують свою репутацію. Ринок CRM-систем в логістиці активно зростає, і впровадження таких систем стає необхідністю для компаній, які прагнуть залишатися конкурентоспроможними.

1.7. Висновки

Проведений аналіз предметної області та сучасних CRM-систем у логістиці засвідчив важливість автоматизації бізнес-процесів у транспортній галузі.

Виявлено, що великі компанії активно використовують інформаційні системи, які забезпечують управління перевезеннями, складськими запасами та документообігом. Однак середні та малі підприємства часто відстають у впровадженні таких рішень, що знижує їх конкурентоспроможність, тому запропоновано впровадження CRM-систем, адаптованих до потреб логістики.

Встановлено, що сучасні CRM-рішення мають великий потенціал для оптимізації логістичних процесів завдяки автоматизації взаємодії з клієнтами,

моніторингу доставки у реальному часі, аналітиці великих даних і прогнозуванню попиту.

Виявлено, що впровадження CRM-систем стикається з проблемами, такими як інтеграція з іншими системами, обробка великих обсягів даних і дотримання вимог до безпеки.

РОЗДІЛ 2

ОБҐРУНТУВАННЯ ВИБОРУ ТЕХНОЛОГІЙ ТА АРХІТЕКТУРИ CRM-СИСТЕМИ

2.1. Встановлення архітектури

Дослідивши предметну область та сценарії використання, ми визначили, що найкращою структурою для нашого веб-застосунку буде трирівнева архітектура (рис. 2.1).

1) Рівень представлення (presentation layer): відповідає за відображення інформації користувачеві та взаємодію з ним.

2) Структурно-логічний рівень (logical layer): обробляє дані та реалізує бізнес-логіку застосунку.

3) Рівень доступу до даних (data access layer): забезпечує взаємодію з базою даних та зберігання інформації.

Такий підхід дозволить створити гнучкий та масштабований застосунок, який легко підтримувати та розвивати [17].

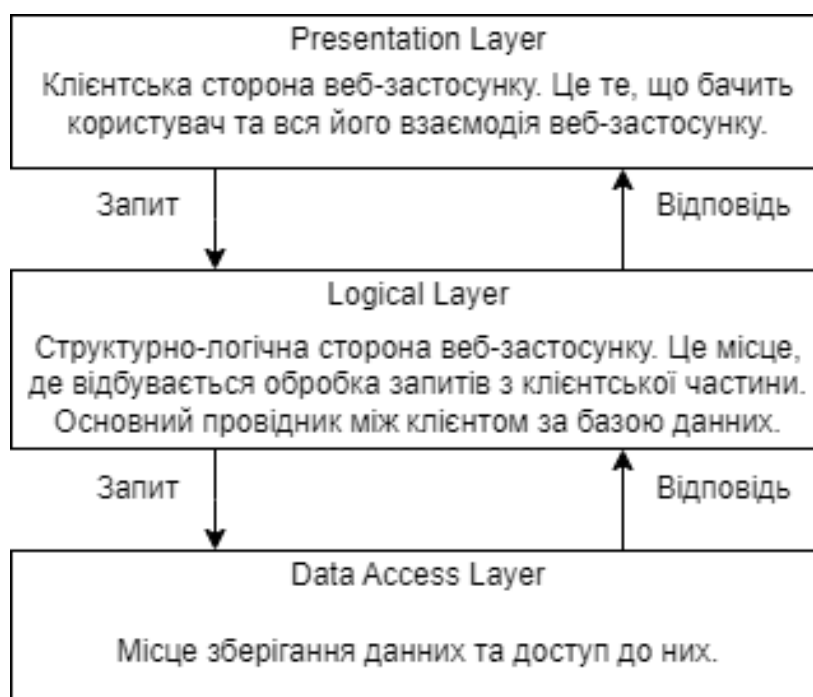


Рис. 2.1. Трирівнева архітектура

Запити до серверної частини відбуваються на рівні клієнта за допомогою бібліотеки Axios. Axios - це JavaScript бібліотека, призначена для спрощення роботи з HTTP-запитами. Вона надає зручний інтерфейс для взаємодії з сервером, дозволяючи легко відправляти запити та обробляти отримані відповіді.

Однією з головних переваг Axios є використання Promise API, що робить асинхронний код більш структурованим та зрозумілим. Це особливо важливо при роботі з великою кількістю запитів, де важливо відстежувати їх стан та обробляти потенційні помилки.

Axios універсальний інструмент, який можна використовувати як у браузері, так і в середовищі Node.js. Це робить його придатним для широкого спектру задач, пов'язаних з веб-розробкою. Бібліотека автоматично перетворює відповіді сервера у формат JSON, що полегшує роботу з даними. Крім того, Axios має вбудовані механізми для перехоплення запитів та відповідей, що дозволяє, наприклад, додавати заголовки авторизації або обробляти помилки [24].

Серед інших корисних функцій Axios варто відзначити можливість скасування запитів та захист від XSRF-атак. Завдяки своїй простоті, гнучкості та широкому набору функцій, Axios є популярним вибором серед розробників, а велика спільнота гарантує наявність документації та підтримки.

Для забезпечення безпеки авторизованих запитів у проекті використовується механізм токенів, реалізований за допомогою Laravel Sanctum. Цей підхід дозволяє створювати токени доступу після успішної авторизації користувача, що забезпечує захищену взаємодію між клієнтом і сервером.

Після надсилання запиту на маршрут /login сервер обробляє дані, перевіряючи правильність введених облікових даних. У цьому процесі здійснюється валідація введених користувачем параметрів, таких як електронна пошта та пароль. Валідація забезпечує відповідність даних певним вимогам, зокрема перевіряється обов'язковість полів і формат електронної пошти.

Після цього система проводить ідентифікацію користувача в базі даних, використовуючи введену електронну адресу як ключ пошуку. Якщо користувач не

знайдений або введений пароль не збігається з хешованим у базі, сервер повертає повідомлення про помилку автентифікації із кодом відповіді 404.

У разі успішного введення облікових даних створюється унікальний токен доступу за допомогою методу `createToken()`. Токен генерується для ідентифікації користувача і зберігається в системі, що дозволяє розпізнавати авторизовані запити в майбутньому. Згенерований токен передається клієнту у відповідь разом із даними про авторизованого користувача.

Для захисту маршрутів використовується `middleware auth:sanctum`, який перевіряє наявність та дійсність токена в кожному запиті до захищених ресурсів. Наприклад, при запиті до маршруту `/user`, токен, переданий у заголовках запиту, використовується для автентифікації користувача. Якщо токен дійсний, сервер повертає відповідні дані про користувача.

Використання унікальних ключів доступу для кожного користувача є запорукою високого рівня безпеки системи. Крім того, `Laravel Sanctum` дозволяє легко інтегрувати цей механізм у проєкт і налаштовувати доступ до ресурсів відповідно до потреб застосунку. Реалізація токенів також спрощує підтримку багатоплатформених застосунків, забезпечуючи безпечний доступ до API з різних пристроїв та клієнтів.

Для створення візуальної частини проєкту застосовано каскадні таблиці стилів (CSS) у поєднанні з фреймворком `Bootstrap`, що є широко використовуваним інструментом фронтенд розробки. Таке рішення дозволяє суттєво скоротити час та зусилля, необхідні для розробки інтерфейсу,

Фреймворк `Bootstrap` пропонує розробникам велику колекцію готових до використання стилів та інтерфейсних елементів, включаючи кнопки, форми, модальні вікна та багато іншого. Це дозволяє зосередитись на функціональності застосунку, не витрачаючи зайвий час на написання базових CSS правил. Крім того, `Bootstrap` забезпечує адаптивність дизайну до різних розмірів екранів, що є критично важливим в сучасному світі мобільних пристроїв.

Завдяки гнучкості `Bootstrap`, ми можемо легко налаштовувати стилі компонентів відповідно до проєктних вимог. Це досягається за допомогою

використання власних CSS правил, які перевизначають або доповнюють стандартні стилі Bootstrap. Таким чином, ми можемо створити унікальний та привабливий інтерфейс, зберігаючи при цьому переваги використання готових компонентів та адаптивності.

Використання CSS з бібліотекою Bootstrap в даному проєкті є оптимальним рішенням, яке дозволяє ефективно поєднувати швидкість розробки, якість дизайну та адаптивність до різних пристроїв.

Для обміну даними між клієнтською та серверною частинами застосовується архітектурний стиль REST API. Він встановлює чіткі принципи та стандарти для проектування веб-сервісів, що дозволяє різним системам ефективно взаємодіяти та обмінюватися інформацією в уніфікований спосіб.

2.2. Використання БД MySQL

Для бази даних у проєкті було використано MySQL.

MySQL - це реляційна база даних, яка чудово підходить для розробки CRM-систем. Її структура дозволяє організувати інформацію про клієнтів, замовлення, маршрути, транспорт та склади у вигляді пов'язаних таблиць, що забезпечує ефективне зберігання та обробку даних. Завдяки універсальній мові SQL, можна легко виконувати складні запити та аналізувати інформацію для прийняття рішень.

Враховуючи використання фреймворку Laravel та специфіку CRM-системи, оптимальним вибором є саме MySQL, її реляційна модель даних дозволяє ефективно організовувати та зберігати структуровану інформацію про клієнтів, замовлення, транспорт та інші аспекти логістичного бізнесу.

В рамках кваліфікаційної роботи MySQL використовується для реалізації ключових функцій CRM-системи, таких як:

- 1) Зберігання та управління інформацією про клієнтів (контактні дані, історія взаємодії, переваги тощо).
- 2) Обробка замовлень (відстеження статусу, формування документів).

3) Координація логістичних процесів з метою оптимізації маршрутів доставки, контролю вантажоперевезень та ефективного управління складськими ресурсами.

4) Аналіз даних та формування звітів для прийняття управлінських рішень.

Ключовими перевагами MySQL є надійність та продуктивність.

Вона вирізняється високою продуктивністю та здатністю ефективно обробляти значні обсяги інформації, що є необхідною умовою для CRM-систем, які працюють в режимі реального часу. Більше того, MySQL — це надійна та перевірена часом система керування базами даних, яка забезпечує безпеку та цілісність даних.

MySQL також легко масштабується, що дозволяє CRM-системі розвиватися разом з бізнесом. Можна збільшувати ресурси сервера або розподіляти дані між кількома серверами для забезпечення високої продуктивності навіть при зростанні обсягів інформації.

Безпека даних є пріоритетом для будь-якої CRM-системи. MySQL пропонує інструменти контролю доступу та шифрування даних, захищаючи інформацію від несанкціонованого доступу. Завдяки цьому забезпечується надійний захист інформації клієнтів та компанії.

Ще однією перевагою MySQL є її доступність. Вона безкоштовна у використанні та надає відкритий вихідний код, що знижує загальні витрати на розробку CRM. Крім того, велика та активна спільнота користувачів MySQL забезпечує підтримку та доступ до різноманітних ресурсів.

У базі даних використано велику кількість таблиць для зберігання даних, список яких можна побачити на рисунку 2.2.

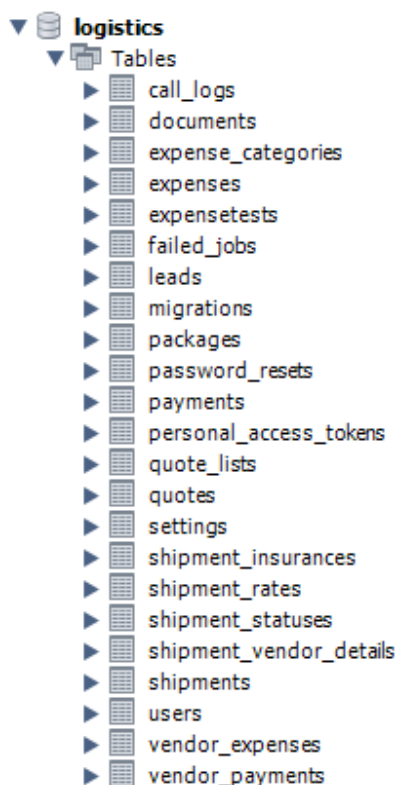


Рис. 2.2. Створені таблиці у MySQL

Також, на рисунку 2.3 представлено вигляд зв'язків між поданими таблицями.

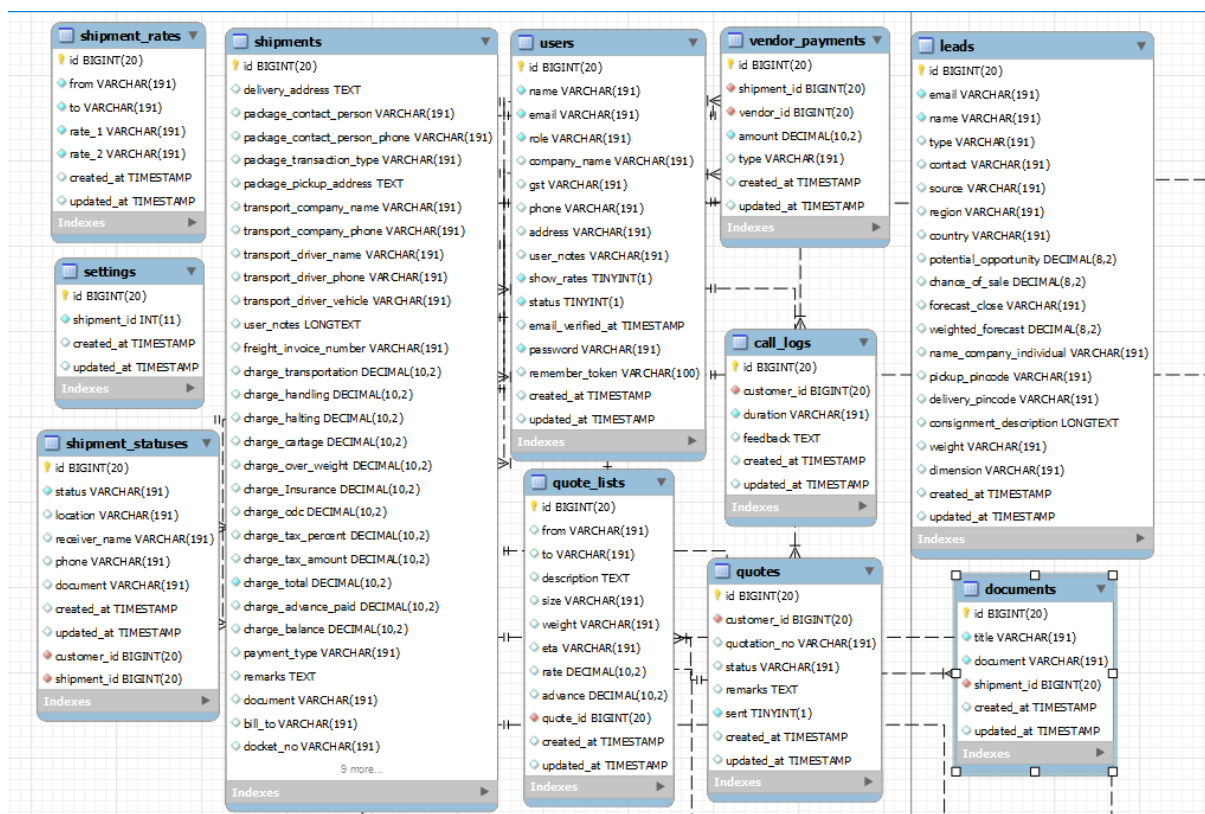


Рис. 2.3. Зв'язки між таблицями у базі даних

Ще однією перевагою MySQL є її простота в налаштуванні та використанні. Вона має інтуїтивно зрозумілий інтерфейс та широку документацію, що спрощує розробку та підтримку застосунку. Крім того, MySQL є безкоштовною та має велику спільноту користувачів, що гарантує доступність підтримки та різноманітних ресурсів.

Використання MySQL дозволяє створити ефективну та надійну CRM-систему, яка сприятиме оптимізації логістичних процесів та покращенню взаємодії з клієнтами.

2.3. Використані фреймворки

Кваліфікаційна робота являє собою CRM-систему з відкритим вихідним кодом, спеціально розроблену для потреб логістичних компаній. Вона пропонує широкий спектр функцій для управління різними аспектами логістичних операцій, такими як створення транспортних документів, відстеження відправлень та контроль витрат.

Для розробки цієї системи було використано сучасний стек технологій, що включає Laravel, Vue.js, Vuex та Laravel Sanctum. Laravel, як популярний PHP фреймворк, забезпечує надійну основу для бекенду, а Vue.js надає інструменти для створення інтерактивного та зручного інтерфейсу користувача. Vuex використовується для ефективного управління станом застосунку, а Laravel Sanctum забезпечує безпечну автентифікацію користувачів та API запитів.

Серверна частина кваліфікаційної роботи розроблена з використанням мови програмування PHP та популярного фреймворку Laravel. Laravel обрано через його елегантну архітектуру, широкому набору інструментів та можливостям, що значно спрощують розробку складних веб-застосунків.

Laravel вирізняється чітким та зрозумілим синтаксисом, що дозволяє розробникам приділяти більше уваги реалізації функціональності, а не заглиблюватися в технічні нюанси. Вбудовані інструменти для роботи з БД, включаючи системи міграцій та об'єктно-реляційне відображення (Eloquent ORM),

значно спрощують взаємодію з даними, усуваючи необхідність в ручному написанні складних SQL-запитів. Крім того, Laravel пропонує інструменти для роботи з чергами, подіями та кешуванням, що дозволяє створювати продуктивні й масштабовані рішення.

Фреймворк також надає розробникам доступ до багатьох готових рішень і бібліотек, які полегшують впровадження типових завдань, таких як відправка електронної пошти, інтеграція API або обробка платежів. Laravel має активну спільноту, яка постійно підтримує й удосконалює платформу, а також розширену документацію, яка робить процес вивчення і роботи з фреймворком значно простішим навіть для новачків.

Фреймворк Laravel має широкий набір інструментів для створення REST API, включаючи механізми маршрутизації, обробки запитів, валідації даних та серіалізації. Це дозволило розробити гнучкий та масштабований API, який легко інтегрується з фронтендом та іншими зовнішніми сервісами. Застосування REST API забезпечує високу продуктивність, надійність та спрощує підтримку бекенду проєкту.

Vue.js - це сучасний JavaScript фреймворк, який значно спрощує створення інтерактивних веб-інтерфейсів та односторінкових застосунків. Розроблений колишнім співробітником Google, він швидко здобув популярність завдяки своєму зручному синтаксису, гнучкості та високій продуктивності. Vue.js дозволяє розробникам будувати інтерфейси з невеликих, незалежних компонентів, які можна легко комбінувати та повторно використовувати.

Vue.js вирізняється своєю реактивною природою. Це означає, що будь-які зміни даних миттєво синхронізуються з користувацьким інтерфейсом, роблячи веб-застосунки більш динамічними та інтерактивними. Крім того, Vue.js використовує шаблонний синтаксис на основі HTML, що полегшує його вивчення та інтеграцію в існуючі проєкти.

Vue.js має багату екосистему інструментів та бібліотек, які допомагають розробникам створювати складні застосунки. Vue Router спрощує створення односторінкових застосунків, Vuex дозволяє ефективно управляти станом

застосунку, а Vue CLI прискорює процес розробки завдяки готовим шаблонам та інструментам.

Vuex - це бібліотека для управління станом у Vue.js застосунках. Вона діє як централізоване сховище даних для всіх компонентів в застосунку, гарантуючи, що стан може бути змінений лише передбачуваним чином. Це особливо корисно у великих та складних проектах, де багато компонентів можуть потребувати доступу та зміни однакових типів даних.

Замість передачі даних між компонентами через props або events, Vuex дозволяє зберігати їх в центральному сховищі, звідки вони можуть бути доступні будь-якому компоненту. Це спрощує структуру проекту, зменшує кількість коду та полегшує відстеження змін стану. Vuex також надає механізми для організації стану в модулі, що дозволяє розділити великі застосунки на менші, більш керовані частини.

Архітектура системи базується на принципах MVC (Model-View-Controller), що забезпечує чіткий розподіл відповідальності між різними компонентами та сприяє підвищенню гнучкості та підтримуваності проекту. Такий підхід дозволяє ефективно організувати розробку та забезпечити високу якість коду (рис. 2.4).

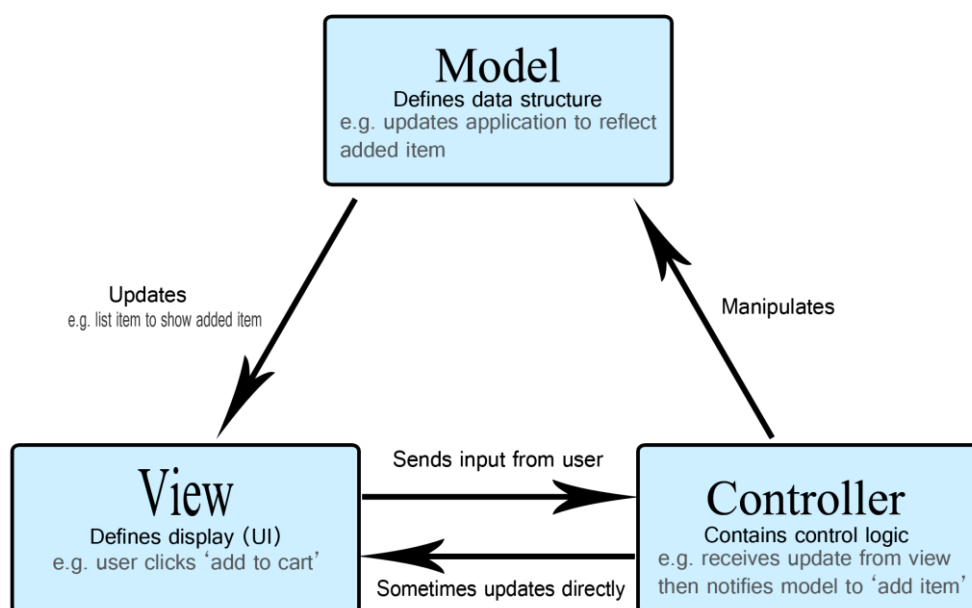


Рис. 2.4. Вигляд принципу MVC

В цілому, цей проект демонструє ефективне використання сучасних технологій для створення потужної та функціональної CRM-системи, орієнтованої на вирішення специфічних задач логістичних компаній.

2.4. Формування основних вимог

Платформа для оптимізації CRM-систем в логістиці повинна бути потужним інструментом, що охоплює всі ключові аспекти управління бізнесом. Вона має забезпечувати комплексне управління клієнтською базою, дозволяючи збирати, аналізувати та сегментувати інформацію про клієнтів для персоналізації взаємодії та підвищення лояльності.

Масштабованість є важливим аспектом розробки сучасних веб-застосунків, особливо для систем, орієнтованих на логістику, де обсяг даних і кількість користувачів можуть значно зростати з часом. Розроблена система має архітектуру, що забезпечує гнучкість у масштабуванні як горизонтальному, так і вертикальному.

Для горизонтального масштабування використовується підхід, при якому запити розподіляються між кількома машинами. Наприклад, сервер додатків може бути розділений на кілька вузлів, де кожен вузол обробляє частину запитів. Це дозволяє збільшити загальну продуктивність системи і забезпечує стійкість до навантажень. Для рівномірного розподілу вхідних запитів між кількома серверами застосовуються спеціальні програмні інструменти — балансувальники навантаження. Прикладами таких інструментів є NGINX та AWS Elastic Load Balancer.

Для забезпечення ефективної обробки зростаючого обсягу даних застосовується кластеризація бази даних. MySQL підтримує реплікацію даних, яка дозволяє створювати кілька реплік основної бази даних. Репліки можуть бути використані для розподілу навантаження на читання, тоді як основна база відповідає за операції запису. Це забезпечує швидкий доступ до даних і дозволяє уникнути перевантаження центральної бази. Наприклад, за великої кількості одночасних запитів на читання система може перенаправляти їх на репліки, що значно зменшує затримки.

Крім того, система підтримує кешування даних за допомогою Redis чи Memcached. Це особливо важливо для часто використовуваних даних, які не потребують постійного оновлення. Кешування знижує навантаження на базу даних і прискорює час відгуку системи.

Вертикальне масштабування забезпечується шляхом збільшення ресурсів окремих серверів, таких як процесорна потужність, оперативна пам'ять чи швидкість доступу до диска (рис. 2.5). Це може бути швидким рішенням у разі раптового зростання навантаження. Проте, на відміну від горизонтального масштабування, вертикальне має обмеження, оскільки ресурси апаратного забезпечення не безмежні.

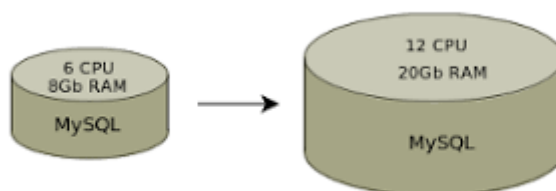


Рис. 2.5. Приклад вертикального масштабування

В REST API реалізовано механізм асинхронної обробки запитів, який гарантує стабільну та швидку роботу сервісу незалежно від кількості активних користувачів. Це забезпечує плавну роботу системи навіть під високим навантаженням.

Усі ці заходи дозволяють системі адаптуватися до зростаючих потреб бізнесу, забезпечуючи її продуктивність, надійність та можливість інтеграції нових функцій без збоїв у роботі.

Важливою складовою є ефективне управління замовленнями, що включає автоматизацію процесів обробки, відстеження статусів та контроль виконання. Платформа повинна також надавати інструменти для оптимізації логістичних операцій, таких як планування маршрутів, моніторинг перевезень та управління складськими запасами. Інтеграція з GPS-моніторингом дозволить відстежувати вантаж в режимі реального часу.

Для прийняття зважених управлінських рішень платформа повинна мати розвинену аналітику та звітність. Вона повинна збирати та аналізувати дані про різні аспекти діяльності компанії, генерувати звіти за різними критеріями та візуалізувати інформацію для легкого сприйняття.

Крім функціональних вимог, важливу роль відіграють також нефункціональні аспекти, такі як продуктивність, масштабованість, надійність, безпека та зручність використання. Окрім інтуїтивно зрозумілого інтерфейсу та простоти використання, платформа повинна підтримувати інтеграцію з іншими системами, наприклад, бухгалтерськими або ERP-системами, для забезпечення ефективного обміну даними. Враховуючи сучасні тенденції, важливо також забезпечити мобільний доступ до платформи та можливість розгортання в хмарі.

2.5. Визначення варіантів використання

Управління клієнтською базою в CRM-системі для логістики передбачає не просто зберігання контактних даних. Це потужний інструмент, який дозволяє менеджерам глибоко аналізувати потреби клієнтів, відстежувати історію взаємодії та сегментувати їх для персоналізованого обслуговування. Застосування цієї стратегії сприяє зміцненню довгострокових відносин з клієнтами та підвищенню результативності маркетингових заходів.

Ефективне управління замовленнями є невід'ємною частиною успішної логістичної діяльності. CRM-система дозволяє автоматизувати процеси обробки замовлень, відстежувати їх статус у будь-який момент часу та генерувати необхідну документацію. Це сприяє прозорості та оперативності роботи з замовленнями, а також мінімізує ризик помилок.

Оптимізація логістичних операцій досягається за рахунок інструментів планування маршрутів, моніторингу транспорту та управління складськими запасами. Інтеграція з GPS-моніторингом дозволяє відстежувати переміщення

вантажів безперервно, що підвищує безпеку та ефективність перевезень. Оптимізація завантаження транспорту допомагає мінімізувати витрати на паливо та амортизацію.

Управління складом в CRM-системі передбачає точний облік замовлень, контроль доставки її оптимізацію. Це дозволяє прискорити процеси прийому та відвантаження замовлень, мінімізувати витрати та уникнути затримок.

Аналітика та звітність є важливими інструментами для прийняття управлінських рішень. CRM-система дозволяє генерувати звіти про ефективність логістичних операцій, аналізувати дані про клієнтів та виявляти тенденції на ринку. Це допомагає компаніям покращувати свою діяльність, оптимізувати витрати та підвищувати конкурентоспроможність.

2.6. Висновки до розділу

Проведено аналіз технологій та методів оптимізації CRM-систем для логістики, зокрема REST API, Bootstrap, MVC, фреймворків Laravel, Vue.js та Vuex.

На основі аналізу обґрунтовано вибір оптимальних методів для реалізації CRM-систем для логістики, зокрема, таких як асинхронна обробка запитів, валідація схем та кешування, та на їх основі запропоновано алгоритм, що дозволить платформі ефективно обробляти великі обсяги даних.

Запропонований інтегрований підхід дозволяє не лише оптимізувати обробку даних та підвищити рівень безпеки, а й забезпечити позитивний користувацький досвід.

РОЗДІЛ 3

ПРАКТИЧНА РЕАЛІЗАЦІЯ CRM-СИСТЕМИ ДЛЯ ЛОГІСТИКИ

3.1. Реалізація серверної частини

Після аналізу методів для розробки API, проектування архітектури БД та дослідження сфери оптимізації логістичних операції можна розпочати створення класів серверної частини для забезпечення її роботи.

Серверна частина написана з допомогою мови PHP та фреймворку Laravel.

Запуск програми відбувається з допомогою файлу `server.php`, який використовується вбудованим веб-сервером PHP для емуляції функціональності «`mod_rewrite`» Apache. Це дозволяє запускати Laravel застосунок без необхідності встановлювати повноцінний веб-сервер, як Apache або Nginx (рис. 3.1).

```
1 <?php
2
3 /**
4  * Laravel - A PHP Framework For Web Artisans
5  *
6  * @package  Laravel
7  * @author   Taylor Otwell <taylor@laravel.com>
8  */
9
10 $uri = urldecode(
11     string: parse_url(url: $_SERVER['REQUEST_URI'], component: PHP_URL_PATH)
12 );
13
14 // This file allows us to emulate Apache's "mod_rewrite" functionality from the
15 // built-in PHP web server. This provides a convenient way to test a Laravel
16 // application without having installed a "real" web server software here.
17 if ($uri !== '/' && file_exists(__DIR__.'/public'.$uri)) {
18     return false;
19 }
20
21 require_once __DIR__.'/public/index.php';
22
```

Рис. 3.1. Функція запуску програми

Робота з розробки серверної частини починається зі створення API та відповідних контролерів для керування відповідним ресурсом. Першим створеним ресурсом є авторизація, він відповідає за автентифікацію користувача. Першочергово виконується функція `middleware`, яка встановлює стандартну роль користувача як «Гість», тобто надає доступ до веб-сайту без авторизації. Далі виконується функція

авторизації, яка створює токен, при успішному виконанні, який використовується для подальших запитів до API (рис. 3.2).

```
Route::post('/login', function (Request $request): mixed {
    $data = $request->validate([
        'email' => 'required|email',
        'password' => 'required'
    ]);

    $user = User::where('email', $request->email)->first();

    if (!$user || !Hash::check($request->password, $user->password)) {
        return response([
            'message' => ['These credentials do not match our records.']
        ], 404);
    }

    $token = $user->createToken('my-app-token')->plainTextToken;

    $response = [
        'user' => $user,
        'token' => $token
    ];

    return response($response, 201);
});
```

Рис. 3.2. Функція створення токена після авторизації

За реєстрацію нового користувача відповідає файл RegisterController.php (рис. 3.3)

```
39 public function __construct()
40 {
41     $this->middleware('guest');
42 }
43
44 /**
45  * Get a validator for an incoming registration request.
46  *
47  * @param array $data
48  * @return \Illuminate\Contracts\Validation\Validator
49  */
50 protected function validator(array $data): mixed
51 {
52     return Validator::make($data, [
53         'name' => ['required', 'string', 'max:255'],
54         'email' => ['required', 'string', 'email', 'max:255', 'unique:users'],
55         'password' => ['required', 'string', 'min:8', 'confirmed'],
56     ]);
57 }
58
59 /**
60  * Create a new user instance after a valid registration.
61  *
62  * @param array $data
63  * @return \App\User
64  */
65 protected function create(array $data): mixed
66 {
67     return User::create([
68         'name' => $data['name'],
69         'email' => $data['email'],
70         'password' => Hash::make($data['password']),
71     ]);
72 }
73 }
```

Рис. 3.3. Файл RegisterController.php

Функція `validator` перевіряє вхідні дані, які надсилаються з авторизацією користувача, такі як адресу електронної пошти та пароль. Перевіряє чи адреса пошти є унікальною, та чи можна створити користувача з такою адресою.

Функція `create` на основі перевірених даних створює нового користувача у базі даних.

Після авторизації відкривається головна сторінка сайту `dashboard`, яка доступна за відповідним ресурсом в API, керування яким у свою чергу відбувається за допомогою відповідної функції у файлі `customercontroller.php` (рис. 3.4).

```
public function dashboard(): mixed
{
    $overview = [
        ['Sun', Shipment::whereDate('created_at', Carbon::parse('last sunday')->startOfDay())->sum('charge_total')],
        ['Mon', Shipment::whereDate('created_at', Carbon::parse('last monday')->startOfDay())->sum('charge_total')],
        ['Tue', Shipment::whereDate('created_at', Carbon::parse('last tuesday')->startOfDay())->sum('charge_total')],
        ['Wed', Shipment::whereDate('created_at', Carbon::parse('last wednesday')->startOfDay())->sum('charge_total')],
        ['Thu', Shipment::whereDate('created_at', Carbon::parse('last thursday')->startOfDay())->sum('charge_total')],
        ['Fri', Shipment::whereDate('created_at', Carbon::parse('last friday')->startOfDay())->sum('charge_total')],
        ['Sat', Shipment::whereDate('created_at', Carbon::parse('last saturday')->startOfDay())->sum('charge_total')],
    ];

    $payments_overview = [
        ['Sun', Payment::whereDate('created_at', Carbon::parse('last sunday')->startOfDay())->sum('amount') + Shipment::whereDate('created_at', Carbon::parse('last sunday')->startOfDay())->sum('charge_total')],
        ['Mon', Payment::whereDate('created_at', Carbon::parse('last monday')->startOfDay())->sum('amount') + Shipment::whereDate('created_at', Carbon::parse('last monday')->startOfDay())->sum('charge_total')],
        ['Tue', Payment::whereDate('created_at', Carbon::parse('last tuesday')->startOfDay())->sum('amount') + Shipment::whereDate('created_at', Carbon::parse('last tuesday')->startOfDay())->sum('charge_total')],
        ['Wed', Payment::whereDate('created_at', Carbon::parse('last wednesday')->startOfDay())->sum('amount') + Shipment::whereDate('created_at', Carbon::parse('last wednesday')->startOfDay())->sum('charge_total')],
        ['Thu', Payment::whereDate('created_at', Carbon::parse('last thursday')->startOfDay())->sum('amount') + Shipment::whereDate('created_at', Carbon::parse('last thursday')->startOfDay())->sum('charge_total')],
        ['Fri', Payment::whereDate('created_at', Carbon::parse('last friday')->startOfDay())->sum('amount') + Shipment::whereDate('created_at', Carbon::parse('last friday')->startOfDay())->sum('charge_total')],
        ['Sat', Payment::whereDate('created_at', Carbon::parse('last saturday')->startOfDay())->sum('amount') + Shipment::whereDate('created_at', Carbon::parse('last saturday')->startOfDay())->sum('charge_total')],
    ];

    $pending = 0;
    $shipments = Shipment::all();
    foreach ($shipments as $key => $shipment) {
        if($shipment->status->status != 'Delivered'){
            $pending++;
        }
    }
}
```

Рис. 3.4. Вигляд функції `dashboard`

Функція збирає різні статистичні дані про відправлення, платежі, клієнтів, котирування та інші показники, а потім повертає їх у форматі JSON.

Загалом, ця функція збирає та обробляє дані для відображення на головній сторінці CRM-системи, надаючи користувачеві огляд ключових показників та статистики.

Наступним було створено ресурс Shipments, який відповідає за вантажі у веб-додатку (рис. 3.5).

```

72 references | 0 implementations
13  class Shipment extends Model
14  {
15      0 references
16      protected $guarded = [];
17
18      0 references | 0 overrides
19      public function package(): mixed{
20          return $this->hasMany(Package::class, 'shipment_id');
21      }
22
23      0 references | 0 overrides
24      public function insurance(): mixed{
25          return $this->hasMany(ShipmentInsurance::class, 'shipment_id');
26      }
27
28      0 references | 0 overrides
29      public function document(): mixed{
30          return $this->hasMany(Document::class, 'shipment_id');
31      }
32
33      0 references | 0 overrides
34      public function sender(): mixed{
35          return $this->hasOne(User::class, 'id', 'sender_id');
36      }
37
38      0 references | 0 overrides
39      public function receiver(): mixed{
40          return $this->hasOne(User::class, 'id', 'receiver_id');
41      }
42
43      0 references | 0 overrides
44      public function payment(): mixed{
45          return $this->hasMany(Payment::class, 'shipment_id');
46      }
47
48      0 references | 0 overrides
49      public function status(): mixed{
50          return $this->hasOne(ShipmentStatus::class, 'shipment_id')->latest();
51      }
52
53      0 references | 0 overrides
54      public function track(): mixed{
55          return $this->hasMany(ShipmentStatus::class, 'shipment_id');
56      }
57
58      0 references | 0 overrides
59      public function vendor_details(): mixed{
60          return $this->hasOne(ShipmentVendorDetail::class, 'shipment_id');
61      }

```

Рис. 3.5. Ресурс Shipments

За керування ресурсом Shipments відповідає контролер ShipmentController.php (рис. 3.6). Він створює нові вантажі, валідує дані кожного з них, дає змогу видаляти та редагувати дані, надає можливості відстеження, формує накладну та її pdf-версію.

```

28 class ShipmentController extends Controller
29 {
30     /**
31      * Display a Listing of the resource.
32      *
33      * @return \Illuminate\Http\Response
34      */
35     public function index(): mixed
36     {
37         $shipments = Shipment::latest()->get();
38         return ShipmentResource::collection($shipments);
39     }
40
41     public function undelivered_shipments(): mixed
42     {
43         // $shipments = Shipment::all();
44         // return $shipments[0]->status->status;
45         // $shipments = Shipment::with(['status' => function ($query) {
46         //     $query->where('status', '<', 'Delivexcred');
47         // }])->get();
48         $shipments = Shipment::where('delivery_date', '<', Carbon::now()->toDateString()->whereHas('status', function($q): void {
49             $q->where('status', '!=', 'Delivered');
50         }->get());
51
52         foreach ($shipments as $key => $shipment) {
53             # code...
54             if($shipment->status->status == 'Delivered'){
55                 unset($shipments[$key]);
56             }
57         }
58         return UndeliveredResource::collection($shipments);
59     }

```

Рис. 3.6. Клас ShipmentController

```

public function store(Request $request): mixed
{
    $request->validate([
        "delivery_address" => "max:255",
        "receiver_id" => "required",
        "package_contact_person" => "max:255",
        "package_contact_person_phone" => "max:255",
        "package_transaction_type" => "max:255",
        "package_pickup_address" => "max:500",
        "transport_company_name" => "max:255",
        "transport_company_phone" => "max:255",
        "transport_driver_name" => "max:255",
        "transport_driver_phone" => "max:255",
        "transport_driver_vehicle" => "max:255",
        "user_notes" => "max:500",
        "freight_invoice_number" => "max:500",
        "charge_transportation" => "max:255",
        "charge_handling" => "numeric",
        "charge_halt" => "numeric",
        "charge_insurance" => "numeric",
        "charge_odc" => "numeric",
        "charge_tax_percent" => "numeric",
        "charge_tax_amount" => "numeric",
        "charge_total" => "numeric",
        "sender_id" => "required",
        "remarks" => "max:500",
        "bill_to" => "required",
        "vendor_id" => "required|numeric",
        "vendor_total" => "required|numeric",
        "vendor_cash" => "numeric",
        "vendor_advance" => "required|numeric",
        "vendor_commission" => "required|numeric",
        "delivery_date" => "required|date",
    ]);
}

```

Рис. 3.7. Функція Store

Однією з основних функцій у ShipmentController є store, яка відповідає за створення нового запису про відправлення (shipment) в базі даних.

Загалом, вона реалізує логіку створення нового запису про відправлення в CRM-системі, включаючи валідацію даних, обробку файлів, збереження пов'язаних даних та генерацію подій (рис. 3.7).

З основних функцій системи також можна виділити quotations, customers, package та vendor.

Quotations – функція, яка відповідає за обробку запитів, пов'язаних з котируваннями (quotes). Вона містить набір методів, які реалізують CRUD (Create, Read, Update, Delete) операції над котируваннями, а також деякі додаткові функції (рис. 3.8).

```

12 class QuoteController extends Controller
13 {
14     /**
15      * Display a listing of the resource.
16      *
17      * @return \Illuminate\Http\Response
18      */
19     public function index(): mixed
20     {
21         $quotes = Quote::all();
22         return QuoteResource::collection($quotes);
23     }
24
25     /**
26      * Show the form for creating a new resource.
27      *
28      * @return \Illuminate\Http\Response
29      */
30     public function create(): void
31     {
32         //
33     }
34
35     /**
36      * Store a newly created resource in storage.
37      *
38      * @param \Illuminate\Http\Request $request
39      * @return \Illuminate\Http\Response
40      */

```

Рис. 3.8. Клас QuoteController

Загалом, цей контролер забезпечує повний цикл управління котируваннями системі, включаючи створення, перегляд, оновлення, видалення, відправлення email та зміну статусів.

Клас CustomerController (рис. 3.9) відповідає за взаємодію з даними про клієнтів в системі. Він надає інструменти для роботи з клієнтами, включаючи їх

пошук, перегляд, створення, редагування та видалення. Крім того, контролер дозволяє отримувати різноманітну статистику про клієнтів, таку як кількість відправлень, статус рахунків та котирувань.

```

0 references | 0 implementations
19 class CustomerController extends Controller
20 {
21     /**
22      * Display a listing of the resource.
23      *
24      * @return \Illuminate\Http\Response
25      */
26     public function index(Request $request): mixed
27     {
28         // $customers = User::where('role', 'customer')->orderBy('name')->get();
29         // return CustomerResource::collection($customers);
30
31
32         $search_query = $request->searchTerm;
33         $customers = User::where('role', 'customer')->where( 'name', 'LIKE', '%' . $search_query . '%'
34             ->get();
35         if ( $search_query ) {
36             $users['searchTerm'] = $search_query ? : '';
37         } else {
38             $users['searchTerm'] = $search_query ? null : '';
39         }
40
41         return CustomerResource::collection($customers);
42     }

```

Рис. 3.9. Вигляд класу CustomerController

CustomerController також містить функції для отримання інформації про рахунки та котирування клієнта. Функції `get_customer_invoices`, `get_customer_paid_invoices`, `get_customer_partial_invoices` та `get_customer_pending_invoices` дозволяють отримати список рахунків з різними статусами оплати. Функція `get_customer_quotes` повертає інформацію про котирування, створені для клієнта.

В цілому, CustomerController забезпечує комплексне управління даними про клієнтів в CRM-системі, надаючи зручний інтерфейс для роботи з ними та отримання необхідної інформації.

PackageController призначений для обробки запитів, пов'язаних з пакетами в CRM-системі.

VendorController призначений для управління інформацією про постачальників в системі (рис. 3.10). Він надає функціональність для створення, перегляду, оновлення

та видалення даних про постачальників, а також для відстеження їх зв'язків з відправленнями.

```

12 class VendorController extends Controller
13 {
14     /**
15      * Display a listing of the resource.
16      *
17      * @return \Illuminate\Http\Response
18      */
19     public function index(): void
20     {
21     }
22
23     0 references | 0 overrides
24     public function all_vendors(): mixed
25     {
26         $vendors = User::where('role', 'vendor')->get();
27         return CustomerResource::collection($vendors);
28     }
29
30     /**
31      * Show the form for creating a new resource.
32      *
33      * @return \Illuminate\Http\Response
34      */
35     public function create(): void
36     {
37         //
38     }
39
40     /**
41      * Store a newly created resource in storage.
42      *
43      * @param \Illuminate\Http\Request $request
44      * @return \Illuminate\Http\Response
45      */

```

Рис. 3.10. Клас VendorController

3.2. Клієнтська частина

Графічний інтерфейс представлено веб-застосунком на основі javascript з використанням фреймворку Vue.

Vue.js — це вільно поширюваний JavaScript фреймворк, який використовується для розробки інтерактивних користувацьких інтерфейсів та односторінкових веб-застосунків. Його головні переваги — це простота вивчення, гнучкість та зручність у

використанні, що робить його популярним вибором серед веб-розробників різного рівня досвіду (рис. 3.13).

```

1 <template>
2 <form class="text-left" @submit.prevent="login">
3   <b-alert
4     :show="dismissCountDown"
5     dismissible
6     variant="danger"
7     @dismissed="dismissCountDown = 0"
8     @dismiss-count-down="countDownChanged"
9   >Invalid Credentials</b-alert>
10
11
12 <div class="form">
13   <div id="username-field" class="field-wrapper input">
14     <svg
15       xmlns="http://www.w3.org/2000/svg"
16       width="24"
17       height="24"
18       viewBox="0 0 24 24"
19       fill="none"
20       stroke="currentColor"
21       stroke-width="2"
22       stroke-linecap="round"
23       stroke-linejoin="round"
24       class="feather feather-user"
25     >
26       <path d="M20 21v-2a4 4 0 0 0-4-4H8a4 4 0 0 0-4 4v2" />
27       <circle cx="12" cy="7" r="4" />
28     </svg>
29     <input
30       id="email"
31       type="email"
32       class="form-control"
33       v-model="email"
34       required
35       autocomplete="email"
36       autofocus
37     />
38   </div>

```

Рис. 3.13. Зовнішній вигляд коду на основі Vue

На рисунку представлено компонент відображення сторінки входу у систему.

Основна концепція Vue.js - це компонентний підхід. Застосунок будується з невеликих, незалежних компонентів, які можна комбінувати та повторно використовувати, створюючи складні інтерфейси. Vue.js використовує шаблонний синтаксис на основі HTML, що дозволяє легко інтегрувати його в проект.

Завдяки вбудованій реактивній системі Vue.js, інтерфейс автоматично реагує на зміни даних, що значно полегшує створення динамічних елементів та мінімізує обсяг

коду, необхідного для обробки дій користувача. Фреймворк також надає широкий набір директив, які розширюють можливості HTML та дозволяють створювати складні ефекти та анімації.

3.3. Перевірка роботи системи

Після завершення роботи над системою потрібно провести її тестування. Перше, що бачить користувач при запуску системи – вікно авторизації (рис. 3.14).

Log In to Gurukul

New Here? [Create an account](#)



Show Password ☐

Log In

☐ Keep me logged in

[Forgot Password?](#)

[Track Shipment](#)

© 2020 All Rights Reserved. Gurukul

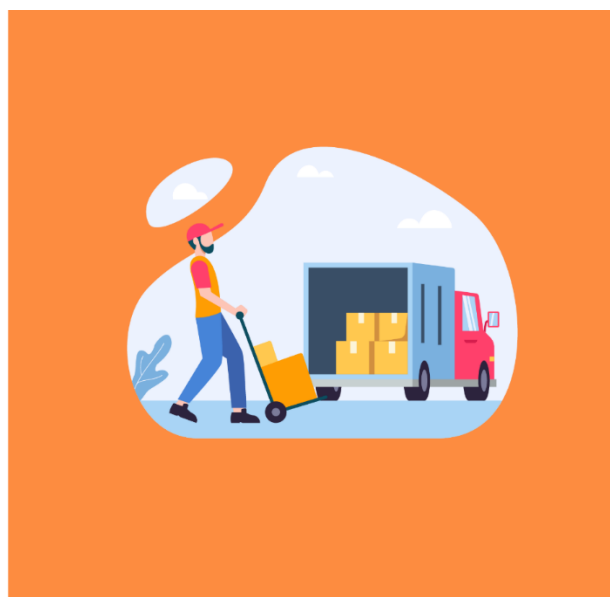


Рис. 3.14. Вигляд вікна авторизації

Після авторизації користувач потрапляє до вікна навігації системи. (рис. 3.15).

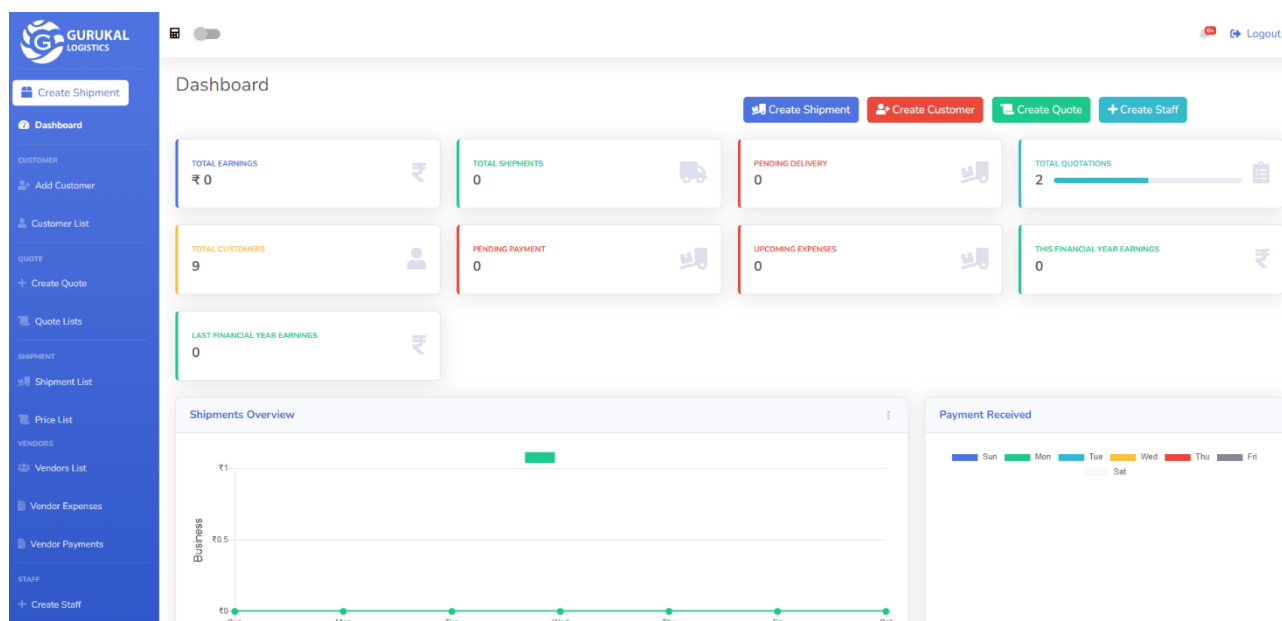


Рис. 3.15. Сторінка навігації системи

В залежності від авторизованої ролі, користувач має різні можливості використання застосунку. Клієнт має змогу створювати запити для замовлення, надаючи відповідну інформацію логістичній компанії (рис. 3.16).

Add Quotation Details

From

To

Denver

Dallas

Description

2019 Chevrolet Suburban

Size

Weight

224"Lt

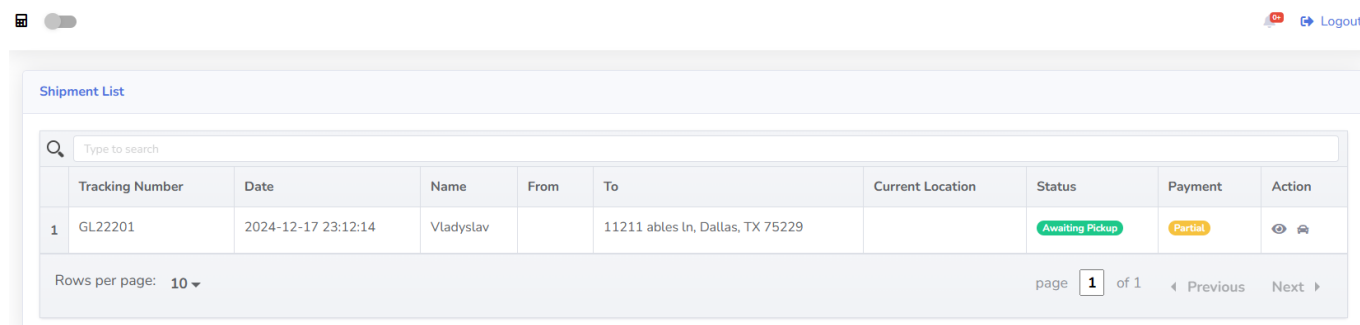
4568lb

Close

Add

Рис. 3.16. Вікно створення замовлення клієнтом

Після створення замовлення, воно з'являється у списку (рис. 3.17) і йому надається відповідний ідентифікатор (номер).

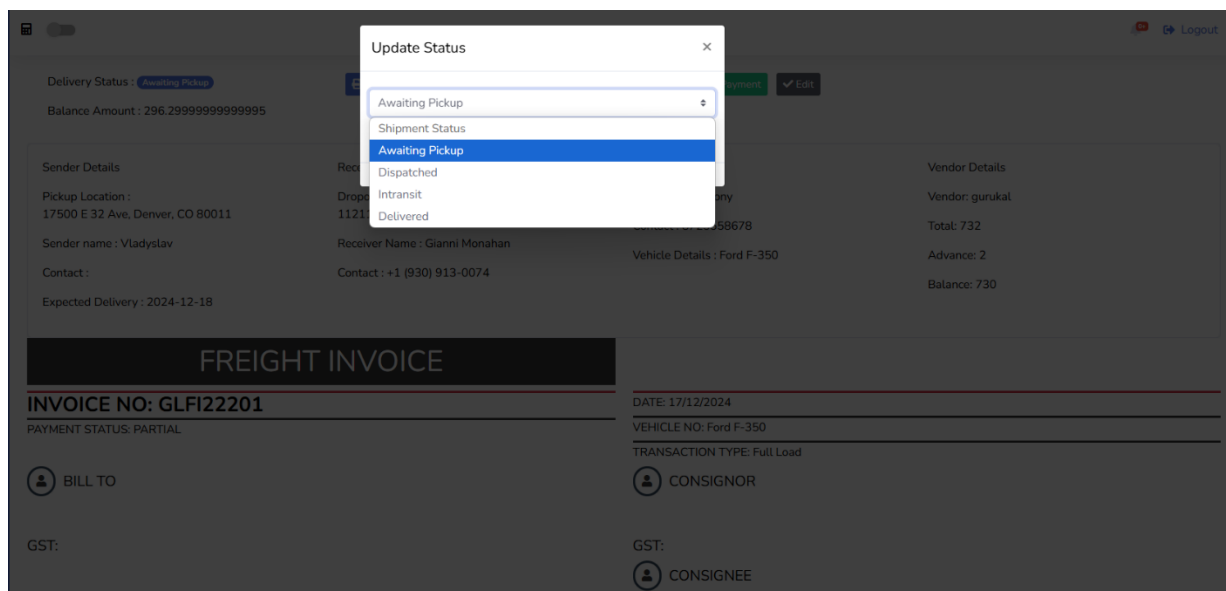


The screenshot shows a web application interface with a 'Shipment List' section. At the top right, there is a 'Logout' button. Below the title, there is a search bar with the placeholder text 'Type to search'. The main part of the interface is a table with the following columns: Tracking Number, Date, Name, From, To, Current Location, Status, Payment, and Action. There is one row of data with the following values: Tracking Number: GL22201, Date: 2024-12-17 23:12:14, Name: Vladyslav, From: (empty), To: 11211 ables ln, Dallas, TX 75229, Current Location: (empty), Status: Awaiting Pickup (in a green box), Payment: Partial (in an orange box), and Action: (icons for view and edit). Below the table, there is a pagination bar showing 'Rows per page: 10' and 'page 1 of 1' with 'Previous' and 'Next' navigation links.

	Tracking Number	Date	Name	From	To	Current Location	Status	Payment	Action
1	GL22201	2024-12-17 23:12:14	Vladyslav		11211 ables ln, Dallas, TX 75229		Awaiting Pickup	Partial	

Рис. 3.17. Створене замовлення у списку

У свою чергу персонал компанії має змогу керувати замовленням, а саме: змінювати його статус (рис. 3.18), створювати накладні (рис. 3.19), відправляти накладні у pdf-форматі, за адресою електронної пошти або смс (рис. 3.20), додавати дані про оплату.



The screenshot shows a 'FREIGHT INVOICE' form with various fields for sender and receiver details, pickup location, contact information, and vehicle details. A 'Delivery Status' dropdown is set to 'Awaiting Pickup'. A 'Balance Amount' of 296.29999999999995 is displayed. An 'Update Status' dialog box is open in the center, showing a list of status options: Awaiting Pickup (selected), Shipment Status, Awaiting Pickup, Dispatched, Intransit, and Delivered. The background form also includes a 'Vendor Details' section with 'Vendor: gurukul', 'Total: 732', 'Advance: 2', and 'Balance: 730'. At the bottom, there are sections for 'BILL TO' and 'CONSIGNOR/CONSIGNEE' with 'GST' fields.

Рис. 3.18. Зміна статусу замовлення

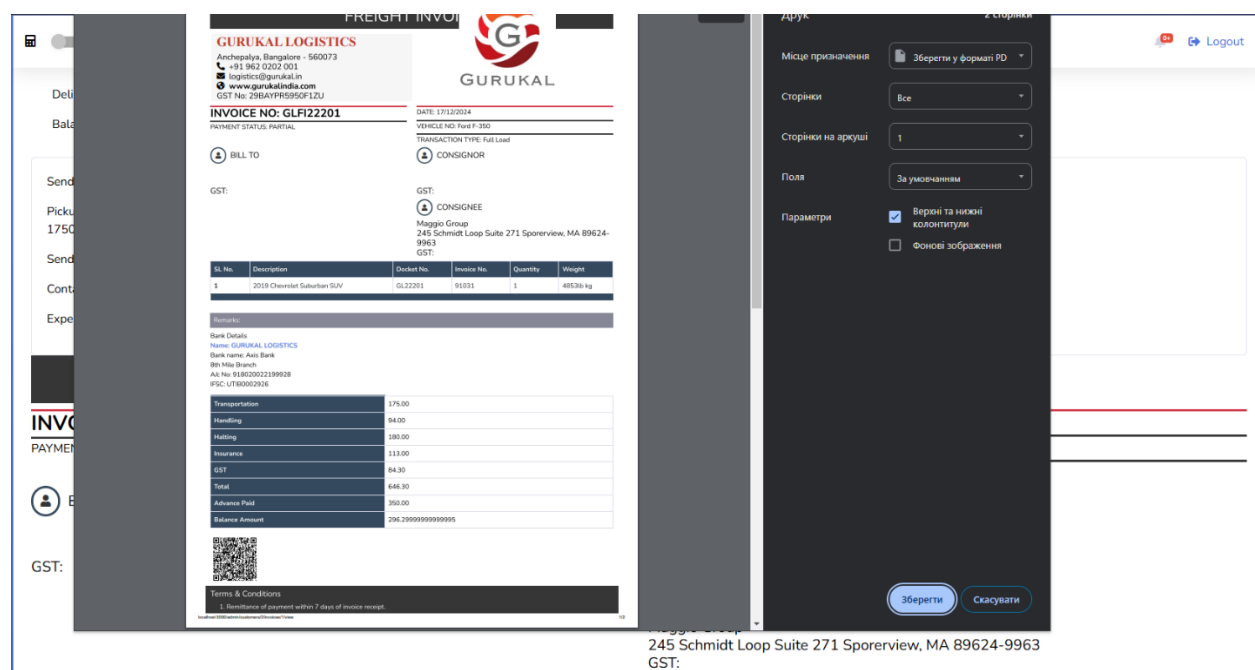


Рис. 3.19. Формування накладної для замовлення

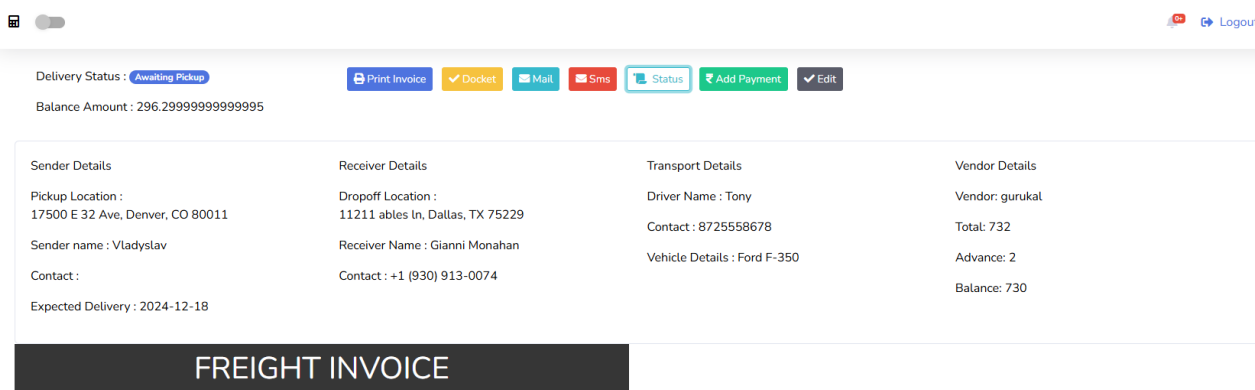


Рис. 3.20. Різні способи відправки накладної

У свою чергу адміністратор системи має змогу додавати та редагувати користувачів (рис 3.21, 3.22).

The screenshot shows the 'Create' form in the GURUKAL LOGISTICS application. The form is a modal window with a close button (X) in the top right corner. It contains the following fields and sections:

- Name:** A text input field with the placeholder 'name'.
- Email:** A text input field with the placeholder 'email'.
- Type:** A dropdown menu.
- Forecast close:** A dropdown menu.
- Contact:** A text input field with the placeholder 'contact'.
- Source:** A text input field with the placeholder 'source'.
- region:** A text input field with the placeholder 'region'.
- country:** A text input field with the placeholder 'India'.
- Potential Opportunity:** A text input field with the placeholder '0'.
- Chance of Sale:** A text input field with the placeholder '0'.
- Weighted forecast:** A text input field with the placeholder '0'.
- Query Form:** A section containing:
 - Name (Company/Individual):** A text input field.
 - Pickup Pincode:** A text input field.
 - Delivery Pincode:** A text input field.
 - Consignment description:** A text input field.

The background shows a sidebar with navigation options like 'Create Shipment', 'Dashboard', 'CUSTOMER', 'QUOTE', 'SHIPMENT', 'VENDOR', and 'STAFF'. The main content area is titled 'Leads' and has buttons for 'Add Lead' and 'Export'.

Рис. 3.21. Додавання нового користувача адміністратором

The screenshot shows the user profile page for Emma Schmitt. The page is divided into two main sections:

- Profile Card (Left):**
 - Avatar:** A placeholder icon for a user profile.
 - Name:** Emma Schmitt, with a green 'Active' status tag.
 - Contact Info:**
 - E-mail:** kromaguera@example.com
 - Phone:** 524-568-4792
 - Address:** 5541 Polly Villages Terrenceborough, UT 04425-1109
 - Registration Date:** 2024-12-17
- User Details (Right):**
 - Actions:** Buttons for 'Edit', 'Invoices', and 'Quotes'.
 - Fields:**
 - Email:** kromaguera@example.com
 - Phone:** 524-568-4792
 - Name:** Emma Schmitt
 - Company:** Smitham Group
 - GST No:** (Empty field)
 - Address:** 5541 Polly Villages Terrenceborough, UT 04425-1109
 - User Notes:** A text area with the placeholder 'User Notes - For internal use only.'
 - Show Rates:** A toggle switch.
 - Buttons:** 'Update' (blue), 'Delete Customer' (red), and 'Return to the dashboard' (grey).

Рис. 3.22. Редагування існуючого користувача адміністратором

3.4. Висновки до розділу

Реалізовано основні компоненти, які були використані для роботи системи. Серверна частина побудована на основі PHP з фреймворком Laravel з акцентом на використання контролерів.

Розроблено графічний інтерфейс з використанням Vue.js, який забезпечує реактивний і компонентний підхід до розробки. Vue.js автоматично оновлює інтерфейс користувача при зміні даних, розбиває його на повторювані компоненти, які полегшують розробку та містить велику кількість шаблонів, завдяки яким легко створювати динамічні інтерфейси.

Завдяки Vuex запити то бази даних зберігаються в окремому файлі програмного коду, що дозволяє виконувати менше запитів до бази даних, а отже підвищити швидкодію системи.

Використання фреймворків Laravel, Vue.js та Vuex, дозволило здійснити оптимізацію CRM системи для логістичних процесів.

РОЗДІЛ 4

ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

4.1. Охорона праці

У кваліфікаційній роботі проводилося дослідження методів та засобів оптимізації CRM-систем для логістики. Виконання роботи проводилося з використанням ЕОМ тому при роботі є необхідним дотримання правил охорони праці та техніки безпеки.

При роботі з ЕОМ потрібно дотримуватись державних стандартів України. Робочі місця працівників, які обладнані комп'ютерами пристроями, повинні відповідати вимогам ДСТУ 8604:2015 «Дизайн і ергономіка. Робоче місце для виконання робіт у положенні сидячи. Загальні ергономічні вимоги» [22].

Основні положення ДСТУ 8604:2015:

- Конструкція робочого місця та взаємне розташування всіх його елементів (сидіння, органів керування, засобів відображення інформації тощо) повинні відповідати антропометричним, фізіологічним і психологічним вимогам, а також характеру виконуваної роботи.

- Конструкцією робочого місця повинно бути забезпечено виконання трудових операцій у межах зони досяжності моторного поля.

- Висоти сидіння та підставки для ніг (у разі нерегульованої висоти робочої поверхні). У цьому разі висоту робочої поверхні встановлюють за номограмою для працюючих зростом 1800 мм. Оптимальна робоча поза для менших за зростом працюючих досягається збільшенням висоти робочого сидіння й підставки для ніг на величину, що дорівнює різниці між висотою робочої поверхні для працюючого зростом 1800 мм і висотою робочої поверхні, оптимальної для зросту даного працюючого.

- Підставка для ніг повинна бути регульованою по висоті. Її ширина повинна бути не менше ніж 300 мм, довжина - не менше ніж 400 мм. Поверхня 70 підставки

повинна бути рифленою. По передньому краю доцільно передбачати бортик висотою 10 мм.

Також на робочих місцях працівників які взаємодіють з екранними пристроями слід забезпечити дотримання вимог, які визначені в НПАОП 0.00-7.15- 18 «Вимог щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями» [8]. Згідно з цим наказом освітлення робочого місця працівника з екранними пристроями має створювати відповідний контраст між екраном і навколишнім середовищем (з урахуванням виду роботи) та відповідати вимогам ДСанПІН 3.3.2.007-98 [10].

Основні положення ДСанПІН 3.3.2.007-98:

- Площа на одне робоче місце має становити не менше ніж $6,0 \text{ м}^2$, а об'єм не менше ніж $20,0 \text{ м}^3$.

- Приміщення для роботи повинні мати природне та штучне освітлення.

- Природне освітлення має здійснюватись через світлові прорізи, орієнтовані переважно на північ чи північний схід і забезпечувати коефіцієнт природною освітленості (КПО) не нижче ніж 1,5%.

- Для внутрішнього оздоблення приміщень слід використовувати дифузно-відбивні матеріали з коефіцієнтами відбиття для стелі 0,7-0,8, для стін 0,5- 0,6.

- Яскравість світильників загального освітлення в зоні кутів випромінювання від 50 до 90 градусів з вертикаллю в повздовжній та поперечній площинах має становити не більше ніж 200 кд/м^2 , захисний кут світильників - не менше ніж 40 градусів

- Показник засліплення у разі використання джерел загального штучного освітлення у виробничих приміщеннях має не перевищувати 20.

- Необхідно обмежувати нерівномірність розподілу яскравості в полі зору працюючих з екранами. При цьому співвідношення яскравості робочих поверхонь має бути не більшим ніж 3:1, а співвідношення яскравості робочих поверхонь та поверхонь стін, обладнання тощо - 5:1.

Таким чином згідно з вимогами щодо охорони праці для працівників які працюють з ЕОМ приміщення, яке використовувалося для виконання кваліфікаційної

роботи магістра було забезпечене природнім і штучним освітленням. Також для роботи з ЕОМ було обрано місце, щоб в поле зору не потрапляли вікна або освітлювальні прилади. Шкідливим є також розміщення вікон за спинами працівників, оскільки на моніторі з'являється відбиття світла. Для уникнення проблем зі засліпленням природнім світлом приміщення було обладнано з використанням жалюзі з можливістю регулювання світлових потоків, щоб захистити робоче місце від попадання прямих сонячних променів. Адже вікна приміщення орієнтовані на південний схід.

Приміщення також було забезпечене штучним освітленням у вигляді комбінованих систем освітлення з використанням люмінесцентних ламп які розташовуються над робочими поверхнями у рівномірному порядку, щоб запобігти освітленню екрану ЕОМ прямими світловими потоками.

На всі робочих місцях для працівників які взаємодіють з ЕОМ забезпечена рівномірна освітленість за допомогою відбитого або розсіяного світла. Світлових відблисків від будь-яких частин робочої поверхні або ЕОМ у напрямку очей працівника немає.

Таким чином кваліфікаційна робота магістра виконувалась з урахуванням ДСТУ 8604:2015 та відповідала вимогам ДСанПІН 3.3.2.007-98, що дозволило працювати в умовах які відповідають правилам охорони праці та техніки безпеки. Робоче місце було обладнано всіма необхідними для комфортної роботи з ЕОМ засобами. Приміщення було забезпечене рівномірним та правильним освітленням, робочі місця розміщені таким чином, щоб природнє освітлення додавало природності робочому середовищу та зменшувало втомленість очей. Штучне освітлення було розподілене рівномірно і мало достатній рівень яскравості для зручного використання комп'ютера та читання документів. Такий підхід сприяє не лише ефективній роботі, але й збереженню здоров'я та підвищенню загального рівня задоволення від роботи з ЕОМ

4.2. Створення комфортних умов праці на логістичному підприємстві

Створення комфортних умов праці на логістичному підприємстві є критичним чинником, який впливає на продуктивність, мотивацію та загальний добробут співробітників. Належні умови праці сприяють зменшенню стресу, підвищенню ефективності виконання обов'язків та формуванню позитивного психологічного клімату в колективі. Досягнення цієї мети вимагає системного підходу, що включає врахування фізичних, технічних, психологічних та соціальних аспектів робочого середовища.

Організація робочого простору має базуватися на принципах ергономіки, які передбачають пристосування робочих місць до фізіологічних особливостей працівників. Меблі, обладнання та інструменти повинні забезпечувати комфортне положення тіла під час роботи, мінімізуючи навантаження на хребет і м'язи. Важливим є також забезпечення належного рівня освітлення, вентиляції та підтримання оптимальної температури в приміщенні, оскільки ці фактори безпосередньо впливають на фізичний і психологічний стан працівників.

Логістичні підприємства повинні забезпечувати своїх співробітників сучасним технічним обладнанням, яке відповідає специфіці виконуваних завдань. Комп'ютери з необхідним програмним забезпеченням, швидкісний інтернет, принтери, сканери, телефони та інше обладнання допомагають оптимізувати робочі процеси. Регулярне технічне обслуговування і своєчасне оновлення обладнання дозволяє запобігти затримкам у роботі та підвищує продуктивність.

Позитивний психологічний клімат є важливим фактором, що сприяє створенню комфортного робочого середовища. Взаєморозуміння, ефективна комунікація та командна співпраця допомагають зменшити стрес і підвищити мотивацію працівників. Для підтримки здорової атмосфери на підприємстві доцільно проводити регулярні командні заходи, створювати умови для обговорення робочих питань і вирішення конфліктів. Додатково керівництву слід заохочувати співпрацю між підрозділами, що сприяє розвитку командного духу.

Дотримання соціальних гарантій є невід'ємною складовою комфортних умов праці. Своєчасна виплата заробітної плати, надання відпусток, забезпечення медичного страхування та інших соціальних пільг підвищує лояльність співробітників і знижує плинність кадрів. Умови праці повинні відповідати чинному законодавству та міжнародним стандартам, таким як Директива 90/270/ЄЕС від 29 травня 1990 року, яка регулює безпеку і здоров'я при роботі з екранними пристроями.

Підтримка професійного розвитку працівників є важливим мотиваційним фактором. Організація тренінгів, курсів підвищення кваліфікації та участь у галузевих конференціях дозволяють співробітникам здобувати нові знання і навички. Такі заходи не тільки сприяють особистісному зростанню, але й підвищують конкурентоспроможність підприємства на ринку.

Комфортні умови праці повинні відповідати встановленим стандартам безпеки. В Україні основні вимоги щодо безпеки і захисту здоров'я працівників при роботі з екранними пристроями затверджено наказом Міністерства соціальної політики від 14 лютого 2018 року № 207. Крім того, державні санітарні правила і норми роботи з візуальними дисплейними терміналами, розроблені Міністерством охорони здоров'я України у 1998 році, спрямовані на збереження здоров'я працівників і мінімізацію ризиків, пов'язаних з використанням комп'ютерної техніки.

Впровадження комплексних заходів щодо створення комфортних умов праці на логістичному підприємстві позитивно впливає на продуктивність, задоволеність працівників і стабільність колективу. Поєднання сучасного технічного оснащення, дотримання ергономічних стандартів, забезпечення соціальних гарантій і підтримка професійного розвитку формують робоче середовище, яке мотивує співробітників і сприяє успішному функціонуванню підприємства.

4.3. Висновки до розділу

В цьому розділі проаналізовано важливі питання охорони праці та безпеки при створенні комфортних умов праці на логістичному підприємстві.

ВИСНОВКИ

В результаті виконання роботи запропоновано методи та засоби підвищення ефективності логістичних процесів завдяки оптимізації CRM-систем. А саме:

- 1) Проведений аналіз предметної області та сучасних CRM-систем у логістиці засвідчив важливість автоматизації бізнес-процесів у транспортній галузі;
- 2) Проаналізовано технології та методи оптимізації CRM-систем для логістики, зокрема REST API, Bootstrap, MVC, фреймворків Laravel, Vue.js та Vuex з метою їх використання;
- 3) Обґрунтовано вибір оптимальних методів для реалізації CRM-систем для логістики, зокрема, таких як асинхронна обробка запитів, валідація схем та кешування, та на їх основі запропоновано алгоритм, що дозволить платформі ефективно обробляти великі обсяги даних;
- 4) Реалізовано основні компоненти, які були використані для роботи системи. Серверна частина побудована на основі PHP з фреймворком Laravel з акцентом на використання контролерів;
- 5) Розроблено графічний інтерфейс з використанням Vue.js, який забезпечує реактивний і компонентний підхід до розробки;
- 6) Використання фреймворків Laravel, Vue.js та Vuex, дозволило здійснити оптимізацію CRM системи для логістичних процесів.

Запропонований інтегрований підхід дозволяє не лише оптимізувати обробку даних та підвищити рівень безпеки, а й забезпечити позитивний користувацький досвід.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Луцик Н.С., Луцків А.М., Осухівська Г.М., Тиш Є.В. Програма та методичні рекомендації з проходження практики за тематикою кваліфікаційної роботи для студентів спеціальності 123 «Комп'ютерна інженерія» другого (магістерського) рівня вищої освіти усіх форм навчання. Тернопіль: ТНТУ. 2024. 45 с.
2. Луцик Н.С., Луцків А.М., Осухівська Г.М., Тиш Є.В. Методичні рекомендації до виконання кваліфікаційної роботи магістра для студентів спеціальності 123 «Комп'ютерна інженерія» другого (магістерського) рівня вищої освіти усіх форм навчання. Тернопіль. 2024. 44 с.
3. Варавін А.В., Лецишин Ю.З., Чайковський А.В. Методичні вказівки до виконання курсового проєкту з дисципліни «Дослідження і проєктування комп'ютерних систем та мереж» для здобувачів другого (магістерського) рівня вищої освіти спеціальності 123 «Комп'ютерна інженерія» усіх форм навчання. Тернопіль: ТНТУ, 2024. 32 с.
4. Бражніков В. А., Осухівська Г.М. Вплив персоналізації клієнтських взаємовідносин на ефективність CRM-систем у логістиці. Матеріали XII міжнародній науково-технічній конференція молодих учених та студентів «Інформаційні моделі, системи та технології». Тернопіль: 2024
5. Бражніков В. А., Осухівська Г.М. Актуальність засобів оптимізації CRM-систем для логістики. Матеріали XII міжнародній науково-технічній конференція молодих учених та студентів «Інформаційні моделі, системи та технології». Тернопіль: 2024
6. Стручок В.С. Техноекологія та цивільна безпека. Частина «Цивільна безпека». Навчальний посібник. Тернопіль: ТНТУ. 2022. 150 с.
7. Стручок В.С. Безпека в надзвичайних ситуаціях. Методичний посібник для здобувачів освітнього ступеня «магістр» всіх спеціальностей денної та заочної (дистанційної) форм навчання. Тернопіль: ТНТУ. 2022. 155 с.
8. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин. ДСанПІН 3.3.2.007-98. URL:

<https://zakon.rada.gov.ua/rada/show/v0007282-98#Text> (дата звернення: 11.12.2024 р.)

9. Кодекс цивільного захисту України. URL: <https://zakon.rada.gov.ua/laws/main/5403-17> (дата звернення: 11.12.2024 р.)

10. Наказ Мінсоцполітики від 14.02.2018, № 207. URL: <https://zakon.rada.gov.ua/laws/show/z0508-18#Text> (дата звернення: 11.12.2024 р.)

11. ДСТУ 8604:2015. URL: https://online.budstandart.com/ua/catalog/doc-page.html?id_doc=71028 (Дата звернення 10.12.2024)

12. Yatsyshyn V., Pastukh O., Palamar A., Zharovsky R. Technology of relational database management systems performance evaluation during computer systems design. Scientific Journal of TNTU, Ternopil, Ukraine, 2023. Vol. 109, No 1. P. 54–65.

13. Харченко О., Яцишин В. Розробка та керування вимогами до програмного забезпечення на основі моделі якості. Вісник ТДТУ. Тернопіль, 2009. Т. 14. №1. С. 201-207

14. Кобилух О., Гірна О. Сучасні підходи до логістичного обслуговування клієнтів на основі використання CRM-системи URL: <https://academy-vision.org/index.php/av/article/view/62/53> (Дата звернення 12.12.2024)

15. Мальнов Д.В. Особливості застосування інформаційних технологій в управлінні взаємовідносинами з клієнтами транспортних підприємств. URL: https://www.researchgate.net/profile/bugajkooleksandrovic/publication/380723570_risks_of_multimodal_transportation_under_the_conditions_of_martial_law/links/664b4206479366623afe3882/risksmultimodaltransportationundertheconditionsofmartiallaw.pdf?__cf_chl__tk=kjo8zvlpa9_3xr9w_mcha6cn1.43gwsnhztr2h72vxs17346908641.0.1.1iyi1gnvcmzg1uubri9aribulqs7o2eh_xtgwjksa6cg#page=126 (Дата звернення 03.12.2024)

16. Янчук Т.В., Боєнко О.Ю. Впровадження CRM-систем як засіб підвищення ефективності маркетингової діяльності. URL: <https://economyandsociety.in.ua/index.php/journal/article/view/2269> (Дата звернення 12.12.2024)

17. Роберт М. Чиста Архітектура. Фабула, 2019. 359 с.

18. Gupta N. R. Crack the Marketing Case and Interview Like a СМО. Bloomsbury Publishing India Pvt. Ltd., 2016. 176 p.

19. Szymonik A. Information Technologies in Logistics. 2012. 214 p.
20. Laravel for CRM. URL: <https://laravelcrm.com/docs/>.
21. MySQL. URL: <https://www.mysql.com/>.
22. Vue.js: The progressive javascript framework. URL: <https://vuejs.org/>.
23. Bootstrap · The most popular HTML, CSS, and JS library in the world.
URL: <https://getbootstrap.com/>.
24. Getting Started | Axios Docs. URL: <https://axios-http.com/docs/intro>.
25. Рентабельність CRM-системи: як використати її потенціал на повну.
URL: https://wezom.com.ua/ua/blog/rentabelnost-crm-sistemy?utm_source=chatgpt.com.
(Дата звернення 05.12.2024)
26. Forrester Research: "CRM Trends and Analytics for 2023".
URL: <https://www.forrester.com/report/customer-relationship-management-market-insights-2023/RES179653>. (Дата звернення 30.11.2024)
27. Designing Data-Intensive Web Applications. Elsevier, 2003.
URL: <https://doi.org/10.1016/b978-1-55860-843-6.x5000-2> (Дата звернення 29.11.2024)
28. Mastering Laravel. Packt Publishing, Limited, 2015.
29. Shaikh D. S. Logistics Management System using Big Data Analytics. Interantional journal of scientific research in engineering and management. 2023. Vol. 07, no. 12. P. 1–13. URL: <https://doi.org/10.55041/ijssrem27750> (Дата звернення 07.12.2024)

Додаток А.
Тези конференцій

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
імені ІВАНА ПУЛЮЯ
НАУКОВЕ ТОВАРИСТВО ім. ШЕВЧЕНКА

МАТЕРІАЛИ

ХІІ НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ

**«ІНФОРМАЦІЙНІ МОДЕЛІ,
СИСТЕМИ ТА ТЕХНОЛОГІЇ»**



18–19 грудня 2024 року

ТЕРНОПІЛЬ
2024

В. Бражніков

АКТУАЛЬНІСТЬ ЗАСОБІВ ОПТИМІЗАЦІЇ CRM-СИСТЕМ ДЛЯ ЛОГІСТИКИ

V. Brazhnikov

RELEVANCE OF CRM SYSTEM OPTIMIZATION TOOLS FOR LOGISTICS

117

227

В. Бражніков, Г. Осухівська

ВПЛИВ ПЕРСОНАЛІЗАЦІЇ КЛІЄНТСЬКИХ ВЗАЄМОВІДНОСИН НА ЕФЕКТИВНІСТЬ CRM-СИСТЕМ У ЛОГІСТИЦІ

V. Brazhnikov, H. Osukhivska

THE IMPACT OF CUSTOMER RELATIONSHIP PERSONALIZATION ON THE EFFICIENCY OF CRM SYSTEMS IN LOGISTICS

118

УДК 004.451:656.025.2

В. Бражніков

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

АКТУАЛЬНІСТЬ ЗАСОБІВ ОПТИМІЗАЦІЇ CRM-СИСТЕМ ДЛЯ ЛОГІСТИКИ

UDC 004.451:656.025.2

V. Brazhnikov

RELEVANCE OF CRM SYSTEM OPTIMIZATION TOOLS FOR LOGISTICS

Сучасна логістика – це складний механізм, де кожна деталь має значення. Зростаючі обсяги перевезень, різноманітність маршрутів та вимоги клієнтів до швидкості доставки ставлять перед логістичними компаніями значні виклики. У цьому контексті ефективне управління взаємовідносинами з клієнтами є одним з ключових факторів успіху. Саме тут на допомогу приходять CRM-системи, оскільки вони дозволяють автоматизувати, аналізувати і планувати всі процеси взаємодії з клієнтами [1, 2].

Однак з розвитком технологій і збільшенням обсягу даних традиційний підхід до використання CRM-системи виявляється недостатнім. Величезні обсяги інформації про клієнтів, замовлення, маршрути та транспортні засоби необхідно аналізувати швидко і точно. Крім того, сучасні логістичні компанії використовують різноманітні інформаційні системи, такі як ERP і WMS, які необхідно інтегрувати з CRM для створення єдиного інформаційного середовища [1, 2]. Все це ставить перед логістичними компаніями нові виклики і вимагає пошуку інноваційних рішень.

Одним з найбільш перспективних напрямків оптимізації CRM-систем є використання машинного навчання та штучного інтелекту. Ці технології дозволяють аналізувати великі обсяги даних, виявляти закономірності та прогнозувати майбутні тенденції. Наприклад, машинне навчання можна використовувати для оптимізації маршрутів доставки, прогнозування попиту на певні послуги та персоналізації пропозицій для клієнтів. Крім того, штучний інтелект може взяти на себе рутинні завдання, такі як обробка замовлень або відповіді на запити клієнтів, дозволяючи співробітникам зосередитися на більш складних завданнях.

Ще одним важливим аспектом оптимізації CRM-системи є інтеграція аналітичних інструментів. Інформаційні панелі та звіти дають змогу менеджерам отримувати детальну інформацію про результати діяльності компанії, виявляти вузькі місця та приймати обґрунтовані рішення. Візуалізація даних дозволяє швидко оцінити ситуацію і вжити необхідних заходів.

Таким чином, оптимізація CRM-систем в логістиці є необхідною умовою для підвищення ефективності бізнесу. Використовуючи новітні технології, такі як машинне навчання, штучний інтелект і аналітичні інструменти, логістичні компанії можуть значно поліпшити якість обслуговування, знизити витрати і підвищити задоволеність клієнтів. Інвестуючи в розвиток CRM-систем, компанії роблять крок до успіху в конкурентному середовищі.

Література

1. 5 benefits of CRM for Logistics Service Providers. URL: <https://www.workbooks.com/resources/blog/5-benefits-of-crm-for-logistics-service-providers/>.
2. CRM for logistics: importance, advantages and features. URL: <https://www.customerization.ca/best-crm-logistics-industry/>.

УДК 004.451:656.025.2

В. Бражніков; Г. Осухівська, канд. техн. наук, доц.

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

ВПЛИВ ПЕРСОНАЛІЗАЦІЇ КЛІЄНТСЬКИХ ВЗАЄМОВІДНОСИН НА ЕФЕКТИВНІСТЬ CRM-СИСТЕМ У ЛОГІСТИЦІ

UDC 004.451:656.025.2

V. Brazhnikov; H. Osukhivska, Ph. D., Assoc. Prof.

THE IMPACT OF CUSTOMER RELATIONSHIP PERSONALIZATION ON THE EFFICIENCY OF CRM SYSTEMS IN LOGISTICS

Сьогодні логістика все більше орієнтується на клієнта, пріоритезуючи індивідуальні підходи до кожного замовника. Такий підхід потребує персоналізованих рішень у взаємовідносинах з клієнтами, які допомагають підвищити якість обслуговування та задоволеність клієнтів. У цьому випадку CRM-системи є ключовим інструментом, який дозволяє ефективно керувати відносинами з клієнтом завдяки збору, аналізу та використання даних [1, 2].

Персоналізація клієнтських взаємовідносин у логістичних операціях стикається з низкою викликів. Важливим аспектом є необхідність обробки великої кількості даних про клієнтів, зокрема, їх вподобання, історію замовлень та особливості поведінки. Такі дані повинні бути доступними в режимі реального часу для забезпечення адаптації послуг до потреб замовників. Тому CRM-системи мають бути не тільки потужними, але й гнучкими та інтегрованими з іншими інструментами інтернету речей та автоматизованими системами підтримки для забезпечення високого рівня персоналізації.

У реалізації персоналізованого підходу вирішальну роль відіграють технології. Штучний інтелект, машинне навчання, аналітика великих обсягів даних і алгоритми кластеризації дозволяють розподілити клієнтів за їх характеристиками та прогнозувати майбутні потреби. Для прикладу це можуть бути індивідуальні рекомендації та пропозиції або попередження щодо доставки, які допомагають покращити досвід клієнта. До того ж, автоматизовані рішення, такі як чат-боти, забезпечують цілодобову підтримку, яка відповідає конкретним запитам клієнтів.

Персоналізація підходу приносить значні переваги як для клієнтів, так і для компанії. Клієнт отримує кращий досвід взаємодії, швидке вирішення проблем та відчуття індивідуального підходу. Перевагою для компанії є підвищення лояльності клієнтів та збільшення обсягів повторних замовлень. До того ж, персоналізація допомагає оптимізувати витрати, спрямовуючи ресурси на цільові сегменти аудиторії. Одночасно, це дозволяє зміцнити репутацію компанії, що важливо у конкурентному середовищі сьогодення.

Отже, персоналізація клієнтських відносин через CRM-системи є корисною не лише як інструмент підвищення ефективності, але й як стратегічна перевага. Вона дозволяє адаптовувати бізнес до мінливості ринку, дозволяючи забезпечувати високий рівень довіри та задоволення клієнтів. Сучасні технології дозволяють логістичним компаніям не просто реагувати на потреби клієнтів, але й передбачати їх, створюючи довгострокові та взаємовигідні відносини.

Література

1. Переваги використання CRM систем для ефективної роботи компаній. URL: <https://firtka.if.ua/blog/view/perevagi-vikoristannia-crm-sistem-dlia-efektivnoyi-roboti-kompanii>.
2. Впровадження CRM-системи: роль CRM-технологій у підвищенні ефективності бізнесу. URL: <https://tqm.com.ua/ua/likbez/crm-systemy/rol-vprovadzhennia-crm>.

Додаток Б.

Скрипт запуску

Лістинг коду 1 – Програмний код запуску системи

```
<?php
    $uri = urldecode(
        parse_url($_SERVER['REQUEST_URI'], PHP_URL_PATH)
    );

    if ($uri !== '/' && file_exists(__DIR__.'/public'.$uri)) {
        return false;
    }

    require_once __DIR__.'/public/index.php';
```

Лістинг коду 2 – Програмний код авторизації в систему

```
Route::middleware('auth:sanctum')->get('/user', function (Request
$request) {

    return $request->user();
});

Route::post('/login', function (Request $request) {
    $data = $request->validate([
        'email' => 'required|email',
        'password' => 'required'
    ]);

    $user = User::where('email', $request->email)->first();

    if (!$user || !Hash::check($request->password, $user->password))
    {
        return response([
            'message' => ['These credentials do not match our
records.'],
        ], 404);
    }

    $token = $user->createToken('my-app-token')->plainTextToken;

    $response = [
        'user' => $user,
        'token' => $token
    ];

    return response($response, 201);
});
```

Лістинг коду 2 – Програмний код функції відображення накладної

```

public function view_invoice($request) {
    $shipment = Shipment::find($request);
    $shipment->sender;
    $shipment->package;
    $shipment->receiver;
    $shipment->payment;

    $total_paid = $shipment->payment->sum('amount');
    if($total_paid > 0)
    {
        $balance = ($shipment->charge_total - $total_paid)
- $shipment->charge_advance_paid;
    }
    else {
        $balance = ($shipment->charge_total - $shipment-
>charge_advance_paid);
    }

    $shipment->qrcode =
"https://crm.gurukal.in/customer/docket/8jZSqbgNmzk25EcBgMsWYyDP4LDEAS
7amrVevmqcTE67ByuajGaks8UqmLmJ/$shipment-
>id/urMrnM6JNuGPCnEdnmDqzfWfDYAUSYb8rkveHF9mWGPgD2XxH4SYRXjRCnmX/view"
;

    // dd($balance);
    $balance_amount = new stdClass();
    $balance_amount->balance_amount = $balance;
    return view('invoice', compact('shipment', 'balance_amount'));
}

```

Лістинг коду 3 – Програмний код функції збереження накладної у PDF-форматі

```

public function invoicePDF($request){
    $shipment = Shipment::find($request);
    $shipment->sender;
    $shipment->package;
    $shipment->receiver;
    $shipment->payment;

    $total_paid = $shipment->payment->sum('amount');
    if($total_paid > 0)
    {
        $balance = ($shipment->charge_total - $total_paid) -
$shipment->charge_advance_paid;
    }
    else {
        $balance = ($shipment->charge_total - $shipment-
>charge_advance_paid);
    }

    $shipment->qrcode =

```

```

"https://crm.gurukal.in/customer/docket/8jZSqbgNmzk25EcBgMsWYyDP4LDEAS
7amrVevmqcTE67ByuajGaks8UqmLmJ/$shipment-
>id/urMrnM6JNuGPCnEdnmDqzfWfDYAUSYb8rkveHF9mWGPgD2XxH4SYRXjRCnmX/view"
;
    //    dd($balance);
    $balance_amount = new \stdClass();
    $balance_amount->balance_amount = $balance;

    return view('invoice_mobile', compact('shipment',
'balance_amount'));
}

```

Лістинг коду 4 – Програмний код функції створення замовлення

```

public function store(Request $request)
{
    $request->validate([
        'name'=>'required|max:255',
        'email'=>'required|email|unique:users|max:255',
        'password'=>'required|max:255',
        'gst'=>'max:255',
        'address'=>'max:500',
        'phone'=>'max:255',
        'company_name'=>'max:255',
    ]);

    $customer = User::create($request->all());
    if($request->show_rates) {
        $customer->show_rates = $request->show_rates;
    }else {
        $customer->show_rates = 0;
    }

    $customer->save();

    $beautymail = app() -
>make(\Snowfire\Beautymail\Beautymail::class);

    $beautymail->send('emails.customer.created',
compact('request'), function($message) use($request)
    {
        $message
            ->from('admin@gurukal.co.in', 'Gurukal Logistics')
            ->to($request->email, $request->name)
            // ->to("virtualarts123@gmail.com")
            // ->cc('gurukallogistics@gmail.com')
            ->subject('Account created');
    }
}

```

```
    });  
    return new CustomerResource($customer);  
}
```