

Прикарпатський національний університет імені Василя Стефаника

Фізико-технічний факультет

Кафедра комп'ютерної інженерії та електроніки

Підсадочий Володимир Петрович

Pidsadochy Volodymyr

УДК 004:42

Спеціальність 123 «Комп'ютерна інженерія»

Кваліфікаційна робота

на здобуття освітнього ступеня бакалавра

Автоматизоване робоче місце диспетчера вантажоперевезень

Automated workplace of the dispatcher of cargo transportation

Науковий керівник:

Доцент Дзундза Б.С.

Рецензент:

Д.ф.-м.н., проф. Салій Я.П.

Івано-Франківськ

2023

АНОТАЦІЯ

Сфера управління вантажними перевезеннями має велике значення в сучасних бізнес-процесах, пов'язаних з вантажними перевезеннями.

У роботі досліджено предметну область і встановлено основні вимоги до програмно-апаратного забезпечення автоматизованого робочого місця диспетчера вантажоперевезень. Визначено функціональні та нефункціональні вимоги та варіанти їх використання. Побудовано діаграми класів і компонентів. Ці діаграми допомагають у процесі розробки. Розглянуто роботу існуючих систем, їх переваги та недоліка.

У результаті проектування системи основні варіанти використання процесу управління вантажами для менеджера вантажів були описані та охоплені кодом програми.

У процесі розробки використовувалися сучасні технології проектування бази даних, серверних сервісів і веб-сторінок. Нові технології були застосовані для безпечного та надійного прототипу програми.

Змн.	Арк.	№ докум.	Підпис	Дата				
Розробив		Підсачочий В.П.			Анотація	Літ.	Арк.	Аркуші
Перевірив		Дзундза Б.С.					3	1
Н. Контр.								
Затвердив								

ABSTRACT

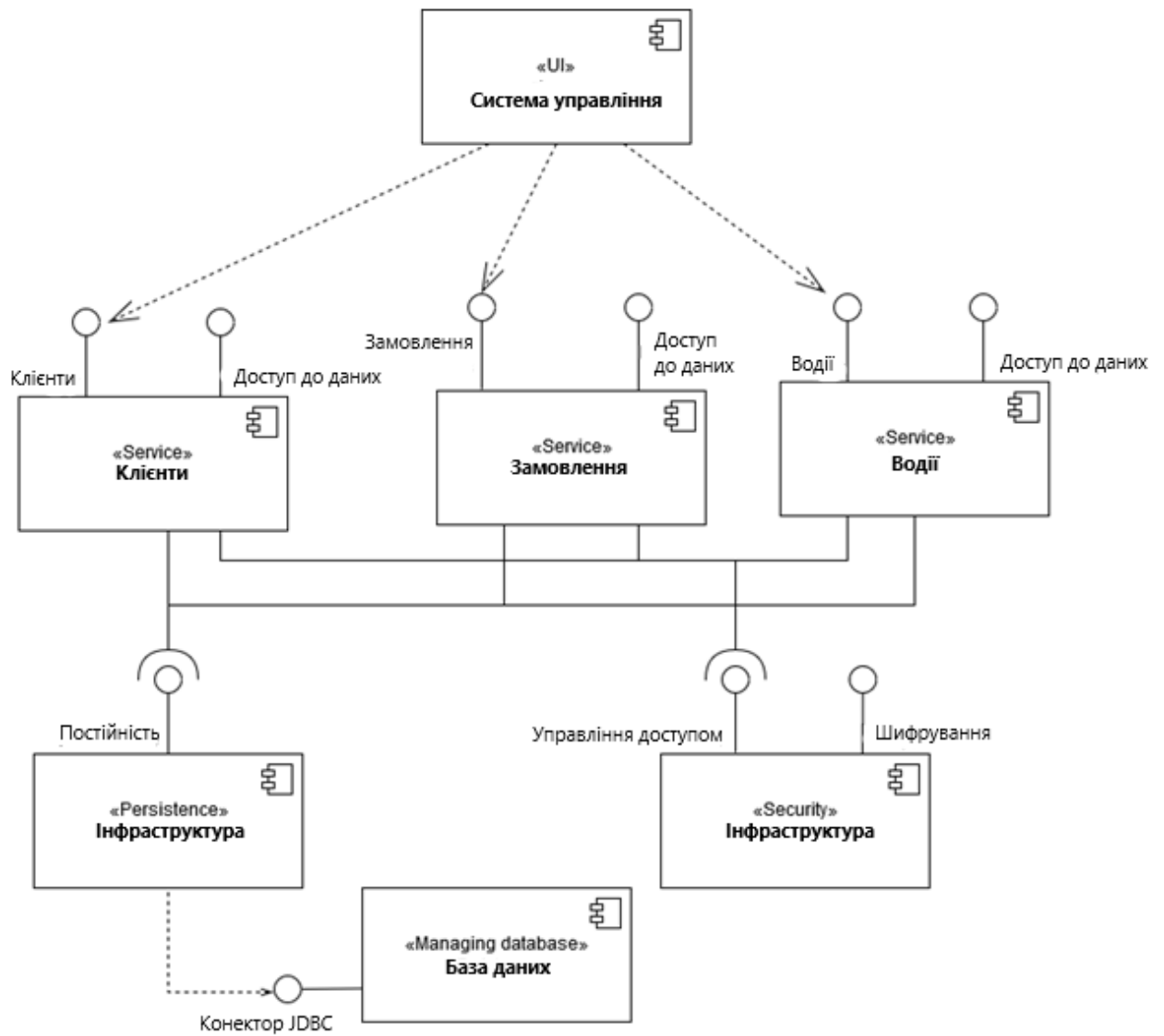
The field of cargo transportation management is of great importance in modern business processes related to cargo transportation.

In the work, the subject area was investigated and the basic requirements for the software and hardware support of the automated workplace of the cargo transportation dispatcher were established. Functional and non-functional requirements and options for their use are defined. Constructed class and component diagrams. These diagrams help in the development process. The work of existing systems, their advantages and disadvantages is considered.

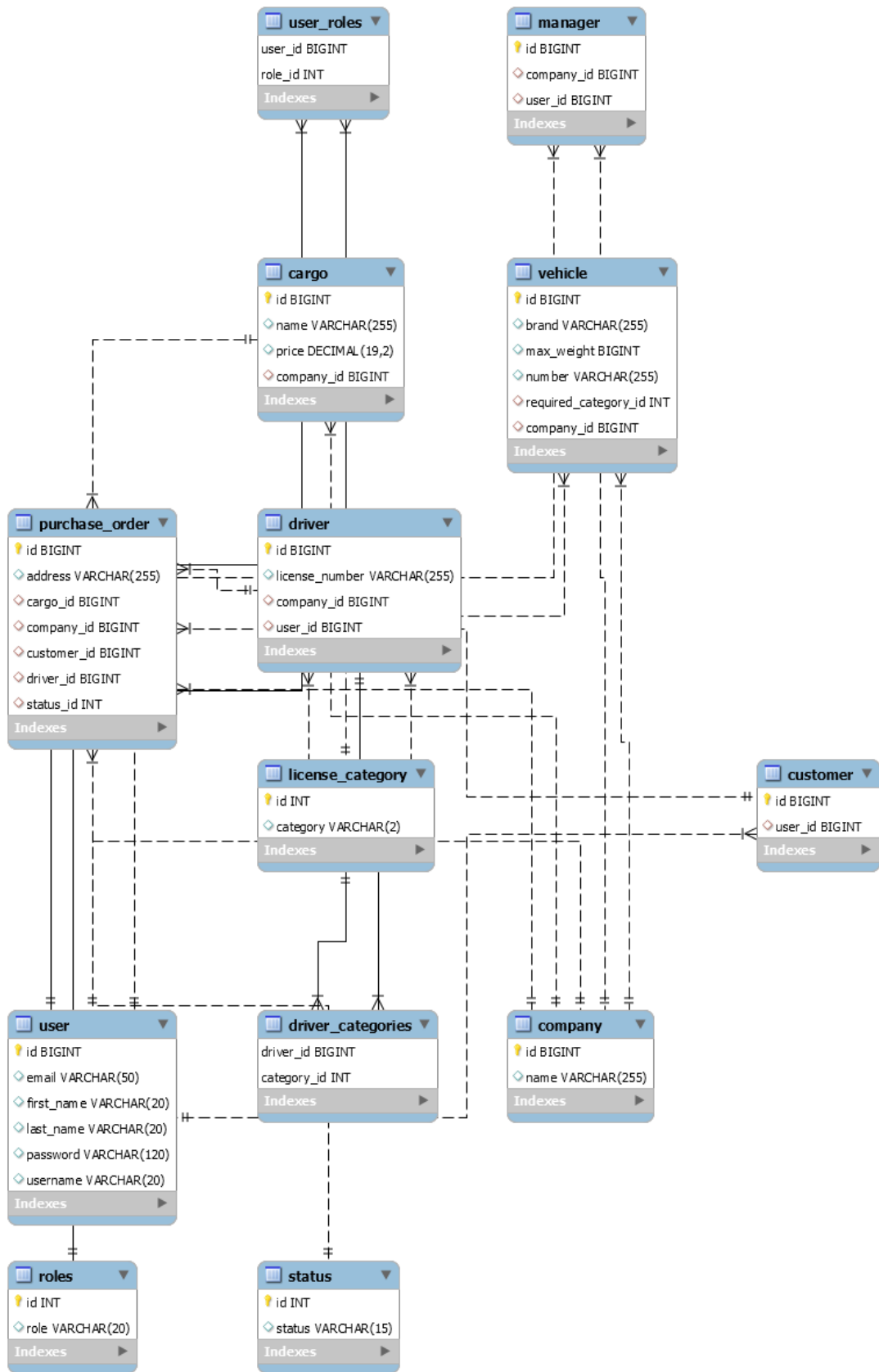
As a result of the system design, the main use cases of the cargo management process for the cargo manager were described and covered by the program code.

In the development process, modern technologies of database design, server services and web pages were used. New technologies were applied for a safe and reliable prototype of the application.

<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
Розробив		Підсадочий В.П.			Abstract	<i>Лім.</i>	<i>Арк.</i>	<i>Аркушіє</i>
Перевірів		Дзундза Б.С.					4	1
Н. Контр.								
Затвердив								



Змн.	Арк.	№ докум.	Підпис	Дата				
Розробив		Підсачочий В.П.			Діаграма взаємодії КОМПОНЕНТІВ	Літ.	Арк.	Аркушів
Перевірив		Дзундза Б.С.					5	1
Н. Контр.								
Затвердив								



Змн.	Арк.	№ докум.	Підпис	Дата
Розробив		Кушнір А.М.		
Перевірив		Дзундза Б.С.		
Н. Контр.				
Затвердив				

Діаграма зв'язку бази даних

Літ.	Арк.	Аркуші
	6	1

Пояснювальна записка

до кваліфікаційної роботи

на тему:

«Автоматизоване робоче місце диспетчера вантажоперевезень»

					123.КІ-41.3			
<i>Змн.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>				
Розробив		Кушнір А.М.			Пояснювальна записка	<i>Літ.</i>	<i>Арк.</i>	<i>Аркушіє</i>
Перевірив		Дзундза Б.С.					7	75
Н. Контр.								
Затвердив								

Зміст

Список скорочень

Вступ

РОЗДІЛ 1. АНАЛІЗ ОБЛАСТІ ЗАСТОСУВАННЯ

- 1.1. Огляд управління вантажними перевезеннями
- 1.2. Види вантажних перевезень
- 1.3. Основні учасники процесу управління вантажними перевезеннями
- 1.4. Види товарів
- 1.5. Аналіз існуючих систем управління вантажами

Висновок

РОЗДІЛ 2. СПЕЦИФІКАЦІЯ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

- 2.1. Функціональні вимоги
- 2.2. Нефункціональні вимоги
- 2.3. Діаграма варіантів використання

Вистовок

РОЗДІЛ 3. АРХІТЕКТУРА ТА ДИЗАЙН ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

- 3.1. Архітектура програми
- 3.2. Діаграма компонентів
- 3.3. Діаграма класів
- 3.4. База даних
- 3.5. Схема розгортання

Висновок

РОЗДІЛ 4. ПРОТОТИП СИСТЕМИ

- 4.1. Використовувані технології
- 4.2. Приклади роботи

Висновок

Висновки

Література

					123.KI-41.3	Арк.
						8
Зм.	Арк.	№ докум.	Підпис	Дата		

СПИСОК СКОРОЧЕНЬ

- API - Інтерфейс прикладного програмування
- UI - Інтерфейс користувача
- JSON - Нотація об'єктів Javascript
- TMS - Система управління транспортом
- REST - Представницький стан передачі
- HTML - Мова розмітки гіпертексту
- CSS - Каскадні таблиці стилів
- GPS - Глобальна система позиціонування
- SRS - Специфікація системних вимог

					<i>123.KI-41.3</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		9

ВСТУП

З кожним роком людство винаходить все нові і нові продукти. Проте не так багато продуктів зникають із повсякденного життя людей. Все, що може лежати перед людиною, на столі, на землі, було якимось чином зроблено і доставлено сюди. Щоб гідно підтримувати повсякденне існування людини, кожен із нас повинен користуватися різними благами. Кожен виріб має своє місце виробництва і не завжди це місце поруч. Більшість товарів везуть з іншого кінця світу.

Розглядаючи перевезення вантажів у сучасному світі, зрозуміло, що обсяги перевезень надзвичайно великі і без автоматизованих систем обійтися неможливо. Для ефективного управління та відстеження транспортування вантажів по всьому світу та можливості розподілу навантаження на перевізників створюються автоматизовані системи вантажних перевезень.

					<i>123.KI-41.3</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		10

РОЗДІЛ 1.
АНАЛІЗ ОБЛАСТІ ЗАСТОСУВАННЯ

1.1. Огляд управління вантажними перевезеннями

«Управління вантажними перевезеннями — це процес нагляду та управління економічною операцією та доставкою товарів. Управління вантажними перевезеннями поєднує досвід логістики, людські ресурси та знання, щоб забезпечити безперебійну координацію між перевізниками та вантажовідправниками».

Управління вантажними перевезеннями — це частина управління логістикою, яка охоплює величезну сферу управління, транспортування, продажу та зберігання товарів і водночас контролює відносини з клієнтами та відповідає вимогам клієнтів.

«Логістика відноситься до загального процесу управління тим, як ресурси отримуються, зберігаються та транспортуються до місця призначення. Управління логістикою передбачає визначення потенційних дистриб'юторів і постачальників і визначення їх ефективності та доступності».

Процес управління вантажними перевезеннями за своєю формою, згідно з цим визначенням, включає:

- Встановлення контакту з клієнтами та перевізниками.
- Підготовка всіх необхідних документів для перевезення документів.
- Управління транспортуванням вантажів у бажаний час.
- Управління рахунками та платежами.
- Оптимізація доставки товару.

Цей список можна розширювати все більшою кількістю елементів. Кожен пункт може характеризувати конкретну область у процесі управління вантажними перевезеннями. Компанії, які використовують вантажні перевезення, можуть персоналізувати різні елементи для свого процесу управління. Такий підхід дозволяє створити гнучку модель управління, індивідуальну для кожної компанії.

					123.KI-41.3	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		11

Світова економіка швидко зростає і вимагає все більше ресурсів для збільшення темпів зростання та вдосконалення технологій, які використовуються в усій системі управління цією складною сферою.

Кожен бізнес, який хоче заробляти гроші та конкурувати з іншими компаніями, повинен вдосконалювати кожну використану технологію та автоматизувати процес управління кожною окремою частиною бізнесу.

Сьогодні неможливо бути конкуруючою компанією із застарілими технологіями та не автоматизованими процесами. В останні роки інформаційні технології стали невід'ємною частиною будь-якої сфери життя людини.

Сфера управління вантажними перевезеннями великих сучасних компаній не може бути ефективно реалізована без використання ІТ. Чим більша компанія, тим важче контролювати внутрішні частини. Існує верхня межа, коли наймання людей стає не вигідним, оскільки комп'ютерні системи можуть виконувати ті самі завдання швидше та безпомилково. Одна особа не може керувати кількістю запитаних або доставлених товарів без будь-яких допоміжних інструментів. Кожен день можуть бути мільйони запитів і замовлень, створених клієнтами. Найменша затримка або помилка призведе до втрати грошей.

Для клієнта бізнес-сфери зворотній зв'язок є найціннішою характеристикою виконаних завдань. Кожна компанія має свій список клієнтів, які покладаються на цінності, надані кожною компанією. Якщо клієнт задоволений наданими послугами, це сприяє хорошим відносинам між клієнтом і компанією. Цього легко досягти, якщо компанія використовує найкращі технології, програмне та апаратне забезпечення. Якщо всі ці речі зручні для клієнта, в результаті буде відбуватися постійне зростання і вдосконалення.

Таким чином, кожна компанія змушена використовувати автоматизовані системи, щоб мати можливість конкурувати з іншими компаніями. Ймовірно, вони використовують певні підходи, але в цілому успішний процес управління схожий у всіх.

1.2. Види вантажних перевезень

					123.KI-41.3	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		12

Існують різні види транспортування вантажів. Залежно від кількості, відстані та часу можна використовувати чотири основних види транспорту. (рис. 1.1.)

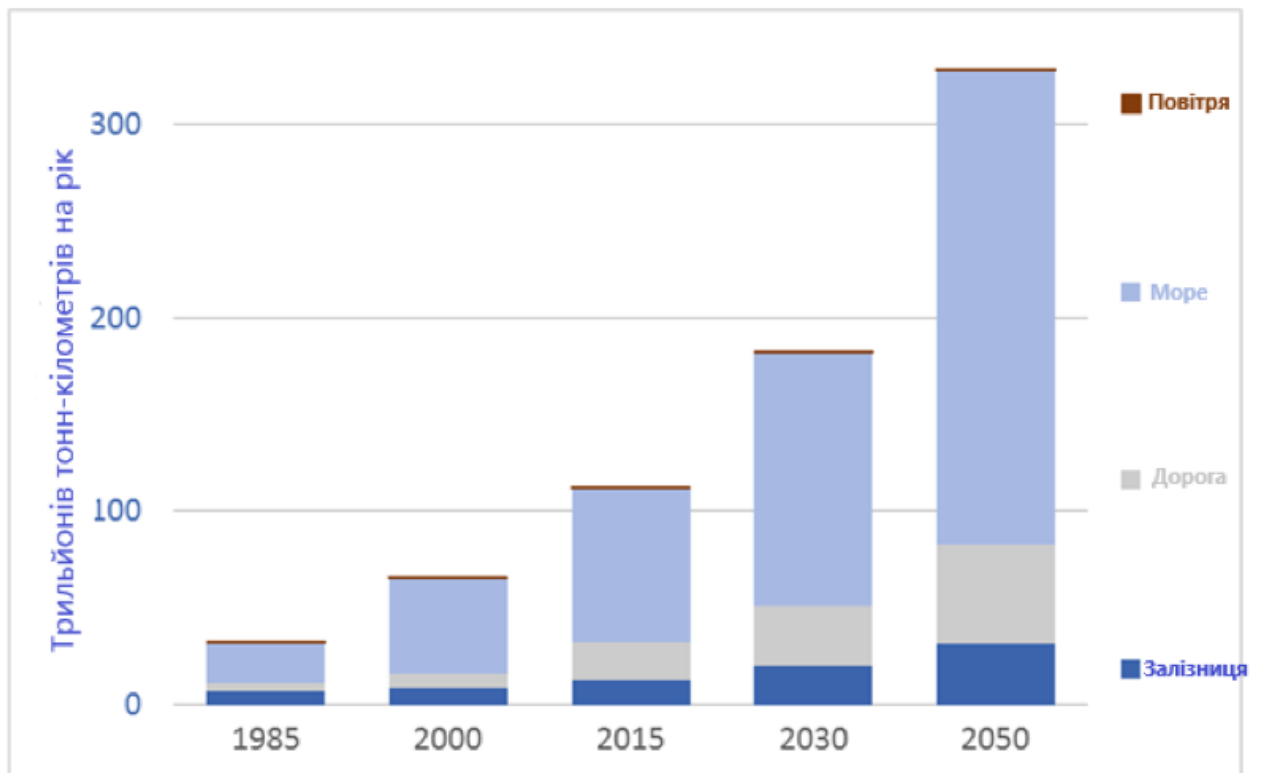


Рис. 1.1. Обсяги світових вантажоперевезень

Морські перевезення становлять найбільшу частину світового обсягу вантажообігу. Щороку він перевозить велику кількість вантажів. Деякі види транспорту неможливо використовувати через питому вагу або об'єм вантажу, але океанські перевезення з цим справляються.

Вантажні перевезення морем є менш дорогим способом доставки товарів, ніж інші, але в той же час він вимагає більше часу в дорозі. Морський транспорт часто використовується для доставки сільськогосподарських товарів, таких як пшениця, кукурудза та інше зерно. Крім того, морські вантажні перевезення використовуються для переміщення великих машин, таких як гвинти вітряків, двигуни для заводів тощо.

Крім того, морські вантажоперевезення більш екологічні, ніж інші види транспорту. Він споживає менше палива на одну одиницю транспорту, ніж будь-який інший тип.

Основні переваги морських перевезень:

- Може переміщати різні хороші типи
- Велика вантажопідйомність
- Найменший вплив на природу
- Різні варіанти доставки.

У випадку, коли необхідно доставити вантаж в короткі терміни на далекі відстані, авіап перевезення є найкращим вибором. Цей вид вантажоперевезень є найдорожчим, але забезпечує найшвидший спосіб доставки товару.

Недоліком такого варіанту транспортного обслуговування є обмеження на тип і габарити вантажу, які суворо прописані в правилах авіап перевезень.

Основні переваги авіап перевезень включають

- Швидка доставка товару
- Менше обробки вантажу
- Надійне прибуття та відправлення
- Підвищений рівень безпеки вашого вантажу

Залізничний тип перевезень дозволяє доставляти великогабаритні та негабаритні вантажі. До складу поїзда може входити будь-яка кількість контейнерів, цистерн, вагонів. Це зручний спосіб транспортування великої кількості вантажів по суші.

Серед інших видів вантажних перевезень залізниця надзвичайно популярна в гірничодобувній та важкій промисловості, оскільки майже весь інший транспорт програє залізниці за вантажопідйомністю, а часто й за вартістю.

Ключові переваги залізничних вантажних перевезень включають

- Надійний час і розклад транспортування
- Велика вантажопідйомність
- Швидка та економічна доставка на великі відстані.

Перевезення автомобільним транспортом в ряді випадків є найбільш вигідним. Такий варіант організації доставки вантажу збільшує швидкість перевезення, а також знижує загальну вартість. Такий тип є найкращим вибором для доставки на короткі відстані та невеликої кількості товару. Це забезпечує гарне співвідношення витрачених ресурсів до кількості перевезених вантажів.

					<i>123.KI-41.3</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		14

Переваги автомобільних перевезень включають можливість переміщення вантажів практично в будь-яку точку країни та за кордон, оскільки є певна гнучкість у підготовці маршруту. Завантаження товару може бути здійснено в будь-якому місці, вказаному відправником.

Недоліками даного типу є обмеження розміру вантажу, кількості товару. Також буває складно доставити вантаж в місця з поганими дорогами.

Автомобільні види вантажних перевезень в більшості випадків це:

- Економічно ефективний
- Швидко та за розкладом
- Легко контролювати
- Доставка товарів «від дверей до дверей».

Однак вантажний транспорт дещо обмежений щодо того, що він може перевозити, розміром транспортних засобів, що використовуються, а також обмеженнями розміру та ваги. Іншим обмеженням є те, що на нього впливають погода, дорожні умови та дорожній рух.

1.3. Основні учасники процесу управління вантажними перевезеннями

Менеджер по вантажу – це особа, яка контролює всі взаємодії з вантажем.

Менеджеру необхідно мати глибокі знання про всі види транспорту та специфіку кожного виду для продуктивної роботи.

Основні завдання менеджера - контролювати виконання всіх замовлень, спілкуватися з клієнтом і отримувати зворотній зв'язок, розподіляти замовлення водіям, які будуть доставляти вантаж. Йому потрібно точно знати графік відправлень, які виконуються щодня, кількість вантажу, який потрібно доставити, види транспорту, які використовуються для доставки товарів, і статистику доходів і результатів у реальному часі.

Менеджер також повинен спілкуватися з усіма відділами компанії, щоб відстежувати, що відбувається в офісах продажів і маркетингу, щоб знати актуальну інформацію про кількість товарів, які можна доставити, відгуки клієнтів та іншу інформацію, яка може допомогти продуктивно керувати процесом.

					123.KI-41.3	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		15

Вантажовласник — це фізична або юридична особа, яка має певний вид вантажу і може надати необхідну кількість товару тому, хто його потребує. Вантажовласник може працювати як з різними компаніями, так і з однією. Це залежить від того, який вантаж він може надати або де територіально розташований склад вантажовласника.

Перевізник — фізична або юридична особа, яка має можливість транспортувати вантаж за допомогою транспорту відповідно до одного з видів вантажних перевезень. Перевізник забезпечує спосіб взаємодії вантажовласника і клієнта. Це така послуга доставки вантажів, яка може бути сторонньою або всередині компанії вантажовласника.

Водій — особа, яка закріплена за певним транспортом і здійснює доставку вантажу. Кожен водій повинен мати права керувати конкретним транспортом, який використовується для транзиту. Наприклад, в автомобільних перевезеннях водії повинні мати категорії В, ВЕ, С, СЕ в загальних випадках. Ці категорії дозволяють керувати певним видом транспорту, таким як легкові автомобілі та вантажівки з причепами.

1.4. Види товарів

Існує п'ять основних видів товарів, які щодня переміщуються разом зі світом. До цих типів належать контейнерні вантажі, рідкі наливні вантажі, сухі наливні вантажі, наливні вантажі та вантажі в рулонах.

Це загальний вид вантажу, який не перевозиться в контейнерах або танкерах. Такі товари можуть бути упаковані в різні коробки або пакети і мати різну кількість незалежно один від одного.

Це рідкий вантаж, який можна перевозити в танкері. Танкер - транспортний засіб, призначений для перевезення або зберігання рідин або газів наливом. Танкери можуть перевозити величезну кількість різноманітних наливних вантажів.

Як от:

- Прісноводні
- Масло

					123.KI-41.3	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		16

- Нафтове паливо
- Різні хімікати

Це всі товари, які перевозяться без упаковки у великих кількостях. Наприклад, більшість сільськогосподарських товарів, таких як зерно або кукурудза, а також гравій і пісок перевозяться як сухі маси

Контейнери - це стандартизовані коробки, які мають свої розміри. Контейнери можуть мати різні розміри, але більшість з них мають однаковий вигляд. Таким чином, ці контейнери можна транспортувати кількома різними видами транспорту. Наприклад, один контейнер можна завантажити на корабель, перемістити через море, а потім у порту контейнер можна завантажити на вантажівку та перемістити по суші.

Вантаж Ро-Ро - це всі вантажі з колесами, які можуть котитися на несучому транспорті та з нього. Наприклад, усі легкові, вантажівки та тягачі є вантажами RoRo. Цей вид вантажу може бути доставлений на спеціальних платформах. Ці платформи повинні мати спеціальні кріплення для коліс.

Вантажна техніка, яка не має коліс, розміщується на платформі, яка може котитися по транспортному модулю. Отже, його можна назвати RoRo cargo.

1.5. Аналіз існуючих систем управління вантажами

Ларді-Транс — міжнародна третя сторона, що займається автоперевезеннями. Він має велику інфраструктуру, розвинену протягом багатьох років. У цієї компанії багато клієнтів і перевізників.

Це зручний спосіб зареєструвати транспорт або вантажопостачання. Ви можете легко шукати вантажні перевезення в головному вікні (рис. 1.2.).

					<i>123.KI-41.3</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		17

Рис. 1.2. Інтерфейс Lardi-trans

Переваги:

- Велика інфраструктура
- Кількість послуг
- Додавання транспорту
- Додавання вашого вантажу
- Керуйте драйверами
- GPS відстеження
- Список усіх перевезень

Недоліки:

- Більшість сервісів відокремлені від основного сайту, тому важко орієнтуватися.
- Половина сайту має новий інтерфейс, інша частина має старий інтерфейс, тому завжди дратує шукати потрібну кнопку, коли інтерфейс змінюється.

Della є міжнародною третьою частиною компанії з автомобільних вантажних перевезень. Це дозволяє знайти перевізника та додати ваші товари для завантаження. У ньому є панелі пошуку для пошуку відповідного перевізника для транспортування ваших товарів, або ви можете знайти товари, які можете транспортувати.

Також ця система дозволяє розміщувати пропозиції на перевезення (рис. 1.3.).

Рис. 1.3. Інтерфейс Della

Переваги:

- Можливість знайти перевізника
- Можливість знайти товар для переміщення
- Можливість додати свій товар і транспорт
- Список усіх перевезень

Недоліки:

- Незручний інтерфейс користувача
- Важко знайти необхідний функціонал

а. Висновок

Управління вантажами - це комплексна система управління доставкою вантажів по певній території: місто, країна, світ. Ця сфера охоплює важливу частину сучасної економіки світу. Усі вантажоперевезення здійснюються чотирма основними видами транспортування: морським, залізничним, повітряним та автомобільним.

Менеджер з вантажів виконує основну роль у контролі процесу доставки між клієнтом і перевізником. У більшості випадків компанія-перевізник і компанія-менеджер вантажу відрізняються. У цьому випадку компанія-перевізник

									123.KI-41.3	Арк.
										19
Зм.	Арк.	№ докум.	Підпис	Дата						

називається стороннім постачальником логістики. Крім того, якщо компанія з управління вантажними перевезеннями продає свої товари і не використовує інших постачальників, ця компанія може використовувати різні системи управління незалежно.

					<i>123.KI-41.3</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		20

РОЗДІЛ 2.

СПЕЦИФІКАЦІЯ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

«Специфікація вимог до системи (SRS) (також відома як специфікація вимог до програмного забезпечення) — це документ або комплект документації, який описує функції та поведінку системи або програмного додатка. Він включає різноманітні елементи (див. нижче), які намагаються визначити передбачувану функціональність, необхідну замовнику для задоволення різних користувачів.

Крім визначення того, як повинна вести себе система, специфікація також визначає на високому рівні основні бізнес-процеси, які підтримуватимуться, які припущення для спрощення були зроблені та які ключові параметри продуктивності повинні відповідати системі». [4]

Основна мета цієї системи – звести до мінімуму зусилля з управління відносинами між клієнтами та транспортними компаніями. В результаті буде створено прототип системи для автоматизації процесу управління замовленнями, вантажами, водіями та транспортом транспортної компанії.

Користувачі системи повинні бути відповідальними за свої дії та усвідомлювати, яка відповідальність лежить на них. Кожен користувач повинен надати правдиву інформацію про себе. Для реєстрації в системі користувачеві необхідно надати персональні дані для ідентифікації особи та вибрати роль. Таким чином, для реєстрації необхідно, щоб водій мав водійське посвідчення та необхідну категорію, менеджер повинен вказати вантажну компанію, а замовником може бути звичайна особа, яка замовить вантаж і отримає доставку.

2.1. Функціональні вимоги

Основною метою системи управління вантажними перевезеннями є спрощення та автоматизація основних завдань, які виконує менеджер. Загалом, така система повинна забезпечувати широкий спектр обов'язків, які можна виконувати в процесі доставки товарів, управління замовленнями, призначення водіїв на ці замовлення або просто змінювати деякі властивості транспорту.

					123.KI-41.3	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		21

Виходячи з того факту, що в системі є кілька акторів, функціональні вимоги пов'язані з кожним актором. Основними дійовими особами системи є менеджер, клієнт і водій.

Перелік функціональних вимог системи записано в таблиці нижче. Функціональні вимоги описують поведінку системи та результат, який отримає користувач під час використання системи.

У таблиці 2.1 описані функціональні вимоги системи.

Таблиця 2.1.

Функціональні вимоги до системи

	Опис функціональної вимоги
	Система відкриває сторінку для створення облікового запису, коли користувач натискає кнопку «Реєстрація».
	Система має відображати три ролі у формі реєстрації, коли відкрита сторінка реєстрації.
	Система повинна надавати можливість вибрати одну з трьох ролей на сторінці реєстрації.
	Під час реєстрації система надає користувачеві поля для введення особистої інформації, такої як ім'я, прізвище, адреса електронної пошти, ім'я користувача та пароль.
	Під час реєстрації в якості водія система повинна надавати користувачеві поля для введення додаткової інформації, такої як номер водійського посвідчення та категорії посвідчення.
	Під час реєстрації в якості менеджера система повинна надавати користувачу поле для введення додаткової інформації, такої як назва компанії.
	Система перевіряє унікальність введених даних, таких як електронна адреса, ім'я користувача, назва компанії та номер водійського посвідчення.

	Система повинна забезпечувати можливість введення облікових даних користувача у формі входу.
	Коли користувач увійшов як адміністратор, з'являється кнопка дошки менеджерів.
0	Система повинна надавати можливість керувати списком вантажів для менеджера.
1	Коли менеджер видалляє вантаж, видалена позиція повинна зникати зі списку вантажів.
2	Коли менеджер відкриває компонент вантажу список, то відображається список доданих вантажів і поля для додавання нового вантажу.
3	Коли менеджер додає новий вантаж, доданий товар повинен з'явитися в списку вантажів.
4	Система повинна надавати можливість керувати замовленнями для менеджера.
5	Коли менеджер переходить до списку замовлень, мають відображатися існуючі замовлення.
6	Коли менеджер натискає кнопку Деталі, з'являється вікно з деталями замовлення.
7	Коли менеджер натискає кнопку Призначити водія, відкривається вікно зі списком водіїв і списком транспорту.
8	Коли менеджер вибирає водія та транспорт і натискає кнопку «Надіслати» у вікні «Призначити водія», вибраний водій і транспорт прив'язуються до замовлення.
9	Коли менеджер натискає кнопку «Водії», з'являється список водіїв, призначених компанії менеджера.
0	Коли менеджер натискає на кнопку Панель, то з'являється загальна інформація про кількість замовлень, водіїв, транспорт і вантажі.

1	Коли користувач увійшов як Клієнт, з'являється кнопка Панель клієнтів.
2	Коли клієнт натискає кнопку Панель клієнтів, з'являється сторінка зі списком доступних вантажів із кнопкою замовлення.
3	Коли клієнт натискає кнопку «Замовити», відкривається сторінка з формою для перевірки нового замовлення з вибраним вантажем.
4	Коли клієнт заповнює форму оформлення замовлення та натискає кнопку відправки, створюється нове замовлення та відкривається сторінка із замовленням користувача.
5	Коли користувач увійшов як водій, з'являється кнопка панелі драйверів.
6	Коли водій натискає кнопку «Замовлення», відкривається список призначених замовлень.
7	Коли водій натискає кнопку «Транспорт», відкривається список призначеного транспорту.
8	Коли водій доставив вантаж, він натиснув кнопку «Доставлено», і замовлення має змінити статус.

2.2. Нефункціональні вимоги

У таблиці 2.2 визначені нефункціональні вимоги. Ці вимоги описують загальні характеристики системних атрибутів, таких як безпека, надійність, продуктивність, зручність обслуговування, масштабованість і корисність. Вони описують цінні властивості системи та очікування користувачів.

Нефункціональні вимоги показують, наскільки висока продуктивність системи, досвід користувача та загальні системні обмеження відповідно до «ISO/IEC 25010» [6]

можуть бути введені такі типи нефункціональних характеристик якості:

функціональна придатність, продуктивність, надійність, зручність використання, безпека, сумісність, ремонтпридатність і портативність.

					123.KI-41.3	Арк.
						24
Зм.	Арк.	№ докум.	Підпис	Дата		

Таблиця 2.2.

Нефункціональні вимоги

	Опис функціональної вимоги
	Усі веб-сторінки завантажуються протягом 4 секунд.
	Система має бути захищена за допомогою маркера безпеки для автентифікації.
	Система повинна бути захищена наданням доступу користувачам із певними ролями.
	Основна тематика системи повинна бути в темних тонах.
	Інтерфейс користувача дозволить користувачам отримати доступ до даних свого профілю з будь-якої сторінки за 3 кліки.
	Система повинна мати можливість змінити серверну або зовнішню частину системи на іншу.
	Система повинна забезпечувати можливість кодування паролів і збереження закодованого значення в базах даних.
	Серверна частина має бути реалізована як служба REST.
	Internet Explorer не підтримується.
0	Система повинна забезпечувати одночасне використання.

2.3. Діаграма варіантів використання

Діаграма варіантів використання (рис. 2.1.) описує основні зв'язки та варіанти використання між суб'єктами системи. У цьому випадку три актори взаємодіють один з одним і будують єдиний потік комунікації.

Менеджер є основною ланкою в системі, яка керує всіма частинами системи та отримує результати взаємодії клієнта та водія.

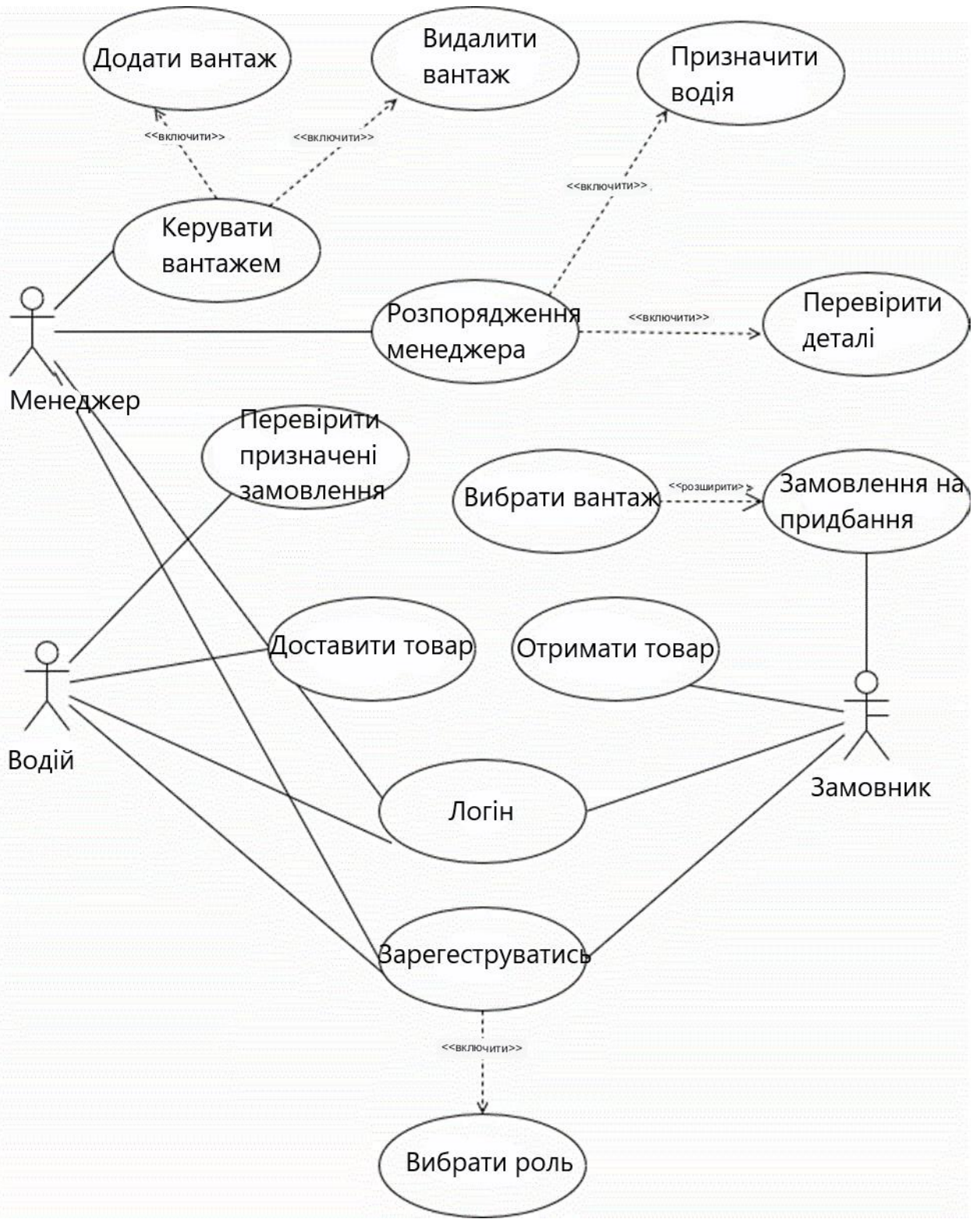


Рис. 2.1. Діаграма варіантів використання

в. Висновок

У цьому розділі встановлено функціональні та нефункціональні вимоги до системи та варіанти її використання. Була побудована діаграма варіантів використання, яка описує основні варіанти використання.

РОЗДІЛ 3.

АРХІТЕКТУРА ТА ДИЗАЙН ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

3.1. Архітектура програми

Загалом, існує велика кількість архітектур, які успішно використовуються для проектування програмної системи. Більшість із них надають суворі вимоги до створення компонентів та їхніх зв'язків. Інші забезпечують гнучку структуру з можливістю налаштування структури та зв'язків.

Цей прототип включає три основні частини: серверний модуль із сервісною логікою, зовнішній модуль зі службами інтерфейсу користувача, який відображає дані для користувача та реагує на введення користувача, і базу даних, яка зберігає всі необхідні дані та надає їх для запитів серверної служби, яка працює з HTTP-запитами від модуля UI.

Кожен компонент прототипу можна визначити як окремий модуль зі своєю логікою. Ці модулі можна легко замінити іншими, які реалізують комунікаційний контракт. Наприклад, система може змінити постачальника бази даних, просто змінивши файл конфігурації, або вона може перейти на інші модулі інтерфейсу користувача, які взаємодіятимуть із серверними службами через HTTP-виклики.

Внутрішня частина має багаторівневу архітектуру з трьома основними рівнями.

По-перше, це рівень даних, який забезпечує шлях зв'язку між базою даних і перетворенням об'єктів бази даних у прості об'єкти, якими можуть маніпулювати служби.

Другий – рівень обслуговування. Ця частина серверної логіки забезпечує різні операції з отриманими даними та їх транспортування до контролерів.

І останнім є рівень контролера, який збирає дані, що повертаються службами, і перетворює їх у JSON перед надсиланням у відповідь. Крім того, контролери отримують HTTP-запити, які описують, який метод слід викликати та які дані потрібно повернути.

					123.KI-41.3	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		27

Бекенд-служба, реалізована як служба REST API, яка надає точки входу для запитів HTTP. Служби REST API працюють з ресурсами, які вони можуть передавати на вимогу запиту користувача або іншого запиту служби. Такі ресурси можуть представляти різні моделі даних або типи даних, які десь зберігаються.

Існує єдиний інтерфейс для взаємодії з таким сервісом. Оскільки REST API використовує запит HTTP для отримання даних, існує чотири основних типи запитів:

- ОТРИМАТИ
- ПОСТАВИТИ
- ПУБЛІКАЦІЯ
- ВИДАЛИТИ

Кожен тип запиту повинен мати власний ідентифікатор і набір параметрів, за допомогою яких сервер визначить, який контролер повинен взяти на обробку вхідний запит.

Основне відображення REST API, описане в таблиці 3.1.

Таблиця 3.1.

Відображення REST API

HTTP method	Mapping	Pre authorize role	Description
GET	/api/auth/user + Request param userID	Manager, Driver, Customer	Returns the user data according to specified parameter.
POST	/api/auth/signin + Request body with credentials	Any	Returns user info with personal authorization token.
POST	/api/auth/signup + Request body with credentials	Any	Creates a user and return result info message.

GET	/api/cargo	Manager, Customer	Return the list of available cargoes.
POST	/api/cargo + Request body with filled info	Manager	Creates the new cargo
DELETE	/api/cargo + Request param name of the cargo	Manager	Deletes the cargo from database
GET	/api/driver/all	Manager	Return list of drivers
GET	/api/order/all	Manager	Return list of orders
POST	/api/order/ + Request body	Customer	Creates new order
GET	/api/order/customer + Request param username	Customer	Return list of order of specified customer
GET	/api/check/all	All	Return success response for users without roles
GET	/api/check/customer	Customer	Return failure response for users without customer role
GET	/api/check/driver	Driver	Return failure response for users without driver role
GET	/api/check/manager	Manager	Return failure response for users without manager role

Зм.	Арк.	№ докум.	Підпис	Дата

123.KI-41.3

Арк.

29

Для керування ролями користувачів використовується конфігурація безпеки. Користувач деталізує інформацію, що зберігається в базі даних. Коли користувач хоче отримати доступ до якогось компонента, йому необхідно надати облікові дані та особистий маркер, щоб відповідати наданій ролі. У випадку, якщо користувач має необхідну роль, конфігурація безпеки приймає запит, якщо користувачеві не надано доступ до певного ресурсу, конфігурація безпеки відхиляє запит.

Внутрішні служби спілкуються з базою даних через структуру об'єктно-реляційного відображення. Структура ORM використовує з'єднувач для конкретної бази даних, тому, щоб змінити базу даних, ви можете просто змінити з'єднувач.

Таблиці бази даних створюються за допомогою ORM. Отже, спочатку потрібно, щоб сутності та їхні зв'язки були описані в кодї. На етапі початкового завантаження розгортання програми будуть створені таблиці бази даних, і залежно від конфігураційних файлів уся структура системної бази даних буде збережена або видалена після завершення роботи програми.

Інтерфейсну частину прототипу системи можна розглядати як окремий модуль з компонентною архітектурою. Цей архітектурний стиль забезпечує зручне рішення для побудови чітко визначених комунікаційних інтерфейсів із багаторазовими бінарними частинами. Відповідно до цього підходу компоненти можна легко замінити, повторно використовувати або розширити необхідним чином.

3.2. Діаграма компонентів

На (рис. 3.1.) описано загальний процес взаємозв'язку між основними компонентами. Діаграма компонентів використовується для візуалізації взаємодії між компонентами системи.

					123.KI-41.3	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		30

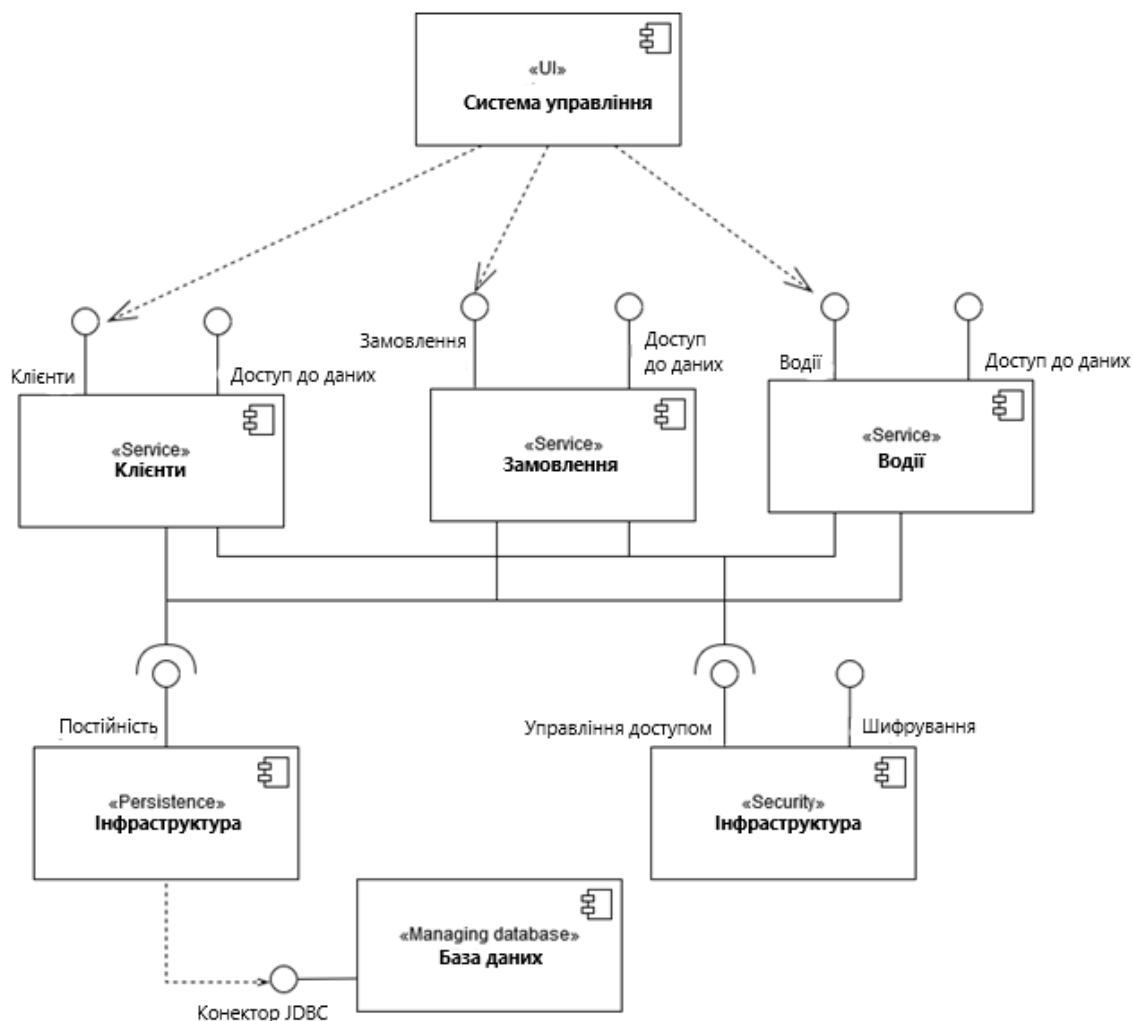


Рис. 3.1. Діаграма компонентів

3.3. Діаграма класів

(рис. 3.2.) визначає основні класи моделі предметної області. Загалом існує три основні класи, які описують користувачів: клієнт, менеджер і водій. Ці класи пов'язані з класом користувача, який інкапсулює основну інформацію про користувача та його роль.

Класи замовлення та компанії містять інформацію про учасників та призначені сутності для конкретного замовлення чи компанії. Ці класи діють як власники даних, які потрібно отримати.

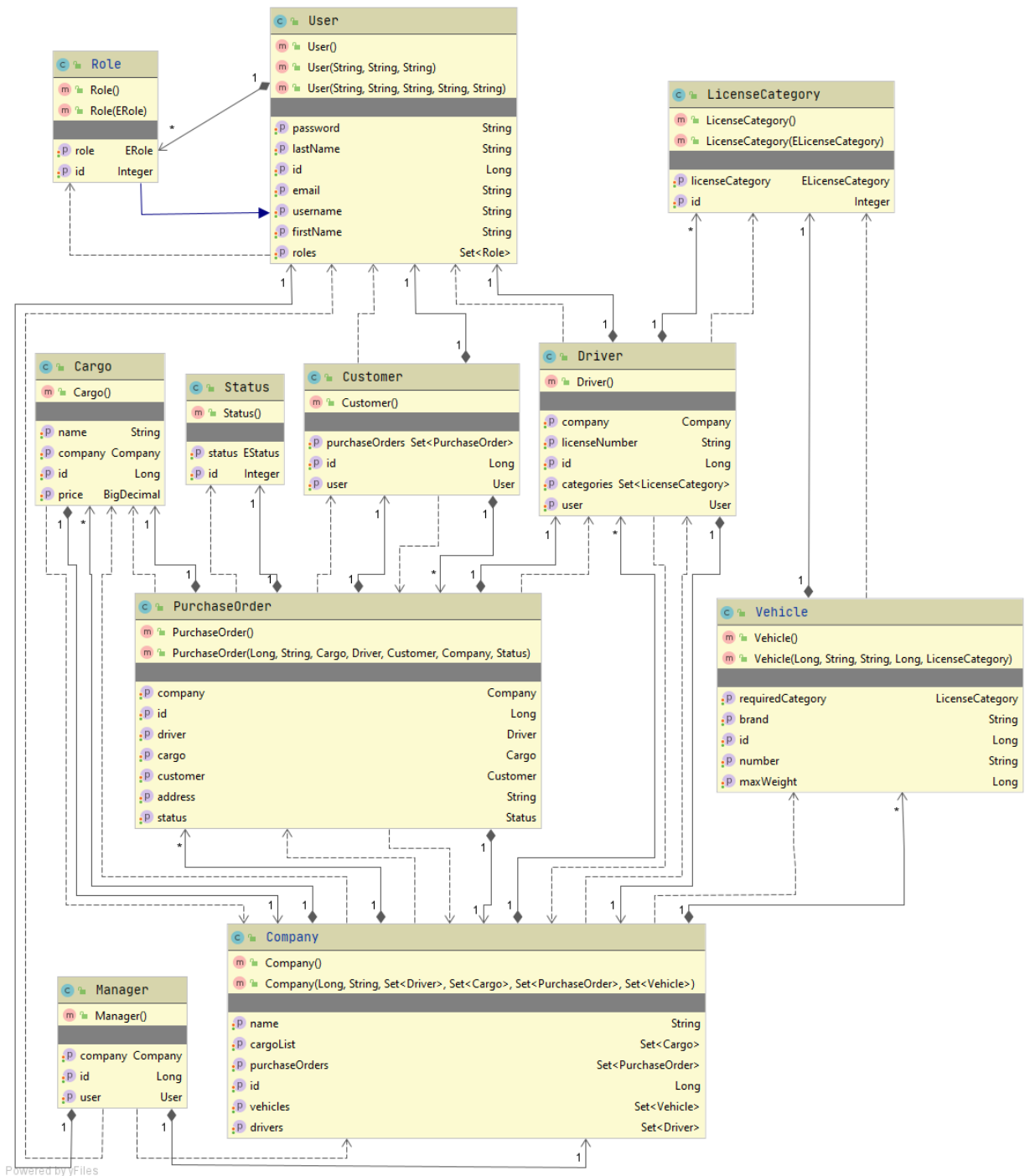


Рис. 3.2. Діаграма класів

3.4. База даних

На (рис. 3.3.) показана діаграма бази даних.

Зм.	Арк.	№ докум.	Підпис	Дата

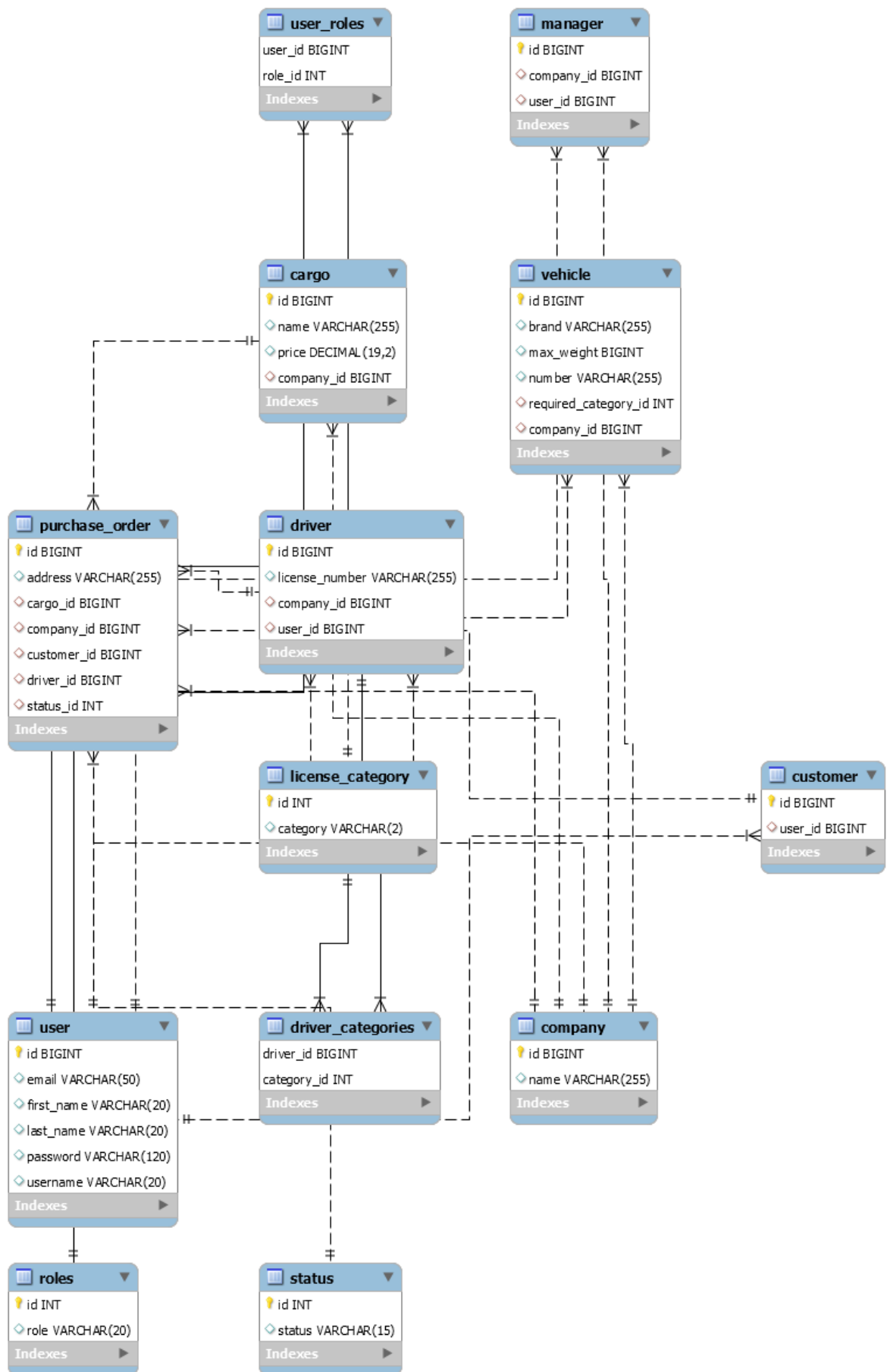


Рис. 3.3. Діаграма зв'язку бази даних

Зм.	Арк.	№ докум.	Підпис	Дата

У таблицях нижче описана структура сутностей бази даних.

Таблиця 3.2.

Відношення сутності `User_Roles`

Column name	Data type	Length	Integrity constraint
user_id	big_int		Foreign key references to the `id` of table `user`, Not null
role_id	Int		Foreign key references to the `id` of table `roles`, Not null

Таблиця 3.3.

Відношення сутності `Roles`

Column name	Data type	Length	Integrity constraint
id	int		Primary key, Not null
role	varchar	20	Not null

Таблиця 3.4.

Відношення сутності `User`

Column name	Data type	Length	Integrity constraint
id	bigint		Primary key, Not null
email	varchar	50	Unique, Not null
first_name	varchar	20	Not null
last_name	varchar	20	Not null
password	varchar	120	Not null
username	varchar	20	Not null

Таблиця 3.5.

Відношення сутності `Manager`

Column name	Data type	Length	Integrity constraint
id	bigint		Primary key, Not null
company_id	bigint		Foreign key references to the `id` of table `company`, Not null
user_id	bigint		Foreign key references to the `id` of table `user`, Not null

Таблиця 3.6.

Відношення сутності `Driver`

Column name	Data type	Length	Integrity constraint
id	bigint		Primary key, Not null
license_number	varchar	50	Not null
company_id	bigint		Foreign key references to the `id` of table `company`, Not null
user_id	bigint		Foreign key references to the `id` of table `user`, Not null

Таблиця 3.7.

Відношення сутності `Customer`

Column name	Data type	Length	Integrity constraint
id	bigint		Primary key, Not null
user_id	bigint		Foreign key references `user_id` of table User, Not null

Таблиця 3.8.

Відношення сутності `Company`

Column name	Data type	Length	Integrity constraint
id	bigint		Primary key, Not null
name	varchar	255	Not null

Таблиця 3.9.

Відношення сутності `Status`

Таблиця 3.10.

Відношення сутності `Driver_Categories`

Column name	Data type	Length	Integrity constraint
driver_id	bigint		Foreign Key references column `id` of the table `driver`, Not null, Unique
category_id	bigint		Foreign Key references column `id` of the table `license_category`, Not null, Unique

Таблиця 3.11.

Відношення сутності `License_Categories`

Column name	Data type	Length	Integrity constraint
id	int		Primary key, Not null
category	varchar	2	Unique , Not null

Таблиця 3.12.

Відношення сутності `Vehicle`

Column name	Data type	Length	Integrity constraint
id	bigint		Primary key, Not null
brand	varchar	255	Unique , Not null
max_weight	bigint		Not null
number	varchar	255	Unique, Not null
required_category_id	int		Foreign key references table `license_category` column `Id`
company_id	bigint		Foreign key references table `company` column `id`

Таблиця 3.13.

Відношення сутності `Cargo`

Column name	Data type	Length	Integrity constraint
id	bigint		Primary key, Not null
name	varchar	255	Unique , Not null
price	decimal	19, 2	Not null
company_id	bigint	255	Foreign key references table `company` column `id`, Not null

Таблиця 3.14.

Відношення сутності `Purchase_Order`

Column name	Data type	Length	Integrity constraint
id	bigint		Primary key, Not null
address	varchar	255	Not null
cargo_id	bigint		Foreign key references table `cargo` column `id`, Not null
company_id	bigint		Foreign key references table `company` column `id`, Not null
customer_id	bigint		Foreign key references table `customer` column `id`, Not null
driver_id	bigint		Foreign key references table `driver` column `id`, Not null
status_id	int		Foreign key references table `status` column `id`, Not null

3.5. Схема розгортання

Схема розгортання наведена на (рис. 3.4.).

					<i>123.KI-41.3</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		38

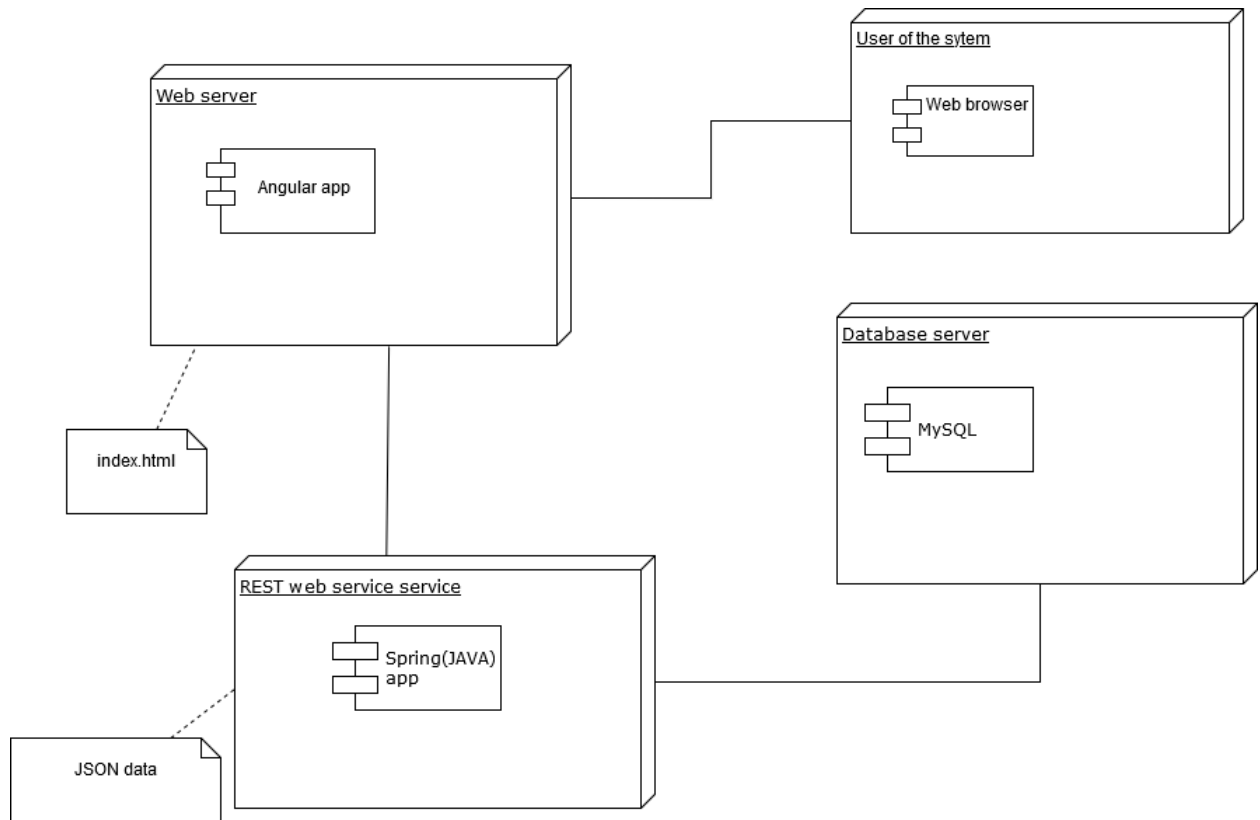


Рис. 3.4. Схема розгортання

с. Висновок

У цій главі було описано архітектуру та технології розробки, використані для розробки прототипу системи. Схема бази даних була охарактеризована, а структура класу була показана на діаграмі класу.

РОЗДІЛ 4. ПРОТОТИП СИСТЕМИ

4.1. Використовувані технології

Технології, які використовуються при розробці прототипу, включають сучасні рішення для зручної розробки програмного забезпечення

Як сховище даних використовується система керування реляційною базою даних MySQL. Він безкоштовно поширюється і простий у використанні. База даних завантажується як образ Docker і розгортається в контейнері Docker.

Docker — це програмне забезпечення для віртуалізації та упаковки програмних компонентів, які називаються контейнерами. Контейнери Docker забезпечують простий спосіб розгортання різних програмних компонентів на різних машинах за кілька кліків.

Для серверної частини використовується мова програмування Java. Для спрощення розробки та зменшення складності коду використовувався Spring Framework, написаний на Java. Spring Framework дозволяє використовувати принцип впровадження залежностей під час розробки таких серверних систем. Цей підхід запобігає дублюванню коду та непотрібній ініціалізації об'єктів через створення служб і компонентів без стану як одиночних компонентів.

Для зручності використання бази даних використовується Spring Data як частина Spring Framework. Цей компонент забезпечує розширення над базою даних, що дозволяє працювати з сутностями бази даних, як із простими об'єктами. Spring Data використовує Hibernate ORM для керування станом сутностей бази даних і створення таблиць бази даних під час запуску програми або просто оновлення існуючої.

					123.KI-41.3	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		40

Для побудови та розгортання бекенд-частини системи використовується інструмент Gradle. Gradle в інструменті збирання, який керує необхідними залежностями та виконує завдання компіляції та збирання для створення виконуваного файлу jar.

Крім того, компонент Spring Security використовується для керування ролями користувачів і забезпечення авторизації та автентифікації користувачів системи.

Для розробки інтерфейсної частини прототипу використовувався Angular Framework. Angular надає компонентну розробку широким набором інструментів для створення веб-інтерфейсів.

Для розробки бізнес-логіки компонентів UI використовується мова програмування TypeScript. TypeScript — це суворо типізована мова, яка працює як розширення над JavaScript.

Для компонентів Angular потрібні класи, написані на TypeScript і шаблонах HTML, які динамічно вставляються на веб-сторінку під час виконання відповідно до вибраного маршруту.

Для стилізації веб-інтерфейсу використовувався фреймворк Bootstrap і CSS. Це дозволяє створити гарний стиль з мінімумом зусиль.

Система Angular побудована за допомогою спеціального інструменту Angular CLI для розгортання програми.

4.2. Приклади роботи

Коли користувач переходить на веб-сайт, з'являється домашня сторінка (рис. 4.1.). Три кнопки: Home, Sign Up і Login відображаються, коли користувач не ввійшов.

Користувач може перейти до форми реєстрації або входу, просто натиснувши навігаційні кнопки. Коли користувач обирає одну з перемикачів для вибору ролі, яку потрібно зареєструвати для менеджера та водія, з'являються додаткові поля (рис. 4.2. а, b, c), для клієнта такі поля відображаються (рис. 4.3.).

					123.KI-41.3	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		41



Рис. 4.1. Домашня сторінка

a)

Username:

Email:

First name:

Last name:

Password:

Signup as:
 Customer
 Driver
 Manager

Company name:

b)

Username:

Email:

First name:

Last name:

Password:

Signup as:
 Customer
 Driver
 Manager

Driver license:

Driver category:

c)

Last name:

Password:

Signup as:
 Customer
 Driver
 Manager

Driver license:

Driver category:

Company name:

Рис. 4.2. а) Форма реєстрації менеджера; б, с) Реєстраційна форма для водія

Username

Email

First name

Last name

Password

Signup as:
 Customer
 Driver
 Manager

Рис. 4.3. Форма реєстрації для клієнта

Менеджер може відкрити спливаюче вікно «Призначити водія» (рис. 4.4.), натиснувши кнопку «Assign driver» у непризначеному порядку.

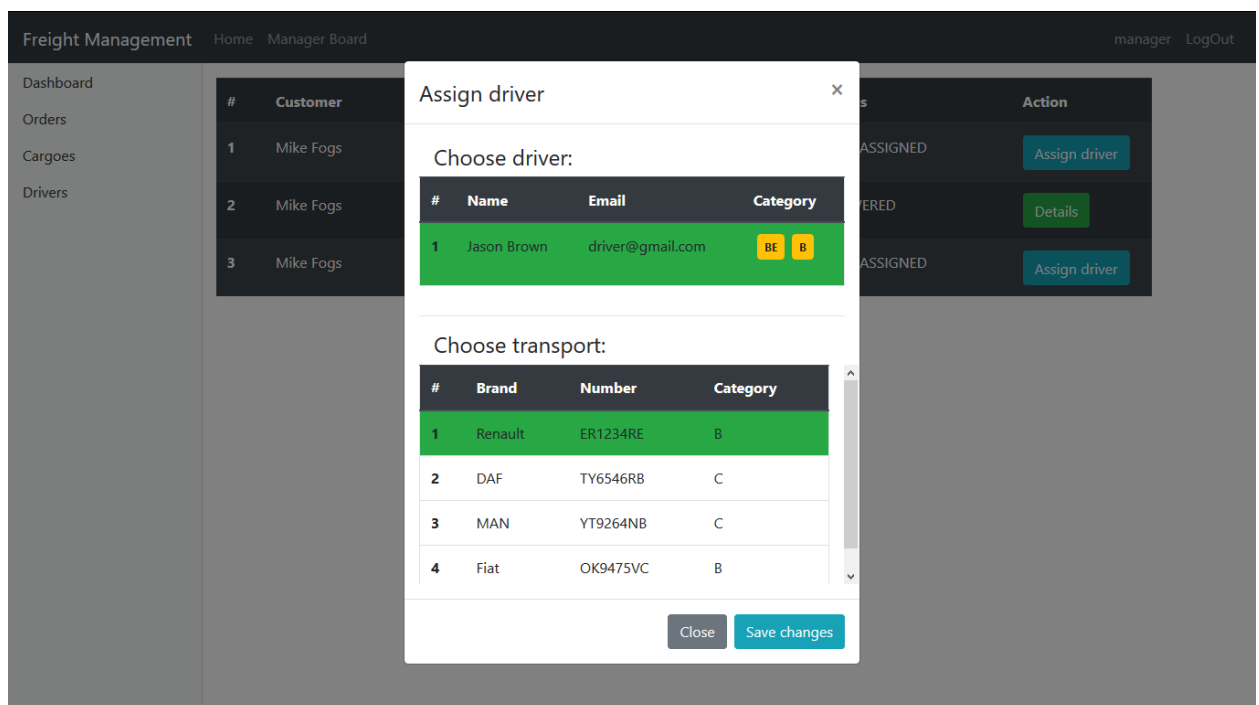


Рис. 4.4. Закріплення водія на замовлення

Після внесення змін оновлюється статус замовлення та змінюється розділ Дія (рис. 4.5.).

#	Customer	Cargo	Price	Address	Status	Action
1	Mike Fogs	Sand	12.4	Kyiv, Peremohy st.	On the way	No action needed
2	Mike Fogs	Sand	12.4	Kyiv	DELIVERED	Details
3	Mike Fogs	Oil	122.4	Poltava, Nauky, 9	NOT_ASSIGNED	Assign driver

Рис. 4.5. Замовлення зі зміненим статусом

Manager can open “Order details” popup (fig. 4.6.) of the order by clicking on details button in order list.

#	Customer	Cargo	Price	Address	Status	Action
1	Mike Fogs	Sand	12.4	Kyiv, Peremohy st.	On the way	No action needed
2	Mike Fogs	Sand	12.4	Kyiv	DELIVERED	Details
3	Mike Fogs	Oil	122.4	Poltava, Nauky, 9	NOT_ASSIGNED	Assign driver

Order details	
Customer info First name: Mike Last name: Fogs Address: Kyiv Email: customer@gmail.com	Cargo info Cargo: Sand Price: 12.4 Company: First Freight
Driver info First name: Jason Last name: Brown Email: driver@gmail.com License: 123QWERTY	Transport info Brand: DAF Number: AA1234AA Required category: B Max weight: 15000
Close	

Рис. 4.6. Деталі замовлення

Менеджер може додавати та видаляти вантаж зі списку. Наприклад, після того, як менеджер заповнить дані для нового вантажу (рис. 4.7.) і натисне кнопку «Додати продукт», у списку з’явиться новий вантаж (рис. 4.8.).

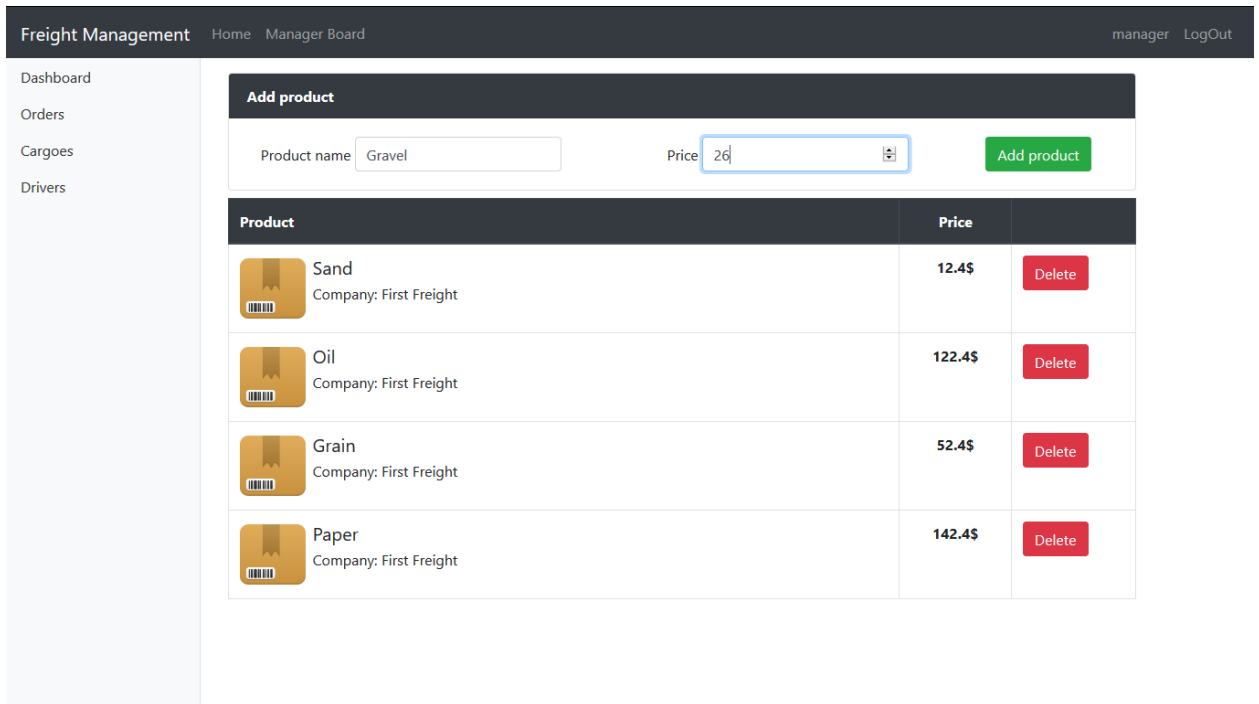


Рис. 4.7. Заповнити дані для нового вантажу

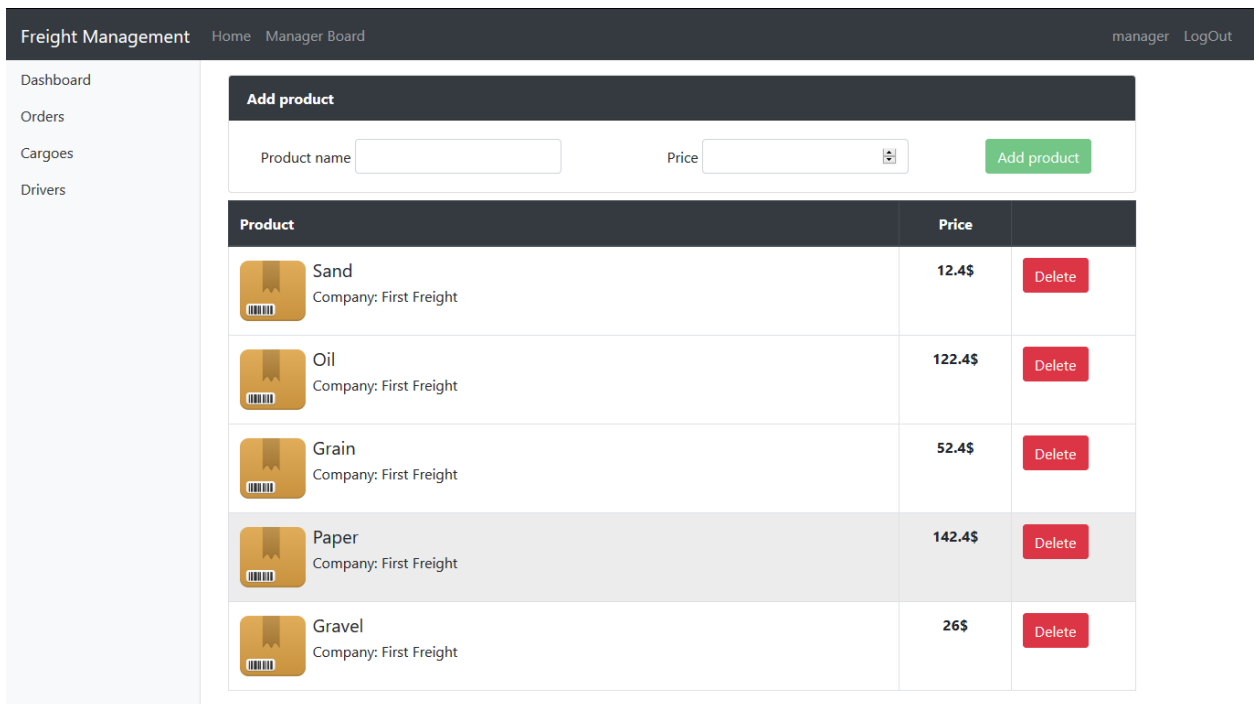


Рис. 4.8. До списку додано новий вантаж

Перелік драйверів та їх статуси менеджер може переглянути на компоненті «Драйвери» (рис. 4.9.).

Freight Management		Home	Manager Board					manager	LogOut
Dashboard	#	First name	Last name	Email	License	License categories	Status		
Orders	1	Arten	Qwert	arten@google.com	RW123123	B BE	Free		
Cargoes	2	Reqw	Tyuf	qweas@google.com	RT3221213	C CE	Free		
Drivers	3	Fasd	Bvdsd	fasf@google.com	FDg3321	B BE	On road		
	4	Avsx	Bdsds	ytir@google.com	AS2123123	B BE C CE	Free		
	5	Udsf	Mhdfd	aseq@google.com	FB65775	B BE	On road		

Мал. 4.9. Список драйверів

d. Висновок

У цьому розділі були описані технології розробки прототипу. Також описано основний процес виконання обов'язків менеджера та додано скріншоти. Менеджер може додавати та видаляти вантаж, перевіряти деталі замовлення та призначати водія до замовлення. Нарешті, менеджер може перевірити інформацію про водіїв та їхні статуси.

ВИСНОВКИ

У цьому дипломному проекті розроблено та проаналізовано програмний прототип системи автоматизованого робочого місця менеджера з вантажних перевезень. Досліджено предметну область і встановлено основні вимоги. Визначено функціональні та нефункціональні вимоги та варіанти їх використання. Побудовано діаграми класів і компонентів. Ці діаграми допомагають у процесі розробки.

Сфера управління вантажними перевезеннями має велике значення в сучасних бізнес-процесах, пов'язаних з вантажними перевезеннями. У результаті проектування системи основні варіанти використання процесу управління вантажами для менеджера вантажів були описані та охоплені кодом програми.

У процесі розробки використовувалися сучасні технології проектування бази даних, серверних сервісів і веб-сторінок. Нові технології були застосовані для безпечного та надійного прототипу програми.

					<i>123.KI-41.3</i>	<i>Арк.</i>
						47
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		

Література

1. What does Freight Management mean?, [Electronic resource] – Mode of access: <https://www.saloodo.com/logistics-dictionary/freight-management/>
2. Logistics, [Electronic resource] – Mode of access: <https://www.investopedia.com/terms/l/logistics.asp>
3. Fraught freight: trade agreements, globalization, and rising global freight transport, [Electronic resource] – Mode of access: <https://www.darrinqualman.com/global-freight-transport/>
4. What is a System Requirements Specification (SRS)?, [Electronic resource] – Mode of access: <https://www.inflectra.com/ideas/topic/requirements-definition.aspx>
5. Systems and software engineering, [Electronic resource] – Mode of access: <https://www.iso.org/obp/ui/#iso:std:iso-iec:25010:ed-1:v1:en>
6. Software Quality Attributes, [Electronic resource] – Mode of access: <https://medium.com/@andreigrignev/non-functional-requirements-quality-attributes-and-better-software-architecture-855425310e60>
7. Cargo types, [Electronic resource] – Mode of access: https://en.wikipedia.org/wiki/List_of_cargo_types#cite_ref-PT_2-0
8. Cargo transportation in Ukraine, [Electronic resource] – Mode of access: <https://open4business.com.ua/volumes-of-cargo-transportation-in-ukraine-in-mln-tons-2/>
9. Functional vs Non-Functional Requirements, [Electronic resource] – Mode of access: <https://qracorp.com/functional-vs-non-functional-requirements/>
10. Component-Based Architecture, [Electronic resource] – Mode of access: https://www.tutorialspoint.com/software_architecture_design/component_based_architecture.htm
11. Rest API, [Electronic resource] – Mode of access: <https://restfulapi.net/https://www.docker.com/>
12. Docker, [Electronic resource] – Mode of access: <https://www.docker.com/>
13. Diagrams design, [Electronic resource] – Mode of access: <https://www.visual-paradigm.com/>

					<i>123.KI-41.3</i>	Арк.
Зм.	Арк.	№ докум.	Підпис	Дата		48

14. Spring Data, [Electronic resource] – Mode of access:
<https://docs.spring.io/spring-data/jpa/docs/current/reference/html/#reference>
15. Angular, [Electronic resource] – Mode of access: <https://angular.io/docs>

					<i>123.KI-41.3</i>	<i>Арк.</i>
<i>Зм.</i>	<i>Арк.</i>	<i>№ докум.</i>	<i>Підпис</i>	<i>Дата</i>		49