

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМ. В. ДАЛЯ  
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ЕЛЕКТРОНІКИ  
КАФЕДРА КОМП'ЮТЕРНИХ НАУК ТА ІНЖЕНЕРІЇ

До захисту допускається  
Завідувач кафедри  
\_\_\_\_\_ І.С. Скарга-Бандурова  
« \_\_\_\_ » \_\_\_\_\_ 20\_\_ р.

**ДИПЛОМНИЙ ПРОЕКТ (РОБОТА) БАКАЛАВРА**

**ПОЯСНЮВАЛЬНА ЗАПИСКА**

НА ТЕМУ:

Інформаційно-комунікаційна система автоматизації обліку  
перевезення вантажів

Освітньо-кваліфікаційний рівень “бакалавр”  
Напрямок підготовки 6.050102 – “Комп’ютерна інженерія”

Керівник проекту:

\_\_\_\_\_ (підпис)

Сафонова С.О.

\_\_\_\_\_ (ініціали, прізвище)

Консультант з охорони праці:

\_\_\_\_\_ (підпис)

Критська Я.О.

\_\_\_\_\_ (ініціали, прізвище)

Здобувач вищої освіти:

\_\_\_\_\_ (підпис)

Іконніков Д.Ю.

\_\_\_\_\_ (ініціали, прізвище)

Група:

\_\_\_\_\_ КІ-14ад

Сєвєродонецьк 2018

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Факультет Інформаційних технологій та електроніки

Кафедра Комп'ютерних наук та інженерії

Освітньо-кваліфікаційний рівень бакалавр

Напрямок підготовки 6.050102 Комп'ютерна інженерія

(шифр і назва)

Спеціальність \_\_\_\_\_

(шифр і назва)

**ЗАТВЕРДЖУЮ:**

Завідувач кафедри \_\_\_\_\_

І.С. Скарга-Бандурова

« \_\_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ р.

**З А В Д А Н Н Я  
НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) БАКАЛАВРА**

Іконнікову Дмитру Юрійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи Інформаційно-комунікаційна система автоматизації  
обліку перевезення вантажів

керівник проекту  
(роботи)

Сафонова С.О., к.т.н., доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від "14" травня 2018 р. № 117/48

2. Термін подання студентом роботи 15.06.2018

3. Вихідні дані до роботи матеріали переддипломної практики

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) Аналіз предметної області і постановка задачі. Моделювання даних предметної області. Вибір і обґрунтування програмних засобів.

Вирішення задачі вибору маршруту. Проектування компонентів комплексу програмного забезпечення. Керівництво користувача. Охорона праці та безпека в надзвичайних ситуаціях.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)  
Електронні плакати

## 6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	ст.викл. Критська Я.О.		

7. Дата видачі завдання 14.05.2018

Керівник

\_\_\_\_\_ (підпис)

Завдання прийняв до виконання

\_\_\_\_\_ (підпис)

## КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту ( роботи )	Примітка
1	Огляд літератури з теми ДП і постановка задачі	14.05.18-15.05.18	
2	Моделювання даних предметної області	16.05.18-19.05.18	
3	Розробка архітектури комплексу програмного забезпечення, вибір технологій для реалізації та методології розробки	20.05.18-23.05.18	
4	Вирішення задачі вибору маршруту	24.05.18-27.05.18	
5	Проектування компонентів комплексу програмного забезпечення	28.05.18-03.06.18	
6	Розробка розділу охорона праці	04.06.18-08.06.18	
7	Оформлення електронних плакатів	09.06.18-10.06.18	
8	Оформлення пояснювальної записки	11.06.18-15.06.18	

Здобувач вищої освіти

\_\_\_\_\_ ( підпис )

Іконніков Д.Ю.

\_\_\_\_\_ (прізвище та ініціали)

Керівник

\_\_\_\_\_ ( підпис )

Сафонова С.О.

\_\_\_\_\_ (прізвище та ініціали)

## РЕФЕРАТ

Пояснювальна записка до дипломного проекту (роботи) бакалавра: 65 с., 14 рис., 2 табл., 20 бібліографічних джерел посилань, 1 додаток.

Об'єкт розробки: задачі вибору ефективних маршрутів і проектів планів перевезень.

Мета роботи: розробка інформаційної системи для автоматизації транспортування вантажів.

В проекті виконано:

1. Проведено аналіз предметної області, сформульована постановка задачі.
2. Моделювання даних предметної області.
3. Розробка архітектури комплексу програмного забезпечення.
4. Вибір технологій для реалізації та методології розробки.
5. Вирішення задачі вибору маршруту.
6. Проектування компонентів комплексу програмного забезпечення .
7. Здійснений аналіз потенційних небезпечних і шкідливих виробничих чинників проєктованого об'єкта, що впливають на персонал.

Отримано наступні результати: система забезпечує графічне відображення маршруту руху та формування і роздрук звітної документації; здійснює контроль за дотриманням термінів виконання замовлень за допомогою надання нотифікації користувачам системи по кожному замовленню.

Практичне значення, галузь застосування роботи: розробка ефективних маршрутів і проектів планів перевезень сприяє своєчасному й безперебійному виконанню поставок продукції та ефективній взаємодії організацій-постачальників, одержувачів та автотранспортних організацій.

**Ключові слова:** ПРЕДМЕТНА ОБЛАСТЬ, JAVA, SPRING MVC, ORACLE PL/SQL, MYBATIS, BOOTSTRAP TWITTER, SERVLET, САЙТ, WEB-СТОРІНКА.

Умови одержання дипломного проекту: СНУ ім. В. Даля, пр. Центральний 59-А., м. Сєвєродонецьк, 93400.

## ЗМІСТ

Скорочення та умовні позначки .....	5
ВСТУП .....	6
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ РОБОТИ .....	8
1.1 Актуальність проблеми розробки систем транспортування вантажів .....	8
1.2 Постановка задачі.....	9
2 МОДЕЛЮВАННЯ ДАНИХ ПРЕДМЕТНОЇ ОБЛАСТІ.....	11
2.1 Створення інформаційної моделі .....	11
2.2 Створення реляційної моделі бази даних (РМБД).....	12
2.4 Архітектура додатку .....	14
2.5 Вибір і обґрунтування програмних засобів.....	16
3 ВИРІШЕННЯ ЗАДАЧІ ВИБОРУ МАРШРУТУ .....	34
3.1 Технічне завдання .....	34
3.2 Обґрунтування вибору алгоритму.....	34
3.3 Virішення задачі.....	37
4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ .....	44
4.1 Аналіз умов праці робочого місця оператора ПК.....	44
4.2 Промислова безпека у виробничому приміщенні НДЛ .....	49
4.3 Виробнича санітарія у приміщенні НДЛ.....	50
4.4 Пожежна безпека виробничого приміщення.....	53
ВИСНОВКИ.....	55
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ.....	56
Додаток А Слайди презентації.....	58

## СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

BOOTSTRAP – це набір інструментів від Twitter (відноситься до класу інструментів: CSS-фреймворк), створений для полегшення розробки web додатків та сайтів

HTTP – (Hypertext Transfer Protocol) протокол передачі гіпертекстових файлів

JAVA – об'єктно-орієнтована, кросплатформена мова програмування

MYBATIS – фреймворк, що автоматизує взаємодію між базами даних і об'єктами Java

PL/SQL – (Procedure Language + Structured Query Language) мова програмування, яка використовується для доступу до баз даних Oracle

SERVLET – стандартизований API для створення динамічного контенту до веб-сервера, використовуючи платформу Java

SPRING MVC – (Spring Model View Controller) архітектурний шаблон, який використовується під час проектування та розробки програмного забезпечення

БД – база даних

ПО – предметна область

СУБД – система управління базами даних

## ВСТУП

Будь-який бізнес може розвиватися лише за умови автоматизації процесів. А будь-яка автоматизація починається зі створення чіткої структури та схеми роботи організації, в якій кожен співробітник перебуває на своєму місці, а його навантаження розподілене максимально ефективно. Керівник в будь-який момент зможе проконтролювати роботу кожного підлеглого або відділу, побачити ступінь виконання завдань і проектів. З таких ланок складається робота будь-якої організації. Ефективна автоматизація бізнесу - це коли всі бізнес-процеси, які можуть бути автоматизовані, передоручені автоматично, а людина виконує тільки ту частину роботи, яку не можна їй довірити. Це можна порівняти з будівництвом будинку: навіщо наймати сотню землекопів, якщо з цієї ж роботою набагато краще впорається екскаватор. І за цим принципом працює будь-який бізнес.

Багатьох проблем вдалося б уникнути і вирішити багато задач легше, якби спочатку було зрозуміло, хто відповідає за ту чи іншу задачу, кому можна задати по ній питання. Адже дуже часто співробітники забувають про свої доручення або не розуміють, хто відповідає за те чи інше завдання. Вирішити дані проблеми допомагає планування. А точніше, система, що об'єднує всі завдання та комунікації по проектах. Таке завдання буде складно не помітити або перекласти відповідальність на колегу, а всі питання можна вирішити безпосередньо в програмі. І завжди можна проконтролювати хід виконання роботи.

Необхідність вирішення великої кількості проблем, поставлених сьогодні перед світовим співтовариством, привело до створення великих комплексних систем. Менеджмент проектів був розроблений для більш оптимального управління задачами та проектами. Комплексні проблеми

вимагають об'єднаних зусиль багатьох галузей. Управління проектами забезпечує саме такий підхід до вирішення складних проблем.

На сьогодні прикладів застосування точних методів для вирішення задач планування розвізних маршрутів невідомо, проте є приклади розробки ефективних алгоритмів, що базуються на модифікації точних та наближених методів.

Основою інформаційних систем є бази даних, які забезпечують зберігання інформації, а також зручний і швидкий доступ до даних. База даних є комп'ютеризованою системою, основне призначення якої зберігати інформацію, надаючи користувачам засоби її витягання і модифікації. До інформації може відноситися все, що заслуговує на увагу окремого користувача або організації, що використовує систему. В даний час бази даних складають основу комп'ютерного забезпечення інформаційних процесів, що входять практично у всі сфери людської діяльності. Процеси обробки інформації мають загальну природу і спираються на опис фрагментів реальності, виражений у вигляді сукупності взаємозв'язаних даних. Бази даних є ефективним засобом представлення структур даних і маніпулювання ними.

Метою даної роботи є розробка інформаційної системи для автоматизації транспортування вантажів. База даних повина містити інформацію про замовлення, строк виконання замовлення, адресу, маршрут доставки та ін.

Реалізація бази даних повина здійснюватися засобами Oracle, а створення клієнтського додатка в середовищі Java з використанням Spring, для здійснення взаємодії кінцевого користувача зі створеною базою даних.



# 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ РОБОТИ

## 1.1 Актуальність проблеми розробки систем транспортування вантажів

На сьогоднішній день маршрутизація перевезень являється найбільш досконалим способом організації матеріалопотоків вантажів, що суттєво впливає на прискорення обігу автомобіля при раціональному і ефективному його використанні та на вартість перевезень. Створення маршрутів дає змогу визначити потрібну кількість рухомого складу, котрий використовується при перевезеннях, сприяє скороченню простою автомобілів під час навантаження та розвантаження, ефективному використанню рухомого складу.

Проблема вдосконалення методів маршрутизації дрібнопартійних перевезень у транспортній мережі міст є актуальною з огляду на наступні фактори: значне загострення конкуренції на ринку автотранспортних послуг, перехід на якісно нові стандарти виробництва, потреба у впровадженні нових технологій транспортного процесу, організація дрібнопартійних перевезень в транспортних системах міст пов'язана з аналізом великих масивів даних (число постачальників, число перевізників, число вантажоодержувачів, кількість і вантажопідйомність автомобілів), при цьому планування розвізних маршрутів зв'язане з необхідністю врахування великої кількості технологічних обмежень (обмеження по швидкості, напрямку руху, за часом та ін.).

Таким чином, розробка ефективних маршрутів і проектів планів перевезень сприяє своєчасному й безперебійному виконанню поставок продукції та ефективній взаємодії організацій-постачальників, одержувачів та автотранспортних організацій.

Таким чином, можна зробити висновок, що задача оптимізації маршрутизації транспортних засобів стає особливо актуальною в умовах даної економічної ситуації.

В останні кілька десятиліть корінні зміни в технології та глобальній конкуренції стали особливо очевидними і відчутними. Бізнес став більш ефективним і динамічним. Здатність успішно і з мінімально можливими витратами здійснювати транспортування дає можливість компанії значно підвищити свою конкурентоспроможність.

## **1.2 Постановка задачі**

Метою даного дипломного проекту є розробка інформаційної системи автоматизації транспортування вантажів. Інформаційна система повинна забезпечувати в клієнтській частині аналіз маршруту транспортування та роздрук і формування звітних шаблонів.

Інформаційна система повинна здійснювати повний контроль за дотриманням термінів виконання замовлень, надавати повну звітність на кожному етапі роботи, кожне завдання повинне бути прив'язане до конкретного працівника. Такий працівник кожного разу при вході на додаток отримує нотифікацію по кожному замовленню, для якого залишилося небагато часу до виконання. Для здійснення поставленого завдання необхідно розробити клієнтський додаток, який зробить процес взаємодії системи та користувача зрозумілим та легким.

Кожна розроблювана система в своєму кінцевому вигляді потрібна бути адаптованою до побажань клієнта та органічно вписатися у бізнес-процеси замовника. Це є фактором успіху кінцевого продукту. Тому вкрай важливо знати та розуміти усі головні бізнес-процеси в роботі фірми-замовника.

Важливим моментом є те, що основною метою інформаційної системи є автоматизація робочого процесу в компанії перевізника.

Схема бізнес-процесу фірми-перевізника наведена на рисунку 1.1.

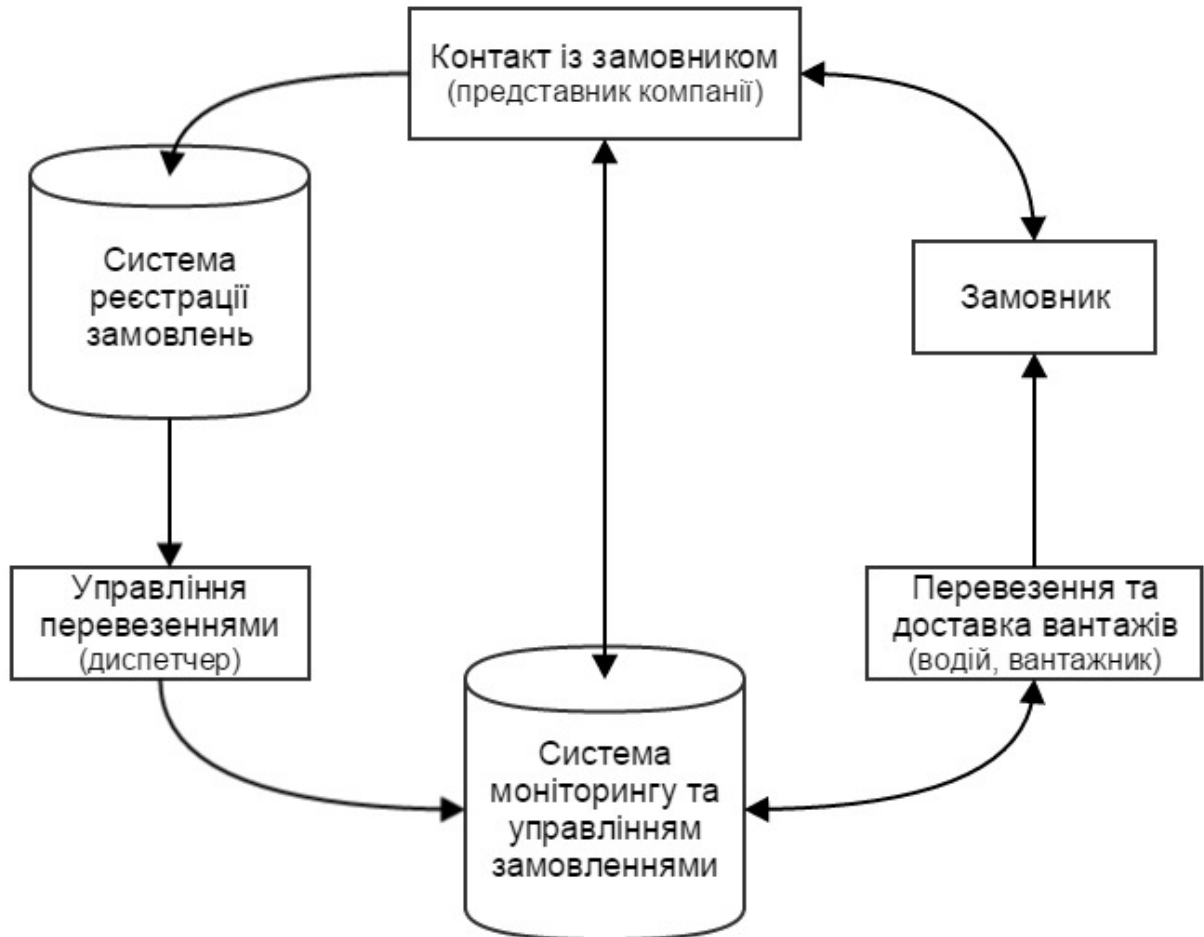


Рисунок 1.1 – Схема бізнес-процесу фірми-перевізника

Таким чином, система повинна реалізувати наступні можливості:

- а) реєстрацію в системі нового замовлення;
- б) реєстрацію в системі користувачів;
- в) створення маршруту в залежності від відстані;
- г) створення маршруту в залежності від кількості товару;
- д) формування звітної документації;
- е) надання нотифікації працівнику, на якого записане замовлення;
- ж) спілкування між співробітниками.

## 2 МОДЕЛЮВАННЯ ДАНИХ ПРЕДМЕТНОЇ ОБЛАСТІ

### 2.1 Створення інформаційної моделі

Модель –знакова або уявна система, що відображає принципи внутрішньої організації або функціонування, певні властивості або характеристики об'єктів дослідження.

За своєю природою відрізняються фізичні моделі (однієї природи з оригіналом), наочно-математичні (еквівалентні оригіналу в сенсі математичного опису), знакові – побудовані у вигляді впорядкованої сукупності символів-знаків (схеми, формули, тексти). Інформаційні моделі – клас знакових моделей. Вони фіксують інформацію про оригінал з метою подальшого використання в комп'ютерних системах.

При інформаційному моделюванні – при створенні і використанні моделей - розрізняють три “світи” або області діяльності.

Перший світ – світ оригіналів. Його модельований фрагмент званий наочною областю. В світі оригіналів відрізняють об'єкти і їх властивості; властивості об'єктів мають значення.

Другий світ – область інформації, тобто існує як відображення реального світу в представленні суб'єктів інформаційної діяльності (розробників, користувачів). Уявні моделі утілюються в концептуальні моделі. Концептуальна модель – це знакова модель, в якій понятійним образам об'єктів зіставляються сукупності символів (знаків). Тут знаки використовуються для узагальнених (абстрактних) понять про об'єкти – так званих концептів; концептам приписуються атрибути, відповідні властивостям об'єктів ПО.

Третій світ – світ даних – це область, в якій використовуються послідовності символів для кодування елементів інформації (значення

атрибутів). Дані – це сукупності знаків, відповідних значенням властивостей об'єктів.

Основними поняттями інформаційної моделі є:

- задача – це процедура обробки або перетворення інформації в системі;
- сховище даних - деякий фізичний об'єкт: комп'ютерний файл або неавтоматизований засіб зберігання даних, наприклад картотека;
- інформаційні потоки – це потоки даних, що переміщуються між зовнішніми об'єктами, завданнями і сховищами даних;
- матеріальні потоки – це потоки матеріальних об'єктів (енергії, матерії, продукції проміжної або кінцевої);
- сховище матеріальних об'єктів.

## 2.2 Створення реляційної моделі бази даних (РМБД)

Система орієнтована на автоматизацію реєстрації та розробки маршруту доставки замовлень на дрібну побутову техніку. На рисунку 2.1 приведена структура бази даних.

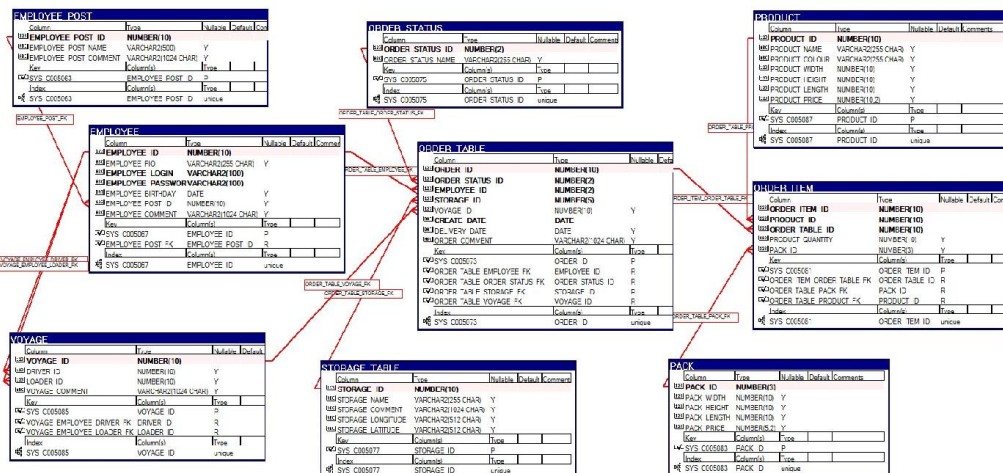


Рисунок 2.1 – Структура БД інформаційної системи

Усі сутності знаходяться у першій нормальній формі, оскільки всі атрибути містять атомарні значення. Поміж атрибутів не міститься груп, що повторюються (кількох значень для кожного екземпляру). Составних первинних ключів немає - отже модель перебуває у другій нормальній формі.

Сутності також знаходяться у третій нормальній формі, оскільки вони знаходяться у другій нормальній формі та ніякий неключовий атрибут не залежить від іншого неключового атрибута (немає зв'язку між не ключовими атрибутами).

У системі передбачено, що голова компанії має право реєструвати в системі нових користувачів, при цьому існує можливість вказати їх посаду, персональні дані та дані аккаунту.

Будь-який працівник має право прийняти замовлення та зареєструвати його в системі.

Працівник може керувати тільки тими замовленнями, які зареєстрував він сам. У системі організований механізм нотифікації для замовлень, терміни поставки яких вже добігають кінця.

Працівник має можливість розробити маршрут руху за кількістю найменувань в замовленні.

Маршрут руху має бути доступний для посад: директор, водій. Також доступ має бути у людини, що обробила замовлення.

Система має формувати товарно-транспортні накладні (ТТН) та інші документи у форматах: \*.rtf, \*.pdf.

### **2.3 Бізнес-правила та ролі користувачів**

Критерієм успіху будь-якого гарного проекту завжди була і буде задоволеність клієнта досягнутим результатом.

Отже, для цього потрібно спочатку прописати технічне завдання і бізнес-правила проекту. Це є відправною точкою для старту аналізу.

Перелік ролей користувачів:

- адміністратор;
- менеджер;
- водій;
- вантажник.

Перелік бізнес-правил:

- адміністратор має повний спектр привілеій;
- аккаунт у інформаційній системі має кожен співробітник;
- кожен співробітник має право зареєструвати замовлення;
- створювати нових користувачів у системі має право лише адміністратор;
- за 2 дні до кінцевого строку доставки замовлення співробітнику, що зареєстрував замовлення, приходить нотифікація;
- у будь-який момент часу користувач з роллю «Водій» має можливість переглянути маршрут доставки замовлення;
- у будь-який момент часу користувач з роллю «Водій» має можливість змодельовати маршрут.

## 2.4 Архітектура додатку

Розроблювана інформаційна система базується на догматиці DDD (Domain Driven Design) та складається з шести взаємопов'язаних модулів (рис. 2.2):

- Business object models. Об'єкти бізнес-логіки, представлені у вигляді класів об'єктно-орієнтованого програмування, що

описують сутності реального світу предметної області та їх поведінку.

- Persistence layer. Модуль з'єднання та взаємодії із базою даних. Включає такі об'єкти: DTO (Data Transfer Object) — описують сутності бази даних, DAO (Data Access Object) — об'єкти оперування даними в БД.
- Translating module. Перетворювач об'єктів, що описують сутності БД (Data transfer object) в об'єкти бізнес-логіки (Business object models) та навпаки.
- Domain service. Прошарок бізнес-логіки, котрий включає функції по взаємодії об'єктів бізнес-логіки та реалізації бізнес-правил. Оперує усіма об'єктами та функціями трьох вище зазначених модулів.
- User interface module. Модуль інтерфейсу користувача. Складається з розмітки web-сторінок, стилів web-сторінок та класів для їх обробки.
- Configuration module. Модуль конфігурації, який відповідає за налаштування запуску і тестування додатку, а також за підключення до бази даних.

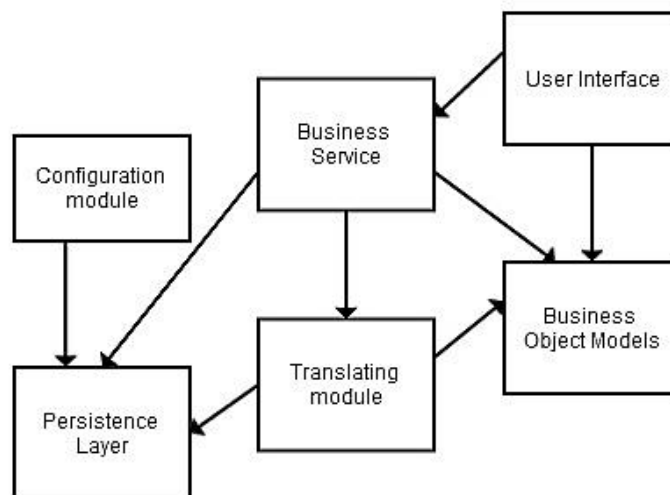


Рисунок 2.2 – Схема архітектури додатку



## 2.5 Вибір і обґрунтування програмних засобів

### 2.5.1 Вибір серверу

Сервер – будь-який окремо взятий комп'ютер в Інтернеті, який дозволяє іншим машинам використовувати себе як «посередника» при передачі даних. Сервер – це саме машина («залізо»), а не логічна частина мережі, вона може мати декілька різних IP-адрес, так що цілком може виглядати з Інтернету як декілька незалежних систем.

Відмінною рисою сервера є те, що він використовує один-єдиний стек TCP/IP, тобто на ньому запущено тільки по одному «екземпляру» драйверів протоколів. Термін «сервер» має декілька визначень – це програма (у термінології TCP – процес), яка обробляє запити клієнтів. Наприклад, додаток, обслуговуючий користувачів WWW, називається web-сервером. Дане визначення ідентичне поняттю «мережевий домен» або «сервіс».

Під час розробки проекту був використан web-сервер Apache. Сьогодні Apache - це повнофункціональний, розширюваний web-сервер, що повністю підтримує протокол HTTP/1.1 і розповсюджується з відкритим вихідним кодом. Сервер може працювати практично на всіх поширених платформах. Існують готові виконувачі файли сервера для Windows NT, Windows 9x, Windows 2003/7 Server XP OS/2, Netware 5.x і декількох UNIX-систем. При цьому він дуже простий в установці і конфігурації.

Apache налаштовується за допомогою текстових конфігураційних файлів. Основні параметри вже налаштовані "за умовчанням" і працюватимуть в більшості випадків. Важливою перевагою є те, що творці активно спілкуються з користувачами і реагують на всі повідомлення про помилки.

Найпростіша функція, яку може виконувати Apache, – стояти на сервері і обслуговувати звичайний HTML-сайт. При отриманні запиту на певну

сторінку сервер відправляє її в відповідь браузеру. Функція, яка закладена в протоколі HTTP/1.1 – аутентифікація користувачів. За допомогою штатних засобів сервера Apache можна розмежувати доступ до певних сторінок сайту для різних користувачів. Це потрібно, наприклад, для того, щоб зробити адміністраторський інтерфейс до сайту.

Користувачі можуть бути розбиті на групи, і для кожної з них можна призначити свої права доступу.

За допомогою сервера Apache можна проводити просту аутентифікацію. Для розділення дизайну і функціональної частини сайту, а також для спрощення зміни статичних об'єктів існує технологія SSI. Вона дозволяє помістити всю інформацію, що повторюється, в один файл (наприклад, `top.inc`), а потім вставляти в сторінки посилання на неї. Потім, якщо знадобиться змінити декілька рядків в цій інформації, то доведеться поміняти їх тільки в одному файлі. Сервер Apache підтримує цю технологію і дозволяє використовувати серверні включення в повному об'ємі.

Сервер Apache підтримує роботу з технологією SSI. Якщо на одному сервері зі встановленою операційною системою сімейства Unix і сервером Apache заведено декілька користувачів, то кожному з них можна створити окрему директорію. Точніше, вона створюватиметься автоматично разом з псевдонімом. Це робиться за допомогою модуля `mod_userdir` і директиви `UserDir`. Так, наприклад, можна папці `public_html` в домашній папці користувача співставити адресу `www.site.ua/~user`. Загалом, так і робиться на серверах більшості сайтів, що надають безкоштовний хостинг. Адміністратор сервера може вирішити або заборонити певним користувачам створювати домашні сторінки, використовувати SSI і інші функції сервера. Повноцінний же хостинг зазвичай передбачає створення окремого віртуального сервера для кожного користувача.

На сервері Apache можна створювати призначені для користувача директорії для web-сайтів. Сервер Apache був одним з перших серверів, які почали підтримувати віртуальні сервера (хости). Ця можливість дозволяє

розміщувати на одному фізичному сервері декілька повноцінних сайтів. У кожного з них може бути свій домен, адміністратор, IP-адрес і так далі.

В даний момент більшість інтернет-сторінок є динамічними. Це означає, що їх зовнішній вигляд і наповнення формується за допомогою програмного скрипта, написаного на одній з "мов" (їх не можна повною мірою назвати мовами, визначення достатнє умовно).

На основі сервера Apache можна створювати не тільки прості сайти, але і ресурси, що вимагають серйозного криптографічного захисту передаваних даних. Спеціально для цього був розроблений протокол SSL/TLS, а його підтримка була вбудована в Apache 2.0. За допомогою спеціального модуля можна здійснювати аутентифікацію на основі іменних сертифікатів, що дозволяє практично напевно гарантувати достовірність користувача.

У останній версії Apache існує модуль для роботи з сильним криптозахистом SSL/TSL.

Сервер Apache може вести протокол всіх дій, що здійснюються з ним. Причому адміністратор може сам вибрати ступінь подробиць протоколу. Протоколи ведуться окремо для помилок, для успішних операцій і для кожного віртуального хосту. Таким чином, Apache є на сьогоднішній день одним з найпоширеніших і непоступливим по функціональних можливостях комерційним web-серверам.

У зв'язку з приведеними вище характеристиками був вибраний web-сервер Apache для реалізації проекту.

### **2.5.2 Вибір мови програмування. Java**

Java – об'єктно-орієнтована мова програмування, випущена компанією Sun Microsystems у 1995 році як основний компонент платформи Java.

Основною властивістю є багато- або кросплатформеність. Кросплатформеність досягається за рахунок використання віртуальної машини JVM, що виконується операційною системою і надає для Java-додатків всі необхідні можливості.

Вихідний код програми на мові Java є текстовими файлами, які можуть бути створені в будь-якому текстовому редакторі або спеціалізованому засобі розробки. Ці файли подаються на вхід Java-компілятора, який транслює їх у спеціальний Java байт-код.

Результат роботи компілятора зберігається в бінарних файлах з розширенням `.class`. Java-додаток, що складається з таких файлів, подається на вхід віртуальній машині, яка починає їх виконувати.

Наступною за важливістю властивістю є об'єктна орієнтованість Java. В Java практично все реалізовано у вигляді об'єктів - потоки виконання (threads) і потоки даних (streams), робота з мережею, робота з зображеннями, з призначенням для користувача інтерфейсом, обробка помилок і т.д. Будь-який додаток на Java - це набір класів, що описують нові типи об'єктів.

В Java застосовується сувора типізація — будь-яка змінна і будь-який вираз має тип, відомий вже на момент компіляції. Такий підхід застосовано для спрощення виявлення проблем, адже компілятор відразу повідомляє про помилки і вказує їх розташування в коді. Пошук виняткових ситуацій (exceptions - так в Java називаються некоректні ситуації) під час виконання програми (runtime) вимагатиме складного тестування [1].

У Java існує всього 8 типів даних, які не є об'єктами. Вони були визначені з самої першої версії і ніколи не змінювалися. Це п'ять цілочисельних типів: `byte`, `short`, `int`, `long`, а також до них відносять символічний `char`. Потім два дробових типу `float` і `double` і булевський тип `boolean` — примітивні типи даних (від англійського `primitive`). Всі інші типи - об'єктні або посилальні (англ. `reference`).

Синтаксис Java створений на основі синтаксису мов C/C++.

Інша важлива властивість Java - легкість в освоєнні і розробці. Автори прагнули позбавити програмістів від найбільш поширених помилок, які часом допускають навіть досвідчені розробники на C/C++. І перше місце тут займає робота з пам'яттю.

У Java з самого початку був введений механізм автоматичного збирання сміття (від англійського *garbage collector*). Припустимо, програма створює деякий об'єкт, працює з ним, а далі настає момент, коли він більше вже не потрібен. Необхідно звільнити займану пам'ять, щоб не заважати операційній системі нормально функціонувати. В C/C++ це необхідно робити явно з програми. При такому підході існує дві небезпеки - або видалити об'єкт, який ще комусь необхідний (і якщо до нього дійсно відбудеться звертання, то виникне помилка), або не видаляти об'єкт, що став непотрібним, а це означає витік пам'яті, тобто програма починає споживати все більшу кількість оперативної пам'яті.

При розробці на Java програміст взагалі не думає про звільнення пам'яті. Віртуальна машина сама підраховує кількість посилань на кожен об'єкт, і якщо воно стає рівним нулю, то такий об'єкт позначається для обробки *garbage collector*. Збирач сміття - це фоновий потік виконання, який регулярно переглядає існуючі об'єкти і видаляє вже не потрібні. З програми ніяк не можна вплинути на роботу *garbage collector*, можна тільки явно ініціювати його черговий прохід з допомогою стандартної функції. Ясно, що це істотно спрощує розробку програм, особливо для починаючих програмістів.

Проте досвідчені розробники були незадоволені тим, що вони не можуть повністю контролювати все, що відбувається з їх системою. Немає точної інформації, коли саме буде видалений об'єкт, що став непотрібним, коли почне працювати (а значить, і займати системні ресурси) потік збирача сміття і т.д. Але при всій повазі до досвіду таких програмістів, необхідно зазначити, що переважна кількість збоїв програм, написаних на C/C++, припадає саме на некоректну роботу з пам'яттю, причому часом це

трапляється навіть із широко поширеними продуктами досить серйозних компаній.

Крім введення `garbage collector`, є і інші кроки для полегшення розробки. Наприклад: відмова від множинного успадкування, спрощення синтаксису та ін.. Можливість створення багатопотокових додатків була реалізована в першій версії Java (дослідження показали, що це дуже зручно для користувачів). Однак те, що створення і підтримка систем дійсно простіше на Java, ніж на C/C++, давно є загальновизнаним фактом. Втім, все-таки ці мови створені для різних цілей, і кожна має свої незаперечні переваги.

Наступна важлива властивість Java - безпека. При роботі будь-якої віртуальної машини Java діє цілий комплекс заходів.

По-перше, це правила роботи з пам'яттю. Вже говорилося, що очищення пам'яті проводиться автоматично. Резервування її також визначається JVM, а не компілятором, або явно з програми, розробник може лише вказати, що він хоче створити ще один новий об'єкт. Вказівники за фізичними адресами відсутні принципово.

По-друге, наявність віртуальної машини-інтерпретатора значно полегшує відсікання небезпечного коду на кожному етапі роботи. Спочатку байт-код завантажується в систему, як правило, у вигляді class-файлів. JVM перевіряє, чи всі вони підкоряються загальним правилам безпеки Java і чи не створені зловмисниками за допомогою якихось інших засобів (і не спотворені при передачі). Під час виконання програми інтерпретатор легко може перевірити кожен дію на допустимість. Можливості класів, які були завантажені з локального диска або по мережі, істотно різняться. Такі вбудовані обмеження є у всіх стандартних бібліотеках Java.

Існує механізм підписання аплетів та інших програм, що завантажуються по мережі. Спеціальний сертифікат гарантує, що користувач отримав код саме в тому вигляді, в якому його випустив виробник. Це дозволяє клієнтові або відмовитися від роботи з додатками ненадійних

виробників, або відразу побачити, що в програму внесені неавторизовані зміни. У гіршому випадку він знає, хто відповідальний за заподіяний збиток.

Сукупність описаних властивостей Java дозволяє стверджувати, що мова досить пристосована для розробки Internet і інтранет (внутрішні мережі корпорацій)-додатків.

Важлива відмінна особливість Java - її динамічність. Мова дуже вдало задумана, в її розвитку беруть участь сотні тисяч розробників і багато великих компаній.

Java-платформа володіє наступними перевагами:

- переносимість або кросплатформеність;
- об'єктна орієнтованість, створена ефективна об'єктна модель;
- звичний синтаксис C/C++;
- вбудована і прозора модель безпеки;
- орієнтація на Internet-завдання, мережеві розподілені додатки;
- динамічність, легкість розвитку і додавання нових можливостей;
- простота освоєння.

### **2.5.3 Вибір СУБД. Oracle**

Не одне підприємство не може обійтися без надійного і зручного місця зберігання інформації. На сьогоднішній момент використовують електронні бази даних. Для проекту використовується Oracle, оскільки це надійна і проста у використанні база даних, окрім цього вона добре функціонує в парі з Java.

Oracle Database – це об'єктно-реляційна система, що підтримує технології, які підтримують об'єктно-орієнтований підхід до створення та експлуатації бази даних.

Ключові переваги об'єктно-реляційної системи Oracle Database над конкурентами на ринку баз зберігання даних (Microsoft SQL Server, MySQL):

- Real Application Cluster (RAC) – дозволяє масштабувати систему при необхідності, керує завантаженістю системи;
- Automatic Storage Management (ASM) – дозволяє автоматично розподіляти дані поміж ресурсами для зберігання, збільшує отказовідказність системи;
- продуктивність;
- порівняно прості засоби розробки;
- великі бази даних. Максимальний розмір одного екземпляру СУБД Oracle дорівнює восьми екзобайтам.

Сучасною формою інформаційних систем є банки даних, що включають в свій склад наступні складові:

- обчислювальну систему (ОС);
- систему управління базою даних;
- одну або декілька баз даних;
- набір прикладних програм (додатків БД).

База даних забезпечує зберігання інформації, а також зручний і швидкий доступ до неї. Вона є сукупністю даних різного характеру, організованих за певними правилами. Інформація в БД повинна бути:

- несуперечливою;
- ненадмірною;
- цілісною.

Система управління базою даних – це сукупність мовних і програмних засобів, призначених для створення, ведення і використання БД. По характеру СУБД розділяють на персональні і багатокористувацькі.

До складу мовних засобів сучасної СУБД входять наступні два типи мов:



- мова опису даних, що призначена для опису логічної структури бази даних (DDL – Data Defenition Language);

- мова маніпулювання даними, що забезпечує основні операції над даними – введення, модифікацію та вибірку (DML – Data Management Language).

Названі мови в різних СУБД можуть мати відмінності. Найбільші поширення набули дві стандартизовані мови: QBE (Query By Example) – мова запитів за зразком і SQL (Structured Query Language) – структурована мова запитів. QBE забезпечує візуальне конструювання запитів до БД і в основному володіє властивостями мови маніпулювання даними. SQL забезпечує керування структурою БД і маніпулювання даними, є стандартним засобом доступу до видалених БД і об'єднує в собі властивості мов обох типів – опису і маніпулювання даними [2].

Для СУБД Oracle була розроблена PL/SQL. PL/SQL – мова програмування, яка використовується для доступу до баз даних Oracle. Сама назва означає англ. Procedural Language (процедурна мова) + SQL (структурована мова запитів). Поява цієї мови є закономірним розвитком засобів обробки даних для баз даних. Як відомо, мова SQL є декларативною мовою програмування (мова програмування 4-го покоління), тобто конструкції цієї мови вказують, що потрібно зробити, а не як. PL/SQL підтримує всі базові програмні конструкції. Синтаксис цієї мови базується на синтаксисі мови програмування Pascal. Ідейно ця мова створена на основі мови програмування Ada. Код програм не чутливий до регістра. Мова тісно інтегрована з мовою SQL [2].

База даних містить дані, використовувані деякою прикладною інформаційною системою. Залежно від виду організації даних розрізняють наступні основні моделі представлення даних в базі:

- ієрархічну;
- мережеву;

- реляційну;
- об'єктно-орієнтовану.

У ієрархічній моделі дані представляються у вигляді деревовидної (ієрархічної) структури. Подібна організація даних зручна для роботи з ієрархічно впорядкованою інформацією, проте при операції даними з складними логічними зв'язками ієрархічна модель виявляється дуже громіздкою.

У мережевій моделі дані організуються у вигляді довільного графа. Недоліком мережевої моделі є жорсткість і висока складність її реалізації.

Крім того, значним недоліком ієрархічної і мережевої моделі є також те, що структура даних задається на етапі проектування БД і може бути змінена на етапі доступу до даних.

У об'єктно-орієнтованій моделі окремі записи БД представляються у вигляді об'єктів. Між записами баз даних і функціями їх обробки встановлюються взаємозв'язки за допомогою механізмів, подібних до відповідних засобів в об'єктно-орієнтованих мовах програмування. Об'єктно-орієнтовані моделі поєднують особливості мережевої і реляційної моделей і використовуються для створення великих БД з складними структурами даних [3].

Реляційна модель отримала свою назву від англійського терміну relation (відношення) і була запропонована в 70-х роках співробітником фірми ІВМ Едгаром Коддом. Реляційна БД є сукупністю таблиць, зв'язаних відносинами. Перевагами реляційної моделі даних є її простота, гнучкість структури, зручність реалізації на комп'ютері, наявність теоретичного опису. Більшість сучасних БД для персональних комп'ютерів є реляційними. При виконанні дипломного проекту використовувалася саме така модель даних.

СУБД Oracle містить наступні типи файлів:

- Керуючі файли (Control files) – містять метадані СУБД. Це є дуже важливі файли для функціонування бази даних.

- Файли даних (Data files) – містять інформацію бази даних.
- Оперативні журнали повторного виконання – необхідні для відновлення бази даних після відмови екземпляру.

Існують інші файли, що формально не входять до бази даних, але робота СУБД без них не є можливою:

- файл параметрів – використовується для опису конфігурації екземпляру;
- файл паролів – дозволяє користувачам підключатися до бази даних для виконання адміністративних задач;
- архівні журнали – містять історію створених журнальних файлів.

Управляючий файл читається при старті екземпляра в ході монтування бази даних. У записах цього файлу зберігаються описи фізичних файлів, що утворюють базу даних. Коли файли додаються до бази даних, в управляючий файл автоматично вносяться зміни.

Місце розташування управляючих файлів задається в параметрі ініціалізації. Для захисту від відмов бази даних, викликаних втратою файлу, що управляє, слід мультиплексувати файл, що управляє, і використати для них, вприняймні, три різні фізичні пристрої. Сервер бази даних Oracle супроводжує копії управляючого файлу, що задані в параметрі ініціалізації.

У оперативні журнальні файли повторного виконання (redo log files) пишуться записи про зміни, що виконуються у базі даних транзакціями і внутрішніми операціями сервера Oracle. Вони дозволяють відновити цілісність бази даних після системних збоїв, викликаних припиненням подачі електроенергії, дискових збоїв і т.п.

Журнальні файли входять в групи журналів. Група містить журнальний файл і його мультиплексовані копії. Кожна така копія - член журнальної групи, і кожна група однозначно визначається її номером. Процес запису даних повторного виконання (log writer - LGWR) пише інформацію з журнального буфера в журнальну групу. Після заповнення файлів журнальної групи або виконання операції переходу з однієї групи в іншу

процес LGWR починає писати в наступну групу. Журнальні групи використовуються «по колу».

База даних розділена на логічні структурні одиниці, що називаються табличними просторами. Вони використовуються для об'єднання логічно пов'язаних структур. Кожна база даних містить один або декілька табличних просторів. Для зберігання інформації створюється один або декілька файлів даних.

Такі об'єкти СУБД, як таблиці і індекси, зберігаються в табличних просторах у вигляді сегментів. Кожен сегмент складається з одного чи більш екстентів. Екстент складається з суміжних блоків даних. Тому кожен екстент може знаходитися тільки в одному файлі даних. Блоки даних – найменші одиниці введення/виведення у базі даних.

Коли база даних просить у операційної системи набір блоків даних, ОС відображає їх у свої реальні блоки на пристрої зберігання. Користувачеві не потрібно знати фізичну адресу інформації у базі даних. Файл даних може бути також розщеплений і зберігатися на декількох дисках із застосуванням зеркалювання.

Розмір блоку даних встановлюється у момент створення СУБД. Стандартний розмір 8к підходить для багатьох баз даних. Якщо СУБД використовується для сховища даних з великими таблицями і індексами, тоді використання блоків більшого розміру може дати вигоду в продуктивності. Якщо БД використовується для транзакційного застосування, в якому читання і записи робляться у випадковому порядку, тоді краще задати менший розмір блоку. Максимальний розмір блоку залежить від ОС.

Основні структури пам'яті, пов'язані з екземпляром Oracle:

- системна глобальна область (System Global Area - SGA), що розділяється усіма серверними і фоновими процесами;
- програмна глобальна область (Program Global Area - PGA), приватна для кожного серверного і фонового процесу. Для кожного процесу виділяється одна PGA;

- системна глобальна область (SGA) - це область пам'яті, що розділяється, в якій містяться дані і інформація екземпляра.

#### 2.5.4 Bootstrap, JavaScript та бібліотека JQuery

Twitter Bootstrap – вільний набір інструментів для створення сайтів і веб-додатків. Включає HTML і CSS шаблони оформлення для типографіки, веб-форм, кнопок, міток, блоків навігації і інших компонентів веб-інтерфейсів, включаючи JavaScript.

Bootstrap використовує найсучасніші напрацювання в області CSS і HTML, тому необхідно бути уважним за підтримки старих браузерів.

Основні переваги Twitter Bootstrap:

- заощадження часу – Bootstrap дозволяє заощадити час і зусилля, використовуючи шаблони дизайну і класи, та сконцентруватися на інших розробках;
- висока швидкість – динамічні макети Bootstrap масштабуються на різні пристрої і дозволи екрану без яких-небудь змін в розмітці;
- гармонійний дизайн – усі компоненти платформи Bootstrap використовують єдиний стиль і шаблони за допомогою центральної бібліотеки. Дизайн і макети веб-сторінок узгоджуються один з одним;
- простота у використанні – платформа проста у використанні, користувач з базовими знаннями HTML і CSS може почати розробку з Twitter Bootstrap;
- сумісність з браузерами – Twitter Bootstrap сумісний з Mozilla Firefox, Google Chrome, Safari, Internet Explorer і Opera;

- відкрите програмне забезпечення - особливість Twitter Bootstrap, яка припускає зручність використання за допомогою відкритості початкових кодів і безкоштовного завантаження.

Основні інструменти Bootstrap:

- сітки – заздалегідь задані розміри колонок, які можна відразу ж використати, наприклад ширина колонки 140px відноситься до класу `.span2`, який можна використати в CSS описі документу;
- шаблони – фіксований або відносний шаблон документу;
- типографіка – описи шрифтів, визначення деяких класів для шрифтів, таких як код, цитати і т.п.;
- медіа – управління зображеннями і відео;
- таблиці – засоби оформлення таблиць, аж до додавання функціональності сортування;
- форми – класи для оформлення форм і деяких подій, що відбуваються з ними;
- навігація – класи оформлення для табів, вкладок, меню і тулбара.

JavaScript – скриптова мова, що найчастіше використовується при створенні сценаріїв поведінки браузера, що вбудовуються у веб-сторінки. Розроблена Бренданом Айком з компанії Netscape мова була включена в браузер Netscape Navigator, починаючи з версії 2.0 в грудні 1995 року. Спочатку мова називалася Mocha, потім вона була перейменована в LiveScript. На синтаксис зробили вплив мови C і Java, і, оскільки технологія Java була у той час дуже модною, LiveScript перейменували в JavaScript, отримавши відповідну ліцензію у Sun. Реалізація компанії Microsoft отримала назву JScript. Internet Explorer підтримує JScript, починаючи з версії 3.0, що випущена в серпні 1996 року. Стандартизація мови асоціацією ECMA ініційована компанією Netscape. Стандартизована версія має назву ECMAScript, описується стандартом ECMA-262. Перша версія даного стандарту приблизно відповідала JavaScript 1.1.

JavaScript в даний момент повністю займає нішу браузерних мов. JavaScript також знаходить застосування як скриптова мова доступу до об'єктів додатків. Платформа Mozilla (XUL/Gecko) використовує JavaScript. Серед сторонніх продуктів, наприклад, Java, починаючи з версії 6, містить вбудований інтерпретатор JavaScript на базі Rhino. Сценарії JavaScript підтримуються в таких додатках Adobe, як Adobe Photoshop, Adobe Dreamweaver, Adobe Illustrator або Adobe InDesign [4].

JavaScript володіє низкою властивостей об'єктно-орієнтованої мови, але завдяки концепції прототипів підтримка об'єктів в ньому відрізняється від традиційних мов ООП. Крім того, JavaScript має ряд властивостей, властивих функціональним мовам, – функції як об'єкти першого рівня, об'єкти як списки, карринг (currying), анонімні функції, замикання (closures) – що додає мові додаткову гнучкість.

JavaScript має С-подібний синтаксис, але в порівнянні з мовою С має наступні корінні відмінності:

- об'єкти з можливістю інтроспективної і динамічної зміни типу через механізм прототипів;
- функції як об'єкти першого класу;
- автоматичне приведення типів;
- автоматична збірка сміття;
- анонімні функції.

JavaScript-код включається в HTML-код сторінки і виконується інтерпретатором, вбудованим в браузер. Для створення кросс-браузерних веб-додатків з використанням JavaScript використовуються спеціальні бібліотеки JavaScript. Вони є збірками JavaScript, що здійснюють підтримку при розробці програм. Серед відомих JavaScript бібліотек виділяють Adobe life, Dojo Toolkit, Ext JS, jQuery, Mootools, Prototype, Qooxdoo.

У даній роботі використовується бібліотека jQuery. jQuery – бібліотека JavaScript, що фокусується на взаємодії JavaScript і HTML. Була опублікована

на комп'ютерній конференції «BarCamp» в Нью-Йорку Джоном Ресигом в 2006 році.

Головні можливості бібліотеки jQuery:

- перехід по дереву DOM, включаючи підтримку XPath як плагіна;
- візуальні ефекти;
- AJAX-доповнення;
- JavaScript-плагіни.

Точно так, як і CSS відокремлює візуалізацію від структури HTML, jQuery відокремлює поведінку від структури HTML. Наприклад, замість прямої вказівки на обробник події натиснення кнопки, керування передається jQuery, що ідентифікує кнопки і потім перетворює його в обробник події кліка. Таке розділення поведінки і структури також називається принципом ненав'язливого JavaScript. Вся робота з jQuery ведеться за допомогою функції \$. Якщо на сайті застосовуються інші JavaScript бібліотеки, де \$ може використовуватися для своїх потреб, то можна використовувати її синонім – jQuery. Роботу з jQuery можна розділити на 2 типи:

- отримання jQuery-об'єкту за допомогою функції `$()` [4]. Наприклад, передавши в неї CSS-селектор, можна отримати jQuery-об'єкт всіх елементів HTML, що потрапляють під критерій і далі працювати з ними за допомогою різних методів jQuery-об'єкту;

- виклик глобальних методів у об'єкту \$, наприклад, зручних ітераторів по масиву.

### 2.5.5 MyBatis

В Java-співтоваристві Hibernate framework де-факто вважається стандартом для зручної роботи з базою даних. Розробнику важко обрати інший фреймворк, тому що часом він не знає про існування альтернатив.



Основна відмінність MyBatis від Hibernate – це те, як проводиться мапінг об'єктів. Hibernate робить мапінг таблиці БД на сутності, даючи нам доступ до даних. Для отримання даних Hibernate генерує SQL запити хороші до пори – до часу, а потім вони з'їдають купу часу, стають громіздкими і не керованими. MyBatis мапиться не на таблиці, а на SQL запити, за формування запитів відповідає розробник і тільки від нього буде залежати, як швидко буде працювати програма.

MyBatis має функціонал, який дозволяє формувати запити SQL динамічно в залежності від параметрів, які були в нього передані. Наприклад, немає необхідності створювати купу SQL на кожен дію (вибірки з однієї таблиці, але за різними параметрами), можна обійтися одним методом, який буде фільтрувати тих же абонентів по декількох колонок або взагалі не буде фільтрувати і поверне всіх в залежності від вхідних даних.

Для динамічного формування SQL запитів в арсеналі MyBatis є досить компонентів для вирішення більшості завдань. MyBatis при розумному використанні може дати відчутний приріст у швидкості роботи програми.

### **2.5.6 Apache FOP**

FOP – це безкоштовний Java додаток, поширюваний через Internet організацією Apache Software. FOP – це Java-додаток, який прочитує дерево об'єктів форматування, після чого перетворить його в документ PDF. Його можна представляти і в декількох інших форматах, включаючи текстовий формат, MIF, PCL, AWT, а також виведення безпосередньо на принтер. FOP приймає дерева FO у вигляді документів XML - FO, які раніше створювалися за допомогою текстового редактора або засобу XSLT. FOP підтримує і безпосередню передачу даних від синтаксичних аналізаторів SAX і DOM без попереднього збереження у вигляді файлу FO.

Поточна версія FOP включає декілька допоміжних бібліотек, до яких відносяться наступні:

- 1) синтаксичний аналізатор XML Xerces(підтримує SAX і DOM);
- 2) XSLT - обробник Xalan;
- 3) SVG - бібліотека w3c.jar, що відповідає кандидатові в рекомендації консорціуму W3C SVG Candidate Recommendation;
- 4) бібліотека для роботи із зображеннями Jini від компанії Sun Microsystems.

## **3 ВИРІШЕННЯ ЗАДАЧІ ВИБОРУ МАРШРУТУ**

### **3.1 Технічне завдання**

Дана транспортна компанія займається перевезенням вантажів з бази збереження техніки на дрібні товарні склади.

Працівники, що розвозять товар, повинні мати змогу цілодобово зайти на сайт компанії з будь-якого пристрою та подивитися інформацію о замовленнях.

Інформація об актуальних замовленнях доступна у вигляді таблиці. При потребі, водій має змогу натиснути кнопку «Вирахувати» та переглянути маршрут доставки товарів. Користувач повинен мати змогу подивитися відстань, що повинен подолати.

Маршрут має бути сформований у вигляді мапи із намальованим маршрутом руху та у текстовому форматі з описанням маршруту.

Рух дозволений виключно через дороги районного значення (через інші склади техніки).

Маршрут, що прокладається системою, має бути сформований по найкоротшому шляху із дотриманням вище зазначених правил і обмежень.

Приоритет доставки замовлень визначається за кількістю одиниць товару, що треба транспортувати на певний склад.

### **3.2 Обґрунтування вибору алгоритму**

У даній роботі вирішується задача вибору маршруту між пунктами із обмеженнями по вибору доріг для доставки. Оскільки всі склади знаходяться

в обласних центрах, а перевізник наклав обмеження стосовно руху тільки по дорогам районного значення, то маємо змогу сформуванати граф. Вершинами цього графу будуть районні склади компанії, а дугами будуть дороги між обласними центрами. Це буде зважений граф, оскільки кожна дуга (дорога) має свою відстань. Граф буде неорієнтованим, оскільки кожна дорога районного значення має полоси в обидва напрямки. На основі цих даних може бути змодельований граф (рис 3.1).

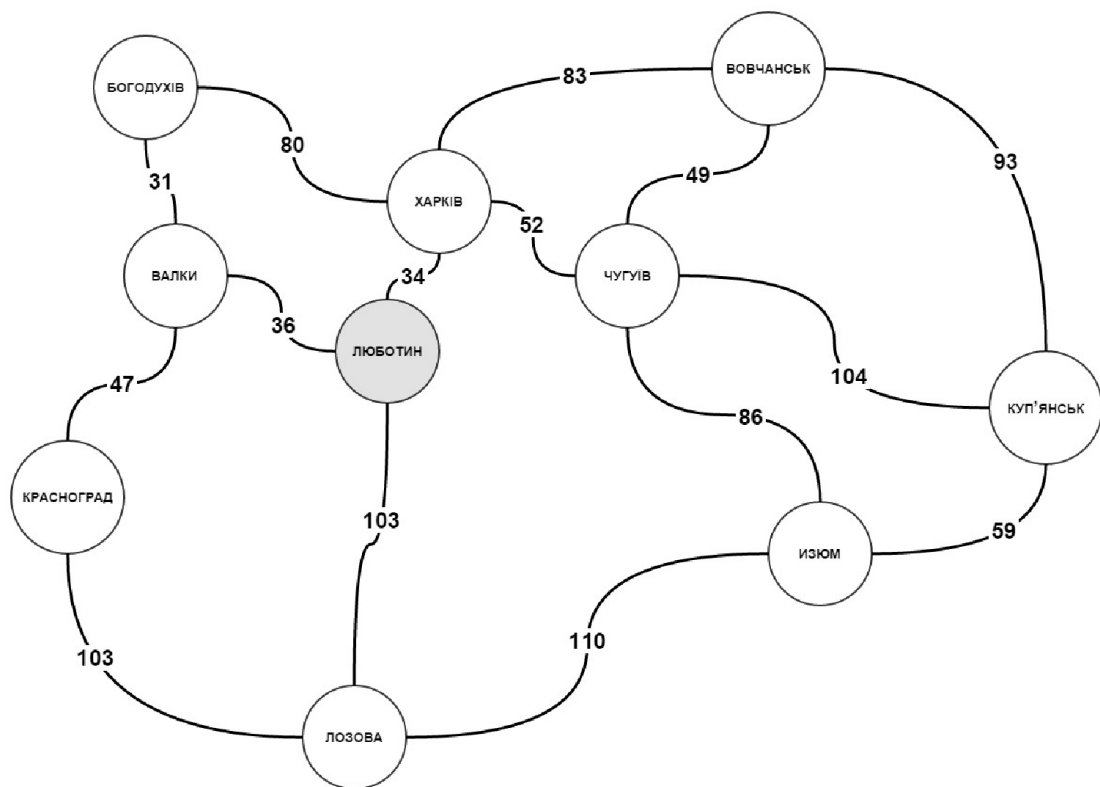


Рисунок 3.1 – Змодельований граф

Як правило, пошук найкоротшого шляху здійснюється за допомогою комбінаторних алгоритмів, тобто алгоритмів спрямованого перебору. Основною перевагою комбінаторних алгоритмів розв'язання завдання пошуку найкоротшого шляху є невисока та заздалегідь відома обчислювальна складність їхньої реалізації. Найефективнішими та найпоширенішими з них є алгоритми Дейкстри (Dijkstra), Беллмана-Форда (Bellman-Ford), Флойда -Уоршела (Floyd-Warshall) та велика кількість різних

їх модифікацій. При цьому перші два алгоритми знаходять найкоротші шляхи від обраного вузла-відправника пакетів до всіх інших вузлів, а третій алгоритм знаходить найкоротші шляхи від всіх вузлів до всіх інших вузлів.

Оскільки алгоритм Флойда – Уоршела знаходить від всіх вузлів до всіх інших вузлів, теоретично він мав би вирішити дану задачу. Єдиним значним мінусом є те, що він генерує велику кількість інформації, що може і не бути необхідною для розв'язку локальної задачі [5].

Для вирішення задачі був обраний алгоритм Дейкстри. Він потребує, щоб довжини всіх дуг були додатні, що в нашому завданні є обов'язковою умовою. Обсяг обчислень у найгіршому разі для цього алгоритму значно менший, ніж в алгоритмі Беллмана — Форда. Основна ідея алгоритму полягає в тому, щоб відшукати найкоротші шляхи в порядку зростання їх довжини. Найкоротшим серед усіх найкоротших шляхів від вузла 1 є шлях, що складається з однієї дуги, що з'єднує вузол 1 з найближчим сусіднім вузлом, оскільки будь-який шлях, який складається з декількох дуг, буде завжди довшим ніж довжина першої дуги, внаслідок припущення про додатність всіх дугових довжин. Наступним найкоротшим серед найкоротших шляхів має бути або шлях з однієї дуги до наступного найближчого сусіда вузла 1, або найкоротший шлях із двох дуг, який проходить через вузол, обраний на першому кроці, тощо.

Формалізований опис алгоритму Дейкстри:

- 1) усім вершинам графу проставляємо мітки, що означають нескінченність. Начальну вершину помічаємо нульовою міткою;
- 2) знаходимо суміжні вершини з даною;
- 3) коректуємо мітки суміжних вершин, знаходячи мінімум з поточної мітки та ваги ребра до суміжної вершини;
- 4) з-поміж множини тимчасових міток обираємо мітку з мінімальним значенням та проставляємо їй постійну мітку;
- 5) переходимо до кроку (2). У якості початкової беремо тільки що знайдену вершину.

Алгоритм закінчує свою роботу, коли усі вершини мають постійні мітки.

### 3.3 Вирішення задачі

Нехай є певна кількість замовлень по доставкам, розміщених у порядку спадання кількості одиниць товару на певний склад.

У даній роботі маємо справу із складною задачею. Для її вирішення застосуємо принцип «розділяй і володай (лат. Divide et impera)». Розділемо нашу задачу на три.

Перша полягає в тому, щоб відсортувати масив замовлень за зазначеним критерієм. Для цього застосуємо алгоритм сортування «бульбашкою». Алгоритм полягає у повторюваних проходах по масиву, що сортується. За кожен прохід елементи послідовно порівнюються попарно і, якщо порядок у парі невірний, виконується обмін елементів. Проходи по масиву повторюються до тих пір, доки на черговому проході не виявиться, що обміни більше не потрібні, що означає — масив вісортовано. При проході алгоритму, елемент, який стоїть не на своєму місці, «спливає» до потрібної позиції як бульбашка у воді, звідси і назва алгоритму.

Оскільки знаємо, що кількості складів прямо пропорційно залежить від кількості районних центрів, а їхня кількість в Україні становить 490, то застосовувати більш ресурсоємкий алгоритм не має сенсу.

Друга задача полягає в тому, що для кожної пари вершин формуємо маршрут за допомогою алгоритма Дейкстри. Із результатів цих ітерацій формуємо кінцевий маршрут.

Третя задача полягає в тому, що отриманий маршрут потрібно представити у вигляді маршруту на мапі. Для вирішення цієї задачі було

обрано Google Maps API. Єдиним недоліком є те, що користувач системи має бути підключений до Інтернету.

### **3.4 Опис розробленого проекту**

#### **3.4.1 Керівництво користувача**

##### **3.4.1.1 Комплект постачання і системні вимоги**

Проект поставляється у вигляді:

- пакету файлів з розширенням «java»;
- файл розгортки з розширенням «war»;
- скрипт для створення бази даних проекту з розширенням «sql».

Специфікація вимог до програмного забезпечення:

- Apache Tomcat v7.0.12 чи новіший;
- будь-яка операційна система зі встановленою JRE 7 (Java Runtime Environment) v7.0.45;
- встановлена Oracle Database 11g;
- браузері: Internet Explorer (9,10,Edge), Chrome (v42.0.2311), Mozilla Firefox(v25.0.1), Safari(v5.1.7).

Для використання сенсорних функцій потрібен пристрій із сенсорним екраном. Доступ до всіх функцій і можливостей також можна отримати за допомогою клавіатури, миші або іншого стандартного чи доступного пристрою вводу. Нові сенсорні функції оптимізовані для використання в ОС Windows 8.

Процесор із тактовою частотою 1 ГГц або більше. Для використання функцій, що працюють через Інтернет, потрібне підключення до Інтернету (за нього може стягуватися плата).

Для апаратного прискорення обробки зображення потрібен відеоадаптер, що підтримує технологію DirectX 10.3, 0 ГБ вільного дискового простору. 1 ГБ оперативної пам'яті (для 32-розрядної версії); 2 ГБ оперативної пам'яті (для 64-розрядної версії).

Корректна робота додатку гарантується тільки за дотримання вище зазначених вимог.

### 3.4.1.2 Інтерфейс сайту

Будь-який користувач, зайшовши на сайт, спочатку потрапляє на сторінку авторизації (рис. 3.1).

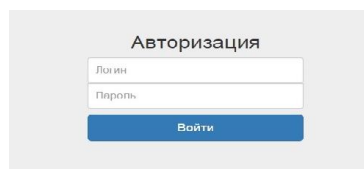


Рисунок 3.1 – Сторінка авторизації

Проходить авторизацію та бачить головну сторінку інформаційної системи (рис.3.2).



Рисунок 3.2 – Головна сторінка



З головної сторінки користувач має змогу переглянути свої особисті дані профілю (рис 3.3).

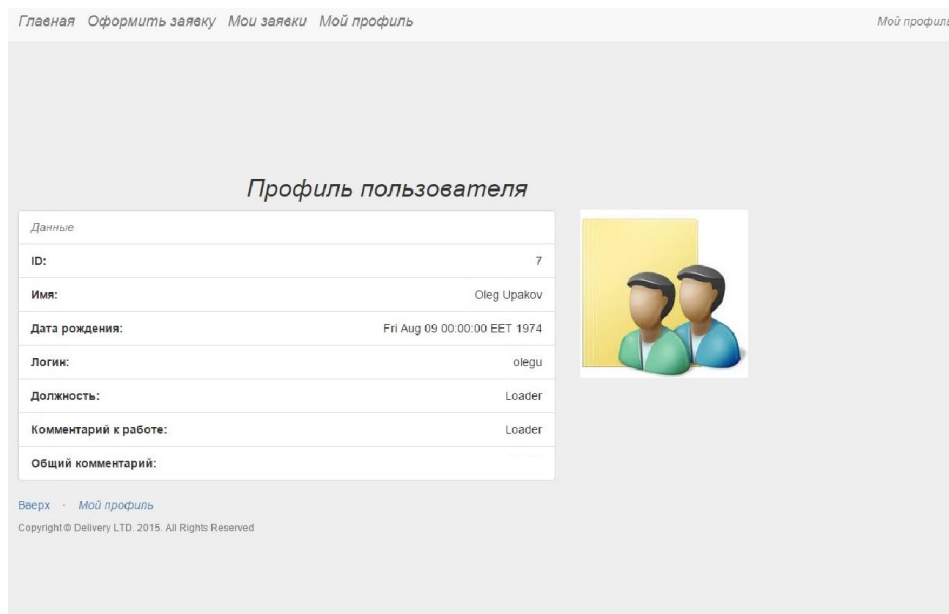


Рисунок 3.3 – Сторінка профілю користувача

Оскільки основними функціями системи є реєстрація замовлення та наглядне представлення маршруту, то найчастіше буде використовуватися сторінка реєстрації замовлення (рис. 3.4).

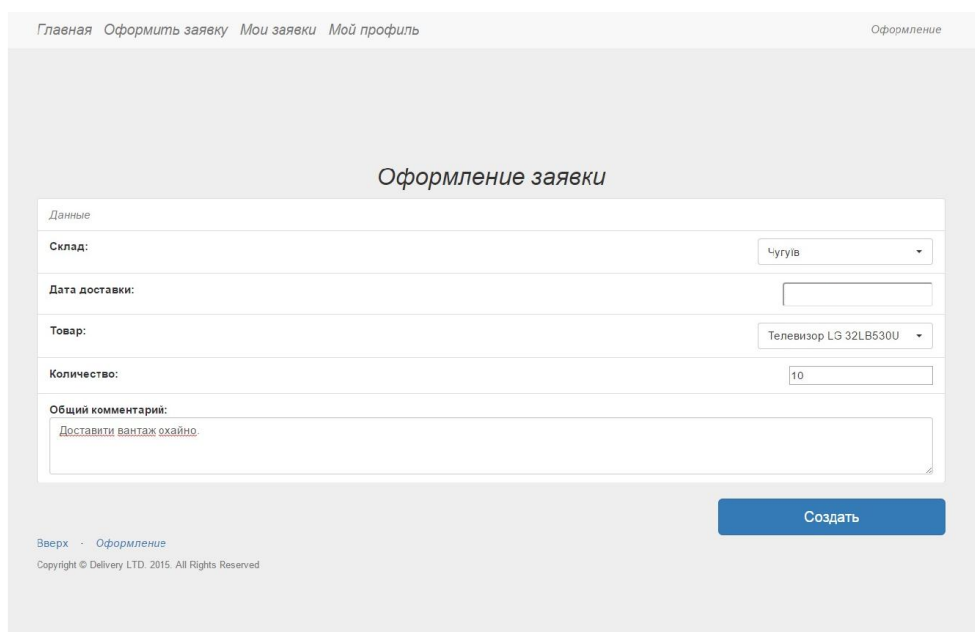


Рисунок 3.4 – Сторінка реєстрації замовлення

Після того, як замовлення зареєстроване, користувач зможе переглянути його з-під свого акаунту у таблиці своїх замовлень. У таблиці замовлень (рис. 3.5) доступна така інформація: найменування товару, колір товару, кількість замовлених одиниць товару, розміри, пункт та дата доставки. Важливою інформацією є статус замовлення.

Главная Оформить заявку Мои заявки Мой профиль Мои заявки

### Мои заявки

Наименование товара	Цвет товара	Размеры: Д x Ш x В	Количество	Цена за шт	Дата создания	Дата доставки	Статус	Склад
Телевизор LG 55LY345C	BLACK	1243.0 x 255.0 x 772.0	100	13799.0	Sun Jun 07 14:53:50 EEST 2015	Wed Jan 07 00:06:00 EET 2015	Новая	Харків
Телевизор LG 32LB530U	BLACK	738.0 x 207.0 x 497.0	150	4099.0	Sun Jun 07 14:54:02 EEST 2015	Wed Jan 07 00:06:00 EET 2015	Новая	Вовчанськ
Телевизор Samsung UE40HG350	BLACK	918.0 x 267.0 x 598.0	200	10599.0	Sun Jun 07 14:54:09 EEST 2015	Wed Jan 07 00:06:00 EET 2015	Новая	Изюм
Телевизор LG 55LY345C	BLACK	1243.0 x 255.0 x 772.0	250	13799.0	Sun Jun 07 14:54:31 EEST 2015	Wed Jan 07 00:06:00 EET 2015	Новая	Красноград

Рисунок 3.5 – Сторінка замовлень

Також система здатна розробляти зручний маршрут прямування для доставки замолень. Кожна поїздка з одного складу на інший наглядно відображена на окремій мапі (рис.3.7-3.9). Кожна мапа має підпис, у якому знаходиться вся важлива інформація. В залежності від того, чи потрібно робити розвантаження вантажу у даній точці маршруту до примітки мапи додається відповідна помітка.

*Путевой лист*

*Маршрут: Люботин, Валки, Красноград, Лозова, Изюм, Вовчанськ, Харків.*

Вычислить

Рисунок 3.6 – Відображення маршруту прямування

Для поліпшення сприйняття інформації про маршрут руху усі ділянки руху розбиті на окремі мапи.

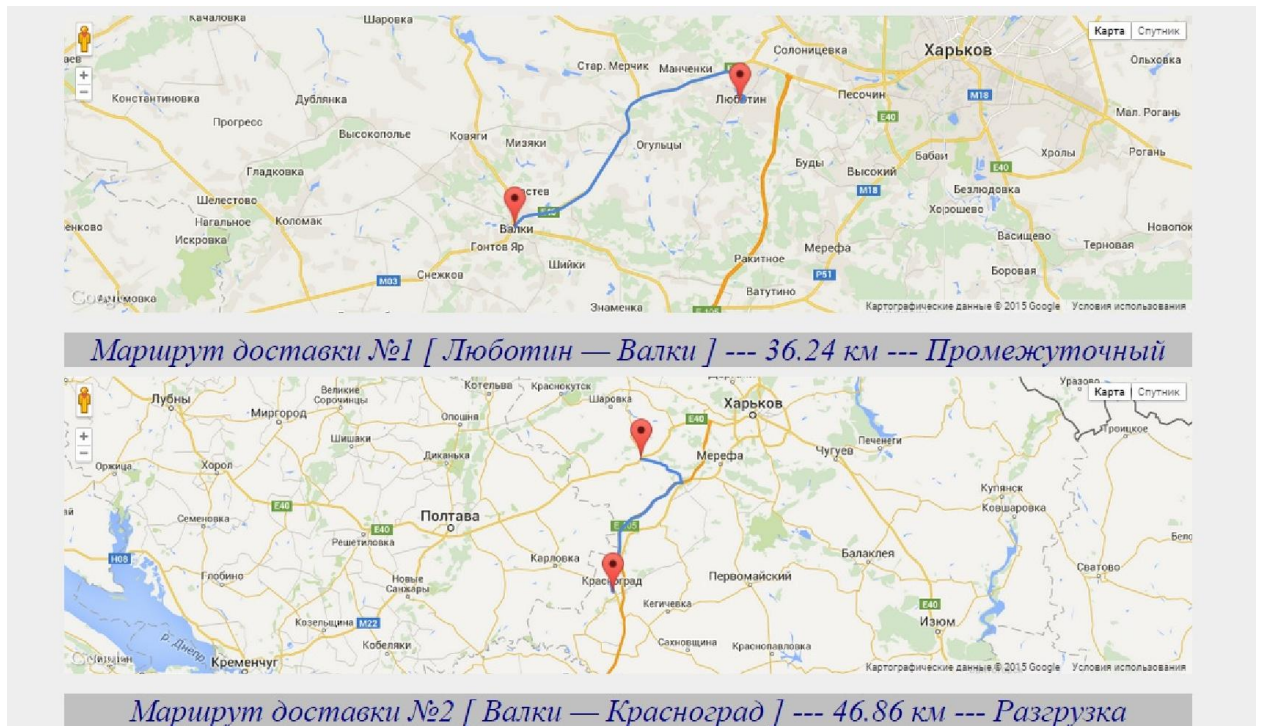


Рисунок 3.7 – Відображення маршруту прямування

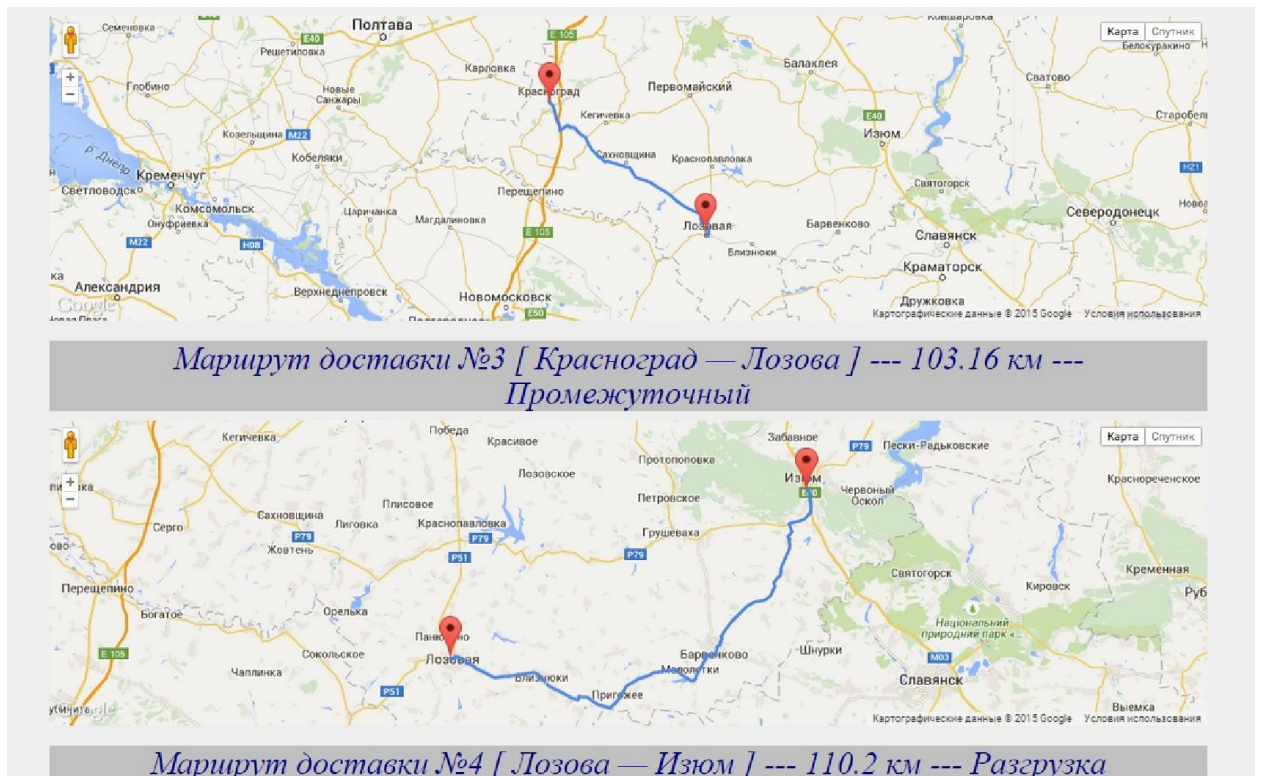


Рисунок 3.8 – Відображення маршруту прямування



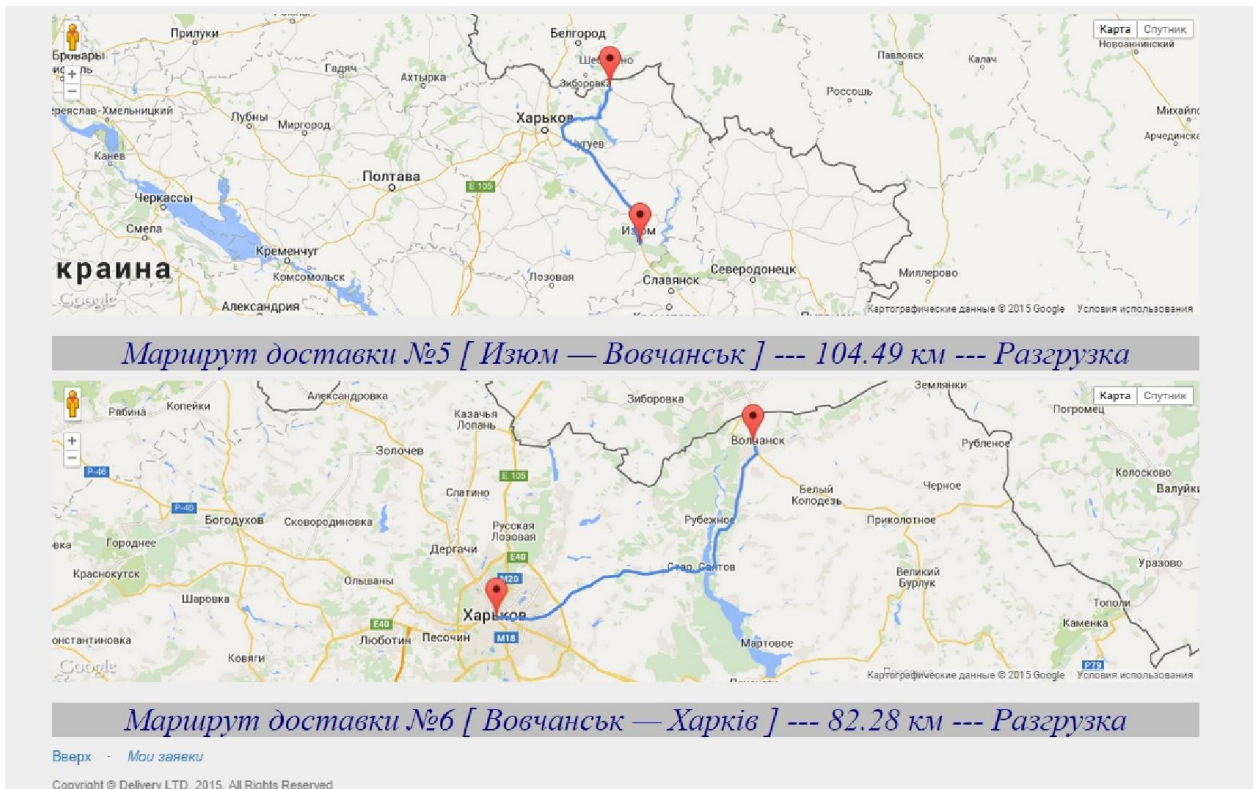


Рисунок 3.9 – Відображення маршруту прямування

Кожна ділянка руху між районними складами розташована на окремій мапі. На мапі доступна наступна інформація:

- 1) номер маршруту доставки;
- 2) початкова та кінцева точка маршруту;
- 3) відстань;
- 4) чи потрібна відгрузка товарів;
- 5) масштаб;
- 6) вичерпна інформація для керування;
- 7) замітки про пункт доставки.

## 4 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

Метою розділу «Охорона праці» є аналіз і розрахунок безпечних умов праці на робочих місцях користувачів ЕОМ, виявлення і оцінка впливу шкідливих виробничих факторів, розроблення заходів щодо промислової безпеки, виробничої санітарії, гігієни праці та пожежної безпеки методом аналізу виділеної системи взаємодії людини, машини і середовища [7].

В даному розділі необхідно дослідити характеристики приміщення, провести аналіз системи «Людина–Машина–Середовище», а саме провести опис шкідливих та небажаних виробничих факторів, що мають місце, їхній вплив на виробниче середовище та людину.

### 4.1 Аналіз умов праці робочого місця оператора ПК

Дипломне дослідження виконувалось в приміщенні науково-дослідної лабораторії (НДЛ). При розробці застосовувалися ПЕОМ.

Приміщення, в якому знаходиться НДЛ, має розміри 2,78x8,24x3,2 м, що становить площу 22,91 м<sup>2</sup>. Працює дві людини. Обладнання: 2 комп'ютери. Перелік обладнання комп'ютера: TFT дисплей 17 дюймів, клавіатура, пристрій типу «мишка», системний блок. На одного працюючого припадає 11,455 м<sup>2</sup> площі і 36,65 м<sup>3</sup> обсягу що відповідає вимогам ДСанПиН 3.3.2-007-98, згідно з яким норма становить 6 м<sup>2</sup> і 20 м<sup>3</sup> відповідно на одне робоче місце з ПК. Електромережа трифазна п'ятипровідна з глухозаземленою нейтраллю, напругою 220В і частотою 50Гц. [7]

У НДЛ створена система «Людина–Машина–Середовище» («Л–М–С»),

елементами якої є:

- а) 2 елементи «людина» – люди, що працюють в НДЛ;
- б) 2 елементи «машина» – ПЕОМ з периферійними пристроями;
- в) «середовище» – виробнича середовище в приміщенні НДЛ;
- г) «предмет праці» – програмне забезпечення у вигляді інформаційної системи.

Кожен елемент «людина» можна умовно розділити на наступні функціональні частини:

Л1 – це людина-оператор, керуючий машиною;

Л2 – це людина, що розглядається з точки зору безпосереднього впливу на навколишнє середовище за рахунок тепло і вологовиділення, споживання кисню.

Л3 – це людина, що розглядається з точки зору його психофізіологічного стану під впливом чинників, які впливають на неї у виробничому процесі.

Кожен елемент «машина» можна умовно розділити на наступні функціональні частини:

М1 – це елемент, який виконує основну технологічну функцію;

М2 – це елемент функції аварійного захисту;

М3 – це елемент впливу на виробниче середовище і людину. [8]

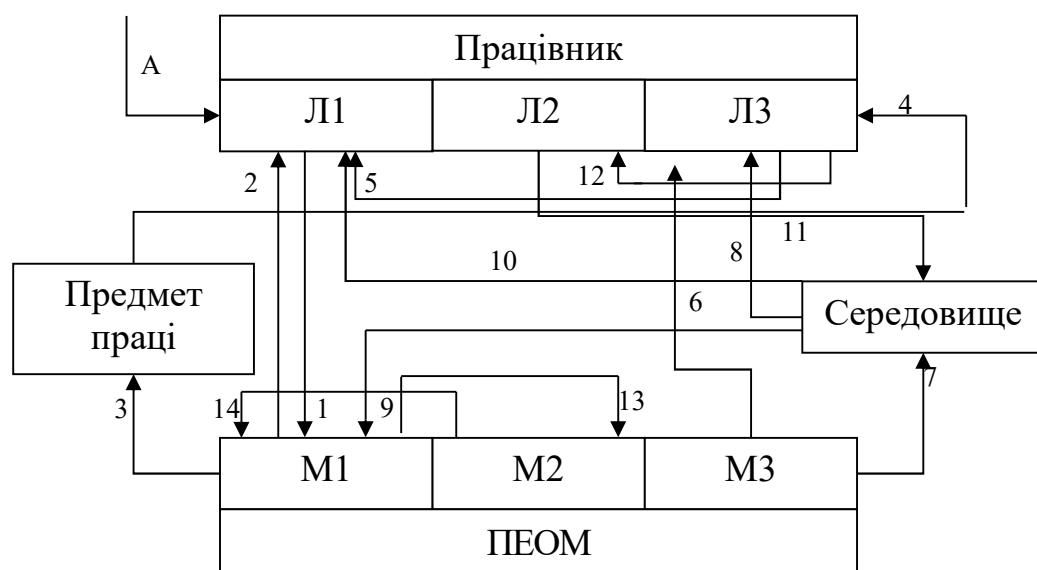


Рисунок 4.1 – Схема системи «Л – М – С»

Характеристика впливів у системі «Людина-Машина-Середовище» представлена в таблиці 4.1.

Таблиця 4.1 – Характеристика впливів у системі «Л–М–С»

Номер зв'язку	Напрямки зв'язку	Зміст зв'язку
1	2	3
1	Л1-М1	Вплив людини на управління «машиною» - комп'ютером, тобто при неправильній експлуатації людина може стати причиною поломки комп'ютера.
2	М1-Л1	Комп'ютер постачає людині необхідну інформацію для обробки
3	М1-ПП	Вплив «машини» на предмет праці, тобто комп'ютер виконує при управлінні людиною необхідні дії при створенні програмного продукту.
4	ПП-Л3	Вплив предмета праці на психофізіологічний стан людини (розумове перенапруження, стреси)
5	Л3-Л1	Вплив стану організму людини на якість роботи людини, тобто втомлена людина не зможе якісно працювати
6	М3-Л3	Психофізіологічний стан людини може постраждати внаслідок неполадок в комп'ютері
7	М3-С	Комп'ютер впливає на виробниче середовище через електромагнітне випромінювання, знижує вологість повітря та підвищує температуру приміщення, є причиною появи «білого шуму» через роботу кулера
8	С-Л3	Вплив виробничого середовища (неправильний мікроклімат в приміщенні, підвищений шум, недостатнє освітлення) може стати причиною різних захворювань людини.

Продовження таблиці 4.1

1	2	3
9	С-М1	При підвищеній температурі можливі перегрів, а також неправильна робота комп'ютера
10	С-Л1	Вплив виробничого середовища (нестача робочого простору, підвищений шум, неправильний мікроклімат) може погано вплинути на якість роботи оператора.
11	Л2-С	Оператор як біологічний об'єкт поглинає кисень із середовища та виділяє тепло
12	Л3-Л2	Психофізіологічний стан людини впливає на ступінь інтенсивності обміну речовин між організмом, середовищем і енерговиділенням людини
13	М1-М2	Інформація, необхідна для виробництва аварійного керуючого впливу, наприклад, перенапруження в мережі
14	М2-М1	Аварійні управляючі дії (при відключенні ПК зупиняється виконання основної технологічної функції)
А	Зовнішня система управління -ЛІ	Керуюча інформація про технологічний процес з зовнішньої системи управління

На підставі наведених зв'язків виділимо небезпечні та шкідливі виробничі фактори (НШВФ) (за ГОСТ 12.0.003-74 [18]) у приміщенні НДЛ:

**Фізичні НШВЧ:**

- Підвищений рівень шуму на робочому місці. Джерело: люди, робота ПЕОМ, принтерів. Можливі наслідки: викликає роздратування, створює незручність мовного спілкування, знижує продуктивність праці може бути причиною зниження слуху.
- Підвищена або знижена температура повітря робочої зони. Джерело: люди, ПЕОМ, джерела світла. Можливі наслідки: дискомфорт для працюючих, розвиток стомлюваності, що зменшує продуктивність праці.
- Підвищена або знижена вологість повітря робочої зони.



- Підвищена або знижена рухомість повітря робочої зони.
- Відсутність або нестача природного світла і штучного освітлення, недостатня освітленість робочої зони. Джерело: малі світлові прорізи, неправильно вибране штучне освітлення. Можливі наслідки: ускладнення розпізнавання зорових образів, що зменшує працездатність, викликаючи стомлення зорових аналізаторів.
- Підвищене значення напруги в електричному ланцюзі, замикання якого може відбутися через тіло людини. Джерело: електрична мережа, електро-обладнання. Можливі наслідки: електротравми та електроудари.
- Підвищене значення електромагнітного випромінювання. Джерело: небезпечний вплив на працюючих можуть надавати електромагнітні поля радіочастот (60 кГц-300 ГГц) і електричні поля промислової частоти (50 Гц). Можливі наслідки: порушення функціонального стану нервової і серцево-судинної систем. [9]

#### Психофізіологічні НШВЧ.

Можливі наслідки: знижується працездатність, розвивається стомлення, емоційні перевантаження в окремих (рідкісних) випадках можуть бути причиною стресу.

- Фізичні (статичні) перевантаження. Джерело: тривале перебування в одній позі.
- Розумова перенапруга. Обробка великої кількості інформації. Джерело: перенапруження зорових аналізаторів.
- Монотонність праці. Виконання одноманітних рухів при роботі з клавіатурою і «мишею».

Хімічні і біологічні фактори відсутні.

Домінуючим НШВЧ є відсутність або нестача штучного освітлення, недостатня освітленість робочої зони.

Штучна освітленість робочого приміщення має допустиме значення згідно з чинними нормами ДБН В.2.5-28-2006 [10]

## 4.2 Промислова безпека у виробничому приміщенні НДЛ

Умови, які можуть створювати підвищену і особливу небезпеку ураження електричним струмом: підвищена вологість та температура, струмопровідний пил, струмопровідні підлоги, можливість одночасного дотику до заземлених металоконструкцій будівлі і металевих поверхонь електроприладів. У зв'язку з використанням в НДЛ електрообладнання необхідно передбачити організаційні і технічні заходи для забезпечення безпеки праці. За ступенем небезпеки ураження електричним струмом, приміщення відноситься до класу приміщень без підвищеної небезпеки, оскільки в приміщенні неможливий одночасний дотик до корпусів ПЕОМ з одного боку і до заземлених металевих конструкцій приміщення (батареї опалення) з іншого боку. Батареї огорожені дерев'яними ґратами. Всі наявні розетки забезпечені попереджувальними написами, мережеві вилки та шнури ізолювані діелектричними матеріалами.

Для захисту людей від ураження електричним струмом слід передбачити подвійну ізоляцію, захисне вимикання. Для захисту людей від ураження електричним струмом передбачена система заземлення типу TN, оскільки використовується електрична мережа напругою до 1000В з глухо заземленою нейтраллю, згідно ДБН В.2.5-28-2006 [13]. Для цього навмисно електрично з'єднані з нульовим захисним провідником мережі корпусу всіх ПЕОМ та принтера, які можуть випадково виявитися під напругою. З'єднання виконано провідником, марка і переріз якого такі ж, як і фазовий провідник мережі. Час відключення пошкодженої ділянки мережі повинен бути не більше 0,1-0,2 с згідно ДБН В.2.5-27-2006 [19]. Для зменшення напруги, доданої до тіла людини при пробі фазы на корпус ПЕОМ або принтера, виконано повторне заземлення нульового проводу.

Контроль стану ізоляції повинен проводитися не рідше одного разу на рік між фазою та нулем між фазами.

Опір повторного заземлення не повинен перевищувати 10 Ом.

Забороняється:

- застосовувати саморобні подовжувачі, які не відповідають до переносних електропроводок;
- застосовувати для опалення нестандартне електронагрівальне обладнання;
- користуватися пошкодженими розетками, коробками, вимикачами та іншими електропристроями.

Згідно з НПАОП 0.00-4.12-05 [16] необхідно проводити вступний, первинний на робочому місці, повторний, а при необхідності - позаплановий і цільовий інструктажі.

### **4.3 Виробнича санітарія у приміщенні НДЛ**

При обслуговуванні, ремонті і наладці ПЕОМ необхідно керуватися «Правилами охорони праці при експлуатації електронно-обчислювальних машин». При організації робочих місць правила охорони праці під час експлуатації електронно-обчислювальних машин в розділі “Вимоги до організації робочого місця користувача ЕОМ” – НПАОП 0.00.-1.28-10 [20] передбачають, що організація робочого місця кожного працівника повинна забезпечувати відповідність усіх елементів робочого місця та їх розташування ергономічним вимогам.

Шум є одним з найбільш поширених у виробництві шкідливим фактором. Вплив шуму на організм людини дуже різноманітний: тривалий вплив шуму з рівнем 100-120дБ на низьких частотах може викликати незворотню втрату слуху, а шум з рівнем 120-140дБ здатний викликати механічні ушкодження органів слуху [6].

Рівень шуму не має перевищувати згідно з ДСН 3.3.6.037-99 [17]:

- 40 дБА – під час робіт, пов'язаних з виробленням концепцій, розробкою нових програм, інших творчих робіт, а також під час навчання;
- 50 дБА – під час робіт, пов'язаних з керівництвом людьми, що виконують розумові роботи (робота керівника);
- 55 дБА – під час висококваліфікованої розумової роботи, що вимагає зосередженості;

Робота користувача комп'ютеризованих систем належить до категорії робіт за енерговитратами організму 1а «легка робота» згідно ГОСТ 12.1.005-81 [15]. Оптимальні параметри мікроклімату у різні періоди року в приміщенні НДЛ згідно ДСН 3.3.6.042-99 [14] наведені у таблиці 4.2.

Таблиця 4.2 – Оптимальні параметри мікроклімату

Період року	Категорія робіт	Температура повітря, град С	Відносна вологість, %	Швидкість руху повітря, м/с, не більше
Холодний	1а	22-24	40-60	0,1
Теплий	1а	23-25	40-60	0,1

Приміщення з комп'ютеризованими робочими місцями повинні мати природне і штучне освітлення згідно з ДБН В.2.5-28-2006 [13]. Природне світло має проникати через бічні світлоотвори, зорієнтовані, як правило, на північ або північний схід. Вікна повинні мати регулюючі пристосування для відкриття, а також жалюзі, штори, зовнішні козирки тощо. Коефіцієнт природної освітленості (КПО) має бути не нижче 1,2 %.

*Розрахунок освітлення.*

Для виробничих та адміністративних приміщень світловий коефіцієнт приймається не менше  $1/8$ , в побутових –  $1/10$ :

$$S_b = \left( \frac{1}{5} \div \frac{1}{10} \right) \cdot S_n, \quad (4.1)$$

де  $S_b$  – площа віконних прорізів,  $m^2$ ;

$S_n$  – площа підлоги,  $m^2$ .

$$S_n = a \cdot b = 2,78 \cdot 8,24 = 22,9072 \text{ м}^2,$$

$$S = 1/8 \cdot 23 = 2,875 \text{ м}^2.$$

Приймаємо 2 вікна площею  $S=1,4 \text{ м}^2$  кожне.

Розрахунок штучного освітлення виробляється по коефіцієнтах використання світлового потоку, яким визначається потік, необхідний для створення заданої освітленості при загальному рівномірному освітленні.

Розрахунок кількості світильників  $n$  виробляється по формулі (4.2):

$$n = \frac{E \cdot S \cdot Z \cdot K}{F \cdot U \cdot M}, \quad (4.2)$$

де  $E$  – нормована освітленість робочої поверхні, визначається нормами – 300 лк;

$S$  – освітлювана площа,  $m^2$ ;  $S = 22,9 \text{ м}^2$ ;

$Z$  – поправочний коефіцієнт світильника ( $Z=1,15$  для ламп розжарювання та ДРЛ;  $Z = 1,1$  для люмінесцентних ламп) приймаємо рівним 1,1;

$K$  – коефіцієнт запасу, що враховує зниження освітленості в процесі експлуатації – 1,5;

$U$  – коефіцієнт використання, залежний від типу світильника, показника індексу приміщення і т.п. – 0,575

$M$  – число люмінесцентних ламп в світильнику – 2;

$F$  – світловий потік лампи – 5400лм (для ЛБ-80).

Підставивши числові значення у формулу (А.2), отримуємо:

$$n = \frac{300 \cdot 22,9 \cdot 1,1 \cdot 1,5}{5400 \cdot 0,575 \cdot 2} \approx 1,83$$

Приймаємо освітлювальну установку, яка складається з 2-х світильників, які

складаються з двох люмінесцентних ламп загальною потужністю 160 Вт, напругою – 220 В. Штучне освітлення приміщення має бути обладнано системою загального рівномірного освітлення. Рівень освітленості на робочому столі в зоні розміщення документів повинен бути в межах 200–500 лк. [ 5]

#### **4.4 Пожежна безпека виробничого приміщення**

За пожежною безпекою приміщення лабораторії відноситься до класу П-Па. Враховуючи специфіку займання ПЕОМ (велику кількість диму), необхідна наявність систем кондиціонування повітря. За вибухопожежною і пожежною небезпекою згідно з НАПБ Б.03.001-2004 [10] дане приміщення відноситься до категорії В, бо застосовуються тверді займисті речовини із температурою займання 61°C. НДЛ розташована в будівлі, виконаної з залізобетонних конструкцій. При роботі тут застосовуються тверді спалимі матеріали, в приміщенні знаходяться тверді і волокнисті горючі речовини. Тому згідно з ДБН В.1.1.7-2002 [12] будівля має 1 ступінь вогнестійкості. Причиною пожежі можуть бути: коротке замикання електропроводки; несправність ПЕОМ та іншого електроустаткування. Для попередження пожежі необхідно проводити ряд технічних та організаційних заходів, спрямованих на дотримання встановленого режиму експлуатації електричної мережі, обладнання та дотримання правил пожежної профілактики.

Відповідно до вимог приміщення повинно бути оснащено:

- 2 димовими сповіщувачами ИП 212-189 (не менше 2 в приміщенні з ПЕОМ), згідно з ДБН В.2.5-56-10 [11];
- протипожежним покривалом.

Необхідно виконувати наступні організаційні заходи: призначити відповідального за пожежну безпеку; включати питання з пожежної профілактики в усі інструктажі з техніки безпеки; контролювати ізоляцію і стан електропроводки та електрообладнання. Згідно проведеного аналізу умов праці та оцінки факторів виробничого середовища і трудового процесу визначимо необхідну кількість та тип вогнегасників в приміщенні:

$l$  – довжина приміщення, 2.78 м;

$w$  – ширина приміщення, 8.24 м;

$h$  – висота приміщення, 3.24 м;

$n$  – кількість працюючих, 2 людини;

$m$  – кількість робочих місць, 2 робочі місця.

Розрахуємо площу робочого приміщення  $S$ :

$$S = 2.78 \times 8.24 = 22.91 \text{ м}^2. \quad (4.1)$$

Приміщення, у яких розташовуються персональні ПК і дисплейні зали, мають бути оснащені переносними вуглекислотними вогнегасниками з розрахунку 1 вогнегасник на 3 комп'ютери, але не менше 2 вогнегасників у приміщенні. [8] Приміщення має бути оснащене переносними вуглекислотними вогнегасниками з розрахунку 1 шт. на кожні 20 м<sup>2</sup> площі з урахуванням гранично допустимих концентрацій (згідно з НАПБ Б.03.001-2004 [10]). Розрахуємо кількість вогнегасників  $N$  типу ВВК-4 для цього приміщення:

$$N = \frac{S}{20} = \frac{22.91}{20} = 1.1455. \quad (4.2)$$

Отже, потрібно не менш ніж два ручних вуглекислотних вогнегасника ВВК-1,4.

## ВИСНОВКИ

В результаті виконання даного дипломного проекту [8] була розроблена інформаційна система автоматизації транспортування вантажів, яка реалізована у вигляді корпоративного веб-сервісу для фіксації клієнтських замовлень та формування маршруту прямування.

Використання сучасних методів управління в організаціях передбачає використання сучасних інформаційних корпоративних веб-сервісів для упорядкування та автоматизації робочого процесу. Була реалізована функція контролю за термінами виконання замовлень, яка надає повну звітність на кожному етапі роботи, а кожне завдання прив'язане до конкретного працівника. Такий працівник кожного разу при вході на додаток отримує нотифікацію по замовленню, для якого залишилося мало часу до виконання.

Були досліджені алгоритми знаходження шляху з однієї вершини до іншої, елементи теорії графів для вирішення поставленої задачі вибору маршруту. Для вирішення поставленої задачі був досліджений та програмно реалізований алгоритм Дейкстри. Отриманий шлях представлений у вигляді маршруту на мапі за допомогою Google Maps API.

Підсумкова програма дозволяє працівникам вести електронну документацію та здійснювати операції, не прив'язуючись до стаціонарного робочого місця.

У розділі «Охорона праці» була складена система «Л–М–С» із зазначенням основних факторів впливу для робочих місць у розглянутій НДЛ. Також був проведений детальний аналіз умов праці з урахуванням діючих норм і нормативів та виконаний аналіз діючих шкідливих та небезпечних факторів в приміщенні НДЛ для заданих умов, які передбачають роботу 2 осіб і експлуатацію 2 ПК. Було розглянуто заходи з промислової безпеки, гігієни праці, виробничої санітарії і пожежної безпеки.



## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Светлов, Н.М. Информационные технологии управления проектами [Текст]: учеб. / Н.М Светлов, Г.Н. Светлова. – М.: ФГОУ ВПО РГАУ–МСХА им. К.А. Тимирязева, 2007. – 207с.
2. Коннор МакДональд, Крістофер Бек. Oracle PL/SQL для профессионалов: практичні рішення. [Текст] / Коннор МакДональд. – Санкт - Петербург: DiaSoft, 2005. – 560с.
3. Бежанишвили Л. Программирование на языке PL/SQL базы данных ORACLE [Текст]/ Л. Бежанишвили. – Тбилиси: Издательский дом "Технический университет", 2009. – 58 с.
4. Бибо, Б. JQuery. Подробное руководство по продвинутому JavaScript [Текст] /Б.Бибо.– Санкт - Петербург: Наука, 2010. – 327с.
5. Иванов Б.М. Дискретная математика алгоритмы и программы [Текст] / Б.М. Иванов. – М.: Известия, 2011. – 521с.
6. Дзюндзюк Б.В. Охрана труда. Сборник задач. [Текст]/ Б.В. Дзюндзюк – Х.: НВП центр ХНУРЕ, 2006. – 242с.
7. Методичні вказівки до виконання розділу дипломного проекту (роботи) бакалавра "Охорона праці та безпека в надзвичайних ситуаціях" (для студентів денної та заочної форм навчання за спеціальностями 122 "Комп'ютерні науки та інформаційні технології", 123 "Комп'ютерна інженерія", 125 "Кібербезпека" (за напрямами 6.050101 "Комп'ютерні науки", 6.050102 "Комп'ютерна інженерія", 6.170101 "Безпека інформаційних і комунікаційних систем")) / Уклад.: Критська Я.О. – Під ред. Скарги-Бандурової І.С. – Северодонецьк: вид-во СНУ ім. В. Даля, 2017. – 71 с.
8. Методичні вказівки до виконання і захисту дипломного проекту (роботи) бакалавра (для студентів денної та заочної форм навчання за спеціальностями 122 "Комп'ютерні науки та інформаційні технології", 123 "Комп'ютерна інженерія", 125 "Кібербезпека" (за напрямами 6.050101 "Комп'ютерні науки", 6.050102 "Комп'ютерна інженерія", 6.170101 "Безпека інформаційних і комунікаційних систем")) / Уклад.: Скарга-Бандурова І.С., Барбарук В.М., Кардашук В.С. – Северодонецьк: СНУ ім. В. Даля, 2017. – 53 с.

9. ДСН 3.3.6.039-99 Санітарні норми виробничої загальної та локальної вібрації
10. НАПБ Б.03.001-2004 Про затвердження Типових норм належності вогнегасників.
11. ДБН В.2.5-56-10 “Системи протипожежного захисту
12. ДБН В.1.1.7-2002 Пожежна безпека об'єктів будівництва
13. ДБН В.2.5-28-2006 Природне і штучне освітлення
14. ДСН 3.3.6.042 99 Санітарні норми мікроклімату виробничих приміщень
15. ГОСТ 12.1.005-81 Система стандартів безпеки труда. Общие санитарно-гигиенические требования к воздуху рабочей зоны.
16. НПАОП 0.00-4.12-05 Положення про порядок проведення навчання і перевірки знань з питань охорони праці
17. ДСН 3.3.6.037-99 Санітарні норми виробничого шуму, ультразвуку та інфразвуку
18. ГОСТ 12.0.003-74 Система стандартів безпеки труда
19. ДБН В.2.5-27-2006. Инженерное оборудование зданий и сооружений. Защитные меры электробезопасности в электроустановках зданий и сооружений
20. НПАОП 0.00-1.28-10 ПРАВИЛА охрaны труда при эксплуатации электронно-вычислительных машин

## Додаток А

### Слайди презентації

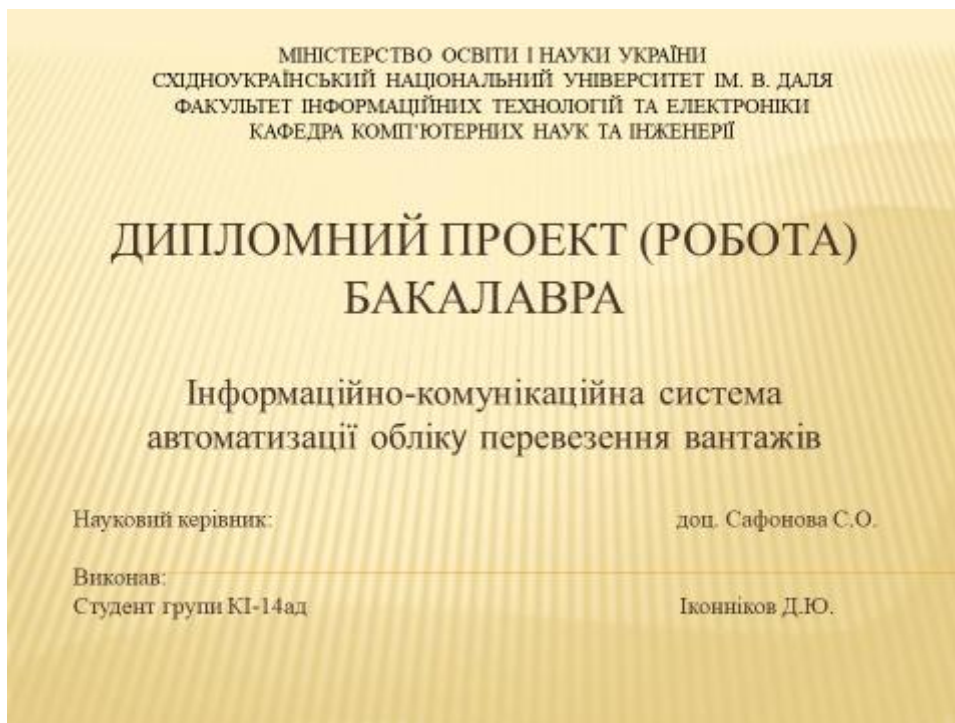


Рисунок А.1 – Слайд 1

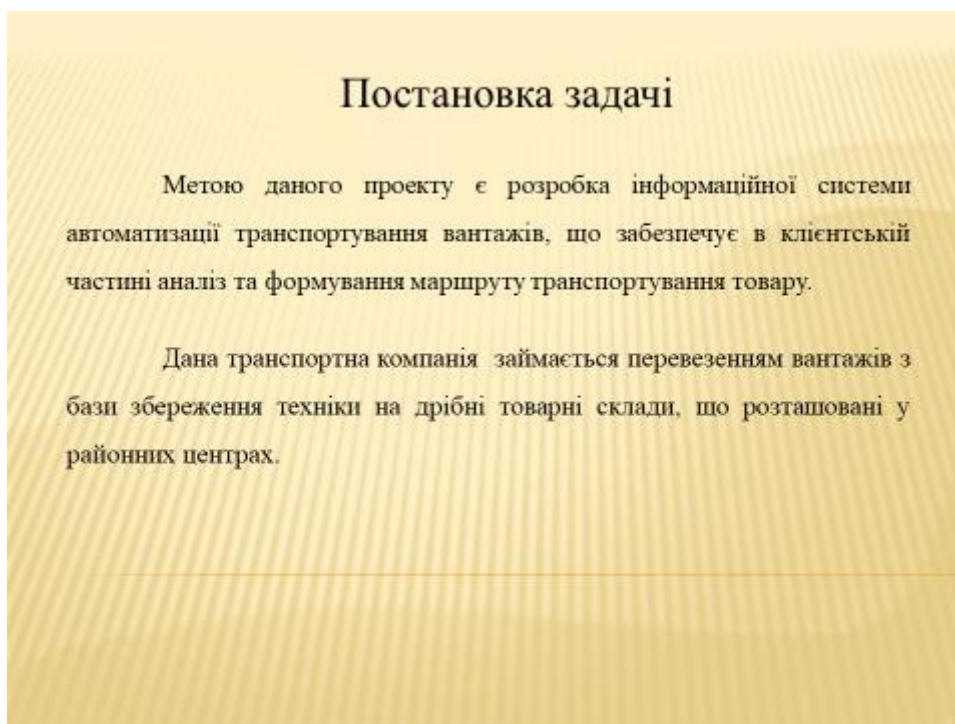


Рисунок А.2 – Слайд 2

## Схема роботи підприємства

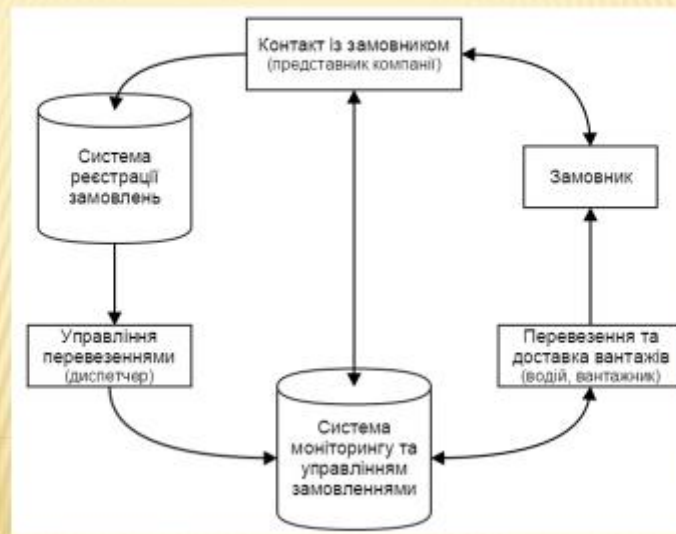


Рисунок А.3 – Слайд 3

## Бізнес-правила

- ❑ Компанія займається перевезенням вантажів з бази збереження техніки на дрібні товарні склади, що розташовані у районних центрах області.
- ❑ Інформаційна система має працювати у режимі 24/7.
- ❑ Інформаційна система має працювати на смартфонах, планшетах та ПК.
- ❑ Перевізник повинен мати змогу подивитися відстань, що повинен подолати.
- ❑ Маршрут має бути сформований у вигляді мапи із намальованим маршрутом руху та у текстовому форматі з описанням маршруту.
- ❑ Рух дозволений виключно через дороги районного значення (через інші склади техніки).
- ❑ Маршрут, що прокладається системою, має бути сформований по найкоротшому шляху із дотриманням вище зазначених правил і обмежень.
- ❑ Приоритет доставки замовлень визначається за кількістю одиниць товару, що треба транспортувати на певний склад.

Рисунок А.4 – Слайд 4

## Структура БД інформаційної системи

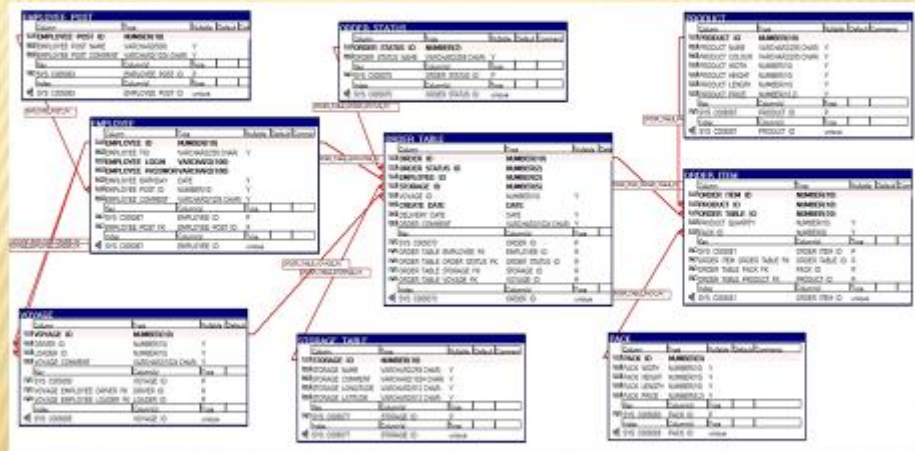


Рисунок А.5 – Слайд 5

## Приклад схеми транспортних розв'язок

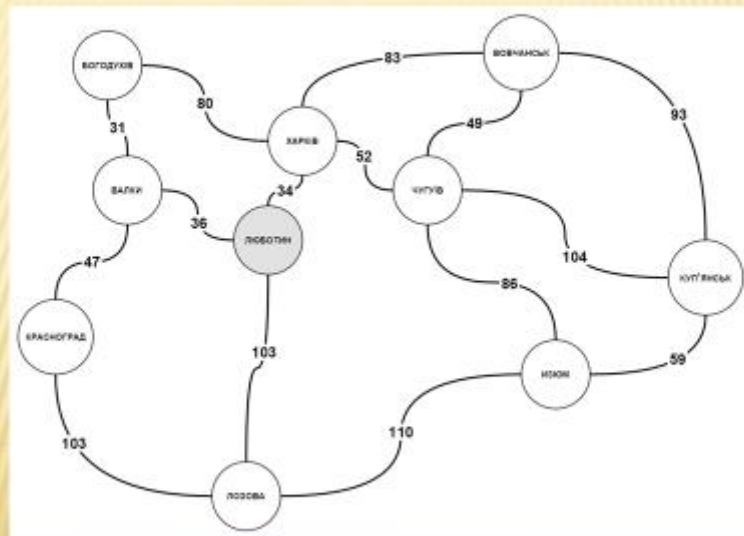


Рисунок А.6 – Слайд 6



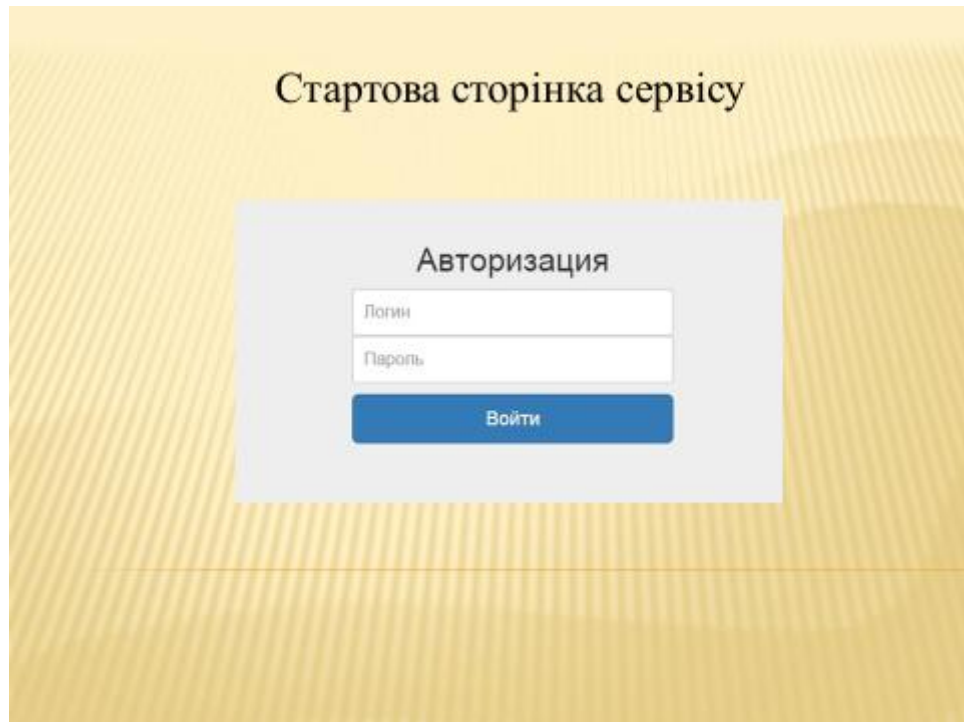


Рисунок А.7 – Слайд 7

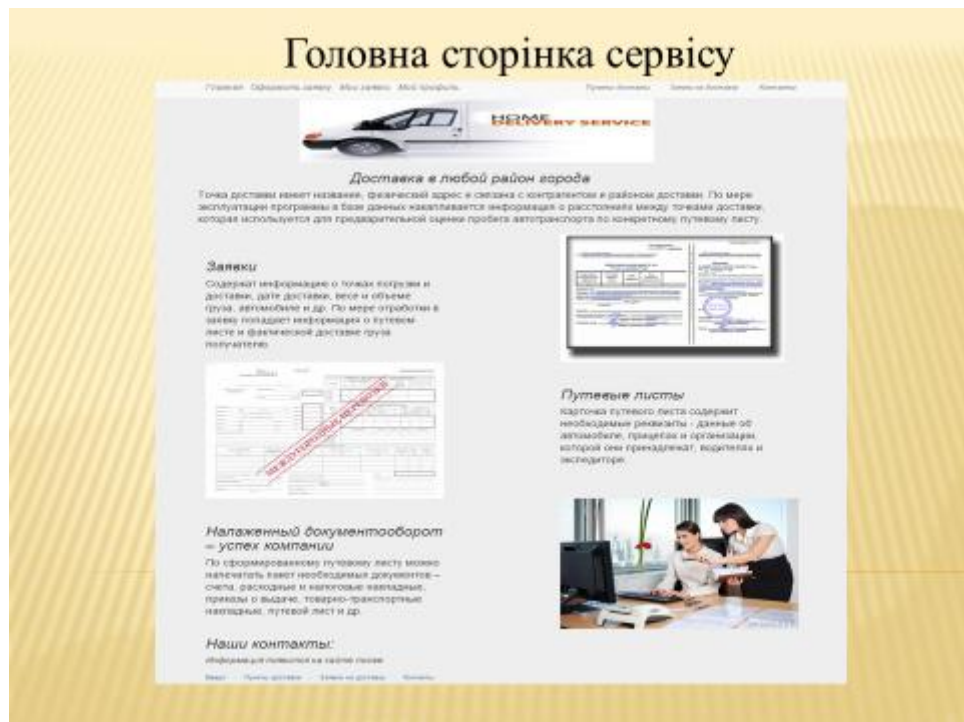


Рисунок А.8 – Слайд 8

## Сторінка профілю користувача

Главная | Оформить заявку | Мои заявки | Мой профиль

Мой профиль

### Профиль пользователя

Данные:	
ИД:	7
Имя:	Олег Орлов
Дата рождения:	11 Aug 99 06:00 GMT+03:00
Пол:	Муж
Должность:	Leader
Комментарий к работе:	Leader
Общий комментарий:	

Вверх | Мой профиль

Copyright © 2015. All Rights Reserved

Рисунок А.9 – Слайд 9

## Сторінка реєстрації замовлення

Главная | Оформить заявку | Мои заявки | Мой профиль

Оформление

### Оформление заявки

Данные:	
Склад:	Чугун
Дата доставки:	
Тип:	Телемаркетинг (01.01.0001)
Валюта:	USD
Общий комментарий:	
Доставка весте ледом	

Вверх | Оформление

Создать

Рисунок А.10 – Слайд 10

## Сторінка замовлень та відображення маршруту

*Мої заявки*

Найменування товару	Ціна товару	Розміри: Д x Ш x В	Кількість	Ціна за шт	Дата створення	Дата доставки	Статус	Склад
Телефон LG 59LY345C	BLACK	1243.0 x 255.0 x 772.0	100	13799.0	Sun Jan 07 14:53:50	Wed Jan 07 00:06:00	Новий	Харків
Телефон LG 32LB38U	BLACK	738.0 x 207.0 x 497.0	100	4099.0	Sun Jan 07 14:54:02	Wed Jan 07 00:06:00	Новий	Вовчанськ
Телефон Samsung UE46H30E	BLACK	918.0 x 267.0 x 688.0	200	16699.0	Sun Jan 07 14:54:09	Wed Jan 07 00:06:00	Новий	Ізюм
Телефон LG 59LY345C	BLACK	1243.0 x 255.0 x 772.0	250	13799.0	Sun Jan 07 14:54:31	Wed Jan 07 00:06:00	Новий	Красноград

*Путевий лист*

*Маршрут: Люботин, Валки, Красноград, Лозова, Ізюм, Вовчанськ, Харків.*

[Вивчити](#)

Рисунок А.11 – Слайд 11

## Графічне відображення маршруту прямування

*Маршрут доставки №1 [ Люботин — Валки ] --- 36.24 км --- Промежуточный*

*Маршрут доставки №2 [ Валки — Красноград ] --- 46.86 км --- Разгрузка*

Рисунок А.12 – Слайд 12



## Огляд інформації на мапі

На мапі доступна наступна інформація:

- ✓ номер маршруту доставки;
- ✓ початкова та кінцева точка маршруту;
- ✓ відстань;
- ✓ чи потрібна відгрузка товарів;
- ✓ масштаб;
- ✓ вичерпна інформація для керування;
- ✓ замітки про пункт доставки.

Рисунок А.13 – Слайд 13

## Охорона праці



Рисунок А.14 – Слайд 14

## Охорона праці

У розділі «Охорона праці» була складена система «І-М-С» із зазначенням основних факторів впливу для робочих місць у розглянутій НДЛ. Також був проведений детальний аналіз умов праці з урахуванням діючих норм і нормативів та виконаний аналіз діючих шкідливих та небезпечних факторів в приміщенні НДЛ для заданих умов, які передбачають роботу 2 осіб і експлуатацію 2 ПК. Було розглянуто заходи з промислової безпеки, гігієни праці, виробничої санітарії і пожежної безпеки.

Рисунок А.15 – Слайд 15

## Висновки

В результаті виконання даного дипломного проекту була розроблена інформаційна система автоматизації транспортування вантажів, яка реалізована у вигляді корпоративного веб-сервісу для фіксації клієнтських замовлень та формування маршруту прямування.

Підсумкова програма дозволяє працівникам вести електронну документацію та здійснювати усі необхідні операції, не прив'язуючись до стаціонарного робочого місця. Система забезпечує графічне відображення маршруту руху та формування і роздрук звітної документації; здійснює контроль за дотриманням термінів виконання замовлень за допомогою надання нотифікації користувачам системи по кожному замовленню.

Розробка ефективних маршрутів і проектів планів перевезень сприяє своєчасному й безперебійному виконанню поставок продукції та ефективній взаємодії організацій-постачальників, одержувачів та автотранспортних організацій.

Рисунок А.16 – Слайд 16