

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова
Навчально-науковий інститут комп'ютерних наук та управління проектами
Кафедра інформаційних управляючих систем та технологій

"Допущений до захисту"

Завідувач кафедри ІУСТ


к.т.н, доц. Михелєв І.Л.

" ____ " _____ 2025 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття ступеня вищої освіти "магістр"

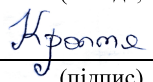
на тему: Дослідження методів та розробка СППР для вибору оптимального маршруту доставки товарів

Виконала: студентка групи 6141мз


(підпис) *Жирнова А.В.*

Керівник роботи:

доцент кафедри ІУСТ, к.т.н.,
(посада, науковий ступінь вчене звання)

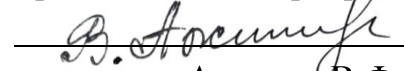

(підпис) *Кронтя А.В.*

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова
Навчально-науковий інститут комп'ютерних наук та управління проектами
Кафедра інформаційних управляючих систем та технологій
Спеціальність 122 "Комп'ютерні науки"
Освітня програма "Комп'ютерні науки"

"ЗАТВЕРДЖУЮ"

Гарант освітньої програми



к.т.н, доц. Ажищев В.Ф.

"___" _____ 2025 р.

*ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ
на здобуття ступеня вищої освіти "магістр"*

Студентці Жирновій Анастасії Віталіївні

(прізвище, ім'я, по батькові)

1. Тема роботи: Дослідження методів та розробка СППР для вибору оптимального маршруту доставки товарів

Керівник роботи к.т.н., доцент Кроптя А.В.

Затверджені наказом ректора № 1355-уч від "14" листопада 2025 року.

2. Термін подання роботи: 15 грудня 2025 р.

3. Вихідні дані до проекту (роботи) ДСТУ щодо обробки інформації, літературні джерела, технічна документація на існуючі аналоги систем, матеріали наукового стажування

4. Перелік питань, що належать до розробки (найменування розділів) Аналіз методів та алгоритмів оптимізації планування випуску продукції. Розробка проекту інформаційної системи. Розробка робочої документації. Розділ з економіки. Розділ з охорони праці. Розділ з охорони навколишнього середовища. Висновки. Список використаної літератури. Додатки

5. Перелік презентаційних матеріалів Актуальність дослідження, Мета, об'єкт, предмет дослідження, наукова новизна, Завдання дослідження, Порівняльна таблиця методів оптимізації маршрутів та СППР, Порівняння СППР у логістиці за трьома критеріями: вартість, функціональність та складність впровадження, Багатокритеріальна модель вибору маршруту для СППР, Схема запропонованої концепції, Діаграма варіантів використання, Діаграма класів, діяльності, компонентів, переходу станів системи, розгортання системи, Висновки

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	к.т.н., доц. Кроптя А.В.	05.09.2024	22.12.2024
2	к.т.н., доц. Кроптя А.В.	23.12.2024	15.06.2025
3	к.т.н., доц. Кроптя А.В.	15.06.2025	25.09.2025
4	к.т.н., доц. Кроптя А.В.	25.09.2025	14.11.2025
5	к.т.н., доц. Кроптя А.В.	25.09.2025	14.11.2025
6	к.т.н., доц. Кроптя А.В.	25.09.2025	14.11.2025

7. Дата видачі завдання 05 вересня 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів роботи	Примітка
1	Вибір теми, визначення мети, завдань, об'єкта та предмета дослідження	22.12.2024	
2	Підбір та опрацювання літератури	15.03.2025	
3	Аналіз стану проблеми планування роботи підприємства	15.06.2025	
4	Аналіз методів розв'язання задач теорії розкладів	25.09.2025	
5	Розробка та реалізація проекту	14.11.2025	
6	Розробка додаткових розділів, оформлення роботи, підготовка презентаційних матеріалів	14.11.2025	
7	Представлення до попереднього захисту	14.11.2025	
8	Отримання відгуку керівника	16.11.2025	
9	Отримання рецензії	20.11.2025	
10	Захист кваліфікаційної роботи	15.12.2025	

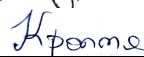
Студент


(підпис)

Жирнова А.В.

(ПІБ)

Керівник роботи


(підпис)

Кроптя А.В.

(ПІБ)

АНОТАЦІЯ

Тема магістерської роботи «Дослідження методів та розробка СППР для вибору оптимального маршруту доставки товарів».

Зростання обсягів електронної комерції та логістичних послуг вимагає ефективних систем планування доставки. Оптимізація маршрутів дозволяє зменшити витрати на паливо, час доставки та негативний вплив на довкілля. Використання систем підтримки прийняття рішень стає критично важливим для компаній, що працюють у конкурентному середовищі. В умовах воєнних та економічних викликів в Україні питання оптимізації логістики має стратегічне значення для бізнесу та гуманітарних поставок.

Мета дослідження полягає в розробці методів та інструментів СППР, які забезпечують вибір оптимального маршруту доставки товарів з урахуванням багатокритеріальних факторів (час, вартість, ризики, екологічність, надійність).

Об'єкт дослідження - процес організації та управління доставкою товарів у логістичних системах.

Предмет дослідження - методи, алгоритми та моделі прийняття рішень для вибору оптимального маршруту доставки товарів у СППР.

Наукова новизна полягає у створенні багатокритеріальної моделі СППР, яка враховує економічні, часові, екологічні та ризик-орієнтовані фактори при виборі маршруту доставки. Запропоновано інтеграцію сучасних алгоритмів оптимізації та адаптивних механізмів, що дозволяють системі реагувати на динамічні зміни умов у реальному часі.

Робота виконана українською мовою та викладена на 92 сторінках, містить 11 рисунків, 4 таблиці та 2 додатки.

Ключові слова: система підтримки прийняття рішень (СППР), оптимізація маршрутів доставки, багатокритеріальна модель, логістика, алгоритми оптимізації, прийняття рішень, інформаційна підсистема.

SUMMARY

The topic of the master's thesis is "Research of methods and development of DSS for selecting the optimal route for delivering goods".

The growth of e-commerce and logistics services requires effective delivery planning systems. Route optimization allows you to reduce fuel costs, delivery time and negative impact on the environment. The use of DSS (decision support systems) is becoming critically important for companies operating in a competitive environment. In the context of military and economic challenges in Ukraine, the issue of logistics optimization is of strategic importance for business and humanitarian supplies.

The purpose of the research is to develop DSS methods and tools that ensure the selection of the optimal route for delivering goods, taking into account multi-criteria factors (time, cost, risks, environmental friendliness, reliability).

The object of the research is the process of organizing and managing the delivery of goods in logistics systems.

The subject of the research is methods, algorithms and decision-making models for choosing the optimal route for delivering goods in DSS.

The scientific novelty lies in the creation of a multi-criteria DSS model that takes into account economic, time, environmental and risk-oriented factors when choosing a delivery route. The integration of modern optimization algorithms and adaptive mechanisms is proposed, allowing the system to respond to dynamic changes in conditions in real time.

The work is in Ukrainian and is presented on 92 pages, contains 11 drawings, 4 tables and 2 appendices.

Keywords: decision support system (DSS), delivery route optimization, multi-criteria model, logistics, optimization algorithms, decision making, information subsystem.

ЗМІСТ

Вступ

1 АНАЛІЗ СТАНУ ПРОБЛЕМИ ОПТИМІЗАЦІЇ ЛОГІСТИЧНИХ МАРШРУТІВ

1.1 Обґрунтування актуальності дослідження

1.2 Особливості процесу транспортування товарів від постачальника до споживача

1.3 Аналіз існуючих підходів до планування маршрутів доставки

1.4 Сучасні методології управління логістичними процесами

1.5 Поняття оперативного планування у сфері транспортної логістики

1.6 Інформаційні системи та СППР для підтримки прийняття рішень у виборі маршрутів

Висновки до розділу 1

2 МАТЕМАТИЧНЕ ОБҐРУНТУВАННЯ ТА МОДЕЛЬ КОНЦЕПЦІЇ СППР

2.1 Математична постановка багатокритеріальної задачі вибору маршруту

2.2 Дослідження властивостей задачі та методів її розв'язання

2.3 Побудова моделі концепції СППР для вибору маршрутів

2.4 Постановка задачі

Висновки до розділу 2

3 РОЗРОБКА ПРОЄКТУ ІНФОРМАЦІЙНОЇ ПІДСИСТЕМИ СППР

3.1 Загальносистемні рішення

3.2 Рішення з інформаційного забезпечення

3.3 Рішення з технічного забезпечення

3.4 Рішення з програмного забезпечення

Висновки до розділу 3

4 РОЗРАХУНОК ЕКОНОМІЧНОЇ ЕФЕКТИВНОСТІ ВПРОВАДЖЕННЯ СИСТЕМИ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ

4.1 Вступ

4.2 Розрахунок витрат на створення та експлуатацію інформаційної системи

4.3 Економічна ефективність розробки і впровадження системи

Висновки до розділу 4

5 ОХОРОНА ПРАЦІ

5.1 Вступ

5.2 Аналіз шкідливих та небезпечних факторів

5.3 Розрахунок системи штучного освітлення приміщення відділу інформаційних технологій

5.4 Розробка заходів щодо зменшення впливу шкідливих та небезпечних факторів при роботі з персональним комп'ютером

5.4.1 Заходи щодо зменшення впливу

5.4.2 Техніка безпеки при роботі на комп'ютері

Висновки до розділу 5

6 ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА

6.1 Вступ

6.2 Забруднення навколишнього середовища

6.3 Розробка заходів щодо зменшення забруднення навколишнього середовища

Висновки до розділу 6

Висновки

Список використаної літератури

ДОДАТКИ

ВСТУП

Магістерська робота присвячена дослідженню оптимізації процесів транспортування товарів від постачальника до кінцевого споживача. Зростання електронної комерції та розвиток логістичних послуг формують потребу у впровадженні сучасних систем планування доставки. Рациональне використання маршрутів дозволяє скоротити витрати на паливо, зменшити час перевезень та знизити негативний вплив на довкілля. У конкурентному середовищі ключову роль відіграє застосування систем підтримки прийняття рішень, які забезпечують ефективність управління логістичними процесами. В умовах воєнних та економічних викликів в Україні питання оптимізації логістики набуває стратегічного значення як для бізнесу, так і для організації гуманітарних поставок.

Метою дослідження є розробка методів та інструментів СППР, які забезпечують вибір оптимального маршруту доставки товарів з урахуванням багатокритеріальних факторів (час, вартість, ризики, екологічність, надійність).

Для досягнення вказаної мети були поставлені наступні задачі дослідження:

1. Провести аналіз сучасних методів оптимізації маршрутів доставки та систем підтримки прийняття рішень у логістиці.
2. Визначити критерії оцінки оптимальності маршруту.
3. Розробити багатокритеріальну модель вибору маршруту для СППР.
4. Інтегрувати алгоритми оптимізації в структуру системи.
5. Побудувати UML моделі та інші діаграми для формалізації архітектури СППР.
6. Реалізувати прототип системи та перевірити його ефективність на тестових сценаріях доставки.
7. Оцінити результати роботи СППР за технічними, економічними та екологічними показниками.

Об'єктом дослідження є процес організації та управління доставкою товарів у логістичних системах.

Предметом дослідження є методи, алгоритми та моделі прийняття рішень для вибору оптимального маршруту доставки товарів у СППР.

Наукова новизна полягає у створенні багатокритеріальної моделі СППР, яка враховує економічні, часові, екологічні та ризик-орієнтовані фактори при виборі маршруту доставки. Запропоновано інтеграцію сучасних алгоритмів оптимізації та адаптивних механізмів, що дозволяють системі реагувати на динамічні зміни умов у реальному часі.

Завданням є аналіз та розробка інформаційної системи для вибору оптимального маршруту доставки товарів.

У першому розділі описується теоретичний огляд.

У другому розділі описується методи прийняття рішень.

У третьому розділі проблеми вибору та інформаційна система.

Четвертий розділ присвячений розрахунку економічної ефективності системи підтримки прийняття рішень.

В п'ятому проводиться аналіз шкідливих та небезпечних факторів, розрахунок системи штучного освітлення та розробка заходів щодо зменшення впливу шкідливих та небезпечних факторів.

В шостому розділі визначається зменшення забруднення навколишнього середовища.

1 АНАЛІЗ СТАНУ ПРОБЛЕМИ ОПТИМІЗАЦІЇ ЛОГІСТИЧНИХ МАРШРУТІВ

1.1 Обґрунтування актуальності дослідження

У сучасних умовах стрімкого розвитку електронної комерції та зростання обсягів логістичних послуг питання оптимізації маршрутів доставки товарів набуває особливої значущості. Ефективне планування транспортних процесів дозволяє підприємствам зменшити витрати на паливо, скоротити час перевезень та мінімізувати негативний вплив на довкілля.

Висока конкуренція на ринку логістики вимагає від компаній впровадження інноваційних інструментів управління, серед яких ключову роль відіграють системи підтримки прийняття рішень (СППР). Вони забезпечують можливість швидкого аналізу альтернативних маршрутів, врахування багатьох критеріїв та вибір оптимального варіанту доставки.

В умовах воєнних та економічних викликів в Україні питання ефективної організації логістики має стратегічне значення не лише для бізнесу, але й для забезпечення гуманітарних поставок. Саме тому дослідження методів оптимізації та розробка СППР для вибору оптимальних маршрутів доставки є актуальним і необхідним напрямом наукових та практичних розробок.

1.2 Особливості процесу транспортування товарів від постачальника до споживача

Процес доставки товарів від виробника чи постачальника до кінцевого споживача складається з низки взаємопов'язаних етапів, які формують єдину логістичну систему. Основними операціями є:

- планування маршруту з урахуванням відстані, дорожніх умов та часу доставки;

- формування транспортних партій відповідно до обсягів замовлень та характеристик продукції;
- завантаження та розподіл товарів у транспортних засобах для забезпечення їхньої збереженості;
- транспортування із застосуванням оптимальних видів транспорту;
- контроль та моніторинг руху за допомогою GPS та інформаційних систем;
- розвантаження та передача товарів у пункті призначення.

Рациональне використання транспортних ресурсів передбачає вибір оптимальних маршрутів, що дозволяє зменшити витрати на паливо, скоротити час перевезення та забезпечити своєчасність доставки. Важливим чинником є також врахування специфіки товарів: для харчової продукції необхідно дотримуватися температурних режимів, для великогабаритних вантажів – вимог до безпеки перевезення.

Організація процесу транспортування здійснюється логістичними підрозділами підприємства або спеціалізованими транспортними компаніями. До складу команди можуть входити:

- диспетчери, які планують та координують маршрути;
- водії та експедитори, що відповідають за фізичне переміщення товарів;
- технічний персонал для обслуговування транспортних засобів;
- аналітики та оператори СППР, які забезпечують вибір оптимальних рішень.

З урахуванням наведених факторів транспортна інфраструктура підприємства включає:

- складські приміщення для зберігання товарів;
- зони навантаження та розвантаження;
- диспетчерські центри для моніторингу доставки;
- допоміжні приміщення для персоналу та технічного обслуговування.

Таким чином, ефективність процесу транспортування визначається комплексним підходом до планування, використанням сучасних інформаційних

систем та впровадженням СППР, що дозволяють враховувати багатокритеріальні умови при виборі оптимального маршруту доставки.

1.3 Аналіз існуючих підходів до планування маршрутів доставки

Ефективність логістичної діяльності підприємства значною мірою залежить від правильного вибору та планування маршрутів транспортування. Планування доставки є комплексним процесом, що включає прогнозування потреб, визначення оптимальних шляхів руху, розподіл ресурсів та координацію дій між усіма учасниками ланцюга постачання.

У практиці логістики можна виділити кілька основних підходів до планування маршрутів:

- Маршрутизація за цілями – орієнтація на досягнення стратегічних завдань компанії, таких як мінімізація витрат, скорочення часу доставки чи підвищення рівня обслуговування клієнтів.

- Маршрутизація для повторюваних перевезень – використовується у випадках регулярних поставок, коли маршрути формуються на основі стабільних замовлень і відносно незмінних умов.

- Маршрутизація для неповторюваних перевезень – застосовується у випадках нестандартних або разових замовлень, що потребують індивідуального підходу.

При плануванні маршрутів організація повинна враховувати:

1. Вимоги клієнтів щодо термінів та якості доставки.
2. Наявність і стан транспортних ресурсів.
3. Особливості дорожньої інфраструктури та можливі обмеження.
4. Законодавчі та екологічні вимоги.
5. Потенційні ризики та надзвичайні обставини.

Сучасні методи планування маршрутів базуються на використанні математичних моделей та алгоритмів оптимізації. Значна частина задач належить до класу NP-складних, що ускладнює пошук точного рішення для

великих систем. Тому на практиці широко застосовуються евристичні та метоевристичні методи (генетичні алгоритми, алгоритми мурашиних колоній, табу-пошук тощо), які дозволяють знаходити наближені оптимальні рішення за прийнятний час.

Таблиця 1.1

Порівняльна таблиця методів оптимізації маршрутів та СППР

Метод / Підхід	Переваги	Недоліки	Типові сфери застосування
Класичні алгоритми	Точність, гарантоване знаходження найкоротшого шляху	Погано масштабуються для великих мереж; не враховують багатокритеріальність	Навігація, локальні маршрути, прості графи
Математичне програмування	Оптимальні рішення, строгі моделі	Висока обчислювальна складність, потреба у потужних ресурсах	Планування доставки у невеликих мережах
Евристики	Швидкість, простота реалізації	Рішення не завжди оптимальні, залежність від початкових умов	Великі мережі, де потрібні швидкі наближені рішення
Метоевристики	Гнучкість, здатність працювати з великими даними, багатокритеріальність	Стохастичність результатів, потреба у налаштуванні параметрів	Логістика e-commerce, кур'єрські служби
Динамічна маршрутизація	Адаптивність, врахування трафіку, погоди, ризиків	Висока вартість впровадження, складність інтеграції	Кур'єрські служби, міська логістика, гуманітарні поставки
Багатокритеріальні методи	Балансування часу, вартості, екології та ризиків	Складність у визначенні ваг критеріїв, суб'єктивність	Стратегічне планування, корпоративна логістика
ШІ та ML	Прогнозування попиту, ETA, адаптивність до змін	Потреба у великих даних, складність навчання	Великі логістичні компанії, глобальні мережі

Порівняльний аналіз показує, що кожен метод має свої переваги та обмеження, а вибір оптимального підходу залежить від масштабу задачі, доступних ресурсів та вимог до доставки. Класичні алгоритми та математичне програмування забезпечують точність, але обмежені у масштабованості, тоді як евристики та метаевристики дозволяють працювати з великими мережами, хоча й не гарантують абсолютної оптимальності.

Важливим інструментом у сучасній логістиці є системи підтримки прийняття рішень (СППР), які забезпечують аналіз альтернативних маршрутів, врахування багатокритеріальних умов та вибір оптимального варіанту доставки. Використання таких систем дозволяє підприємствам підвищити продуктивність, зменшити витрати та забезпечити гнучкість у реагуванні на зміни ринкових умов.

Використання СППР, інтегрованих із сучасними алгоритмами та технологіями ШІ, забезпечує багатокритеріальний аналіз і дає можливість підприємствам досягати балансу між швидкістю, вартістю та якістю логістичних процесів.

Для більшої наочності проведено порівняння п'яти сучасних СППР у сфері логістики за трьома критеріями (рисунок 1.1):

- вартість рішення (синій стовпчик, шкала 0–10),
- функціональність (зелений стовпчик),
- складність впровадження (світло-зелений стовпчик).

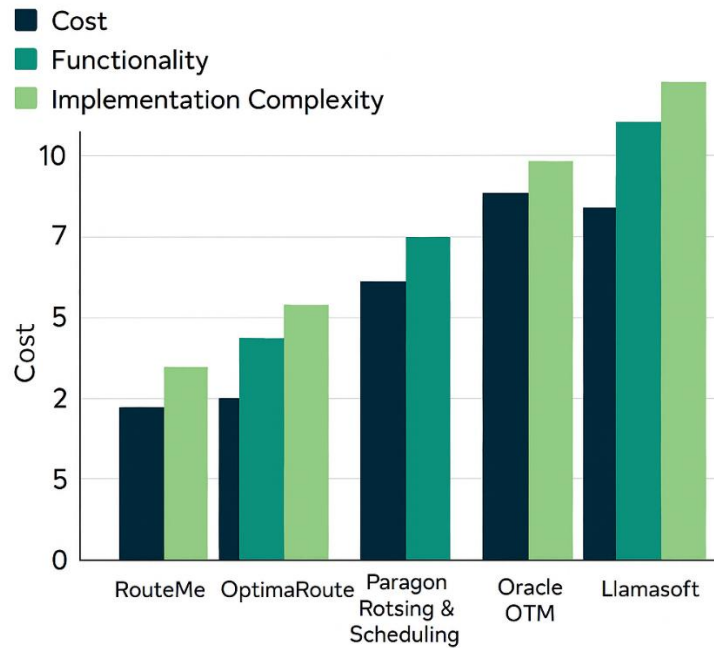


Рисунок 1.1 - Багатошарова стовпчикова діаграма

Багатошарова стовпчикова діаграма демонструє, що:

- Route4Me та OptimoRoute – бюджетні рішення з базовим функціоналом, прості у впровадженні;
- Paragon – збалансований варіант середнього рівня за всіма критеріями;
- Oracle OTM та Llamasoft – високофункціональні системи, але дорогі та складні у впровадженні, орієнтовані на великі корпорації.

Вибір СППР залежить від масштабу підприємства та його стратегічних цілей: для малого та середнього бізнесу доцільні прості й бюджетні рішення, тоді як великі корпорації потребують комплексних систем із розширеним функціоналом, навіть попри високу вартість та складність інтеграції.

1.4 Сучасні методології управління логістичними процесами

Ефективність управління логістикою визначається здатністю підприємства швидко адаптуватися до змін ринку, мінімізувати витрати та забезпечувати високу якість обслуговування клієнтів. Сучасні методології спрямовані на координацію дій усіх учасників ланцюга постачання, оптимізацію використання ресурсів, підвищення швидкості та гнучкості системи. Вони формуються на

основі узагальнення практичного досвіду та закономірностей розвитку транспортних і складських систем і визначають правила, за якими функціонує та розвивається менеджмент доставки.

Серед найбільш поширених методологій можна виділити:

- Ощадна логістика (Lean Logistics) – орієнтована на усунення втрат у процесах транспортування та складування, мінімізацію зайвих переміщень і простоїв, оптимізацію використання ресурсів. Основна ідея полягає у створенні максимально ефективного ланцюга постачання, де кожна операція додає цінність.

- Концепція «точно в строк» (Just In Time, JIT) – забезпечує доставку товарів саме тоді, коли вони потрібні, без накопичення надлишкових запасів. Це дозволяє знизити витрати на зберігання, але водночас підвищує ризик зривів у випадку затримок постачання.

- Швидке реагування (Quick Response Logistics, QRL) – орієнтоване на скорочення часу виконання замовлення. Завдяки інтеграції інформаційних систем та гнучкому управлінню транспортними ресурсами компанії можуть швидко реагувати на зміни попиту та забезпечувати своєчасну доставку.

- Гнучка логістика (Agile Logistics) – стратегія, що робить підприємство стійким до кризових ситуацій та непередбачуваних змін. Її перевага полягає у здатності швидко перебудовувати ресурси та адаптуватися до нових умов, що особливо важливо в умовах високої невизначеності ринку.

- Логістика якості (Total Quality Logistics, TQL) – акцентує увагу на відповідності міжнародним стандартам та високому рівні контролю всіх етапів доставки. Вона забезпечує довіру клієнтів і стабільність процесів, але потребує значних інвестицій у сертифікацію та контроль.

Кожна з методологій має власну стратегічну орієнтацію: Lean – на ліквідацію втрат і підвищення ефективності; JIT – на точність і синхронність поставок; QRL – на швидкість реагування; Agile – на гнучкість і стійкість до змін; TQL – на якість та стандартизацію процесів.

Сучасні методології управління логістичними процесами дозволяють підприємствам досягати балансу між швидкістю, вартістю та якістю доставки. Їхнє впровадження є ключовим чинником конкурентоспроможності компаній у динамічному ринковому середовищі.

1.5 Поняття оперативного планування у сфері транспортної логістики

Управління логістичними процесами передбачає багаторівневу систему планування, де особливе місце займає оперативне планування. Воно забезпечує узгоджену роботу всіх елементів транспортної системи з метою своєчасної доставки товарів при оптимальному використанні ресурсів.

Оперативне планування у сфері транспортної логістики включає:

- оперативно-календарне планування – визначення часу та послідовності виконання перевезень, формування графіків руху транспортних засобів;
- диспетчеризацію – контроль фактичного ходу перевезень, виявлення відхилень від плану та їхнє оперативне усунення;
- регулювання процесів доставки – коригування маршрутів та графіків у разі зміни умов (трафік, погодні фактори, технічні несправності).

Завданням оперативного планування є організація ритмічної та взаємоузгодженої роботи транспортних підрозділів, що дозволяє досягти високої якості обслуговування клієнтів і мінімізувати витрати.

Система оперативного управління транспортною логістикою складається з кількох рівнів:

- 1) Стратегічне планування – визначення довгострокових цілей і бізнес-моделі доставки.
- 2) Планування перевезень – розподіл транспортних ресурсів відповідно до обсягів замовлень.
- 3) Головний календарний план – створення графіків руху транспортних засобів.

- 4) Планування потреби в ресурсах – визначення необхідної кількості транспорту, пального, персоналу.
- 5) Закупівлі та контроль – забезпечення матеріально-технічної бази та моніторинг виконання планів.

Метою оперативного планування є суворе дотримання термінів доставки, оптимальне використання транспортних ресурсів та мобілізація внутрішніх резервів. Для цього застосовуються сучасні інформаційні системи, які автоматизують процеси планування, обліку та контролю.

Функції системи оперативного управління транспортною логістикою включають:

- управління як процес прийняття рішень;
- планування як визначення оптимальної лінії поведінки транспортної системи;
- облік як контроль виконання перевезень та аналіз відхилень;
- регулювання як усунення проблем і підтримання стабільності доставки.

Таким чином, оперативне планування у транспортній логістиці є ключовим інструментом забезпечення ефективності перевезень. Воно дозволяє підприємствам швидко реагувати на зміни, підтримувати ритмічність роботи та досягати високого рівня сервісу при мінімальних витратах. На рисунку 1.2 представлена схема оперативного планування у транспортній логістиці.



Рисунок 1.2 – Схема оперативного планування у транспортній логістиці

Ця схема демонструє п'ять ключових етапів: стратегічне планування → планування перевезень → оперативно-календарне планування → диспетчеризація → контроль і коригування. Вона узагальнює логіку оперативного управління та показує послідовність дій у системі доставки.

1.6 Інформаційні системи та СППР для підтримки прийняття рішень у виборі маршрутів

У сучасній транспортній логістиці інформаційні системи та системи підтримки прийняття рішень відіграють ключову роль у виборі оптимальних маршрутів. Якщо у попередньому підрозділі ми розглядали СППР як загальну концепцію, то тут акцент робиться на їхньому практичному застосуванні.

Інформаційні системи забезпечують збір і обробку даних про замовлення, транспортні ресурси та дорожні умови, формують графіки перевезень і дозволяють здійснювати моніторинг доставки у реальному часі. Вони інтегруються з іншими бізнес-системами, такими як ERP чи CRM, що дає змогу узгоджувати логістичні процеси з виробничими та комерційними завданнями підприємства.

Системи підтримки прийняття рішень у виборі маршрутів дозволяють аналізувати альтернативні варіанти доставки з урахуванням часу, вартості, ризиків та екологічних показників. Вони застосовують алгоритми оптимізації - від класичних методів математичного програмування до сучасних евристичних і машинного навчання. Завдяки цьому маршрути можуть адаптуватися до змінних умов, таких як затори, погодні фактори чи технічні несправності.

На практиці такі системи представлені різними рівнями складності. Простими рішеннями для малого бізнесу є Route4Me чи OptimoRoute, які забезпечують базову оптимізацію маршрутів. Середній рівень функціональності демонструє Paragon, що поєднує доступність і ширші можливості. Для великих корпорацій використовуються комплексні платформи на кшталт Oracle OTM чи Llamasoft, які дозволяють здійснювати багатокритеріальний аналіз і повну інтеграцію з управлінням ланцюгами постачання.

Таким чином, інформаційні системи та СППР у виборі маршрутів є практичним інструментом, що забезпечує підприємствам можливість приймати обґрунтовані рішення, підтримувати гнучкість і адаптивність логістичної системи та створювати конкурентні переваги. Якщо у пункті 1.3 йшлося про

концептуальні основи СППР, то у пункті 1.6 показано їхнє реальне застосування у транспортній логістиці, що робить структуру дослідження більш повною та послідовною.

Уявімо кур'єрську службу, яка щодня має доставити сотні замовлень у великому місті. Традиційне планування маршрутів займає багато часу і не враховує непередбачувані фактори, наприклад затори чи погодні умови. Використання інформаційної системи, інтегрованої зі СППР, дозволяє автоматично збирати дані про стан дорожнього руху, прогнозувати затримки та пропонувати альтернативні маршрути.

Наприклад, якщо на головній магістралі утворився затор, система у реальному часі перебудовує графік руху, перенаправляючи кур'єра на менш завантажені вулиці. При цьому враховується не лише час доставки, а й витрати на паливе та пріоритетність замовлень. Завдяки багатокритеріальному аналізу СППР може обрати маршрут, який забезпечить мінімальні витрати і водночас дозволить виконати доставку у встановлений термін.

У результаті компанія отримує ритмічну та стабільну роботу транспортної мережі, знижує операційні витрати та підвищує рівень задоволеності клієнтів. Саме такі приклади демонструють, що інформаційні системи та СППР у виборі маршрутів - це не лише теоретичні моделі, а реальні інструменти, які щодня впливають на ефективність логістики.

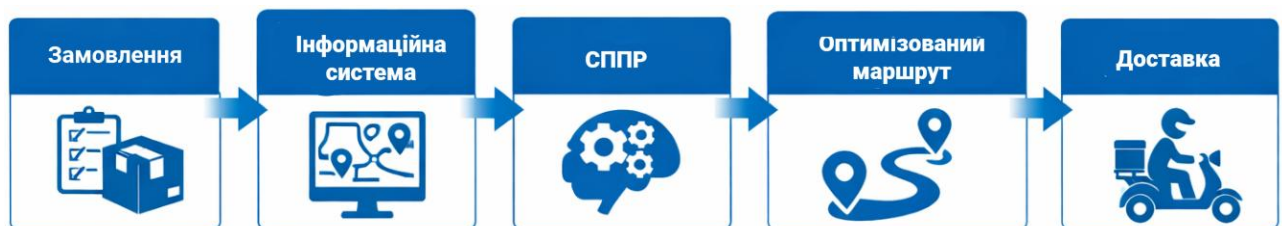


Рисунок 1.3 – Схема взаємодії інформаційної системи та СППР у виборі оптимального маршруту доставки

Рисунок 1.3 ілюструє послідовність процесу: від моменту отримання замовлення до його виконання через аналіз даних, оптимізацію маршруту та доставку. Він доповнює приклад кур'єрської служби, показуючи, як

інформаційні системи та СППР працюють разом для досягнення ефективної логістики.

Висновки до розділу 1

У першому розділі було розглянуто фундаментальні поняття, методології та інструменти, що лежать в основі сучасного управління логістикою. Визначено, що логістичні процеси є складною системою взаємодії матеріальних, інформаційних та фінансових потоків, ефективність якої залежить від узгодженості дій усіх учасників ланцюга постачання.

Сучасні методології управління логістикою, такі як Lean, JIT, Agile, QRL та TQL, демонструють різні стратегічні підходи до оптимізації витрат, підвищення швидкості доставки та забезпечення якості обслуговування. Їхнє впровадження дозволяє підприємствам адаптуватися до змін ринку та підвищувати конкурентоспроможність.

Оперативне планування у транспортній логістиці розглядається як ключовий інструмент забезпечення ритмічної роботи системи доставки. Воно включає стратегічне планування, формування графіків перевезень, диспетчеризацію та контроль виконання. Візуалізація цього процесу у вигляді блок-схеми дозволяє краще зрозуміти логіку взаємодії між етапами.

Особливу увагу приділено інформаційним системам та системам підтримки прийняття рішень (СППР), які забезпечують обґрунтований вибір маршрутів. На прикладі кур'єрської служби показано, як СППР дозволяє адаптувати маршрути до змінних умов, знижувати витрати та підвищувати якість логістичного сервісу.

2 МАТЕМАТИЧНЕ ОБҐРУНТУВАННЯ ТА МОДЕЛЬ КОНЦЕПЦІЇ СППР

2.1 Математична постановка багатокритеріальної задачі вибору маршруту

Задача вибору оптимального маршруту у транспортній логістиці належить до класу багатокритеріальних задач оптимізації. Нехай маємо множину можливих маршрутів

$$R = \{r_1, r_2, \dots, r_m\}.$$

Кожен маршрут r_i характеризується набором показників (критеріїв)

$$x_{ij}, \text{ де } j = 1, \dots, n,$$

де n - кількість критеріїв (наприклад, час доставки, вартість, ризики, екологічність).

Для узгодження критеріїв застосовується система вагових коефіцієнтів

$$w_j, \text{ де } j = 1, \dots, n,$$

$$\sum_{j=1}^n w_j = 1,$$

що відображає пріоритетність кожного критерію.

Після нормалізації показників формується інтегральна оцінка маршруту. У найпростішому випадку використовується метод зваженої суми (WSM):

$$S_i = \sum_{j=1}^n w_j \cdot x_{ij}$$

Оптимальний маршрут визначається як

$$r^* = \arg \max_{r_i \in R} S_i$$

Альтернативно можуть застосовуватися інші методи багатокритеріального аналізу, наприклад TOPSIS, де обчислюється відстань кожного маршруту до ідеального та антиідеального рішень, або метод ϵ -обмежень, що дозволяє виділити множину Парето-оптимальних рішень.

Таким чином, математична постановка задачі включає:

- множину можливих маршрутів;
- систему критеріїв оцінки;
- вагові коефіцієнти для узгодження пріоритетів;
- метод інтеграції критеріїв (WSM, TOPSIS, ϵ -обмеження);
- правило вибору оптимального маршруту.

Цей формалізований підхід дозволяє перейти від опису логістичних процесів до побудови моделі концепції СППР, яка реалізує багатокритеріальний вибір маршруту.

2.2 Дослідження властивостей задачі та методів її розв'язання

Багатокритеріальна задача вибору маршруту має низку властивостей, які визначають її складність та особливості розв'язання.

По-перше, вона є множинною за критеріями: кожен маршрут оцінюється не одним показником, а цілою системою критеріїв (час, вартість, ризики, екологічність тощо). Це означає, що оптимальне рішення залежить від узгодження різних, часто суперечливих вимог.

По-друге, задача є чутливою до вагових коефіцієнтів. Зміна пріоритетів користувача або підприємства (наприклад, підвищення ваги критерію «час доставки» порівняно з «вартістю») може суттєво змінити результат ранжування маршрутів. Таким чином, система повинна забезпечувати гнучкість у налаштуванні ваг.

По-третє, задача має множину Парето-оптимальних рішень. Це означає, що існує набір маршрутів, які не можна покращити за одним критерієм без погіршення іншого. Вибір конкретного маршруту з множини Парето залежить від додаткових пріоритетів або методів інтеграції критеріїв (WSM, TOPSIS, ϵ -обмеження).

По-четверте, задача є динамічною: значення критеріїв можуть змінюватися в реальному часі (трафік, погодні умови, технічні несправності). Це вимагає

адаптивності моделі та можливості швидкого перерахунку оптимального рішення.

Нарешті, задача має властивість обчислювальної складності. Генерація кандидатів маршрутів у задачі VRP належить до NP-складних проблем, тому для практичного застосування використовуються евристичні та метаевристичні методи, які забезпечують прийнятну якість рішень за обмежений час.

Таким чином, багатокритеріальна задача вибору маршруту характеризується багатокритеріальністю, чутливістю до ваг, наявністю множини Парето-оптимальних рішень, динамічністю та високою обчислювальною складністю. Це визначає необхідність використання спеціалізованих методів і систем підтримки прийняття рішень для її ефективного розв'язання.

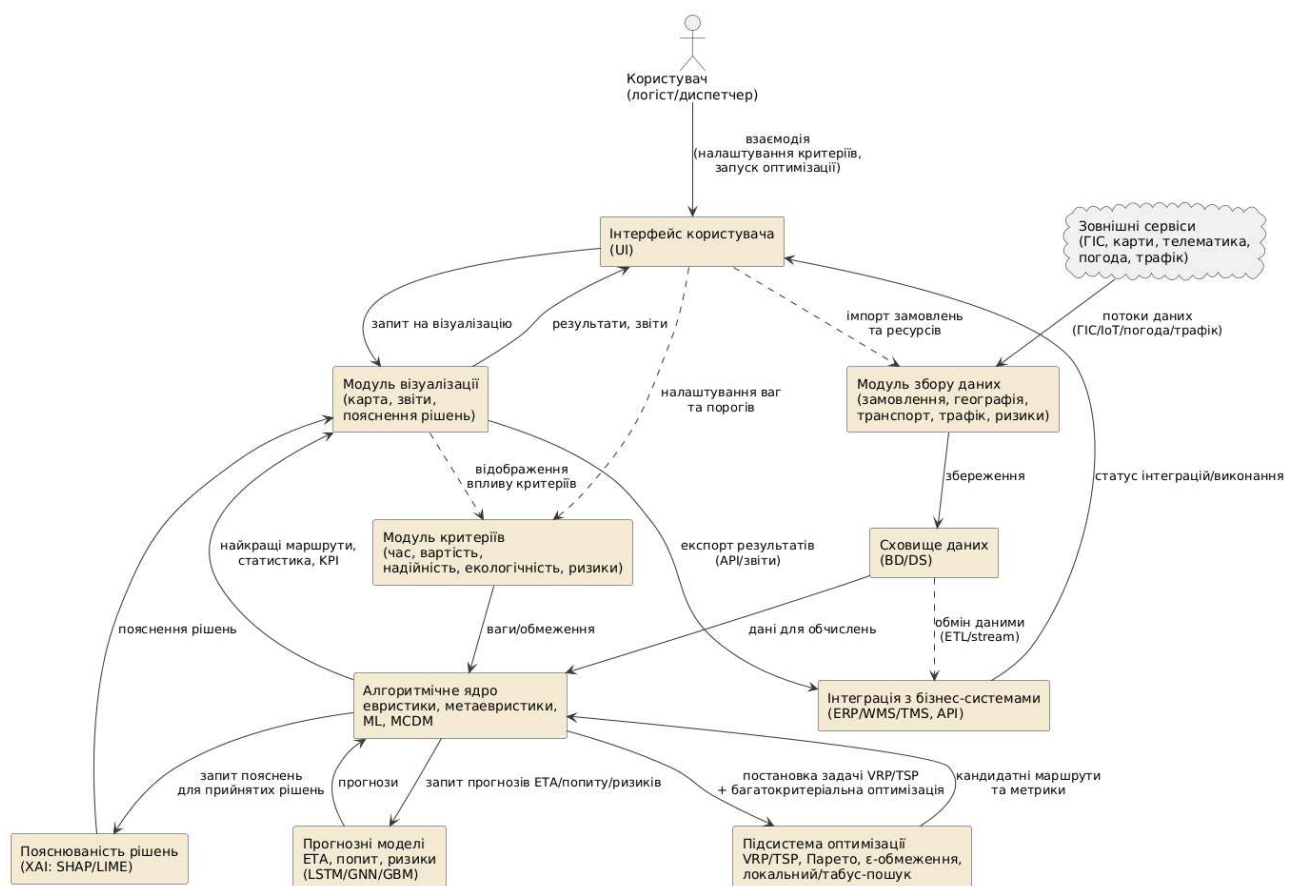
2.3 Побудова моделі концепції СППР для вибору маршрутів

Запропонована концепція системи підтримки прийняття рішень у виборі маршрутів ґрунтується на інтеграції даних, багатокритеріальній оптимізації та пояснюваності результатів. Вона передбачає взаємодію користувача (логіста чи диспетчера) з інтерфейсом системи, де задаються критерії та запускається процес оптимізації.

Зовнішні сервіси (геоінформаційні системи, карти, телематика, погодні та трафікові дані) забезпечують надходження інформації про замовлення, транспортні ресурси та ризики. Ці дані збираються модулем збору та інтегруються у сховище, яке взаємодіє з бізнес-системами підприємства (ERP, WMS, TMS).

Ключовим елементом є підсистема оптимізації, що реалізує генерацію кандидатів маршрутів (VRP/TSP), багатокритеріальний аналіз та відбір Парето-оптимальних рішень. Алгоритмічне ядро поєднує евристики, метаевристики, методи машинного навчання та багатокритеріальні методи прийняття рішень (MCDM).

Особливу роль відіграє модуль критеріїв, де визначаються ваги та порогові значення для таких показників, як час, вартість, надійність, екологічність та ризику. Результати оптимізації відображаються у модулі візуалізації у вигляді карт, звітів та пояснень рішень. Для забезпечення прозорості застосовуються методи пояснюваного штучного інтелекту (XAI: SHAP, LIME), що дозволяють користувачу зрозуміти, чому саме було обрано той чи інший маршрут.



На рисунку 2.1 показано взаємодію користувача з інтерфейсом системи, інтеграцію зовнішніх сервісів та бізнес-систем, роботу підсистеми оптимізації, алгоритмічного ядра, прогнозних моделей і модулів пояснюваності та візуалізації.

Формалізована багатокритеріальна задача інтегрується у концепцію СППР через алгоритмічне ядро та модуль критеріїв. Математична модель визначає множину маршрутів, систему критеріїв та методи їх інтеграції, тоді як концепція системи забезпечує практичну реалізацію цього процесу: збір даних, генерацію кандидатів, ранжування та пояснення результатів. Таким чином, математична постановка і концептуальна архітектура утворюють єдину основу для побудови ефективної СППР у транспортній логістиці.

2.4 Постановка задачі

Для досягнення поставленої мети та реалізації запропонованої концепції необхідно виконати такі завдання:

1. Провести аналіз сучасних методів оптимізації маршрутів доставки та систем підтримки прийняття рішень у логістиці.
2. Визначити критерії оцінки оптимальності маршруту (час, вартість, ризики, екологічність, надійність) та сформулювати систему вагових коефіцієнтів.
3. Розробити багатокритеріальну математичну модель вибору маршруту для СППР.
4. Інтегрувати алгоритми оптимізації у структуру системи для генерації та відбору кандидатів маршрутів.
5. Побудувати UML-моделі та інші діаграми для формалізації архітектури СППР.
6. Розробити прототип інформаційної підсистеми та перевірити його ефективність на тестових сценаріях доставки.
7. Оцінити результати роботи СППР за технічними, економічними та екологічними показниками, а також забезпечити пояснюваність прийнятих рішень.

Висновки до розділу 2

Було здійснено математичне обґрунтування задачі вибору маршруту та побудовано модель концепції СППР.

Формалізація задачі показала, що вибір маршруту є багатокритеріальною оптимізацією, де рішення залежить від узгодження кількох показників, часто суперечливих між собою.

Дослідження властивостей задачі виявило її чутливість до вагових коефіцієнтів, наявність множини Парето-оптимальних рішень, динамічність та високу обчислювальну складність.

Запропонована модель концепції СППР інтегрує зовнішні джерела даних, алгоритмічні методи оптимізації, прогностичні моделі та інструменти пояснюваності рішень, забезпечуючи прозорість і практичну корисність системи.

Таким чином, створена основа для подальшого проектування інформаційної підсистеми СППР.

3 РОЗРОБКА ПРОЄКТУ ІНФОРМАЦІЙНОЇ ПІДСИСТЕМИ СППР

3.1 Загальносистемні рішення

Запропонована система підтримки прийняття рішень для вибору оптимального маршруту доставки товарів є інструментом, що поєднує математичні методи багатокритеріальної оптимізації та сучасні інформаційні технології. Її головне призначення полягає у забезпеченні ефективного планування логістичних процесів, зниженні витрат та підвищенні надійності доставки в умовах динамічного середовища.

Система орієнтована на використання у:

- компаніях електронної комерції, де швидкість і точність доставки визначають конкурентоспроможність;
- транспортних та кур'єрських службах, що прагнуть оптимізувати витрати на паливо та час перевезень;
- гуманітарних поставках, де оптимізація маршрутів має стратегічне значення в умовах воєнних та економічних викликів.

Основна мета системи – надати користувачеві інструмент для вибору маршруту, який враховує одночасно кілька критеріїв:

- час доставки;
- вартість перевезення;
- ризики та надійність;
- екологічність та вплив на довкілля.

Діаграма варіантів використання (рисунок 3.1) відображає взаємодію користувача (логіста/диспетчера) із системою СППР та зовнішніми сервісами (GIS, трафік, погода). Вона демонструє основні функції системи та їхні залежності, що дозволяє формалізувати сценарії роботи та визначити ключові вимоги до інформаційної підсистеми.

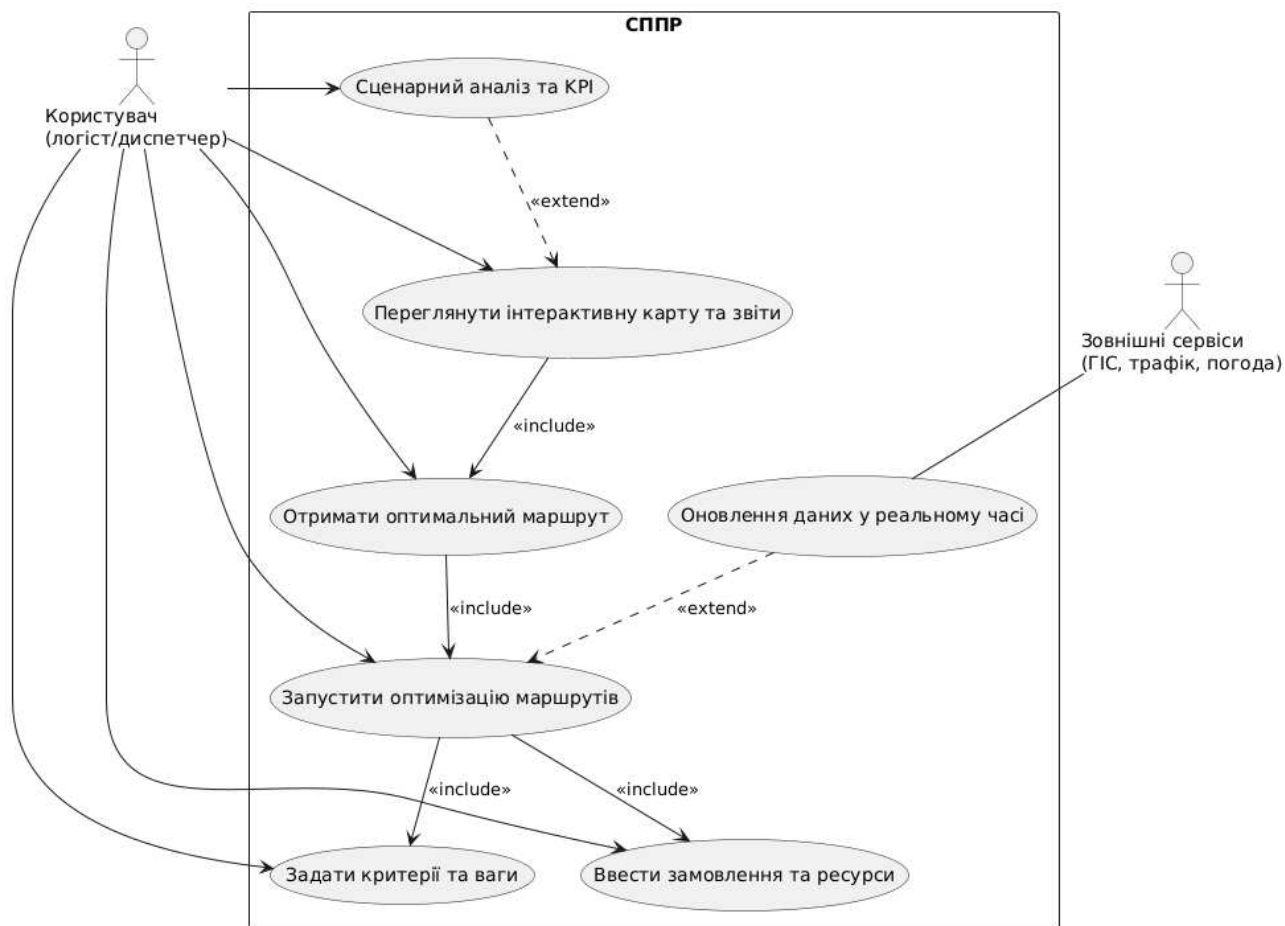


Рисунок 3.1 – Діаграма варіантів використання СПДР для вибору маршруту доставки

На рисунку 3.1 показано, що центральним процесом є оптимізація маршрутів, тоді як інші функції (отримання результатів, візуалізація, сценарний аналіз, оновлення даних) є похідними від нього. Така структура забезпечує логічну послідовність роботи користувача та гнучкість системи у змінних умовах.

Відповідно до діаграми варіантів використання, система реалізує такі ключові функції:

- Введення замовлень та ресурсів – користувач вводить дані про вантажі, транспортні засоби та обмеження.
- Задання критеріїв та ваг – визначаються пріоритети (час, вартість, ризики, екологічність, надійність).

- Запуск оптимізації маршрутів – система генерує кандидати маршрутів та виконує багатокритеріальний аналіз.

- Отримання оптимального маршруту – користувач отримує результат у вигляді найкращого маршруту з поясненням вибору.

- Перегляд інтерактивної карти та звітів – результати відображаються у візуальній формі (карти, графіки, KPI).

- Сценарний аналіз та KPI – користувач може моделювати різні сценарії доставки та оцінювати їх за ключовими показниками ефективності.

- Оновлення даних у реальному часі – система враховує зміни у трафіку, погоді чи ризиках і коригує маршрути.

Логіка «include» та «extend» демонструє залежності між функціями:

- «include» вказує на обов’язкові кроки (наприклад, оптимізація неможлива без введення замовлень та критеріїв).

- «extend» показує додаткові можливості (наприклад, сценарний аналіз розширює перегляд звітів, а оновлення даних у реальному часі розширює оптимізацію).

Очікуваний ефект від використання системи

- Економічний ефект: зниження витрат на паливо та експлуатацію транспорту.

- Часовий ефект: скорочення часу доставки та підвищення точності планування.

- Екологічний ефект: зменшення негативного впливу на довкілля за рахунок оптимізації маршрутів.

- Соціальний ефект: підвищення надійності та безпеки доставки, особливо у гуманітарних та критичних логістичних сценаріях.

3.2 Рішення з інформаційного забезпечення

Інформаційне забезпечення системи підтримки прийняття рішень визначає структуру та організацію даних, необхідних для реалізації функціональних

можливостей системи. Ключовим етапом є формалізація основних сутностей, що відображають об'єкти та процеси логістичної діяльності.

Сутності інформаційної підсистеми

- Клієнт (Customer) – відображає інформацію про замовника доставки.

Атрибути: ідентифікатор клієнта, назва, адреса, контактні дані.

- Замовлення (Order) – описує параметри вантажу та умови його транспортування. Атрибути: ідентифікатор замовлення, клієнт, місце відправлення та призначення, часові вікна, об'єм і вага вантажу.

- Ресурси (Vehicle) – характеризують транспортні засоби, що залучаються до виконання замовлень. Атрибути: ідентифікатор транспортного засобу, вантажопідйомність, тип пального, екологічний показник.

- Маршрут (Route) – визначає оптимізований шлях доставки. Атрибути: ідентифікатор маршруту, транспортний засіб, загальний час, вартість, показники надійності, екологічності та ризику.

- Сегмент маршруту (RouteSegment) – деталізує маршрут як послідовність відрізків. Атрибути: ідентифікатор сегмента, початкова та кінцева точка, відстань, очікуваний час проходження.

- Критерії та ваги (CriteriaWeights) – визначають систему багатокритеріальної оцінки маршрутів. Атрибути: вагові коефіцієнти для часу, вартості, надійності, екологічності та ризиків.

- Запуск оптимізації (OptimizationRun) – фіксує процес оптимізації та результати вибору маршруту. Атрибути: ідентифікатор запуску, критерії, маршрут, дата виконання.

- Дані про трафік (TrafficData) – забезпечують актуалізацію інформації у реальному часі. Атрибути: ідентифікатор даних, сегмент маршруту, час фіксації, рівень завантаженості, погодні умови.

- Звіти та KPI (Reports) – формують аналітичну інформацію для користувача. Атрибути: ідентифікатор звіту, сценарій доставки, ключові показники ефективності, дата формування.

3.3 Рішення з технічного забезпечення

Технічне забезпечення системи підтримки прийняття рішень визначає апаратне та програмно-апаратне середовище, у якому функціонує система, а також вимоги до його продуктивності, надійності та масштабованості.

Вимоги до технічного забезпечення передбачають наявність достатніх обчислювальних ресурсів для виконання багатокритеріальної оптимізації та обробки великих масивів даних у реальному часі. Система повинна забезпечувати безперервність роботи, мати механізми резервування критичних компонентів і підтримувати масштабованість, що дозволяє розширювати обчислювальні потужності та сховища даних у разі зростання кількості замовлень чи транспортних ресурсів. Важливим є також забезпечення інтеграційної сумісності з зовнішніми сервісами (GIS, трафік, погодні дані) та корпоративними системами (ERP, WMS, TMS), а також дотримання вимог інформаційної безпеки, включно із захистом даних клієнтів і використанням сучасних протоколів шифрування та автентифікації. Додатковою вимогою є мобільність та доступність, що передбачає можливість використання системи через веб-інтерфейс або мобільні додатки.

Склад технічного забезпечення включає серверну частину, клієнтську частину, мережеву інфраструктуру та засоби безпеки. Серверна частина охоплює високопродуктивні сервери або кластер для виконання оптимізаційних алгоритмів, систему управління базами даних для зберігання замовлень, ресурсів, маршрутів та результатів оптимізації, а також модулі інтеграції з зовнішніми сервісами. Клієнтська частина представлена робочими станціями логістів і диспетчерів, що мають доступ до веб-інтерфейсу, а також мобільними пристроями для оперативної роботи з даними. Мережеве забезпечення включає канали зв'язку з високою пропускнуою здатністю та захищені протоколи передачі даних. Засоби безпеки охоплюють системи автентифікації та авторизації користувачів, механізми резервного копіювання і відновлення даних, а також інструменти моніторингу та журналювання подій.

Технічне забезпечення СППР формує комплексне середовище, яке поєднує серверні ресурси, клієнтські інтерфейси, мережеву інфраструктуру та засоби захисту інформації, що забезпечує ефективне функціонування системи у реальних умовах логістики.

3.4 Рішення з програмного забезпечення

Програмне забезпечення системи підтримки прийняття рішень (СППР) визначає архітектуру та функціональну структуру, яка забезпечує реалізацію алгоритмів оптимізації, інтеграцію з зовнішніми сервісами та взаємодію з користувачем. Архітектура системи побудована за модульним принципом, що дозволяє розділити її на окремі компоненти та забезпечити гнучкість у масштабуванні й модернізації.

До складу програмного забезпечення входить серверна частина, яка реалізує ядро системи. Вона відповідає за виконання алгоритмів багатокритеріальної оптимізації, управління базою даних, обробку зовнішніх даних та формування результатів. У серверній частині функціонують модулі інтеграції з GIS-сервісами, системами моніторингу трафіку та погодними ресурсами, що забезпечує актуалізацію інформації у реальному часі. Окремий модуль відповідає за реалізацію алгоритмів оптимізації, включаючи евристичні, еволюційні та нейромережеві методи, а також за формування множини Парето-рішень.

Клієнтська частина представлена веб-інтерфейсом та мобільними додатками, які забезпечують доступ користувачів до функціоналу системи. Вона реалізує можливості введення замовлень і ресурсів, задання критеріїв та вагових коефіцієнтів, запуску оптимізації маршрутів, перегляду інтерактивних карт і звітів, а також проведення сценарного аналізу та оцінки KPI. Інтерфейс орієнтований на зручність роботи логістів і диспетчерів, забезпечуючи прозорість процесу прийняття рішень.

Програмне забезпечення також включає модулі безпеки, що реалізують автентифікацію та авторизацію користувачів, захист даних від несанкціонованого доступу та механізми резервного копіювання. Важливим елементом є модуль моніторингу, який забезпечує контроль за станом системи, журналювання подій та своєчасне реагування на відмови.

Діаграма класів

Діаграма класів відображає внутрішню структуру системи підтримки прийняття рішень та формалізує основні об'єкти, їхні атрибути й взаємозв'язки. Вона дозволяє чітко визначити логіку роботи системи, забезпечуючи узгодженість між інформаційною підсистемою та алгоритмічними модулями.

У системі виділено кілька ключових класів. Клас *Customer* описує клієнтів, що формують замовлення, і містить атрибути ідентифікатора, назви, адреси та контактних даних. Клас *Order* деталізує параметри доставки вантажу, включаючи часові вікна, місце відправлення та призначення, об'єм і вагу. Клас *Vehicle* характеризує транспортні ресурси, що використовуються для виконання замовлень, із зазначенням вантажопідйомності, типу пального та екологічного показника.

Центральним елементом є клас *Route*, який описує оптимізований шлях доставки. Він пов'язаний із класом *Vehicle* та асоціюється з класом *Order* через проміжний клас *RouteOrder*, що дозволяє реалізувати відношення «багато-до-багатьох». Клас *RouteSegment* деталізує маршрут як послідовність відрізків, а його атрибути включають початкову та кінцеву точку, відстань і очікуваний час проходження. Для врахування змінних умов середовища використовується клас *TrafficData*, який зберігає інформацію про рівень завантаженості та погодні умови, пов'язану із сегментами маршруту.

Клас *CriteriaWeights* визначає систему багатокритеріальної оцінки, задаючи вагові коефіцієнти для часу, вартості, надійності, екологічності та ризиків. Він використовується у класі *OptimizationRun*, що фіксує процес оптимізації та результати вибору маршруту, включаючи дату запуску та

інтегральну оцінку. Завершує структуру клас Reports, який формує аналітичну інформацію для користувача, зберігаючи сценарії доставки та ключові показники ефективності.

Діаграма класів (рисунок 3.2) демонструє цілісну архітектуру СППР, де кожен клас виконує чітко визначену функцію, а їхні взаємозв'язки забезпечують узгодженість даних і підтримку процесів багатокритеріальної оптимізації. Це дозволяє системі інтегрувати інформацію про клієнтів, замовлення, ресурси та зовнішні фактори, формувати оптимальні маршрути й надавати користувачеві прозорі та обґрунтовані рішення.

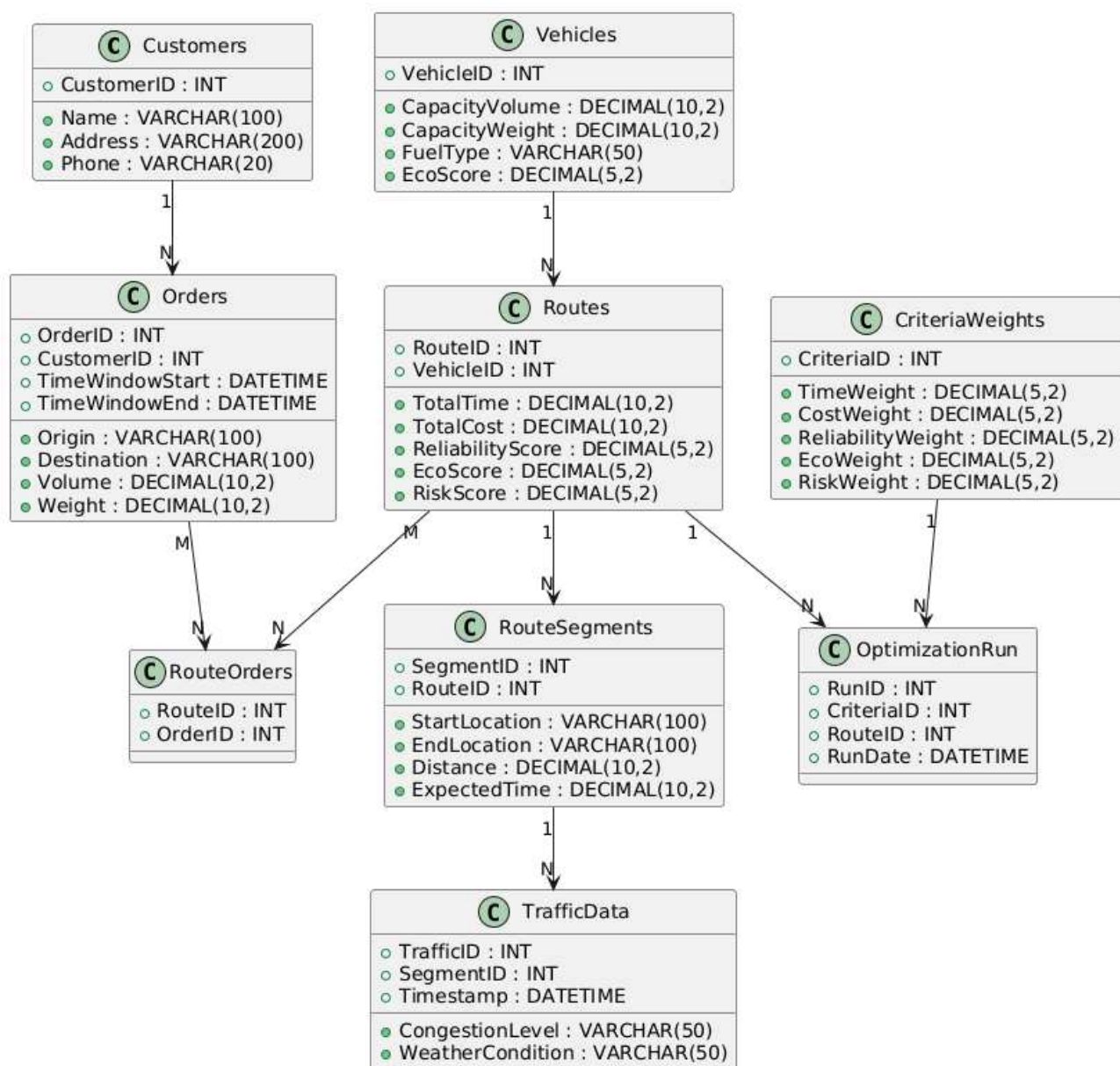


Рисунок 3.2 – Діаграма класів СППР для вибору маршруту доставки

Діаграма діяльності

Діаграма діяльності відображає послідовність дій користувача та системи у процесі вибору оптимального маршруту доставки. Вона демонструє логіку переходів між етапами, починаючи від збору даних і закінчуючи отриманням результатів у вигляді інтерактивної карти та звітів (рисунок 3.3).

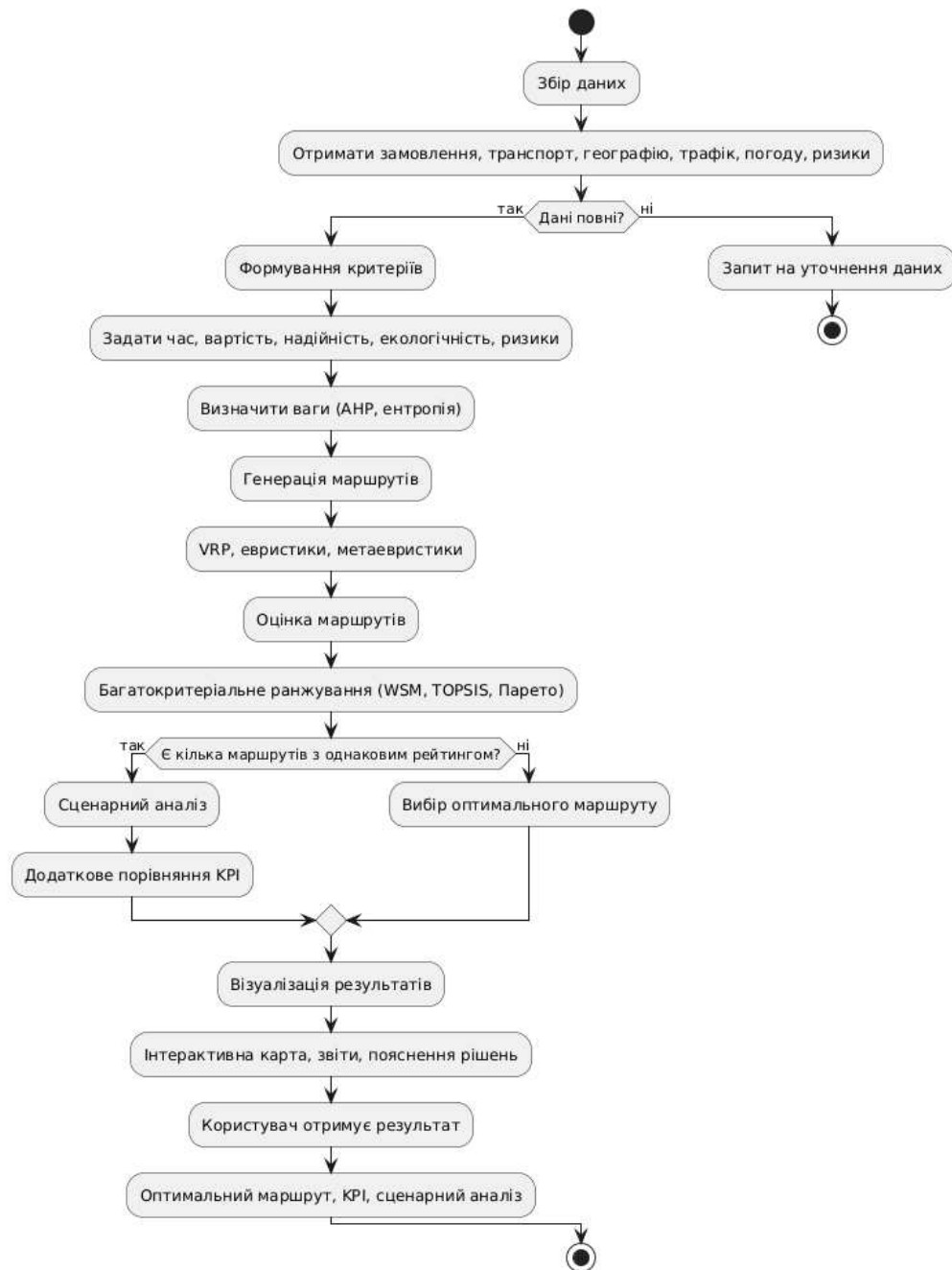


Рисунок 3.3 – Діаграма діяльності процесу оптимізації маршруту у СППР

Процес розпочинається зі збору даних, що включає інформацію про замовлення, транспортні ресурси, географічні умови, трафік, погодні фактори та

ризиків. Якщо дані є неповними, система формує запит на їх уточнення; у випадку повноти даних відбувається формування критеріїв оцінки. Користувач задає параметри часу, вартості, надійності, екологічності та ризиків, після чого визначаються вагові коефіцієнти.

Далі система переходить до генерації маршрутів. Згенеровані маршрути оцінюються за визначеними критеріями, після чого здійснюється багатокритеріальне ранжування. Якщо кілька маршрутів мають однаковий рейтинг, система виконує сценарний аналіз із додатковим порівнянням КРІ; у протилежному випадку відбувається безпосередній вибір оптимального маршруту.

Завершальним етапом є візуалізація результатів у вигляді інтерактивної карти, звітів та пояснень прийнятих рішень. Користувач отримує результат у формі оптимального маршруту, ключових показників ефективності та, за потреби, сценарного аналізу.

Діаграма компонентів

Діаграма компонентів відображає архітектуру системи підтримки прийняття рішень та взаємодію її основних модулів. Вона демонструє, як окремі програмні компоненти інтегруються між собою та із зовнішніми сервісами, забезпечуючи цілісність функціонування системи (рисунок 3.4).

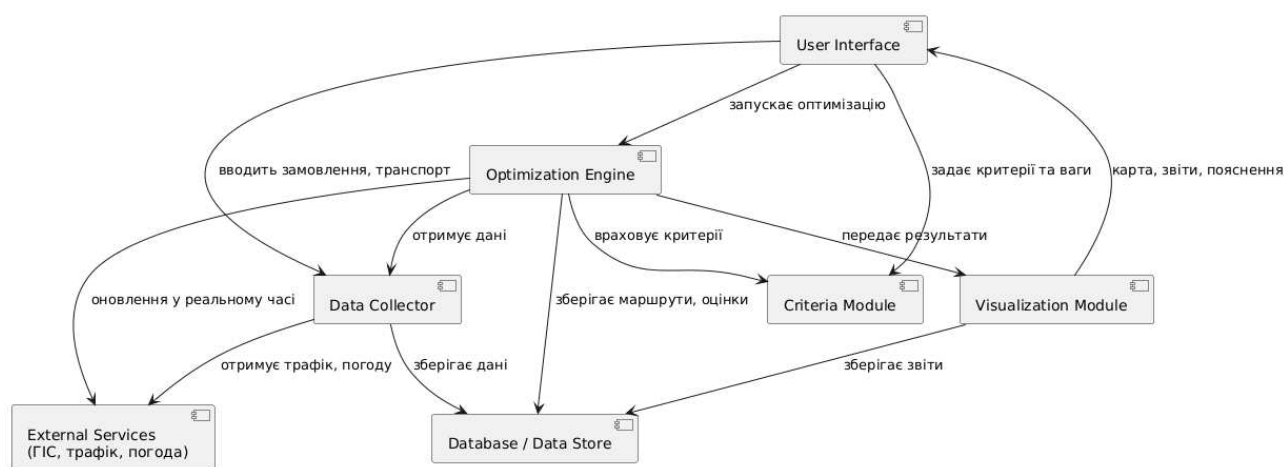


Рисунок 3.4 – Діаграма компонентів СППР для вибору маршруту доставки

У структурі системи виділяється користувацький інтерфейс, який забезпечує взаємодію логіста чи диспетчера із системою. Через нього здійснюється введення замовлень та ресурсів, задання критеріїв і вагових коефіцієнтів, запуск оптимізації та отримання результатів. Інтерфейс пов'язаний із модулем візуалізації, що формує інтерактивні карти, звіти та пояснення прийнятих рішень.

Центральним елементом є оптимізаційний модуль (Optimization Engine), який виконує алгоритми багатокритеріальної оптимізації, генерує маршрути, оцінює їх за критеріями та зберігає результати у базі даних. Він взаємодіє з модулем критеріїв, що відповідає за обробку вагових коефіцієнтів та формування інтегральних оцінок, а також із модулем збору даних (Data Collector), який отримує інформацію про трафік, погодні умови та географічні дані від зовнішніх сервісів.

Зовнішні сервіси (GIS, трафік, погодні ресурси) постачають актуальні дані, що інтегруються у систему через модуль збору даних. Усі дані зберігаються у базі даних, яка є спільним сховищем для замовлень, ресурсів, маршрутів, результатів оптимізації та звітів.

Діаграма переходу станів системи

Діаграма переходу станів демонструє життєвий цикл маршруту доставки у системі підтримки прийняття рішень. Вона відображає (рисунок 3.5) можливі стани маршруту та умови переходу між ними, що дозволяє формалізувати процес управління доставкою та врахувати ризики й непередбачувані ситуації.

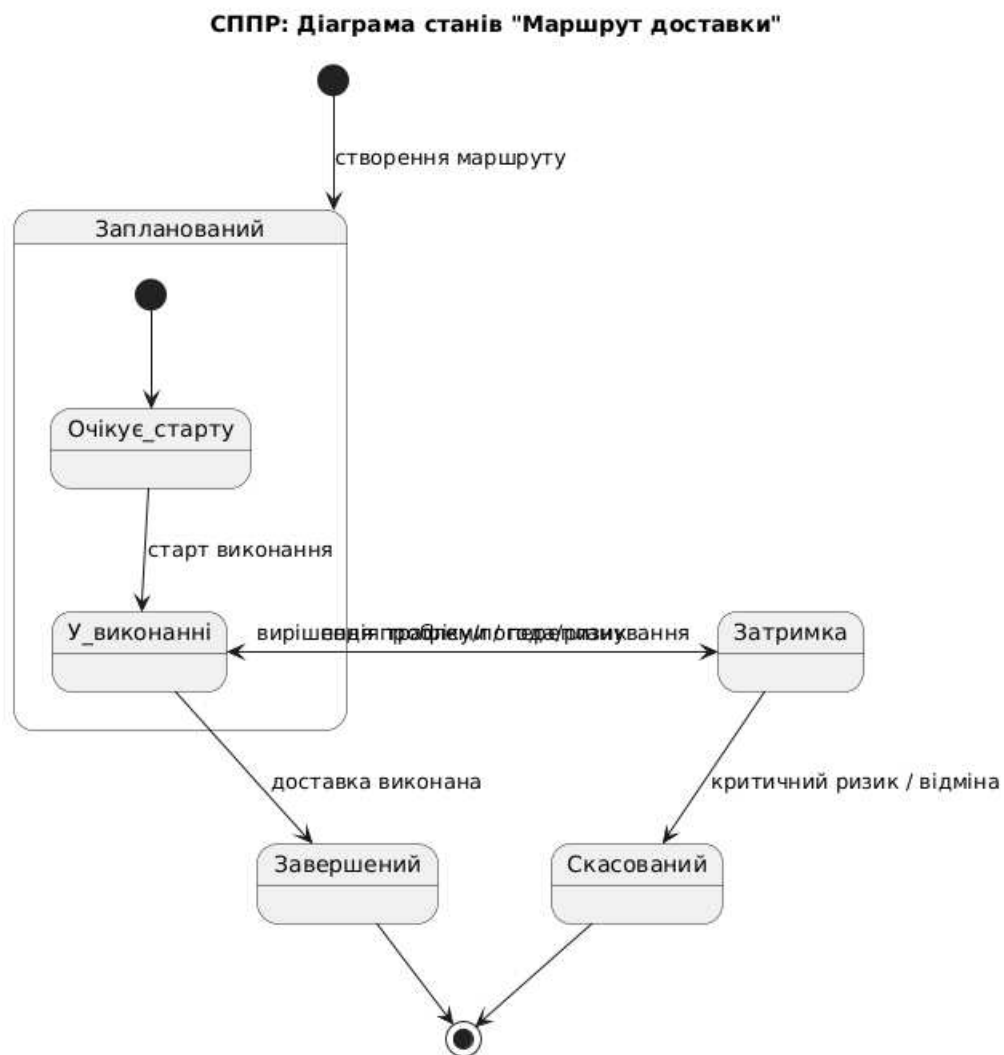


Рисунок 3.5 – Діаграма переходу станів «Маршрут доставки» у СППР

Маршрут створюється у системі та переходить у стан «Запланований», де він очікує старту виконання. Після початку транспортування маршрут переходить у стан «У виконанні», що відображає процес доставки. У цьому стані можливі два варіанти розвитку подій: якщо виникають проблеми, пов’язані з логістикою чи ризиками, маршрут переходить у стан «Затримка»; якщо доставка завершується успішно, він переходить у стан «Завершений».

У випадку критичних ризиків або відміни доставки маршрут може перейти зі стану «Затримка» у стан «Скасований», що означає припинення виконання. Завершальним станом є символ кінця процесу, який фіксує завершення життєвого циклу маршруту незалежно від його результату.

Діаграма розгортання системи

Діаграма розгортання відображає фізичне середовище функціонування системи підтримки прийняття рішень та показує, як її програмні модулі розміщуються на апаратних вузлах і взаємодіють між собою через мережеву інфраструктуру.

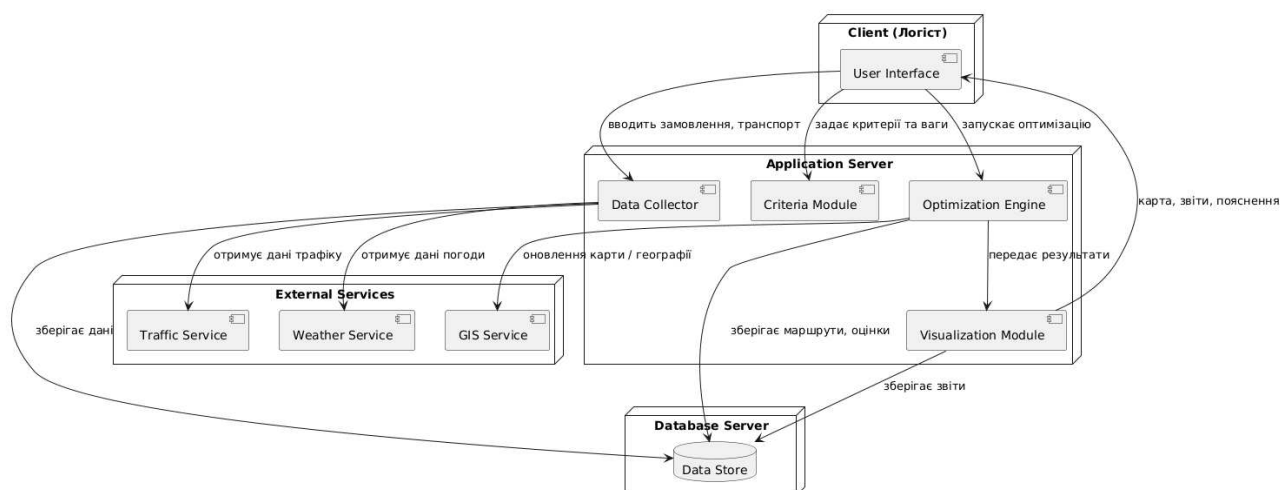


Рисунок 3.6 – Діаграма розгортання СППР для вибору маршруту доставки

У системі виділяється клієнтський рівень, який представлений робочими станціями логістів та диспетчерів, а також мобільними пристроями. Користувачі через інтерфейс вводять замовлення та ресурси, задають критерії й вагові коефіцієнти, запускають оптимізацію та отримують результати у вигляді інтерактивних карт і звітів.

Основні обчислювальні процеси зосереджені на сервері додатків, де функціонують модулі збору даних, критеріїв, оптимізаційний модуль та модуль візуалізації. Сервер додатків забезпечує інтеграцію із зовнішніми сервісами, виконує алгоритми багатокритеріальної оптимізації, формує маршрути та передає результати користувачам.

Зовнішні сервіси постачають актуальні дані, які надходять до модуля збору даних і використовуються для уточнення маршрутів у реальному часі.

Сервер бази даних виконує функцію централізованого сховища, де зберігаються замовлення, транспортні ресурси, маршрути, результати

оптимізації та звіти. Він забезпечує узгодженість даних і доступність для всіх модулів системи.

Висновки до розділу 3

У третьому розділі було здійснено комплексне моделювання системи підтримки прийняття рішень для вибору оптимальних маршрутів доставки. Розглянуто рішення з інформаційного, технічного та програмного забезпечення, а також побудовано низку діаграм, що формалізують архітектуру та логіку функціонування системи.

Інформаційне забезпечення визначило ключові сутності підсистеми, які відображають клієнтів, замовлення, транспортні ресурси, маршрути, сегменти, критерії та результати оптимізації. Це створило основу для узгодженого зберігання даних та їх використання алгоритмічними модулями.

Технічне забезпечення окреслило вимоги до продуктивності, надійності, масштабованості та безпеки системи, а також визначило склад апаратних і мережових компонентів, що забезпечують її функціонування. Програмне забезпечення було представлено як модульна архітектура, що включає оптимізаційний модуль, модуль збору даних, модуль критеріїв, модуль візуалізації та клієнтські інтерфейси.

Побудовані діаграми відобразили різні аспекти системи: від взаємодії користувача із СППР до внутрішньої структури даних, алгоритмічних процесів та фізичного середовища розгортання. Вони забезпечили цілісне уявлення про систему, її функціональні можливості та архітектурну узгодженість.

4 РОЗРАХУНОК ЕКОНОМІЧНОЇ ЕФЕКТИВНОСТІ ВПРОВАДЖЕННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

4.1 Вступ

На даний момент сучасні обчислювальні засоби широко використовуються в повсякденній діяльності людини. Сьогодні обчислювальна техніка дозволяє вирішувати найрізноманітніші задачі: від звільнення людини від ручної праці до розробки складних технічних розрахунків, що не під силу виконати одній людині. Сучасні засоби програмування дозволяють вирішувати задачі, що стояли в нещодавно перед цілими колективами наукових співробітників.

Розвиток нових технологій і ріст застосування обчислювальної техніки привели до широкого застосування ЕОМ для вирішення проблем, пов'язаних з обробкою інформації, її контролем і аналізом. Сьогодні на основі засобів обчислювальної техніки розробляються і впроваджуються різні автоматизовані інформаційні системи (проектування, технологічної підготовки виробництва).

Використання даного програмного продукту приведе до значної економії, що виражається в наступних показниках:

- підвищення продуктивності праці;
- прискорення обробки інформації;
- економія часу на обробку інформації.

Найбільш важливим моментом для розробника, з економічної точки зору, є процес формування вартості програмного продукту. Очевидно, що вона являє собою дуже специфічний товар з безліччю особливостей.

Створення інформаційної системи вимагає одноразових витрат на її розробку, придбання необхідних технічних засобів і поточних витрат на функціонування системи. Економія від функціонування ІС визначається з урахуванням витрат на її експлуатацію. Відношення цієї економії до витрат на створення системи характеризує економічну ефективність капітальних вкладень.

Економічні показники визначаються по діючим на момент розрахунку оптовим цінам, тарифам і ставкам заробітної плати.

4.2 Розрахунок витрат на створення та експлуатацію інформаційної системи

Витрати на розробку системи складаються з витрат на зарплату розробника, на амортизацію ЕОМ, на якій виконується розробка, на експлуатацію цієї ЕОМ, на засоби розробки та витрат на матеріали і комплектуючі.

Виходячи з того, що в середньому заробітна плата розробників програмного забезпечення становить 50 000 грн./місяць, вартість сучасного комп'ютера, який відповідає висунутим вимогам в третьому розділі, становить 30 000 грн. і вартість кіловат-годин електроенергії становить 4,0 грн, розрахуємо вартість розробки системи. Витрати на допоміжні матеріали наведені в таблиці 4.1.

Таблиця 4.1

Витрати на допоміжні матеріали

Найменування витрат	Сума, грн.
Папір	250
Заправлення картриджа до принтера	350
Непередбачені витрати	600
Разом матеріали і комплектуючі вироби	1200

Капітальні витрати на розробку системи K_{π} розраховуються за формулою:

$$K_{\pi} = C + (Z_{\pi p} + Z_e) \cdot T + Z_d \quad (1)$$

де C - первісна вартість програмного продукту, грн.;

$Z_{\pi p}$ - заробітна плата розробника, грн.;

Z_e – експлуатаційні витрати, грн.;

T - тривалість розробки, міс.;

Z_d - додаткові витрати, грн.;

Заробітна плата фахівця розраховується за формулою:

$$Z_{\text{пр}} = Z_{\text{озп}} \cdot \left(1 + \frac{A_c}{100}\right) \quad (2)$$

де $Z_{\text{озп}}$ – основна заробітна плата фахівця;

A_c - відсоток відрахувань на соціальне страхування;

$$Z_{\text{пр}} = 50000 \cdot (1 + 0,22) = 61000$$

Розрахуємо Z_e при споживанні потужності 0,3 Вт, тривалості роботи на місяць рівної $21 \cdot 8 = 168$ годин і вартості кіловат-години електроенергії 4,0 грн.,

$$Z_e = 168 \cdot 4,0 \cdot 0,3 = 201,6 \text{ грн.}$$

Поточні витрати на розробку системи представленні в таблиці 4.2

Таблиця 4.2

Витрати на розробку інформаційної системи

Найменування витрат	Одиниця виміру	Кількість
Тривалість розробки	міс.	4
Заробітна платня	грн.	50 000
Відрахування на соціальні заходи	грн.	11 000
Витрати на допоміжні матеріали	грн.	1200
Витрати на електроенергію	грн.	201,6

Відповідно до формули (1) капітальні витрати на розробку системи складають:

$$K_{\text{п}} = 92401,6 + (61000 + 201,6) \cdot 4 + 1200 = 338408 \text{ грн}$$

Амортизаційні відрахування на устаткування складають 60% балансової вартості в рік:

$$A_{\text{об}} = 35000 \cdot 0,6 = 21000 \text{ грн}$$

У масштабах підприємства річні витрати на основні і допоміжні матеріали визначаються в розмірі 5% вартості основного устаткування:

$$B_{\text{м}} = 35000 \cdot 0,05 = 1750 \text{ грн}$$

Річний обсяг робіт ПЕОМ у годинах визначається в такий спосіб:

$$\Phi_{\text{м}} = 253,3 \cdot T_3$$

де T_z - це середнє місячне завантаження устаткування (близько 7 годин), 253,3
 - середня кількість робочих днів у році:

$$\Phi_M = 253,3 \cdot 7 = 1773,1 \text{ годин}$$

Витрати на електроенергію Z_e при 1773,1 годинах роботи устаткування в рік складуть:

$$Z_e = 1773,1 \cdot 0,3 \cdot 4,0 = 17734,13 \text{ грн}$$

4.3 Економічна ефективність розробки і впровадження системи

Критерієм ефективності створення і впровадження нових засобів автоматизації є очікуваний економічний ефект. Він визначається за формулою:

$$E = \Delta P - E_n \cdot K_n \quad (3)$$

де ΔP - економія від підвищення продуктивності праці;

E_n - нормативний коефіцієнт ($E_n = 0.15$);

K_n - капітальні витрати на впровадження системи.

Якщо користувач системи при економії i -го виду робіт із застосуванням програми економить T_i годин, то підвищення продуктивності праці P_i (у %) визначається за формулою:

$$P_i = \left(\frac{\Delta T_j}{F_j - \Delta T_j} \right) \cdot 100\% \quad (4)$$

де F_j - час, який планувалося затратити користувачем для виконання роботи j -виду до впровадження системи.

В таблиці 4.3 представлена інформація про вид робіт, виконуваних користувачем, та час який він витрачає.

Таблиця 4.3

Таблиця робіт користувача

Вид робіт	До автоматизації, хв. F_j	Економія часу, хв. ΔT_j	Підвищення продуктивності праці P , %
Введення інформації	40	20	100
Проведення розрахунків	90	70	350

Аналіз і вибірка даних	50	25	100
Підготовка та друк звітів	30	15	100

Економія, пов'язана з підвищенням продуктивності праці користувача Р визначимо за формулою:

$$\Delta P = Z_{\pi} \cdot \sum_i \frac{P_i}{100}$$

де Z_{π} - середня заробітна плата менеджера.

Заробітна плата менеджера становить відповідно до формули (2)

$$Z_{\pi} = 27000 \cdot (1 + 0,22) = 32940 \text{ грн.}$$

Відповідно до формули (4) економія, пов'язана з підвищенням продуктивності праці користувача:

$$\Delta P = 32940 \cdot 6,5 = 214110 \text{ грн.}$$

У підсумку, відповідно до формули (4), отримуємо наступну очікувану економічну ефективність:

$$E = 214110 - 0,15 \cdot 338408 = 163348,8 \text{ грн.}$$

Строк окупності системи розраховується за формулою:

$$T = \frac{K_{\pi}}{\Delta P} = \frac{338408}{214110} \approx 1,58 \text{ року}$$

Отже строк окупності системи складає приблизно 1 рік 9 місяці.

Висновки до розділу 4

Було здійснено економічне обґрунтування створення та експлуатації інформаційної системи. Показано, що використання сучасних обчислювальних засобів і програмних технологій дозволяє значно підвищити продуктивність праці, прискорити обробку інформації та зекономити час.

Розрахунок витрат на розробку та функціонування системи підтвердив економічну доцільність її впровадження. Економічна ефективність визначається співвідношенням між витратами на створення та отриманою економією від використання, що свідчить про раціональність капітальних вкладень.

Впровадження СППР у логістичних процесах є економічно виправданим і забезпечує оптимізацію управлінських рішень.

5 ОХОРОНА ПРАЦІ

5.1 Вступ

Охорона праці – це система правил і заходів, які забезпечують безпечну роботу на даному виробництві.

Законодавство про працю містить норми і вимоги з техніки безпеки і виробничої санітарії, норми, що регулюють робочий час і час відпочинку, звільнення та переведення на іншу роботу, норми праці щодо жінок, молоді, гігієнічні норми і правила тощо.

Загальний нагляд за додержанням норм охорони праці покладено на прокуратуру, спеціальний - на професійні спілки. Контроль за безпекою праці здійснюють також державні й відомчі спеціалізовані інспекції.

У сучасному житті комп'ютер широко застосовується в житті людини: і удома, і в офісі, і в магазині, і у виробництві, і навіть в побутовій техніці - іншими словами комп'ютери міцно увійшли до повсякденного життя людей і їх використання постійно збільшується.

В офісах комп'ютери в основному використовуються як допоміжні засоби обробки інформації, і таке введення комп'ютерних технологій принципово змінило характер праці офісних працівників і вимоги до організації і охорони праці.

Недотримання вимог безпеки приводить до того, що через деякий час роботи за комп'ютером співробітник починає відчувати певний дискомфорт: у нього виникають головні болі і різь в очах, з'являються втома і дратівливість. У деяких людей порушується сон, погіршується зір, починають хворіти руки, шия, поперек і тому подібне.

5.2 Аналіз шкідливих та небезпечних факторів

При роботі з комп'ютером шкідливими і небезпечними чинниками є:

- електростатичні поля;
- електромагнітне випромінювання;
- наявність потужних іонізуючих випромінювань;
- локальне стомлення, загальне стомлення;
- стомлюваність очей;
- небезпека поразки електричним струмом;
- пожежонебезпека.

Всі ці чинники викликані двома основними причинами: неправильною роботою за обладнанням і неправильним вибором обладнання.

Негативний вплив комп'ютера на людину є комплексним.

По-перше, комп'ютер (особливо його дисплей, або монітор, сконструйований на основі ЕПТ (електронно-променевої трубки)) є джерелом електростатичного поля; електромагнітних випромінювань в низькочастотному, наднизькочастотних та високочастотному діапазонах (2 Гц - 400 кГц); випромінювання оптичного діапазону (ультрафіолетового, інфрачервоного і видимого світла), рентгенівського випромінювання.

Вчені встановили, що випромінювання низької частоти в першу чергу негативно впливає на центральну нервову систему, викликаючи головний біль / запаморочення, нудоту, депресію, безсоння, відсутність апетиту, стрес.

Низькочастотне електромагнітне поле може з'явитися причиною шкірних захворювань (вугрі, екзема, рожевий лишай та ін), хвороб серцево-судинної системи та шлунково-кишкового тракту; воно впливає на білі кров'яні тіลця, що призводить до виникнення пухлин, у тому числі і злоякісних.

Електростатичне поле великої напруженості здатне змінювати і переривати клітинне розбиток, а також в окремих випадках викликати катаракту.

Дисплеї з ЕПТ завдяки електростатичного поля інтенсивно притягають і збирають пил, а значить, можуть бути причиною появи дерматитів на шкірі обличчя, загострення бронхіальної астми, подразнення слизових оболонок.

По-друге, нерухома і напружена поза.

По-третє, інтенсивна робота з клавіатурою викликає больові відчуття в ліктьових суглобах, передпліччях, зап'ястях, в кистях та пальцях рук.

Відповідно до законодавчих актів площа робочого місця користувача ПК повинна складати не менше 4,5м². У приміщеннях повинні проводитися щоденне вологе прибирання і систематичне провітрювання після кожної години роботи. Устаткування, що виробляє багато шуму (друкуючі пристрої, сканери, сервери і тому подібні), рівні шуму якого перевищують нормативні, повинне розміщуватися поза робочими місцями співробітників.

Робочі столи слід розміщувати так, щоб монітори були орієнтовані бічною стороною до світлових отворів, щоб природне світло падало переважно зліва.

При розміщенні робочих місць відстань між робочими столами має бути не менше 2,0 м-коду, а відстань між бічними поверхнями відеомоніторів - не менше 1,2 м. Робочі місця співробітників, що виконують творчу роботу і що вимагає значної розумової напруги або високої концентрації уваги, рекомендується ізолювати один від одного перегородками заввишки від 1,5 м.

Конструкція робочого столу повинна забезпечувати оптимальне розміщення на робочій поверхні використовуваного устаткування. Висота робочої поверхні столу повинна складати 725 мм, робоча поверхня столу повинна мати ширину 800..1400 мм і глибину 800..1000 мм. Робочий стіл повинен мати простір для ніг заввишки не менше 600 мм, шириною - не менше 500 мм, завглибшки на рівні колін - не менше 450 мм і на рівні витягнутих ніг - не менше 650 мм.

Конструкція робочого стільця або крісла повинна забезпечувати підтримку раціональної робочої пози працівника і дозволяти змінювати позу з метою зниження статичної напруги м'язів шийно-плечової області і спини. Робочий стілець або крісло мають бути підйнятно-поворотними, регульованими по

висоті і кутам нахилу сидіння і спинки, а також відстані спинки від переднього краю сидіння, при цьому регулювання кожного параметра має бути незалежним, легко здійснюваним і мати надійну фіксацію.

Клавіатуру слід розташовувати на поверхні столу на відстані 100..300 мм від краю, зверненого до користувача, або на спеціальній поверхні, відокремленій від основної.

Екран відеомонітора повинен знаходитися від очей користувача на відстані 600..700 мм, але не ближче 500.

5.3 Розрахунок системи штучного освітлення приміщення відділу інформаційних технологій

У відділі інформаційних технологій використовується загальне штучне освітлення, тобто освітлення приміщень штучним світлом за допомогою електричних ламп, яке використовується при недостатньому природному освітленню.

Необхідно зробити розрахунок загального освітлення виробничого приміщення. Довжина $A = 40$ м, ширина $B = 14$ м, висота $H = 3,5$ м. Висота виробничої поверхні $h_p = 1$ м. Для освітлення приймаємо світильники типу НСП 07. Мінімальна освітленість лампи розжарювання за нормами $E_{\min} = 100$ лк. Коефіцієнт відображення стелі $S_n = 70\%$, стін $S_c = 50\%$, робочої поверхні $S_p = 30\%$. Напруга мережі 220 В.

Розрахунок здійснюємо за методом коефіцієнта використання світлового потоку.

Відстань від стелі до робочої поверхні

$$H_o = H - h_p = 3,5 - 1 = 2,5 \text{ м.}$$

Відстань від стелі до світильника

$$h_c = 0,2 \cdot H_o = 0,2 \cdot 2,5 = 0,5 \text{ м.}$$

Висота підвішування світильника над освітлювальною поверхнею

$$h = H_o - h_c = 2,5 - 0,5 = 2 \text{ м.}$$

Висота підвішування світильника над підлогою

$$H_n = h + h_p = 2 + 1 = 3 \text{ м.}$$

Для досягнення найбільшої рівномірності освітлення приймаємо відношення $L/h = 1,5$. Тоді відстань між центром світильників складає

$$L = 1,5 \cdot h = 1,5 \cdot 2 = 3 \text{ м.}$$

Необхідна кількість світильників

$$N = S/L^2 = (40 \cdot 14) / 3^2 = 62,22 \text{ світ.}$$

Приймаємо 63 світильники - 7 рядів по 9 світильників (рисунок 5.1).

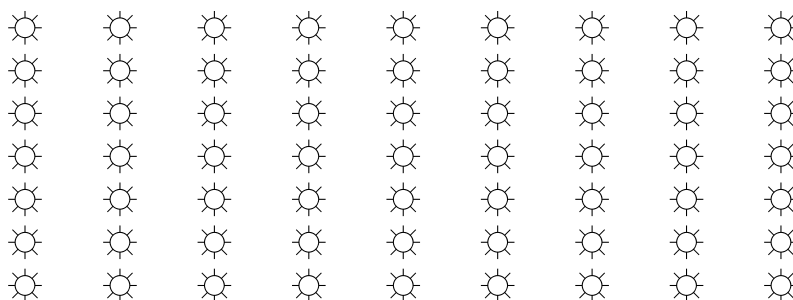


Рисунок 5.1 - Розташування світильників у виробничому приміщенні

Індекс приміщення дорівнює

$$i = A \cdot B / (h \cdot (A + B)) = (40 \cdot 14) / (2 \cdot (40 + 14)) = 560 / 108 = 5,185$$

При $i = 5,185$; $S_n = 70\%$, $S_c = 50\%$, $S_p = 30\%$ для світильника типу НСП 07 коефіцієнт використання світлового потоку $\eta = 0,9$.

Світловий потік однієї лампи

$$\Phi = (E_{min} S K_3 Z) / (N \eta) = (100 \cdot 560 \cdot 1,3 \cdot 1,15) / (63 \cdot 0,9) = 1476,54 \text{ лм,}$$

де $K_3 = 1,3$ за таблицею значень коефіцієнту запасу; $Z = 1,15$ для ламп розжарювання.

За знайденим світловим потоком з таблиці параметрів ламп розжарювання загального призначення вибираємо лампу потужністю 100 Вт, що має світловий потік $\Phi_{\text{л}} = 1450$ лм, найбільш близький до розрахункового.

При цьому фактична освітленість

$$E_{\text{ф}} = E_{min} \cdot (\Phi_{\text{л}} / \Phi_{\text{р}}) = 100 \cdot (1450 / 1476,54) = 98,2 \text{ лк.}$$

Загальна потужність освітлювальної установки

$$P_3 = P_{\text{л}} \cdot N = 100 \cdot 63 = 6300 \text{ Вт} = 6,3 \text{ кВт.}$$

Отже для оптимальної роботи в приміщенні необхідно встановити 7 рядів по 9 світильників. Загальна потужність усіх світильників складає 6,3 кВт.

5.4 Розробка заходів щодо зменшення впливу шкідливих та небезпечних факторів при роботі з персональним комп'ютером

5.4.1 Заходи щодо зменшення впливу

Заходи щодо усунення небезпеки поразки електричним струмом зводяться до правильного розміщення устаткування та електричних кабелів. Інші заходи щодо забезпечення електробезпеки, збігаються з загальними заходами пожежо- та електробезпеки.

В якості профілактичних заходів для забезпечення пожежної безпеки слід використовувати скриту електромережу, надійні розетки з пожежобезпечних матеріалів, силові мережі живлення устаткування виконувати кабелями, розрахованими на підключення в 3-5 разів більшого навантаження, включати й виключати живлення обладнання за допомогою штатних вимикачів. Треба регулярно робити очистку внутрішніх частин комп'ютерів, іншого устаткування від пилу, розташовувати комп'ютери на окремих не спалюваних столах. Для запобігання іскріння необхідно рідше встромляти і виймати штепсельні вилки з розеток.

Екран дисплея повинен бути розташованим перпендикулярно до напрямку погляду. Якщо він розташований під кутом, то стає причиною сутулості. Відстань від дисплея до очей повинна трохи перевищувати звичну відстань між книгою та очима. Перед екраном монітора, особливо старих типів, повинен бути спеціальний захисний екран. При його відсутності треба сидіти на відстані витягнутої руки від монітора. Ще одним моментом, який стосується зору, є необхідність створення неоднорідного поля зору. Для цього можна розвісити на поверхнях (стінах) плакати та картини, виконані у спокійних тонах. Наприклад, пейзажі.

Важливою є форма спинки крісла, яка повинна повторювати форму спини. Висота крісла повинна бути такою, щоб користувач не відчував тиску на куприк або стегна. Крісло бажано обладнати бильцями. Його потрібно встановити так, щоб не треба було тягтися до клавіатури. Періодично користувачу необхідно рухатися, вчасно змінювати положення тіла і робити перерви у роботі.

При напруженій роботі за комп'ютером щогодини необхідно робити перерву на 15 хвилин через кожну годину і треба займатися іншою справою. Декілька разів на годину бажано виконувати серію легких вправ для розслаблення.

5.4.2 Техніка безпеки при роботі на комп'ютері

Загальні положення:

1) При виконанні робіт на персональних комп'ютерах (ПК) необхідно дотримуватись вимог загальної та даної інструкцій з охорони праці.

2) До самостійної роботи на комп'ютерах допускаються особи, які пройшли медичний огляд, навчання з професії, ввідний інструктаж з охорони праці і первинний інструктаж з охорони праці на робочому місці. В подальшому вони проходять повторні інструктажі з охорони праці на робочому місці один раз на півріччя.

3) При роботі на комп'ютерах, до складу яких входять відеодисплейні термінали (ВДТ), на працівників можуть діяти наступні шкідливі та погрожуючі життю і здоров'ю виробничі фактори:

а) фізичні :

- підвищений рівень шуму (від вентилятора мікропроцесора та плат);
- високий рівень напруги в електричній мережі, торкання до якої може викликати нещасний випадок;
- підвищений рівень статичної електрики;
- підвищений рівень електромагнітного випромінювання;
- підвищений рівень напруженості електричного поля;
- пряма і відбита від екрана блискіть;

- несприятливе розподілення яскравості об'єктів в полі зору.

б) психофізіологічні:

- фізичні перевантаження статичної та динамічної дії;
- нервово-психічні перевантаження (розумове перенапруження, перенапруження зорових аналізаторів, монотонність праці, емоційні перевантаження).

4) Основним обладнанням робочого місця користувача ПК є: ВДТ, системний блок, клавіатура, друкуючий пристрій (принтер), зовнішні пристрої пам'яті, сканер, "миша", блок безперервного живлення, робочий стіл, стілець та інше; допоміжним – шафи, полиці, сейф та ін.

Робочі місця з ВДТ повинні бути розташовані на відстані не менше 1,5 м від стіни з вікнами, від інших стін – на відстані 1 м, між собою – на відстані не менше 1,5 м. Відносно вікон робочі місця доцільно розташувати таким чином, щоб природне світло падало на нього збоку, переважно зліва.

5) Робочі місця, обладнані ВДТ, слід розташовувати таким чином, щоб запобігти попаданню в очі прямого світла. Джерела освітлення слід розташовувати з обох боків екрану, паралельно напрямку зору.

Для запобігання світлових відблисків від екрана, клавіатури в напрямку очей користувача від освітлювачів загального призначення або сонячних променів необхідно застосовувати антивідблискові сітки, спеціальні фільтри, захисні козирки, на вікнах регульовані жалюзі.

Фільтри із металевої або нейлонової сітки використовувати не рекомендується тому, що сітка спотворює зображення внаслідок інтерференції світла. Найкраща якість зображення забезпечується скляними поляризаційними фільтрами – вони усувають практично усі відблиски і забезпечують чітке та контрастне зображення.

6) При роботі з текстовою інформацією (в режимі введення даних, редагування тексту і читання з екрану ВДТ) найбільш фізіологічним є зображення чорних знаків на світлому (білому) фоні.

7) Розташовувати ВДТ на робочому місці слід так, щоб поверхня екрану знаходилась в центрі поля зору на відстані 400-700 мм від очей користувача. Пропонується розташовувати елементи робочого місця таким чином, щоб підтримувалась однакова відстань від очей користувача до екрана, клавіатури, пюпітра.

8) Робота комп'ютера супроводжується електромагнітним випромінюванням різних частот і рівнів, інтенсивність котрого зменшується пропорційно квадрату відстані до екрану. Оптимальною для працюючого за ПК є відстань 50 см від екрану ВДТ.

9) Раціональною робочою позою працюючого за ПК вважається таке розташування тіла, при якому ступні працівника розташовані горизонтально на підлозі або на підставці для ніг, стегна зорієнтовані в горизонтальній площині, верхні кінцівки рук – вертикально, кут ліктьового суглоба коливається в межах 70-90 град., зап'ястя зігнуті під кутом, не перевищує 20 град., нахил голови – в межах 15-20 град.

10) Для нейтралізації зарядів статичної електрики в приміщенні, де виконуються роботи на комп'ютерах, в тому числі на лазерних та світлодіодних принтерах, пропонується збільшувати вологість повітря за допомогою кімнатних зволожувачів. Не рекомендується носити одяг з синтетичних матеріалів.

Вимоги безпеки перед початком роботи:

1) Перевірити надійність встановлення апаратури на робочому столі. ВДТ не повинен стояти на краю столу. Повернути ВДТ так, щоб було зручно дивитись на екран – під прямим кутом (а не збоку) і трохи згори вниз, а екран повинен бути нахиленим під невеликим кутом (нижній край ближче до оператора).

2) Перевірити загальний стан апаратури, справність проводки електромережі, з'єднувальних шнурів, штепсельних вилок та розеток, заземлення захисного екрана.

4) Відрегулювати освітленість робочого місця.

5) Відрегулювати і зафіксувати висоту стільця (крісла), зручний для користувача нахил його спинки.

6) Під'єднати до системного блока необхідну апаратуру (принтер, сканер і т.п.). Всі кабелі, що з'єднують системний блок з іншими пристроями слід вставляти і виймати тільки при вимкненому електричному живленні.

7) Увімкнути апаратуру ПК вмикачами на корпусах в такій послідовності: стабілізатор (або блок безперервного живлення), ВДТ, системний блок, принтер (якщо передбачається друкування).

8) Відрегулювати яскравість світіння екрану ВДТ, фокусування, контрастність. Не слід робити яскравість екрану занадто великою, тому що це втомлює очі (і зменшує термін роботи електронно-променевої трубки).

Рекомендується:

- яскравість світіння екрану – не менше 100 кд/ кв.м (кандел на кв.м);
- відношення яскравості екрану ВДТ до яскравості оточуючих його поверхонь в робочій зоні – не більше 3:1;
- мінімальний розмір точки світіння – не більше 0,4 мм для монохромного ВДТ і не більше 0,6 мм для кольорового;
- контрастність зображення знака – не менше 0,8.

9) При появі несправностей комп'ютерної техніки повідомити керівника.

Вимоги безпеки при закінченні роботи на ПК:

1) Закінчити і зберегти в пам'яті ПК файли, які знаходились у роботі. Виконати всі дії для коректного завершення роботи в оперативній системі.

2) Вимкнути принтер та інші периферійні пристрої, вимкнути ВДТ і системний блок. Вимкнути стабілізатор (або блок безперервного живлення при його наявності). Штепсельні вилки витягнути з розеток. Накрити клавіатуру кришкою для попередження попадання в неї пилу.

3) Навести порядок на робочому місці.

4) Вимкнути освітлення та електроприлади.

Вимоги безпеки при аварійних ситуаціях:

1) При раптовому припиненні подачі електричної енергії вимкнути всі пристрої ПК в такій послідовності: периферійні пристрої, ВДТ, системний блок, стабілізатор (або блок безперервного живлення). Витягнути вилки з розеток.

2) При наявності ознак горіння (дим, запах горілого) необхідно вимкнути всі пристрої ПК. Знайти місце загоряння і виконати всі можливі заходи для його ліквідації, попередивши терміново про це керівництво.

3) У випадку виникнення пожежі негайно попередити про це пожежну частину та керівництво, виконати усі можливі заходи по евакуації людей з приміщення і розпочати гасіння пожежі первинними засобами пожежогасіння.

Висновки до розділу 5

Було розглянуто питання охорони праці при роботі з інформаційними технологіями та персональними комп'ютерами. Проведено аналіз шкідливих і небезпечних факторів, що можуть впливати на працівників у процесі експлуатації комп'ютерної техніки, зокрема вплив електромагнітного випромінювання, недостатнє освітлення, тривале статичне навантаження та психоемоційні ризики.

Виконано розрахунок системи штучного освітлення приміщення відділу інформаційних технологій, що забезпечує відповідність санітарним нормам і створює комфортні умови для роботи персоналу. Окрему увагу приділено розробці заходів щодо зменшення впливу шкідливих факторів, включаючи організацію робочого місця, регламентовані перерви, використання ергономічних меблів та дотримання правил техніки безпеки.

Результати розділу підтверджують, що впровадження комплексу організаційних, технічних та санітарно-гігієнічних заходів дозволяє мінімізувати негативний вплив роботи з комп'ютером на здоров'я працівників, підвищити рівень безпеки та створити оптимальні умови для ефективної діяльності у сфері інформаційних технологій.

6 ОХОРОНА НАВКОЛИШНЬОГО СЕРЕДОВИЩА

6.1 Вступ

Охорона навколишнього середовища - система заходів щодо раціонального використання природних ресурсів, збереження особливо цінних та унікальних природних комплексів і забезпечення екологічної безпеки, також це сукупність державних, адміністративних, правових, економічних, політичних і суспільних заходів, спрямованих на раціональне використання, відтворення і збереження природних ресурсів землі, обмеження негативного впливу людської діяльності на навколишнє середовище.

Відносини у галузі охорони навколишнього середовища регулюються Законом України «Про охорону навколишнього природного середовища», а також земельним, водним, лісовим законодавством, законодавством про надра, про охорону атмосферного повітря, про охорону і використання рослинного і тваринного світу та іншим спеціальним законодавством.

Метою охорони навколишнього середовища є протидія негативним змінам у навколишньому середовищі, які мали місце в минулому, відбуваються зараз або можуть бути.

Природоохоронні заходи мають на меті поліпшення стану навколишнього природного середовища або створення умов для цього. Ознаками природоохоронних заходів є:

- підвищення екологічності продукції, що випускається;
- скорочення споживання природних ресурсів на одиницю продукції, що випускається та здійснення господарської діяльності;
- зниження забруднення природних комплексів викидами, стоками, відходами, випромінюваннями;
- зниження концентрації шкідливих речовин у викидах, стоках, відходах;
- поліпшення стану середовища існування людей.

Актуальність охорони довкілля, що перетворилася в глобальну проблему, пов'язана зі зростаючим антропогенним впливом. Це зумовлено демографічним вибухом, урбанізацією, що прискорюється розвитком комунікацій, забрудненням навколишнього середовища відходами, надмірним навантаженням на орні землі, пасовища, ліси, водойми.

6.2 Забруднення навколишнього середовища

При виробництві різних композиційних матеріалів відбувається викид великої кількості техногенних відходів в навколишнє середовище, що суттєво забруднює атмосферу.

Оскільки на підприємствах для виробництва КМ використовуються різноманітні технології, то і в довкілля потрапляють різні забруднюючі речовини.

Основними джерелами утворення і викидів в атмосферу забруднюючих речовин є наступні технологічні процеси: підготовка сировини (збагачування, агломерація), виробництво коксу, доменне виробництво, виробництво сталі, виробництво прокату.

Наприклад бетонні виробництва потребують близько 20 технологічних процесів (розмелювання, висушування, подрібнення, нагрівання в печах, охолодження в баштах, транспортування на стрічкових конвеєрах, транспортування готового продукту тощо), що супроводжуються забрудненням повітря, причому кожен з них має особливості і створює власні технічні проблеми. Найбільш складними технологічними блоками є хімічні підприємства. На одному заводі ряд виробничих процесів може спричинити викиди різних забруднюючих речовин, включно з газоподібними (наприклад, при виробництві азотної, сірчаної кислот, віскози та добрив, а також теплової енергії в котельнях).

Для аглофабрики кількість агломераційних газів, що викидаються в атмосферу, становить - $1,83 \cdot 10^6 \text{ м}^3 / \text{год}$, їх запиленість $8-10 \text{ г} / \text{м}^3$, зміст CO_2 -

4,2%, CO - 0,6-0,9%, SO₂ - 0,05%. Під час агломерації руди відбувається вигорання сірки з тиритів. Сульфідні руди містять до 10% сірки, а після агломерації її залишається лише 0,2 - 0,8%. Викид сірчистого газу при агломерації може бути у розмірі 190 кг. на 1 т. руди, тобто одна стрічкова машина дає близько 700 т. сірчистого газу за добу.

У доменному виробництві мають місце викиди в атмосферу продуктів згоряння з доменної печі (колошникових газів) у процесах завантаження шихти, випуску чавуну, охолодження шлаку. Також разом з доменним газом в атмосферу в невеликих кількостях викидаються також сполуки миш'яку, фосфору, сурми, свинцю, пари ртуті і рідких металів, ціанистий водень і смолисті речовини.

У загальному балансі пилогазових викидів сталеплавільних агрегатів на частку виробництва сталі припадає 75% пилу, 85% оксидів азоту, 90% діоксиду сірки. Запиленість димів газів становить 0,5-2 г / м³.

Значну роль у забрудненні атмосфери грають викиди мартенівських і конверторних сталеплавильних цехів. При веденні мартенівського процесу пил утворюється з металевої шахти (сталевий брухт при її окислюванні, зі шлаку, руди, вапняку, що йдуть на окислювання домішок шихти, і з доломіту, що застосовується для заправлення підлоги печі). У період кипіння сталі виділяються також пари окисів шлаку і металу, газу. Розміри часток пилу не перевищує 3 мкм. Переважна частина пороши мартенівських печей складається з триокису заліза (67%) і триокису алюмінію (8,7%). При безкисневому процесі на 1 т. мартенівської сталі виділяється 3000 - 4000 м³ газів з концентрацією пилу в середньому 0,5 г/м³. У період подачі кисню в зону розплавленого металу пилоутворення багаторазово збільшується, досягаючи 15-52 г/м³. Крім того, плавлення сталі супроводжується вигоранням деякої кількості вуглецю і сірки, у зв'язку з чим у газах мартенівських печей, що відходять при кисневому дутті, міститься до 60 кг. окису вуглецю і до 3 кг. сірчистого газу в розрахунку на 1 т. виданої сталі. Головною особливістю конверторного процесу є одержання сталі з рідкого чавуну без застосування палива. Виплавка сталі за таким принципом

здійснюється в конвертерах ємністю 50,100, 250 і більше тонн шляхом продування рідкого чавуну киснем. Це забезпечує вигорання небажаних домішок, наприклад, марганцю, фосфору і вуглецю, що містяться у чавуні. Процес одержання конверторної сталі носить циклічний характер і при кисневому дутті триває 25-30 хвилин. Димові гази, що утворюються, складаються з часток окисів кремнію, марганцю і фосфору. У складі диму міститься значна кількість окису вуглецю - до 80%. Концентрація пилу в газах що відходять, складає приблизно 15 г/м³. Основна маса пилу -90%, складається з часток розміром 0,2 -1 мкм.

У процесі виробництва коксу мають місце викиди в атмосферу коксового газу. Коксохімічні підприємства забруднюють атмосферне повітря пилом і сумішшю летких сполук. Процес коксування полягає в термічній, без доступу повітря, обробці кам'яновугільної шихти при температурі 900 - 1100вС. При коксуванні 1т. вугілля утворюється 300-320 м³ коксового газу. У його сполуки входять: водень - 50-62% (об'ємних), метан -20-34%, окис вуглецю - 4,5-4,7%, вуглекислий газ -1,8-4%, азот - 5-10%, вуглеводні - 2-2,6% і кисень -0,2-0,5%.

Основна маса коксового газу вловлюється і направляється на хімічну переробку й утилізацію, близько 6% газу надходить в атмосферу внаслідок втрат під час завантаження і вивантаження печей і негерметичності апаратури. У деяких випадках, наприклад, при порушенні режиму роботи батарей коксування чи допоміжних відділень утилізації, в атмосферу викидаються значні кількості неочищеного коксового газу.

6.3 Розробка заходів щодо зменшення забруднення навколишнього середовища

Охорона навколишнього середовища на підприємстві характеризується комплексом вжитих заходів, які спрямовані на попередження негативного впливу людської діяльності підприємства на навколишню природу, що забезпечує сприятливі та безпечні умови людської життєдіяльності. Враховуючи

стрімкий розвиток науково-технічного прогресу, перед людством постала складна задача - охорона найважливіших складових навколишнього середовища (земля, вода, повітря), схильних сильному забрудненню техногенними відходами і викидами, що призводить до окислення ґрунту і води, руйнування озонового шару землі та кліматичним змінам. Промислова політика всього світу привела до таких незворотних і суттєвих змін в навколишньому середовищі, що це питання (охорона навколишнього середовища на підприємстві) став загальносвітовою проблемою і примусив державні апарати розробити довгострокову екологічну політику зі створення внутрішньодержавного контролю.

Основними умовами для поліпшення екології в країні є: раціональне використання, охорона і трата запасів природного резерву, забезпечення безпеки екології та протирадіаційні заходи, підвищення і формування екологічного мислення у населення, а також контроль над екологією в промисловості.

Охорона навколишнього середовища на підприємстві визначила ряд заходів для зниження рівня забруднень, що виробляється підприємствами:

- виявлення, оцінка, постійний контроль та обмеження викиду шкідливих елементів в атмосферу, а також створення технологій і техніки, які охороняють і зберігають природу і її ресурси.
- розробка правових законів, спрямованих на охоронні заходи навколишнього середовища та матеріальне стимулювання виконаних вимог і профілактики комплексу природоохоронних заходів.
- профілактика екологічної обстановки шляхом виділення спеціально відведених територій (зон).

Суттєвого зменшення забруднення навколишньої середовища підприємствами можна досягти впровадженням таких заходів:

- 1) Реалізація засад ресурсозбереження в усьому ланцюзі суспільного виробництва на шляху його інтенсифікації і зниження ресурсомісткості (водо-, земле-, метало-, енерго-);

2) Впровадження безвідходних технологій, що означає (і має своїм наслідком): комплексне використання сировини і енергоносіїв, створення замкнених газо- і водооборотних систем, застосування принципово нових підходів до вилучення, збагачення, перероблення сировини і матеріалів – біотехнологій, геотехнологій.

При вирішенні проблеми безвідходності виробництва слід мати на увазі дві сторони єдиного процесу. Перше – це найбільш раціональних видобуток та повне використання ресурсів і як наслідок зменшення утворення відходів. Друге – це розширення використання відходів, що утворюються. Ці шляхи не виключають, а взаємно доповнюють один одного.

Поняття безвідходних технологій дещо умовним, оскільки повної безвідходності досягти практично не можливо. Більш коректним буде говорити про маловідходні технології. При цьому мається на увазі можливість створення технологічних систем, вплив яких на природу не буде перевищувати її відновлювального потенціалу.

В основі концепції безвідходних технологій лягли три основні положення, а саме:

- створення максимально замкнених систем, організованих за аналогією з природними екосистемами;
- раціональне використання всіх компонентів сировини;
- неминучі впливи на навколишнє середовище не повинні порушувати його функціонування.

Висновки до розділу 6

Було розглянуто питання охорони навколишнього середовища у контексті функціонування інформаційної системи. Проведено аналіз основних джерел забруднення, що можуть виникати при використанні комп'ютерної техніки та супутньої інфраструктури, зокрема утворення електронних відходів, споживання електроенергії та вплив на атмосферу.

Розроблено заходи, спрямовані на зменшення негативного впливу на довкілля. Серед них – раціональне використання енергоресурсів, застосування енергоощадних технологій, організація збору та утилізації електронних відходів, а також впровадження екологічно безпечних рішень у процесі експлуатації системи.

Результати розділу підтверджують, що впровадження комплексу екологічних заходів дозволяє мінімізувати негативний вплив інформаційної системи на навколишнє середовище, сприяє раціональному використанню ресурсів та формує умови для сталого розвитку.

ВИСНОВОК

У магістерській роботі проведено дослідження сучасних методів підтримки прийняття рішень у сфері логістики та доставки товарів. Виконано аналіз процесу пошуку оптимального маршруту з урахуванням багатокритеріальних факторів: часу, вартості, надійності, екологічності та ризиків. Розглянуто існуючі системи підтримки прийняття рішень у транспортно-логістичній галузі, визначено їхні переваги та обмеження.

На основі проведеного аналізу запропоновано модель пошуку оптимального маршруту доставки, реалізовану за методами СППР. Розроблено проект системи, що включає фізичну модель даних, архітектуру компонентів, алгоритми оптимізації та сценарії використання. Створено робочу документацію, яка охоплює UML-діаграми, опис процесів та інтерфейсні рішення.

Результати роботи мають як теоретичну, так і практичну цінність: вони формують основу для впровадження інтелектуальної СППР у сфері логістики, здатної забезпечити прозорість процесів, гнучкість у прийнятті рішень та підвищення ефективності доставки товарів.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

- 1 Балабанов І.Т. Логістика: сучасні концепції та практика. – Київ: КНЕУ, 2020.
- 2 Козловський В.В. Оптимізація транспортних процесів у логістиці. – Львів: ЛНУ, 2022.
- 3 Winston W.L. Operations Research: Applications and Algorithms. – Cengage Learning, 2020.
- 4 Saaty T.L., Vargas L.G. Models, Methods, Concepts & Applications of the Analytic Hierarchy Process. – Springer, 2019.
- 5 Hwang C.L., Yoon K. Multiple Attribute Decision Making: Methods and Applications. – Springer, Reprint 2020.
- 6 Glover F., Kochenberger G., Hao J.K. Handbook of Metaheuristics. – Springer, 2019.
8. Laporte G., Toth P., Vigo D. Vehicle Routing: Problems, Methods, and Applications. – SIAM, 2019.
9. Bertsimas D., Dunn J. Machine Learning Under a Modern Optimization Lens. – Dynamic Ideas, 2019.
10. Bertsekas D.P. Dynamic Programming and Optimal Control. – Athena Scientific, 2020.
11. Korte B., Vygen J. Combinatorial Optimization: Theory and Algorithms. – Springer, 2021.
12. Talbi E.G. Metaheuristics: From Design to Implementation. – Wiley, 2019.
13. Michalewicz Z., Fogel D.B. How to Solve It: Modern Heuristics. – Springer, 2020.
14. Goodfellow I., Bengio Y., Courville A. Deep Learning. – MIT Press, 2019.
15. Піддубна Н. Оптимізація поставок та розподілу в логістичних системах: методичні матеріали. – Одеський національний морський університет, 2025.

16. Методичні матеріали з дисципліни «Сучасні інформаційні технології в управлінні залізничними підрозділами». – Харківський державний університет залізничного транспорту, 2023.
17. Інструкція з охорони праці при роботі на комп'ютерах. – Наказ Мінінформа України №75 від 05 червня 1996 р.
18. Тубальцев А.М. Виробниче освітлення та його розрахунок: Навчальний посібник. – Миколаїв: УДМТУ, 2001.
19. Джигирей В.С. Екологія та охорона навколишнього природного середовища: Навчальний посібник. – Київ: Знання, 2004.
20. Семенов В.Ф., Михайлюк О.Л., Галушкіна Т.П., Крусір Г.В. та ін. Екологічний менеджмент: Навчальний посібник. – Київ: Центр навчальної літератури, 2004.
21. Закон України «Про охорону навколишнього природного середовища». – Постанова Верховної Ради України №1268-ХІІ від 26 червня 1991 р., чинна редакція від 18 листопада 2012 р.

ДОДАТОК А – Технічне завдання

1 Загальні положення

Назва інформаційної системи: «СППР Оптимальний маршрут».

Створюваний програмний продукт являє собою систему підтримки прийняття рішень для вибору оптимальних маршрутів доставки товарів. Система забезпечує багатокритеріальну оптимізацію транспортування з урахуванням часу, вартості, надійності та екологічних факторів.

Підстави для розробки: Наказ №1355-уч від "14" листопада 2025 року «Про затвердження тем магістерських робіт».

2 Призначення та мета створення системи

Система призначена для автоматизації процесів планування та вибору маршрутів доставки товарів від постачальника до споживача.

Метою створення є підвищення ефективності логістичних процесів, скорочення витрат та часу транспортування, а також забезпечення прозорості прийняття рішень у сфері доставки.

3 Характеристика об'єкта автоматизації

Об'єктом автоматизації є процес вибору та оптимізації маршрутів доставки. Система охоплює ведення бази даних замовлень, транспортних ресурсів, постачальників, а також генерацію альтернативних рішень із використанням сучасних методів оптимізації.

4 Цілі та завдання розробки системи

Цілями є зменшення суб'єктивних помилок при плануванні, підвищення достовірності даних, створення єдиної інформаційної системи для логістики та забезпечення її надійності. Завдання включають ведення баз даних замовлень і

ресурсів, генерацію альтернативних маршрутів, багатокритеріальне ранжування та формування звітів.

5 Вимоги до системи

Функціональні вимоги: введення даних про замовлення та транспорт, генерація маршрутів, корекція планів, багатокритеріальна оцінка, збереження та затвердження оптимального маршруту, формування документації.

Вимоги до надійності: безперебійне живлення, використання ліцензійного ПЗ, антивірусний контроль, перевірка коректності введених даних.

Вимоги до експлуатації: можливість роботи у різних умовах, включаючи мобільний доступ, захист від зовнішніх факторів та простота використання для логістів.

Технічні вимоги: серверна платформа для виконання оптимізаційних алгоритмів, клієнтські пристрої для диспетчерів, мережеве середовище для інтеграції з зовнішніми сервісами.

6 Вимоги до програмної документації

Документація повинна містити технічне завдання, опис системи, програму та методики випробувань, інструкцію користувача.

7 Стадії та етапи розробки

Розробка включає передпроектне дослідження, аналіз предметної області, розробку концепції, ескізне та технічне проектування, а також введення системи в дію.

8 Порядок прийому та контролю

Приймання здійснюється представником замовника у присутності розробника відповідно до методики випробувань. За результатами складається акт, який затверджується керівництвом обох сторін.

Додаток В – Текст програми

```
package dssOptimalRoute;

public class Main {

    mainForm mf;

    public static void main(String[] args) {
        Main prog = new Main();
        prog.run();
    }

    public void run() {
        mf = new mainForm();
        mf.setVisible(true);
    }
}

package DssOptimalRoute;

public class mainForm extends javax.swing.JFrame {

    projectForm pf;

    public mainForm() {
        initComponents();
    }

    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated
Code">
    private void initComponents() {

        jPanel1 = new javax.swing.JPanel();
        jPanel2 = new javax.swing.JPanel();
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();
        jComboBox1 = new javax.swing.JComboBox();
        jTextField1 = new javax.swing.JTextField();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE
);

        jPanel1.setBorder(javax.swing.BorderFactory.createTitledBorder(new
```

```

javax.swing.border.LineBorder(new java.awt.Color(0, 0, 0), 1,
true), " IC Ярмарок смаколиків "));

jPanel2.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(0, 0, 0)));

    jLabel1.setText("Загальна інформація:");

    jLabel2.setText("Назва проекту:");

    jLabel3.setText("Автор:");

    javax.swing.GroupLayout jPanel2Layout = new
javax.swing.GroupLayout(jPanel2);
    jPanel2.setLayout(jPanel2Layout);
    jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignmen
t.LEADING)
        .addGroup(jPanel2Layout.createSequentialGroup()
            .addGroup(jPanel2Layout.createParallelGroup(
                javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel1)
                .addGroup(jPanel2Layout.createSequentialGroup()
                    .addComponent(jLabel2)
                    .addPreferredGap(javax.swing.La
                        outStyle.ComponentPlacement.RELATED)
                    .addComponent(jTextField1,
javax.swing.GroupLayout.PREFERRED_SIZE, 162,
javax.swing.GroupLayout.PREFERRED_SIZE))
                .addComponent(jLabel3)
                .addGroup(jPanel2Layout.createSequentialGroup()
                    .addComponent(jComboBox1, 0,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
                .addGap(Short.MAX_VALUE)
            )
        );
    jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignmen
t.LEADING)
        .addGroup(jPanel2Layout.createSequentialGroup()
            .addGroup(jPanel2Layout.createParallelGroup(
                javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel1)
                .addGroup(jPanel2Layout.createSequentialGroup()
                    .addComponent(jLabel2)
                    .addPreferredGap(javax.swing.La
                        outStyle.ComponentPlacement.RELATED)
                    .addComponent(jTextField1,
javax.swing.GroupLayout.PREFERRED_SIZE, 162,
javax.swing.GroupLayout.PREFERRED_SIZE))
                .addComponent(jLabel3)
                .addGroup(jPanel2Layout.createSequentialGroup()
                    .addComponent(jComboBox1, 0,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
                .addGap(Short.MAX_VALUE)
            )
        );
    jPanel2Layout.setDefaultCloseOperation(
        javax.swing.WindowConstants.DISPOSE_ON_CLOSE);
    jPanel2.setVisible(true);
}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
{
    // TODO add your handling code here:
}

```

```

        .addContainerGap()
        .addComponent(jLabel1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATE
D)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout
t.Alignment.BASELINE)
        .addComponent(jLabel2)
        .addComponent(jTextField1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATE
D)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout
t.Alignment.BASELINE)
        .addComponent(jLabel3)
        .addComponent(jComboBox1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );

    jButton1.setText("X");
    jButton1.addActionListener(new
java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent
evt) {
            jButton1ActionPerformed(evt);
        }
    });

    jButton2.setText("Початок оптимізації");
    jButton2.addActionListener(new
java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent
evt) {
            jButton2ActionPerformed(evt);
        }
    });

    javax.swing.GroupLayout jPanel1Layout = new
javax.swing.GroupLayout(jPanel1);
    jPanel1.setLayout(jPanel1Layout);
    jPanel1Layout.setHorizontalGroup(

```

```

jPanell1Layout.createParallelGroup(javax.swing.GroupLayout.Alignmen
t.LEADING)
    .addGroup(jPanell1Layout.createSequentialGroup())
        .addContainerGap()

    .addGroup(jPanell1Layout.createParallelGroup(javax.swing.GroupLayou
t.Alignment.LEADING)
        .addComponent(jPanel2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

    .addGroup(jPanell1Layout.createSequentialGroup()
        .addComponent(jButton1)

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATE
D)
        .addComponent(jButton2,
javax.swing.GroupLayout.DEFAULT_SIZE, 222, Short.MAX_VALUE)))
        .addContainerGap()
    );
jPanell1Layout.setVerticalGroup(

jPanell1Layout.createParallelGroup(javax.swing.GroupLayout.Alignmen
t.LEADING)
    .addGroup(jPanell1Layout.createSequentialGroup())
        .addComponent(jPanel2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATE
D)

    .addGroup(jPanell1Layout.createParallelGroup(javax.swing.GroupLayou
t.Alignment.BASELINE)
        .addComponent(jButton1)
        .addComponent(jButton2))

    .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );

    javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADI
NG)
        .addGroup(layout.createSequentialGroup())

```



```

        .addContainerGap()
        .addComponent(jPanell1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );
    layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jPanell1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        );

    pack();
} // </editor-fold>

private void
jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    this.setVisible(false);
}

private void
jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    pf = new projectForm();
    pf.setVisible(true);
    this.setVisible(false);
}

public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {

        public void run() {
            new mainForm().setVisible(true);
        }
    });
}
// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JComboBox jComboBox1;
private javax.swing.JLabel jLabel1;

```

```

        private javax.swing.JLabel jLabel2;
        private javax.swing.JLabel jLabel3;
        private javax.swing.JPanel jPanel1;
        private javax.swing.JPanel jPanel2;
        private javax.swing.JTextField jTextField1;
        // End of variables declaration
    }
package DssOptimalRoute;

public class projectForm extends javax.swing.JFrame {

    MainForm mf;
    processForm pf;

    public projectForm() {
        initComponents();
    }

    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated
Code">
    private void initComponents() {

        jPanel1 = new javax.swing.JPanel();
        jPanel2 = new javax.swing.JPanel();
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jTextField1 = new javax.swing.JTextField();
        jLabel3 = new javax.swing.JLabel();
        jLabel4 = new javax.swing.JLabel();
        jComboBox1 = new javax.swing.JComboBox();
        jComboBox2 = new javax.swing.JComboBox();
        jLabel5 = new javax.swing.JLabel();
        jComboBox3 = new javax.swing.JComboBox();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();
        jButton3 = new javax.swing.JButton();
        jButton4 = new javax.swing.JButton();
        jPanel3 = new javax.swing.JPanel();
        jScrollPane1 = new javax.swing.JScrollPane();
        jTextArea1 = new javax.swing.JTextArea();
        jLabel6 = new javax.swing.JLabel();
        jCheckBox1 = new javax.swing.JCheckBox();
        jComboBox4 = new javax.swing.JComboBox();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE
);

        jPanel1.setBorder(javax.swing.BorderFactory.createTitledBorder(jav

```

```

ax.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0,
0)), " Оптимізація "));

jPanel2.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(0, 0, 0)));

        jLabel1.setFont(new java.awt.Font("Tahoma", 1, 11)); //
NOI18N
        jLabel1.setText("Інформація про продукцію:");

        jLabel2.setText("Назва:");

        jLabel3.setText("ДСТУ:");

        jLabel4.setText("Матеріал:");

        jLabel5.setText("Модель:");

        javax.swing.GroupLayout jPanel2Layout = new
javax.swing.GroupLayout(jPanel2);
        jPanel2.setLayout(jPanel2Layout);
        jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignmen
t.LEADING)
                .addGroup(jPanel2Layout.createSequentialGroup()
                    .addComponent(jLabel1)

                .addGroup(jPanel2Layout.createParallelGroup(javax.s
wing.GroupLayout.Alignment.LEADING)
                    .addGroup(jPanel2Layout.createSequentialGroup()
                        .addComponent(jLabel5)

                    .addPreferredGap(javax.swing.LayoutStyle.Comp
onentPlacement.RELATED)
                        .addComponent(jComboBox3, 0, 192,
Short.MAX_VALUE))
                    .addComponent(jLabel1)

                .addGroup(jPanel2Layout.createSequentialGroup()
                    .addComponent(jLabel2)

                .addPreferredGap(javax.swing.LayoutStyle.Comp
onentPlacement.RELATED)
                        .addComponent(jTextField1,
javax.swing.GroupLayout.DEFAULT_SIZE, 237, Short.MAX_VALUE))
                .addGroup(jPanel2Layout.createSequentialGroup()
                    .addComponent(jLabel3)

```

```

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATE
D)
                                .addComponent(jComboBox2, 0, 239,
Short.MAX_VALUE))

.addGroup(jPanel2Layout.createSequentialGroup()
                                .addComponent(jLabel4)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATE
D)
                                .addComponent(jComboBox1, 0, 221,
Short.MAX_VALUE))
                                .addContainerGap())
);
jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignmen
t.LEADING)
                                .addGroup(jPanel2Layout.createSequentialGroup()
                                    .addContainerGap()
                                    .addComponent(jLabel11)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATE
D)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout
.Alignment.BASELINE)
                                .addComponent(jLabel2)
                                .addComponent(jTextField1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATE
D)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout
.Alignment.BASELINE)
                                .addComponent(jLabel3)
                                .addComponent(jComboBox2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATE
D)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout
.Alignment.BASELINE)
                                .addComponent(jLabel4)

```

```

        .addComponent(jComboBox1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATE
D)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout
t.Alignment.BASELINE)
        .addComponent(jLabel5)
        .addComponent(jComboBox3,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );

    jButton1.setText("<");
    jButton1.addActionListener(new
java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent
evt) {
            jButton1ActionPerformed(evt);
        }
    });

    jButton2.setText("Проектувати ТО");
    jButton2.addActionListener(new
java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent
evt) {
            jButton2ActionPerformed(evt);
        }
    });

    jButton3.setText("X");
    jButton3.addActionListener(new
java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent
evt) {
            jButton3ActionPerformed(evt);
        }
    });

    jButton4.setText("Підібрати заготівку");
    jButton4.addActionListener(new
java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent
evt) {

```

```

        jButton4ActionPerformed(evt);
    }
});

jPanel3.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(0, 0, 0)));

    jTextArea1.setColumns(20);
    jTextArea1.setRows(5);
    jScrollPane1.setViewportView(jTextArea1);

    jLabel6.setFont(new java.awt.Font("Tahoma", 1, 11)); //
NOI18N
    jLabel6.setText("Альтернативи заготовок:");

    javax.swing.GroupLayout jPanel3Layout = new
javax.swing.GroupLayout(jPanel3);
    jPanel3.setLayout(jPanel3Layout);
    jPanel3Layout.setHorizontalGroup(

jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignmen
t.LEADING)
        .addGroup(jPanel3Layout.createSequentialGroup()
            .addGap(10, 10, 10)
            .addGroup(jPanel3Layout.createParallelGroup(javax.swing.Gro
upLayout.Alignment.LEADING)
                .addComponent(jLabel6)
                .addComponent(jScrollPane1,
javax.swing.GroupLayout.DEFAULT_SIZE, 153, Short.MAX_VALUE))
            .addContainerGap(10, true))
        .addGroup(jPanel3Layout.createSequentialGroup()
            .addComponent(jCheckBox1)
            .addContainerGap(10, true))
        .addGroup(jPanel3Layout.createSequentialGroup()
            .addComponent(jComboBox4,
javax.swing.GroupLayout.PREFERRED_SIZE, 97,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(10, 10, 10)
            .addComponent(jLabel6)
            .addContainerGap(10, true))
    );
    jPanel3Layout.setVerticalGroup(

jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignmen
t.LEADING)
        .addGroup(jPanel3Layout.createSequentialGroup()
            .addGap(10, 10, 10)
            .addGroup(jPanel3Layout.createParallelGroup(javax.swing.Gro
upLayout.Alignment.TRAILING)
                .addComponent(jLabel6)
                .addComponent(jScrollPane1,
javax.swing.GroupLayout.DEFAULT_SIZE, 153, Short.MAX_VALUE))
            .addContainerGap(10, true))
        .addGroup(jPanel3Layout.createSequentialGroup()
            .addComponent(jCheckBox1)
            .addContainerGap(10, true))
        .addGroup(jPanel3Layout.createSequentialGroup()
            .addComponent(jComboBox4,
javax.swing.GroupLayout.PREFERRED_SIZE, 97,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addGap(10, 10, 10)
            .addComponent(jLabel6)
            .addContainerGap(10, true))
    );

```

```

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATE
D, javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayou
t.Alignment.LEADING)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel3Layout.createSequentialGroup())

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayou
t.Alignment.LEADING)
                .addComponent(jCheckBox1)
                .addComponent(jComboBox4,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGap(91, 91, 91))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel3Layout.createSequentialGroup()
                .addComponent(jScrollPane1,
javax.swing.GroupLayout.PREFERRED_SIZE, 102,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addContainerGap()))
);

    javax.swing.GroupLayout jPanel1Layout = new
javax.swing.GroupLayout(jPanel1);
    jPanel1.setLayout(jPanel1Layout);
    jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignmen
t.LEADING)
                .addGroup(jPanel1Layout.createSequentialGroup()
                    .addContainerGap()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayou
t.Alignment.TRAILING, false)
                    .addComponent(jPanel3,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                    .addComponent(jPanel2,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

.addGroup(javax.swing.GroupLayout.Alignment.LEADING,
jPanel1Layout.createSequentialGroup()
                    .addComponent(jButton3)

```

```

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addComponent(jButton1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addComponent(jButton2,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
        .addComponent(jButton4,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

.addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );
    jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel1Layout.createSequentialGroup()
            .addComponent(jPanel2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addComponent(jButton4)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addComponent(jPanel3,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(jButton3)
        .addComponent(jButton1)
        .addComponent(jButton2))
        .addContainerGap())
    );

    javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());

```



```

        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .add(layout.createParallelGroup(
                    javax.swing.GroupLayout.PREFERRED_SIZE,
                    javax.swing.GroupLayout.DEFAULT_SIZE,
                    javax.swing.GroupLayout.PREFERRED_SIZE)
                .add(layout.createParallelGroup(
                    javax.swing.GroupLayout.DEFAULT_SIZE,
                    Short.MAX_VALUE)
                );
            layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .add(layout.createParallelGroup(
                    javax.swing.GroupLayout.DEFAULT_SIZE,
                    javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .add(layout.createParallelGroup(
                    javax.swing.GroupLayout.DEFAULT_SIZE,
                    Short.MAX_VALUE)
                );
            pack();
        } // </editor-fold>

private void
jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
    this.setVisible(false);
}

private void
jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    mf = new mainForm();
    this.setVisible(false);
    mf.setVisible(true);
}

private void
jButton4ActionPerformed(java.awt.event.ActionEvent evt) {

}

private void
jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    pf = new processForm();
    this.setVisible(false);
    pf.setVisible(true);
}

```

```

    }

    public static void main(String args[]) {
        java.awt.EventQueue.invokeLater(new Runnable() {

            public void run() {
                new projectForm().setVisible(true);
            }

        });
    }
    // Variables declaration - do not modify
    private javax.swing.JButton jButton1;
    private javax.swing.JButton jButton2;
    private javax.swing.JButton jButton3;
    private javax.swing.JButton jButton4;
    private javax.swing.JCheckBox jCheckBox1;
    private javax.swing.JComboBox jComboBox1;
    private javax.swing.JComboBox jComboBox2;
    private javax.swing.JComboBox jComboBox3;
    private javax.swing.JComboBox jComboBox4;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JLabel jLabel4;
    private javax.swing.JLabel jLabel5;
    private javax.swing.JLabel jLabel6;
    private javax.swing.JPanel jPanel1;
    private javax.swing.JPanel jPanel2;
    private javax.swing.JPanel jPanel3;
    private javax.swing.JScrollPane jScrollPane1;
    private javax.swing.JTextArea jTextArea1;
    private javax.swing.JTextField jTextField1;
    // End of variables declaration
}
package DssOptimalRoute;

public class processForm extends javax.swing.JFrame {

    public processForm() {
        initComponents();
    }

    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated
Code">
    private void initComponents() {

        jPanel1 = new javax.swing.JPanel();
        jPanel2 = new javax.swing.JPanel();
        jLabel1 = new javax.swing.JLabel();
        jScrollPane1 = new javax.swing.JScrollPane();
        jTextArea1 = new javax.swing.JTextArea();

```

```

jButton1 = new javax.swing.JButton();
jButton2 = new javax.swing.JButton();
jCheckBox1 = new javax.swing.JCheckBox();
jComboBox1 = new javax.swing.JComboBox();
jPanel3 = new javax.swing.JPanel();
jLabel2 = new javax.swing.JLabel();
jScrollPane2 = new javax.swing.JScrollPane();
jTextArea2 = new javax.swing.JTextArea();
jLabel3 = new javax.swing.JLabel();
jScrollPane3 = new javax.swing.JScrollPane();
jTextArea3 = new javax.swing.JTextArea();
jButton3 = new javax.swing.JButton();
jCheckBox2 = new javax.swing.JCheckBox();
jComboBox2 = new javax.swing.JComboBox();
jButton4 = new javax.swing.JButton();
jButton5 = new javax.swing.JButton();
jButton6 = new javax.swing.JButton();
jButton7 = new javax.swing.JButton();
jButton8 = new javax.swing.JButton();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE
);

jPanel1.setBorder(javax.swing.BorderFactory.createTitledBorder(javax
.swing.BorderFactory.createLineBorder(new java.awt.Color(0, 0,
0)), " Оптимізація "));

jPanel2.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(0, 0, 0)));

    jLabel1.setText("Планування випуску:");

    jTextArea1.setColumns(20);
    jTextArea1.setRows(5);
    jScrollPane1.setViewportView(jTextArea1);

    jButton1.setText("Затвердити план");

    jButton2.setText("Корегувати шаблон");

    javax.swing.GroupLayout jPanel2Layout = new
javax.swing.GroupLayout(jPanel2);
    jPanel2.setLayout(jPanel2Layout);
    jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignmen
t.LEADING)
        .addGroup(jPanel2Layout.createSequentialGroup())
            .addContainerGap()

```

```

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING, false)
    .addComponent(jScrollPane1,
javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jLabel1,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jButton2,
javax.swing.GroupLayout.DEFAULT_SIZE, 193, Short.MAX_VALUE)

.addGroup(jPanel2Layout.createSequentialGroup()
    .addComponent(jCheckBox1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(jComboBox1, 0, 170,
Short.MAX_VALUE))
    .addComponent(jButton1,
javax.swing.GroupLayout.DEFAULT_SIZE, 193, Short.MAX_VALUE))
    .addContainerGap()
);
jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanel2Layout.createSequentialGroup()
        .addContainerGap()
        .addComponent(jLabel1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel2Layout.createSequentialGroup())

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
    .addComponent(jComboBox1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

```

```

        .addComponent(jCheckBox1))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATE
D, 40, Short.MAX_VALUE)
        .addComponent(jButton1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATE
D)
        .addComponent(jButton2))
        .addComponent(jScrollPane1,
javax.swing.GroupLayout.DEFAULT_SIZE, 113, Short.MAX_VALUE))
        .addContainerGap()
    );

jPanel3.setBorder(javax.swing.BorderFactory.createLineBorder(new
java.awt.Color(0, 0, 0)));

    jLabel2.setText("Згенеровані альтернативи:");

    jTextArea2.setColumns(20);
    jTextArea2.setRows(5);
    jScrollPane2.setViewportView(jTextArea2);

    jLabel3.setText("Порядок планування:");

    jTextArea3.setColumns(20);
    jTextArea3.setRows(5);
    jScrollPane3.setViewportView(jTextArea3);

    jButton3.setText("Копіювати");

    jButton4.setText("Затвердити план");

    jButton5.setText("Зберегти у якості шаблону");

    javax.swing.GroupLayout jPanel3Layout = new
javax.swing.GroupLayout(jPanel3);
    jPanel3.setLayout(jPanel3Layout);
    jPanel3Layout.setHorizontalGroup(

jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignmen
t.LEADING)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel3Layout.createSequentialGroup()
            .addContainerGap()

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.
Alignment.TRAILING)
                .addComponent(jButton5,
javax.swing.GroupLayout.DEFAULT_SIZE, 317, Short.MAX_VALUE)
            )
    );

```

```

        .addComponent(jButton4,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.DEFAULT_SIZE, 317, Short.MAX_VALUE)

        .addGroup(jPanel3Layout.createSequentialGroup())

        .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.
t.Alignment.LEADING)
            .addComponent(jScrollPane2,
javax.swing.GroupLayout.DEFAULT_SIZE, 140, Short.MAX_VALUE)
            .addComponent(jLabel2,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATE
D)

        .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.
t.Alignment.LEADING)
            .addComponent(jScrollPane3,
javax.swing.GroupLayout.DEFAULT_SIZE, 171, Short.MAX_VALUE)
            .addComponent(jButton3,
javax.swing.GroupLayout.DEFAULT_SIZE, 171, Short.MAX_VALUE)
            .addComponent(jLabel3,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))

        .addGroup(jPanel3Layout.createSequentialGroup())
            .addComponent(jCheckBox2)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATE
D)
            .addComponent(jComboBox2, 0, 294,
Short.MAX_VALUE))
        .addContainerGap())
    );
    jPanel3Layout.setVerticalGroup(

jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.Alignmen
t.LEADING)
        .addGroup(jPanel3Layout.createSequentialGroup())
            .addContainerGap()
            .addComponent(jLabel2)

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATE
D)

        .addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout.
t.Alignment.LEADING, false)

        .addGroup(jPanel3Layout.createSequentialGroup())
            .addComponent(jLabel3)

```

```

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATE
D)
        .addComponent(jScrollPane3)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATE
D)
        .addComponent(jButton3)
        .addComponent(jScrollPane2,
javax.swing.GroupLayout.PREFERRED_SIZE, 147,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATE
D)

.addGroup(jPanel3Layout.createParallelGroup(javax.swing.GroupLayout
t.Alignment.TRAILING)
        .addComponent(jCheckBox2)
        .addComponent(jComboBox2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATE
D)
        .addComponent(jButton4)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATE
D)
        .addComponent(jButton5)

.addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );

jButton6.setText("<");

jButton7.setText("X");

jButton8.setText("Роздрукувати документацію по проекту");

javax.swing.GroupLayout jPanel1Layout = new
javax.swing.GroupLayout(jPanel1);
jPanel1.setLayout(jPanel1Layout);
jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignmen
t.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup())
        .addContainerGap()

```

```

.addGroup(jPanell1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(jPanel3,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
    .addComponent(jPanel2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addGroup(jPanell1Layout.createSequentialGroup()
    .addComponent(jButton6)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(jButton8,
javax.swing.GroupLayout.DEFAULT_SIZE, 247, Short.MAX_VALUE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(jButton7))
    .addContainerGap()
);
jPanell1Layout.setVerticalGroup(

jPanell1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(jPanell1Layout.createSequentialGroup()
        .addComponent(jPanel2,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(jPanel3,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(jPanell1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(jButton6)
    .addComponent(jButton7)
    .addComponent(jButton8))

.addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))

```



```

    );

    javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jPanell1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        );
    layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

        .addGroup(layout.createSequentialGroup()
            .addContainerGap()
            .addComponent(jPanell1,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

            .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
        );

    pack();
} // </editor-fold>

public static void main(String args[]) {
    java.awt.EventQueue.invokeLater(new Runnable() {

        public void run() {
            new processForm().setVisible(true);
        }

    });
}
// Variables declaration - do not modify
private javax.swing.JButton jButton1;
private javax.swing.JButton jButton2;
private javax.swing.JButton jButton3;
private javax.swing.JButton jButton4;
private javax.swing.JButton jButton5;
private javax.swing.JButton jButton6;

```

```
private javax.swing.JButton jButton7;
private javax.swing.JButton jButton8;
private javax.swing.JCheckBox jCheckBox1;
private javax.swing.JCheckBox jCheckBox2;
private javax.swing.JComboBox jComboBox1;
private javax.swing.JComboBox jComboBox2;
private javax.swing.JLabel jLabel1;
private javax.swing.JLabel jLabel2;
private javax.swing.JLabel jLabel3;
private javax.swing.JPanel jPanel1;
private javax.swing.JPanel jPanel2;
private javax.swing.JPanel jPanel3;
private javax.swing.JScrollPane jScrollPane1;
private javax.swing.JScrollPane jScrollPane2;
private javax.swing.JScrollPane jScrollPane3;
private javax.swing.JTextArea jTextArea1;
private javax.swing.JTextArea jTextArea2;
private javax.swing.JTextArea jTextArea3;
// End of variables declaration
}
```