

ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «Оптимізація моделі взаємодії замовлень
постачальника та електронних накладних з динамічним
оновленням товарних позицій для підвищення точності обліку»

на здобуття освітнього ступеня магістра
зі спеціальності 121 Інженерія програмного забезпечення
освітньо-професійної програми «Інженерія програмного забезпечення»

*Кваліфікаційна робота містить результати власних досліджень. Використання
ідей, результатів і текстів інших авторів мають посилання на відповідне джерело*

_____ Антон КОНСВ
(підпис)

Виконав: здобувач вищої освіти групи ПДМ-61

Антон КОНСВ

Керівник: Богдан ХУДІК

доктор філософії (PhD)

Рецензент:

науковий ступінь,
вчене звання

_____ Ім'я, ПРІЗВИЩЕ

Київ 2026

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ**

Навчально-науковий інститут інформаційних технологій

Кафедра Інженерії програмного забезпечення

Ступінь вищої освіти Магістр

Спеціальність 121 Інженерія програмного забезпечення

Освітньо-професійна програма «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ

Завідувач кафедри

Інженерії програмного забезпечення

_____ Ірина ЗАМРІЙ

« _____ » _____ 2025 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

Конєву Антону Олександровичу

1. Тема кваліфікаційної роботи: «Оптимізація моделі взаємодії замовлень постачальника та електронних накладних із динамічним оновленням товарних позицій для підвищення точності обліку»

керівник кваліфікаційної роботи Богдан ХУДІК, доктор філософії, доцент кафедри ІІЗ,

затверджені наказом Державного університету інформаційно-комунікаційних технологій від «30» жовтня 2025 р. № 467.

2. Строк подання кваліфікаційної роботи «19» грудня 2025 р.

3. Вихідні дані до кваліфікаційної роботи: науково-технічна література, нормативні документи, моделі процесів постачання, підходи до формалізації статусних систем, методи синхронізації документів, принципи оптимізації облікових процесів та структура взаємодії документів у сучасних ERP-системах.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

1. Аналіз існуючих підходів до обробки замовлень постачальника та електронних накладних і виявлення причин їхньої неузгодженості.

2. Формалізація вимог та побудова моделі взаємодії документів із використанням статусної логіки та механізмів динамічного оновлення товарних позицій.

3. Розробка та програмна реалізація алгоритмів синхронізації даних між замовленням і накладною з урахуванням часткових поставок.
4. Проведення експериментального дослідження, оцінка точності та ефективності запропонованої моделі та формування висновків.
5. Перелік ілюстративного матеріалу: *презентація*
1. Мета об'єкт та предмет дослідження.
 2. Актуальність роботи.
 3. Структура моделі.
 4. Алгоритм формування картки постачальника
 5. Практичний результат.
 6. Результати моделювання/порівняння.
 7. Висновки.
 8. Публікації та апробація роботи.
6. Дата видачі завдання «31» жовтня 2025 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз наявної науково-технічної літератури	31.10-01.11.2025	
2	Вивчення матеріалів для аналізу моделей взаємодії замовлення постачальника та електронної накладної	02.11-05.11.2025	
3	Дослідження підходів до синхронізації товарних позицій та їх впливу на точність обліку	06.11-08.11.2025	
4	Аналіз особливостей функціонування статусних моделей у процесах обробки постачальницької документації	10.11-14.11.2025	
5	Дослідження методів формалізації процесів постачання та оптимізації взаємодії документів	15.11-25.11.2025	
6	Застосування формальної моделі статусної логіки та алгоритмів синхронізації у взаємодії документів “замовлення-накладна”	26.11-02.12.2025	
7	Оформлення роботи: вступ, висновки, реферат	03.12-14.12.2025	
8	Розробка демонстраційних матеріалів	15.12-19.12.2025	

Здобувач вищої освіти

(підпис)

Антон КОНЄВ

Керівник

кваліфікаційної роботи

(підпис)

Богдан ХУДІК

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня магістра: __80_ стор., _2__ табл., _13_ рис., 37 джерел.

Мета роботи – підвищення точності товарного обліку шляхом оптимізації моделі взаємодії між замовленням постачальника та електронною накладною з використанням динамічного оновлення товарних позицій.

Об'єкт дослідження – процес інформаційної взаємодії між замовленням постачальника та електронною накладною.

Предмет дослідження – процес інформаційної взаємодії між замовленням постачальника та електронною накладною.

Короткий зміст роботи: У роботі проведено аналіз проблем узгодженості між замовленням постачальника та електронною накладною, визначено основні причини розбіжностей у товарних позиціях та недоліки традиційних підходів до їх синхронізації. Розроблено оптимізовану модель інформаційної взаємодії документів, що ґрунтується на формалізованій статусній логіці та інваріантах цілісності, які забезпечують коректне та передбачуване оновлення даних. Запропоновано механізми динамічної синхронізації товарних позицій із підтримкою часткових поставок та реалізовано програмний модуль, що інтегрує ці алгоритми в єдину облікову систему. Проведено експериментальне дослідження, яке підтвердило підвищення точності, стабільності та прозорості обробки постачальницької документації завдяки використанню розробленої моделі.

КЛЮЧОВІ СЛОВА: ЗАМВОЛЕННЯ ПОСТАЧАЛЬНИКА, ЕЛЕКТРОННА НАКЛАДНА, ДИНАМІЧНЕ ОНОВЛЕННЯ ПОЗИЦІЙ, СТАТУСНА МОДЕЛЬ, СИНХРОНІЗАЦІЯ ДАНИХ, ОПТИМІЗАЦІЯ ОБЛІКУ.

ABSTRACT

Text part of the master's qualification work: 88 pages, 2 pictures, 13 table, 37 sources.

The purpose of the work is to develop an optimized model for synchronizing data between supplier purchase orders and electronic invoices by means of formalizing status logic and ensuring dynamic updating of item positions to improve the accuracy of accounting processes.

Object of research – processes of interaction between supplier purchase orders and electronic invoices within information systems.

Subject of research – methods, models, and mechanisms of synchronization of item positions between related accounting documents based on status logic and integrity constraints.

Summary of the work: The work analyzes the problems of data inconsistency between supplier purchase orders and electronic invoices and substantiates the need for developing a formalized model that ensures reliable synchronization of item positions. The study identifies the main causes of discrepancies and limitations of traditional approaches to processing supplier documentation. An optimized interaction model based on status logic and integrity invariants has been designed to provide predictable and accurate updates of item data, including support for partial deliveries. Algorithms for dynamic synchronization of item positions were developed and implemented within a software module integrated into the accounting workflow. Experimental results confirmed improved accuracy, stability, and transparency of data processing achieved through the proposed solution.

KEYWORDS: SUPPLIER ORDER, ELECTRONIC INVOICE, DYNAMIC ITEM UPDATE, STATUS MODEL, DATA SYNCHRONIZATION, ACCOUNTING OPTIMIZATION.

ЗМІСТ

ВСТУП	11
РОЗДІЛ 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ ПІДХОДІВ ДО ВЗАЄМОДІЇ “ЗАМОВЛЕННЯ ПОСТАЧАЛЬНИКА - ЕЛЕКТРОННА НАКЛАДНА”	14
1.1 Характеристика бізнес-процесу взаємодії “замовлення постачальника – електронна накладна”	14
1.2 Аналіз сучасних інформаційних систем і моделей обробки документації постачальників	15
1.3 Проблематика узгодженості товарних позицій та джерела розбіжностей між документами	17
1.4 Обмеження традиційних моделей та необхідність формалізації процесу синхронізації	19
1.5 Вимоги до оптимізованої моделі взаємодії із підтримкою динамічного оновлення товарних позицій	21
РОЗДІЛ 2 АНАЛІЗ ІСНУЮЧИХ ПІДХОДІВ ТА ФОРМАЛІЗАЦІЯ МОДЕЛІ ВЗАЄМОДІЇ “ЗАМОВЛЕННЯ ПОСТАЧАЛЬНИКА - ЕЛЕКТРОННА НАКЛАДНА”	24
2.1 Аналіз існуючих моделей взаємодії замовлення та електронної накладної	24
2.2 Формалізація структури даних та моделі взаємодії «замовлення — накладна»	29
2.2.1 Структура множин товарних позицій	30
2.2.2 Модель синхронізації замовлення та накладної	30
2.2.3 Залежності між замовленням та накладною	31
2.2.4 Формалізація статусної моделі	31
2.3 Формальна специфікація алгоритму синхронізації даних	32
2.3.1 Інваріанти та базовані співвідношення	32
2.3.2 Система правил алгоритму	33
2.3.3 Модель поведінки системи як скінченний автомат	34
2.3.4 Алгоритм взаємодії замовлення постачальника та електронної накладної	37
2.3.5 Сценарії роботи алгоритму синхронізації	38
2.4 Моделювання потоків даних у системі	40
2.4.1 Контекстна модель потоків даних	41
2.4.2 Деталізована модель потоків даних процесу обробки замовлення постачальника	42
2.4.3 Критичні точки обробки даних у процесі взаємодії замовлення та накладної	45
2.5 Моделювання структури даних системи	46

2.5.1	Модель сутностей та зв'язків	46
2.5.2	Опис сутностей та логіки зв'язків	48
2.6	Моделювання функціональних можливостей системи	49
РОЗДІЛ 3 РЕАЛІЗАЦІЯ МОДЕЛІ ТА ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ		53
3.1	Програмні засоби та середовище розробки	53
3.2	Архітектура розробленої системи	55
3.2.1	Логічна архітектура системи	55
3.2.2	Структура основних модулів	57
3.2.3	Архітектурні рішення та шаблон MVC	59
3.2.4	Структура бази даних	62
3.3	Алгоритмічне забезпечення системи	64
3.4	Експериментальна оцінка ефективності розробленої моделі	67
3.4.1	Мета та методика експерименту	67
3.4.2	Результати тестування та порівняння методів	68
3.4.3	Оцінка ефективності оптимізованої моделі на основі експериментальних даних	70
3.5	Інтерфейс програмного забезпечення	72
ВИСНОВКИ		77
ПЕРЕЛІК ПОСИЛАНЬ		79
ДОДАТОК А		83
ДОДАТОК Б. ЛІСТИНГ ОСНОВНИХ МОДУЛІВ		90

ВСТУП

Сучасні підприємства дедалі частіше стикаються з проблемою ефективної взаємодії між процесами постачання, обліку товарів та документообігу. Традиційні системи управління закупівлями і замовленнями постачальників часто функціонують окремо від модулів обліку або документообігу, що призводить до дублювання даних, невідповідності між замовленнями та накладними, а також ускладнює контроль залишків товарів.

В умовах цифрової трансформації бізнесу все більшого значення набуває автоматизація процесів обміну інформацією між замовленнями постачальників та електронними накладними. Одним із ключових напрямів удосконалення таких систем є впровадження динамічних моделей оновлення товарних позицій, що дозволяють синхронізувати дані у реальному часі, мінімізуючи людський фактор і підвищуючи точність обліку.

Мета переддипломної практики полягає у оптимізації моделі взаємодії замовлень постачальника та електронних накладних із можливістю динамічного оновлення товарних позицій для підвищення точності обліку та ефективності управління постачаннями.

Об'єктом дослідження є процес інформаційної взаємодії між замовленнями постачання та електронними накладними у веб-системах управління постачаннями.

Предметом дослідження – методи та програмні засоби оптимізації моделі взаємодії замовлень постачання і накладних із динамічною синхронізацією товарних позицій.

Завдання переддипломної практики включають:

1. Аналіз існуючих підходів до управління постачаннями та електронним документообігом;
2. Розробку структури взаємодії між замовленнями постачальників та накладними;
3. Створення моделі динамічного оновлення товарних позицій та контролю змін;

4. Проектування бази даних і логіки статусів постачальників;
5. Реалізацію веб-інтерфейсу для роботи із замовленнями та накладними;
6. Оцінку ефективності запропонованої моделі з точки зору точності обліку та зменшення часу обробки документів.

Методи дослідження, що застосовуються у роботі:

- методи системного аналізу бізнес-процесів постачання;
- моделювання структури баз даних і потоків даних;
- розробка програмних модулів засобами веб-технологій;
- експериментальна перевірка точності та узгодженості даних;
- порівняльний аналіз ефективності запропонованої моделі з традиційними методами обліку.

Інформаційною базою дослідження є сучасні наукові публікації, стандарти електронного документообігу, матеріали щодо оптимізації логістичних процесів та інформаційні системи управління постачаннями (ERP, CRM, SCM).

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ІСНУЮЧИХ ПІДХОДІВ ДО ВЗАЄМОДІЇ “ЗАМОВЛЕНЬ ПОСТАЧАЛЬНИКА - ЕЛЕКТРОННИХ НАКЛАДНИХ”

1.1 Характеристика бізнес-процесу взаємодії “замовлень постачальника – електронних накладних”

Бізнес-процес постачання на підприємстві передбачає послідовну обробку інформації, яка стосується планування закупівель, узгодження умов постачання, фактичного отримання товарів та їх подальшого складського й фінансового обліку. Основними документами, що забезпечують цей процес, є замовлення постачальника та електронна накладна. Замовлення визначає прогнозовану потребу у товарних позиціях, формує перелік продукції, узгодженої до постачання, і слугує вихідною точкою для взаємодії між підприємством та зовнішнім контрагентом. Електронна накладна відображає фактичний результат поставки, містить кількісні та цінові дані щодо товарів, які були реально отримані, та є основою для подальшого бухгалтерського й складського обліку.

Типовий процес взаємодії починається зі створення замовлення постачальника, у межах якого менеджер формує перелік товарних позицій відповідно до потреб підприємства чи його клієнтів. На цьому етапі кількість, вартість і характеристики товарів можуть змінюватися, оскільки документ перебуває у стадії підготовки. Після відправлення замовлення зовнішньому контрагенту починається етап узгодження, протягом якого постачальник підтверджує можливість поставки кожної позиції, може уточнювати обсяг, ціну або повідомляти про часткову доступність товарів. Результати такого узгодження формують узгоджений набір товарних позицій, який має залишатися незмінним, оскільки саме на нього спирається подальший документообіг.

Електронна накладна створюється після остаточного узгодження умов постачання та відображає фактичні дані поставки. Вона містить перелік товарних позицій, фактичну кількість кожного товару, ціну та інші параметри, що характеризують здійснену операцію. Важливо, що накладна є документом фінального характеру в частині товарообігу: після її формування вона впливає на складські залишки, облікові показники та фінансову звітність підприємства.

Проблемність взаємодії між замовленням та накладною виникає у випадках, коли бізнес-процес не має формалізованої логіки та контролю на кожному етапі життєвого циклу документа. Якщо замовлення продовжує редагуватися після узгодження або накладна формується незалежно, без чіткого зв'язку із початковими позиціями, виникають розбіжності у кількостях, невідповідності у цінах, некоректні облікові дані та складності в аналітиці. Часткові поставки є додатковим чинником ускладнення, оскільки вони вимагають можливості зафіксувати фактичний обсяг постачання без порушення структури замовлення.

У реальних умовах ефективність процесу значною мірою залежить від того, наскільки система обробки документів забезпечує незмінність узгоджених позицій, контроль можливих дій користувача на кожному етапі, цілісність даних та автоматичну синхронізацію інформації між документами. Коли бізнес-процес формалізований, зменшується кількість помилок, підвищується якість обліку та прискорюється обробка документів.

1.2 Аналіз сучасних інформаційних систем і моделей обробки документації постачальників

Сучасні інформаційні системи, що підтримують процеси закупівель та постачання, реалізують різні підходи до організації взаємодії між замовленням постачальника та електронною накладною. У більшості випадків ці підходи історично сформувалися як частина загальних облікових систем, а не як окремі оптимізовані моделі синхронізації двох взаємопов'язаних документів. Це призводить до того, що кожна система забезпечує власну логіку обробки

документів, яка має певні переваги, але також зберігає властиві їй обмеження щодо узгодженості даних, фіксації позицій та дотримання процесної логіки.

Найпоширенішими у практиці підприємств є ERP-системи, такі як SAP, Odoo, Microsoft Dynamics та 1С, які передбачають класичну модель життєвого циклу закупівлі. У цих системах замовлення постачальнику формується як окремий документ, який містить перелік товарних позицій, прогнозовану кількість та контрактні дані. Накладна створюється на основі замовлення, однак при цьому зберігає можливість незалежного редагування. Така особливість призводить до ситуацій, коли зміни, внесені у накладну, не відображаються у замовленні, а зміни у замовленні після узгодження не мають жодних обмежень. У складних сценаріях, таких як часткові поставки або коригування ціни, ERP-системи змушують користувача створювати надлишкові документи або вручну звіряти відповідність позицій.

У веб-орієнтованих системах, що розробляються для малого та середнього бізнесу, часто використовується спрощена модель CRUD, у межах якої кожен документ — замовлення, накладна, акт або звіт — існує як автономна сутність. Зазвичай система не містить механізмів контролю відповідності між документами, тому узгодження позицій покладається на оператора. У таких системах структура товарних позицій легко порушується, оскільки редагування можливе у будь-який момент, незалежно від етапу обробки документа. Це створює передумови для втрати цілісності даних та значно збільшує ймовірність помилок.

Документоорієнтовані платформи, орієнтовані на маршрутизацію та workflow-процеси, забезпечують зручний інструментарій для керування статусами документа, проте не мають узагальнених механізмів підтримки взаємної синхронізації між замовленням і накладною. У таких системах кожен документ у ланцюзі постачання зберігає власну копію товарних позицій, що знову ж таки формує незалежні набори даних. У разі відхилення фактичної кількості товарів від запланованих коригування виконуються окремо для кожного документа, що не гарантує збереження узгодженості.

Особливої уваги потребують випадки застосування таблиць або простих електронних інструментів на кшталт Excel чи Google Sheets. У таких випадках замовлення та накладна приймають форму окремих таблиць, які не зв'язані між собою формальними відношеннями та не мають механізмів перевірки правильності даних. Відсутність контрольованої структури та автоматизованих обмежень призводить до численних неточностей, дублювання даних, складнощів під час інвентаризації та неможливості відстежити історію змін.

Узагальнюючи аналіз, можна зазначити, що жоден із розглянутих підходів не забезпечує повної відповідності між замовленням та накладною на рівні структури даних. У більшості систем форма накладної лише частково залежить від замовлення, а сам процес формування не передбачає жорсткого зв'язку між узгодженими позиціями та фактичними показниками поставки. Також відсутня можливість гарантовано зафіксувати позиції після узгодження, що створює ризик втручання у дані на критичних етапах документообігу. Додатковою проблемою є незалежність статусних механізмів, що не завжди відображають реальну логіку бізнес-процесу та допускають некоректні переходи між етапами.

1.3 Проблематика узгодженості товарних позицій та джерела розбіжностей між документами

Узгодженість товарних позицій між замовленням постачальника та електронною накладною є критично важливим аспектом інформаційного забезпечення процесів постачання. Незважаючи на логічну взаємозалежність цих документів, у практиці підприємств між ними часто виникають розбіжності, які ускладнюють як складський, так і фінансовий облік. Джерела таких розбіжностей мають як організаційний, так і технічний характер, проте основною причиною є відсутність формалізованого механізму зв'язку між документами у більшості існуючих систем.

Одним із найбільш поширених джерел проблем є незалежність структур даних у замовленні та накладній. Коли кожен документ зберігає власні копії

товарних позицій, то навіть незначне коригування одного з них призводить до порушення цілісності. У випадках, коли менеджер змінює кількість або ціну в замовленні після узгодження або коли фактичні дані накладної вводяться вручну без автоматичної перевірки, система не гарантує відповідності між документами. Відсутність єдиного джерела правди ускладнює аудит, призводить до неточностей у залишках та створює ризики для аналітичних звітів.

Ще однією причиною розбіжностей є відсутність статусної моделі, яка б визначала, на якому етапі життєвого циклу документа допускається редагування товарних позицій. У багатьох системах замовлення можна змінювати навіть після того, як постачальник підтвердив поставку, що суперечить логіці процесу. Неконтрольоване редагування руйнує зв'язок між запланованою та фактичною кількістю товарів, оскільки система не запобігає внесенню коригувань у моменти, коли дані вже не можуть бути змінені без порушення облікових принципів.

Часткові поставки є додатковим складним випадком, що вимагає особливого механізму синхронізації. Якщо постачальник не може поставити повну кількість товару, накладна має відобразити це коректно, без зміни узгоджених позицій замовлення. У багатьох системах такий сценарій не підтримується формально, через що менеджери змушені вручну змінювати або замовлення, або накладну. Це створює значні ризики появи помилок і негативно впливає на подальші операції інвентаризації.

Окремої уваги потребує проблема дублювання товарних позицій. У випадках, коли накладна формується вручну або існує можливість додати позиції, яких не було у замовленні, виникають ситуації невідповідності, що не можуть бути виявлені автоматично. Подібні випадки свідчать про відсутність механізмів контролю допустимості операцій на рівні моделі даних. Без таких механізмів будь-яке вручну введене значення може порушити структуру документа й унеможливити автоматичну звірку.

Джерелом розбіжностей є також різні підходи до інтерпретації статусу документа користувачами. У системах, де статуси не впливають на функціональність, а виконують лише інформативну роль, менеджери можуть не

усвідомлювати, що зміни, які вони вносять у документ, суперечать реальному бізнес-процесу. Відсутність статусних обмежень дозволяє редагувати документ у будь-який момент, у тому числі після формування накладної, що робить інформацію нестабільною та недостовірною.

Технічним джерелом помилок є також відсутність алгоритмів перевірки відповідності між множиною позицій замовлення та множиною позицій накладної. Якщо система не контролює, що кожна позиція накладної має відображення у замовленні, або якщо не гарантується сталість ідентифікаторів товарів, то навіть коректно створена накладна може містити позиції, не пов'язані з первинним документом. Це призводить до неможливості однозначно інтерпретувати дані та відстежити їх походження.

Загалом проблема узгодженості між замовленням і накладною полягає у відсутності єдиної, формалізованої, незмінної та контрольованої структури даних, яка б гарантувала відповідність між очікуваними та фактичними поставками. Складність бізнес-процесу, наявність людського фактора, відсутність статусної логіки та незалежність документів у більшості систем створюють сукупність умов, у яких розбіжності стають не винятком, а системною проблемою. Саме тому оптимізація цієї моделі є необхідною для підвищення точності обліку та мінімізації ризиків, пов'язаних з обробкою постачальницької документації.

1.4 Обмеження традиційних моделей та необхідність формалізації процесу синхронізації

Традиційні моделі взаємодії між замовленням постачальника та електронною накладною історично формувалися як частина загальних облікових систем, тому їхня логіка нерідко не враховує складність сучасних бізнес-процесів постачання. Попри те, що такі моделі здатні відображати базову послідовність операцій — створення замовлення, отримання підтвердження і формування накладної — вони не забезпечують достатнього рівня контролю узгодженості товарних позицій і не містять механізмів, які б гарантували стабільність та незмінність критичних даних.

Це призводить до накопичення системних неточностей і потреби додаткового ручного контролю, що значно ускладнює роботу менеджерів та негативно впливає на точність обліку.

Одним із ключових обмежень традиційних моделей є їхня орієнтація на копіювання даних замовлення в накладну без подальшого забезпечення зв'язку між цими документами. Після копіювання дані фактично існують окремо, що означає відсутність механізму автоматичної перевірки відповідності. У разі, коли користувач змінює позиції в замовленні після узгодження або коригує позиції в накладній, система не відслідковує такі зміни і не повідомляє про порушення узгодженості. Така поведінка свідчить про структурний недолік, за якого множини товарних позицій у двох документах не є формально пов'язаними між собою.

Суттєвим обмеженням є також відсутність статусної логіки, яка б визначала дозволені операції на кожному етапі обробки документів. У більшості систем статуси виконують виключно інформативну функцію і не впливають на можливість редагування. Наприклад, замовлення, що вже узгоджено постачальником, може залишатися доступним для зміни кількості, ціни або структури позицій. У той самий час накладна, яка мала б відображати виключно фактичні дані, у деяких системах також допускає неконтрольоване редагування. Відсутність механізму блокування ключових операцій у критичні моменти життєвого циклу документів створює умови для появи розбіжностей, які виявляються лише під час інвентаризації або перевірки бухгалтерських даних.

Часткові поставки, які є типовим сценарієм у логістичних процесах, також виявляють недосконалість традиційних моделей. У багатьох системах відсутній формалізований підхід до відображення ситуацій, коли фактично поставлена кількість товару відрізняється від замовленої. У таких випадках користувачі або вручну змінюють накладну, або коригують замовлення, що порушує його первинний зміст. За відсутності чіткого механізму відображення часткових поставок система не може коректно синхронізувати позиції, а зміни вносяться без фіксації причин і подій, що їх спричинили.

Ще одним обмеженням є нездатність традиційних систем підтримувати автоматичну перевірку інваріантів, що мають визначати незмінні властивості даних. Наприклад, кількість товару в накладній не повинна перевищувати кількості в замовленні; позиції накладної повинні мати відображення у замовленні; зміни узгоджених даних мають бути забороненими. За відсутності формально описаних інваріантів система не має змоги здійснювати логічний контроль даних, що веде до порушення їхньої цілісності.

Перераховані обмеження пояснюють необхідність формалізації процесу синхронізації між замовленням і накладною, яка передбачатиме чітке визначення структур даних, їхніх взаємних відношень, послідовності допустимих операцій та логічних обмежень. Формалізація повинна забезпечувати незмінність базових даних після узгодження, визначати жорстку взаємозалежність між документами та гарантувати правильність переходів між статусами, що відповідають реальним етапам бізнес-процесу. Важливо також врахувати можливість часткових поставок, для яких має існувати алгоритм побудови накладної на основі узгоджених позицій без порушення структури замовлення.

1.5 Вимоги до оптимізованої моделі взаємодії із підтримкою динамічного оновлення товарних позицій

Удосконалення процесу взаємодії між замовленням постачальника та електронною накладною потребує побудови моделі, яка усуває структурні недоліки традиційних рішень і забезпечує повну синхронізацію даних на всіх етапах життєвого циклу документа. Оскільки аналіз існуючих систем показав відсутність механізмів контролю цілісності, нездатність обробляти часткові поставки в єдиному інформаційному просторі та необмеженість операцій редагування, оптимізована модель повинна відповідати певному комплексу вимог, що визначають її функціональні, логічні та інформаційні властивості.

Основою такої моделі має бути концепція єдиного джерела істини щодо товарних позицій. Це означає, що після етапу узгодження всі позиції замовлення

повинні набувати статусу незмінних, а накладна має формуватися виключно на основі цього зафіксованого набору даних. Відмова від дублювання позицій між документами і суворе відокремлення узгоджених та фактичних значень дозволять уникнути неузгодженостей, які виникають у випадку паралельного редагування. Модель повинна гарантувати, що зміна позиції в одному документі автоматично впливає на доступність операцій в іншому, забезпечуючи жорстку логічну взаємозалежність між ними.

Оптимізована модель також має забезпечувати можливість динамічного оновлення товарних позицій на етапах, коли це дозволено логікою бізнес-процесу. До моменту узгодження користувач повинен мати можливість змінювати замовлення без обмежень, проте після підтвердження постачальника всі дані мають бути зафіксовані, а будь-які зміни повинні бути або заблоковані, або виконуватися лише через додаткові операції, що не впливають на узгоджений набір. Динамічність моделі не може суперечити принципу незмінності узгоджених даних, тому вона повинна враховувати контекст і статус документа, а також забезпечувати механізми зворотного контролю правильності.

Важливим елементом вимог до моделі є підтримка часткових поставок. У реальних умовах постачальник далеко не завжди може поставити повну кількість товару, тому система має дозволяти формування накладної, у якій фактична кількість менша за узгоджену, але при цьому не допускається зміна структури замовлення або втручання в узгоджені значення. Це вимагає наявності механізму математичного зв'язку між множиною позицій замовлення та множиною позицій накладної, що дозволяє ідентифікувати їхню відповідність навіть за наявності часткових відхилень. Такий механізм повинен гарантувати, що накладна не може включати позиції, яких не було у замовленні, і навпаки, що узгоджені позиції не можуть бути вилучені або замінені.

Статусна модель відіграє ключову роль у забезпеченні передбачуваної поведінки системи. Оптимізована структура повинна мати чітко визначений набір статусів, кожен з яких регламентує, які дії користувача дозволені, а які ні. Послідовність переходів між статусами має бути лінійною і незворотною, щоб

виключити можливість порушення логіки бізнес-процесу. Статуси повинні виступати не лише індикаторами стану, але й механізмами управління можливостями редагування, синхронізації даних та формування вихідних документів.

Окремої уваги потребує питання алгоритмізації синхронізації між замовленням і накладною. Оптимізована модель повинна включати формалізовані правила та інваріанти, які визначають поведінку множин товарних позицій та станів документів. Для коректної роботи системи необхідно, щоб такі правила були математично визначеними, однозначними, перевірюваними та виконувалися автоматично без залучення користувача. Наявність інваріантів забезпечує стабільність даних, а формальна специфікація дозволяє уникнути ситуацій, коли різні частини системи інтерпретують одну й ту саму подію по-різному.

Вимоги до оптимізованої моделі також охоплюють питання прозорості процесу та можливості аудиту. Кожна зміна повинна мати чітку причину, бути пов'язана зі статусом документа і не порушувати узгодженості товарних позицій. Це дозволяє не лише забезпечити облікову точність, але й створити умови для автоматизованого контролю та відтворюваності даних, що особливо важливо для систем, що працюють з великими обсягами товарів та великою кількістю постачальників.

2 АНАЛІЗ ІСНУЮЧИХ ПІДХОДІВ ТА ФОРМАЛІЗАЦІЯ МОДЕЛІ ВЗАЄМОДІЇ “ЗАМОВЛЕННЯ ПОСТАЧАЛЬНИКА - ЕЛЕКТРОННА НАКЛАДНА”

У цьому розділі здійснюється аналіз існуючих методів та моделей взаємодії між замовленням постачальника та електронною накладною, що застосовуються у сучасних інформаційних системах управління постачанням. Розглядаються підходи, реалізовані у класичних ERP-системах, документноорієнтованих рішеннях, табличних моделях, структурі Master–Detail та CRUD-підходах. Особливу увагу приділено визначенню причин виникнення розбіжностей у товарних позиціях, порушення цілісності даних та необхідності ручного узгодження.

Аналіз дозволяє виявити обмеження традиційних моделей роботи з документами постачання та сформулювати вимоги до оптимізованої моделі, яка забезпечує узгодженість даних, чітку статусну логіку, мінімізацію ручних операцій та інтегровану взаємодію «замовлення постачальника – електронна накладна».

2.1 Аналіз існуючих моделей взаємодії замовлення та електронної накладної

У сучасних інформаційних системах управління закупівлями використовуються різні підходи до організації взаємодії між замовленням постачальника та електронною накладною. Найпоширенішим є підхід, реалізований у класичних ERP-системах, де процес формалізований у вигляді послідовності «замовлення → накладна → оплата». У цьому випадку накладна створюється як копія товарних

позицій із замовлення, однак документи можуть редагуватися незалежно, що призводить до виникнення розбіжностей при зміні кількості чи ціни.

Ще одним розповсюдженим рішенням є модель Master–Detail, у якій замовлення виступає головним документом, а товарні позиції — підлеглими записами. Такий підхід забезпечує цілісність структури даних у межах одного документа, проте накладна в більшості реалізацій створюється як окремий документ із власним набором позицій. Це зумовлює дублювання даних та відсутність механізмів автоматичного узгодження між документами.

У простіших веб-системах застосовуються CRUD-моделі, де кожен документ (замовлення, накладна, акт) є самостійною сутністю, для якої підтримуються операції створення, читання, редагування та видалення. У таких підходах логіка узгодження реалізується виключно на рівні інтерфейсу та ручних дій користувача, що підвищує ймовірність помилок.

Документоорієнтовані інформаційні системи розглядають замовлення та накладну як елементи єдиного потоку документів. Попри зручність маршрутизації та фіксації історії змін, дані про товарні позиції дублюються у кожному документі, що ускладнює підтримку їх узгодженості при часткових поставках або зміні умов постачальника.

У невеликих підприємствах досі широко застосовуються табличні підходи (Excel, Google Sheets), де замовлення та накладні ведуться у вигляді окремих таблиць. За відсутності формалізованої структури даних, механізмів перевірки та обмежень такі рішення є найбільш схильними до появи розбіжностей та помилок, особливо під час ручного копіювання інформації.

Порівняння ключових властивостей розглянутих моделей подано у таблиці 2.1.

Таблиця 2.1

Назва моделі	Переваги	Недоліки
ERP-модель	Формалізований процес постачання; Автоматичне формування накладної; Стандартизація документообігу	Немає зворотної синхронізації позицій; Дані у замовленні та накладній легко розходяться; Неможливість часткового узгодження позицій
Модель Master - Detail	Структуроване представлення головного документа і позицій; Чіткі зв'язки між записами; Добре підходить для роботи з каталогами позицій	Накладна створюється як окремий документ – дублювання; Немає спільного життєвого циклу документів; Дані між документами не синхронізуються
CRUD модулі документів	Проста реалізація в будь-якій системі; Мінімальні вимоги до інфраструктури; Гнучкість у створенні окремих документів	Весь процес узгодження виконується вручну; Висока ймовірність помилок користувача; Немає перевірок цілісності даних
Document Flow	Підтримка маршрутизації та історії змін; Зручне середовище для управління статусами документів	Дублювання товарних позицій у кожному документі; Розбіжності між документами при зміні умов постачальника; Немає механізму спільної синхронізації
Табличні моделі (Excel/Sheets)	Простота використання; Низький поріг входу; Не потребують складної системи	Відсутність контролю цілісності; Вразливість до ручних помилок; Немає статусної моделі або зв'язків між документами

Узагальнення порівняльного аналізу показує, що жодна з розглянутих моделей не забезпечує цілісності даних між замовленням постачальника та накладною, не підтримує узгоджений життєвий цикл документів і не мінімізує ручну працю користувача.

2.2 Формалізація структури даних та моделі взаємодії «замовлення — накладна»

У цьому підрозділі формалізовано структуру даних та логіку взаємодії між замовленням постачальника й електронною накладною. Описана модель визначає, які множини товарних позицій використовуються в системі, як саме формується накладна на основі замовлення, яким чином забезпечується узгодженість позицій та як статуси впливають на можливість редагування даних.

На рисунку 2.1 наведено узагальнену структурну схему моделі взаємодії. У лівій частині показано вхідні дані (параметри замовлення, множина товарних позицій, власне замовлення постачальника), у центральній частині — модуль обробки та потік даних «замовлення → накладна», а у правій частині — вихідні дані у вигляді електронної накладної та пов'язаних із нею об'єктів (позиції накладної, картки постачальників, інформація про постачальника)

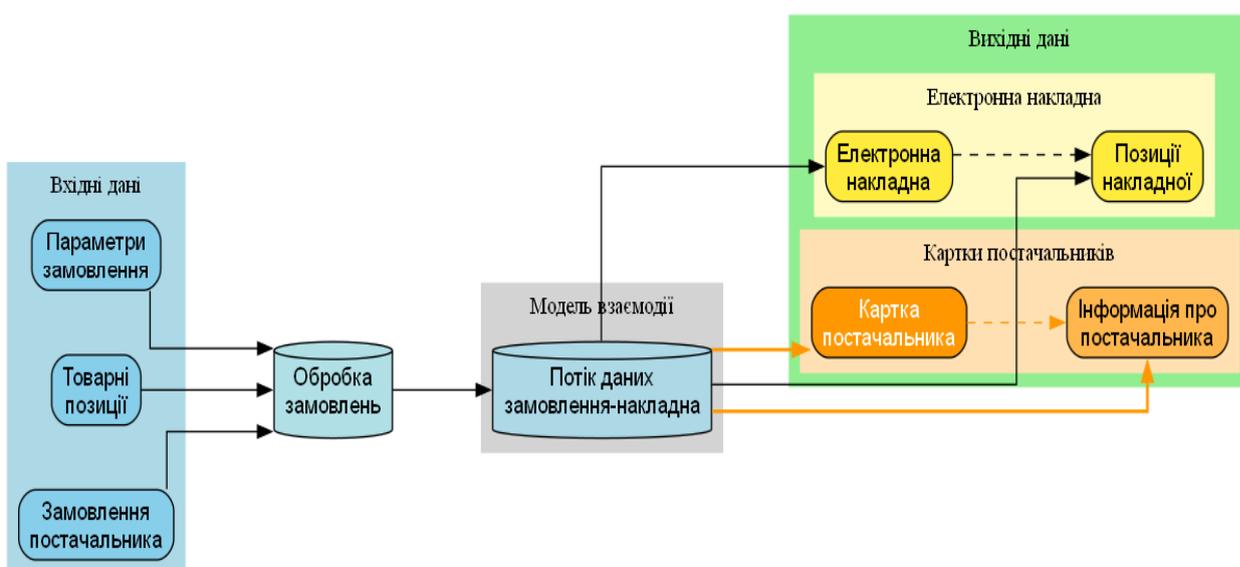


Рис. 2.1 Структурна модель взаємодії

“замовлення постачальника - електронна накладна”

2.2.1 Структура множин товарних позицій

Нехай:

- P — множина товарних позицій, визначених у замовленні постачальника:

$$P = \{ p_1, p_2, \dots, p_n \}$$

де кожна позиція p_i описується кортежем:

$$p_i = (\text{id}, \text{count}, \text{price}, \text{supplier_id})$$

- I — множина товарних позицій, що увійшли до електронної накладної.

У загальному вигляді накладна повинна відображати стан узгоджених позицій замовлення, тобто:

$$I \subseteq P$$

2.2.2 Модель синхронізації замовлення та накладної

Сутність взаємодії можна подати як відображення:

$$f: (P, S) \rightarrow I$$

де:

- P — множина позицій замовлення,
- S — поточний статус картки постачальника,
- I — сформована множина позицій накладної.

Функція відображення підпорядковується правилам:

$$I = \{P, \text{ якщо } S = \text{Узгоджено}, I, \text{ якщо } S = \text{Отримано}\}$$

Після статусу «Узгоджено» позиції замовлення переходять у режим фіксації:

$$\Delta P = 0 \text{ при } S \geq S_{\text{Узгоджено}}$$

що унеможливорює їх зміну, дублювання або пересинхронізацію.

2.2.3 Залежності між замовленням та накладною

Формально встановлюється зв'язок:

$$\forall p_i \in P: \exists i_k \in I \text{ такий, що } i_k.\text{id} = p_i.\text{id}$$

Це означає:

- кожна позиція в накладній є відображенням конкретної позиції замовлення,
- зміна кількості або ціни у накладній не порушує відповідність позицій,

під час часткової поставки формується коректне відображення:

$$i_k \text{ кількість} \leq p_i \text{ кількості}$$

У разі розбіжностей система не дозволяє змінити вихідні дані замовлення, а фіксує зміни лише в накладній, що забезпечує цілісність обліку.

2.2.4 Формалізація статусної моделі

Нехай:

$$S \in \{\text{Створено, Відправлено, Узгоджено, Отримано, Скасовано}\}$$

Правила переходу:

$$f(S) = \{\text{Відправлено,}$$

якщо обрано спосіб відправки Узгоджено, якщо постачальник підтвердив позиції

Обмеження:

$$\text{Створено} \rightarrow \text{Отримано}$$

$$S_{\text{Узгоджено}} \rightarrow \Delta P = 0$$

Це забезпечує:

- неможливість “пропустити” етапи,
- гарантію, що зміни у позиціях виконуються лише до моменту узгодження,
- уникнення розривів даних між документами.

2.3 Формальна специфікація алгоритму синхронізації даних

У цьому підрозділі наведено формальну специфікацію алгоритму синхронізації множин товарних позицій між замовленням постачальника та електронною накладною. Алгоритм подано як систему переходів між станами,

інваріантів та обмежень, що забезпечують однозначність і коректність відображення даних.

2.3.1 Інваріанти та базовані співвідношення

Інваріант 1 – відповідність позицій

$$\forall i_k \in I \exists p_i \in P : i_k.id = p_i.id,$$

де

P – множина позицій замовлення;

I – множина позицій накладної.

Інваріант 2 – Обмеження фактичної кількості

$$i_k.qty \leq p_i.qty,$$

Де

qty – кількість товару, фактично поставлена.

Інваріант 3 – незмінність позицій після узгодження.

$$S \geq S_{узгоджено} \rightarrow \Delta P = 0,$$

де

ΔP – зміна будь-якого елемента p_i .

2.3.2 Система правил алгоритму

Перед формальним описом вводимо чіткі позначення.

Позначення:

- $P = \{p_1, p_2 \dots, p_n\}$ – множина товарних позицій;
- $I = \{i_1, i_2 \dots, i_n\}$ – множина позицій у накладній.
- S – поточний статус картки постачальника.
- $O = \{edit(P), freeze(P), form(I), lock(I)\}$ – множина операцій.
- $perm(o, S) \in \{0, 1\}$ – функція дозволеності операції
(1 – дозволено, 0 – заборонено).
- T – множина дозволених переходів між статусами.

Правило r1 – редагування позицій замовлення постачальника.

Редагування множини позицій P можливе лише у статусах “Створено” та “Відправлено”.

$$perm(edit(P), S) = \{1,$$

$$S \in \{\text{Створено, Відправлено}\} \ 0, \quad S \notin \{\text{Створено, Відправлено}\}$$

Правило r2 – фіксація позицій у статусі “Узгоджено”.

У статусі “Узгоджено” редагування позицій заборонено.

Виконується операція фіксації.

$$perm(freeze(P), S) = \{1, \quad S = \text{Узгоджено} \ 0, \quad S \neq \text{Узгоджено}$$

Після фіксації:

$$\Delta P = 0$$

Правило r3 – формування електронної накладної.

Формування накладної можливе лише після узгодження:

$$perm(form(I), S) = \{1,$$

$$S = \text{Узгоджено} \ 0, \quad S \neq \text{Узгоджено}$$

$$S \neq \text{Узгоджено}$$

Правило r4 – блокування накладної

У стані “Отримано” накладна переходить у режим повної незмінності:

$$perm(lock(I), S) = \{1, \quad S = \text{Отримано} \ 0, \quad S \neq \text{Отримано}$$

Правило r5 – множина дозволених переходів

Функцію переходів:

$$\delta : S \times action \rightarrow S$$

визначаємо на множині дозволених переходів:

$$T = \{\text{Створено} \rightarrow \text{Відправлено}, \text{Відправлено} \rightarrow \text{Узгоджено}, \text{Узгоджено} \\ \rightarrow \text{Отримано}, \text{будь-який} \rightarrow \text{Скасовано}\}$$

Будь-який інший перехід заборонений:

$$\delta(S_1, a) = \text{undefined, якщо } (S_1 \rightarrow S_2) \notin T$$

2.3.3 Модель поведінки системи як скінченний автомат

У цьому підрозділі наведено модель поведінки системи у вигляді детермінованого скінченного автомата (FSM — Finite State Machine). Така модель дозволяє формально описати життєвий цикл картки постачальника, визначити можливі статуси, події, які їх змінюють, та реакцію системи на ці події. Використання FSM забезпечує чіткість логіки роботи процесу, контроль допустимих переходів і виключає можливість некоректних або несанкціонованих змін статусів.

Множина статусів.

Позначимо множину статусів картки постачальника:

$$S = \{S_{cr}, S_{sn}, S_{ap}, S_{rc}, S_{cn}\}$$

де:

- S_{cr} – “Створено”;
- S_{sn} – “Відправлено”;
- S_{ap} – “Узгоджено”;
- S_{rc} – “Отримано”;
- S_{cn} – “Скасовано”.

Кожен статус відповідає певному етапу життєвого циклу замовлення постачальника.

Множина подій.

Події, що можуть спричинити зміну статусу, утворюють множину:

$$A = \{a_1 = \text{sendOrder}, a_2 = \text{confirm}, a_3 = \text{receiveGoods}, a_4 = \text{cancel}\}.$$

Кожна подія відповідає конкретній дії користувача або результату взаємодії постачальником.

Функція переходів між статусами:

$$\delta : S \times A \rightarrow S$$

Яка повертає новий статус на основі поточного статусу та події.

Поведінка автомата описується такими допустимими переходами між статусами

$$\delta(S_{cr}, sendOrder) = S_{sn},$$

$$\delta(S_{sn}, confirm) = S_{ap},$$

$$\delta(S_{ap}, receiveGoods) = S_{rc},$$

$$\delta(S, cancel) = S_{cn}, \forall S \in S.$$

Якщо комбінація “поточний статус + подія” не виходить до визначеної множини переходів вважається недопустимим:

$$\delta(S, a) = \text{undefined}$$

Це гарантує захист даних від некоректних послідовностей операцій.

На рисунку 2.2 зображено UML-діаграму статусів замовлення постачальника.



Рис. 2.2 UML-діаграма статусів замовлення постачальника.

2.3.4 Алгоритм взаємодії замовлення постачальника та електронної накладної

Алгоритм роботи системи описує повну послідовність дій, пов'язану зі створенням, узгодженням та опрацюванням замовлення постачальника, а також із формуванням відповідної електронної накладної. Основною метою алгоритму є забезпечення узгодженості між двома документами, контроль допустимих операцій на кожному етапі та запобігання появі некоректних або несанкціонованих змін у даних.

Процес взаємодії починається із формування замовлення. Менеджер створює нове замовлення та додає до нього товарні позиції. На цьому етапі система дозволяє редагувати перелік товарів, їх кількість, вартість та інші параметри. Позиції можуть бути змінені, видалені або додані без будь-яких обмежень, оскільки документ перебуває у початковому статусі та не пов'язаний із реальними діями постачальника.

Після того як замовлення сформовано, менеджер відправляє його постачальнику за обраним каналом зв'язку, після чого система переводить картку постачальника у статус «Відправлено». На цьому етапі допускаються незначні коригування, однак основна задача системи — очікувати відповідь постачальника. Замовлення ще не є зафіксованим остаточно, тому користувач може вносити зміни, якщо це необхідно.

Наступним етапом є узгодження товарних позицій. Після того як постачальник підтвердив замовлення, картка переходить у статус «Узгоджено». Це ключовий момент алгоритму, оскільки в цьому статусі редагування позицій стає недоступним. Система фіксує їхній остаточний перелік, після чого автоматично створює електронну накладну на основі зафіксованих даних. Усі позиції замовлення копіюються у накладну, що забезпечує узгодженість між запланованою та фактичною поставкою.

На етапі формування електронної накладної система створює документ, який містить всі товарні позиції, узгоджені з постачальником. Якщо постачальник

здійснює часткову поставку, у накладній може бути зменшена кількість окремих позицій порівняно з початковим замовленням, проте така зміна не впливає на зафіксовані дані замовлення. Накладна слугує документальним підтвердженням фактичного виконання замовлення.

Після отримання фактичної поставки менеджер здійснює приймання товару, і накладна переходить у статус «Отримано». На цьому етапі система блокує будь-які зміни як у накладній, так і в картці постачальника. Документ набуває фінального статусу, і подальше редагування можливе лише шляхом створення додаткових документів, таких як коригувальні накладні чи нові замовлення.

Завершення процесу означає, що всі дані — перелік позицій, їхні кількості, статуси та відповідність між документами — остаточно зафіксовано. З цього моменту вони можуть бути використані для внутрішніх облікових операцій, фінансової звітності або аналітичних розрахунків. Алгоритм забезпечує цілісність та узгодженість даних протягом усього життєвого циклу документа.

2.3.5 Сценарії роботи алгоритму синхронізації

Робота алгоритму синхронізації товарних позицій між замовленням постачальника та електронною накладною може бути проілюстрована низкою типових сценаріїв, що демонструють його поведінку в практичних умовах. Такі сценарії дозволяють простежити, як змінюються статуси картки постачальника та які дії стають доступними або недоступними залежно від контексту роботи користувача.

У базовому сценарії повної поставки менеджер створює замовлення постачальника, додає перелік товарних позицій і зберігає його. Після цього замовлення відправляється постачальнику, а картка переходить у статус «Відправлено». Менеджер очікує відповідь, після чого, отримавши підтвердження всіх позицій, переводить картку у статус «Узгоджено». На цьому етапі система автоматично фіксує остаточний перелік товарних позицій і формує на їх основі електронну накладну. Після фактичного отримання товару менеджер здійснює

приймання, і документ переходить у статус «Отримано». Усі зміни відтепер заблоковані, а дані замовлення та накладної є повністю узгодженими.

Інший поширений сценарій — часткова поставка. У цьому випадку постачальник не може поставити всю кількість товару, зазначену в замовленні. Після відправлення замовлення менеджер отримує відповідь від постачальника, у якій зазначається фактично доступна кількість товару. Під час узгодження система фіксує перелік позицій замовлення, але під час формування накладної дозволяє зазначити меншу кількість товару для окремих позицій, що відображає реальний обсяг поставки. При цьому замовлення залишається незмінним, а накладна містить фактичні дані. Після прийняття часткової поставки накладна блокується, а можливість коригувати дані доступна лише через створення додаткових документів, наприклад нової накладної для решти товару.

Окремий сценарій описує випадок, коли постачальник відмовляється від виконання замовлення. У такій ситуації менеджер переводить картку постачальника у статус «Скасовано». Це може статися як до моменту узгодження позицій, так і після відправлення замовлення. Після скасування жодні дії над картою та товарними позиціями не допускаються, що запобігає випадковому використанню некоректних або неактуальних даних.

Ще один сценарій передбачає повторне відправлення або коригування замовлення до моменту узгодження. Оскільки в статусах «Створено» та «Відправлено» редагування доступне, менеджер може змінити кількість, замінити товари або відкоригувати параметри замовлення. Алгоритм забезпечує повну свободу редагування до моменту переходу у статус «Узгоджено», після чого всі зміни блокуються для забезпечення цілісності даних.

Усі наведені сценарії демонструють, що алгоритм синхронізації забезпечує логічну послідовність виконання операцій, гарантує цілісність даних і запобігає зниженню точності товарного обліку. За рахунок контролю статусів та жорсткого обмеження можливих дій у кожному з них система працює передбачувано й стабільно незалежно від сценарію взаємодії з постачальником.

2.4 Моделювання потоків даних у системі

У цьому підрозділі описується моделювання потоків даних між основними компонентами інформаційної системи, що забезпечує роботу із замовленнями постачальників та електронними накладними. Таке моделювання дозволяє визначити, яким чином інформація проходить через систему, які об'єкти беруть участь у взаємодії, які дані передаються між ними та в яких точках відбуваються критичні перетворення. Результатом побудови моделей потоків даних є формування цілісного уявлення про логіку обробки замовлення — від моменту його створення менеджером до формування та прийняття електронної накладної.

2.4.1 Контекстна модель потоків даних

На контекстному рівні система розглядається як єдиний логічний процес, що взаємодіє із зовнішніми акторами. Основними учасниками обміну інформацією є менеджер, постачальник та облікова підсистема підприємства.

Менеджер формує замовлення постачальника і передає до системи інформацію про товарні позиції, дату замовлення, спосіб оплати та інші параметри. Система повертає менеджеру інформацію про статуси замовлення, результати узгодження та сформовані електронні накладні. Постачальник отримує від системи сформоване замовлення та повертає підтвердження, відмову або уточнення щодо кількості товарів. Після обробки результатів узгодження система формує електронну накладну та передає її дані до облікової підсистеми, де інформація використовується для подальшого бухгалтерського та складського обліку.

Контекстна діаграма дозволяє визначити межі системи, джерела та одержувачів даних, а також основні зовнішні потоки інформації.

На рисунку 2.3 зображена контекстна модель потоків даних системи управління замовленнями постачальників.

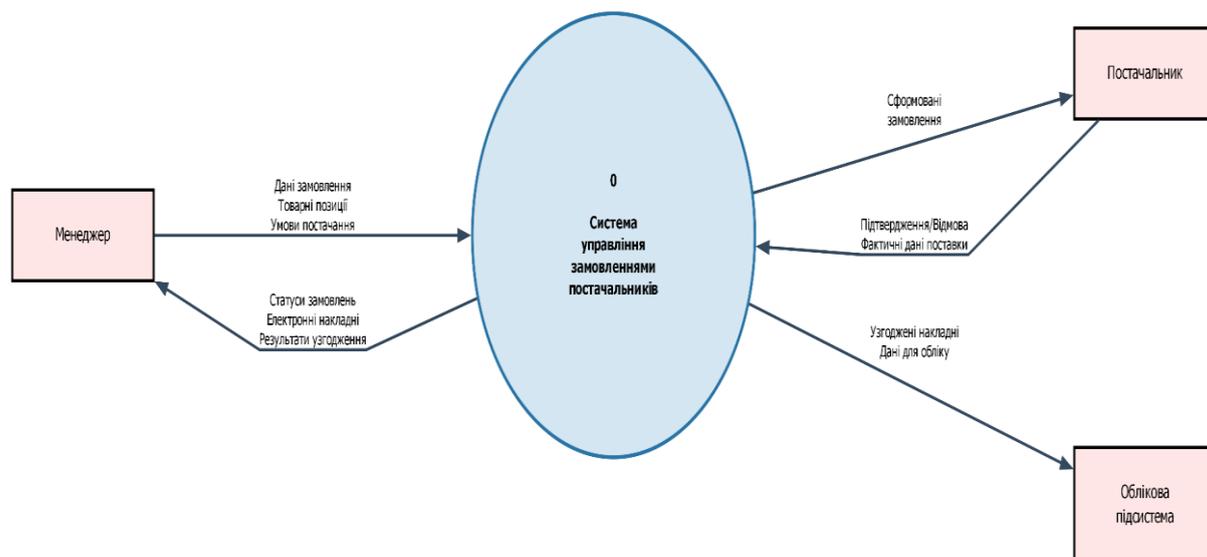


Рис. 2.3 Контекстна модель потоків даних системи управління замовленнями постачальників

2.4.2 Деталізована модель потоків даних процесу обробки замовлення постачальника

На наступному рівні деталізації контекстна модель розкладається на окремі підпроцеси, що формують життєвий цикл замовлення постачальника. Основними функціональними блоками системи є:

- формування замовлення постачальника;
- обробка відповіді постачальника;
- узгодження товарних позицій;
- формування електронної накладної;
- передача результатів до облікової підсистеми.

Підпроцес формування замовлення постачальника включає створення нового замовлення та додавання товарних позицій. Менеджер передає відповідні дані до системи, після чого вони зберігаються у сховищі замовлень та позицій. Після відправлення замовлення постачальнику система переходить до наступного підпроцесу — обробки відповіді постачальника. У цьому блоці обробляється

підтвердження або відмова щодо окремих позицій, що дозволяє визначити перелік узгоджених товарів.

Після узгодження даних система переходить до підпроцесу формування електронної накладної, у якому автоматично створюється документ поставки, синхронізований із карткою постачальника. Накладна містить фактичні кількості товару, що можуть відрізнятися від початкового замовлення у випадку часткової поставки. На завершальному етапі інформація про накладну передається до облікової підсистеми, де використовується для формування складських залишків та фінансової звітності.

На рисунку 2.4 зображена деталізована модель потоків даних обробки замовлення постачальника.

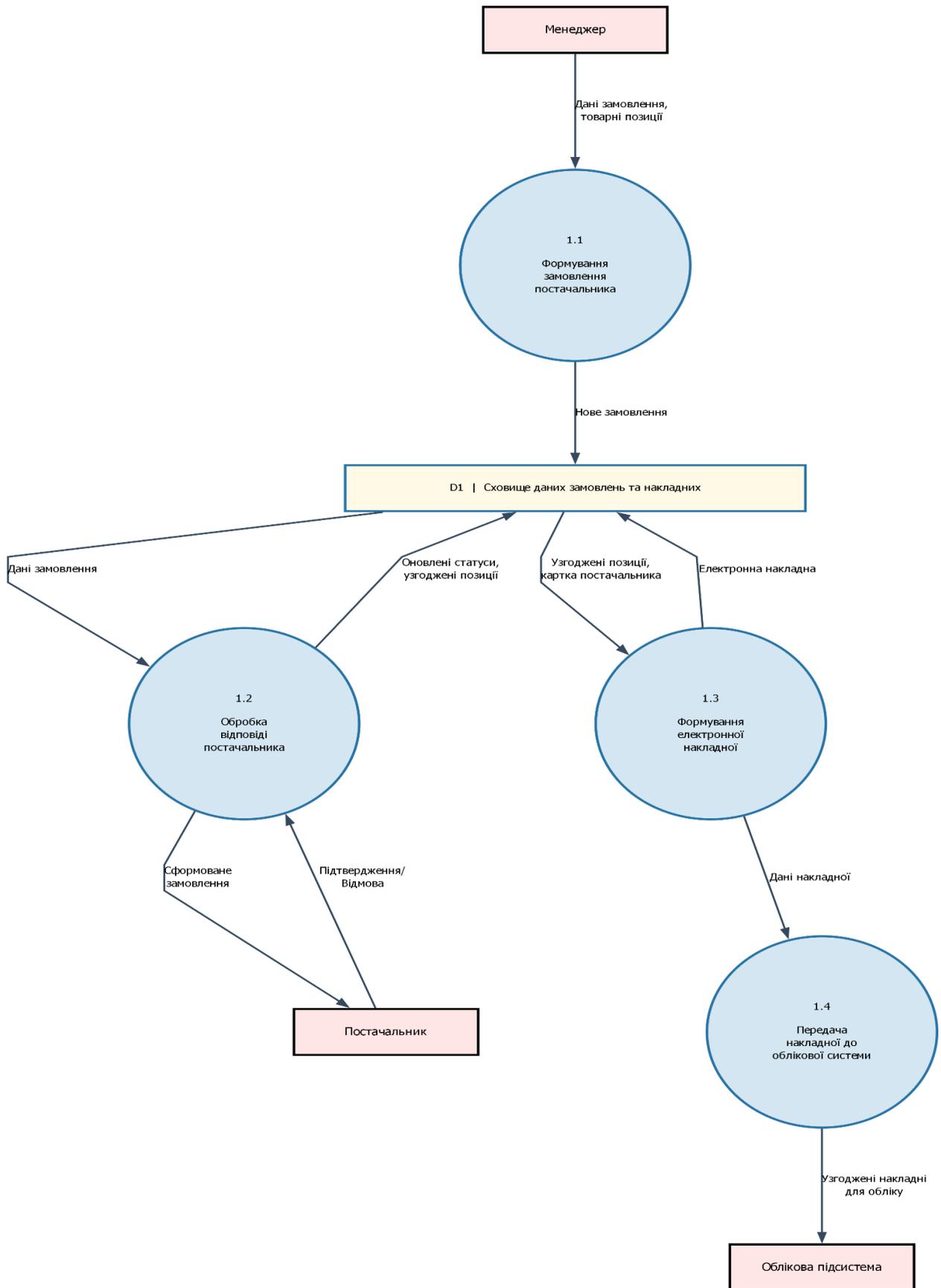


Рис. 2.4 Деталізована модель потоків даних обробки замовлення

2.4.3 Критичні точки обробки даних у процесі взаємодії замовлення та накладної

У процесі обробки замовлення постачальника та формування електронної накладної система проходить через низку критичних точок, у яких відбуваються важливі зміни стану даних. Виділення таких точок дає можливість визначити етапи, на яких здійснюється фіксація інформації, синхронізація товарних позицій та контроль цілісності.

Першою критичною точкою є момент створення замовлення, коли менеджер формує первинний набір товарних позицій і визначає параметри постачальника. На цьому етапі дані доступні для коригування, а система не накладає обмежень щодо редагування.

Наступною точкою є відправлення замовлення постачальнику, після чого система переходить до режиму очікування відповіді. На цьому етапі система фіксує структуру замовлення та реєструє факт передачі замовлення зовнішньому учаснику.

Ключовою точкою є узгодження замовлення, під час якого система переводить картку постачальника у стан, що забороняє подальше редагування товарних позицій. Усі дані переносяться в узгоджений вигляд, який надалі використовується як база для формування накладної.

Під час формування електронної накладної виконується друга фіксація даних — цього разу за фактичними результатами поставки. На цьому етапі система звіряє узгоджені позиції із даними постачальника та створює остаточний документ поставки.

Завершальною критичною точкою є приймання накладної, після якого система блокує можливість редагування документів, забезпечуючи стабільність та цілісність інформації протягом усього подальшого циклу роботи.

2.5 Моделювання структури даних системи

У цьому підрозділі виконується формалізація структури даних, що використовується в системі управління замовленнями постачальників та електронними накладними. Моделювання структури даних дозволяє визначити основні сутності предметної області, встановити зв'язки між ними та описати логіку збереження інформації на різних етапах життєвого циклу замовлення постачальника.

2.5.1 Модель сутностей та зв'язків

ER-модель системи включає набір сутностей, які відповідають об'єктам реального бізнес-процесу: замовленню постачальника, карткам постачальників, товарним позиціям, електронним накладним та довідниковій інформації. Структура даних організована таким чином, щоб забезпечити узгодженість позицій між замовленням і накладною, а також можливість простежити повний ланцюг взаємодії — від створення замовлення до приймання фактичної поставки.

На рисунку 2.5 наведена ER-діаграма, що відображає сутності та їх взаємозв'язки.

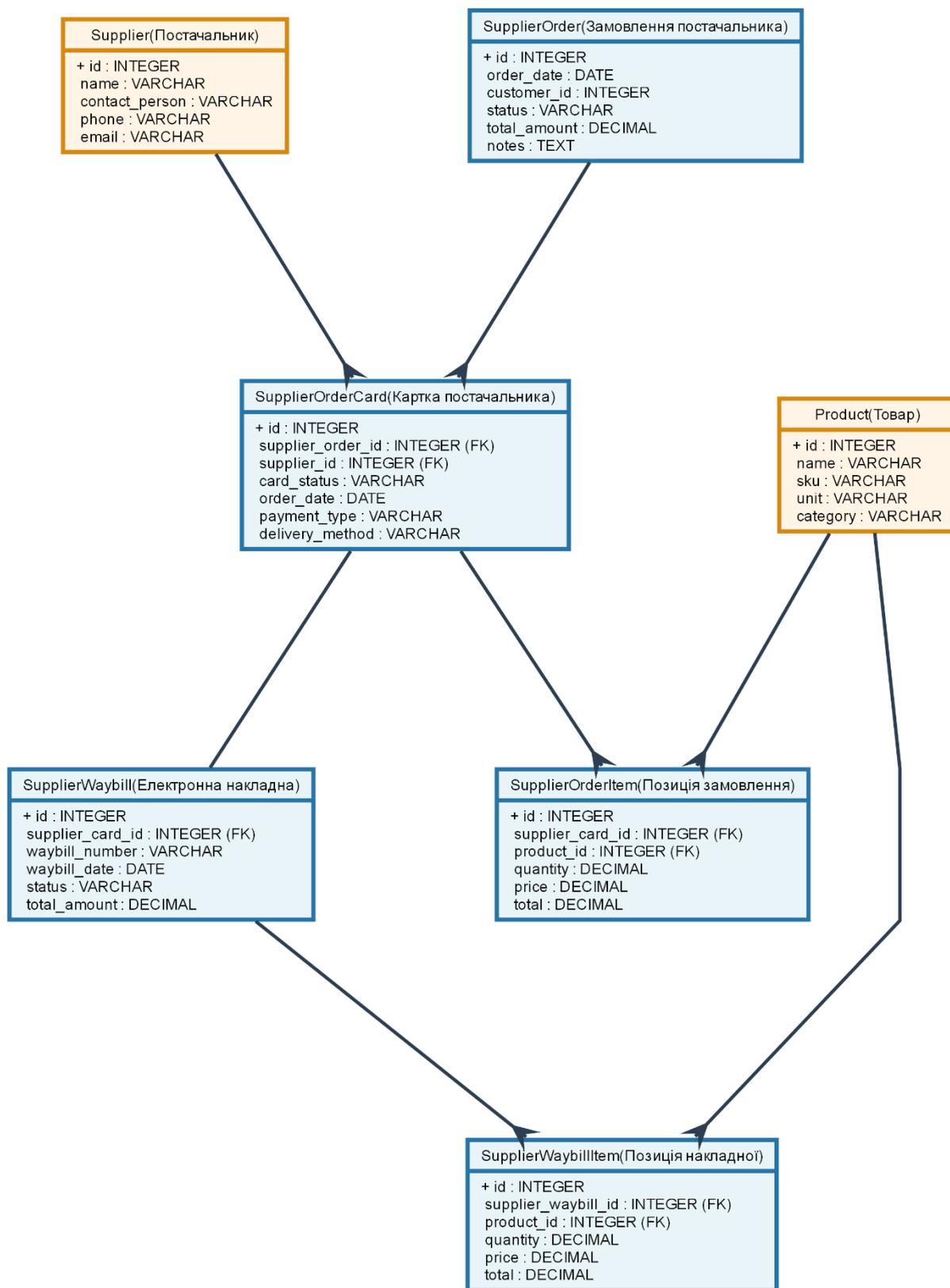


Рис. 2.5. ER-діаграма

Основою моделі є сутність «Замовлення постачальника», яка містить загальну інформацію про замовлення: дату замовлення, клієнта, тип оплати та загальний статус. З одним замовленням може бути пов'язано декілька карток постачальників, якщо у межах одного документа працюють з різними постачальниками.

Кожна картка постачальника має прив'язку до довідника постачальників та містить власний життєвий цикл зі статусами «Створено», «Відправлено», «Узгоджено», «Отримано» чи «Скасовано». У межах картки формується перелік товарних позицій, які відображають асортимент запланованого постачання. Для кожної позиції фіксуються товар, кількість, ціна та інші параметри.

Після узгодження система формує електронну накладну, яка пов'язана з конкретною картою постачальника. Накладна містить фактичні товарні позиції, кількість яких може відрізнятись від запланованих у випадку часткової поставки. Обидва набори позицій — у замовленні та у накладній — пов'язані через товар із довідника «Товар», що забезпечує уніфіковану номенклатуру.

2.5.2 Опис сутностей та логіки зв'язків

Сутність “Замовлення постачальника(SupplierOrder)” містить узагальнену інформацію про замовлення: дату створення, загальний статус, тип оплати, а також ідентифікатор клієнта або відповідального менеджера. Замовлення є кореневою сутністю моделі.

Сутність «Картка постачальника» (SupplierOrderCard) є деталізацією замовлення за постачальником. Включає дату замовлення, спосіб оплати, статус життєвого циклу та посилання на конкретного постачальника. Для кожної картки формується власний перелік товарних позицій.

Сутність «Позиція замовлення» (SupplierOrderItem) містить інформацію про товар, його кількість і ціну. Позиції пов'язуються з картою постачальника через відповідний ідентифікатор. Саме ці позиції підлягають узгодженню з постачальником.

Сутність «Електронна накладна» (SupplierWaybill) створюється після узгодження замовлення. Містить інформацію про фактичну поставку товарів та має прив'язку до відповідної картки постачальника. Статус накладної залежить від етапу обробки поставки.

Сутність «Позиція накладної» (SupplierWaybillItem) відображає фактичні дані поставки: товар, кількість та ціну. Позиції накладної завжди формуються на основі узгоджених даних замовлення, що забезпечує синхронізацію між документами.

Сутність «Постачальник» (Supplier) зберігає довідникову інформацію про постачальників: назву, контактні дані та інші властивості.

Сутність «Товар» (Product) є довідником номенклатури товарів, що використовується при формуванні як замовлення, так і накладної. Це забезпечує єдиний каталог для обох документів.

2.6 Моделювання функціональних можливостей системи

У межах процесу оптимізації моделі взаємодії замовлення постачальника та електронної накладної важливо визначити перелік функцій, які забезпечують повний життєвий цикл опрацювання замовлення. Для цього використовується підхід моделювання варіантів використання (Use Case Model), що дозволяє формалізувати взаємодію між користувачами системи та її функціональними компонентами. Варіанти використання описують зовнішню поведінку системи, відокремлюючи логіку роботи від внутрішніх технічних деталей. Такий підхід є корисним для опису бізнес-процесів, оскільки дозволяє визначити ролі, відповідальність та можливі дії кожного учасника, а також забезпечує узгодженість між моделлю даних та алгоритмами обробки замовлень.

Безпосередніми акторами системи є менеджер та адміністратор системи. Менеджер працює із замовленнями постачальників, формуючи їх структуру, редагуючи товарні позиції, ініціюючи процес узгодження та завершуючи його

шляхом приймання фактичної накладної. Взаємодія менеджера із системою охоплює всі етапи обробки замовлення — від створення до завершення.

Постачальник виступає зовнішнім учасником бізнес-процесу, який отримує замовлення поза межами системи (через телефонний дзвінок, електронну пошту чи Viber), аналізує його зміст і надає відповідь щодо узгодження товарних позицій. Результат цієї взаємодії фіксується в системі менеджером і впливає на подальший стан картки постачальника та можливість формування електронної накладної. Постачальник фактично підтверджує або уточнює запланований обсяг поставки, але не є користувачем інтерфейсу системи.

Адміністратор системи виконує допоміжну роль, відповідаючи за налаштування довідникових даних, зокрема переліку постачальників та товарів, а також правил опрацювання замовлень. Його функціонал забезпечує підтримку стабільної роботи системи та узгодженість структури даних.

На рисунку 2.6 наведено модель варіантів використання, що узагальнює функціональні можливості системи.

У моделі визначено основні сценарії, які реалізуються менеджером: створення замовлення, редагування позицій, відправлення замовлення постачальнику, узгодження товарних позицій та робота з електронною накладною. Постачальник, у свою чергу, виконує операції з отримання замовлення, підтвердження або відмови та надання фактичних даних поставки.

Use Case-модель показує межі відповідальності акторів та дозволяє відобразити логіку роботи системи з точки зору користувача. Вона також демонструє, що система підтримує чітку послідовність дій, яка узгоджується з побудованими раніше моделями даних і алгоритмами обробки замовлення. Це забезпечує цілісність процесу та формує основу для подальшої реалізації бізнес-логіки.

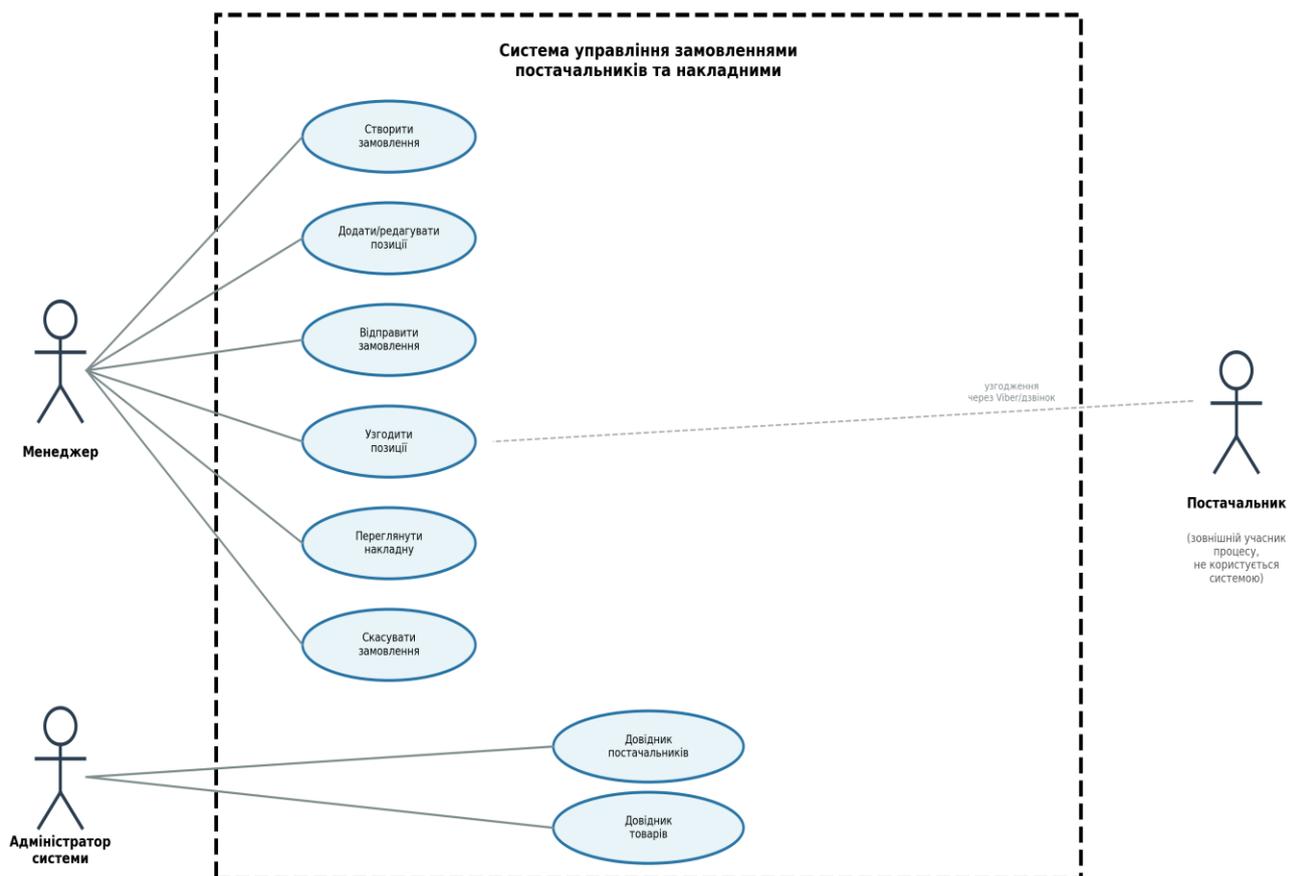


Рис. 2.6. Діаграма варіантів використання системи управління замовленнями постачальників та накладними

3 РЕАЛІЗАЦІЯ МОДЕЛІ ТА ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ

Розроблена модель взаємодії замовлення постачальника та електронної накладної була реалізована у вигляді веб-орієнтованої інформаційної системи. У цьому розділі наведено опис використаних технологій, архітектурних рішень, механізмів реалізації основних модулів, а також результати експериментального дослідження, яке проводилось для оцінювання ефективності запропонованого підходу. Особливу увагу приділено контролю статусів, фіксації товарних позицій після узгодження та формуванню електронної накладної на основі незмінного набору даних, оскільки ці елементи визначають коректність життєвого циклу документа.

У межах розділу розглядаються як практичні аспекти реалізації системи, так і результати тестування, що дозволяють оцінити вплив запропонованої моделі на точність обліку та стабільність процесу обробки замовлень. Структура розділу відповідає загальним вимогам до магістерських робіт і включає опис середовища розробки, архітектурні рішення, деталізацію механізмів реалізації моделі, а також набір тестових сценаріїв, метрики оцінювання та результати експериментів.

3.1 Програмні засоби та середовище розробки

Розробка інформаційної системи здійснювалася із використанням сучасних інструментів веб-розробки, що забезпечують високу продуктивність, масштабованість та зручність реалізації бізнес-логіки. Основою програмної частини виступає фреймворк Laravel, який використовується як універсальна серверна платформа для побудови REST-орієнтованих модулів, реалізації контролерів, маршрутизації HTTP-запитів, моделювання структури даних та забезпечення взаємодії з базою даних через ORM-механізм Eloquent.

У якості серверної мови програмування використовувалася PHP, що дозволило поєднати Laravel з іншими компонентами системи та забезпечити гнучку реалізацію бізнес-процесів, зокрема логіки статусних переходів, фіксації товарних позицій та формування накладної. Фронтенд-частина побудована з використанням JavaScript та бібліотеки jQuery, що забезпечує динамічне оновлення інтерфейсу, роботу з AJAX-запитами та інтерактивну обробку дій користувача у режимі реального часу. Для побудови форм, таблиць та структурованого інтерфейсу використовувалися Blade-шаблони та компонентні можливості Laravel.

В якості системи керування базами даних застосовано MySQL, що забезпечує швидку обробку транзакцій, підтримку складних запитів та можливість моделювання зв'язків між сутностями «замовлення», «картка постачальника», «товарні позиції» та «електронна накладна». Для тестування структури даних та аналізу ефективності статусної моделі також використовувалися SQL-запити, створені у межах експериментального дослідження.

Розробка виконувалась у середовищі Visual Studio Code, а запуск локального серверного середовища — засобами Laravel Sail та Apache/PHP під Windows. Для візуалізації UML-діаграм використовувалися засоби Graphviz та відповідні Python-бібліотеки. Усе це забезпечило повний цикл розробки — від проектування до тестування та експлуатаційного аналізу.

3.2 Архітектура розробленої системи

Архітектура інформаційної системи побудована таким чином, щоб забезпечити коректну реалізацію життєвого циклу замовлення постачальника, механізму узгодження товарних позицій та формування електронної накладної. Основна увага приділена модульності, керованості статусами та розмежуванню рівнів обробки даних, що дозволяє досягти стабільності процесів і гарантувати цілісність інформації на всіх етапах роботи.

Система реалізує функціональні можливості відповідно до концепції розділення обов'язків між рівнем представлення, бізнес-логікою та рівнем даних. Такий підхід забезпечує можливість незалежного розвитку окремих модулів, а також спрощує підтримку механізмів синхронізації позицій між замовленням та накладною.

3.2.1 Логічна архітектура системи

Логічна архітектура розробленої системи відображає структуру основних компонентів та взаємодію між ними у процесі опрацювання замовлення постачальника та формування електронної накладної. Архітектура побудована таким чином, щоб забезпечити послідовний та керований рух даних між рівнем введення, бізнес-логікою та сховищем даних.

У межах системи виділено три ключові логічні рівні.

Перший рівень — рівень представлення, який відповідає за взаємодію користувача з системою та забезпечує формування інтерфейсів для створення замовлення, редагування товарних позицій, узгодження та перегляду накладної. На цьому рівні виконується первинна перевірка введених даних, а також відправлення запитів до бізнес-логіки.

Другий рівень — рівень бізнес-логіки, який реалізує основні алгоритми системи. Саме тут здійснюється аналіз товарних позицій, автоматичне створення картки постачальника, формування множини позицій для накладної, контроль переходів між статусами та блокування редагування після узгодження. Уся логіка реалізована у вигляді окремих функціональних модулів, що дозволяє підтримувати розділення відповідальностей та спрощує розширення системи в майбутньому.

Третій рівень становить рівень даних, який відповідає за збереження станів документів, товарних позицій, статусів та історії змін. Структура бази даних організована у вигляді взаємопов'язаних таблиць, що забезпечують цілісність інформації при переході між етапами життєвого циклу замовлення. На цьому рівні

реалізується зберігання фіксованих позицій після узгодження, що унеможливорює появу розбіжностей між замовленням та накладною.

На рисунку 3.1 зображено трьохрівнева архітектура системи (Presentation → Business Logic → Data).

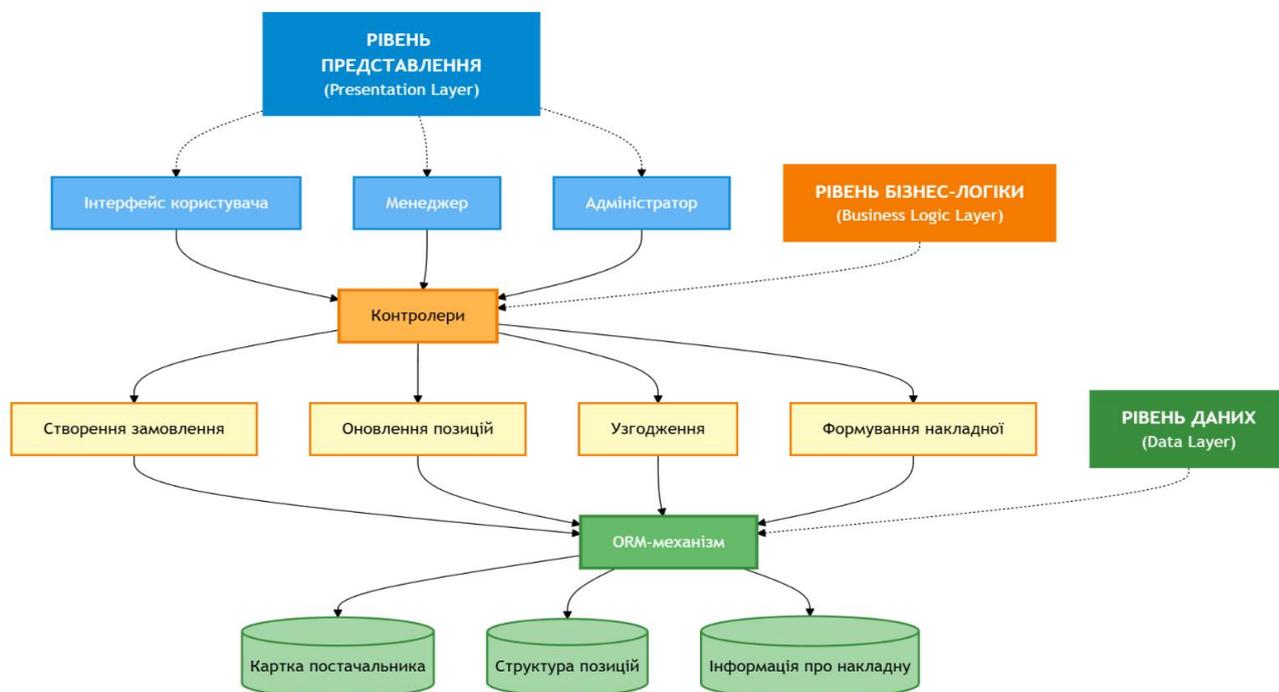


Рис. 3.1 Трьохрівнева архітектура системи

3.2.2 Структура основних модулів

Структура основних модулів системи визначає логічне розмежування функціональних компонентів, що забезпечують повний життєвий цикл опрацювання замовлення постачальника — від моменту його створення до формування електронної накладної та фіксації результатів поставки. Кожен модуль

реалізує окремий сегмент бізнес-процесу, при цьому взаємодія між ними здійснюється відповідно до визначеної послідовності обробки даних.

Модуль створення замовлення відповідає за ініціалізацію документа та формування первинного набору даних. У межах цього модуля користувач визначає параметри замовлення, додає товарні позиції та може редагувати їх до моменту відправлення постачальнику. Після збереження даних система автоматично аналізує їх та створює картки постачальників, пов'язаних із товарами, що входять до замовлення. Таким чином формується множина окремих контекстів, у яких відбувається подальше узгодження.

Модуль управління картою постачальника забезпечує обробку даних у межах кожного окремого постачальника. Він акумулює товарні позиції, розраховує суму замовлення, зберігає дату, тип оплати та поточний статус взаємодії. Саме цей модуль виконує роль проміжної ланки між первинним замовленням і формуванням накладної, оскільки саме на його рівні відбувається узгодження позицій. Крім того, модуль контролює можливість редагування даних відповідно до статусів картки.

Наступним компонентом є модуль узгодження, у межах якого менеджер підтверджує позиції, отримані від постачальника. На цьому етапі система забезпечує фіксацію множини товарних позицій, що унеможлиблює подальші зміни у замовленні. Зафіксований набір позицій стає основою для формування електронної накладної. Узгодження є критичним етапом, оскільки саме тут завершується етап планування й відбувається фіксація інформації для подальшого документального оформлення.

Після узгодження даних активується модуль формування електронної накладної. Він копіює зафіксовані позиції картки постачальника та створює документ накладної, який використовується для приймання поставки та подальшого обліку. У межах цього модуля допускається уточнення кількості товару на основі фактичної поставки, однак зміни не впливають на вихідні дані замовлення, що забезпечує захист від розбіжностей і зберігає історичну точність документів.

3.2.3 Архітектурні рішення та шаблон MVC

Архітектурна побудова системи ґрунтується на шаблоні проектування Model–View–Controller (MVC), який використано як базову концепцію для розподілу функцій між рівнями застосунку. Застосування цієї архітектури забезпечує чітку логічну ізоляцію між бізнес-логікою, обробкою запитів та відображенням даних інтерфейсу, що дає можливість підтримувати систему у масштабованому, структурованому та розширюваному стані.

На рівні моделі реалізовано структуру даних, що описує замовлення постачальника, товарні позиції, картку постачальника та електронну накладну. Моделі відповідають за відображення цих сутностей у базі даних, а також за визначення зв'язків між ними. Функціонування моделей забезпечується ORM-механізмом Eloquent, який дозволяє працювати з даними у вигляді об'єктів та гарантує цілісність зв'язків під час виконання CRUD-операцій. Важливою частиною моделі є механізм фіксації товарних позицій після узгодження, який реалізується за допомогою відповідних правил доступу та логічних обмежень.

На рисунку 3.3 зображено сценарій взаємодії компонентів у архітектурі MVC.

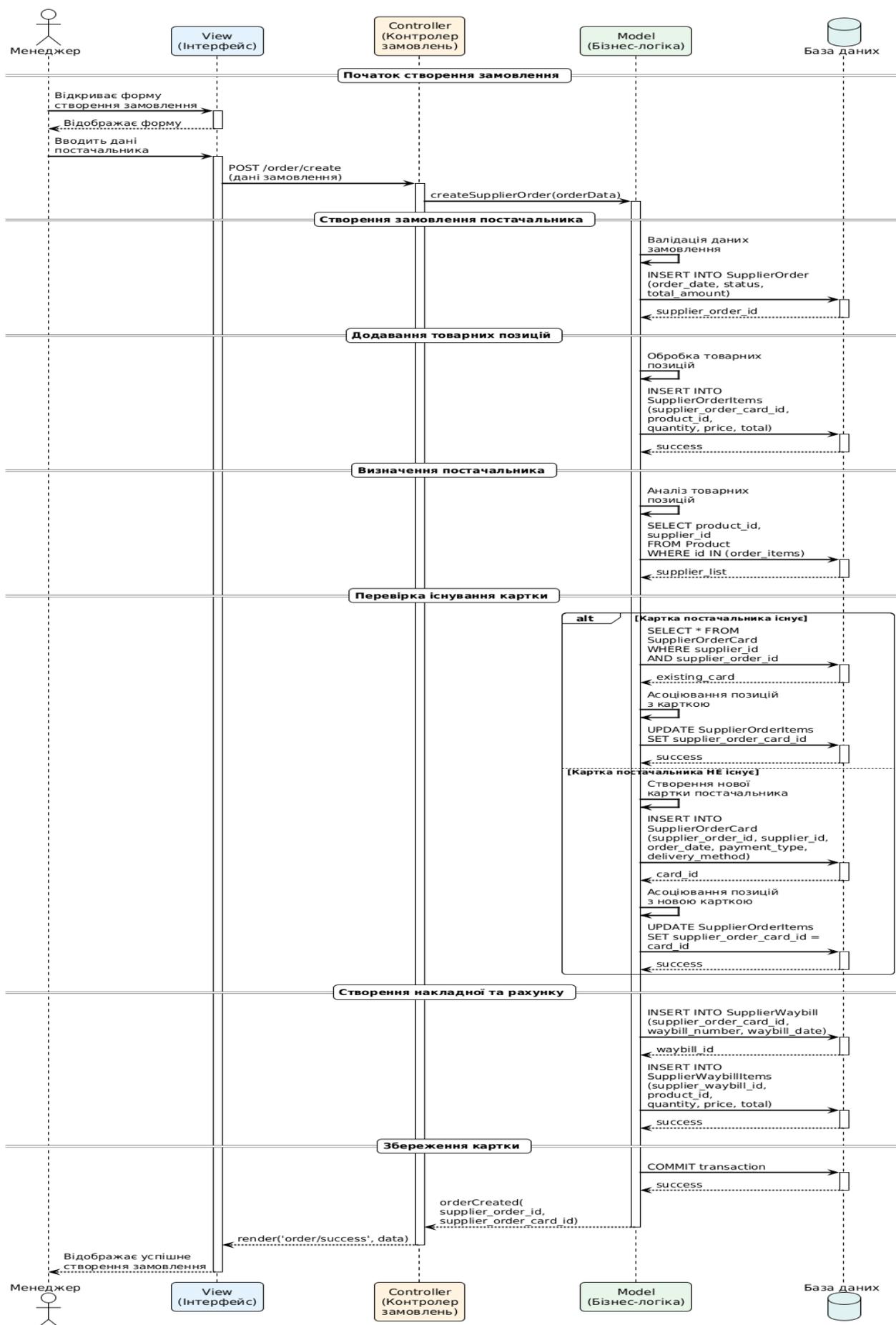


Рис. 3.3. Сценарій взаємодії компонентів у архітектурі MVC

3.2.4 Структура бази даних

Структура бази даних системи розроблена таким чином, щоб забезпечити цілісне подання життєвого циклу взаємодії між замовленням постачальника та електронною накладною, а також підтримувати узгодженість даних на всіх етапах роботи з документами. У центрі моделі даних знаходяться сутності, що описують замовлення, товарні позиції та пов'язані з ними довідникові об'єкти. Логіка взаємодії між таблицями відображає послідовність обробки даних у системі та дозволяє фіксувати ключові події процесу.

Основою структури є таблиця `SupplierOrder`, яка містить загальні параметри замовлення постачальника — дату створення, тип оплати, пов'язаного контрагента та інші атрибути, що визначають його контекст. Кожне замовлення пов'язане з множиною товарних позицій, які зберігаються у таблиці `SupplierOrderItem`. Саме тут фіксуються кількість, ціна та ідентифікатор товару, що забезпечує деталізацію змісту замовлення. Важливою характеристикою цих записів є стабільність їхньої структури: позиції замовлення існують незалежно від подальших дій з накладною й відображають первинний намір поставки.

Для кожного постачальника, чийі товари входять до замовлення, формується спеціальна структурна одиниця — `SupplierCard`. Картка слугує логічним контейнером, у межах якого проводиться узгодження товарних позицій. Вона містить агреговану інформацію щодо відповідного постачальника, суму товарів, параметри поставки та поточний статус. На рівні бази даних картка постачальника пов'язана з таблицями замовлень та позицій через відповідні зовнішні ключі, що забезпечує прямий доступ до всіх даних, необхідних для узгодження.

Після завершення процесу узгодження система генерує електронну накладну, яка представлена таблицею `SupplierWaybill`. Накладна створюється на основі зафіксованого набору позицій картки постачальника й містить інформацію про дату формування, тип документа та пов'язаного постачальника. Товарні позиції накладної зберігаються в таблиці `SupplierWaybillItem`, де фіксуються фактичні кількості товару, зазначені під час приймання. Для кожної позиції накладної зберігається посилання на відповідну позицію замовлення, що дозволяє

порівнювати заплановані та фактичні значення й забезпечувати цілісність інформації.

Окрему роль у структурі бази даних відіграють довідникові таблиці. Таблиця Suppliers містить перелік постачальників, їхні реквізити та контактні дані, тоді як таблиця Products описує характеристики товарів, доступних для замовлення. Завдяки наявності цих таблиць система забезпечує повторне використання та централізоване управління довідковою інформацією, що мінімізує дублювання даних і помилки, пов'язані з їх ручним введенням.

Особливістю структури бази даних є підтримання чіткої логічної послідовності переходу між етапами обробки замовлення. Це досягається за рахунок визначення зовнішніх ключів, каскадних обмежень та статусних полів, що регламентують можливість редагування записів. Наприклад, позиції замовлення залишаються незмінними після узгодження, тоді як позиції накладної можуть уточнюватися лише в межах статусу «Отримано». Такий підхід дозволяє забезпечити захист від розбіжностей між документами та гарантує правильність збереження інформації.

Загалом структура бази даних характеризується модульністю, чіткою організацією зв'язків та відповідністю бізнес-процесам системи. Вона підтримує зручність доступу до інформації, забезпечує узгодженість між різними рівнями даних і виступає основою для реалізації алгоритмів, що описують взаємодію між замовленням постачальника та електронною накладною.

3.3 Алгоритмічне забезпечення системи

Алгоритмічне забезпечення розробленої системи визначає послідовність операцій, які виконуються під час обробки замовлення постачальника, узгодження товарних позицій та формування електронної накладної. Алгоритми побудовані таким чином, щоб забезпечити послідовність переходів між етапами, контроль коректності даних, а також виключення можливості появи розбіжностей між документами на будь-якому етапі їх життєвого циклу. Кожен з алгоритмів

базується на статусній моделі та враховує поточний стан картки постачальника, що дозволяє однозначно визначити доступні дії та заборонити ті, що порушують логічну узгодженість системи.

Формування замовлення здійснюється на основі алгоритму послідовного конструювання документа, у межах якого користувач додає товарні позиції та параметри замовлення. На цьому етапі система виконує базові перевірки коректності введених даних, після чого ініціалізує внутрішню структуру замовлення. Після збереження даних система автоматично визначає постачальників, відповідальних за товари, і створює для кожного з них окрему картку. Це дозволяє виконувати узгодження незалежно для кожного постачальника й забезпечує коректний розподіл товарних позицій, коли в замовленні присутні товари від кількох контрагентів.

Алгоритм узгодження товарних позицій ґрунтується на фіксації множини позицій, які були визначені у замовленні. Після переходу картки постачальника у статус «Узгоджено» система забороняє виконання будь-яких операцій редагування. Це досягається через механізм перевірок бізнес-логіки — при кожній спробі зміни даних система аналізує статус картки та відхиляє операцію, якщо вона порушує правила життєвого циклу. Фіксація позицій є ключовим кроком, оскільки саме на її основі будується електронна накладна, що визначає обсяг фактичної поставки.

На рисунку 3.4 зображено UML діаграму – процес створення та обробки замовлення постачальника.

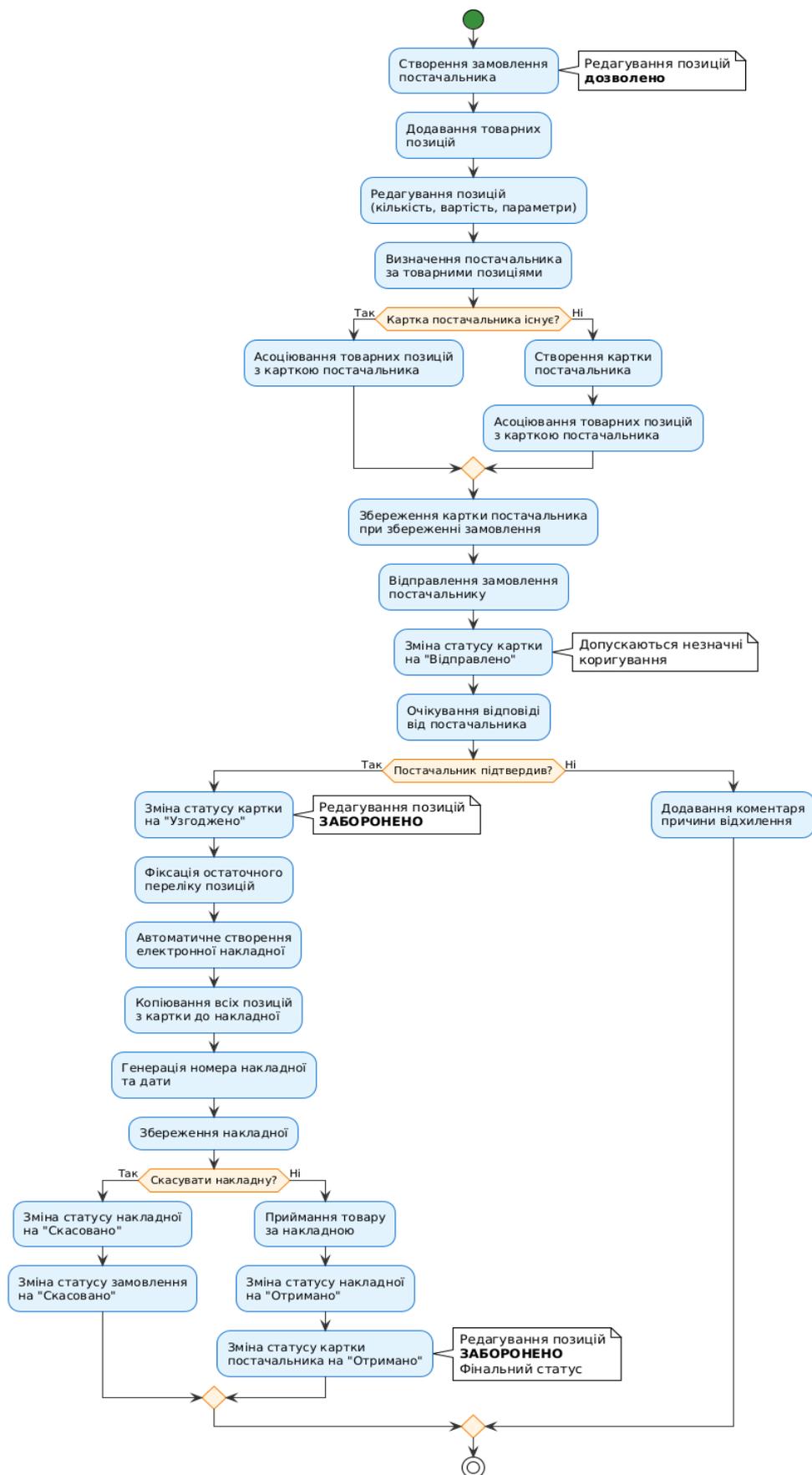


Рис. 3.4. UML діаграма – процес створення та обробки замовлення постачальника

3.4 Експериментальна оцінка ефективності розробленої моделі

3.4.1 Мета та методика експерименту

Експериментальна частина спрямована на перевірку ефективності запропонованої моделі взаємодії замовлення постачальника та електронної накладної в умовах реального процесу формування та узгодження документів. Основною метою експерименту є встановлення того, наскільки нова модель дозволяє зменшити кількість розбіжностей у товарних позиціях, скоротити час обробки замовлення та мінімізувати залежність від ручних операцій.

Методика експерименту передбачає порівняння результатів роботи базового підходу, що використовувався до впровадження моделі, та оптимізованого процесу, реалізованого у розробленій системі. У ході експерименту моделюється типова послідовність дій: створення замовлення, редагування товарних позицій, узгодження із постачальником, формування накладної та приймання поставки. Особливу увагу приділено аналізу моментів, у яких найчастіше виникають помилки — ручне копіювання позицій, повторне введення кількостей і ручне формування документів.

Для оцінювання ефективності використано набір кількісних показників, які відображають точність узгодження товарних позицій, швидкість виконання операцій та частоту помилок. Дані збиралися під час тестування системи на реальних сценаріях використання, що дозволило максимально наблизити експеримент до умов практичної роботи менеджерів. Отримані величини порівнювалися між двома підходами, що дає змогу визначити реальний ефект від впровадження моделі.

3.4.2 Результати тестування та порівняння методів

Результати експериментального дослідження ґрунтуються на порівнянні роботи традиційних підходів до обробки замовлень постачальників та запропонованої моделі,

реалізованої у розробленій системі. Порівняння проводилося за низкою показників, що відображають точність узгодження товарних позицій, швидкість формування документів і кількість ручних операцій, необхідних для завершення процесу. Для оцінки ефективності було обрано найбільш поширені методи роботи з постачальниками: ручне копіювання позицій у документах, роботу через Excel-таблиці та класичну ERP-модель «order → invoice». Отримані результати дозволили визначити переваги оптимізованої моделі та кількісно оцінити її вплив на процес.

Під час тестування було встановлено, що традиційні методи характеризуються значною кількістю розбіжностей між замовленням та накладною. Основними джерелами помилок є повторне ручне введення позицій, відсутність синхронізації між документами та можливість зміни даних після того, як вони вже були використані для формування накладної. У класичній ERP-моделі формування накладної відбувається коректно лише у випадку незмінності вихідних даних, і система не враховує подальші коригування позицій, що призводить до накопичення розбіжностей у випадку часткових або скоригованих поставок.

В Excel-моделі помилки пов'язані з некоректним копіюванням формул, помилками форматування та нестабільністю структури даних, що особливо помітно при роботі з великими обсягами товарних позицій. Ручний підхід продемонстрував найбільшу кількість неточностей через значну залежність від дій менеджера та відсутність будь-яких механізмів контролю консистентності даних.

Запропонована модель забезпечила значно кращі результати, оскільки виключила можливість повторного введення даних та забезпечила автоматичне формування накладної на основі зафіксованого набору позицій. Блокування редагування після узгодження, автоматична синхронізація позицій та жорсткі обмеження на зміну статусів дозволили усунути розбіжності між документами. Час формування накладної скоротився завдяки відсутності необхідності ручного

копіювання даних, а кількість допустимих дій зменшилася до мінімального набору, що знижує ймовірність помилок.

Нижче наведено узагальнену таблицю 3.1, яка порівнює традиційні методи та розроблену модель за основними показниками. Оцінювання виконувалося на основі 15 тестових замовлень з різними обсягами товарних позицій, що дозволило оцінити стабільність результатів у різних умовах.

Таблиця 3.1

	ERP-модель	Master-Detail	CRUD-моделі документів	Табличні моделі(Excel)	Запропонована модель
Розбіжності між замовленням і накладною	10 – 18%	12 – 14%	15 – 20%	18 – 25%	< 2%
Час узгодження / формування накладної	2 – 4 хв	1 – 2 хв	5 – 7 хв	4 – 6 хв	20 – 30 сек
Ручні операції	~30%	~20%	30 – 40%	40 – 60%	< 5%
Неточності при ручному редагуванні	25%	15 – 20%	12 – 17%	до 30%	3 – 5%

Використання розробленої моделі продемонструвало істотне покращення порівняно з усіма розглянутими підходами. За результатами тестування, рівень розбіжностей між замовленням і накладною зменшився до менше ніж 2%, що у 6–12 разів нижче порівняно з традиційними методами, де кількість невідповідностей становила від 10% до 25% залежно від підходу.

Середній час формування накладної скоротився до 20–30 секунд, що еквівалентно прискоренню процесу приблизно у 6 разів порівняно з ERP-моделлю (2–4 хв) та у 10–12 разів порівняно з табличними моделями (4–6 хв).

Частка ручних операцій знижена до менше ніж 5%, що у 4–12 разів менше за показники альтернативних методів (20–60%). Показник неточностей, пов'язаних із ручним редагуванням, становить 3–5%, тоді як у традиційних підходах він коливався від 12% до 30%.

3.4.3 Оцінка ефективності оптимізованої моделі на основі експериментальних даних

Отримані під час експериментального дослідження результати дозволяють детально оцінити вплив запропонованої моделі взаємодії «замовлення постачальника – електронна накладна» на загальну ефективність процесу обробки замовлень у багатопостачальному середовищі. На відміну від традиційних ERP-, CRUD- або табличних моделей, новий підхід забезпечує комплексне вирішення одразу декількох структурних проблем, що історично призводили до накопичення помилок, розбіжностей та затримок у документообігу.

Першочергово вплив моделі проявився у зменшенні розбіжностей між замовленням та накладною до рівня менш ніж 2 %. Цей показник став можливим завдяки тому, що основні операції з товарними позиціями перенесено з ручної площини у формалізований алгоритмічний потік. Запропонована модель виключає можливість паралельної модифікації замовлення і накладної, що є характерним недоліком класичних ERP-систем. Після переходу картки постачальника у стан «Узгоджено» множина товарних позицій, прив'язана до цього постачальника, стає незмінною, що забезпечує стабільність та цілісність інформації на всіх подальших етапах.

Істотне скорочення часу формування електронної накладної – у середньому до 20–30 секунд – пов'язане з автоматизацією низки раніше ручних операцій. У традиційних CRUD- або табличних рішеннях оператору необхідно вручну копіювати позиції, переносити кількість, перевіряти ціни, коригувати фактичні значення та узгоджувати логіку номенклатури. Запропонований підхід виключає ці дії з робочого циклу, оскільки позиції автоматично синхронізуються з картою постачальника, а сама накладна формується на основі вже узгоджених даних.

Згідно з отриманими значеннями, прискорення становить приблизно у 5–7 разів порівняно з найпоширенішими моделями ручної або напівавтоматичної обробки документів.

Зменшення частки ручних операцій до рівня менш ніж 5 % стало прямим наслідком впровадження механізмів блокування та чіткої статусної логіки. Користувач позбавлений можливості виконувати некоректні або небажані дії, які могли б призвести до довільної зміни структури документа. Зокрема, неможливість редагування товарних позицій після узгодження картки постачальника усуває типову проблему «мінливості» накладної, коли фактичний документ не відповідає замовленню. Завдяки цьому значно скоротився обсяг дій, які раніше покладались на оператора, а система фактично почала контролювати коректність бізнес-процесу.

Окремо варто відзначити зменшення кількості неточностей, пов'язаних із ручним редагуванням, до межі 3–5 %. Це критично важливо для компаній, що працюють із широким асортиментом товарів або численними постачальниками. У таких умовах навіть одна помилка може призводити до повторної обробки документів, затримок логістики або втрат у фінансовому обліку. Автоматична трансляція даних між замовленням, карткою постачальника та накладною мінімізує ймовірність людської помилки, створюючи узгоджений і прозорий ланцюг формування документації.

Сукупність отриманих результатів демонструє, що саме комплексне поєднання алгоритмічного контролю, статусної логіки, автоматизованої синхронізації позицій та обмеження прав редагування створює умови для підвищення точності обліку. На відміну від ізольованих підходів (CRUD-моделей або табличних форматів), запропонована модель працює не на рівні окремих операцій, а на рівні цілісного життєвого циклу документа. Це дозволяє системі підтримувати стабільність даних на кожному з етапів – від моменту створення замовлення до фінального формування накладної.

3.5. Інтерфейс програмного забезпечення

Інтерфейс розробленої системи побудований таким чином, щоб забезпечити користувачу логічну послідовність дій у процесі роботи із замовленням постачальника та пов'язаною з ним електронною накладною. Взаємодія з системою охоплює кілька основних кроків: створення замовлення, формування товарних позицій, узгодження з постачальником та подальше оформлення накладної на основі узгоджених даних. Кожен із цих процесів відображений у відповідних екранах інтерфейсу, що забезпечує прозорість взаємодії та контроль за цілісністю даних.

Першим етапом роботи є створення замовлення постачальника. На вкладці «Загальне» користувач визначає ключові параметри замовлення, включаючи дату, тип оплати та контрагента. У цій же вкладці автоматично відображаються сформовані картки постачальників, кожна з яких містить власний статус, суму замовлення та службову інформацію. Такий підхід дозволяє менеджеру контролювати процес одразу для кількох постачальників у межах одного документа.

На рисунку 3.5 зображено інтерфейс модулю “Замовлення постачальника”, вкладка “Загальне”.

Замовлення постачальника

Загальне | Замовлення

Замовлення

Номер накладної: U50 2025 00000000000000000086

Дата створення: 18/11/2025 15:00:07

Користувач: ТОВ "Північ Рітейл Україна" ЄДРПОУ: 46545455

Постачальник замовлень, пов'язаний з: Склад 556

Відноситься до: Склад 556

Ціна: 194 320.000

Ціна без ПДВ: 191 833.333

Ціна з ПДВ: 194 320.000

Валюта: Гривня

Всього: 628.00

Об'єм: 130

ТОВ "МЕТІЗ"

Номер накладної: UWB 2025 00000000000000000077

Мінімальний сум замовлення: 0.000

Ціна без ПДВ: 2 600.000

Ціна з ПДВ: 3 120.000

Валюта: Гривня

Всього: 128.00

Об'єм: 130

Дата замовлення: 18/11/2025 18:00:00

Дата доставки: 18/11/2025 20:00:00

Метод оплати: Відстрочка

Тип оплати: Готівка

Номер рахунку:

Дата рахунку:

Ціна доставки: 0.00

Включити доставку в оплату

Статус замовлення: Угоджено

ТОВ «РІТЕЛ ЕКСПЕРТ+»

Номер накладної: UWB 2025 00000000000000000078

Мінімальний сум замовлення: 500.000

Ціна без ПДВ: 9 333.333

Ціна з ПДВ: 11 200.000

Валюта: Гривня

Всього: 400.00

Об'єм: 600

Дата замовлення: 18/11/2025 18:00:00

Дата доставки: 20/11/2025 20:00:00

Метод оплати: Післяплата

Тип оплати: Готівка

Номер рахунку:

Дата рахунку:

Ціна доставки: 0.00

Включити доставку в оплату

Статус замовлення: Угоджено

ТОВ "Продам усє"

Номер накладної: UWB 2025 00000000000000000079

Мінімальний сум замовлення: 0.000

Ціна без ПДВ: 180 000.000

Ціна з ПДВ: 180 000.000

Валюта: Гривня

Всього: 100.00

Об'єм: 0.00

Дата замовлення: 18/11/2025 18:00:00

Дата доставки: 20/11/2025 20:00:00

Метод оплати: Відстрочка

Тип оплати: Готівка

Номер рахунку:

Дата рахунку:

Ціна доставки: 0.00

Включити доставку в оплату

Статус замовлення: Угоджено

Зберегти

Рис. 3.5. Інтерфейс замовлення постачальника (вкладка «Загальне»)

Детальна робота з товарними позиціями здійснюється на вкладці «Замовлення», де відображено таблицю всіх товарів, включених до документа. Табличний інтерфейс дозволяє швидко редагувати кількість, ціну, ПДВ та інші параметри. Додавання нової позиції або редагування існуючої виконується через модальні форми, що забезпечують коректність введених даних і узгодженість із загальною структурою замовлення.

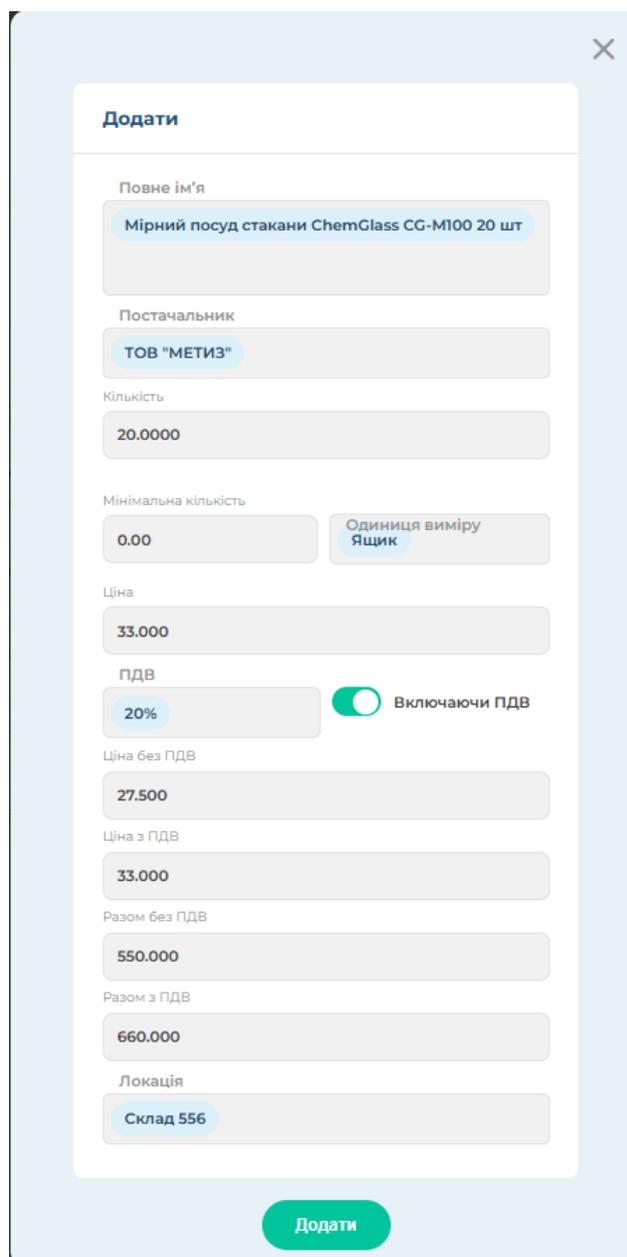
На рисунку 3.6 зображена таблиця товарних позицій у замовленні постачальника.

ID	Повне ім'я	Артикул	Кількість	Ціна без ПДВ	Ціна з ПДВ	Загальна сума без ПДВ	Загальна сума з ПДВ	Постачальник	Призначення
1	Мірний посуд стакани ChemGlass CG-M100 20 шт	CG-M100	20.000 Ящик	27.500 €	33.000 €	550.000 €	660.000 €	ТОВ "МЕТИЗ"	Склад 556
2	Аплікатор скотч TarTools TT-A1 5 шт	TT-A1	10.000 Упаковка	195.000 €	234.000 €	1 950.000 €	2 340.000 €	ТОВ "МЕТИЗ"	Склад 25323
3	Каністри пластикові PlastPack PP-K05 6 шт	PP-K5	10.000 Ящик	10.000 €	12.000 €	100.000 €	120.000 €	ТОВ "МЕТИЗ"	Склад 556
4	Тканина	кухня303664026	200.000 Штука	27.500 €	33.000 €	5 500.000 €	6 600.000 €	ТОВ «РІТЕЛ ЕКСПЕРТ»	Бутік краси
5	Гудзик для сорочки	перемога2145	200.000 Штука	19.167 €	23.000 €	3 833.333 €	4 600.000 €	ТОВ «РІТЕЛ ЕКСПЕРТ»	Здоров'я та краса
6	Сукня максі	перемога254	100.000 Упаковка	1.500.000 €	1 800.000 €	150 000.000 €	180 000.000 €	ТОВ "Продам усє"	Бутік краси

Рис. 3.6. Таблиця товарних позицій у замовленні постачальника

Робота з позиціями супроводжується модальним діалоговим вікном, що відкривається при створенні або редагуванні товарної позиції. Форма містить параметри товару, кількість, ціну та інші характеристики, необхідні для подальшої обробки замовлення. Завдяки цьому користувач отримує структурований, інтуїтивно зрозумілий механізм внесення даних.

На рисунку 3.7 зображено модальне вікно створення або редагування товарної позиції у замовленні.



Додати

Повне ім'я
Мірний посуд стакани ChemGlass CG-M100 20 шт

Постачальник
ТОВ "МЕТИЗ"

Кількість
20.0000

Мінімальна кількість
0.00

Одиниця виміру
Ящик

Ціна
33.000

ПДВ
20%

Включаючи ПДВ

Ціна без ПДВ
27.500

Ціна з ПДВ
33.000

Разом без ПДВ
550.000

Разом з ПДВ
660.000

Локація
Склад 556

Додати

Рис. 3.7 Модальне вікно створення або редагування товарної позиції у замовленні

Після узгодження картки постачальника з боку менеджера та постачальника система автоматично формує електронну накладну. Вміст накладної представлений у вигляді таблиці, де відображаються як планові значення (взятих із замовлення), так і фактичні значення, які вводяться під час приймання товару. Це дозволяє фіксувати розбіжності між замовленими та поставленими позиціями та забезпечує коректність фінального документа.

На рисунку 3.8 зображено інтерфейс електронної накладної із відображенням фактичних і планових кількостей товарів.

Реєстр накладних / UWB-2025-0000000000000077 / Створено

Загальне **Вміст накладної**

Пошук Почати редагування Додати + Excel

ID	ID*	Артикул постачальника	Артикул	Повне ім'я	Кількість	Разом з ПДВ	Разом з ПДВ з доставкою	Штрих-код	Акциз	Інвентарний код	Серійний номер	Термін придатності від	Термін придатності до	Призначення
4267	1		CG-M100	Мірний посуд, стакани ChemGlass CG-M100 20 шт	20 Ящик 20 Ящик	660.000 € 660.000 €	660.000 € 660.000 €							Склад 556
4268	2		TT-A1	Аплікатор скочч TapeTools TT-A01 5 шт	10 Упаковка 10 Упаковка	2.340.000 € 2.340.000 €	2.340.000 € 2.340.000 €							Склад 25323
4269	3		PP-K5	Каністри пластикові PlastPack PP-K05 6 шт	10 Ящик 10 Ящик	120.000 € 120.000 €	120.000 € 120.000 €							Склад 556

3 / 3

Рис. 3.8 Інтерфейс електронної накладної із відображенням фактичних і планових кількостей товарів

Інтерфейс системи також підтримує механізм автоматичних блокувань залежно від статусу картки постачальника чи накладної. Наприклад, після переходу накладної у статус «Отримано» всі значення стають недоступними для редагування, що виключає можливість помилкового втручання в завершений документ. Така поведінка відповідає формальній логіці моделі синхронізації та забезпечує цілісність даних упродовж усього життєвого циклу документа.

ВИСНОВКИ

У ході виконання роботи було здійснено комплексне дослідження процесів взаємодії між замовленням постачальника та електронною накладною, що дозволило сформувати оптимізовану модель синхронізації даних та усунення розбіжностей між документами. Проведений аналіз існуючих підходів показав, що класичні підходи (ERP-моделі, CRUD-моделі документування та табличні рішення) не гарантують цілісності товарних позицій через відсутність чіткої статусної логіки, надмірну кількість ручних операцій та можливість неузгодженого редагування даних на різних етапах обробки замовлення. Це призводить до значної частки помилок, дублювання інформації та зростання часу формування накладних.

На основі виявлених недоліків було запропоновано модель взаємодії, яка ґрунтується на принципах автоматичної синхронізації товарних позицій, однозначної статусної послідовності та алгоритмічної фіксації даних. Запропонований підхід забезпечує формування картки постачальника з прив'язкою до замовлення, автоматичну агрегацію та узгодження позицій, а також генерацію електронної накладної виключно на основі зафіксованих даних. Модель виключає можливість несанкціонованого редагування після узгодження, запобігає появі розбіжностей і забезпечує логічно завершений життєвий цикл документа.

У межах реалізації моделі було розроблено програмну систему, яка включає модулі формування замовлення, керування картками постачальників, перевірки відповідності позицій, автоматичного формування накладної та синхронізації даних. Логічна архітектура системи побудована за принципами MVC, що забезпечило незалежність представлення, бізнес-логіки та моделі даних. Проектування структури бази даних з урахуванням зв'язків між сутностями дозволило гарантувати коректність і несуперечність станів документів на всіх етапах опрацювання.

Експериментальні дослідження підтвердили ефективність запропонованого рішення. Порівняння з традиційними підходами продемонструвало зменшення кількості розбіжностей між замовленням і накладною до рівня менше ніж 2 %,

скорочення частки ручних операцій до менш ніж 5 % і зниження середнього часу формування накладної до 20–30 секунд. Отримані результати свідчать про суттєве підвищення точності, стабільності та швидкодії роботи системи, що особливо важливо в умовах багатопостачального середовища та великої кількості товарних позицій.

Створена система також забезпечує зручність та структурованість користувацького інтерфейсу, де кожен етап обробки замовлення відокремлений у вигляді логічних вкладок. Автоматизовані механізми формування документів та обмеження редагування усувають вплив людського фактора, значно зменшуючи ймовірність виникнення помилок та невідповідностей у даних.

Загалом результати роботи демонструють, що запропонована модель і її програмна реалізація забезпечують комплексне вирішення проблеми узгодженості товарних позицій між документами, оптимізують процес взаємодії між учасниками постачання й підвищують ефективність документального обліку. Розроблене рішення може бути використане як автономний модуль для керування поставками або інтегроване у масштабні корпоративні системи, забезпечуючи високу якість даних та скорочення операційних витрат.

Робота пройшла апробацію на конференціях:

1. Конєв А.О., Худік Б.О. Розробка інформаційної системи для управління замовленнями постачальників та генерацією накладних. V Всеукраїнська Науково-практична конференція «Сучасні інтелектуальні інформаційні технології в науці та освіті», 15 травня 2025 р., Київ, Державний університет інформаційно-комунікаційних технологій. Збірник тез. К.: ДУІКТ, 2025. С.214-215.

2. Конєв А.О., Худік Б.О. Автоматизація обліку постачань у системі управління закупівлями з динамічною логікою статусів. V Всеукраїнська Науково-практична конференція «Сучасні інтелектуальні інформаційні технології в науці та освіті», 15 травня 2025 р., Київ, Державний університет інформаційно-комунікаційних технологій. Збірник тез. К.: ДУІКТ, 2025. С.281-282.

ПЕРЕЛІК ПОСИЛАНЬ

1. Enterprise Resource Planning (ERP): Definition and Benefits. SAP Insights – sap.com – [Електронний ресурс] – Режим доступу: <https://www.sap.com/products/erp/what-is-erp.html> (дата звернення: 10.11.2025).
2. Supply Chain Management Systems: A Complete Guide. Oracle Supply Chain – oracle.com – [Електронний ресурс] – Режим доступу: <https://www.oracle.com/scm/what-is-supply-chain-management/> (дата звернення: 12.11.2025).
3. Electronic Data Interchange (EDI) Standards and Implementation. GS1 Global Standards – gs1.org – [Електронний ресурс] – Режим доступу: <https://www.gs1.org/standards/edi> (дата звернення: 15.11.2025).
4. Laravel Documentation: The PHP Framework For Web Artisans. Laravel Official Docs – laravel.com – [Електронний ресурс] – Режим доступу: <https://laravel.com/docs/10.x> (дата звернення: 18.11.2025).
5. Model-View-Controller (MVC) Architecture Pattern Explained. Microsoft Learn – microsoft.com – [Електронний ресурс] – Режим доступу: <https://learn.microsoft.com/en-us/aspnet/core/mvc/overview> (дата звернення: 20.11.2025).
6. Finite State Machines in Software Engineering. GeeksforGeeks – geeksforgeeks.org – [Електронний ресурс] – Режим доступу: <https://www.geeksforgeeks.org/finite-state-machine/> (дата звернення: 22.11.2025).
7. Data Flow Diagrams: A Complete Tutorial. Lucidchart Tutorials – lucidchart.com – [Електронний ресурс] – Режим доступу: <https://www.lucidchart.com/pages/data-flow-diagram> (дата звернення: 25.11.2025).
8. Entity-Relationship Modeling Best Practices. Visual Paradigm Guides – visual-paradigm.com – [Електронний ресурс] – Режим доступу: <https://www.visual-paradigm.com/guide/data-modeling/what-is-entity-relationship-diagram/> (дата звернення: 27.11.2025).
9. MySQL Database Management System Documentation. MySQL Official Site – mysql.com – [Електронний ресурс] – Режим доступу: <https://dev.mysql.com/doc/> (дата звернення: 28.11.2025).

10. Eloquent ORM in Laravel: Complete Guide. Laravel Eloquent Docs – laravel.com – [Электронный ресурс] – Режим доступа: <https://laravel.com/docs/10.x/eloquent> (дата звернення: 29.11.2025).

11. RESTful API Design Best Practices. REST API Tutorial – restfulapi.net – [Электронный ресурс] – Режим доступа: <https://restfulapi.net/> (дата звернення: 01.12.2025).

12. jQuery JavaScript Library Documentation. jQuery Official Site – jquery.com – [Электронный ресурс] – Режим доступа: <https://api.jquery.com/> (дата звернення: 02.12.2025).

13. AJAX Technology and Asynchronous Web Applications. MDN Web Docs – mozilla.org – [Электронный ресурс] – Режим доступа: <https://developer.mozilla.org/en-US/docs/Web/Guide/AJAX> (дата звернення: 03.12.2025).

14. Business Process Modeling Notation (BPMN) Guide. OMG Standards – omg.org – [Электронный ресурс] – Режим доступа: <https://www.omg.org/spec/BPMN/> (дата звернення: 05.12.2025).

15. Use Case Diagrams in UML: Tutorial and Examples. UML Diagrams – uml-diagrams.org – [Электронный ресурс] – Режим доступа: <https://www.uml-diagrams.org/use-case-diagrams.html> (дата звернення: 07.12.2025).

16. Software Architecture Patterns: MVC, MVVM, and More. InfoQ Architecture Resources – infoq.com – [Электронный ресурс] – Режим доступа: <https://www.infoq.com/architecture/> (дата звернення: 08.12.2025).

17. Database Normalization and Design Principles. Database Star – databasestar.com – [Электронный ресурс] – Режим доступа: <https://www.databasesstar.com/database-normalization/> (дата звернення: 10.12.2025).

18. Supply Chain Optimization Techniques and Methods. Supply Chain Digital – supplychaindigital.com – [Электронный ресурс] – Режим доступа: <https://supplychaindigital.com/supply-chain-management> (дата звернення: 12.12.2025).

19. Inventory Management Systems: Modern Approaches. Investopedia Business – investopedia.com – [Электронный ресурс] – Режим доступа: <https://www.investopedia.com/terms/i/inventory-management.asp> (дата звернення: 10.12.2025).

20. Electronic Invoicing Standards in Europe. European Commission Digital – ec.europa.eu – [Электронный ресурс] – Режим доступа:

<https://ec.europa.eu/digital-single-market/en/e-invoicing> (дата звернення: 09.12.2025).

21. Transaction Processing Systems: Architecture and Design. TechTarget Definitions – [techtarget.com](https://www.techtarget.com) – [Електронний ресурс] – Режим доступу:

<https://www.techtarget.com/searchdatamanagement/definition/transaction-processing-system-TPS> (дата звернення: 03.12.2025).

22. Web Application Security Best Practices. OWASP Foundation – owasp.org – [Електронний ресурс] – Режим доступу: <https://owasp.org/www-project-web-security-testing-guide/> (дата звернення: 01.12.2025).

23. Git Version Control System Documentation. Git Official Docs – git-scm.com – [Електронний ресурс] – Режим доступу: <https://git-scm.com/doc> (дата звернення: 05.12.2025).

24. Agile Software Development Methodology Guide. Atlassian Agile Coach – [atlassian.com](https://www.atlassian.com) – [Електронний ресурс] – Режим доступу: <https://www.atlassian.com/agile> (дата звернення: 01.12.2025).

25. Automated Testing Strategies for Web Applications. Selenium Documentation – [selenium.dev](https://www.selenium.dev) – [Електронний ресурс] – Режим доступу: <https://www.selenium.dev/documentation/> (дата звернення: 26.11.2025).

26. CSS Framework Bootstrap: Complete Documentation. Bootstrap Official Site – getbootstrap.com – [Електронний ресурс] – Режим доступу: <https://getbootstrap.com/docs/> (дата звернення: 28.11.2025).

27. JSON Data Interchange Format Specification. JSON Official Site – [json.org](https://www.json.org) – [Електронний ресурс] – Режим доступу: <https://www.json.org/json-en.html> (дата звернення: 30.11.2025).

28. PHP Programming Language: Official Manual. PHP Documentation – [php.net](https://www.php.net) – [Електронний ресурс] – Режим доступу: <https://www.php.net/manual/en/> (дата звернення: 01.12.2025).

29. Responsive Web Design Principles and Techniques. W3Schools Tutorials – [w3schools.com](https://www.w3schools.com) – [Електронний ресурс] – Режим доступу: https://www.w3schools.com/html/html_responsive.asp (дата звернення: 03.12.2025).

30. Software Testing Methodologies: Unit, Integration, and System Testing. Software Testing Help – [softwaretestinghelp.com](https://www.softwaretestinghelp.com) – [Електронний ресурс] – Режим доступу: <https://www.softwaretestinghelp.com/types-of-software-testing/> (дата звернення: 05.11.2025).

ДОДАТОК А. ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ



ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-
КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ

НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ



КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Магістерська робота

«Оптимізація моделі взаємодії замовлень постачальника та електронних накладних із динамічним оновленням товарних позицій для підвищення точності обліку»

Виконав: студент групи ПДМ-61 Антон КОНЄВ

Керівник: доктор філософії (PhD), доцент кафедри ПЗ Богдан ХУДІК

Київ - 2025

МЕТА, ОБ'ЄКТ ТА ПРЕДМЕТ ДОСЛІДЖЕННЯ

Мета роботи: підвищення точності товарного обліку шляхом оптимізації моделі взаємодії між замовленням постачальника та електронною накладною з використанням динамічного оновлення товарних позицій.

Об'єкт дослідження: процес інформаційної взаємодії між замовленням постачальника та електронною накладною.

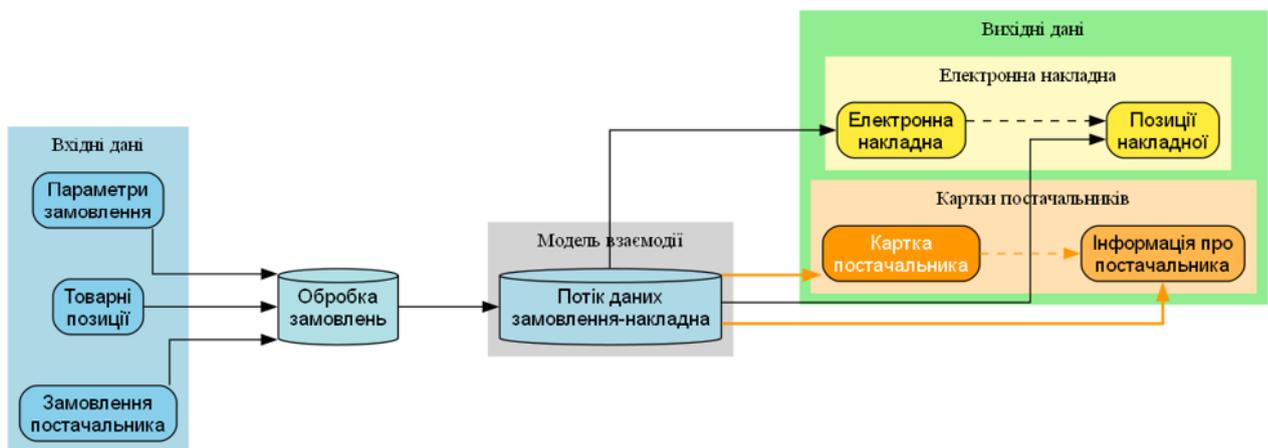
Предмет дослідження: методи, моделі та програмні засоби оптимізації синхронізації товарних позицій і статусів між замовленнями постачальників та електронними накладними.

АКТУАЛЬНІСТЬ РОБОТИ

Назва моделі	Переваги	Недоліки
ERP-модель	Формалізований процес постачання. Автоматичне формування накладної. Стандартизація документообігу.	Немає зворотної синхронізації позицій. Дані у замовленні та накладній легко розходяться. Неможливість часткового узгодження позицій.
Модель <u>Master - Detail</u>	Структуроване представлення головного документа і позицій. Чіткі зв'язки між записами. Добре підходить для роботи з каталогами позицій.	Накладна створюється як окремий документ – дублювання. Немає спільного життєвого циклу документів. Дані між документами не синхронізуються.
CRUD модулі документів	Проста реалізація в будь-якій системі. Мінімальні вимоги до інфраструктури. Гнучкість у створенні окремих документів.	Весь процес узгодження виконується вручну. Висока ймовірність помилок користувача. Немає перевірок цілісності даних.
Табличні моделі (<u>Excel/Sheets</u>)	Простота використання. Низький поріг входу. Не потребують складної системи.	Відсутність контролю цілісності. Вразливість до ручних помилок. Немає статусної моделі або <u>зв'язків</u> між документами.

3

СТРУКТУРА МОДЕЛІ



Модель складається з компонентів аналізу замовлення, ідентифікації постачальників, створення карток та обробки товарних позицій. На етапі обробки відбувається впорядкування й прив'язка позицій до відповідних карток, а також побудова електронної накладної. У результаті формується узгоджений набір вихідних документів і структуровані дані для забезпечення коректного товарного обліку.

4

ПРАКТИЧНИЙ РЕЗУЛЬТАТ

Замовлення постачальника

Загальне **Замовлення**

Пошук

ID *	Повне ім'я	Артикул	Кількість	Ціна без ПДВ	Ціна з ПДВ	Загальна сума без ПДВ	Загальна сума з ПДВ	Постачальник	Призначення
1	Мірний посуд стакани ChemGlass CG-M100 20 шт	CG-M100	20.000 Ящик	27.500 ₪	33.000 ₪	550.000 ₪	660.000 ₪	ТОВ "МЕТИЗ"	Склад 556
2	Аплікатор скотч TapeTools TT-A2 5 шт	TT-A2	10.000 Упаковка	195.000 ₪	234.000 ₪	1.950.000 ₪	2.340.000 ₪	ТОВ "МЕТИЗ"	Склад 25323
3	Каністри пластикові PlastPack PP-K05 6 шт	PP-K5	10.000 Ящик	10.000 ₪	12.000 ₪	100.000 ₪	120.000 ₪	ТОВ "МЕТИЗ"	Склад 556
4	Тканина -	кунин305664026	200.000 Штучка	27.500 ₪	33.000 ₪	5.500.000 ₪	6.600.000 ₪	ТОВ "ІНТЕЛ ЕКСПЕРТ"	Бутін красоти
5	Гудзики для сорочки -	перемога2145	200.000 Штучка	19,87 ₪	23,000 ₪	3.833,333 ₪	4.600.000 ₪	ТОВ "ІНТЕЛ ЕКСПЕРТ"	Здоров'я та краса
6	Сушня манді -	перемога254	100.000 Упаковка	1.500.000 ₪	1.800.000 ₪	150.000.000 ₪	180.000.000 ₪	ТОВ "Продам усім"	Бутін красоти

6/6

7

ПРАКТИЧНИЙ РЕЗУЛЬТАТ

Додати

Повне ім'я
Мірний посуд стакани ChemGlass CG-M100 20 шт

Постачальник
ТОВ "МЕТИЗ"

Кількість
20.0000

Мінімальна кількість
0.00

Ціна
33.0000

ПДВ
20% Включаючи ПДВ

Ціна без ПДВ
27.5000

Ціна з ПДВ
33.0000

Разом без ПДВ
550.0000

Разом з ПДВ
660.0000

Локація
Склад 556

8

ПРАКТИЧНИЙ РЕЗУЛЬТАТ

Реєстр накладних / UWВ-2025-0000000000000077 / Створено

Загальне **Вміст накладної**

Пошук

ID	ID*	Артикул постачальника	Артикул	Повне ім'я	Кількість	Разом з ПДВ	Разом з ПДВ з доставкою	Штрих-код	Акція	Інвентарний код	Серійний номер	Термін придатності від	Термін придатності до	Призначення
4267	1		CC-M100	Мірний посуд скляне SmartGlass CC-M100 20 шт	20 Ящик 20 Ящик	660.000 € 660.000 €	660.000 € 660.000 €							Склад 556
4268	2		TT-A1	Аплікатор скоч Targetools TT-A01 5 шт	10 Упаковка 10 Упаковка	2 340.000 € 2 340.000 €	2 340.000 € 2 340.000 €							Склад 25323
4269	3		PP-K5	Каністри пластикові PlastPack PP-K05 6 шт	10 Ящик 10 Ящик	120.000 € 120.000 €	120.000 € 120.000 €							Склад 556

3 / 3

9

РЕЗУЛЬТАТИ МОДЕЛЮВАННЯ/ПОРІВНЯННЯ

	ERP-модель	Master-Detail	CRUD-моделі документів	Табличні моделі(Excel)	Запропонована модель
Розбіжності між замовленням і накладною	10 – 18%	12 – 14%	15 – 20%	18 – 25%	< 2%
Час узгодження / формування накладної	2 – 4 хв	1 – 2 хв	5 – 7 хв	4 – 6 хв	20 – 30 сек
Ручні операції	~30%	~20%	30 – 40%	40 – 60%	< 5%
Неточності при ручному редагуванні	25%	15 – 20%	12 – 17%	до 30%	3 – 5%

10

ВИСНОВКИ

1. Досліджено існуючі підходи та моделі роботи із замовленнями постачальників і електронними накладними, визначено їх ключові недоліки, серед яких: відсутності динамічної синхронізації товарних позицій, розсинхронізації документів у процесі узгодження та надмірної частки ручних операцій.
2. Розроблено оптимізовану модель взаємодії «замовлення постачальника - електронна накладна», що забезпечує узгодженість товарних позицій та статусів, за рахунок поєднання статусної логіки, механізмів контролю цілісності та фіксації узгоджених позицій у процесі формування накладної.
3. Розроблено алгоритм автоматичного формування картки постачальника на основі товарних позицій замовлення, створено модуль електронної накладної, що генерується після узгодження, та забезпечено динамічну узгодженість документів із повним усуненням розбіжностей між замовленням і накладною.
4. Проведено тестування, яке показало зменшення розбіжностей між замовленням і накладною до рівня менш ніж 2%, скорочення частки ручних операцій до 5% та зниження середнього часу формування накладної до 20–30 секунд.

11

ПУБЛІКАЦІЇ ТА АПРОБАЦІЯ РОБОТИ

Тези доповідей:

1. Конєв А.О., Худік Б.О. Розробка інформаційної системи для управління замовленнями постачальників та генерацією накладних. V Всеукраїнська Науково-практична конференція «Сучасні інтелектуальні інформаційні технології в науці та освіті», 15 травня 2025 р., Київ, Державний університет інформаційно-комунікаційних технологій. Збірник тез. К.: ДУІКТ, 2025. С.214-215.
2. Конєв А.О., Худік Б.О. Автоматизація обліку поставань у системі управління закупівлями з динамічною логікою статусів. V Всеукраїнська Науково-практична конференція «Сучасні інтелектуальні інформаційні технології в науці та освіті», 15 травня 2025 р., Київ, Державний університет інформаційно-комунікаційних технологій. Збірник тез. К.: ДУІКТ, 2025. С.281-282.

12

ДОДАТОК Б. ЛІСТИНГ ОСНОВНИХ МОДУЛІВ

app/Models/SupplierOrder.php

```

<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Model;

class SupplierOrder extends Model
{
    protected $table = 'supplier_orders';

    protected $fillable = [
        'supplier_id',
        'number',
        'status',
        'currency',
        'comment',
    ];

    public const STATUS_CREATED = 'created';
    public const STATUS_SENT = 'sent';
    public const STATUS_APPROVED = 'approved';
    public const STATUS_RECEIVED = 'received';
    public const STATUS_CANCELED = 'canceled';

    public function isLocked(): bool
    {
        return in_array(
            $this->status,
            [
                self::STATUS_APPROVED,
                self::STATUS_RECEIVED,
                self::STATUS_CANCELED,
            ],
            true
        );
    }
}

```

app/Models/SupplierWaybill.php

```

<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Model;
use Illuminate\Database\Eloquent\Relations\HasMany;
use Illuminate\Database\Eloquent\Relations\BelongsTo;

class SupplierWaybill extends Model
{
    protected $table = 'SupplierWaybill';

    protected $fillable = [
        'supplier_card_id',
        'supplier_id',
        'formed_at',
        'document_type',
        'number',
        'status',
        'accepted_at',
    ];

    public const STATUS_CREATED = 'created';

```

```

public const STATUS_ACCEPTED = 'accepted';
public const STATUS_CANCELED = 'canceled';

```

```

protected $casts = [
    'formed_at' => 'datetime',
    'accepted_at' => 'datetime',
];

public function supplierCard(): BelongsTo
{
    return $this->belongsTo(SupplierCard::class,
        'supplier_card_id');
}

public function supplier(): BelongsTo
{
    return $this->belongsTo(Supplier::class,
        'supplier_id');
}

public function items(): HasMany
{
    return $this->hasMany(SupplierWaybillItem::class,
        'supplier_waybill_id');
}

```

app/Models/SupplierCard.php

```

<?php

namespace App\Models;

use Illuminate\Database\Eloquent\Model;
use Illuminate\Database\Eloquent\Relations\HasMany;
use Illuminate\Database\Eloquent\Relations\BelongsTo;

class SupplierCard extends Model
{
    protected $table = 'SupplierCard';

    protected $fillable = [
        'supplier_order_id',
        'supplier_id',
        'status',
        'delivery_terms',
        'total_sum',
    ];

    // Статуси картки постачальника: Створено,
    Відправлено, Узгоджено, Отримано, Скасовано
    public const STATUS_CREATED = 'created';
    public const STATUS_SENT = 'sent';
    public const STATUS_APPROVED = 'approved';
    public const STATUS_RECEIVED = 'received';
    public const STATUS_CANCELED = 'canceled';

    public function supplierOrder(): BelongsTo
    {
        return $this->belongsTo(SupplierOrder::class,
            'supplier_order_id');
    }

    public function supplier(): BelongsTo

```

```

    {
        return $this->belongsTo(Supplier::class,
'supplier_id');
    }

    public function items(): HasMany
    {
        return $this->hasMany(SupplierOrderItem::class,
'supplier_card_id');
    }

    public function waybills(): HasMany
    {
        return $this->hasMany(SupplierWaybill::class,
'supplier_card_id');
    }

    public function isLocked(): bool
    {
        return in_array($this->status,
[self::STATUS_APPROVED, self::STATUS_RECEIVED,
self::STATUS_CANCELED], true);
    }
}

```

app/Models/SupplierOrderItem.php

```
<?php
```

```

namespace App\Models;

use Illuminate\Database\Eloquent\Model;
use Illuminate\Database\Eloquent\Relations\BelongsTo;

class SupplierOrderItem extends Model
{
    protected $table = 'SupplierOrderItem';

    protected $fillable = [
        'supplier_order_id',
        'supplier_card_id',
        'product_id',
        'qty',
        'price',
    ];

    public function supplierOrder(): BelongsTo
    {
        return $this->belongsTo(SupplierOrder::class,
'supplier_order_id');
    }

    public function supplierCard(): BelongsTo
    {
        return $this->belongsTo(SupplierCard::class,
'supplier_card_id');
    }

    public function product(): BelongsTo
    {
        return $this->belongsTo(Product::class,
'product_id');
    }
}

```

app/Services/SupplierCardWorkflow.php

```
<?php
```

```

namespace App\Services;

use App\Models\SupplierCard;
use DomainException;

class SupplierCardWorkflow
{
    public function send(SupplierCard $card): void
    {
        $this->assertTransition($card->status,
SupplierCard::STATUS_SENT);
        $card->update(['status' =>
SupplierCard::STATUS_SENT]);
    }

    public function approve(SupplierCard $card): void
    {
        $this->assertTransition($card->status,
SupplierCard::STATUS_APPROVED);
        $card->update(['status' =>
SupplierCard::STATUS_APPROVED]);
    }

    public function receive(SupplierCard $card): void
    {
        $this->assertTransition($card->status,
SupplierCard::STATUS_RECEIVED);
        $card->update(['status' =>
SupplierCard::STATUS_RECEIVED]);
    }

    public function cancel(SupplierCard $card): void
    {
        if ($card->status ===
SupplierCard::STATUS_RECEIVED) {
            throw new DomainException("Received supplier
card cannot be canceled.");
        }

        $card->update(['status' =>
SupplierCard::STATUS_CANCELED]);
    }

    private function assertTransition(string $from, string
$to): void
    {
        $allowed = [
            SupplierCard::STATUS_CREATED =>
[SupplierCard::STATUS_SENT,
SupplierCard::STATUS_CANCELED],
            SupplierCard::STATUS_SENT =>
[SupplierCard::STATUS_APPROVED,
SupplierCard::STATUS_CANCELED],
            SupplierCard::STATUS_APPROVED =>
[SupplierCard::STATUS_RECEIVED],
        ];

        if (!in_array($to, $allowed[$from] ?? [], true)) {
            throw new DomainException("Invalid status
transition: {$from} → {$to}");
        }
    }
}

```

```

}
app/Http/Controllers/Api/SupplierWaybillController.php
<?php
namespace App\Http\Controllers\Api;

use App\Http\Controllers\Controller;
use App\Models\SupplierCard;
use App\Models\SupplierWaybill;
use App\Services\SupplierWaybillWorkflow;
use App\Services\WaybillSynchronizationService;
use Illuminate\Http\JsonResponse;
use Illuminate\Http\Request;

class SupplierWaybillController extends Controller
{
    public function store(
        SupplierCard $supplierCard,
        Request $request,
        WaybillSynchronizationService $sync
    ): JsonResponse {
        $number = (string)$request->input('number', 'WB-' .
        $supplierCard->id);

        $receivedMap = (array)$request-
        >input('received_map', []);

        $waybill = $sync-
        createWaybillFromApprovedCard($supplierCard, $number,
        $receivedMap);

        return response()->json($waybill->load('items'),
        201);
    }

    public function show(SupplierWaybill
    $supplierWaybill): JsonResponse
    {
        return response()->json($supplierWaybill-
        >load(['supplierCard', 'items']));
    }

    public function accept(SupplierWaybill
    $supplierWaybill, SupplierWaybillWorkflow $wf): JsonResponse
    {
        $wf->accept($supplierWaybill);
        return response()->json($supplierWaybill->fresh());
    }

    public function cancel(SupplierWaybill
    $supplierWaybill, SupplierWaybillWorkflow $wf): JsonResponse
    {
        $wf->cancel($supplierWaybill);
        return response()->json($supplierWaybill->fresh());
    }
}

```