

Міністерство освіти і науки України  
Національний технічний університет  
«Дніпровська політехніка»

Факультет інформаційних технологій  
(факультет)

Кафедра системного аналізу та управління  
(повна назва)

**ПОЯСНОВАЛЬНА ЗАПИСКА**  
кваліфікаційної роботи ступеня бакалавра

Студента Красіліч Вікторії Олексіївни

академічної групи 124-20-1

спеціальності 124 Системний аналіз

на тему: «Аналіз та автоматизація діяльності з використанням OLAP кубу в умовах харчового виробництва»

Керівники	Прізвище, ініціали	Оцінка за шкалою		Підпис
		рейтинговою	Інституційною	
кваліфікаційної роботи	<i>к.ф.-м.н., доц. Хом'як Т.В.</i>			
розділів:				
Інформаційно-аналітичний	<i>к.ф.-м.н., доц. Хом'як Т.В.</i>			
Спеціальний розділ	<i>к.ф.-м.н., доц. Хом'як Т.В.</i>			
Рецензент				
Нормоконтролер	<i>к.ф.-м.н., доц. Хом'як Т.В.</i>			

Дніпро  
2024

ЗАТВЕРДЖЕНО:

**завідувач кафедри**  
**Системного аналізу та управління**  
(повна назва)

\_\_\_\_\_ к.т.н., доц. Желдак Т.А.  
(підпис) (прізвище, ініціали)

« \_\_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_\_ року

**ЗАВДАННЯ**  
**на кваліфікаційну роботу**  
**ступеня бакалавра**

студенту Красіліч В.О. академічної групи 124- 20-1  
спеціальності: 124 Системний аналіз

на тему «Аналіз та автоматизація діяльності з використанням OLAP кубу  
в умовах харчового виробництва»

затверджену наказом ректора НТУ «Дніпровська політехніка»  
від 23.05.2024 р. №469-с

Розділ	Зміст	Терміни виконання
1. Інформаційно-аналітичний розділ	<i>Дослідити об'єкт аналізу. Встановити предметну область та визначити проблему, яка потребує розв'язання. Обговорити та обґрунтувати методи, які будуть використані для виконання поставлених завдань</i>	10.09.2023 – 01.03.2024
2. Спеціальний розділ	<i>Розв'язати поставлені задачі: здійснити моделювання бізнес-процесів у сфері харчового виробництва, розробити OLAP куб для харчової промисловості, візуалізувати аналіз отриманих даних та розробити прогноз на 2024 рік.</i>	01.03.2024 – 30.05.2024

Завдання видано \_\_\_\_\_ доц. Хом'як Т.В.  
(підпис) (прізвище, ініціали)

Дата видачі: 06.09.2023 р.

Дата подання до екзаменаційної комісії: \_\_\_\_\_

Прийнято до виконання \_\_\_\_\_ Красіліч В.О.  
(підпис студента) (прізвище, ініціали)

## РЕФЕРАТ

Пояснювальна записка: 103 с., 53 рис., 7 табл., 4 додатків, 19 джерел.

*Об'єктом дослідження* в роботі є малий бізнес з виробництва та продажу продукції швидкого харчування у місті Кривий Ріг.

*Предметом дослідження* є процеси аналізу та автоматизації діяльності малого харчового бізнесу з використанням OLAP кубу для покращення рентабельності та ефективності бізнесу.

*Метою* даної кваліфікаційної роботи є аналіз та автоматизація діяльності малого харчового бізнесу з використанням OLAP кубу для покращення рентабельності та ефективності бізнесу.

*Методи* дослідження: моделювання бізнес-процесів за допомогою AllFusion Process Model, створення програмного забезпечення підтримки прийняття рішення за допомогою технології OLAP, створення мір за допомогою спеціалізованої мови програмування DAX, створення візуалізації даних за допомогою Power BI, прогнозування за допомогою моделі SARIMA та методом експоненціального згладжування, принципи об'єктно-орієнтованого програмування на мові Python.

*В інформаційно-аналітичному розділі* наведено аналіз об'єкту дослідження, проведено дослідження літератури та всіх необхідних методів та технологій для реалізації мети дослідження

*У спеціальному розділі* побудовано модель бізнес-процесів за допомогою AllFusion Process Model, побудовано модель даних OLAP, написано міри на мові DAX, створено звіти за допомогою Power BI, побудовано прогноз на 2024 рік.

Практична цінність отриманих результатів полягає в тому, що використання OLAP кубів для аналізу даних у харчовому виробництві

дозволяє керівникам та аналітикам більш ефективно управляти виробничими процесами. Агрегування даних у багатовимірних структурах забезпечує швидкий доступ до інформації для прийняття рішень, що може сприяти зниженню витрат та оптимізації ресурсів.

*Ключові слова:* OLAP КУБИ, ХАРЧОВЕ ВИРОБНИЦТВО, БІЗНЕС-АНАЛІТИКА, АНАЛІЗ ДАНИХ, ВІЗУАЛІЗАЦІЯ ДАНИХ, ПРОГНОЗУВАННЯ ДАНИХ, POWER PIVOT, АВТОМАТИЗАЦІЯ ПРИЙНЯТТЯ РІШЕНЬ.

## ABSTRACT

Explanatory Note: 103 pages, 53 figures, 7 tables, 4 appendices, 19 references.

*Object of Study:* The small business involved in the production and sale of fast food products in the city of Kryvyi Rih.

*Subject of Study:* The processes of analysis and automation of the small food business operations using OLAP cubes to improve profitability and efficiency.

*Purpose of the Qualification Work:* To analyze and automate the activities of a small food business using OLAP cubes to enhance profitability and operational efficiency.

*Research Methods:* Business process modeling using AllFusion Process Model, developing decision support software using OLAP technology, creating measures using the specialized programming language DAX, data visualization using Power BI, forecasting using the SARIMA model and exponential smoothing method, object-oriented programming principles in Python.

*Informational-Analytical Section:* This section presents an analysis of the research object, reviews the literature, and examines all necessary methods and technologies to achieve the research goals.

*Special Section:* This section includes the construction of a business process model using AllFusion Process Model, development of an OLAP data model, writing measures in DAX, creating reports with Power BI, and developing a forecast for the year 2024.

*Practical Value:* The practical value of the results lies in the use of OLAP cubes for data analysis in food production, which allows managers and analysts to more effectively manage production processes. Aggregating data in multidimensional structures ensures quick access to information for decision-making, potentially leading to cost reduction and resource optimization.

*Keywords:* OLAP CUBES, FOOD PRODUCTION, BUSINESS ANALYTICS, DATA ANALYSIS, DATA VISUALIZATION, DATA FORECASTING, POWER PIVOT, DECISION AUTOMATION.

## ЗМІСТ

ВСТУП .....	9
РОЗДІЛ 1 ІНФОРМАТИЧНО-АНАЛІТИЧНИЙ .....	11
1.1 Основи бізнес-аналітики .....	11
1.2 Основні поняття AllFusion Process Model .....	12
1.3 Основні поняття про OLAP .....	13
1.3.1 Вступ до OLAP .....	13
1.3.2 Різниця між OLAP та транзакційними базами даних .....	14
1.3.3 Процес створення OLAP куба.....	14
1.3.4 Категорії зберігання OLAP кубів .....	18
1.3.5 Куби даних та OLAP операції.....	18
1.3.6 Вимоги до продуктів OLAP .....	20
1.3.7 Використання OLAP в умовах харчового виробництва .....	22
1.4 Мова DAX як інструмент написання мір в Power Pivot та Power BI .	22
1.4.1 Порівняння мови DAX та мови Excel .....	23
1.4.2 Типи даних і оператори в DAX .....	24
1.4.3 Функції в DAX.....	26
1.5 Візуалізація даних .....	28
1.5.1 Значення візуалізації даних.....	28
1.5.2 Принципи створення графіків.....	29
1.5.3 Панелі управління.....	30
1.5.4 Power BI як інструмент візуалізації даних .....	30
1.5.5 Типи візуалізацій даних та їх порівняння .....	32
1.6 Прогнозування даних .....	34

1.6.1	Прогнозування за допомогою моделі SARIMA.....	34
1.6.2	Прогнозування за допомогою методу експоненціального згладжування. ....	35
1.7	Висновки до розділу 1.....	37
РОЗДІЛ 2 СПЕЦІАЛЬНИЙ.....		38
2.1	AllFusion Process Model.....	38
2.2	Нормалізація бази даних.....	60
2.3	Створення OLAP куба в Power Pivot.....	64
2.3.1	Внесення даних в модель.....	64
2.3.2	Побудова зв'язків між таблицями.....	66
2.3.3	Написання мір на мові DAX.....	68
2.4	Створення візуалізації в Power BI.....	77
2.5	Побудова прогнозу на 2024 рік.....	86
2.5.1	Прогнозування доходу за допомогою моделі SARIMA.....	86
2.5.2	Прогноз кількості проданої продукції методом експоненціального згладжування.....	93
1.6	Висновки до розділу 2.....	98
ВИСНОВКИ.....		100
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....		102
Додаток А. Відомість матеріалів кваліфікаційної роботи.....		104
Додаток Б. Відгук керівника кваліфікаційної роботи.....		105
Додаток Г. Програмний код.....		105



## ВСТУП

Щоб залишатися конкурентоспроможними в сучасному діловому світі, вам потрібні швидкі та якісні рішення. За останні кілька років в Україні відбулося багато подій, які суттєво вплинули на діловий клімат. Зокрема, пандемія коронавірусу у 2019 році та повномасштабне вторгнення в Україну у 2022 році стали серйозною проблемою, особливо для малого та середнього бізнесу. Багатьом компаніям довелося припинити свою діяльність через нездатність адаптуватися до нових умов.

Малий бізнес зазвичай приймає рішення на основі попереднього досвіду, що часто не підкріплюється системами підтримки прийняття рішень. А для того щоб бізнес залишався рентабельним керівникам та працівникам необхідно відслідковувати одночасно багато факторів. Люди обмежені в своїх можливостях обробляти та запам'ятовувати інформацію. Джордж А. Міллер в своєму дослідженні «The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information» від 1956 року стверджує, що середня людина здатна обробити приблизно сім одиниць інформації одночасно, плюс-мінус дві. Це означає, що людина може тримати в своїй короткочасній пам'яті від п'яти до дев'яти елементів одночасно [1].

Сучасні бізнес-процеси генерують великі обсяги даних, тому вони повинні бути організовані таким чином, щоб спеціалісти могли швидко отримувати інформацію, необхідну для прийняття рішень.

Дослідження, проведене консалтинговою агенцією Accenture, виявило, що приблизно 50% рішень у бізнесі приймаються на основі інтуїції через недостатню доступність, швидкість та зрозумілість отримання даних.

Це підкреслює важливість структурування, зберігання та представлення даних для підвищення прибутковості бізнесу. Важливість цієї теми зумовлена необхідністю забезпечення швидкого,

точного та обґрунтованого прийняття рішень в умовах мінливого ринкового середовища. Події останніх років в Україні, такі як пандемія COVID-19 та повномасштабне вторгнення, значно вплинули на малий бізнес, змусивши його адаптуватися до нових викликів. Системи підтримки прийняття рішень, зокрема OLAP-куби, можуть допомогти малому бізнесу залишатися рентабельним, оптимізуючи процеси та підвищуючи ефективність управління.

Об'єктом дослідження є малий бізнес з виробництва та продажу продукції швидкого харчування у місті Кривий Ріг.

Предметом дослідження є процеси аналізу та автоматизації діяльності малого харчового бізнесу з використанням OLAP кубу для покращення рентабельності та ефективності бізнесу.

Метою цієї дипломної роботи є аналіз та автоматизація діяльності малого харчового бізнесу з використанням OLAP кубу для покращення рентабельності та ефективності бізнесу. Це досягається шляхом:

1. Аналізу поточного стану бізнесу: Оцінка існуючих бізнес-процесів, ідентифікація слабких та сильних місць.
2. Формування OLAP кубу: Створення багатовимірної моделі даних, яка дозволить здійснювати швидкі та ефективні аналітичні запити.
3. Аналізу рентабельності бізнесу: Використання зібраних даних для оцінки фінансових показників та визначення областей для покращення.
4. Візуалізація отриманих даних.
5. Прогнозування даних на наступний рік.

## РОЗДІЛ 1 ІНФОРМАТИЧНО-АНАЛІТИЧНИЙ

### 1.1 Основи бізнес-аналітики

В 1971 році Горрі та Мортон зробили значний внесок в розвиток бізнес-аналітики. Вони опублікували роботу, яка стала поштовхом для подальшого розвитку цієї сфери. Це була концепція систем підтримки прийняття рішень (Decision Support Systems, DSS), яка повинна була допомогти керівникам і менеджерам приймати більш обґрунтовані рішення за допомогою комп'ютерних систем. З того часу було проведено багато вдосконалень і досліджень подібних систем. За цей час система встигла змінити свою назву з систем управлінських рішень на системи підтримки прийняття рішень, а згодом і на бізнес-аналітику.

Бізнес-аналітика складається з додатків, технологій, інструментів і процесів, які допомагають користувачам приймати кращі рішення шляхом збору, зберігання, обробки та аналізу даних. Факти, або іншими словами дані, є основою створення рішень. Дані можуть привести нас до інформації, яка, в свою чергу, може призвести до знань. Отже, ми можемо стверджувати, що знання будуються на фактах. Наступним кроком виступає інформація: дані стають інформацією, оскільки вони також містять релевантність і мету. Останнім кроком є знання. Знання - це концепція, яку набагато складніше зрозуміти, майже кожен «знає», що таке знання. Тому чіткого визначення для цього не існує. Описуючи знання, ми говоримо про комбінацію досвіду, цінностей, розуміння, контексту та інформації. Ми можемо побачити з цього опису, що знання знаходяться всередині людей і не можуть легко існувати поза людьми.

Зазвичай вважають, що системи бізнес-аналітики складаються з кількох компонентів. Основними та найбільш використовуваними є компоненти, які зберігають дані (сховища даних) і які відображають їх. Досконаліші системи також складаються з таких компонентів, як аналіз даних та візуалізація даних.

## 1.2 Основні поняття AllFusion Process Model

Як я вже зазначала раніше, мета бізнес-аналізу полягає в тому, щоб працівники інтелектуальної діяльності могли приймати кращі рішення шляхом збору, зберігання, обробки та аналізу даних. Створення або перевірка моделей процесів допомагає отримати уявлення про те, як працює бізнес, і може показати процеси, які потребують оптимізації.

AllFusion Process Model (APM) є потужним інструментом для моделювання бізнес-процесів, який дозволяє візуалізувати, аналізувати та покращувати процеси в організації.

Бізнес-процес – це сукупність взаємозв'язаних дій або задач, що виконуються для досягнення конкретного результату в рамках організації.

Моделювання процесів – це створення візуальних моделей бізнес-процесів для аналізу і покращення їх ефективності.

AllFusion Process Model володіє трьома основними функціями: IDEF0, IDEF3 та DFD.

IDEF0 (Моделювання бізнес-процесів) - це метод який дозволяє аналізувати бізнес, фокусуючись на повсякденних функціях та контролі, який підтримує ці функції.

IDEF3 (Моделювання потоків процесів) – це метод який описує і документує процеси шляхом захоплення інформації про потік процесів,

відносини між процесами та важливі об'єкти. IDEF3 діаграми використовуються для створення робочих процесів, допомагають у реінжинірингу бізнес-процесів.

DFD (Діаграми потоків даних) – Це такі моделі системи як мережа активностей, з'єднаних потоками даних. DFD діаграми використовуються для документування руху та обробки інформації в бізнесі чи організації.

Які є переваги у використанні AllFusion Process Model в харчовому виробництві?

- Покращення управління бізнес-процесами: АРМ дозволить візуалізувати та аналізувати поточні процеси, що допоможе виявити вузькі місця та оптимізувати робочі потоки.
- Зниження витрат і ризиків: за допомогою детального моделювання процесів, АРМ допоможе знизити загальні витрати та ризики, пов'язані з адаптацією до змін операційної діяльності.

### **1.3 Основні поняття про OLAP**

#### **1.3.1 Вступ до OLAP**

OLAP (Online Analytical Processing) – це технологія, яка використовується для створення програмного забезпечення підтримки прийняття рішення. OLAP дозволяє користувачам швидко аналізувати інформацію, яка була узагальнена в багатовимірних уявленнях та ієрархіях. Ці дані зазвичай є критичними для прийняття обґрунтованих бізнес-рішень у різних галузях [1].

### 1.3.2 Різниця між OLAP та транзакційними базами даних

Технологія OLAP передбачає зберігання даних у вигляді, придатному для швидкого та ефективного вилучення великих за обсягом фрагментів. На відміну від інших технологій зберігання інформації, бази даних OLAP спочатку розраховану на максимально ефективне вилучення даних для побудови аналітичних звітів [2].

Є декілька відмінностей між сховищами OLAP та транзакційними базами даних, таких як Access та SQL:

1. Транзакційні бази мають в контексті часті додавання, вилучення та оновлення даних. Коли ж бази даних OLAP, навпаки, працюють тільки з «знімками» даних. В більшості випадків в базах OLAP знаходяться архівні дані, ціллю яких є їх аналіз за допомогою звітів. І навіть якщо данні постійно додаються, то старі данні залишаються на місці.
2. Спосіб зберігання інформації в базах OLAP та SQL відрізняються. Данні в кубі OLAP рідко зберігаються в «сирому» вигляді. Зазвичай їх до внесення в базу даних структурують та агрегують представлення, в яких вже визначенні правила групування, сортування та сумування. Це суттєво відрізняється від підходу в транзакційних базах даних. Там вам доведеться групувати, сортувати та агрегувати записи в стрічці.

### 1.3.3 Процес створення OLAP куба

Щоб створити модуль, спочатку необхідно зібрати необхідні дані. Оскільки оптимальний дизайн залежить від інструменту управління

справами, заданих питань бізнес-аналітики та бізнес-аналітики, а також даних, що містяться в інструменті управління справами, отже, це вхідні дані, необхідні для наступного кроку.

Першим кроком є визначення того, що таке інструмент управління справами, як зберігаються дані та як вони пов'язані. Відносини між різними об'єктами визначають, які дані ви можете декомунізувати і що ці дані означають.

Другий крок-визначити проблему, яку повинен вирішити інструмент. Це визначає, як ви хочете згрупувати дані пізніше. Вимірювання та ієрархії також впливають на ці проблеми.

Третій крок-перегляд доступних даних. Якщо неможливо відповісти на деякі запитання на попередньому кроці, можливо, вам доведеться змінити інструмент управління справами, щоб зберегти дані, необхідні для аналізу. Інші проблеми, які можуть виникнути з даними, полягають у декомунізації деякої інформації (постійно) або в тому, що очікувані відносини між об'єктами даних відрізняються від того, як вони виглядають.

Четвертий крок полягає в розробці куба OLAP. Куб OLAP складається з вимірювань, ієрархій, заходів декомунізації, агрегатів та взаємозв'язків між вимірами та заходами.

Але перед тим як почати створення OLAP куба, треба обрати для себе інструмент для його створення. Зараз існує багато інструментів в яких це можливо зробити, ось самі популярні з них:

- 1) Microsoft SQL Server Analysis Services (SSAS): інструмент для створення баговимірних структур та моделей даних від Microsoft. Є частиною Microsoft SQL Server. Має велику популярність в бізнес-аналітиці.

Переваги:

- Інтегрування таких інструментів, як Excel, Power BI та SharePoint.
- Оптимізація, яка допомагає працювати з великими обсягами даних і складними аналітичними задачами.
- Інтуїтивно зрозумілий інтерфейс.
- Високий рівень безпеки.

Недоліки:

- Висока вартість ліцензії
- Необхідність спеціальних знань для налаштування.
- Обмежена інтеграція з продуктами інших виробників.

2) Oracle OLAP: інструмент, який дозволяє створювати багатовимірні структури даних. Є частиною Oracle Database

Переваги:

- Гарно підходить для великих середовищ.
- Висока потужність.
- Інтегрування таких інструментів, як Oracle Database і Oracle BI.

Недоліки:

- Висока вартість ліцензії.
- Необхідність високого рівня технічних знань для налаштування.
- Обмежена підтримка з інструментами інших виробників.

3) IBM Cognos Analytics: платформа для бізнес-аналітики. Має повний набір інструментів для створення OLAP кубів, звітів та дашбордів.

Переваги:

- Має високий спектр для аналітичних можливостей.
- Інтегрування з іншими інструментами IBM.
- Підтримує різні джерела і види аналітики.

Недоліки:



- Висока вартість.
- Вимагає великих зусиль для налаштування.
- Необхідність спеціальних знань для використання.

4) Microsoft Excel: інструмент для аналізу даних, який підтримує створення OLAP кубів з використанням надбудови Power Pivot.

Переваги:

- Інтуїтивний інтерфейс.
- Має можливість підключення до різних джерел даних, таких як SQL Server, Oracle, Excel, веб-сервіси тощо.
- Має підтримку складних обчислень, формул, зведених таблиць, діаграм.
- За допомогою надбудови Power Pivot можна створювати моделі даних і OLAP куби.
- Доступність. Майже кожен користувався цим інструментом.

Недоліки:

- Може не справлятися з дуже великими об'ємами даних.
- При обробці великих обсягів даних, може працювати повільно.
- Неможливо працювати декільком працівникам з одним файлом в режимі реального часу.
- Захист даних в цьому інструменті менш надійний.

Для себе я обрала Microsoft Excel, як інструмент для створення OLAP куба.

По-перше, я вже мала досвід в створенні OLAP куба саме на цьому інструменті.

По-друге, в нашому випадку, так як ми досліджуємо малий бізнес, то потужності даного інструменту буде достатньо.

### **1.3.4 Категорії зберігання OLAP кубів**

Для реалізації OLAP баз даних існує три основні підходи до зберігання даних: у реляційних базах даних (ROLAP), багатовимірних базах даних (MOLAP) та гібридних рішеннях зберігання (HOLAP).

ROLAP (Relational OLAP): Використовує реляційні бази даних для зберігання даних і забезпечує гнучкість у роботі з великими обсягами розріджених даних. Реляційні бази даних часто нормалізовані, що зменшує надмірність даних та підвищує ефективність їх обробки.

MOLAP (Multidimensional OLAP): Використовує багатовимірні бази даних для зберігання даних у вигляді багатовимірних кубів. Це дозволяє швидко виконувати складні запити, але може вимагати більше місця для зберігання, особливо якщо дані розріджені.

HOLAP (Hybrid OLAP): Комбінує підходи ROLAP та MOLAP, забезпечуючи гнучкість і ефективність. HOLAP використовує реляційні бази даних для зберігання великих обсягів розріджених даних і багатовимірні бази даних для забезпечення швидкого доступу до агрегованих даних.

### **1.3.5 Куби даних та OLAP операції**

Куби даних є однією з ключових концепцій у OLAP, оскільки вони забезпечують інтуїтивно зрозумілий спосіб навігації різними рівнями зведеної інформації в базі даних.

У кубі даних атрибути категоризуються у вимірювальні атрибути (dimensions) та міри (measures). Вимірювальні атрибути визначають, за якими категоріями буде виконане групування даних. Наприклад, у харчовому виробництві вимірювальними атрибутами можуть бути "продукт", "регіон", "час" тощо. Міри є результатами агрегатних функцій, таких як сума, середнє значення, мінімум або максимум, і показують, як ці дані змінюються за різними вимірюваннями. Наприклад, міри можуть включати "кількість проданих одиниць", "загальний дохід" тощо.

Ключовими елементами куба OLAP є виміри, ієрархії, рівні, елементи та міри.

- Виміри (dimensions) – основні класифікатори даних. Вони містять виміри для аналізу.
- Ієрархії (hierarchies) – встановлені агрегації за рівнями в рамках виміру. Цей елемент дозволяє розгортати та аналізувати декілька рівнів одночасно.
- Рівні (levels) – це категорії даних, що агрегуються в рамках ієрархії.
- Елементи (members) – є конкретними елементами даних у межах виміру.
- Міри (measures) – значення даних всередині куба [2].

Куб даних з  $n$  вимірювальними атрибутами може мати до  $2^n$  різних подань або представлень.

Виміри часто організовані в ієрархії вимірів, які можуть бути представлені решіткою. Ієрархії дозволяють групувати дані на різних рівнях деталізації. Наприклад, ієрархія для виміру "час" може включати рівні "рік",

"квартал", "місяць", "день". У цьому випадку можна агрегувати дані за будь-яким з цих рівнів, що дозволяє проводити аналіз на різних рівнях деталізації.

OLAP включає набір операцій для маніпулювання вимірювальними даними, організованими на кількох рівнях абстракції. Основні OLAP операції включають:

**Roll-up:** Підвищення рівня агрегування. Це означає агрегування даних на вищому рівні ієрархії. Наприклад, дані про продажі за днями можуть бути агреговані до рівня місяців.

**Drill-down:** Зниження рівня агрегування. Це зворотна операція до roll-up, яка дозволяє перейти від більш загальних даних до більш деталізованих. Наприклад, від рівня місяців до рівня днів.

**Slice-and-dice:** Вибірка та проекція. Операція slice дозволяє вибрати підмножину куба даних, виділяючи певний зріз (slice) по одному виміру. Операція dice дозволяє створити підмножину даних, вибравши кілька вимірів і обмеживши діапазони їх значень.

**Pivot:** Переорієнтація багатовимірного подання даних. Операція pivot змінює орієнтацію куба даних, наприклад, замінюючи рядки та стовпці місцями. Це дозволяє краще візуалізувати і порівнювати різні аспекти даних.

OLAP операції дозволяють користувачам здійснювати гнучкий аналіз даних, отримувати різні зрізи інформації, порівнювати результати та знаходити закономірності, що сприяє прийняттю обґрунтованих бізнес-рішень.

### **1.3.6 Вимоги до продуктів OLAP**

Едгар Ф. Кодд і його колеги запропонували 12 вимог, за якими можна оцінювати продукти OLAP (Online Analytical Processing). Ці вимоги

допомагають визначити, чи відповідає продукт OLAP стандартам, необхідним для ефективного аналізу даних. Ось ці 12 вимог:

- **Мультимірна концептуальна модель:** Дані повинні бути представлені у багатовимірному вигляді, що дозволяє користувачам аналізувати їх за різними вимірами, такими як час, продукт, регіон тощо.
- **Прозорість:** OLAP має бути інтегрованим у звичні для користувача інструменти (наприклад, електронні таблиці) так, щоб користувач не помічав джерела даних.
- **Доступність:** Інструмент OLAP повинен мати можливість доступу до різномірних джерел даних і перетворювати їх у зручний для користувача вигляд.
- **Стабільна продуктивність звітності:** Продуктивність OLAP не повинна значно знижуватися при збільшенні кількості вимірів.
- **Клієнт-серверна архітектура:** Серверний компонент OLAP повинен бути досить інтелектуальним, щоб підтримувати різні клієнтські інтерфейси з мінімальними зусиллями.
- **Універсальна вимірність:** Всі виміри даних повинні бути рівнозначні за своєю структурою та функціональними можливостями.
- **Динамічна обробка розріджених матриць:** Структура OLAP повинна оптимально обробляти розріджені матриці.
- **Підтримка багатокористувацького режиму:** OLAP інструмент повинен забезпечувати одночасний доступ до даних для кількох користувачів, зберігаючи їх цілісність та безпеку.
- **Необмежені крос-вимірні операції:** OLAP інструмент повинен дозволяти обчислення та маніпуляції даними між будь-якими вимірами без обмежень.
- **Інтуїтивне маніпулювання даними:** Маніпулювання даними, такими як деталізація чи зведення, повинно здійснюватися безпосередньо на клітинках аналітичної моделі.

- Гнучка звітність: Інструмент OLAP повинен надавати можливість створення звітів у будь-якому вигляді, зручному для користувача.
- Необмежені виміри та рівні агрегації: Кількість підтримуваних вимірів та рівнів агрегації повинна бути практично необмеженою, забезпечуючи можливість визначення користувацьких рівнів агрегації у будь-якому шляху консолідації [3].

### **1.3.7 Використання OLAP в умовах харчового виробництва**

В умовах харчового виробництва OLAP куби можуть бути використані для аналізу різних аспектів діяльності підприємства, таких як продажі, запаси, виробництво та інші ключові показники ефективності. Використання багатовимірного аналізу дозволяє отримувати детальну інформацію про кожен аспект діяльності, що сприяє прийняттю обґрунтованих рішень.

Таким чином, OLAP є потужним інструментом для аналізу і автоматизації діяльності підприємств, забезпечуючи можливість детального і багатовимірного аналізу даних, що сприяє покращенню ефективності і прийняттю обґрунтованих бізнес-рішень.

### **1.4 Мова DAX як інструмент написання мір в Power Pivot та Power BI**

Міри є ключовою складовою OLAP кубів. Вони представляють собою числові дані, які можна агрегувати та аналізувати. Міри в Power Pivot пишуться на мові DAX.

DAX (Data Analysis eXpressions) – це спеціалізовано розроблена мова програмування для обчислення бізнес-формул на основі моделі даних і є

мовою програмування Microsoft Power BI, Microsoft Analysis Services і Microsoft Power Pivot for Excel. Ця мова програмування була створена в 2010 році з першим випуском PowerPivot для Microsoft Excel 2010. З 2013 року DAX набув популярності в спільноті користувачів, які використовували Excel та мову DAX для побудови моделей в Power Pivot та користувачів які використовували цю мові для побудови моделей в Power BI. DAX присутній у багатьох різних інструментах, усі вони мають один і той самий внутрішній рушій під назвою Tabular [4].

#### **1.4.1 Порівняння мови DAX та мови Excel**

Думаю кожен в якійсь мірі знайом з мовою формул в Excel. DAX чимось нагадує цю мову, що і не дивно, адже коріннями DAX знаходиться в Power Pivot для Excel і розробники намагались залишити ці дві мови схожими один на одну. Але все ж відмінності є.

Однією з таких відмінностей є те, що Excel виконує обчислення через комірки таблиць, посилаючись на них за допомогою їх координат. В DAX не існує поняття комірки взагалі. DAX працює з таблицями та стовпцями, посилаючись на назву стовпця в певній таблиці. Через те, що DAX працює з моделлю даних ми завжди повинні прописувати і назву таблиці, і назву стовпця, так як в різних таблицях стовпці можуть називатися однаково.

Багато функцій з Excel залишилися не змінними і для DAX та працюють вони за однаковою логікою. Прикладом такої функції може бути функція IF.

Одним з аспектів, де ці дві мови схожі, є те, що і Excel, і DAX є функціональними мовами. Функціональна мова складається з виразів, які в основному є викликами функцій. В Excel і DAX поняття операторів, циклів і переходів не існують, хоча вони є загальними для багатьох мов програмування. У DAX все є виразом. Цей аспект мови часто є проблемою

для програмістів, які працюють з різних мов, але це зовсім не повинно бути сюрпризом для користувачів Excel [4].

Нова концепція, якої не має в Excel, але є в DAX, це концепція ітераторів. Працюючи в Excel, ми виконували розрахунки крок за кроком. Наприклад, щоб порахувати суму продажів, нам необхідно створити окремий стовпець, де помножити кількість на ціну, а потім, як наступний крок, підсумувати цей стовпець, щоб отримати загальну суму продажів. В DAX же ми можемо виконати таку саму операцію за один крок, за допомогою ітераторів. Ітератор перебирає таблицю та виконує обчислення для кожного рядка таблиці, агрегуючи результат. На виході ми отримуємо єдине запитане значення.

#### 1.4.2 Типи даних і оператори в DAX

DAX може виконувати обчислення з різними числовими типами, яких сім. В таблиці 1.1 показані 7 типів даних, та різні їх назви.

Таблиця 1.1

##### Типи даних та їх назви в різних середовищах

Тип даних в DAX	Тип даних в Power BI	Тип даних в Power Pivot та Analysis Services	Тип даних в SQL Server
Integer	Whole Number	Whole Number	Integer / INT
Decimal	Decimal Number	Decimal Number	Floating point / DOUBLE
Currency	Fixed Decimal Number	Currency	Currency / MONEY
DateTime	DateTime, Date, Time	Date	Date / DATETIME
Boolean	True/False	True/False	Boolean / BIT
String	Text	Text	String / NVARCHAR(MAX)
Binary	Binary	Binary	Blob / VARBINARY(MAX)



Щоб писати міри нам необхідно не тільки знати, як називаються типи даних в цьому середовищі, особливо якщо враховувати, що DAX підбирає необхідний тип даних автоматично, а й знати оператори доступні в DAX. Таблиця 1.2 дає уявлення про доступні оператори та способи їх використання.

Таблиця 1.2

### Оператори в DAX

Тип оператора	Символ	Використання	Приклад
Дужки	()	Порядок пріоритету та групування аргументів	(5+1)*2
Арифметика	+ - * /	Додаток Віднімання Множення Ділення	4+1 4-1 4*1 4/1
Порівняння	= <> > >= < <=	Дорівнює Не дорівнює Більше Більше або дорівнює Менше Менше або дорівнює	CountryRegion] = "USA" [CountryRegion] <> "USA" [Quantity] > 0 [Quantity] >= 100 [Quantity] < 0 [Quantity] <= 100
Об'єднання тексту	&	Об'єднання рядків	"Value is" & [Amount]
Логічні	&&      IN  NOT	Умова AND між двома булевими виразами  АБО умова між двома булевими виразами  Включення елемента в список Логічне заперечення	[CountryRegion] = "USA" && [Quantity]>0  [CountryRegion] = "USA"    [Quantity] > 0  [CountryRegion] IN {"USA", "Canada"}

			NOT [Quantity] >0
--	--	--	-------------------

### 1.4.3 Функції в DAX

Самих функцій DAX дуже багато, щоб описувати їх всі, тому я дам опис тих функцій, які використовувала при написанні цієї роботи.

Функція SUMX повертає суму виразу, який обчислюється для кожного рядка в таблиці [5].

SUMX(<table>, <expression>) 1.1

де <table>— назва таблиці, <expression> - вираз, який буде обчислюватися для кожної строки таблиці.

Функція ALL повертає всі строки в таблицях або всі значення в стовпці, ігноруючи всі фільтри, які могли бути застосовані [6].

ALL(<table>) 1.2

де <table>— назва таблиці.

Функція DIVIDE виконує ділення і повертає альтернативний результат або вираз BLANK() при діленні на 0 [7].

DIVIDE(<numerator>, <denominator> [, <alternateresult>]) 1.3

де <numerator>— ділиме число, <denominator> - дільник, <alternateresult>- значення, яке повертається при діленні на 0.

Функція CALCULATE обчислює вираз у зміненому контексті фільтра [8].

CALCULATE(<expression>[, <filter1> [, ...]]) 1.4

де <expression> - обчислювальний вираз, <filter1> - не обов'язковий логічний вираз або табличний вираз обзначаючий фільтр або функцію модифікатора фільтра.

Функція LOOKUPVALUE повертає значення строки, яка відповідає всім критеріям, заданими одним або декількома умовами пошуку [9].

```
LOOKUPVALUE (
    <result_columnName>,
    <search_columnName>,
    <search_value>
    [, <search2_columnName>, <search2_value>]...
    [, <alternateResult>]
)
```

де <result\_columnName> - назва стовпця, який містить повертаєм значення, <search\_columnName> - назва стовпця, <search\_value> – значення для пошука в <search\_columnName>, <alternateResult> - значення, яке повертається при фільтрації контекста до нуля.

Функція VALUES повертає таблицю з одним стовпцем, який містить різні значення з указанного стовпця, якщо вхідний параметр є назвою стовпця. Значення які повторюються видаляються й повертаються тільки унікальні [10].

```
VALUES(<TableNameOrColumnName>) 1.6
```

де <TableNameOrColumnName> - стовбець з якого повертаються унікальні значення, чи таблиця з якої повинні повертатися строки.

Функція AVERAGE повертає середнє значення всіх чисел в стовпці [11].

AVERAGE(<column>) 1.7

де <column> - стовпець, який містить числа.

Функція FILTER повертає таблицю, що представляє підмножину іншої таблиці або виразу [12].

FILTER(<table>,<filter>) 1.8

де <table> - таблиця для фільтрування, <filter> - Логічний вираз, який необхідно оцінити для кожної строки таблиці.

Функція SUM додає всі числа в стовпці [13].

SUM(<column>) 1.9

де <column> - стовпець, який містить числа для суми.

Функція MAX повертає найбільше значення в стовпці [14].

MAX(<column>) 1.10

де <column> - стовпець, в якому треба знайти максимальне значення.

## 1.5 Візуалізація даних

### 1.5.1 Значення візуалізації даних

Як я вже зазначала раніше люди обмежені щодо можливості обробки і запам'ятовування інформації. Техніки для підвищення здатності зберігати більше інформації включають використання багатовимірних вимірів. Це дозволяє представити інформацію в більш зручному і наочному вигляді, що сприяє кращому розумінню і запам'ятовуванню даних.

Іншим підходом до покращення обробки інформації є "інформаційне подрібнення". Наприклад, легше запам'ятати рядок бінарних чисел, якщо об'єднати їх у групи десяткових чисел. Також легше запам'ятовувати слова, ніж окремі букви. Це дозволяє ефективніше обробляти великі обсяги даних, оскільки подрібнена інформація легше сприймається і запам'ятовується[3].

Оскільки збирання інформації дає величезну кількість даних, для аналітиків важливо представляти їх таким чином, щоб необхідну інформацію можна було швидко знайти та зрозуміти. Візуалізація даних дозволяє використовувати візуальну систему людини для вилучення інформації та виявлення цікавих частин даних, таких як структури, тренди, аномалії, шляхом надання огляду даних.

Типові форми візуалізації даних включають таблиці, лінійні графіки, точкові діаграми, стовпчикові та кругові діаграми, радарні графіки, карти тощо. Кожен тип візуалізації має свої переваги для певних видів даних і може бути використаний для різних цілей аналізу. Про це ми поговоримо далі.

## **1.5.2 Принципи створення графіків**

Едвард Тафт запропонував правила для створення чітких і легко зрозумілих графіків:

- Показувати дані.
- Використовувати графіку лише тоді, коли вона додає інформацію.
- Уникати надмірного використання графічних елементів, які відволікають від даних.
- Використовувати мінімальну кількість чорнила для зображення даних.
- Використовувати підписи.
- Використовувати макро- та мікро-рівні деталізації (загальний огляд визначається деталями).

- Відокремлювати шари інформації.
- Використовувати множинні графіки для полегшення порівняння даних.
- Використовувати колір, але уникати кольорів, які не додають значення.
- Розуміти наратив (показувати час і простір відповідно до очікувань)[4].

### **1.5.3 Панелі управління**

Панелі управління (Dashboards) є важливими інструментами в бізнес-аналітиці, оскільки дозволяють користувачам отримати огляд найбільш важливих даних в одному місці. Панель управління може відображати як просторову інформацію (порівняння, сприйняття взаємозв'язків), так і символічну інформацію (таблиці з даними). Сучасні панелі управління можуть включати різні типи візуалізацій для ефективного відображення ключових показників продуктивності .

### **1.5.4 Power BI як інструмент візуалізації даних**

Одним із найпопулярніших інструментів для візуалізації даних є Power BI від Microsoft. Power BI є потужним інструментом, що дозволяє створювати інтерактивні звіти та панелі управління, які можуть об'єднувати дані з різних джерел і представляти їх у зрозумілому вигляді.

Power BI – це комплексне програмне забезпечення бізнес-аналітики від компанії Microsoft, об'єднуюче декілька програмних продуктів, які мають спільний технологічний і візуальний дизайн. Також Power BI є частиною єдиної платформи Microsoft Power Platform, туди також входять PowerApps та Microsoft Flow.

Роботу в Power BI можна розбити декілька базових блоків:

- Візуалізація – це візуальне представлення даних, наприклад карта, графік або діаграма.
- Набір даних – це колекції даних, на основі якої створюють візуалізації.
- Звіти – це колекція візуалізацій, які відображають разом на одній або декількох сторінках, для більш комплексного розуміння бізнес-процесів.
- Панелі моніторингу – це інформаційна панель з колекції візуальних елементів на одній сторінці, до якої можна надати спільний доступ.
- Плитки – це одна візуалізація в звіті або на інформаційній панелі.

#### Переваги використання Power BI:

**Інтерактивність:** Power BI дозволяє користувачам взаємодіяти з даними в режимі реального часу, що забезпечує більш глибокий аналіз.

**Інтеграція з іншими джерелами даних:** Power BI підтримує підключення до різноманітних джерел даних, таких як SQL Server, Excel, веб-сервіси та багато інших.

**Легкість використання:** Інтуїтивно зрозумілий інтерфейс дозволяє користувачам швидко створювати звіти без потреби в глибоких технічних знаннях.

**Можливості для спільної роботи:** Power BI забезпечує можливість спільного доступу до звітів та панелей управління, що дозволяє командам працювати разом над аналізом даних.

**Мобільність:** Доступ до звітів та панелей управління можна отримати з будь-якого пристрою, що забезпечує гнучкість у роботі з даними.

Power BI буде використовуватись у моїй дипломній роботі для створення візуалізацій та панелей управління, що дозволить ефективно аналізувати дані та підтримувати прийняття рішень.

### 1.5.5 Типи візуалізацій даних та їх порівняння

Розглянемо деякі з найбільш поширених типів візуалізацій та їх застосування (Табл. 1.3).

Барові діаграми (стовпчикові діаграми) використовуються для порівняння різних категорій даних. Вони показують величини за допомогою стовпчиків різної висоти або довжини. Барові діаграми є особливо корисними для порівняння значень між різними групами, наприклад, продажів різних продуктів або результатів опитувань.

Кругові діаграми використовуються для відображення часток цілого. Вони показують пропорції кожної категорії у вигляді сегментів кола. Кругові діаграми найкраще використовувати, коли потрібно показати, як частини співвідносяться з цілим.

Точкові діаграми використовуються для візуалізації розподілу даних та виявлення кореляцій між двома змінними. Вони відображають кожен запис як точку на координатній площині. Це особливо корисно для аналізу великих наборів даних, де потрібно знайти тренди або взаємозв'язки.

Картографічні діаграми використовуються для географічного аналізу даних. Вони відображають інформацію на карті, що дозволяє побачити географічні тенденції та розподіл даних. Це особливо корисно для аналізу даних, пов'язаних з місцезнаходженням, таких як демографічні дані, продажі в різних регіонах або поширення захворювань.

Радарні графіки використовуються для порівняння декількох змінних. Вони відображають дані у вигляді променів, вихідних з однієї точки. Це дозволяє побачити сильні та слабкі сторони різних показників одночасно.

Комбіновані стилі використовують різні типи візуалізацій для комплексного аналізу даних. Наприклад, можна поєднувати лінійні графіки з баровими діаграмами для детального аналізу тенденцій та порівнянь.



Таблиця 1.3

## Порівняльний аналіз типів візуалізацій

Тип візуалізації	Опис	Переваги	Недоліки
-1-	-2-	-3-	-4-
Барові діаграми	Використовуються для порівняння різних категорій.	Легко зрозуміти та порівнювати категорії. Можливість горизонтальних/вертикальних стовпчиків.	Обмежена кількість категорій. Може стати перевантаженою великою кількістю даних.

Продовження табл. 1.3

-1-	-2-	-3-	-4-
Кругові діаграми	Відображають частки цілого.	Легко показати пропорції. Інтуїтивно зрозумілі.	Не підходять для великої кількості категорій. Важко порівнювати сегменти
Точкові діаграми	Візуалізують розподіл даних та кореляції між двома змінними.	Виявлення кореляцій та трендів. Підходять для великих наборів даних	Важко інтерпретувати без додаткового аналізу.
Картографічні діаграми	Використовуються для географічного аналізу даних.	Візуалізація на географічній основі. Легке визначення регіональних тенденцій.	Залежні від точності географічних даних
Радарні графіки	Порівняння декількох змінних.	Виявлення сильних і слабких сторін. Візуально привабливі.	Важко читати при великій кількості змінних.
Комбіновані стилі	Поєднують різні типи візуалізацій для комплексного аналізу	Гнучкість та багатогранний погляд на дані.	Можуть бути складними для інтерпретації.
Лінійні графіки	Відображають зміни даних з часом.	Легко показати тенденції та зміни. Зрозумілий та простий.	Менш ефективний для великих наборів категорій.
Таблиці	Організоване представлення	Зручні для детального представлення даних.	Можуть бути неефективними

	даних у рядках та стовпцях.	Простота у читанні та порівнянні.	для великих наборів даних.
--	-----------------------------	-----------------------------------	----------------------------

## 1.6 Прогнозування даних

Прогнозування – це передбачення майбутнього з використанням історичних даних і знань про майбутні події, які можуть вплинути на наші прогнози.

### 1.6.1 Прогнозування за допомогою моделі SARIMA

Сезонне авторегресійне інтегроване ковзне середнє, SARIMA або Seasonal ARIMA, є розширенням ARIMA, яке явно підтримує одновимірні дані часових рядів із сезонним компонентом.

Він додає три нові гіперпараметри для визначення авторегресії (AR), диференціації (I) і ковзного середнього (MA) для сезонної складової ряду, а також додатковий параметр для періоду сезонності.

Авторегресійне інтегроване ковзне середнє, або ARIMA, є методом прогнозування для одновимірних даних часових рядів.

Як випливає з назви, він підтримує елементи авторегресії та ковзного середнього. Інтегрований елемент відноситься до розрізнення, що дозволяє методу підтримувати дані часових рядів із тенденцією.

Проблема з ARIMA полягає в тому, що він не підтримує сезонні дані. Це часовий ряд із повторюваним циклом.

ARIMA очікує даних, які або не є сезонними, або містять сезонний компонент, наприклад, сезонно скориговані за допомогою таких методів, як сезонна різниця.

Налаштування SARIMA вимагає вибору гіперпараметрів як для тенденції, так і для сезонних елементів ряду [15].

$$\text{SARIMA}(p,d,q)(P,D,Q)m$$

де  $p$  - порядок авторегресії тренду,  $d$  - порядок різниці тенденцій,  $q$  - порядок ковзного середнього тренду,  $P$  - Сезонний порядок авторегресії,  $D$  - Порядок сезонної різниці,  $Q$  - Порядок сезонного ковзного середнього,  $m$  - кількість часових кроків для одного сезонного періоду.

### **1.6.2 Прогнозування за допомогою методу експоненціального згладжування.**

Приблизно в 1950-х роках з'явився клас методів прогнозування на основу експоненціального згладжування. Вони відносяться до набору методів, які в універсальний спосіб можуть бути використані для моделювання та отримання прогнозів.

Експоненціальне згладжування - це простий і прагматичний підхід до прогнозування, при якому прогноз будується на основі експоненціально зваженого середнього минулого спостереження. Найбільша вага надається теперішньому спостереженню, менша вага безпосередньо попередньому спостереженню, ще менша вага спостереженню до цього і так далі.

Ідея експоненціального згладжування полягає в тому, щоб згладити вихідний ряд так, як це робить ковзне середнє, і використовувати згладжений ряд для прогнозування майбутніх значень змінної, що цікавить [16].

Найвідоміші методи на основі експоненціального згладжування:

1. Просте експоненціальне згладжування використовується, коли часовий ряд не має явних трендів чи сезонних компонентів.

$$S_t = \alpha X_t + (1 - \alpha)S_{t-1}$$

де  $S_t$  – прогнозоване значення на момент часу  $t$ ,  $X_t$  – фактичне значення на момент часу  $t$ ,  $\alpha$  – коефіцієнт згладжування,  $S_{t-1}$  – прогнозоване значення на момент часу  $t-1$ .

2. Лінійний метод Холта — просте експоненціальне згладжування добре працює, коли немає тренду, сезонності чи інших закономірностей. Холт в 1957 році розширив його до лінійного експоненціального згладжування, що дозволяє прогнозувати дані з трендом.

$$\begin{aligned} \hat{X}_{t+m} &= (\alpha X_t + (1 - \alpha)(L_{t-1} + T_{t-1})) + m(\beta(L_t - L_{t-1}) \\ &+ (1 - \beta)T_{t-1}) \end{aligned}$$

де  $\hat{X}_{t+m}$  – прогнозоване значення на  $m$  періодів вперед,  $\alpha$  – коефіцієнт згладжування для рівня,  $\beta$  – коефіцієнт згладжування для тренду,  $X_t$  – фактичне значення на момент часу  $t$ ,  $T_t$  – оцінка тренду (зміни рівня) у момент часу  $t$ ,  $L_t$  – оцінка рівня (або вирівнюваного середнього) у момент часу  $t$ .

3. Метод трендів і сезонності Холта-Вінтерса — попередні методи не підходять, якщо дані демонструють сезонні закономірності. Холт в 1957 році запропонував метод сезонних даних, пізніше в 1960 році його вдосконалив Вінтерс [17].

$$\begin{aligned} \hat{X}_{t+m} &= (\alpha(X_t - S_{t-s}) + (1 - \alpha)(L_{t-1} + T_{t-1})) \\ &+ m(\beta(L_t - L_{t-1}) + (1 - \beta)T_{t-1}) + S_{t-s+m \text{ mod } s} \end{aligned}$$

де  $\hat{X}_{t+m}$  – прогнозоване значення на  $m$  періодів вперед,  $L_t$  – оцінка рівня у момент часу  $t$ ,  $T_t$  – оцінка тренду у момент часу  $t$ ,  $S_t$  – оцінка сезонної компоненти у момент часу  $t$ ,  $X_t$  – фактичне значення на момент часу  $t$ ,  $s$  – довжина сезонного циклу,  $\alpha$ ,  $\beta$ ,  $\gamma$  –

коефіцієнти згладжування для рівня, тренду та сезонності відповідно.

## 1.7 Висновки до розділу 1

В інформатично-аналітичному розділі ми розглянули основні положення про бізнес-аналітику. Розібрали необхідність моделювання бізнес-процесів використовуючи AllFusion Process Model, а також створення OLAP кубу для малого харчового виробництва.

Також ми розглянули вимоги які допоможуть перевірити , чи відповідає наш майбутній продукт OLAP стандартам, необхідним для ефективного аналізу даних.

Розібрали необхідність візуалізації даних за допомогою інтерактивних звітів, таких які можна створити в Power BI. Також визначили правила для створення чітких і легко зрозумілих візуалізацій, які будемо використовувати в практичній частині.

Аналіз літературних джерел дав нам інформаційну базу, яка необхідна для подальшого виконання роботи, також ми знайшли підтвердження, що використання та оптимізація OLAP систем для підтримання продуктивності бізнес-аналітики є актуальним питанням.

Отриманні знання по створенню та оптимізації OLAP куба лягли в основу створення власної моделі для аналізу даних малого харчового виробництва в місті Кривий ріг. Створення та оптимізація бізнес процесів за допомогою AllFusion Process Model, та аналіз створеного OLAP куба, як і представлення результатів за допомогою Power BI будуть описанні у практичній частині цієї роботи.

## **РОЗДІЛ 2 СПЕЦІАЛЬНИЙ**

### **2.1 AllFusion Process Model**

Першим кроком оцінимо бізнес-процеси харчового підприємства.

Для якісної оцінки бізнес-процесів потрібно добре знати підприємство та хід його роботи. В дипломній роботі я провожу дослідження для бізнесу моєї сім'ї. Це невелика будівля з найманими працівниками, яка виготовлює та продає харчові продукти.

На підставі зібраних даних, а саме: проведення опитування серед працівників і власників та досвід роботи на цьому підприємстві, визначаємо ключові фактори успіху виходячи з даних про функціонування підприємства.

Записуємо ключові фактори та бізнес процеси в таблицю 2.1 та оцінюємо за шкалою від 1 до 5 (1 – найгірше організовані, 5 – найліпше організовані).

Таблиця 2.1

### Оцінка важливості бізнес-процесів харчового виробництва

Бізнес-процес (БП)	Ключові фактори успіху	Важливість бізнес-процесу
-1-	-2-	-3-
БП1 Забезпечення якісними продуктами (закупівля)	Забезпечення персоналу продуктами високої якості для виготовлення якісної продукції	5
БП2 Забезпечення справності обладнання	Контроль, ремонтні роботи та вчасна заміна технічного обладнання	3

Продовження табл. 2.1

-1-	-2-	-3-
БП3 Моніторинг попиту на товар	Моніторинг попиту кожного товару для вчасної заміни або корективу технології виготовлення	2
БП4 Контроль якості товару	Контроль терміну придатності та якості виготовленого товару	4
БП5 Управління охороною праці	Забезпечення працівників необхідними знаннями та обладнанням для підтримування безпеки під час роботи	3
БП6 Контроль рентабельності	Моніторинг за підвищенням цін на витратні матеріали та оренду і	4

	вчасне підвищення цін на продукцію	
БП7 Забезпечення охорони приміщення	Забезпечення працівників безпечним місцем роботи та охороною майна в неробочий час	2
БП8 Управління фінансовими ресурсами	Забезпечення вчасної виплати зарплатні працівникам та оплати оренди та комунальних послуг	3

Таблиця 2.2 відображає рівень організації бізнес-процесів.

Таблиця 2.2

### Оцінка рівня організації бізнес-процесів

Бізнес-процес (БП)	Рівень організації БП
БП1 Забезпечення якісними продуктами (закупівля)	5
БП2 Забезпечення справності обладнання	2
БП3 Моніторинг попиту на товар	2
БП4 Контроль якості товару	4
БП5 Управління охороною праці	3
БП6 Контроль рентабельності	4
БП7 Забезпечення охорони приміщення	2



БП8 Управління фінансовими ресурсами	3
БП9 Забезпечення працівників технологічними картами та процесами	2

Діаграма на рисунку 2.1 відображає «профіль розвитку» підприємства на основі оцінки рівня організації бізнес - процесів за шкалою від 1 до 5 балів (1 – найгірше організовані, 5 – найліпше організовані).

Важливість	5					БП1
	4				БП4/БП6	
	3		БП2	БП5/БП8		
	2		БП3/БП7/БП9			
	1					
		1	2	3	4	5
Рівень організації						

Рисунок 2.1 – Діаграма профілю розвитку

З цієї діаграми видно, що бізнес-процес «Забезпечення справності обладнання» потрібно переглянути в першу чергу, та ввести в нього зміни. Бізнес-процеси «Моніторинг попиту на товар» «Контроль якості товару» «Управління фінансовими ресурсами», «Управління охороною праці» та «Забезпечення працівників технологічними картами та процесами» не потребують реінжинірингу.

Побудуємо діаграми потоків робіт бізнес-процесів в нотації IDEF3, що відображають різні сценарії протікання бізнес-процесів (рис 2.2).

Для контекстного процесу «Харчове виробництво» визначимо необхідну інформацію:

- ВХІД – фінансове постачання, запит на проведення аналізу рентабельності;
- УПРАВЛІННЯ - законодавство України, внутрішні документи організації;
- МЕХАНІЗМИ - працівники, технічне обладнання;
- ВИХІД – виконаний договір куплі-продажу, прибуток.

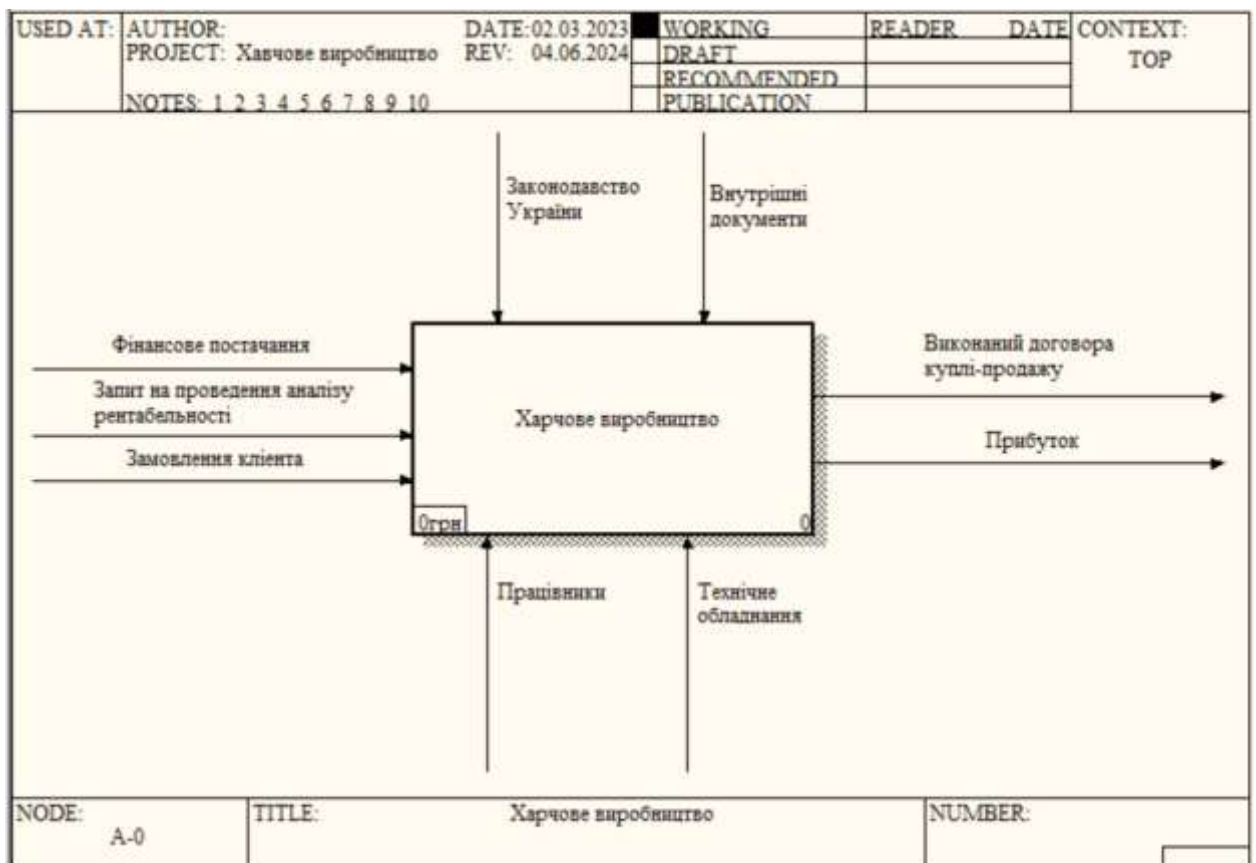


Рисунок 2.2 - Контекстна діаграма процесу "Харчове виробництво"

Маємо такі цільові функції:

- Маркетинговий аналіз;
- здійснення закупівлі;
- процес виготовлення продукції;
- здійснення продажу.

Та наступні нормативні документи:

- ЗУ «Захист прав споживачів»;
- ДСанПіН 2.2.4-171-10 "Гігієнічні вимоги до виробництва та обігу харчової продукції, харчових добавок та матеріалів, які контактують з харчовою продукцією".
- стандарти якості;
- технологічна карта;
- інструкції з техніки безпеки;
- закон України "Про безпеку та якість харчових продуктів".

Та наступні підрозділи підприємства:

- відділ логістики;
- відділ маркетингу;
- відділ виробництва;
- відділ продажу.

З цього випливає декомпозиція, яка представлена на рисунку 2.3. Це діаграма, яка описує бізнес-процеси харчового виробництва, включаючи в себе також документацію. Для бізнесу ця діаграма несе важливу інформацію, а саме: детально показує потоки операцій від закупівлі сировини до виробництва готової продукції.

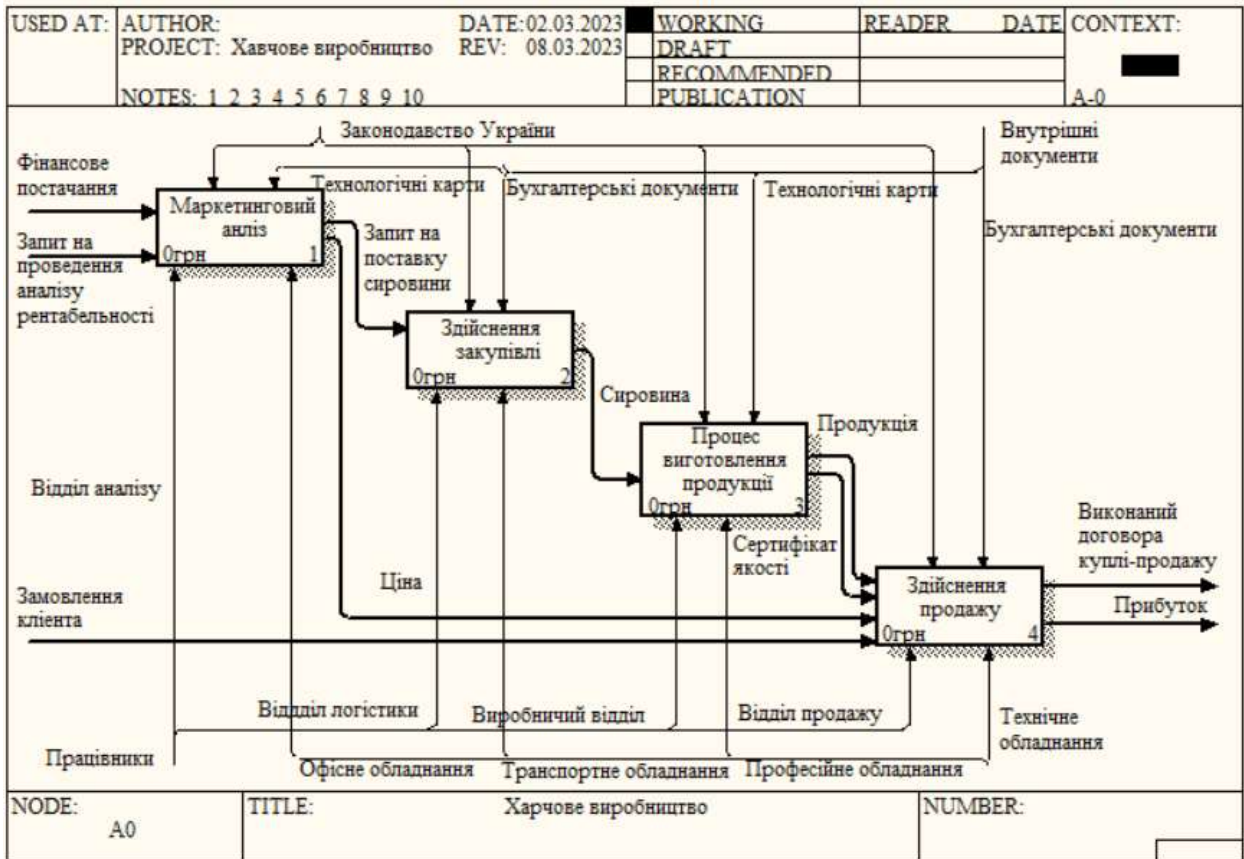


Рисунок 2.3 – Декомпозиційна діаграма для харчового виробництва

На наступній діаграмі (рис. 2.4) проводиться декомпозиція процесу «Маркетинговий аналіз». Іншими словами розбиває цей процес на декілька більш простих та зрозумілих. Ми можемо бачити, що на вході нам потрапляє фінансове постачання та запит на проведення аналізу рентабельності. Після проходження всіх етапів, а в даному процесі їх 4, на виході ми отримуємо запит на постачання сировини та ціну, яку розраховали в процесі дослідження.

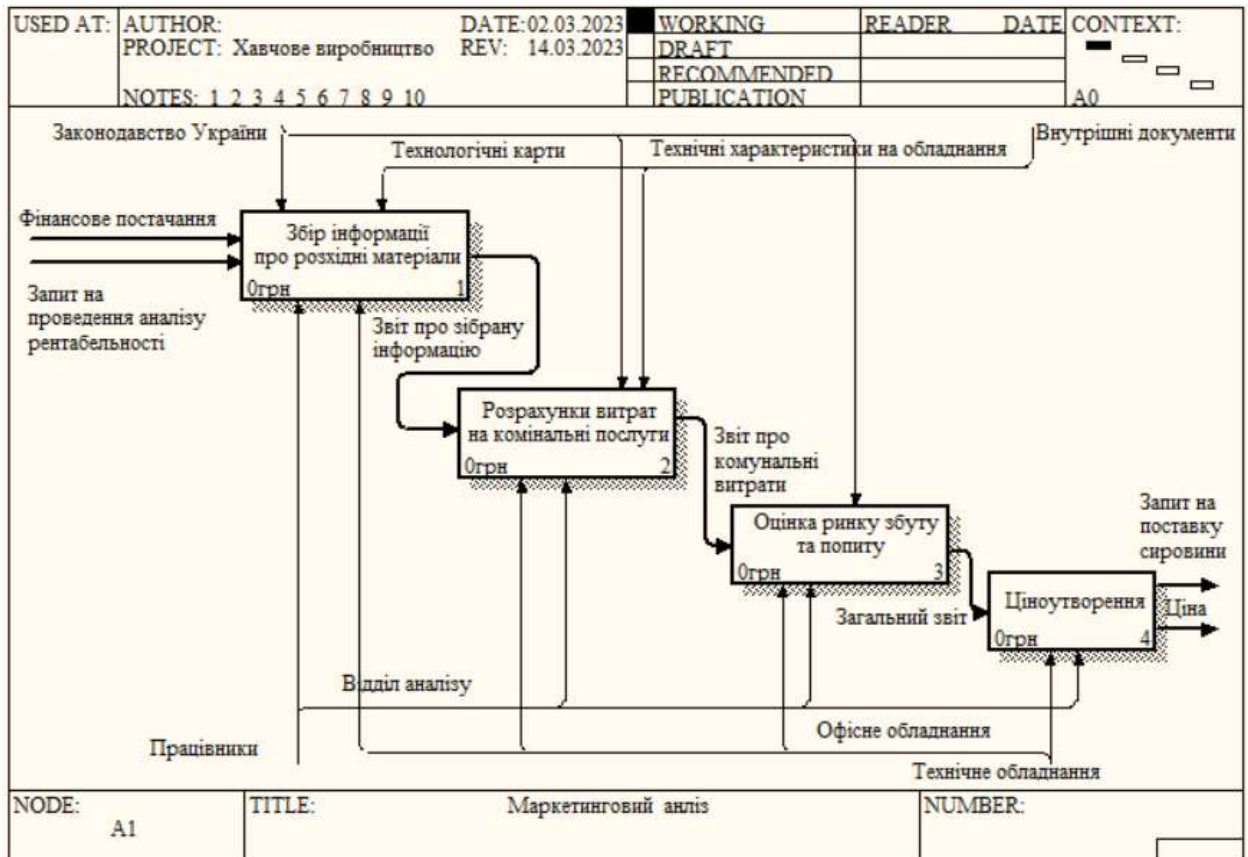


Рисунок 2.4 - Декомпозиційна діаграма для процесу «Маркетинговий аналіз»

На рисунку 2.5 ми продовжуємо розбивати процеси на більш дрібні. На цей раз ми декомпозуємо процес збору інформації. На цій діаграмі ми можемо побачити значок виключного «або», він означає, що тільки один наступний процес буде запущений. Таким чином в нашій діаграмі є розгалуження.

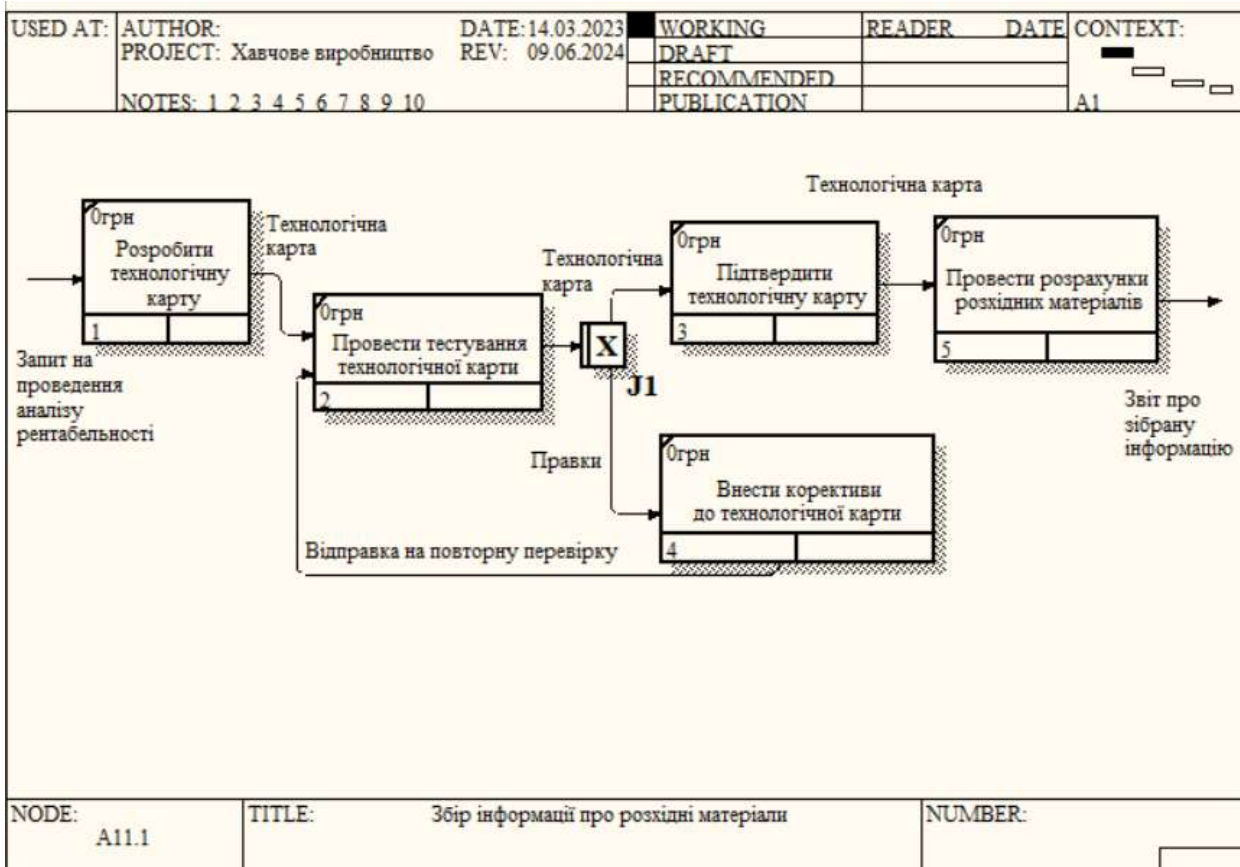


Рисунок 2.5 - Декомпозиційна діаграма для процесу «Збір інформації про розхідні матеріали»

На рисунку 2.6 представлена декомпозиція для процесу розрахунку витрат на комунальні послуги. На цій діаграмі ми можемо бачити значок, який означає асинхронне «і». Отже, всі наступні або попередні процеси повинні бути запущені або завершенні.

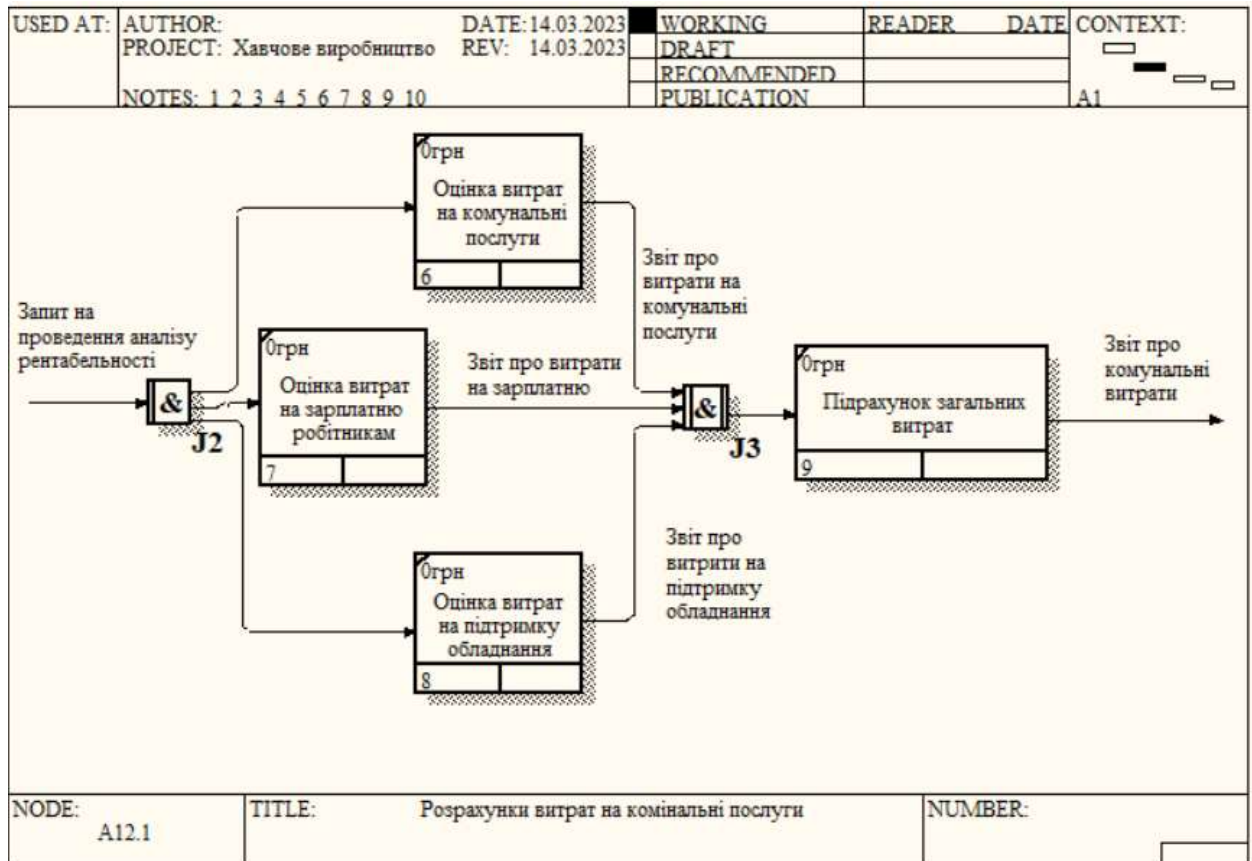


Рисунок 2.6 - Декомпозиційна діаграма для процесу «Розрахунки витрат на комунальні послуги»

На рисунку 2.7 ми бачимо діаграму декомпозиції процесу оцінки ринку збуту. Тут ми можемо побачити вже знайоме нам асинхронне «і». Для того, щоб оцінити рентабельність нам необхідно мати результати оцінки попиту та конкурентів. Отже, спочатку повинні закінчитися ці процеси і тільки після цього можна буде провести оцінку рентабельності.

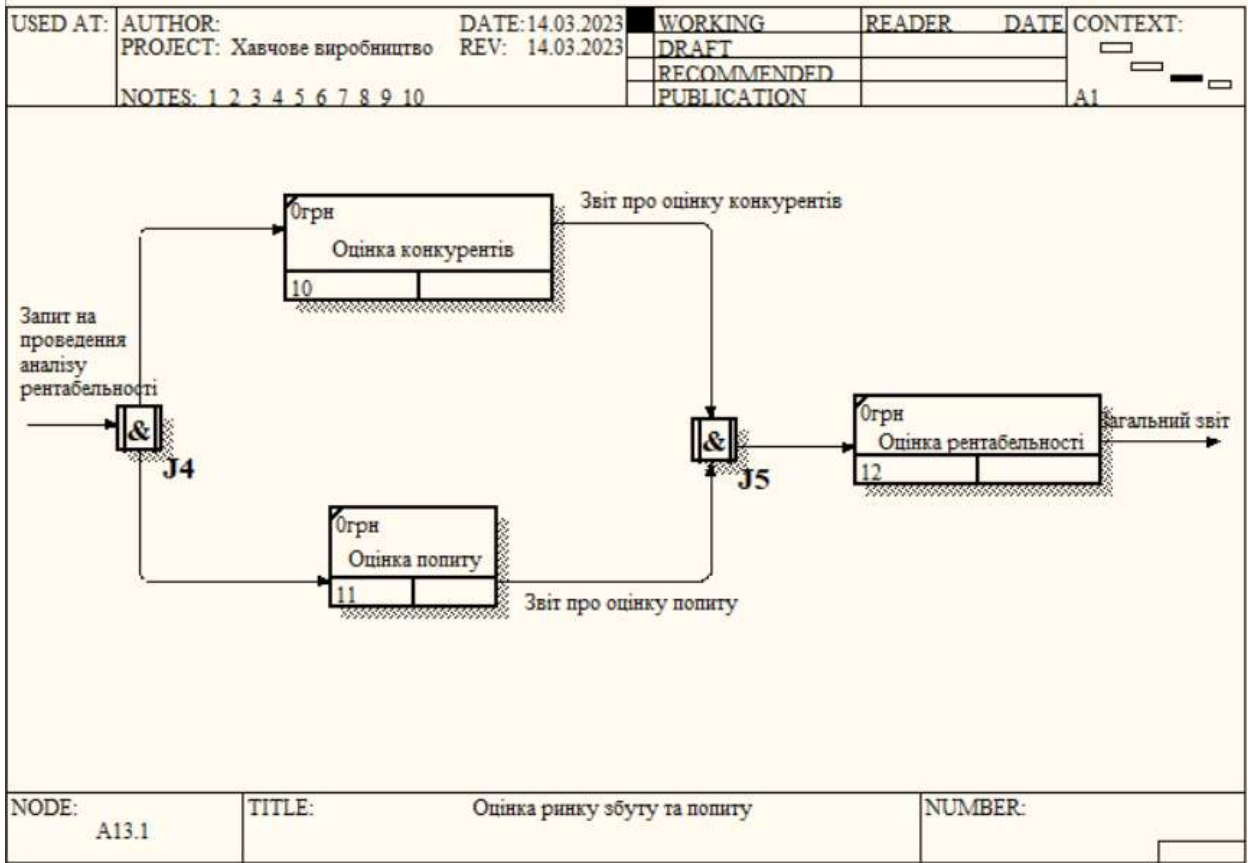


Рисунок 2.7 - Декомпозиційна діаграма для процесу «Оцінка ринку збуту»

На рисунку 2.8 ми бачимо декомпозицію процесу ціноутворення. Також зверніть увагу, що тут ми користуємось загальним звітом, який був в нас на виході на минулій декомпозиції (рис. 2.7).



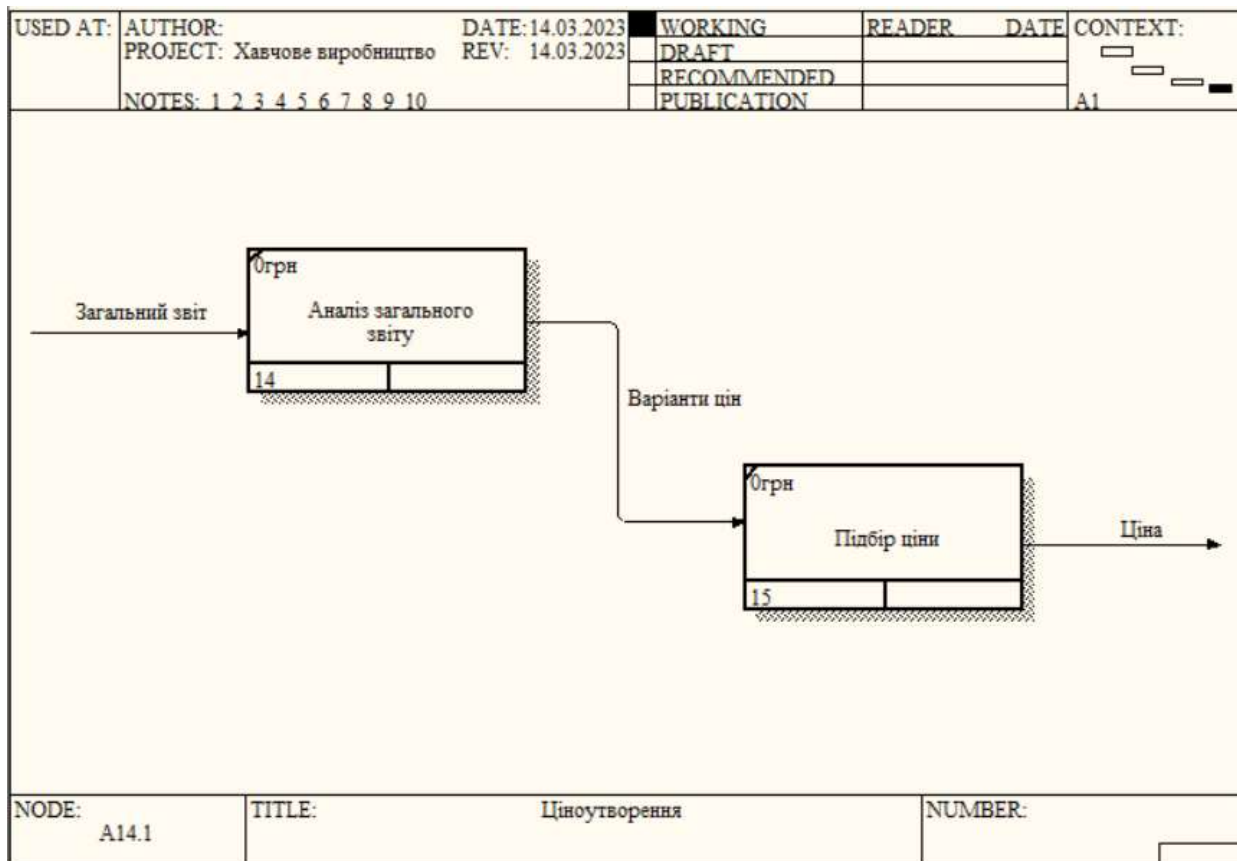


Рисунок 2.8 - Декомпозиційна діаграма для процесу «Ціноутворення»

Декомпозиційна діаграма процесу здійснення закупівлі представлена на рисунку 2.9. Ми можемо побачити, що нам надходить запит на поставку сировини, а після проходження ланцюга процесів ми отримуємо сировину.

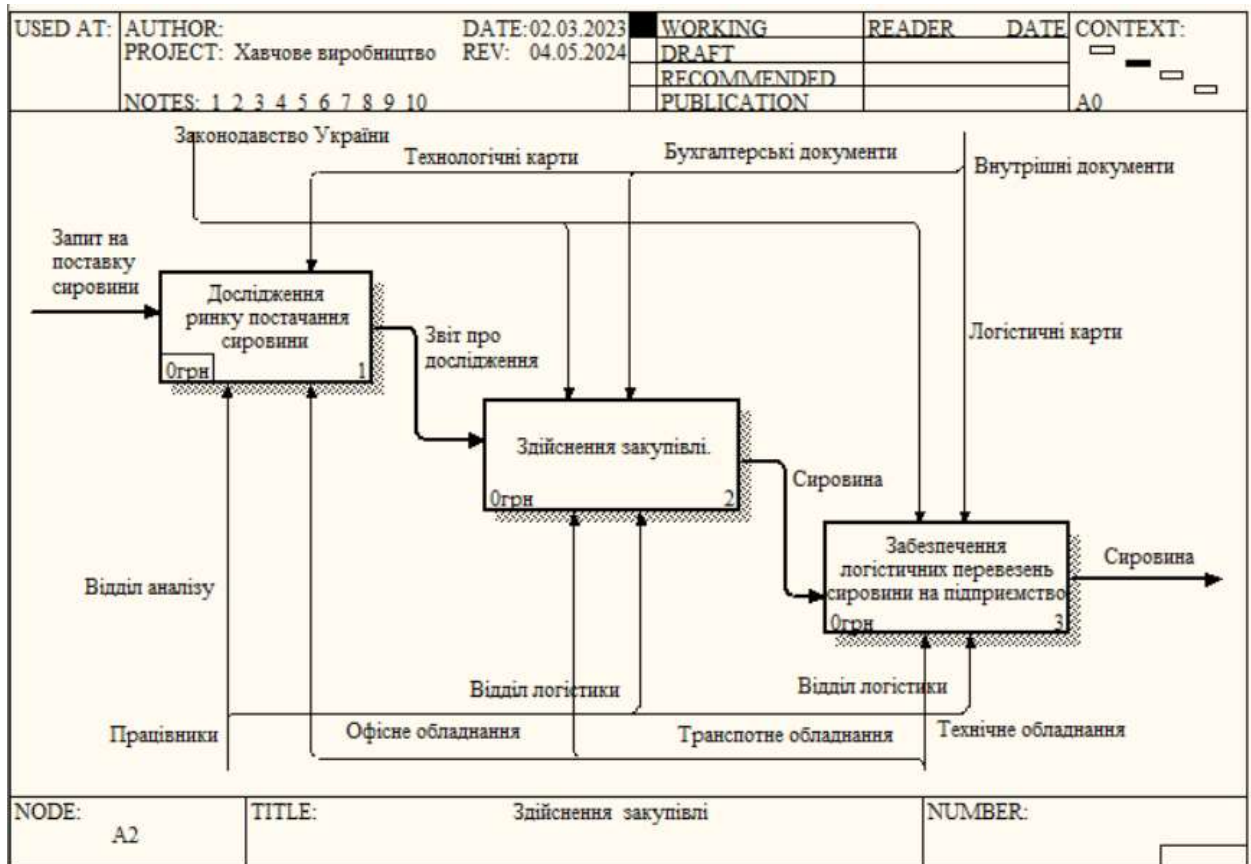


Рисунок 2.9 - Декомпозиційна діаграма для процесу «Здійснення закупівлі»

В декомпозиційній діаграмі процесу дослідження ринку постачання сировини ми можемо бачити на виході звіт про дослідження (рис 2.10). На рисунку 2.11 в діаграмі декомпозиції процесу здійснення закупівлі ми на основі цього звіту здійснюємо закупівлю.

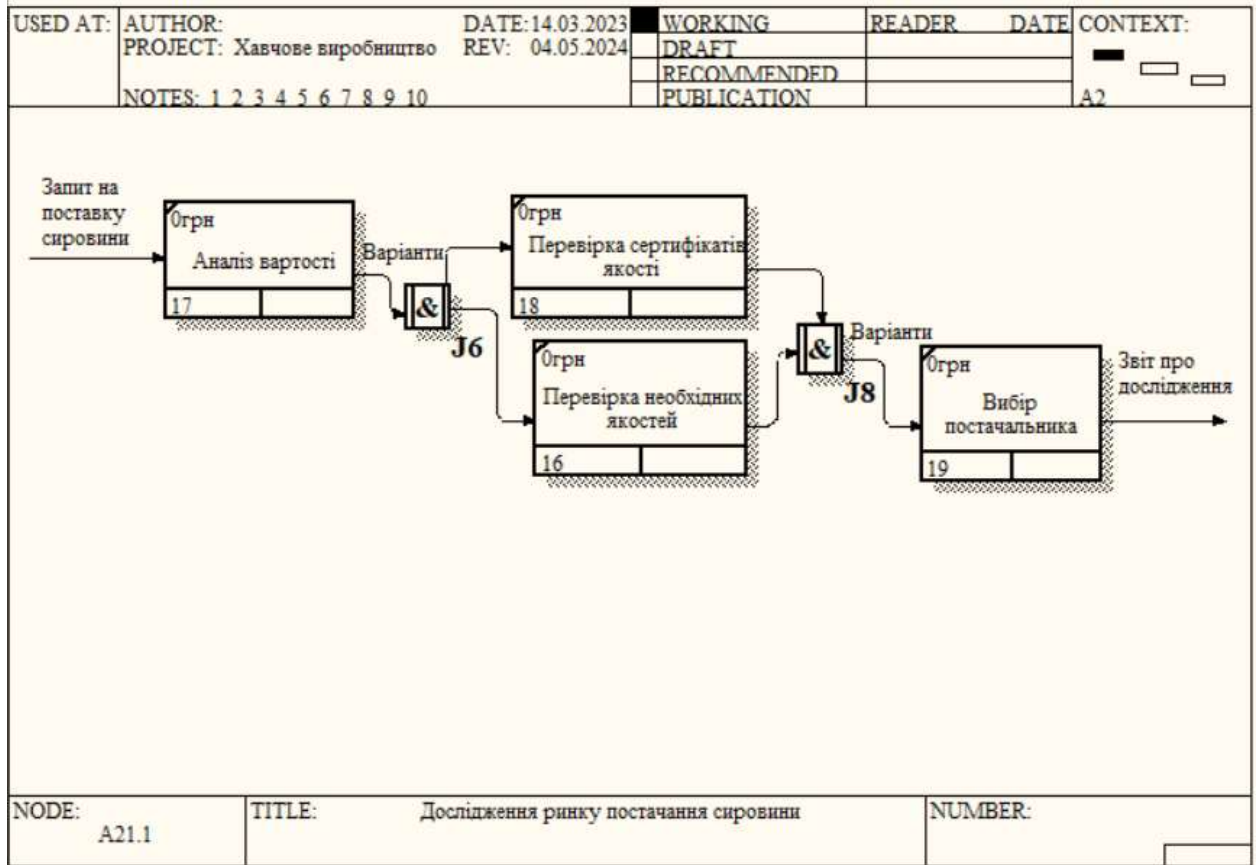


Рисунок 2.10 - Декомпозиційна діаграма для процесу «Дослідження ринку постачання сировини»

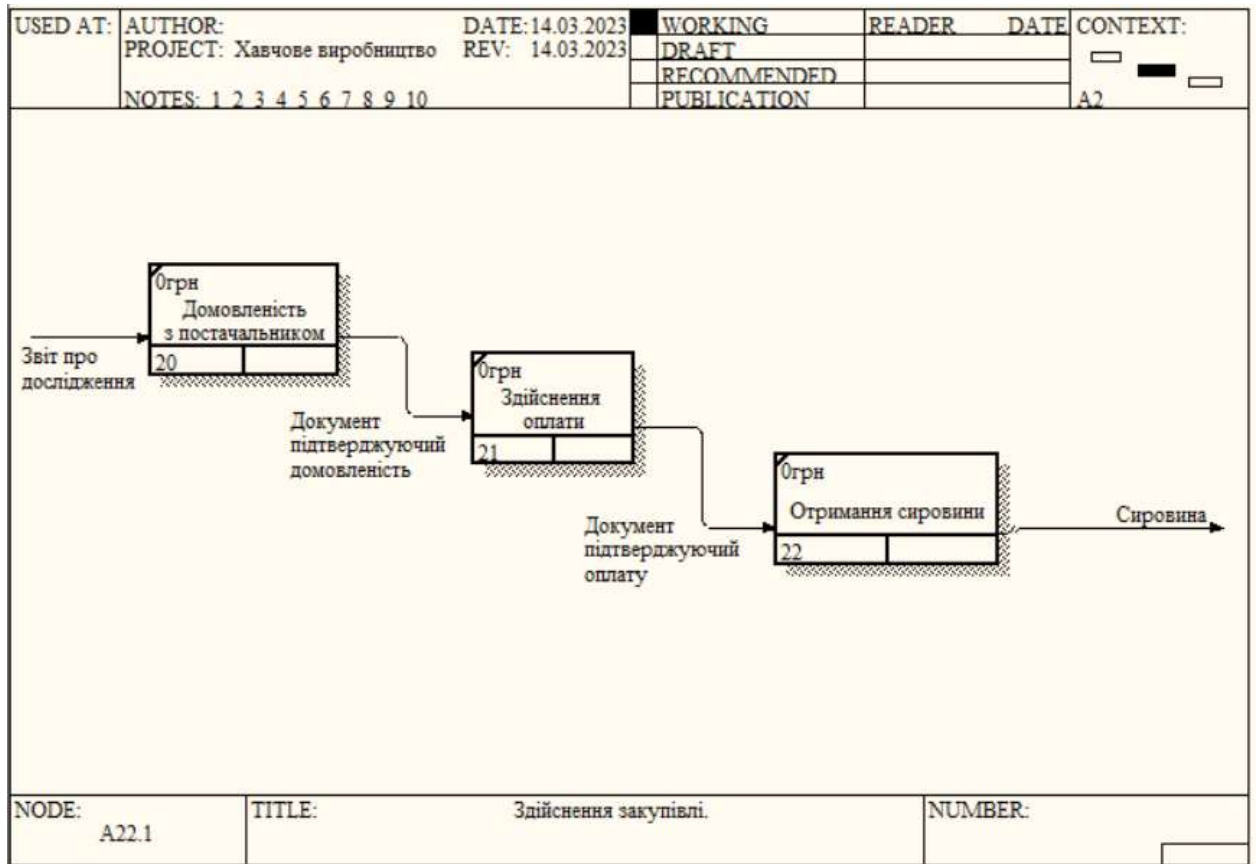


Рисунок 2.11 - Декомпозиційна діаграма для процесу «Здійснення закупівлі»

Декомпозицію зображену на рисунку 2.12, можна пропустити, якщо постачальник, з яким ви домовились, сам постачає сировину або ж ви забираєте її самі на своєму транспорті.

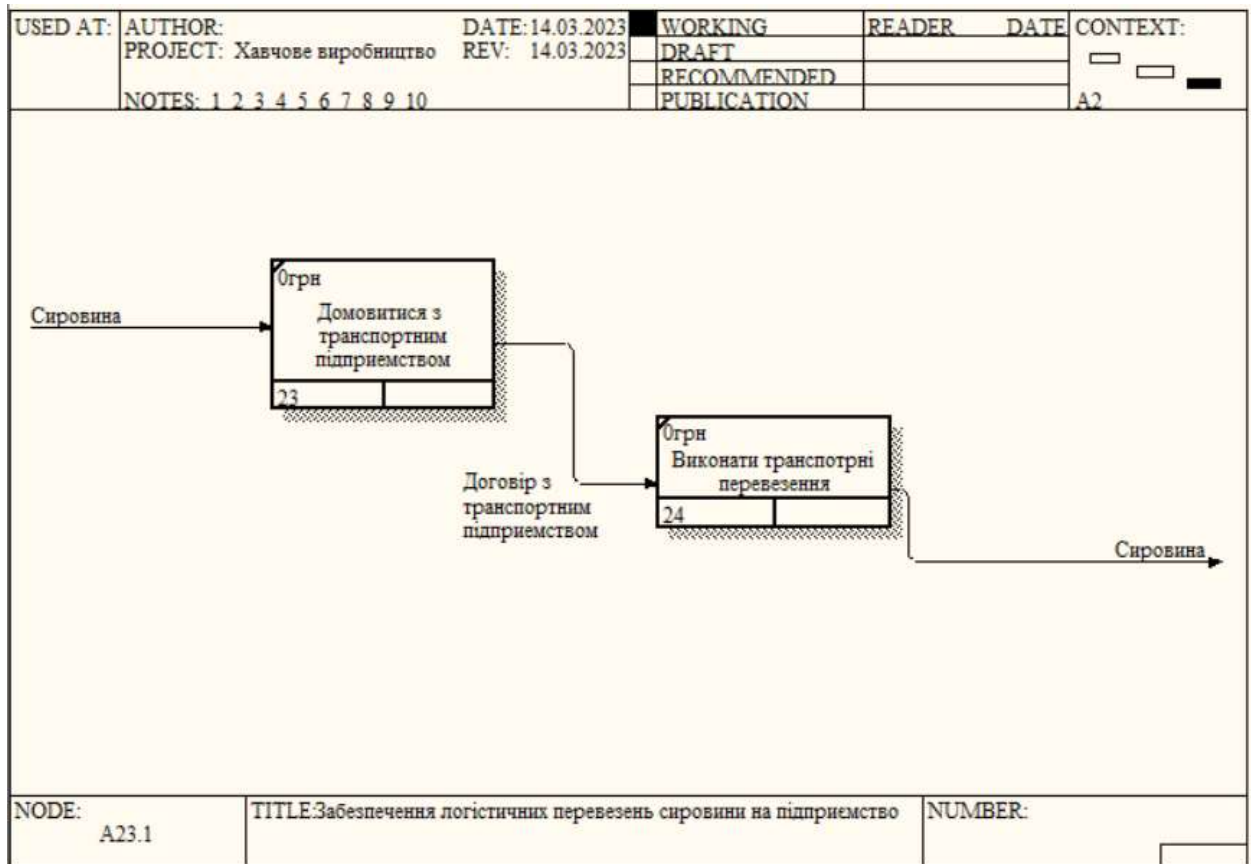


Рисунок 2.12 - Декомпозиційна діаграма для процесу «Забезпечення логістичних перевезень сировини на підприємство»

На рисунку 2.13, зображена декомпозиція процесу виготовлення продукції. Цей процес є дуже важливим. В ньому ми на вході отримуємо сировину, а на виході готову продукцію для продажу.

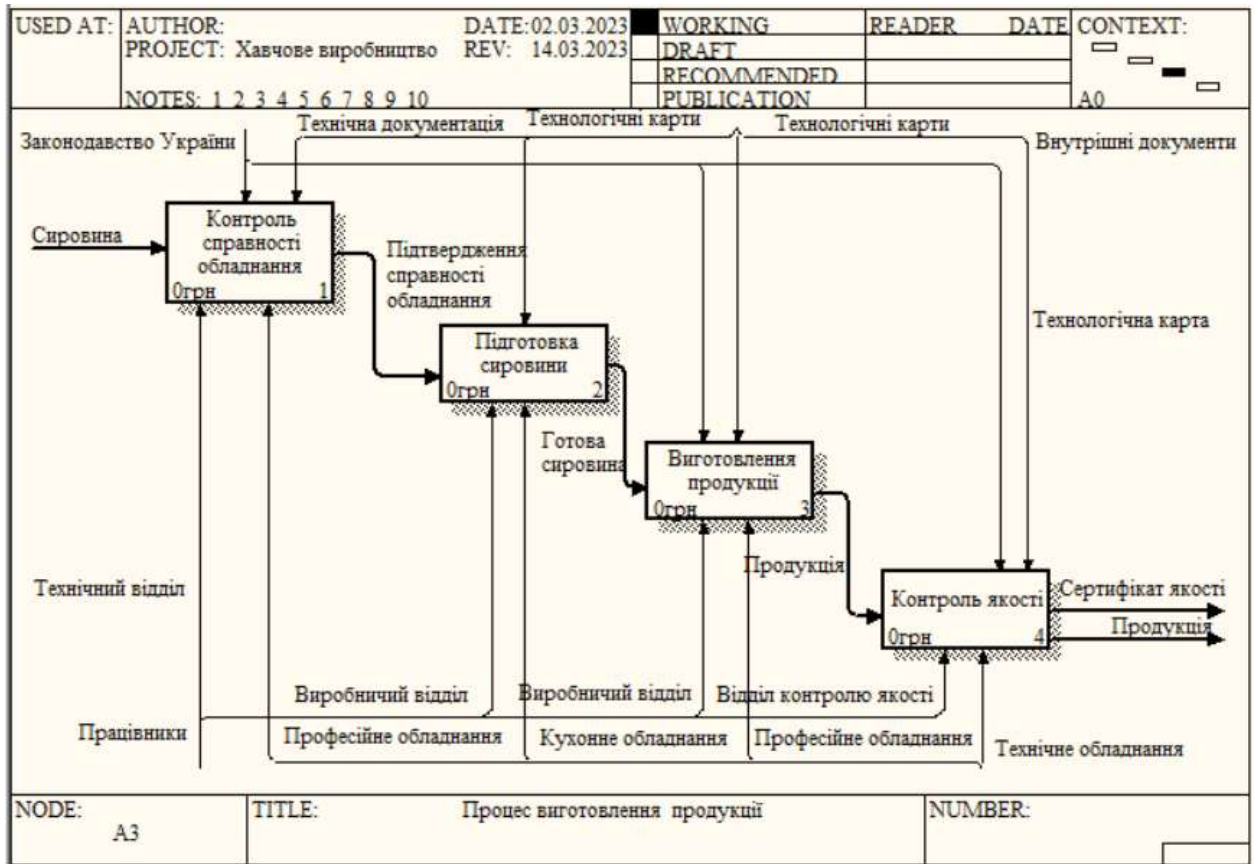


Рисунок 2.13 - Декомпозиційна діаграма для процесу «Процес виготовлення продукції»

На діаграмі представлений на рисунку 2.14 ми бачимо декомпозицію процесу контролю якості обладнання. Також там присутній асинхронний «або», який означає, що тільки один з наступних процесів повинен бути запущений.

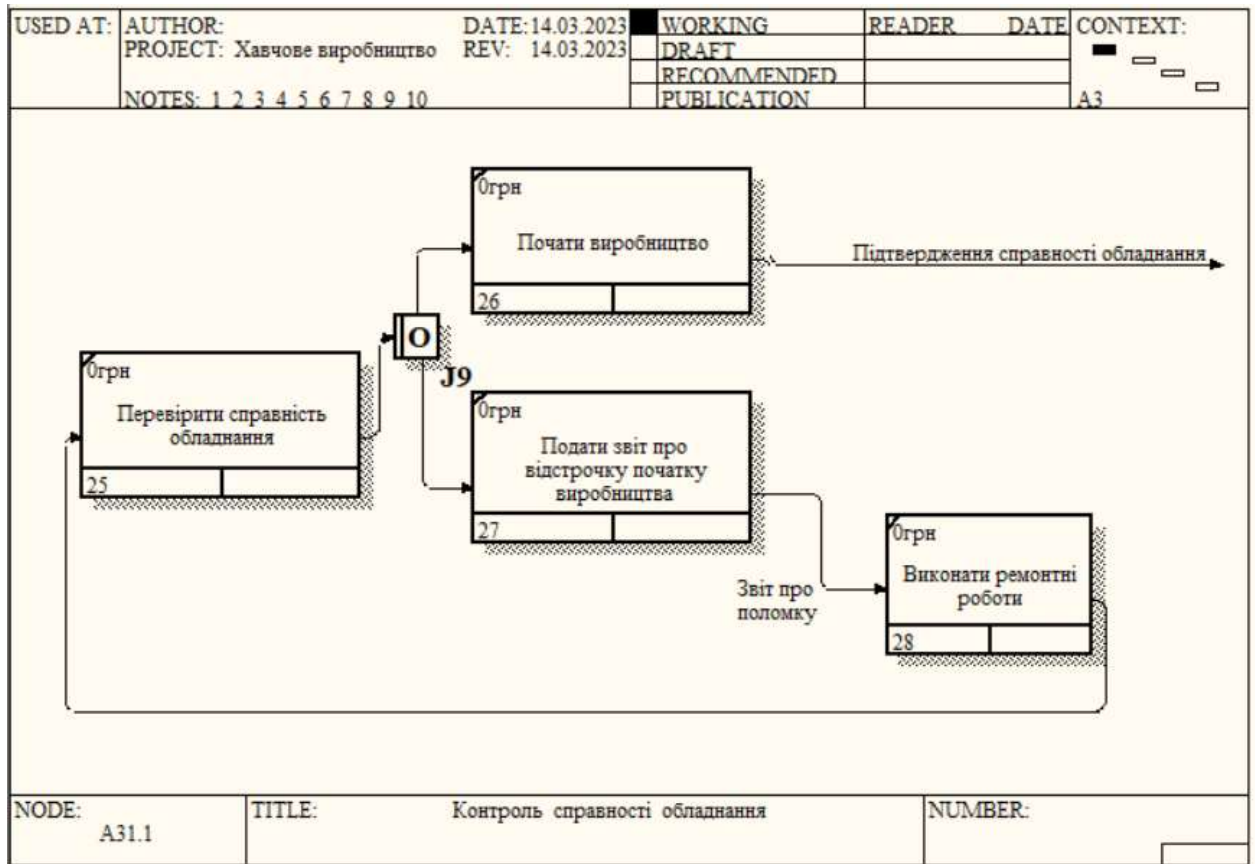


Рисунок 2.14 - Декомпозиційна діаграма для процесу «Контроль справності обладнання»

На декомпозиції процесу підготовки сировини (рис. 2.15) ми можемо бачити вже знайоме виключне «або». В цьому прикладі його можна пояснити таким чином: не вся продукція потребує розмороження, тому та, яка його не потребує, буде розміщена на своє місце, а інша пройде процес розмороження.

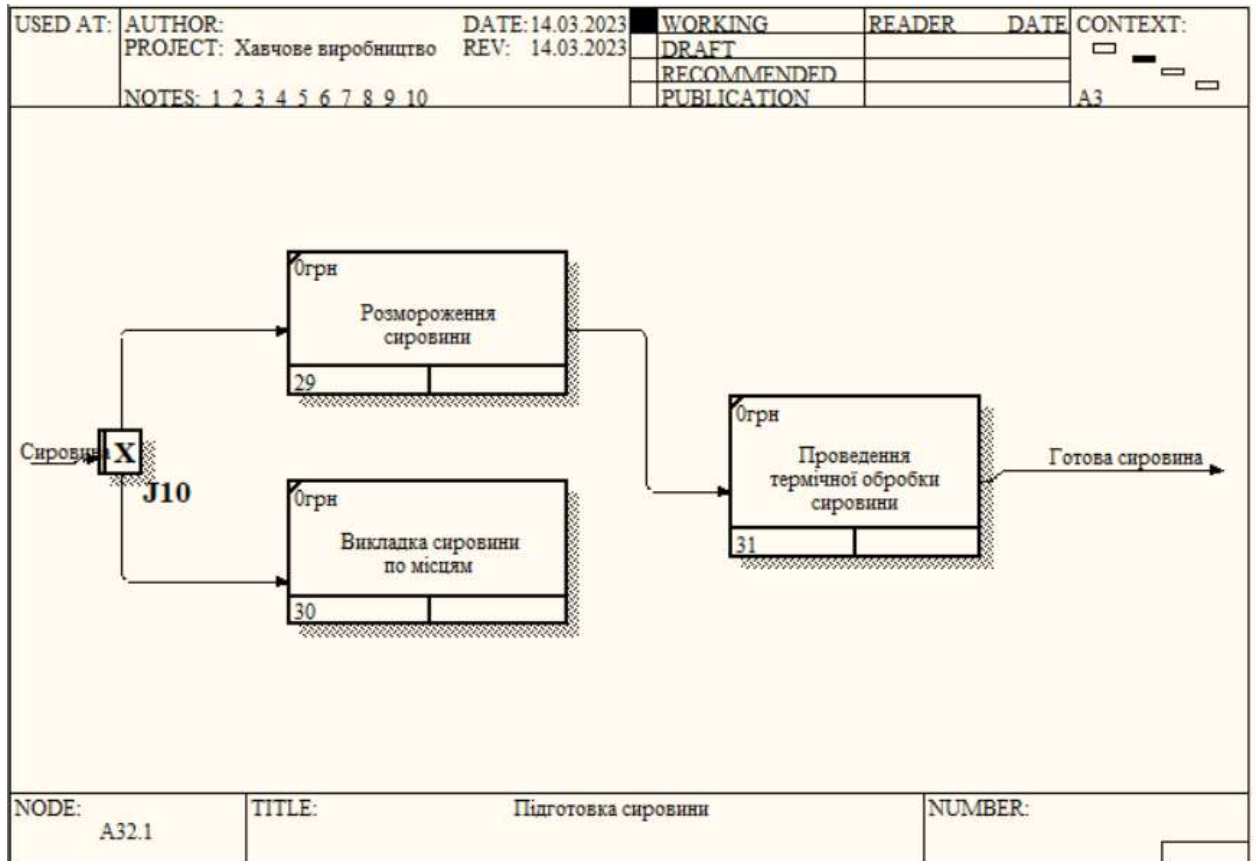


Рисунок 2.15 - Декомпозиційна діаграма для процесу «Підготовка сировини»

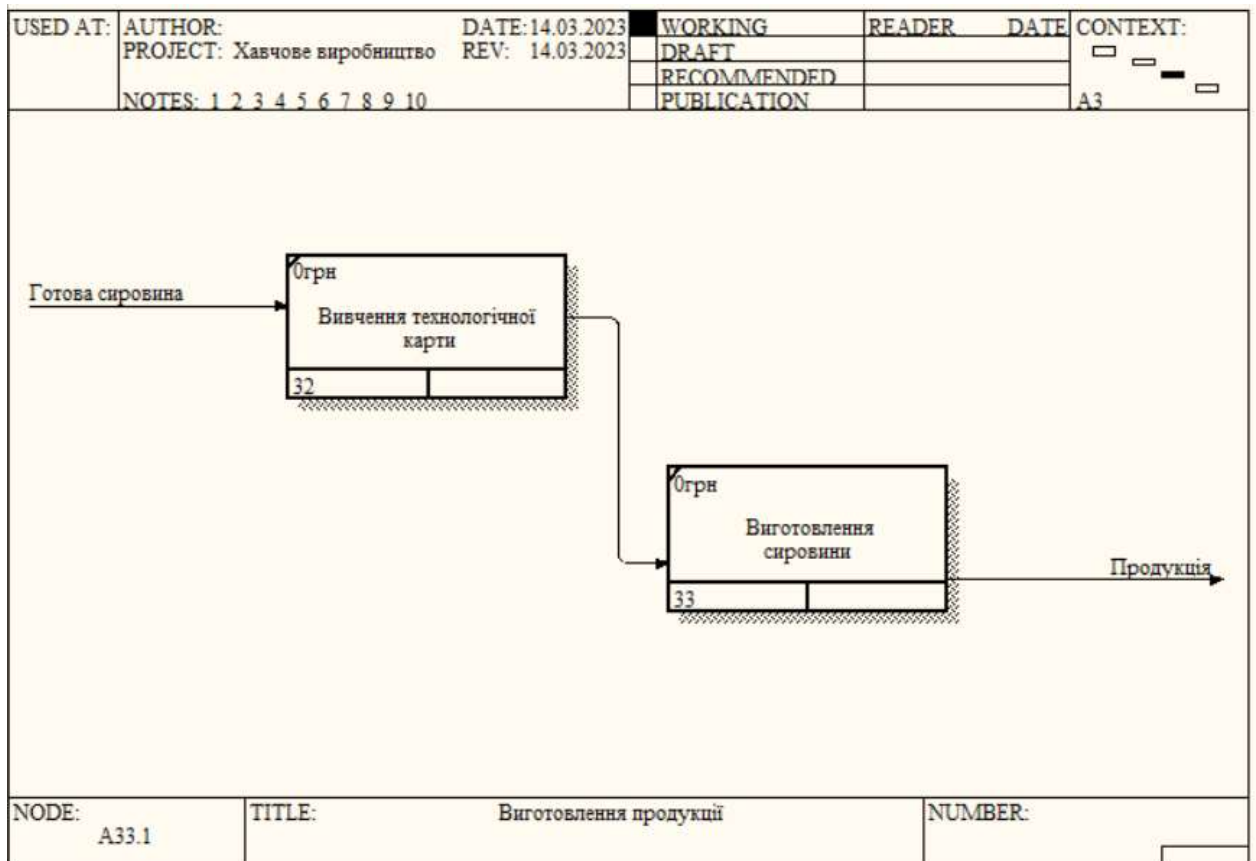




Рисунок 2.16 - Декомпозиційна діаграма для процесу «Виготовлення продукції»

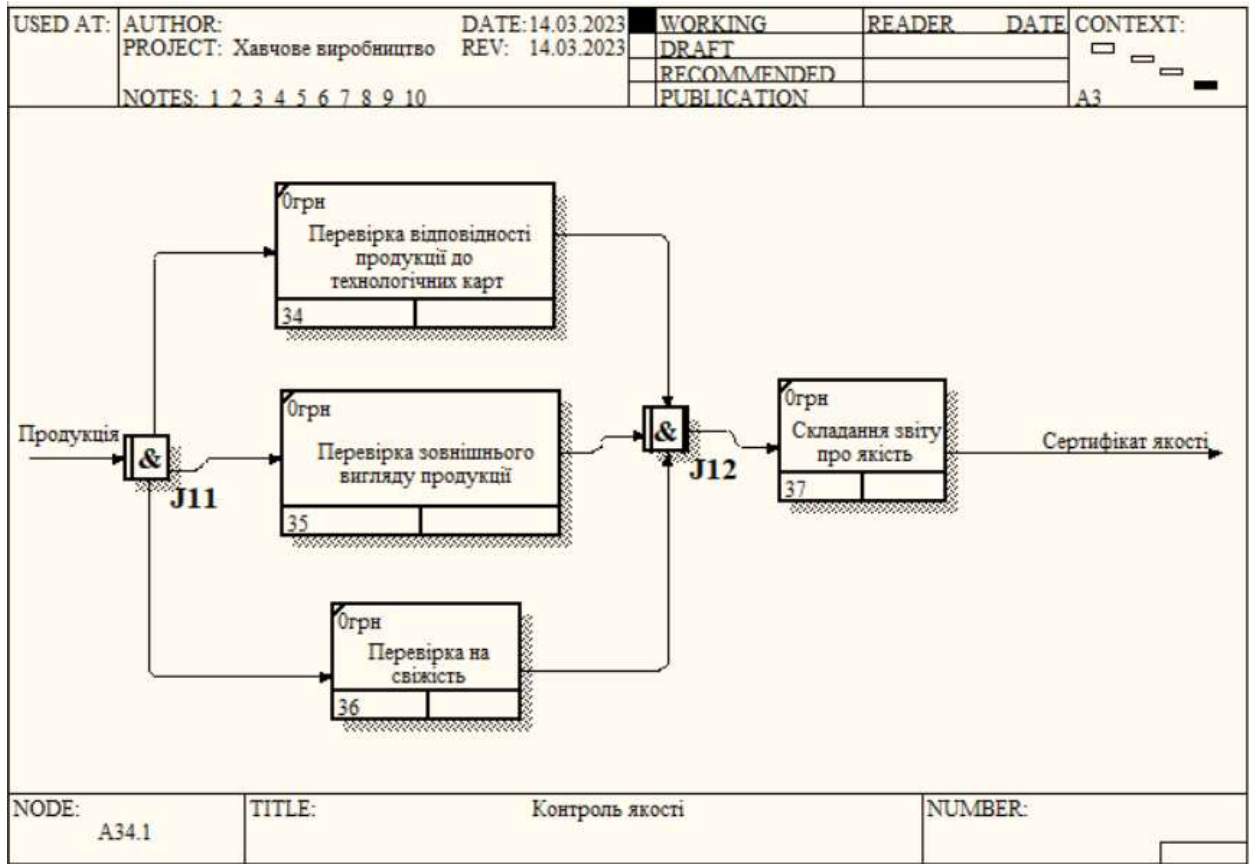


Рисунок 2.17 - Декомпозиційна діаграма для процесу «Контроль якості»

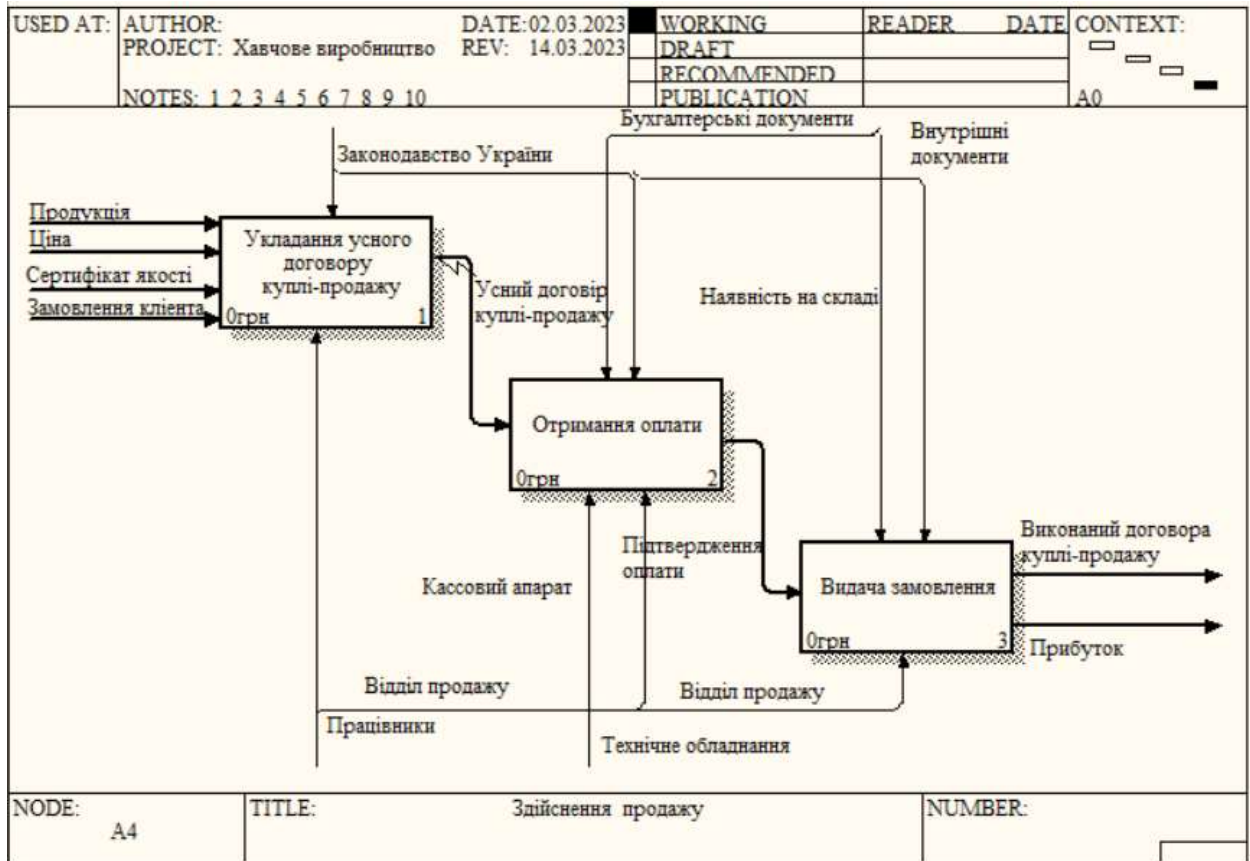


Рисунок 2.18 - Декомпозиційна діаграма для процесу «Здійснення продажу»

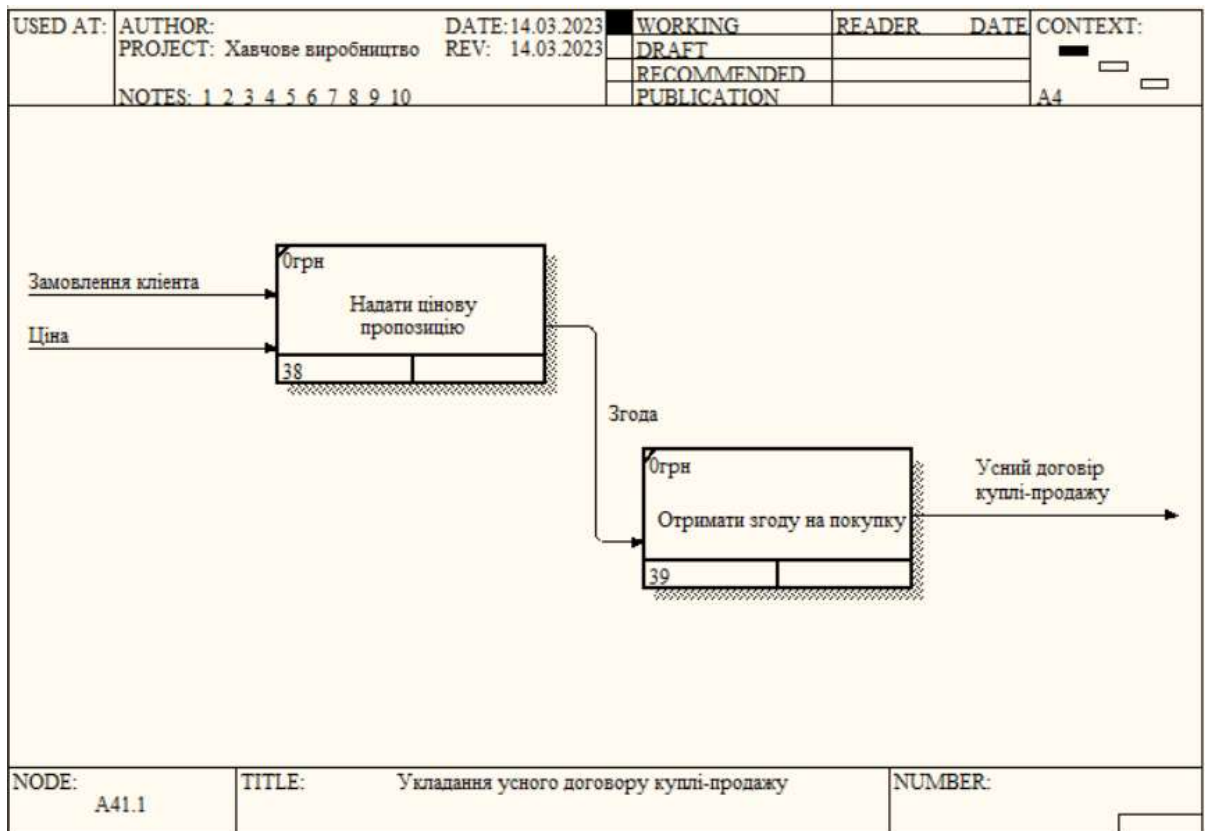


Рисунок 2.19 - Декомпозиційна діаграма для процесу «Укладання усного договору купівлі-продажу»

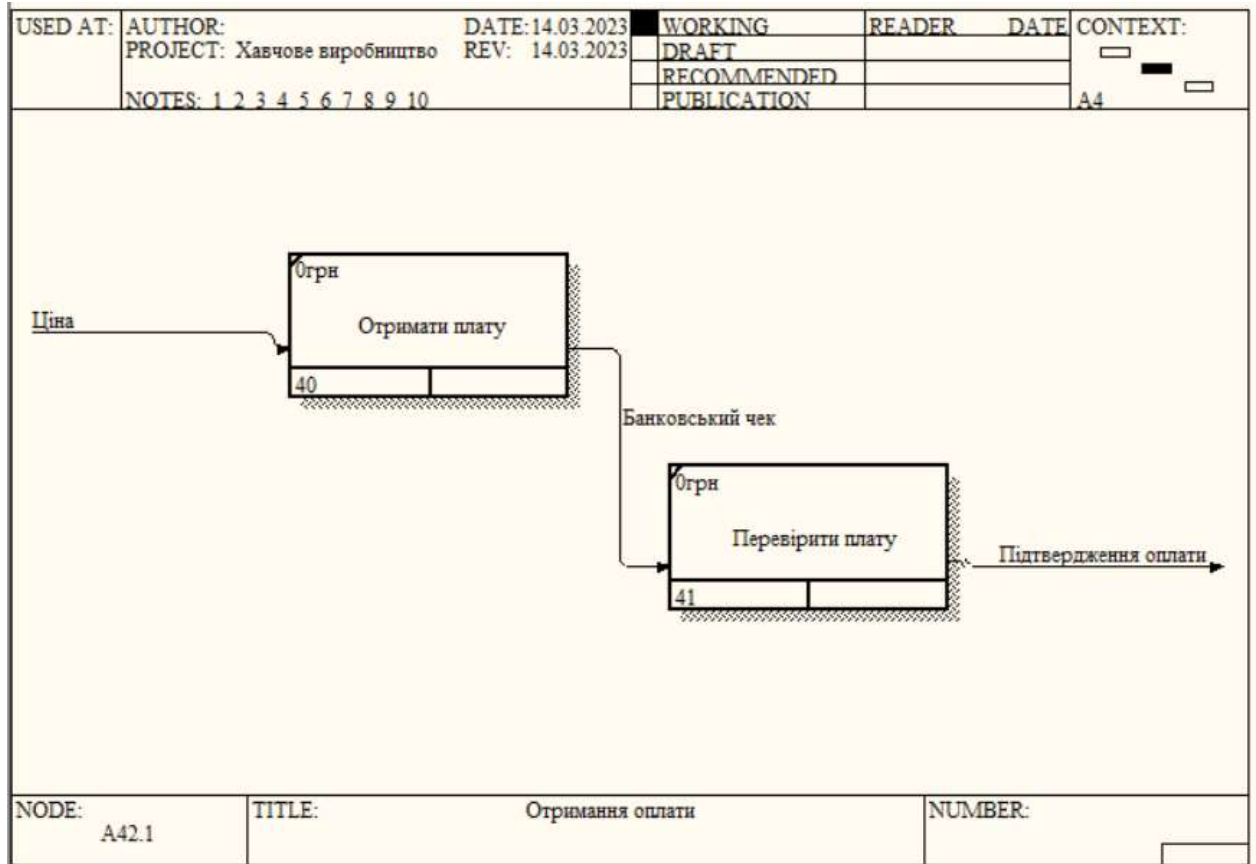


Рисунок 2.20 - Декомпозиційна діаграма для процесу «Отримання оплати»

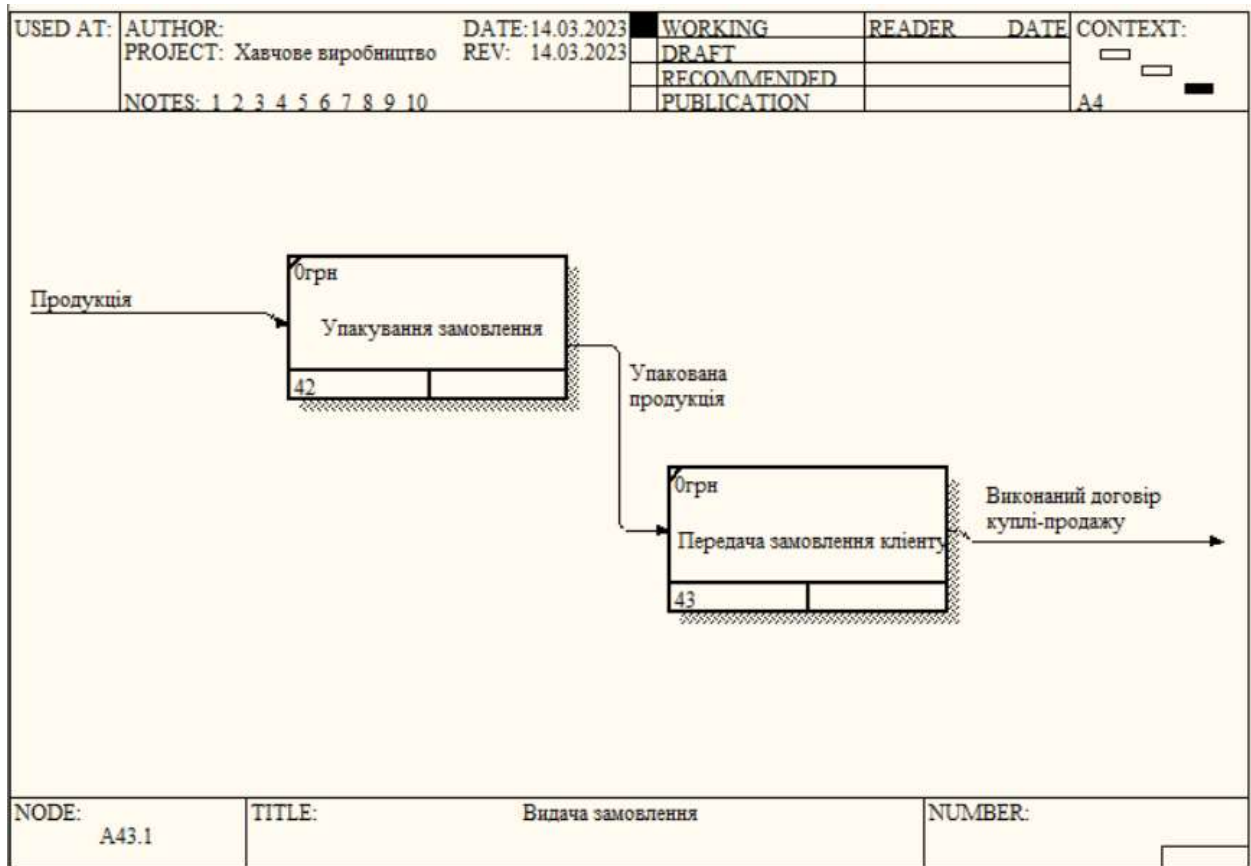


Рисунок 2.21 - Декомпозиційна діаграма IDEF0 для процесу «Видача замовлення»

## 2.2 Нормалізація бази даних

Нормалізація бази даних включає в себе процес організації таблиць і стовпців бази даних для зменшення надлишковості даних і забезпечення цілісності даних.

Отже, нам необхідно нормалізувати всі таблиці бази даних до третьої нормальної форми. Давайте роздивимось процес нормалізації на прикладі моєї таблиці Sales.

На початку нормалізації таблиця має такі поля:

1. date
2. product\_name
3. quantity

#### 4. price

Перший крок: Перевірка на першу нормальну форму.

Всі атрибути містять атомарні значення, отже таблиця знаходиться в першій нормальній формі.

Другий крок: Перевірка на другу нормальну форму.

Всі неключові атрибути повністю залежать від первинного ключа, як ми можемо бачити це не так, отже таблиця не знаходиться в другій нормальній формі. З цього ж випливає. Що вона не знаходиться і в третій нормальній формі.

Третій крок: Перехід до другої нормальної форми.

Щоб перейти до другої нормальної форми, необхідно розділити наші дані на дві таблиці, але я вирішила розділити їх на три таблиці. Це пов'язано з тим, що в нас відбувається багато продажів в один день, а також одна продаж може розбиватися на декілька для того, щоб не було атомарних значень. Це відбувається тоді, коли один покупець купує два і більше видів продукції. Отже, я вирішила виділити окремо дані про дату продажу.

На виході отримаємо три таблиці:

1) Sales:

- sale\_number,
- date.

2) Sale\_details:

- sale\_number,
- product\_id,
- quantity\_sold.

3) Products:

- product\_id,
- product\_name,

- quantity,
- unit,
- price.

Тепер наші таблиці знаходяться в другій нормальній формі, це означає, що неключові атрибути залежать тільки від первинного ключа.

Четвертий крок: Перевірка на третю нормальну форму.

Таблиці не містить транзитивних зв'язків, отже наші таблиці знаходяться в третій нормальній формі.

Далі я повторила ці кроки з кожною таблицею вихідних значень. На виході в мене вийшло 11 таблиць.

Опис таблиць:

**Products:** таблиця містить інформацію, про товар який продає підприємство. В ній знаходиться унікальний первинний ключ (`product_id`), ціну на продукцію, та вимір для якого формується ціна (кілограм або штука)

**Preparations:** таблиця містить інформацію про заготовки, з яких виготовляється товар на продаж. Іона містить унікальний первинний ключ (`preparation_id`), назву заготовки та вимір в якому вона фіксується.

**Ingredients:** таблиця містить інформацію про інгредієнти, які необхідні для приготування заготовок. В таблиці знаходиться унікальний первинний ключ(`ingredient_id`), назва інгредієнта та вимір в якому він фіксується.

**Technological\_map\_preparations:** таблиця містить інформацію про технологічні карти заготовок. Тут вказано скільки кожного інгредієнта необхідно використати, щоб виготовити 1 кілограм заготовки. Первинний ключ тут складається з комбінації полів `preparation_id` і `ingredient_id`.

**Technological\_map\_products:** Таблиця містить інформацію про технологічну карту продукції. В цій таблиці вказано скільки необхідно

використати кожної заготовки, щоб виготовити один продукт. Первинний ключ тут складається з комбінації полів `product_id` і `preparation_id`.

`Sales`: таблиця містить інформацію про продажі, а саме унікальний номер продажі як первинний ключ (`sale_number`) та дату коли покупка була здійснена.

`Sale_details`: Таблиця містить більш детальні дані про продажі, а саме скільки кожного продукту придбала один покупець. Первинний ключ тут складається з комбінації полів `sale_number` і `product_id`.

`Expenses`: таблиця містить інформацію про витрати, а саме дату витрати, номер типу витрати, суму витрати та первинний ключ `expense_id`.

`Expense_types`: ця таблиця містить первинний ключ `expense_type_id`, який відображає номер типу витрати, та назву типу витрати.

`Suppliers`: Таблиця містить інформацію про постачальників, а саме первинний ключ `supplier_id` та назву постачальника.

`Supplier_prices`: Таблиця містить інформацію про ціни постачальника на інгредієнти. Первинний ключ складається з комбінації полів `supplier_id` і `ingredient_id`.

Таблиця 2.3 демонструє аналіз на відповідність до кожної нормальної форми, а також атрибути і первинні ключі які входять в кожну таблицю.

Таблиця 2.3

### Аналіз нормалізації таблиць бази даних

Назва таблиці	Атрибути	Первинний ключ	1 НФ	2 НФ	3 НФ
-1-	-2-	-3-	-4-	-5-	-6-
Supplier_prices	supplier_id (PK), ingredient_id (PK), quantity, unit, price	(supplier_id, ingredient_id)	+	+	+
Suppliers	supplier_id (PK), supplier_name	supplier_id	+	+	+

Продовження табл. 2.3

-1-	-2-	-3-	-4-	-5-	-6-
Expense_types	expense_type_id (PK), expense_type_name	expense_type_id	+	+	+
Expenses	expense_id (PK), date, expense_type_id, value	expense_id	+	+	+
Sale_details	sale_number (PK), product_id (PK), quantity_sold	(sale_number, product_id)	+	+	+
Sales	sale_number (PK), date	sale_number	+	+	+
Technological_map_products	product_id (PK), preparation_id (PK), quantity, unit	(product_id, preparation_id)	+	+	+
Technological_map_preparations	preparation_id (PK), ingredient_id (PK), quantity, unit	(preparation_id, ingredient_id)	+	+	+
Preparations	preparation_id (PK), preparation_name, quantity, unit	preparation_id	+	+	+
Ingredients	ingredient_id (PK), ingredient_name, quantity, unit	ingredient_id	+	+	+
Products	product_id (PK), product_name, quantity, unit, price	product_id	+	+	+

## 2.3 Створення OLAP куба в Power Pivot

### 2.3.1 Внесення даних в модель

Щоб розпочати проектування нашої моделі даних, нам необхідно включити надбудову Power Pivot, тому що саме з нею ми і будемо працювати.



Наступним кроком, буде внести всі наші таблиці в Microsoft Excel.

Після того, як всі наші таблиці знаходяться в одному документі Microsoft Excel, можемо починати додавати їх до нашої моделі даних. Для цього заходимо на вкладку Power Pivot. На цій вкладці в спливаючій стрічці знаходимо кнопку «Додати до моделі даних». В спливаючому вікні вибираємо діапазон з якого буде створена наша таблиця. Після підтвердження дій відкриється вікно Power Pivot, де ми зможемо побачити нашу створену таблицю (рис 2.22).

The screenshot shows the Power Pivot window in Microsoft Excel. The window title is 'Power Pivot для Excel - OLAP-Диплом (version 1).xlsx'. The ribbon includes 'Файл', 'Головна', 'Конструктор', and 'Додатково'. The 'Додатково' ribbon has options like 'Буфер обміну', 'Отримати зовнішні дані', 'Оновити', 'Таблиця PivotTable', 'Тип даних', 'Формат', 'Сортування з фільтрацією', 'Знайти', 'Обчислення', 'Подання даних', 'Подання схеми', 'Відобразити приховані об'єкти', and 'Область обчислення'. The main area displays a table with the following data:

	suppl...	ingredie...	quantity	unit	price	Додати стовпець
1	1	1	1	кг	30	
2	1	2	2	кг	15	
3	2	7	7	кг	23	
4	3	13	13	кг	45	
5	3	9	9	кг	50	
6	4	4	4	кг	13	
7	4	6	6	кг	29	
8	5	8	8	кг	80	
9	6	15	15	л	53	
10	6	16	16	кг	100	
11	7	10	10	кг	13	
12	7	11	11	кг	50	
13	7	3	3	шт	3	
14	4	5	5	кг	18	
15	7	12	12	кг	100	

The bottom of the window shows a taskbar with several worksheets: 'Preparations', 'Ingredients', 'Products', 'Technological\_map\_preparations', 'Expense\_types', and 'Supplier\_prices'.

Рисунок 2.22- Вікно Power Pivot у Microsoft Excel для створення OLAP-куба.

### 2.3.2 Побудова зв'язків між таблицями

Після внесення всіх наших таблиць в модель даних, наступним кроком буде побудувати зв'язки між таблицями.

Стовпці, які використовують для створення зав'язків, зазвичай мають однакові імена в різних таблицях і називаються ключами зв'язку. На одному боці зв'язка стовпець повинен мати унікальні значення для кожного рядка, коли на іншому значення може повторюватися. Коли стовпець має унікальне значення для кожного рядка, то його називають ключем.

Зв'язки між таблицями будуються на основі первинного ключа та зовнішнього ключа. Самі зв'язки поділяються на 3 типи:

- один-до-одного (1:1),
- один-до-багатьох (1:N),
- багато-до-багатьох (N:N).

В Power Pivot всі зв'язки в моделі даних повинні мати тип «один до багатьох»(one-to-many). Зв'язки типу «багато до багатьох»(many-to-many) є неприпустимими в цьому середовищі.

Кожен зв'язок має фільтрацію, вона може бути односпрямована і двонаправлена. Фільтрація завжди відбувається від одного до багатьох.

На рисунку 2.23 зображені зв'язки між таблицями в Power Pivot у Microsoft Excel.

Активний	Таблиця 1	Кількість елементів	Напрямок фільтра	Таблиця 2
Так	Technological_map_preparations [preparation_id]	Багато до одного (*:1)	<< До Technological_map_preparations	Preparations [preparation_id]
Так	Technological_map_preparations [ingredient_id]	Багато до одного (*:1)	<< До Technological_map_preparations	Ingredients [ingredient_id]
Так	Technological_map_products [product_id]	Багато до одного (*:1)	<< До Technological_map_products	Products [product_id]
Так	Technological_map_products [preparation_id]	Багато до одного (*:1)	<< До Technological_map_products	Preparations [preparation_id]
Так	Sale_details [sale_number]	Багато до одного (*:1)	<< До Sale_details	Sales [sale_number]
Так	Sale_details [product_id]	Багато до одного (*:1)	<< До Sale_details	Products [product_id]
Так	Expenses [expense_type_id]	Багато до одного (*:1)	<< До Expenses	Expense_types [expense_type_id]
Так	Supplier_prices [supplier_id]	Багато до одного (*:1)	<< До Supplier_prices	Suppliers [supplier_id]
Так	Supplier_prices [ingredient_id]	Багато до одного (*:1)	<< До Supplier_prices	Ingredients [ingredient_id]

Рисунок 2.23 - Вікно управління зв'язками в Power Pivot у Microsoft Excel з налаштованими зв'язками для OLAP кубу харчового підприємства.

На рисунку 2.24 представлена схема, яка демонструє як таблиці пов'язані між собою. Стрілки між таблицями вказують напрямок фільтрації.

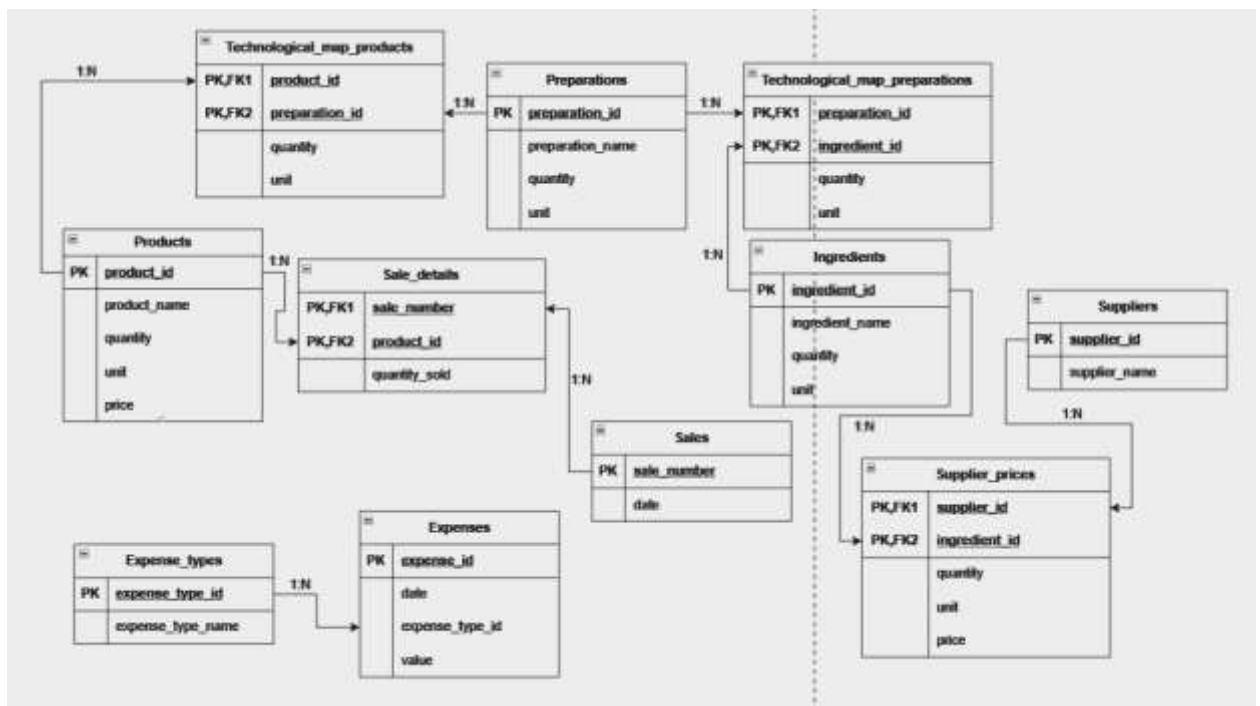


Рисунок 2.24 - Схема бази даних створеного OLAP куба.

Коли зв'язки між таблицями встановлені, можемо завершити створення OLAP куба, для цього в вікні Power Pivot на вкладці «Головна» натискаємо на кнопку «Таблиця PivotTable».

З такою таблицею вже можна працювати. Наприклад на рисунку 2.25 показана таблиця PivotTable, яка відображає суму по кількості продаж для кожного товару в залежності від місяця.

Сумма по столбцу quantity_sold	Позначки стовпців												
	янв	фев	мар	апр	май	июн	июл	авг	сен	окт	ноя	дек	Загальний підсумок
Позначки рядків													
Біляш	609	555	1096	1093	970	537	604	537	1059	1063	994	615	9732
Пиріжок з горохом	469	566	1012	1085	1173	568	535	600	1032	1066	973	630	9709
Пиріжок з капустою	593	523	1089	1031	987	583	585	573	1078	1011	907	688	9648
Пиріжок з картоплею	605	592	983	1180	899	575	614	578	949	989	1004	659	9627
Пиріжок з картоплею та грибами	585	520	1088	1013	969	613	600	538	1135	876	946	653	9536
Пиріжок з картоплею та печінкою	541	554	1026	1041	1087	593	653	597	1026	1112	936	691	9857
Пиріжок з рисом, яйцем та зеленню	548	559	1112	1023	960	524	598	571	1039	995	993	658	9580
Пончики	516	474	839	770	807	495	572	481	801	746	794	538	7833
Сосиска в тісті	610	508	1177	1046	1005	595	613	559	1103	1031	986	654	9887
Чебурек	572	523	1073	984	1042	506	613	574	1040	896	1086	662	9571
<b>Загальний підсумок</b>	<b>5648</b>	<b>5374</b>	<b>10495</b>	<b>10266</b>	<b>9899</b>	<b>5589</b>	<b>5987</b>	<b>5608</b>	<b>10262</b>	<b>9785</b>	<b>9619</b>	<b>6448</b>	<b>94980</b>

Рисунок 2.25 - Зведена таблиця продажів за місяцями у Power Pivot.

Кожен місяць можна розгорнути і подивитися продажі по дням. Це зображено на рисунку 2.26.

Сумма по столбцу quantity_sold	Позначки							
	янв							
Позначки рядків	01.01.2023	04.01.2023	06.01.2023	07.01.2023	08.01.2023	11.01.2023	13.01.2023	
Біляш	33	41	29	46	33	30	23	
Пиріжок з горохом	29	27	23	30	25	36	25	
Пиріжок з капустою	46	32	27	27	47	53	39	
Пиріжок з картоплею	49	39	38	33	20	20	43	
Пиріжок з картоплею та грибами	49	38	31	37	31	30	32	
Пиріжок з картоплею та печінкою	32	32	26	24	28	37	38	
Пиріжок з рисом, яйцем та зеленню	24	32	35	29	37	30	36	
Пончики	32	20	28	36	35	29	30	
Сосиска в тісті	22	29	33	52	32	37	31	
Чебурек	26	35	32	22	46	44	31	
<b>Загальний підсумок</b>	<b>342</b>	<b>325</b>	<b>302</b>	<b>336</b>	<b>334</b>	<b>346</b>	<b>328</b>	

Рисунок 2.26 - Зведена таблиця продажів за днями у Power Pivot.

### 2.3.3 Написання мір на мові DAX

Тепер, коли наш OLAP куб сформований, ми можемо прописати в ньому необхідні міри, які будемо використовувати в нашому дослідженні.

Перша міра яку я створила, це міра яка обчислює загальну кількість проданих одиниць продукції. В ній я використала функцію SUM, яка обчислює суму всіх значень в колонці, яка містить інформацію про кількість проданих одиниць в кожній покупці.

$$\text{TotalUnitsSold}=\text{SUM}(\text{Sale\_details}[\text{quantity\_sold}])$$

Далі я створила міру, яка рахує загальні щомісячні витрати.

$$\begin{aligned} \text{MonthlyExpenses} &= \text{CALCULATE}(\text{SUM}(\text{Expenses}[\text{value}]); \\ &\text{FILTER}(\text{ALL}(\text{Calendar}); \\ &\text{Calendar}[\text{Year}] = 2023 \ \&\& \\ &\text{Calendar}[\text{Month}] = \text{MAX}(\text{'Calendar'}[\text{Month}]) \\ &); \end{aligned}$$

Функція працює наступним чином:

1. Функція ALL(Calendar) ігнорує всі фільтри, які могли бути накладені на цю таблицю.
2. Функція FILTER відбирає, лише рядки в яких значиться 2023 рік.
3. Для відфільтрованих даних функція SUM(Expenses[value]) рахує суму по колонці value в таблиці Expenses.
4. Функція CALCULATE змінює контекст розрахунків і обчислює суму витрат для відфільтрованих даних.

Наступна міра написана для розрахунку загальної кількості одиниць продукції, проданої в певний місяць.

```
TotalUnitsProduced=CALCULATE(
    SUM(Sale_details[quantity_sold]);
    FILTER(
        ALL(Calendar);
        Calendar[Year] = 2023 &&
        'Calendar'[Month] = MAX('Calendar'[Month])
    )
)
```

Формула працює наступним чином:

1. Функція ALL(Calendar) ігнорує всі фільтри, які могли бути накладені на цю таблицю.
2. Функція FILTER відбирає, лише рядки в яких значиться 2023 рік.
3. Функція SUM(Sale\_details[quantity\_sold]) розраховує суму відфільтрованих значень у колонці quantity\_sold в таблиці Sale\_details.
4. Функція CALCULATE змінює контекст розрахунків і обчислює суму витрат для відфільтрованих даних.

Міра для розрахунку середніх щомісячних витрат.

```
AverageMonthlyExpenses= AVERAGEX(
    VALUES('Calendar'[Month]);
    [MonthlyExpenses]
)
```

Вона працює таким чином:

1. Функція VALUES('Calendar'[Month]) визначає унікальні місяці в таблиці Calendar.
2. [MonthlyExpenses] є посиланням на вже створену міру.

3. Функція AVERAGEX обчислює середнє значення суми витрат.

Міра написана для обчислення середньої кількості проданих одиниць продукції.

```
AverageMonthlyUnitsProduced=AVERAGEX(
VALUES(Calendar[Month]);
CALCULATE(
SUM(Sale_details[quantity_sold]);
ALL(Products)
)
)
```

Міра працює наступним чином:

1. Функція VALUES(Calendar[Month]) визначає кількість унікальних місяців в таблиці Calendar.
2. Функція CALCULATE обчислює для кожного місяця загальну кількість проданих одиниць (SUM(Sale\_details[quantity\_sold])) в контексті, де ігнорується всі можливі фільтри таблиці Products.
3. Функція AVERAGEX, обчислює середнє значення сім проданих одиниць товару.

Функція, яка розраховує середні витрати на одиницю продукції.

```
AverageExpensePerUnit=CALCULATE(
DIVIDE(
[AverageMonthlyExpenses];
[AverageMonthlyUnitsProduced];
0
);
```

ALL(Products)

)

Функція працює наступним чином:

1. Функція DIVIDE обчислює частку від ділення середніх щомісячних витрат на середню кількість проданих одиниць товару.
2. Якщо знаменник буде дорівнювати 0, то функція поверне 0.
3. Функція CALCULATE, допомагає забезпечити, що всі розрахунки будуть виконанні без врахування фільтри з таблиці Products

Міра, яка розраховує загальні витрати на заготовки, враховуючи кількість інгредієнтів та їх ціну.

```
PreparationCost=SUMX(
    Technological_map_preparations;
    Technological_map_preparations[quantity] *
    LOOKUPVALUE(
        Supplier_prices[price];
        Supplier_prices[ingredient_id];
        Technological_map_preparations[ingredient_id]
    )
)
```

Функція працює наступним чином:

1. Функція SUMX ітерує через кожний рядок таблиці Technological\_map\_preparations.
2. Для кожного рядка обчислюється кількість інгредієнтів помножена на ціну інгредієнта.



3. LOOKUPVALUE шукає ціну для кожного інгредієнта в таблиці Supplier\_prices, на основі первинного ключа ingredient\_id.
4. Функція SUMX підсумовує попередні обчислення для кожного рядка.

Міра, для розрахунків загальної вартості продукту.

```
ProductCos = SUMX(
    Technological_map_products;
    Technological_map_products[quantity] *
    PreparationCost
)
```

Вона працює таким чином:

1. Функція SUMX ітерує через кожний рядок таблиці Technological\_map\_products.
  2. Перемножує кількість використаної заготовки на її вартість
  3. Функція SUMX підсумовує попередні обчислення для кожного рядка.
- Міра, яка розраховує витрати на одиницю продукції.

```
ExpensePerUnit=CALCULATE(
    DIVIDE(
        [MonthlyExpenses];
        [TotalUnitsProduced];
        0
    );
    ALL(Products)
)
```

Вона працює таким чином:

1. Функція `DIVIDE` обчислює частку від ділення загальних щомісячних витрат на загальну кількість проданої продукції в місяць.
2. Якщо знаменник дорівнює 0, то `DIVIDE` повертає 0.
3. Функція `CALCULATE`, забезпечує, що розрахунки були проведенні без врахування фільтрів для таблиці `Products`

Міра, яка рахує загальну вартість продуктів з врахуванням середніх витрат на одиницю продукції та витрат на підготовку інгредієнтів

$$\begin{aligned} \text{TotalProductCostWithExpenses} = & \text{SUMX}( \\ & \text{Products}, \\ & \text{Products}[\text{quantity}] * ( \\ & \quad [\text{AverageExpensePerUnit}] + [\text{ProductCost}] \\ & ) \\ & ) \end{aligned}$$

Вона працює таким чином:

1. Функція `SUMX` ітерує кожний рядок в таблиці `Products`.
2. Для кожного продукту, береться його кількість.
3. Для кожного продукту обраховується сума середніх витрат на одиницю продукції та собівартість продуктів, використаних на одиницю продукції.
4. Функція `SUMX` підсумовує всі обчисленні значення для кожного рядка таблиці `Products`.

Щоб спростити розуміння написаних мір, в таблиці 2.4 представлений їх короткий опис.

Функція, яка рахує загальний дохід від продажу продукції.

$$\begin{aligned} \text{TotalRevenue} = & \text{SUMX}( \\ & \text{Sale\_details}, \end{aligned}$$

$$\text{Sale\_details[quantity\_sold]} * \text{Sale\_details[price\_per\_unit]}$$

)

Функція, яка рахує прибуток на одиницю продукції.

$$\text{ProfitPerUnit} = \text{SUMX}(\text{Products}, \text{Products[price]} - [\text{TotalProductCostWithExpenses}])$$

Функція яка рахує загальну вартість проданих товарів

$$\text{TotalCOGS} = \text{SUMX}(\text{Products}; [\text{TotalUnitsSold}] * [\text{TotalProductCostWithExpenses}])$$

Функція, яка рахує валовий прибуток

$$\text{GrossMargin} = [\text{TotalRevenue}] - [\text{TotalCOGS}]$$

Функція, яка рахує відсоток валового прибутку.

$$\text{GrossMarginPercentage} = \text{DIVIDE}([\text{GrossMargin}], [\text{TotalRevenue}], 0)$$

Таблиця 2.4

### Короткий опис прописаних мір на мові DAX

Назва міри	Короткий опис міри
-1-	-2-
TotalUnitsSold	Загальна кількість проданих одиниць продукції.
MonthlyExpenses	Загальні щомісячні витрати за певний місяць у 2023 році
TotalUnitsProduced	Загальна кількість одиниць продукції, вироблених у певний місяць 2023 року
AverageMonthlyExpenses	Середні щомісячні витрати.
AverageMonthlyUnitsProduced	Середня кількість вироблених одиниць на місяць

## Продовження табл. 2.4

-1-	-2-
AverageExpensePerUnit	Середні витрати на одиницю продукції.
PreparationCost	Загальна вартість підготовки інгредієнтів.
ProductCost	Загальна вартість продукту з врахуванням підготовки інгредієнтів
ExpensePerUnit	Витрати на одиницю продукції
TotalProductCostWithExpenses	Загальна вартість продуктів з врахуванням середніх витрат на одиницю продукції та витрат на підготовку інгредієнтів.
TotalRevenue	Загальний прибуток
ProfitPerUnit	Прибуток на одиницю продукції
TotalCOGS	Загальна вартість проданих товарів
GrossMargin	Валовий прибуток
GrossMarginPercentage	Відсоток валового прибутку

На рисунку 2.27 зображений приклад використання мір у розрізі місяця, а на рисунку 2.28 застосування мір в розрізі назви продукту.

Month	Значення						
	TotalUnitsSold	TotalUnitsProduced	MonthlyExpenses	TotalRevenue	TotalCOGS	GrossMargin	GrossMarginPercentage
1	5648	5648	34304,89	163825	79658,15145	84166,84855	0,513760711
2	5374	5374	32614,37	153315	74312,3907	79002,6093	0,51529602
3	10495	10495	36029,74	292462	117015,5753	175446,4248	0,599894772
4	10266	10266	35948,11	279513	112143,8966	167369,1035	0,598788262
5	9899	9899	34255,97	276858	109739,6934	167118,3066	0,603624626
6	5589	5589	34371,06	160035	78438,36765	81596,63235	0,509867419
7	5987	5987	35974,78	176074	84458,51205	91615,48795	0,520323773
8	5608	5608	34037,87	158959	77439,19685	81519,80315	0,512835405
9	10262	10262	36065,34	283761	114035,7893	169725,2108	0,598127335
10	9785	9785	34369,83	267752	107723,9145	160028,0855	0,597672792
11	9619	9619	34395,15	270509	108470,242	162038,7581	0,599014295
12	6448	6448	37858,74	181435	87469,01275	93965,98725	0,517904413
<b>Загальний підсумок</b>	<b>94980</b>	<b>6448</b>	<b>37858,74</b>	<b>2664498</b>	<b>1150904,742</b>	<b>1513593,258</b>	<b>0,568059446</b>

Рисунок 2.27 – Зведена таблиця на основі побудованої OLAP-моделі, яка представляє роботу написаних мір в розрізі по місяцям

Позначка продукту	TotalUnitsSold	TotalUnitsProduced	ProductCost	ExpensePerUnit	AverageExpensePerUnit	TotalProductCostWithExpenses	ProfitPerUnit	TotalCOGS	GrossMargin	GrossMarginPercentage
Вівця	9732	615	4,6322	3,87139268	4,43634945	9,09483445	11,96518056	87627,27238	100712,7289	0,546259928
Партіюк з горосом	9789	636	3,8995	3,87139268	4,43634945	7,36431445	9,72668555	70529,18981	94521,86019	0,572887962
Партіюк з малуством	9648	688	4,26545	3,87139268	4,43634945	7,89951445	9,310188055	74181,30682	89624,69318	0,547838115
Партіюк з картоплею та грибами	9627	629	4,47445	3,87139268	4,43634945	7,89881445	9,101188055	76041,82378	87612,14222	0,535364031
Партіюк з картоплею та грибами	9536	653	3,9881	3,87139268	4,43634945	8,412483445	11,58788056	80221,23213	110488,7877	0,579179928
Партіюк з картоплею та печивом	9657	693	4,0281	3,87139268	4,43634945	8,453483445	11,54788056	81115,91246	113824,0875	0,577279928
Партіюк з рисом, айцям та зеленим	9580	628	4,47515	3,87139268	4,43634945	8,89951445	8,100488055	85253,21994	77662,68006	0,476499227
Печиво	7833	538	35,94	3,87139268	4,43634945	35,46438144	86,53568056	309124,3412	650825,6588	0,671180221
Салонок в піст	9687	654	34,88775	3,87139268	4,43634945	18,29211144	5,707888055	180741,1829	56431,85415	0,228113542
Чабуриг	9571	662	5,1502	3,87139268	4,43634945	9,77481445	15,22518056	93253,18889	146719,8811	0,606602542
<b>Загальний підсумок</b>	<b>94680</b>	<b>648</b>	<b>81,93865</b>	<b>3,87139268</b>	<b>4,43634945</b>	<b>128,1835044</b>	<b>173,8314356</b>	<b>1150894,742</b>	<b>1513593,258</b>	<b>0,56695948</b>

Рисунок 2.28 – Зведена таблиця на основі побудованої OLAP-моделі, яка представляє роботу написаних мір в розрізі по назві продукту

## 2.4 Створення візуалізації в Power BI

Щоб почати створювати візуалізації в Power BI, спочатку необхідно імпортувати нашу модель OLAP в Power BI Desktop. Для цього створюємо порожній звіт, заходимо на вкладку «Файл» і на вкладці «Імпортувати» обираємо необхідну кнопку, яка зображена на рисунку 2.29.

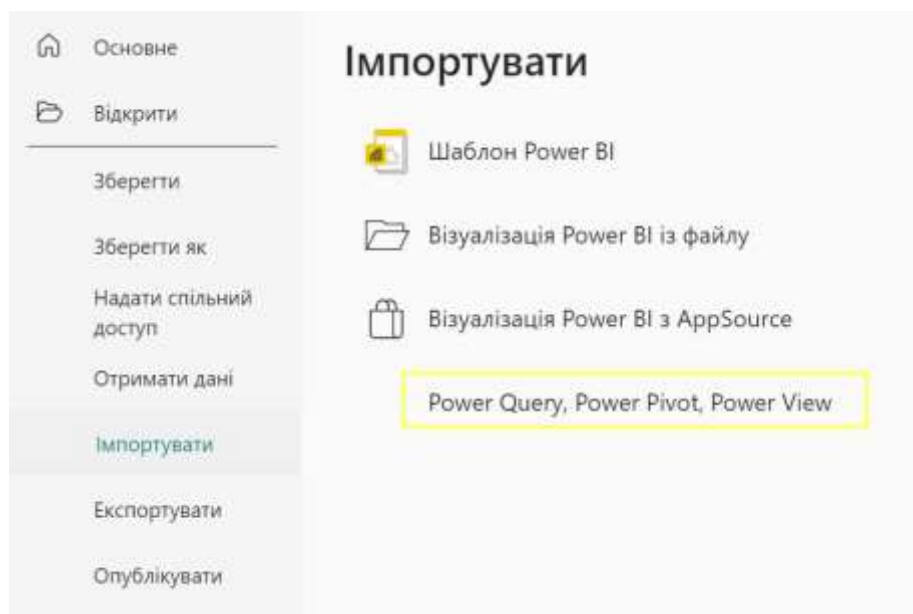


Рисунок 2.29 – Імпортування моделі даних Power Pivot в Power BI.

Всі наші таблиці імпортуються разом з написаними мірами та встановленими зв'язками (рис. 2.30).

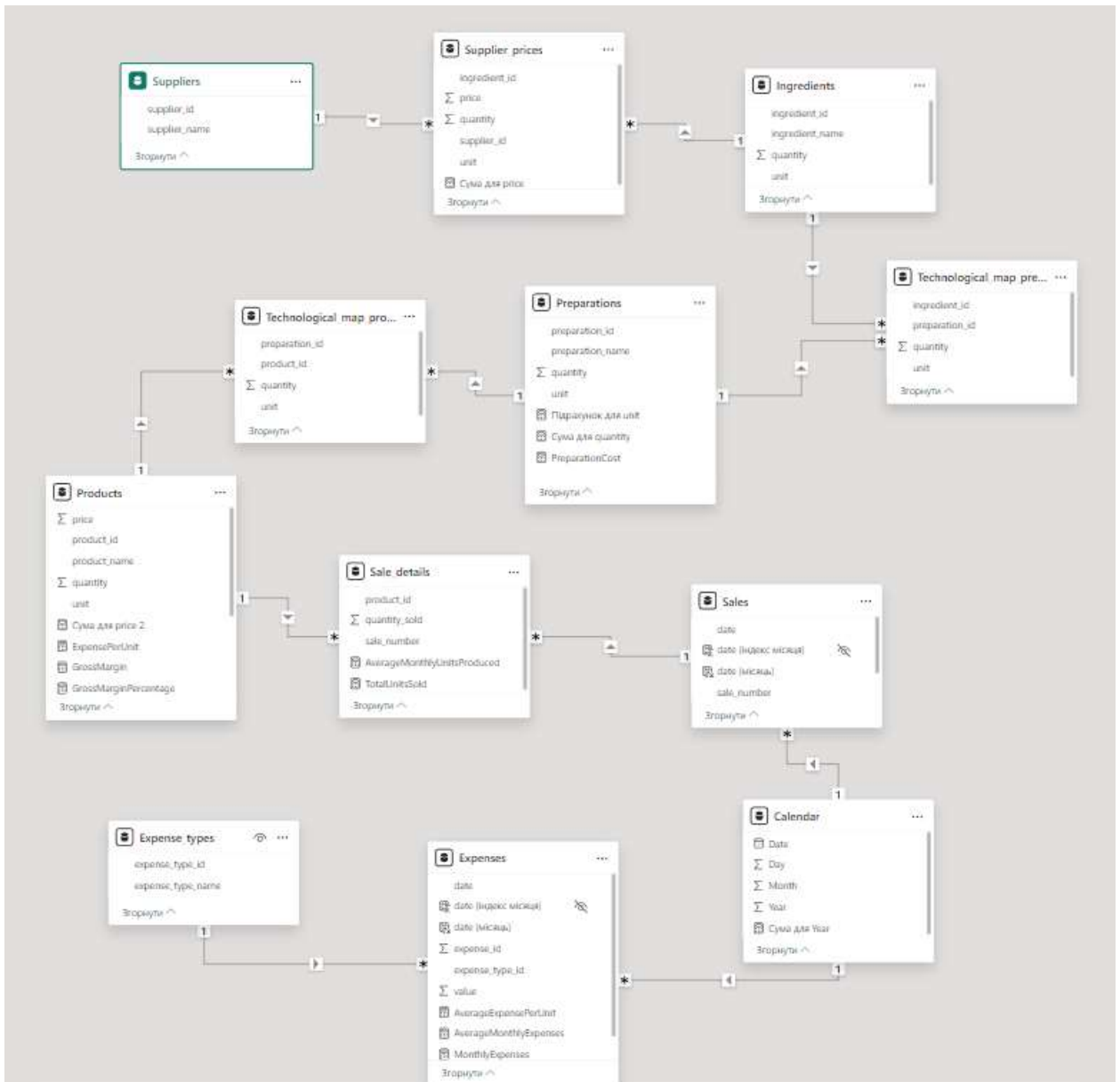


Рисунок 2.30 – Подання імпортованих даних в Power BI.

Тепер можемо приступити до візуалізації даних. Перше, що я вирішила додати на наш звіт, це «Картка». В неї я в поля перетягнула необхідну міру. Всього я створила 6 карток на першій візуалізації. На рисунку 2.31 можна подивитися відношення мір до значень в картках. Нагадаю, що всі міри описані в розділі 2.3.3.

Кількість проданих одиниць продукції допомагає бізнесу оцінити обсяги продажів.

Дохід – це важливий показник для розуміння фінансового здоров'я бізнесу та планування бюджету.

Кількість чеків дозволяє оцінити активність клієнтів та частоту покупок.

Прибуток важливий для аналізу рентабельності бізнесу та визначення ефективності витрат.

Середній чек допомагає в аналізі покупок клієнтів та виявлення можливостей для збільшення середнього чеку.

Маржа важливий показник для оцінки ефективності бізнес-стратегії та прийняття рішень щодо ціноутворення та витрат.



Рисунок 2.31 – Візуалізація типу «Картка» в Power BI: середній чек, маржа, прибуток, кількість чеків, дохід, продано одиниць продукту.

За допомогою «Лійки» візуалізую розподіл витрат по типам витрат (рис. 2.32). Ця візуалізація допомагає бізнесу детально аналізувати структуру своїх витрат, виявляти основні статті витрат та можливості для їх оптимізації. Вона дозволяє приймати більш обґрунтовані рішення щодо управління фінансами та зменшення витрат.



Рисунок 2.32 – Візуалізація типу «Лійка» в Power BI: розподіл витрат по категоріям.

Використовуючи візуалізацію типу «Таблиця» демонструю ціну, собівартість і прибуток по назвам товару (рис 2.33). Ця візуалізація допомагає бізнесу детально аналізувати ефективність кожного товару, ідентифікувати найбільш прибуткові позиції та оптимізувати асортимент продукції. Вона також дозволяє приймати обґрунтовані рішення щодо зміни цінової політики та стратегії управління витратами.

Назва товару	Ціна	Собівартість	Прибуток
Чебурек	25	9,77	15,23
Сосиска в тісті	25	19,29	5,71
Пончики	120	39,46	80,54
Пиріжок з рисом, яйцем та зеленню	17	8,90	8,10
Пиріжок з картоплею та печінкою	20	8,45	11,55
Пиріжок з картоплею та грибами	20	8,41	11,59
Пиріжок з картоплею	17	7,90	9,10
Пиріжок з капустою	17	7,69	9,31
Пиріжок з горохом	17	7,26	9,74
Біляш	20	9,03	10,97

Рисунок 2.33 – Візуалізація типу «Таблиця» в Power BI: вартість, собівартість та прибуток різних товарів.



За допомогою звичайної стовпчастої діаграми демонструю загальний дохід і маржу в розрізі назв товарів (рис 2.34). Ця візуалізація допомагає бізнесу отримати комплексне розуміння фінансових показників по асортименту товарів, приймати обґрунтовані рішення щодо оптимізації витрат, підвищення доходів і максимізації прибутковості.

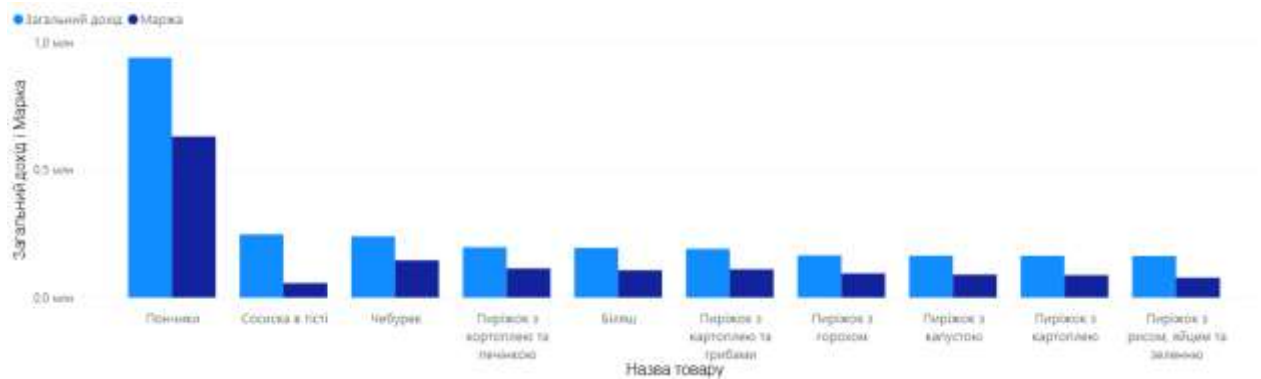


Рисунок 2.34 – Візуалізація типу "Стовпчаста діаграма" в Power BI: Загальний дохід і маржа по товарам.

На рисунку 2.35 представлений загальний вигляд звіту, на якому присутні всі зазначені вище візуалізації. Хочу відмітити, що на звіті присутній роздільник, за допомогою якого можна подивитися на дані в розрізі одного або декількох місяців (рис 2.36).



Рисунок 2.35 – Звіт в Power BI на якому представлені дані за 2023 рік по малому харчовому підприємстві в місті Кривий Ріг.



Рисунок 2.36 – Звіт в Power BI на якому представлені дані за березень 2023 року по малому харчовому підприємстві в місті Кривий Ріг.

Я створила ще один лист візуалізації, з фільтром по назві товару. На цьому звіті присутні картки з даними, які змінюються по назві продукції (рис. 2.37).



Рисунок 2.37 – Візуалізація типу «Роздільник» та «Картка» в Power BI: фільтрація маржі, прибутку, доходу та кількості проданих одиниць продукції за назвою продукції «Пончики».

На рисунку 2.38 представлена лінійчата діаграма, яка демонструє залежність суми витрат від місяця. Ця візуалізація є важливим інструментом для фінансового аналізу та управління витратами. Вона надає можливість детально розглянути динаміку витрат, виявити проблемні місця та розробити ефективні стратегії для зменшення витрат і підвищення фінансової стійкості бізнесу

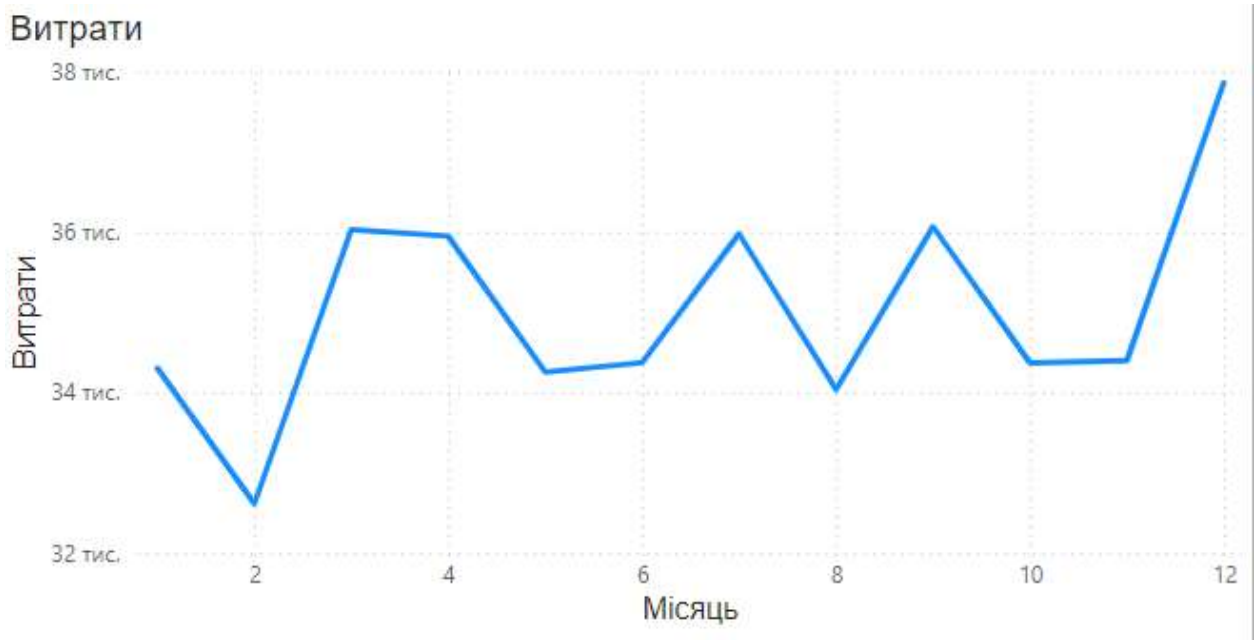


Рисунок 2.38 – Візуалізація типу «Лінійчата діаграма» в Power BI: залежність суми витрат від місяця.

За допомогою комбінованої діаграми я представила залежність кількості чеків та середнього чеку від місяця (рис 2.39). Ця візуалізація є важливим інструментом для моніторингу продажів та аналізу поведінки клієнтів. Вона надає комплексне розуміння ключових показників, що дозволяє приймати обґрунтовані рішення щодо підвищення ефективності бізнесу. А також загальної кількості проданих одиниць продукції та маржу від місяця (рис 2.40). Ця візуалізація є потужним інструментом для аналізу продажів та фінансових показників, що дозволяє отримати комплексне розуміння динаміки продажів і прибутковості продукції. Вона допомагає бізнесу приймати обґрунтовані рішення для підвищення ефективності та прибутковості.



Рисунок 2.39 – Візуалізація типу «Лінійчаста та звичайна стовпчаста діаграма» в Power BI: залежність кількості чеків та середнього чеку від місяця.



Рисунок 2.40 – Візуалізація типу «Лінійчаста та звичайна стовпчаста діаграма» в Power BI: залежність загальної кількості проданих одиниць продукції та маржи від місяця.

На рисунку 2.41 зображено загальний вигляд другої сторінки візуалізації з наведеними вище візуалізаціями.

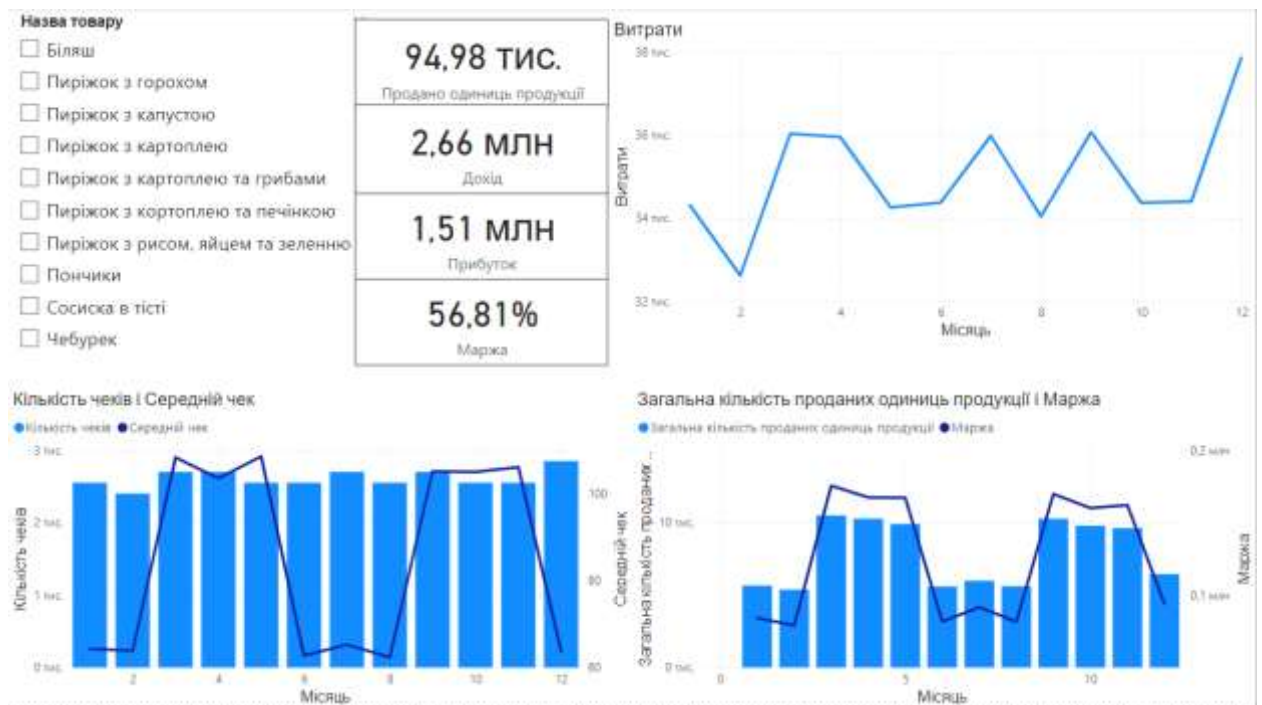


Рисунок 2.41 – Звіт в Power BI на якому представлені дані за 2023 рік по малому харчовому підприємстві в місті Кривий Ріг.

## 2.5 Побудова прогнозу на 2024 рік

### 2.5.1 Прогнозування доходу за допомогою моделі SARIMA

На рисунку 2.42 зображена сума доходів по місяцям за 2023 рік, це і буде наш вхідний часовий ряд, на якому ми будемо будувати прогноз.

Month	TotalCOGS
1	79658,1515
2	74312,3907
3	117015,575
4	112143,897
5	109739,693
6	78438,3677
7	84458,5121
8	77439,1969
9	114035,789
10	107723,915
11	108470,242
12	87469,0128
<b>Загальний підсумок</b>	<b>1150904,74</b>

Рисунок 2.42 – Вхідний часовий ряд для побудови прогнозу доходу на 2024 рік за моделлю прогнозування SARIMA.

Спочатку побудуємо лінійчату діаграму на наших даних, щоб подивитися, чи є сезонна залежність (рис 2.43).

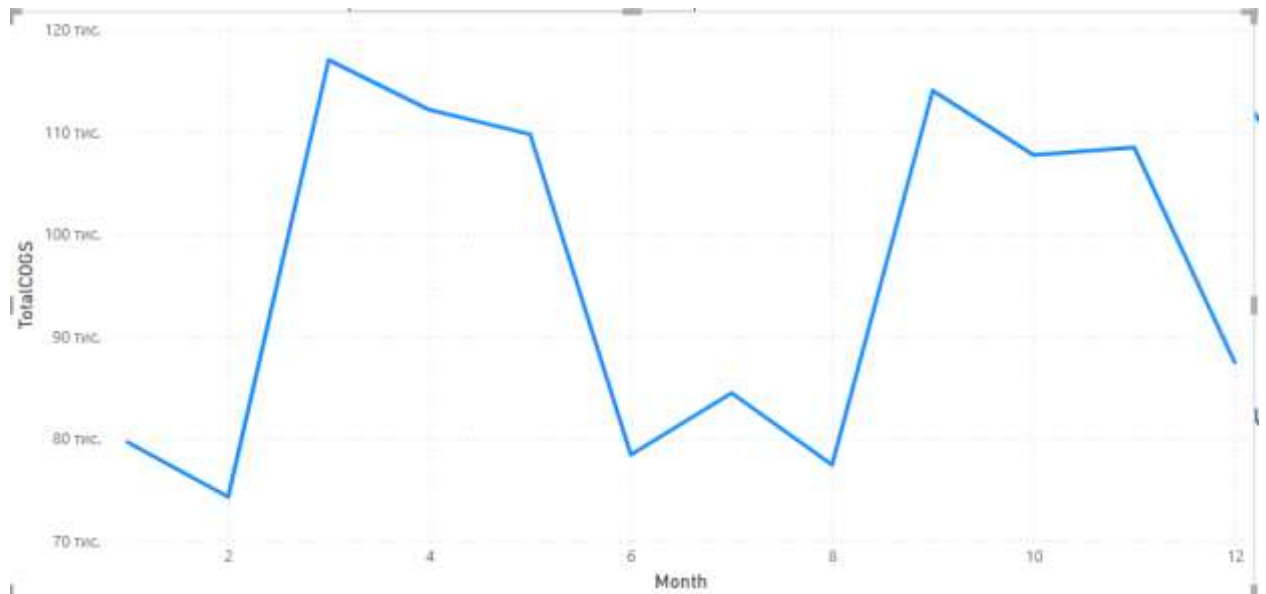


Рисунок 2.43 – Графічне представлення вхідних даних за допомогою лінійчатої діаграми.

Як ми можемо бачити з діаграми, то в нас доволі чітко відслідковується сезонна залежність. Влітку та взимку дохід менше ніж восени та навесні. Це обумовлено деякими факторами. Взимку холодно, та світає пізно. Харчове підприємство на базу якого я проводжу аналіз знаходиться на ринку і працює з шостої ранку до другої дня. Отже, людям холодно ходити на ринок, холодно стояти в черзі, а також продажі будуть набирати оберти під кінець робочого дня. Влітку навпаки, продажі будуть набирати оберти вранці, поки не так спекотно, і закінчуватися приблизно о 12, коли стає дуже спекотно. Також люди влітку будуть менше хотіти їсти, що також пояснюється погодними умовами.

Отже, враховуючи, що в нас є сезонна залежність будемо використовувати прогнозування за допомогою моделі SARIMA.

Прогнозування за допомогою цієї моделі є доволі складним, щоб реалізувати його вручну або за допомогою Excel, тому ми будемо використовувати мову програмування Python з бібліотекою pmdarima.

Ця бібліотека допоможе нам знайти саму функціональну та точну модель, методом підборів параметрів  $p, d, q, P, D, Q, s$ . Також ця бібліотека допоможе скоротити кількість коду та уникнути людських помилок.

Також я буду використовувати допоміжні бібліотеки. Бібліотеку `pandas` для роботи з даними у вигляді `DataFrame`, та бібліотеку `matplotlib.pyplot` для побудови графіку.

На першому кроці коду я імпортую необхідні бібліотеки, потім створюю словник, що містить два ключі 'Month' та 'Value'. З цього словника перетворюю дані в вигляд `DataFrame`. За допомогою `auto_arima` з бібліотеки `rmnarima` автоматично підбираю параметри для моделі SARIMA. Для цього прописую наступні параметри:

- `seasonal=True`: вказує, що модель враховує сезонність.
- `m=3`: вказує, що сезонний період складає 3 місяці.
- `stepwise=True`: використовує ступінчастий алгоритм пошуку для ефективного підбору параметрів.
- `trace=True`: виводить процес пошуку параметрів у консоль.
- `error_action='ignore'`: ігнорує помилки під час пошуку параметрів.
- `suppress_warnings=True`: пригнічує попередження для чистоти виводу.
- `method= powell`: метод Пауелла

Далі використовуючи цю ж бібліотеку роблю прогноз на наступні 12 місяців. І наприкінці будую графік з вхідних та вихідних даних.

Повний програмний код наведений в додатку Г.

Коли ми запустимо нашу програму, вона почне підбирати оптимальну модель для прогнозування і виводити нам дані про кожну модель (рис 2.44).



```

Performing stepwise search to minimize aic
ARIMA(2,0,2)(1,0,1)[3] intercept : AIC=inf, Time=0.45 sec
ARIMA(0,0,0)(0,0,0)[3] intercept : AIC=270.513, Time=0.02 sec
ARIMA(1,0,0)(1,0,0)[3] intercept : AIC=260.566, Time=0.36 sec
ARIMA(0,0,1)(0,0,1)[3] intercept : AIC=266.394, Time=0.12 sec
ARIMA(0,0,0)(0,0,0)[3] intercept : AIC=311.695, Time=0.02 sec
ARIMA(1,0,0)(0,0,0)[3] intercept : AIC=272.108, Time=0.04 sec
ARIMA(1,0,0)(2,0,0)[3] intercept : AIC=inf, Time=0.45 sec
ARIMA(1,0,0)(1,0,1)[3] intercept : AIC=254.951, Time=0.68 sec
ARIMA(1,0,0)(0,0,1)[3] intercept : AIC=266.548, Time=0.14 sec
ARIMA(1,0,0)(2,0,1)[3] intercept : AIC=inf, Time=0.82 sec
ARIMA(1,0,0)(1,0,2)[3] intercept : AIC=inf, Time=0.97 sec
ARIMA(1,0,0)(0,0,2)[3] intercept : AIC=263.972, Time=0.25 sec
ARIMA(1,0,0)(2,0,2)[3] intercept : AIC=inf, Time=1.08 sec
ARIMA(0,0,0)(1,0,1)[3] intercept : AIC=253.469, Time=0.41 sec
ARIMA(0,0,0)(0,0,1)[3] intercept : AIC=264.737, Time=0.09 sec
ARIMA(0,0,0)(1,0,0)[3] intercept : AIC=252.070, Time=0.36 sec
ARIMA(0,0,0)(2,0,0)[3] intercept : AIC=inf, Time=0.54 sec
ARIMA(0,0,0)(2,0,1)[3] intercept : AIC=inf, Time=0.76 sec
ARIMA(0,0,1)(1,0,0)[3] intercept : AIC=inf, Time=0.71 sec
ARIMA(1,0,1)(1,0,0)[3] intercept : AIC=inf, Time=0.89 sec
ARIMA(0,0,0)(1,0,0)[3] intercept : AIC=293.307, Time=0.04 sec

```

Рисунок 2.44 – Автоматичний підбір моделі SARIMA за допомогою бібліотеки `pyrmdarima` на мові Python.

Далі програма виводить детальний звіт про оцінку параметрів і статистику моделі (рис 2.45).

Model описує тип моделі, в нашому випадку це SARIMAX(1, 0, 0)(1, 0, 0)[3], що означає один лаг авторегресії, нульовий порядок інтеграції, нульовий порядок середньоквадратичної помилки, один сезонний лаг авторегресії, без сезонного диференціювання і без сезонної середньоквадратичної помилки на період сезонності 3.

No. Observations описує кількість спостережень, в нас це 12.

Log Likelihood показує логарифмічну правдоподібність моделі, що вказує на якість моделі.

AIC/BIC/HQIC вказують на значення критерія Акаїке, Байєсівського інформаційного критерія і Ганнан-Квінн інформаційного критерія відповідно. Ці значення використовуються для порівняння моделей, чим нижче значення, тим краще.

Intercept показує на коефіцієнт перехоплення з стандартною помилкою.

ar.S.L3 вказує на коефіцієнт сезонної авторегресії при лагу 3. Цей показник показує сильний негативний вплив.

sigma2 відповідає за оцінку дисперсії помилок і вказує на величину варіабельності.

Тест на випадковість залишків (Ljung-Box (Q)) показує значення вище за загальноприйнятий поріг 0,05. Саме через це ми не відкидаємо нульову гіпотезу про випадковість залишків.

Тест на нормальність залишків (Jarque-Bera (JB)) також не дає нам відкинути гіпотезу про нормальність.

Тест на варіабельність залишків (Heteroskedasticity (H)) не вказує на проблеми по частині варіабельності.

Отже, ми можемо зробити висновки, що модель показує гарні статистичні показники, показники випадковості, нормальності залишків, але має високі значення AIC/BIC. Це можна виправити шляхом додавання даних. Якби в нас були значення за 2 роки, то наш прогноз був би більш точним.

```

Best model: ARIMA(0,0,0)(1,0,0)[3] intercept
Total fit time: 9.228 seconds

                                SARIMAX Results
=====
Dep. Variable:                    y      No. Observations:                12
Model:                            SARIMAX(1, 0, 0, 3)  Log Likelihood                -123.035
Date:                            Wed, 12 Jun 2024    AIC                          252.070
Time:                            10:49:52         BIC                          253.525
Sample:                            01-31-2023      HQIC                         251.531
                                - 12-31-2023
Covariance Type:                  opg
=====
              coef      std err          z      P>|z|      [0.025      0.975]
-----
intercept    1.866e+05    7591.113     24.587     0.000     1.72e+05     2.02e+05
ar.S.L3      -0.9451             0.074    -12.778     0.000     -1.090     -0.800
sigma2       2.731e+07         0.039     7.02e+08     0.000     2.73e+07     2.73e+07
=====
Ljung-Box (L1) (Q):                1.37    Jarque-Bera (JB):                0.87
Prob(Q):                            0.24    Prob(JB):                        0.65
Heteroskedasticity (H):              0.81    Skew:                            0.19
Prob(H) (two-sided):                 0.84    Kurtosis:                        1.74
=====

```

Рисунок 2.45 – Статистичні показники обраної моделі SARIMA.

Після цього програма виводить прогнозовані значення (рис. 2.46) та креслить графік на якому зображені початкові та прогнозовані дані (рис. 2.47). По графіку ми бачимо, що прогнозовані значення зберегли сезонні коливання. Для більш якісного прогнозу нам необхідно більше даних

	Forecast
2024-01-31	84830.368948
2024-02-29	84124.994651
2024-03-31	103973.824177
2024-04-30	106467.677769
2024-05-31	107134.346087
2024-06-30	88374.680695
2024-07-31	86017.672244
2024-08-31	85387.585994
2024-09-30	103117.852918
2024-10-31	105345.525326
2024-11-30	105941.036875
2024-12-31	89183.682273

Рисунок 2.46 – Прогнозовані значення доходу на 2024 рік за допомогою програмного коду на мові Python.

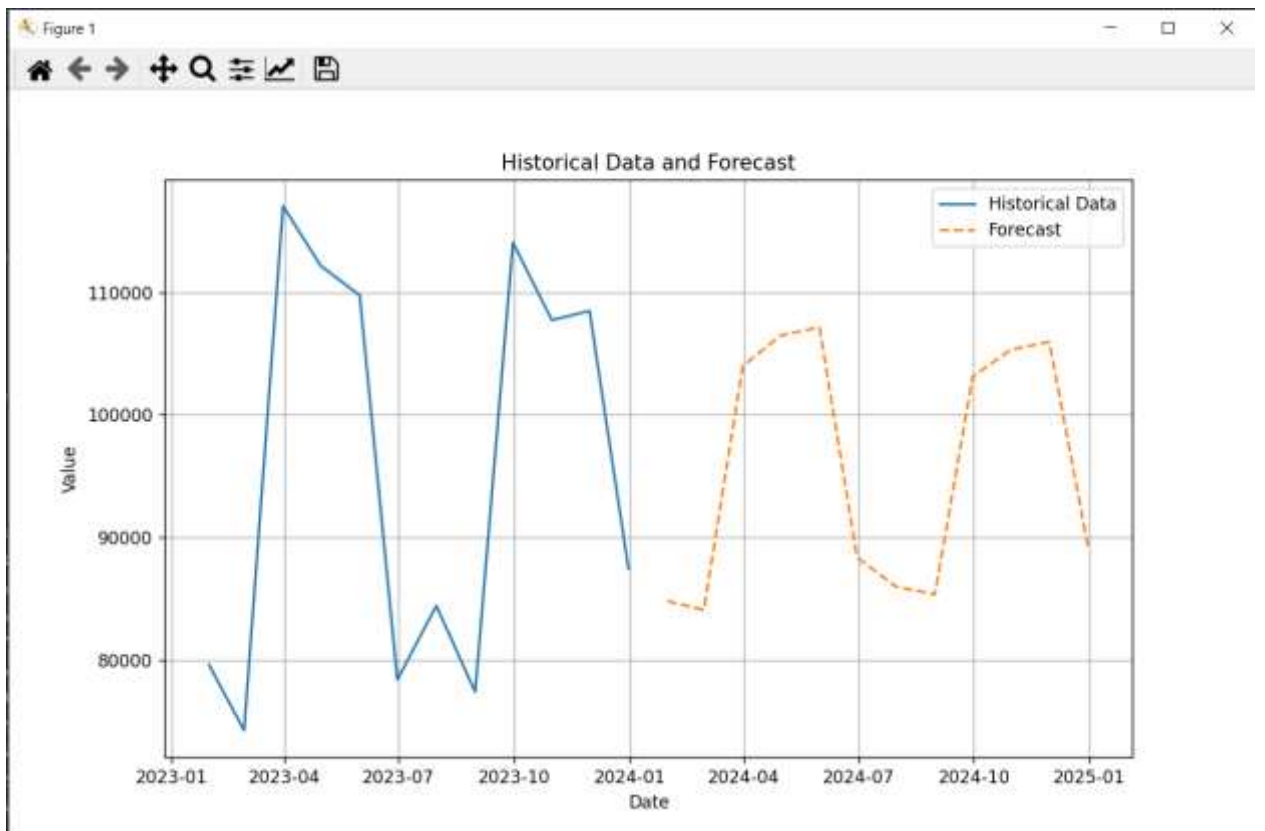


Рисунок 2.47 – Графік вхідних та вихідних даних після прогнозування даних моделлю SARIMA.

## 2.5.2 Прогноз кількості проданої продукції методом експоненціального згладжування

Початкові дані для прогнозування представлені на рисунку 2.48. Тут зображена кількість проданих одиниць товарів в кожному місяці. Ми можемо звернути увагу що в цих даних відслідковується сезонна залежність. Це гарно видно на рисунку 2.49. Тут ми можемо відслідковувати схожу залежність на ту, яку ми розглядали в розділі 2.5.1. Це не дивно, адже дохід напряду залежить від кількості проданої продукції, особливо, якщо ціна на продукцію не підіймалась, як в нашому випадку. Отже, й причини такої сезонності будуть ті самі.

Month	TotalUnitsProduced
1	5648
2	5374
3	10495
4	10266
5	9899
6	5589
7	5987
8	5608
9	10262
10	9785
11	9619
12	6448
<b>Усього</b>	<b>6448</b>

Рисунок 2.48 – Вхідні данні для прогнозування методом експоненціального згладжування

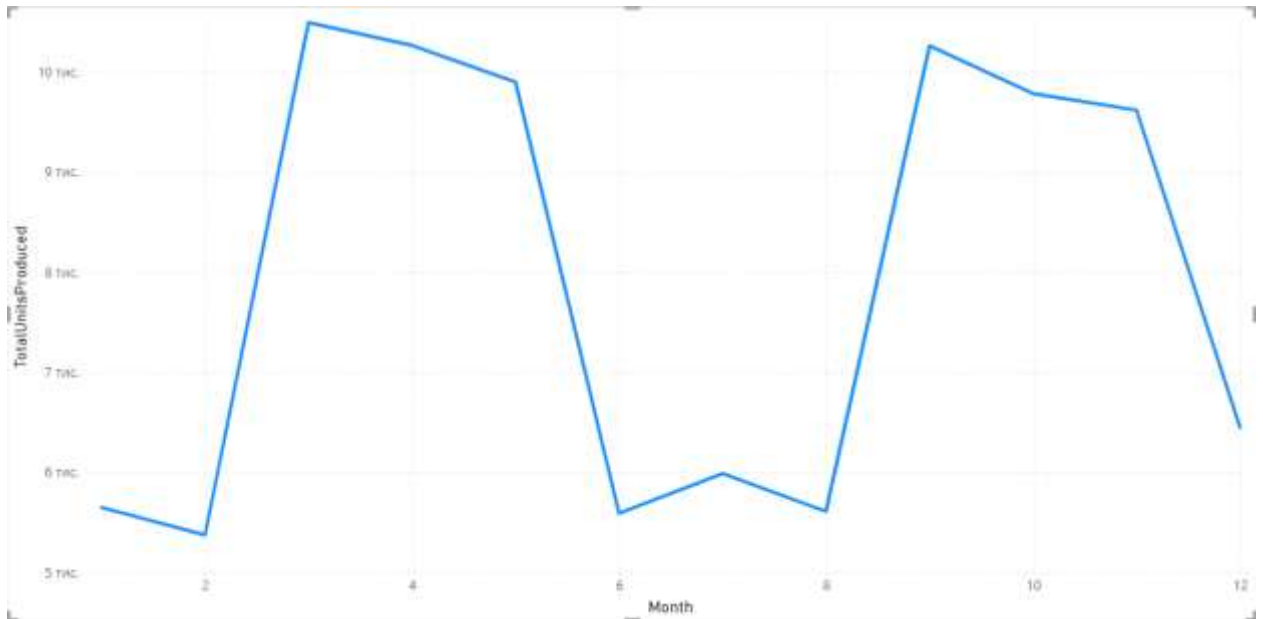


Рисунок 2.49 – Графічне представлення вхідних даних за допомогою лінійчатої діаграми.

Прогнозувати за методом експоненціального згладжування я буду в Microsoft Excel. Для такого випадку вже є готова функція FORECAST.ETS. Ця функція прогнозує майбутні значення, беручи за основу наявні. Використовуючи при цьому алгоритм експоненціального згладжування.

Створюємо таблицю та прописуємо формулу (рис 2.50). Спочатку я вказала значення сезонності 3, що дало не дуже гарний результат.

Дата	Вхідні значення	Прогнозованні значення
01.01.2023	5648	
01.02.2023	5374	
01.03.2023	10495	
01.04.2023	10266	
01.05.2023	9899	
01.06.2023	5589	
01.07.2023	5987	
01.08.2023	5608	
01.09.2023	10262	
01.10.2023	9785	
01.11.2023	9619	
01.12.2023	6448	
01.01.2024		8489,985323
01.02.2024		8286,940225
01.03.2024		8802,767074
01.04.2024		8810,139312
01.05.2024		8607,094214
01.06.2024		9122,921063
01.07.2024		9130,293301
01.08.2024		8927,248203
01.09.2024		9443,075052
01.10.2024		9450,44729
01.11.2024		9247,402193
01.12.2024		9763,229041

Рисунок 2.50 – Таблиця прогнозування кількості проданих одиниць товару в Excel методом експоненціального згладжування при значенні сезонності 3.

На рисунку 2.51, чітко видно, що сезонна динаміка не прослідковується на прогнозованих значеннях.

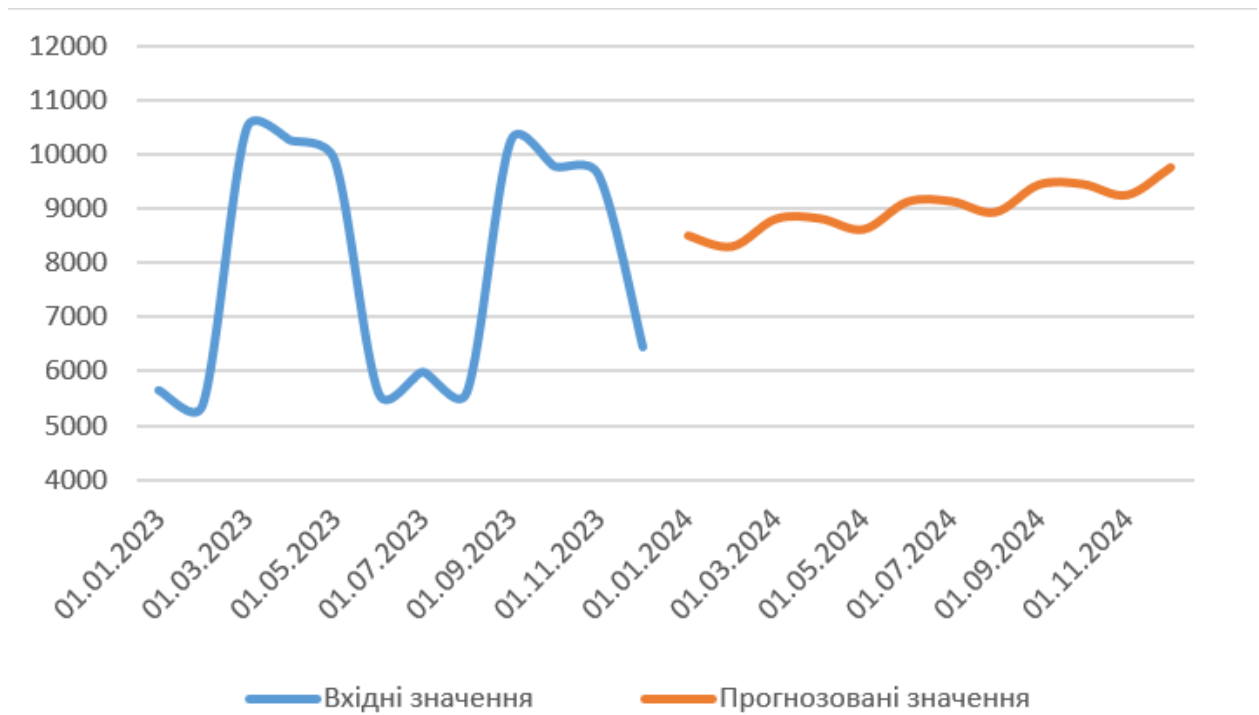


Рисунок 2.51 – Лінійчатая діаграма вхідних та вихідних значенні методом експоненціального згладжування при значенні сезонності 3.

Прогнозовані значення виглядають більш реально при значенні сезонності 6. Що видно на рисунках 2.52 та 2.53. В цьому прогнозі в нас зберігається сезонна залежність.



Дата	Вхідні значення	Прогнозовані значення
01.01.2023	5648	
01.02.2023	5374	
01.03.2023	10495	
01.04.2023	10266	
01.05.2023	9899	
01.06.2023	5589	
01.07.2023	5987	
01.08.2023	5608	
01.09.2023	10262	
01.10.2023	9785	
01.11.2023	9619	
01.12.2023	6448	
01.01.2024		6175,274791
01.02.2024		5799,448946
01.03.2024		10609,65259
01.04.2024		10270,72281
01.05.2024		10013,94762
01.06.2024		5866,520681
01.07.2024		6266,692432
01.08.2024		5890,866587
01.09.2024		10701,07023
01.10.2024		10362,14045
01.11.2024		10105,36526
01.12.2024		5957,938322

Рисунок 2.52 – Таблиця прогнозування кількості проданих одиниць товару в Excel методом експоненціального згладжування при значенні сезонності б.



Рисунок 2.53 – Лінійчата діаграма вхідних та вихідних значенні методом експоненціального згладжування при значенні сезонності 6.

## 1.6 Висновки до розділу 2

В другому розділі цієї дипломної роботи ми дослідили та проаналізували ключові аспекти організації бізнес-процесів на прикладі харчового виробництва. Ми оцінили важливість різних бізнес-процесів, використовуючи AllFusion Process Modeler для виявлення критичних аспектів управління якістю, контролю закупівель, обладнання, моніторингу попиту та інших. Також ми приділили увагу оцінці рівня організації кожного процесу.

Далі, з особливою увагою до нормалізації баз даних, ми проаналізували процеси організації даних для зменшення дублювання і підвищення інтегритету даних. Ми розробили структуру бази даних до третьої нормальної форми, зокрема, розділивши дані на кілька таблиць, що дозволило оптимізувати зберігання та доступ до даних.

Крім того, ми створили OLAP куб у Power Pivot, який дозволяє виконувати багатовимірний аналіз даних, надаючи можливості для глибокого

дослідження фінансових, логістичних та операційних показників. Використання DAX для розробки мір, які оцінюють загальні продажі, витрати, прибутковість, інші критичні для бізнесу метрики, забезпечує потужний інструмент для аналізу ефективності.

Ми створили ряд візуалізацій у Power BI, що дозволяють візуально оцінити поточний стан бізнесу. Ми розробили різні діаграми і зведені таблиці, які демонструють динаміку продажів, витрати і прибутковість за різними параметрами. Таке візуалізаційне представлення допомагає швидше приймати обґрунтовані рішення та адаптуватися до змін у ринкових умовах.

Також ми провели програзування даних на 2024 рік, що може допомогти бізнесу бачити свої майбутні показники та корегувати бізнес-процеси завчасно, щоб досягти бажаного результату.

Розділ підкреслює значення детального аналізу та візуалізації бізнес-процесів для оптимізації управління харчовим виробництвом. Завдяки використанню сучасних інструментів аналітики та моделювання даних, як OLAP і Power BI, ми значно підвищили можливості прогнозування та ефективного рішення стратегічних завдань. Наш підхід забезпечує комплексний погляд на бізнес-операції, що є критично важливим для підтримки стійкого розвитку та конкурентоспроможності підприємства.

## ВИСНОВКИ

Метою даної дипломної роботи було аналізування та автоматизація процесів малого харчового бізнесу з метою покращення рентабельності за допомогою створення та використання OLAP кубу. Це дослідження сприяло ефективному управлінню бізнесом шляхом забезпечення швидкого та точного аналізу даних, що є важливим для прийняття обґрунтованих управлінських рішень.

У роботі було проведено аналіз існуючих бізнес-процесів з використанням сучасних інструментів бізнес-аналітики, таких як AllFusion Process Modeler для моделювання процесів, а також використання Power BI та DAX для обробки даних та їх візуалізації. Особлива увага приділялася OLAP технологіям, які дозволяють виконувати складні аналітичні запити з високою швидкістю та ефективністю.

Результати дослідження показали, що впровадження OLAP кубу значно покращує можливості аналізу даних, особливо у складних сценаріях харчового виробництва, де потрібно враховувати багато факторів та оперативно реагувати на зміни умов бізнес-середовища. Використання DAX дозволило забезпечити гнучкість у створенні мір та аналітичних звітів, забезпечуючи глибоке розуміння внутрішніх процесів бізнесу.

Запропоновані методика та інструменти були застосовані для оптимізації роботи малого харчового бізнесу в місті Кривий Ріг. Розроблена OLAP-система встановлювала кореляції між різними даними (продажі, запаси, витрати) та допомагала у прийнятті рішень, що засновані на аналізі реального стану речей.

Впровадження OLAP кубів і використання бізнес-аналітики суттєво сприяють покращенню ефективності управління в харчовому виробництві. Дослідження підкреслило значення систематичного аналізу та автоматизації

процесів, які допомагають у підвищенні прибутковості та забезпеченні стійкості бізнесу в умовах непередбачуваного ринкового середовища. Отримані дані та аналітичні звіти мають велике значення для стратегічного планування та розвитку підприємства.

Дане дослідження показало ефективність використання OLAP кубів для аналізу даних у харчовому виробництві. Ця робота підкреслює значення стабільності та надійності використаних методів в умовах специфічного застосування.

Практичне значення роботи полягає у підтвердженні можливості адаптації відомих рішень до специфічних умов малого харчового виробництва. Використання OLAP кубів допомагає підприємствам ефективно управляти виробничими даними, оптимізувати запаси та планувати виробництво на основі детального аналізу.

Наукове значення роботи полягає в демонстрації того, як стандартні аналітичні інструменти можуть бути адаптовані для специфічних потреб малого бізнесу. Результати можуть бути корисними для подальших досліджень в області бізнес-аналітики, допомагаючи в розробці більш гнучких аналітичних систем, що враховують особливості різних галузей.

На основі отриманих результатів можна зробити припущення, що подальший розвиток аналітичних систем для харчового виробництва буде зосереджений на підвищенні гнучкості та індивідуалізації аналітичних рішень. Важливою буде інтеграція з новітніми технологіями обробки даних і штучного інтелекту для автоматизації прийняття рішень

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Miller, J.A. The Magical Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information // Psychological Review. 1956. Vol. 63, No. 2. Pp. 81-97.
2. Jelen, B. Microsoft Excel Pivot Table Data Crunching (Office 2021 and Microsoft 365). 1st ed. Microsoft Press, 2021. 544 p.
3. Tufte, E.R. The Visual Display of Quantitative Information. – Cheshire, Connecticut: Graphics Press, 2001. – 197 p.
4. Russo, M., Ferrari, A. The Definitive Guide to DAX: Business Intelligence with Microsoft Excel, SQL Server Analysis Services, and Power BI (2nd Edition). – Redmond, Washington: Microsoft Press, 2019. – 768 p.
5. Функція SUMX (DAX) - DAX | Microsoft Learn. URL: <https://learn.microsoft.com/en-us/dax/sumx-function-dax> (дата звернення: 01.06.2024).
6. Функція ALL (DAX) - DAX | Microsoft Learn. URL: <https://learn.microsoft.com/en-us/dax/all-function-dax> (дата звернення: 01.06.2024).
7. Функція DIVIDE (DAX) - DAX | Microsoft Learn. URL: <https://learn.microsoft.com/en-us/dax/divide-function-dax> (дата звернення: 01.06.2024).
8. Функція CALCULATE (DAX) - DAX | Microsoft Learn. URL: <https://learn.microsoft.com/en-us/dax/calculate-function-dax> (дата звернення: 01.06.2024).
9. Функція LOOKUPVALUE (DAX) - DAX | Microsoft Learn. URL: <https://learn.microsoft.com/en-us/dax/lookupvalue-function-dax> (дата звернення: 01.06.2024).

10. Функція VALUES (DAX) - DAX | Microsoft Learn. URL: <https://learn.microsoft.com/en-us/dax/values-function-dax> (дата звернення: 01.06.2024).
11. Функція AVERAGE (DAX) - DAX | Microsoft Learn. URL: <https://learn.microsoft.com/en-us/dax/average-function-dax> (дата звернення: 01.06.2024).
12. Функція FILTER (DAX) - DAX | Microsoft Learn. URL: <https://learn.microsoft.com/en-us/dax/filter-function-dax> (дата звернення: 01.06.2024).
13. Функція SUM (DAX) - DAX | Microsoft Learn. URL: <https://learn.microsoft.com/en-us/dax/sum-function-dax> (дата звернення: 01.06.2024).
14. Функція MAX (DAX) - DAX | Microsoft Learn. URL: <https://learn.microsoft.com/en-us/dax/max-function-dax> (дата звернення: 01.06.2024).
15. Peixeiro, M. Time Series Forecasting in Python. Shelter Island: Manning Publications Co., 2022. – 416 p.
16. Ostertagová, E., Ostertag, O. (2011). The Simple Exponential Smoothing Model. In: Proceedings of the 4th International Conference on Modelling of Mechanical and Mechatronic Systems, September 20-22, 2011, Herľany, Slovakia. Technical University of Košice, pp. 380-384.
17. Neves, M. M., & Cordeiro, C. (2010). Exponential Smoothing and Resampling Techniques in Time Series Prediction. *Discussiones Mathematicae, Probability and Statistics*, 30, pp. 87-101.
18. Bogatova, Mariia. Improving Recruitment, Selection and Retention of Employees. Case: Dpointgroup Ltd. – South-Eastern Finland University of Applied Sciences, 2017. – 86 p
19. Kool, V.A. OLAP Cubes: Enabling Business Intelligence in Case Management Tools: Master's Thesis. Master, "Computer Science". Radboud University Nijmegen, 2012. – 85 p.

### Додаток А. Відомість матеріалів кваліфікаційної роботи

№ з/п	Позначення				Найменування	Кількість аркушів	Примітки		
1									
2					Документація				
3									
4	САУ.КР.24.10.ПЗ				Пояснювальна записка	105	Формат А4		
5									
6					Демонстраційний матеріал	№2	Презентація на CD-R		
7									
8					Копія роботи	1	Диск CD-R		
9									
10									
11									
12									
13									
14									
15									
16									
17									
18									
					САУ.КР.24.10.ДА.ПЗ.				
Змін.	Аркуш	№ докум.	Підпис	Дата					
Розроб.		Красіліч			<b>Матеріали кваліфікаційної роботи</b>	Літ.	Аркуш	Аркушів	
К. розд.		Хом'як							
Керівн.		Хом'як							
Н.контр.		Хом'як							
Зав. каф.		Хом'як							
						НТУ «ДП», 12; 124-20-1			



## Додаток Б. Відгук керівника кваліфікаційної роботи

### Відгук на кваліфікаційну роботу бакалавра студентки групи 124 – 20 – 1 Красіліч Вікторії Олексіївни спеціальності 124 Системний аналіз

Тема кваліфікаційної роботи: «Аналіз та автоматизація діяльності з використанням OLAP кубу в умовах харчового виробництва»

Обсяг кваліфікаційної роботи 105 стор.

Мета кваліфікаційної роботи: аналіз та автоматизація діяльності малого харчового бізнесу з використанням OLAP кубу для покращення рентабельності та ефективності бізнесу.

Актуальність теми: може значно збільшити ефективність управління виробничими процесами та посилити конкурентні позиції на ринку.

Тема кваліфікаційної роботи безпосередньо пов'язана з об'єктом діяльності бакалавра спеціальності 124 Системний аналіз, оскільки сформована модель бізнес-процесів, побудована база даних на основі якої створений OLAP куб, створена візуалізація даних, побудований прогноз значень.

Виконані в кваліфікаційній роботі завдання відповідають вимогам ступеня бакалавра. Оригінальність наукових рішень полягає в використанні OLAP технологій в малому бізнесі

Практичне значення результатів кваліфікаційної роботи полягає в можливості використання розроблених методів для покращення управління виробництвом та збільшення його ефективності.

Висновки підтверджують можливість використання результатів роботи в практичній діяльності для оптимізації процесів та підвищення продуктивності виробництва. Оформлення пояснювальної записки та демонстраційного матеріалу до неї виконано згідно з вимогами. Роботу виконано самостійно, відповідно до завдання та у повному обсязі

У роботі відзначено такі недоліки: недостатня кількість даних для проведення якісного прогнозу

Кваліфікаційна робота в цілому заслуговує оцінки: 95 – «відмінно»

З урахуванням висловлених зауважень автор заслуговує присвоєння освітньої кваліфікації «бакалавр з системного аналізу».

Керівник кваліфікаційної роботи бакалавра

К. ф.-м. н., доц.,

доцент кафедри системного аналізу та управління

\_\_\_\_\_ / Хом'як Т.В.

## Додаток Г. Програмний код

```

import pandas as pd

from pmdarima import auto_arima

import matplotlib.pyplot as plt

# Підготовка даних

data = {

    'Month': pd.date_range(start='2023-01-01', periods=12, freq='ME'), # Використання 'ME' для кінця місяця

    'Value':
[79658.15145,74312.3907,117015.5753,112143.8966,109739.6934,78438.36765,84458.51205,77439.19685,114035.7893,10772
3.9145,108470.242,87469.01275]}

df = pd.DataFrame(data)

df.set_index('Month', inplace=True)

# Автоматичний підбір параметрів моделі SARIMA з квартальною сезонністю

model = auto_arima(df['Value'], seasonal=True, m=3, stepwise=True, trace=True, error_action='ignore',
suppress_warnings=True, method='powell')

# Виведення вибраних параметрів

print(model.summary())

# Прогнозування на наступний рік з урахуванням сезонності

forecast = model.predict(n_periods=12)

# Створення DataFrame для прогнозованих значень

forecast_index = pd.date_range(start='2024-01-01', periods=12, freq='ME') # Також вказуємо 'ME' для кінця місяця

forecast_df = pd.DataFrame(forecast, index=forecast_index, columns=['Forecast'])

# Вивід прогнозованих значень

print(forecast_df)

# Візуалізація даних

plt.figure(figsize=(10, 6))

plt.plot(df.index.to_numpy(), df['Value'].to_numpy(), label='Historical Data')

plt.plot(forecast_index.to_numpy(), forecast_df['Forecast'].to_numpy(), label='Forecast', linestyle='--')

plt.title('Historical Data and Forecast')

plt.xlabel('Date')

plt.ylabel('Value')

plt.legend()

plt.grid(True)

plt.show()

```