

НУБІП України

ЗАТВЕРДЖУЮ

Завідувач кафедри

комп'ютерних наук

НУБІП України

/ Голуб Б.Л., доцент, к.т.н. /

підпис

« »

20 р.

ЗАВДАННЯ**ДО ВИКОНАННЯ МАГІСТЕРСЬКОЇ РОБОТИ СТУДЕНТУ****ПОНЗЕЛЮ ЯРОСЛАВУ ЮРІЙОВИЧУ**

НУБІП України

Спеціальність 121 – «Інженерія програмного забезпечення»

Освітня програма «Програмне забезпечення інформаційних систем»

Орієнтація освітньої програми освітньо-професійна

1. Тема магістерської роботи: Підсистема адміністрування системи підтримки прийняття рішень з управління шкільним харчуванням затверджена наказом ректора НУБІП України від 29.10.2020 № 4636 "С"

2. Термін подання завершеної роботи на кафедру

2021

рік, місяць, число

3. Вихідні дані до магістерської роботи: замовлення страв

4. Перелік питань, що підлягають дослідженню:

1. Визначення критеріїв аналізу даних.
2. Дослідження алгоритмів аналізу та технологій роботи з даними.
3. Визначення підходів, методів та інструментів для автоматизації аналізу.
4. Проектування сховища даних.
5. Дослідження роботи обраних технологій.

Дата видачі завдання

2020 . 10 . 29
рік, місяць, число

Керівник магістерської роботи

підпис

/ Голуб Б.Л. /

ініціали та прізвище

Завдання прийняв до виконання

підпис

/ Понзель Я.Ю. /

ініціали та прізвище

НУБІП України

ЗМІСТ		
ВСТУП		4
1 СИСТЕМНИЙ АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ		6
1.1 Аналіз предметної області		6
1.2 Технічне завдання		8
2 МОДЕЛЮВАННЯ ПРЕДМЕТНОЇ ОБЛАСТІ		10
2.1 Загальні положення.		10
2.2 Діаграма прецедентів.		11
2.3 Архітектура системи		19
3 ОПИС ТА ПРОЕКТУВАННЯ СИСТЕМИ		22
3.1 Логічна модель Бази Даних		22
3.2 Опис вузлів системи, які поставляють дані по сховищу		25
3.3 Публікація бази даних		26
3.4 Проектування сховища даних		27
3.4.1 Загальні поняття в напрямку OLAP технології		27
3.4.2 Поняття сховища даних		29
3.5 Постачання даних до гіперкубу		34
3.5.1 Побудова розгорнутого кубу		34
3.5.2 Заповнення кубу даними		41
4 АНАЛІЗ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ		46
4.1 Технології, алгоритми та методи аналізу даних		46
4.2 Використання OLAP-технологій для аналізу даних		56
4.2.1 Вибір інструментарію для аналізу		56
4.2.2 Результати аналізу даних на основі звітів		57
4.2.3 Розрахунок KPI		62
ВИСНОВКИ		67
СПИСОК ВИКОНАНИХ ДЖЕРЕЛ		69
ДОДАТОК А		72
ДОДАТОК Б		74

НУБІП УКРАЇНИ

ВСТУП

На сьогодні гостро стає питання стосовно дитячого харчування в шкільних закладах. Саме від нього залежить здоров'я дитини. Багато батьків

незадоволені харчуванням дітей у школі. По-перше, рідко коли додержуються вимоги санітарних норм в шкільних їдальнях, по-друге, у шкільній їдальні немає індивідуального підходу до здорового харчування кожної дитини з

врахуванням норм БЖУ та її здоров'я.

Незалежно від типу і форми власності закладу освіти планування та організація харчування дітей повинно здійснюватися згідно до Конституції України та законів України «Про освіту», «Про повну загальну середню освіту» та «Про основні принципи та вимоги до безпечності та якості харчових продуктів».

Процес цей організовується заради забезпечення учнів енергією та поживними речовинами для різностороннього розвитку, зміцнення здоров'я та покращення успіхів у навчанні. Необхідно враховувати калорійність їжі, яка потрібна для певної вікової групи хлопчиків або дівчат. При плануванні меню

враховувати раціон поживних речовин та вітамінів, які необхідні для певної пори року, особливості національної кухні, можливі релігійні обмеження, алергійні протипоказання та індивідуальні смакові переваги школярів.

Вирішення цих, а також не менш важливих інших, питань бере на себе «Система підтримки прийняття рішень з управління шкільним харчуванням».

Дана робота дає змогу чітко розпланувати шкільне меню для дітей, щоб забезпечити їх необхідними поживними речовинами протягом всього дня.

Система підходить для функціонування на національному рівні, так як дає можливість складати індивідуальне шкільне меню для кожного окремого освітнього закладу. В системі присутня функція замовлення, яка реалізує

змогу вибору індивідуального замовлення меню для дитини, яке буде враховувати його індивідуальні вподобання, нівелювати вміст в продуктах алергічних речовин та добавляти додаткові продукти до меню школяра.

Метою даної магістерської роботи є аналіз замовлення страв шкільного харчування за певними критеріями для подальшої оптимізації шкільного меню.

Об'єктом дослідження являється процес планування та реалізації харчування для учнів різного типу закладів освіти.

Предмет дослідження – це підсистема аналізу замовлення страв шкільного харчування.

Під час виконання магістерської роботи були використані такі засоби: мова програмування SQL, система управління базами даних MS SQL Server, середовище BI MS SQL Server.

Для реалізації поставленої мети визначено такі основні завдання:

- 1) проаналізувати предметну область;
- 2) змодельовати підсистему;
- 3) розробити підсистему адміністрування системи підтримки прийняття рішень з управління шкільним харчуванням.

Дана магістерська робота складається з 78 сторінок. Вона включає 4 розділи.

Апробація результатів дослідження:

- XI міжнародна науково-практична конференція молодих вчених “Інформаційні технології: економіка, техніка, освіта” - публікація “Підсистема адміністрування системи підтримки прийняття рішень з управління шкільним харчуванням”.
- IV Всеукраїнська науково-практична інтернет-конференція студентів та аспірантів “Теоретичні та прикладні аспекти розробки комп'ютерних систем” - публікація “Система підтримки прийняття рішень з управління шкільним харчуванням”.

1 СИСТЕМНИЙ АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Аналіз предметної області

«Системний аналіз – це методологія теорії систем, що полягає в дослідженні будь-яких об'єктів, що представляються в якості систем, проведенні їх структуризації і подальшого аналізу. Головна особливість системного аналізу полягає в тому, що він включає в себе не тільки методи аналізу (від грец. analysis - розчленування об'єкта на елементи), а й методи синтезу (від грец. synthesis - з'єднання елементів в єдине ціле). Головна мета системного аналізу – виявити й усунути невизначеність при вирішенні складної проблеми на основі пошуку найкращого рішення з існуючих альтернатив.»[1]

Згідно з постанови Кабінету Міністрів України від 24 березня 2021 р. № 305 «Про затвердження норм та Порядку організації харчування у закладах освіти та дитячих закладах оздоровлення та відпочинку» виділені наступні норми:

«1. Енергетична та поживна цінність їжі (вміст макро- і мікронутрієнтів) у закладах дошкільної, загальної середньої освіти та інших закладах освіти, що провадять освітню діяльність на певному рівні (рівнях) повної загальної середньої освіти, дитячих закладах оздоровлення та відпочинку (далі - заклади) повинна відповідати загальним віковим потребам здобувачів освіти/дітей згідно з нормами фізіологічних потреб в основних харчових речовинах та енергії, визначеними МОЗ.

2. Планування частки калорійності їжі у загальних енергетичних потребах залежить від режиму (кратності) харчування в закладі.

3. У разі споживання лише сніданку калорійність повинна становити 25-30 відсотків добової потреби, відповідна частка для обіду збільшується до 30-

35 відсотків. Орієнтовна кількість калорій на сніданок, обід та вечерю, добова калорійність для різних вікових груп наведена в таблиці 1.1.»[2]

Таблиця 1.1

Вікова група	Енергетична цінність сніданку, вечері, ккал	Енергетична цінність обіду, ккал	Добова енергетична цінність, ккал
1-4 роки	350-415	415-485	1385
4-6 (7) років	425-510	510-595	1700
6*-11 років	525-630	630-735	2100
11-14 років	600-720	720-840	2400
14-18 років	675-810	810-945	2700

Також, разом з тим згідно з Міністерством Охорони Здоров'я України за наказом від 03.09.2017 №1073 Про затвердження Норм фізіологічних потреб населення України в основних харчових речовинах і енергії, встановлена наступна добова норма дитячого населення в білках, жирах, вуглеводах та енергії яка наведена в таблиці 1.2.[3]

Таблиця 1.2

Вікова група	Стать	Енергія	Білки, г		Жири, г	Вуглеводи, г
		ккал	загальна кількість	Тваринні		
6 років (учні)	хлопчики та дівчатка	1800	60	43	58	260
7-10 років	хлопчики та дівчатка	2100	72	51	70	295
11-13 років	хлопчики	2400	84	62	84	327

Продовження табл. 1.2

11-13 років	дівчатка	2300	78	55	76	326
14-17 років	юнаки	2700	93	68	92	375
14-17 років	дівчата	2400	83	59	81	334

1.2 Технічне завдання

Планування закупівель продовольчої сировини та харчових продуктів для приготування страв або для асортименту шкільних буфетів слід проводити відповідно до вимог Санітарного регламенту для закладів загальної середньої освіти.

Закупівля плануються з огляду на сезонність доступності продовольства, рецептури та необхідність ротації страв у двотижневому меню з певною частотою їх включення.

Керівники закладів освіти несуть відповідальність за відповідність страв у меню вимогам додатку 9 до Санітарного регламенту для закладів загальної середньої освіти, зокрема, щодо особливостей кулінарної обробки продуктів, вмісту солі, цукру та жиру у стравах.

При плануванні меню слід враховувати традиції української національної кухні, сучасні тенденції міжнародної кулінарії, можливі релігійні обмеження у споживанні певних продуктів та страв. Примірні меню мають включати різноманітні та смачні страви, надавати можливість учням пробувати нові варіанти корисної їжі.

Ротаційний цикл для страв у примірному меню має бути достатньо довгим для забезпечення різноманітності та варіабельності. Шість або вісім тижнів можна вважати достатнім періодом для належної ротації з метою запобігання частій появі у меню одних і тих самих страв.

Оптимальне планування харчування та варіабельності примірних меню повинно включати весь навчальний рік. Така річна модель планування дозволяє враховувати сезонну доступність продуктів, святкові особливості меню, а також координувати навчальні заходи, присвячені харчуванню.

Основним завданням магістерської роботи було розробити підсистему адміністрування системи підтримки прийняття рішень з управління шкільним харчуванням в залежності від різних факторів впливу.

Для вирішення проблеми системи аналізу даної предметної області необхідно розробити СД та за допомогою BI MS SQL Server дослідити залежності та вирахувати KPI.

Сховище даних має базуватися на основі бази даних, створеної на основі комунікації зі студентами факультету харчових технологій.

Сховище даних має містити інформацію про:

- позиції продуктів;
- фактори впливу;
- час.

Це мають бути виміри сховища. Таблиця фактів «Замовлення» має містити дату замовлення, назву продукту, назву фактору впливу та кількість.

Сховище даних має мати можливість надавати всю потрібну інформацію для аналізу даних, зокрема, для відповіді на питання:

- які продукти найчастіше замовляються користувачами?
- які фактори найчастіше впливають на реалізацію певного товару або меню за певний період часу в певному регіоні?

2 МОДЕЛЮВАННЯ ПРЕДМЕТНОЇ ОБЛАСТІ

2.1 Загальні положення.

Більшість методів, що існують, об'єктно-орієнтованого аналізу і проектування (ООАП) включають мову моделювання і опис процесу моделювання. Мова моделювання — це нотация (переважно графічна), яка використовується методом для опису проектів.

Уніфікована мова моделювання (UML) є стандартним інструментом створення "креслень" програмного забезпечення. За допомогою UML можна візуалізувати, специфікувати, конструювати та документувати артефакти програмних систем.

UML підходить для моделювання будь-яких систем: від інформаційних систем масштабу підприємства до розподілених Web-додатків і навіть вбудованих систем реального часу. Це дуже виразна мова, що дозволяє розглянути систему з усіх точок зору, що стосуються її розробки та подальшого розгортання.

UML — це мова для візуалізації, специфікації, конструювання та документування артефактів програмних систем.

Мова складається зі словника і правил, що дозволяють комбінувати слова, що входять до нього, і отримувати осмислені конструкції. У мові моделювання словник та правила орієнтовані на концептуальне та фізичне уявлення системи. Мова моделювання, подібна до UML, є стандартним засобом для складання "креслень" програмного забезпечення.

Моделювання необхідне для розуміння системи. При цьому єдиної моделі ніколи не буває достатньо. Навпаки, для розуміння будь-якої нетривіальної системи доводиться розробляти велику кількість взаємозалежних моделей. У застосуванні до програмних систем це означає, що потрібна мова, за допомогою якої можна з різних точок зору описати уявлення архітектури системи протягом циклу її розробки.

Словник і правила такої мови, як UML, пояснюють, як створювати та читати добре певні моделі, але нічого не повідомляють про те, які моделі та в яких випадках потрібно створювати. Це завдання процесу розробки

програмного забезпечення. Добре організований процес має підказати, які потрібні артефакти, які ресурси необхідні для їх створення, як можна використовувати ці артефакти, щоб оцінити виконану роботу та керувати проектом загалом. [4]

Переваги UML:

- UML об'єктно-орієнтований, внаслідок чого методи опису результатів аналізу та проектування семантично близькі до методів програмування на сучасних об'єктно-орієнтованих мовах;

- UML дозволяє описати систему практично з усіх можливих точок зору та різні аспекти поведінки системи;

- Діаграми UML порівняно прості для читання після досить швидкого ознайомлення з його синтаксисом;

- UML розширює та дозволяє вводити власні текстові та графічні стереотипи, що сприяє його застосуванню не лише у сфері програмної інженерії;

- UML набув широкого поширення та динамічно розвивається.

2.2 Діаграма прецедентів.

Візуальне моделювання в UML можна як деякий процес порівневого спуску від найбільш загальної і абстрактної концептуальної моделі вихідної системи до логічної, а потім і до фізичної моделі відповідної програмної системи. Для досягнення цих цілей спочатку будується модель у формі так

званої діаграми прецедентів (use case diagram), яка описує функціональне

призначення системи або, іншими словами, те, що система робить у процесі свого функціонування. Діаграма прецедентів є вихідним концептуальним

уявленням чи концептуальною моделлю системи у процесі її проектування та розробки.

Розробка діаграми прецедентів має на меті:

- Визначити загальні межі та контекст модельованої предметної області на початкових етапах проектування системи.
- Сформулювати загальні вимоги до функціональної поведінки проектованої системи.
- Розробити вихідну концептуальну модель системи для її подальшої деталізації у формі логічних та фізичних моделей.
- Підготувати вихідну документацію для взаємодії розробників системи з її замовниками та користувачами.

Суть цієї діаграми полягає в наступному: проектована система представляється у вигляді безлічі сутностей або акторів, що взаємодіють із системою за допомогою так званих прецедентів.

Актором або дійовою особою називається будь-яка сутність, що взаємодіє із системою ззовні. Це може бути людина, технічний пристрій, програма або будь-яка інша система, яка може бути джерелом впливу на систему, що моделюється так, як визначить сам розробник.

Прецедент служить для опису сервісів, які система надає акторові. Іншими словами, кожен прецедент визначає деякий набір дій, який здійснюється системою при діалозі з актором. При цьому нічого не йдеться про те, яким чином буде реалізовано взаємодію акторів із системою.

У загальному випадку, діаграма прецедентів є графом спеціального вигляду, який є графічною нотацією для представлення конкретних прецедентів, акторів, можливо деяких інтерфейсів, і відносин між цими елементами. При цьому окремі компоненти діаграми можуть бути укладені у прямокутник, який позначає проектовану систему загалом. Слід зазначити, що відносинами даного графа можуть бути лише деякі фіксовані типи

взаємозв'язків між акторами та прецедентами, які в сукупності описують сервіси або функціональні вимоги до системи, що моделюється.

Головне призначення діаграми прецедентів полягає у формалізації функціональних вимог до системи за допомогою понять відповідного пакета та можливості узгодження отриманої моделі із замовником на ранній стадії проектування. Будь-який з прецедентів може бути піданий подальшій декомпозиції на безліч прецедентів для окремих елементів, які утворюють вихідну сутність.[5]

У мові UML є кілька стандартних видів відношень між акторами і варіантами використання:

- асоціації (association relationship)
- включення (include relationship)
- розширення (extend relationship)
- узагальнення (generalization relationship)

При цьому загальні властивості варіантів використання можуть бути представлені трьома різними способами, а саме — за допомогою відношень включення, розширення і узагальнення.[6]

Відношення асоціації — використовується при побудові всіх графічних моделей систем у формі канонічних діаграм. Стосовно до діаграм варіантів використання асоціація служить для позначення специфічної ролі актора при його взаємодії з окремим варіантом використання. Інакше кажучи, асоціація специфікує семантичні особливості взаємодії акторів і варіантів використання в графічній моделі системи.

У контексті діаграми варіантів використання відношення асоціації між актором і варіантом використання може вказувати на те, що актор ініціює відповідний варіант використання. Такого актора називають головним. В інших випадках подібна асоціація може вказувати на актора, якому надається довідкова інформація про результати функціонування системи, що моделюється. Таких акторів часто називають другорядними.

Відношення включення (include) – це різновид відносин залежності між базовим варіантом використання і його спеціальним випадком. При цьому відношенням залежності є таке відношення між двома елементами моделі, при якому зміна одного елемента (незалежного) приводить до зміни іншого елемента (залежного).

Відношення включення встановлюється тільки між двома варіантами використання й вказує на те, що задана поведінка для одного варіанта використання включається як складений фрагмент у послідовність поведінки іншого варіанта використання. Дане відношення є спрямованим бінарним відношенням у тому розумінні, що пари екземплярів варіантів використання завжди впорядковані відносно включення.

Відношення розширення (extend) визначає взаємозв'язок базового варіанта використання з іншим варіантом використання, функціональне поводження якого задіється базовим не завжди, а тільки при виконанні додаткових умов.

Відношення розширення є залежністю, спрямованою до базового варіанта використання й з'єднаною з ним у так названій крапці розширення.

Відношення розширення між варіантами використання позначаються як відношення залежності у формі пунктирної лінії зі стрілкою, спрямованою від того варіанта використання, що є розширенням для базового варіанта використання. Дана лінія зі стрілкою повинна бути позначена стереотипом `<<extend>>`.

Відношення узагальнення – визначає зв'язок, коли два й більше актори мають загальні властивості, тобто взаємодіють з тією самою множиною варіантів використання однаковим чином. Така спільність властивостей і поведінки представляється у вигляді відношення узагальнення з іншим, можливо, абстрактним актором, що моделює відповідну сукупність ролей.

Графічне відношення узагальнення позначаються суцільною лінією зі стрілкою у формі не зафарбованого трикутника, що вказує на батьківський варіант використання [7].

На рис. 1 зображено відношення між акторами та прецедентами в системі. На цій діаграмі виділені три актори: керівник-технолог, адміністратор, аналітик. Виділені прецеденти дозволяють специфікувати поведінку розробленої системи та отримати відповідь на запитання, що має робити система. В таблицях 1.3 та 1.4 детально описано суть актора для системи та дії акторів на прецеденти.

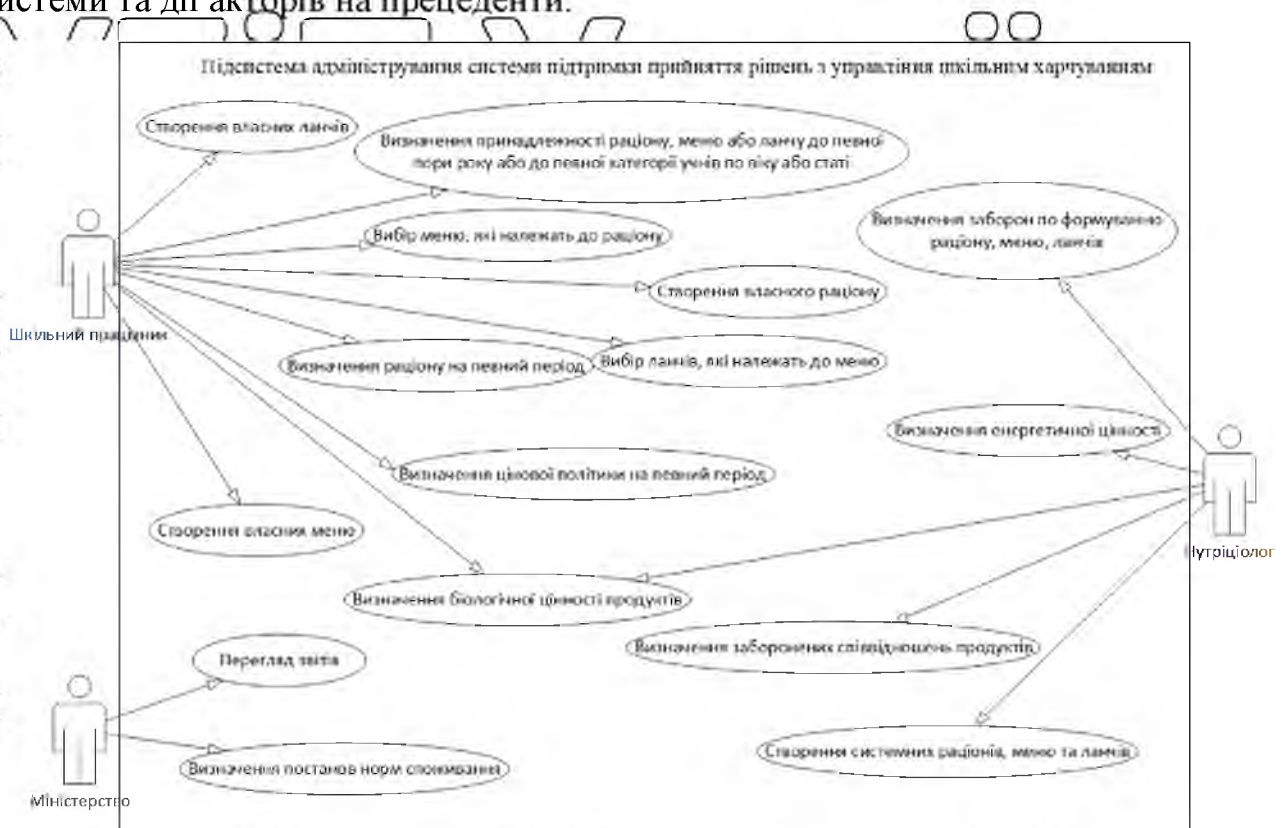


Рис. 1 Діаграма прецедентів

НУБІП України

Визначення акторів

Таблиця 1.3

Актор	Короткий опис
Шкільний працівник	Це людина, яка керує плануванням меню шкільного закладу
Нутріціолог	Це людина, яка встановлює взаємозв'язок між харчуванням та здоров'ям людини. Вона займається питаннями гігієни харчування
Міністерство	Це інститут, який встановлює норми шкільного та дитячого харчування

Виявлення дій акторів на прецеденті

Таблиця 1.4

Основний актор	Назва	Формулювання
Міністерство	Визначення постанов норм споживання	Міністерство має визначити та встановити норми споживання для шкільних закладів відповідно до вимог харчування дітей
Міністерство	Перегляд звітів	Міністерство переглядає звіти по школах на дотримання вимог встановлених меню

Продовження табл. 1.4

Нутріціолог	Визначення заборон по формуванню раціону, меню, ланчів	Нутріціолог має встановити заборони стосовно харчування дітей опираючись на встановлені норми харчування міністерством
Нутріціолог	Визначення енергетичної цінності	Нутріціолог має визначити енергетичну цінність раціону, меню та ланчів
Нутріціолог	Визначення біологічної цінності продуктів	Нутріціолог має визначити засвоюваність продуктів організмом та їх амінокислоти білків
Нутріціолог	Визначення заборонених співвідношень продуктів	Нутріціолог має визначити які продукти не можна поєднувати в раціоні, меню та ланчах
Нутріціолог	Створення системних раціонів, меню та ланчів	Нутріціолог має на основі всіх визначених вище прецедентів створити стандартизовані раціони, меню та ланчі, які можуть використовувати школи

Продовження табл. 1.4

Шкільний працівник	Створення власних ланчів	Шкільний працівник створює на основі встановлених норм міністерством та нутріціологом ланчі для школи
Шкільний працівник	Визначення приналежності раціону, меню або ланчу до певної пори року або до певної категорії учнів по віку або статі	Шкільний працівник визначає який раціон, меню або ланчу підходить для певної пори року, категорії учнів або віку та статі.
Шкільний працівник	Вибір меню, які належать до раціону	Шкільний працівник має визначитись з меню шкільного закладу з вже встановлених меню раніше нутріціологом
Шкільний працівник	Створення власного раціону	Шкільний працівник може створити власний раціон з набору продуктів та визначати період їх вживання
Шкільний працівник	Вибір ланчів, які належать до меню	Шкільний працівник обирає ланчі з вже встановлених раніше нутріціологом

Продовження табл. 1.4

Шкільний працівник	Визначення раціону на певний період	Визначається раціон харчування для дітей на певний період, наприклад, семестр або чверть
Шкільний працівник	Визначення цінової політики на певний період	Шкільним працівником встановлюються ціни на позиції в меню на певний час, наприклад, чверть
Шкільний працівник	Визначення біологічної цінності продуктів	Шкільний працівник має самостійно визначити біологічну цінність тих продуктів, які є в їх раціоні шкільного меню, які не встановлені раніше нутріціологом
Шкільний працівник	Створення власних меню	Шкільний працівник може встановити власне меню, якщо їх не задовольняє або не підходить повністю те меню, яке раніше було визначене нутріціологом
2.3 Архітектура системи		

Рівень розвитку сучасних технологій настільки високий, що дозволяє побудувати інформаційну систему будь-якого масштабу, складності й функціональності. Однак, з огляду на вимоги бізнесу, засновані на показниках різних бізнес-оцінок, виникають додаткові складнощі, вирішення яких зводиться до забезпечення раціонального підходу до

процесу проектування, реалізації й подальшій експлуатації інформаційних систем. Виходячи із цього, можна однозначно вважати обрану архітектуру одним з основних показників ефективності створюваної інформаційної системи, а, отже, і успішності бізнесу. [8]. Саме діаграма розгортання зазвичай зображує архітектуру системи.

Діаграма розгортання — діаграма, на якій відображаються обчислювальні вузли під час роботи програми, компоненти, та об'єкти, що виконуються на цих вузлах. Компоненти відповідають представленню робочих екземплярів одиниць коду. Компоненти, що не мають представлення під час роботи програми на таких діаграмах не відображаються; натомість, їх можна відобразити на діаграмах компонентів. Діаграма розгортання відображає робочі екземпляри компонентів, а діаграма компонентів, натомість, відображає зв'язки між типами компонентів.

Діаграма розгортання в UML моделює фізичне розгортання артефактів на вузлах.

Вузли представляються, як прямокутні паралелепіпеди з артефактами, розташованими в них, зображеними у вигляді прямокутників. Вузли можуть мати підвузли, які представляються, як вкладені прямокутні паралелепіпеди. Один вузол діаграми розгортання може концептуально представляти безліч фізичних вузлів, таких, як кластер серверів баз даних.

Існує два типи вузлів:

- Вузол пристрою;
- Вузол середовища виконання.

Вузли пристроїв — це фізичні обчислювальні ресурси зі своєю пам'яттю і сервісами для виконання програмного забезпечення, такі як звичайні ПК, мобільні телефони.

Вузол середовища виконання — це програмний обчислювальний ресурс, який працює всередині зовнішнього вузда і який надає собою сервіс, який виконує інші виконувани програмні елементи. [9]

Отже, загальну архітектуру системи можна побачити на рис. 2

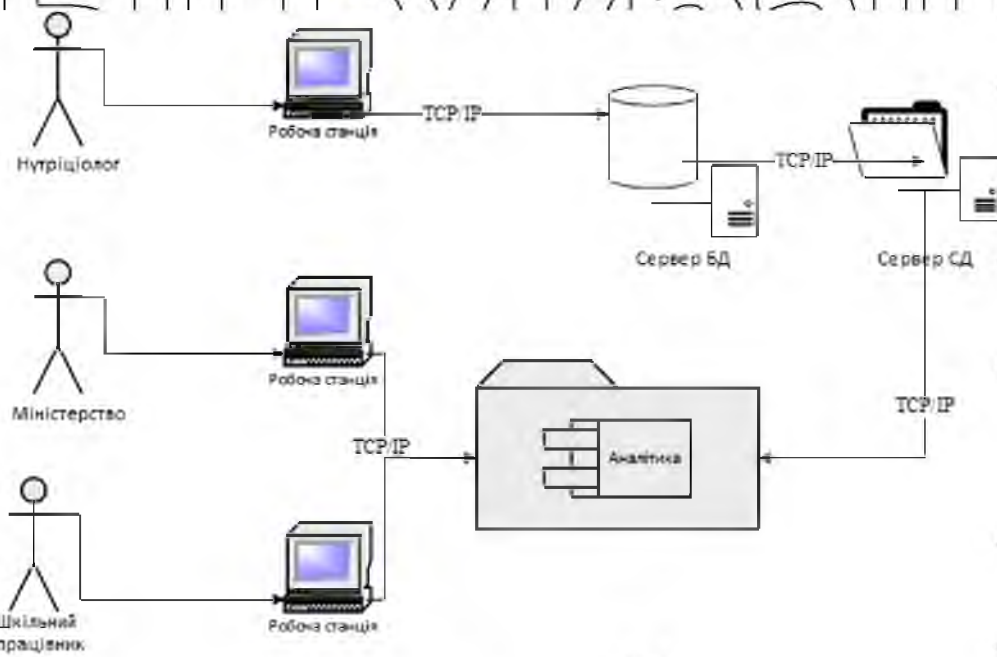


Рис. 2 Топологія системи

На діаграмі виділено три користувача:

- Шкільний працівник – взаємодіє з модулем аналітики, формує рішення про створення меню, ланчів та раціонів.

- Міністерство – розробка постанов стосовно дитячого харчування та перегляд звітів.

- Нутріціолог – взаємодіє з БД та наповнює систему даними про меню, ланчі та раціони.

Окрім користувачів, на діаграмі виділено підсистему аналізу, базу даних та сховище даних. В подальших підрозділах буде розібрано та досліджено ці компоненти.

3 ОНІС ТА ПРОЕКТУВАННЯ СИСТЕМИ

2.1 Логічна модель Бази Даних

Під оптимальною логічною моделлю баз даних розуміють модель, яка не має аномалій, пов'язаних з модифікацією БД, тобто проблем, що можуть виникнути у зв'язку із замінами, вставками і вилученнями даних із БД.

Для створення такої моделі баз даних незалежно від того, яка СУБД використовується — ієрархічна, сіткова чи реляційна — застосовується теорія нормалізації реляційних баз даних. Використання реляційного підходу дає змогу спроектувати оптимальну логічну модель БД, яка потім досить просто трансформується в ієрархічну чи сіткову модель.

В основу реляційних моделей покладено поняття відношення, яке подають у вигляді двовимірної таблиці.

Реляційна БД — це набір взаємопов'язаних відношень. Кожне відношення (таблиця) в ЕОМ подається як файл. Відношення можна поділити на два класи: об'єктні і зв'язкові.

Об'єктні відношення зберігають дані про інформаційні об'єкти предметної області. Наприклад: КЛІЄНТ (код клієнта, назва клієнта, адреса, телефон) є об'єктним відношенням.

В об'єктному відношенні один з атрибутів однозначно ідентифікує окремий об'єкт. Такий атрибут називається первинним ключем відношення.

У наведеному відношенні роль ключа виконує атрибут «код клієнта».

Ключ може вміщувати кілька атрибутів, тобто бути складеним. В об'єктному відношенні не повинно бути рядків з однаковим ключем, тобто не допускається дублювання об'єктів. Це основне обмеження реляційної моделі для забезпечення цілісності даних.

Зв'язкове відношення зберігає первинні ключі двох або більше об'єктних відношень. Ключі зв'язкового відношення мають на меті встановлення зв'язків між об'єктними відношеннями.

Розглянемо, наприклад, ще одне об'єктне відношення БАНК (код банку, назва банку, адреса банку).

Тоді зв'язкове відношення БАНК-КЛІЄНТ (код банку, код клієнта) буде сполучним між двома об'єктними відношеннями БАНК і КЛІЄНТ. У

зв'язковому відношенні можуть дублюватися ключові атрибути. Крім ключів, за якими встановлюють зв'язок у зв'язковому відношенні, можуть бути ще й інші атрибути, які функціонально залежать від цього складового ключа.

Ключі в зв'язкових відношеннях називаються вторинними або зовнішніми ключами, оскільки вони є первинними ключами об'єктів інших відношень. Реляційна модель накладає на зовнішні ключі обмеження, яке називають посилковою цілісністю. Воно необхідне для забезпечення цілісності даних.[10]

Нормалізація відношень — це ітераційний зворотний процес декомпозиції початкового відношення на кілька простіших відношень меншої розмірності. Під зворотністю процесу розуміють те, що операція об'єднання відношень, здобутих у результаті декомпозиції, має дати початкове відношення. У результаті нормалізації склад атрибутів відношень

БД має відповідати та-ким вимогам:

- між атрибутами мають виключатися небажані функціональні залежності;
- групування атрибутів не повинно мати збиткового дублювання даних;
- забезпечувати обробку і поновлення атрибутів без ускладнень.

Апарат нормалізації був розроблений американським вченим Е.Ф.

Коддом. Кожна нормальна форма обмежує тип допустимих залежностей між атрибутами. Кодд виділив три нормальні форми (скорочена назва 1НФ, 2НФ

і 3НФ). Найдосконаліша з них — це 3НФ. Тепер уже відомі і визначені 4НФ, 5НФ.

Відношення в 1НФ мають відповідати таким вимогам:

- усі атрибути відношення мають бути атомарними, тобто

не-подільними;

- усі рядки таблиці мають бути однакової структури, тобто мати одну й ту саму кількість атрибутів з іменами, що відповідно збігаються;

- імена стовпців мають бути різними, а значення однорідними

(мати однаковий формат);

- порядок рядків у таблиці неістотний.

Відношення знаходиться в другій нормальній формі (2НФ), якщо воно знаходиться в першій нормальній формі (1НФ) і немає неключових атрибутів, залежних від частини складного ключа.

Відношення знаходиться в третій нормальній формі (3НФ), якщо воно знаходиться в 2НФ і всі неключові атрибути взаємно незалежні.

Кожне відношення БД містить як структурну, так і семантичну інформацію. Структурна інформація задається схемою відношення, а

семантична виражає функціональні зв'язки між атрибутами. [11]

Логічна модель є основою бази даних, вона повинна відображати взаємозв'язки між реляційними таблицями. Між реляційними таблицями

можуть бути наступні типи зв'язків 1:1, 1:Б та Б:Б. Найбільш поширеним

зв'язком є зв'язок 1:Б. Зв'язок 1:1 зустрігається рідше, тому що дані між якими існує такий тип зв'язку в переважній більшості випадків входять до складу однієї реляційної таблиці. Зв'язок Б:Б безпосередньо не

підтримується в реляційних СУБД. Для реалізації такого зв'язку необхідно

створювати додаткову реляційну таблицю, яка буде відігравати роль зв'язкової. Зв'язкова таблиця має обов'язково містити первинні ключі таблиць, між якими встановлюється зв'язок.

Зв'язок встановлюється між полем первинного ключа об'єктного відношення та полем вторинного ключа зв'язкового відношення. Зв'язок між полем первинного та полем вторинного ключа встановлюється, якщо поля задані одним типом і подані в од-ному форматі. [12]

Erwin Data Modeler (раніше назва була стилізована ERwin) – є комп'ютерним програмним забезпеченням для моделювання даних.

Механізм роботи програмного забезпечення заснований на методології системного представлення даних IDEF1X, хоча тепер він також підтримує інші методології, включаючи просторове моделювання. Програма дозволяє будувати діаграми, зручні для читання та аналізу, що відображають потоки даних та керування ними в будь-якій системі (не обов'язково в цілому інформаційною). Умовно це можна представити як взаємодія сутностей і зв'язків, де сутності – це різні процеси у системі, а зв'язки – це потоки даних, інформації, документів, розпоряджень та іншого. Кожна сутність може бути декомпозована та бути представлена, в свою чергу, як сукупність дрібніших і найшвидших процесів та обміну інформацією, необхідного реалізації.

Erwin знаходить найбільш актуальне застосування у сфері бізнес-аналітики, для детального вивчення роботи підприємства та оптимізації його діяльності, але в цілому може бути застосований практично у всіх сферах життя для багатьох цілей, аж до менеджменту особистого часу. [13]

В додатку А представлено логічну модель бази даних в ЗНФ

2.2 Опис вузлів системи, які поставляють дані по сховищу

Для реалізації інформаційного забезпечення було обрано базу даних MS SQL Server – це система управління реляційними базами даних. У реляційній базі даних, дані зберігаються в окремих таблицях, завдяки чому досягається вигран у швидкості й гнучкості. Таблиці зв'язуються між собою за допомогою відносин, завдяки чому забезпечується можливість поєднувати при виконанні запиту дані з декількох таблиць. SQL як частина системи MS

SQL можна охарактеризувати як мову структурованих запитів, що використовується для доступу до баз даних.

Microsoft SQL Server – система керування базами даних, розроблена корпорацією Microsoft і працює тільки під Windows та Linux.

Для адміністрування БД і розробки додатків для СУБД SQL Server може використовуватися ряд інструментів, що надаються корпорацією Microsoft:

1. SQLCMD - інструмент з інтерфейсом типу командного рядка, призначений для виконання команд SQL інтерактивно або зі скрипта; використовується в якості інсталяційного інтерфейсу за замовчуванням.

2. Microsoft Visual Studio включає підтримку розробки для SQL Server.

3. SQL Server Management Studio - IDE для конфігурації і управління БД SQL Server. [14]

Тож, які переваги ми отримуємо при роботі з SQL Server?

a. Підтримка. Так як SQL Server є платним середовищем, то ми отримуємо при цьому «повний пакет» всіх послуг та інструментів без додаткових сторонніх рішень;

b. Зберігання даних. SQL Server використовує єдину систему збереження, створену Microsoft, що збільшує швидкість роботи, порівняно з іншими засобами;

c. Відміна запиту. На відміну від MySQL, в SQL Server дозволяє всередині виконання запиту зробити його відміну;

d. Безпека. В SQL Server є власна сучасна система безпеки. [15]

2.3 Публікація бази даних

Базу даних було опубліковано на інтернет ресурсі для швидкого доступу із різних локальних комп'ютерів. Так, враховуючи вартість розміщення та необхідні функціональні вимоги для нормальної роботи бази даних, було обрано інтернет ресурс: somee.com. На цьому сайті можна безкоштовно

розмістити своє невелике програмне забезпечення (до 30 МБ) та заодно отримати мінімально-необхідну функціональність для внесення даних та подальшого тестування програми в інтернет середовищі. Дана база даних доступно по адресі: SchoolMeals.mssql.somee.com (рис. 3)

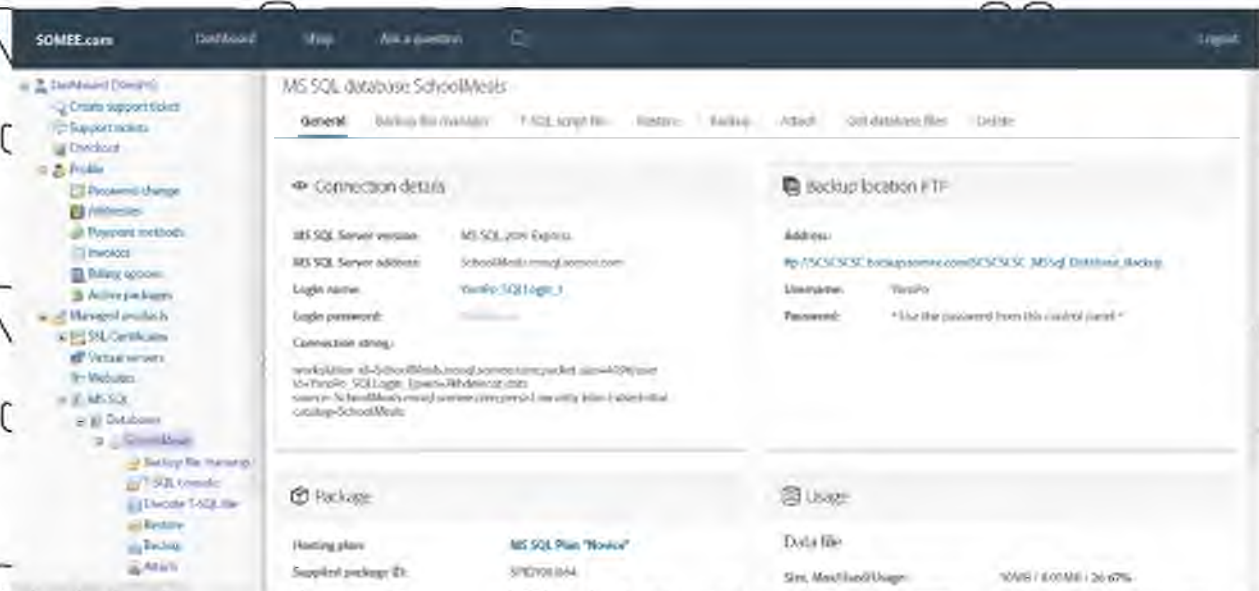


Рис. 3 Публікація БД на сайті somee.com

2.4 Проектування сховища даних

2.4.1 Загальні поняття в напрямку OLAP технології. OLAP (on-line analytical processing) набір технологій для оперативної обробки інформації, що включають динамічну побудову звітів у різних розрізах, аналіз даних, моніторинг та прогнозування ключових показників бізнесу. В основі OLAP-технологій лежить подання інформації у вигляді OLAP-кубів.

OLAP-куби містять бізнес-показники, що використовуються для аналізу та прийняття управлінських рішень, наприклад прибуток, рентабельність продукції, сукупні кошти (активи), власні кошти, позикові кошти тощо.

Бізнес-показники зберігаються у кубах не у вигляді простих таблиць, як і звичайних системах обліку чи бухгалтерських програмах, а в розрізах, що

становлять основні бізнес-категорії діяльності організації: товари, магазини, клієнти, час продажу тощо.

Завдяки детальному структуруванню інформації OLAP-куби дозволяють оперативно здійснювати аналіз даних та формувати звіти у

різних розрізах та з довільною глибиною деталізації. Звіти можуть створюватися аналітиками, менеджерами, фінансистами, керівниками підрозділів в інтерактивному режимі для того, щоб швидко отримати відповіді на запитання, що виникають щодня, і прийняти правильне рішення.

При цьому співробітникам для створення звітів не потрібно вдаватися до послуг програмістів, на що зазвичай витрачається чимало часу. [16]

Автор реляційної моделі даних Е.Ф. Кодя сформулював 18 правил OLAP, які розділив на 4 групи:

а) Основні властивості:

- багатовимірне представлення даних;
- інтуїтивне оперування даними (без застосування меню);
- доступність;
- пакетне вилучення замість інтерпретації;
- моделі аналізу OLAP (за категоріями, тлумачний, абстрактний і стереотипний);
- архітектура "клієнт-сервер";
- прозорість;
- можливість одночасного обслуговування багатьох користувачів.

б) Спеціальні властивості:

- обробка ненормалізованих даних;
- збереження результатів OLAP;
- вилучення значень, яких немає (відрізняються від нульових значень);
- обробка значень, яких немає (ігноруються OLAP-аналізатором без врахування їх джерел).

с) Особливості представлення звітів:

- гнучкість формування звітів;
- стандартна продуктивність звітів (не знижується із зростанням кількості вимірів і об'єму бази даних);

- автоматичне налаштування фізичного рівня.

д) Управління вимірами:

- універсальність вимірів;
- необмежена кількість вимірів і рівнів агрегації;
- необмежені операції над розмірностями.

Перевагами OLAP-систем є:

1. простота використання і сприйняття зведених таблиць;
2. повнота аналітичних даних;
3. повна і легка настройка звіту користувачем без залучення програміста;
4. можливість деталізувати звіт в процесі аналізу даних;
5. швидке формування звітів;
6. несуперечність даних між звітами;
7. консолідація інформації з різних БД;
8. підвищений захист даних.

2.4.2 Поняття сховища даних. Різновидом баз даних є сховище даних

(Data Warehouse). Поняття сховищ даних виникло зовсім недавно.

Необхідність розробки нової концепції сховищ даних обумовлена такими факторами:

Розвиток інформаційних технологій привів до систем нового типу, які дістали назву систем підтримки прийняття рішень. Ці системи основані на новій технології, яка дістала назву OLAP-технології. Основою OLAP-технології є реалізація аналітичних запитів

Системи підтримки прийняття рішень, основані на формуванні аналітичних запитів, почали конфліктувати з транзакційними системами оперативної обробки даних (OLTP-системами). Одночасне вирішення оперативних і аналітичних запитів на одній базі даних часто призводить до нестачі ресурсів.

Формування аналітичних звітів на основі традиційних баз даних, які вміщують оперативну інформацію, займає дуже багато часу. Причому витрати часу, необхідні для формування аналітичних звітів, невпинно зростають зі зростанням обсягів оперативної інформації в базі даних. Це призводить до того, що менеджери не встигають готувати відповідні рішення на основі отриманих аналітичних звітів.

Дуже часто на підприємстві чи в організації функціонує декілька OLTP-систем, кожна з яких має свою окрему базу даних, в яких використовуються різні структури даних, способи кодування, одиниці вимірювання. Побудова зведеного аналітичного запиту на основі декількох баз даних є дуже складною проблемою, яка спочатку потребує вирішення проблеми узгодженості даних, що зберігаються в різних базах даних.

Вирішення перерахованих вище проблем було знайдено в розробці концепції сховища даних. Сховище даних має виконувати функції попереднього добору, агрегації та підготовки оперативних даних OLTP-системам. Тобто в сховищі даних зберігаються не первинні дані, а певним чином інтегровані дані, які створюють основу для вирішення аналітичних задач і функціонування систем підтримки прийняття рішень. [17]

Основне призначення сховища даних – надання точних даних та інформації з найменшими затратами часу і коштів.

Основний принцип його роботи був сформований автором: дані, одного разу занесені до сховища даних, у подальшому багаторазово витягуються з нього і використовуються для аналізу. Звідси випливає одна з основних переваг використання сховища даних в роботі підприємства – контроль за

критично важливою інформацією, отриманою з різних джерел, як за виробничим ресурсом.

Сховища даних є основою для побудови систем підтримки прийняття рішень. Основна мета створення СД в тому, щоб зробити усі значимі для

управління бізнесом дані доступними в стандартизованій формі, придатними для аналізу та отримання необхідних звітів. Для досягнення цього потрібно

отримати дані із існуючих внутрішніх та зовнішніх, доступних для комп'ютера, джерел. Незважаючи на відмінності в підходах та реалізаціях,

усім сховищам даних властиві такі спільні риси: предметна орієнтованість, інтегрованість, прив'язка до часу, незмінність.

Предметна орієнтованість. Інформація в сховищі даних організована у відповідності до основних аспектів діяльності підприємства (замовники, продажі, склад тощо). Це відрізняє сховище даних від оперативної БД, де

дані організовані відповідно до процесів (виписка рахунків, відвантаження товару тощо). Предметна організація даних в сховищі сприяє як значному

спрощенню аналізу, так і підвищенню швидкості виконання аналітичних запитів. Вона виражається, зокрема, в використанні інших, порівняно з

оперативними системами, систем організації даних. У випадку зберігання даних в реляційній СУБД використовується схема «зірки» (star) чи

«сніжинки» (snowflake). Крім цього, дані можуть зберігатися в спеціальній багатовимірній СУБД в n-вимірних кубах.

Інтегрованість. Вихідні дані отримуються із оперативних БД, перевіряються, очищуються, приводяться до єдиного виду, в потрібній мірі

агрегуються (вираховуються сумарні та інші статистичні показники) і завантажуються в сховище. Такі інтегровані дані набагато простіше

аналізувати.

Прив'язка до часу. Дані в сховищі завжди напряму зв'язані з певним періодом часу. Дані, отримані із оперативних БД, накопичуються в сховищі

у виді «історичних шарів», кожен з яких стосується конкретного періоду часу. Це дозволяє аналізувати тенденції в розвитку бізнесу.

Незмінність. Потрапивши в певний «історичний шар» сховища, дані уже ніколи не мінятимуться. Це також відрізняє сховище від оперативної БД,

в якій дані постійно змінюються, у зв'язку з чим один і той же запит, виконаний в різні моменти часу, може дати різні результати. Стабільність даних також полегшує їх аналіз. [18]

Як вже зазначалось раніше, сховище даних має бути в формі «зірки» або «сніжинки», а також має складатися з вимірів та фактів.

Виміри – це визначальні атрибути фактів, і зазвичай відповідають на будь-які питання: коли стався факт, над чим або з чим саме, хто був об'єктом або суб'єктом і т.п. В основному, виміри мають більш описовий (тобто текстовий) характер, так як кінцевому користувачеві буде набагато легше сприймати результати описані текстом (наприклад, назва місяця), ніж цифрами (номер місяця в році).

Один вимір куба може міститися як в одній таблиці (в тому числі і при наявності декількох рівнів ієрархії), так і в декількох зв'язаних таблицях, що відповідають різним рівням ієрархії у вимірі. Якщо кожне вимірювання міститься в одній таблиці, така схема сховища даних носить назву «зірка».

Якщо ж хоча б один вимір міститься в декількох зв'язаних таблицях, така схема сховища даних носить назву «сніжинка». Додаткові таблиці вимірів в такій схемі, зазвичай відповідні верхнім рівням ієрархії вимірювання і знаходяться в співвідношенні «один до багатьох» в головній таблиці вимірів, що відповідає нижньому рівню ієрархії, іноді називають консольними таблицями (outrigger table).

Факти – це фактичні записи про якомусь процесі, який ми хочемо аналізувати. Таблиця фактів, як правило, містить унікальний складений ключ, який об'єднує первинні ключі таблиць вимірів. Найчастіше це цілочисельні значення або значення типу «дата / час» - адже таблиця фактів

може містити сотні тисяч або навіть мільйони записів, і зберігати в ній повторювані текстові описи, як правило, не вигідно - краще помістити їх в менші за обсягом таблиці вимірів. При цьому як ключові, так і деякі

неключових поля повинні відповідати майбутнім вимірам OLAP-куба. Крім

цього таблиця фактів містить одне або кілька числових полів, на підставі яких в подальшому будуть отримані агрегатні дані. Структура сховища даних магістерської роботи типу «зірка» зображена на рис 4

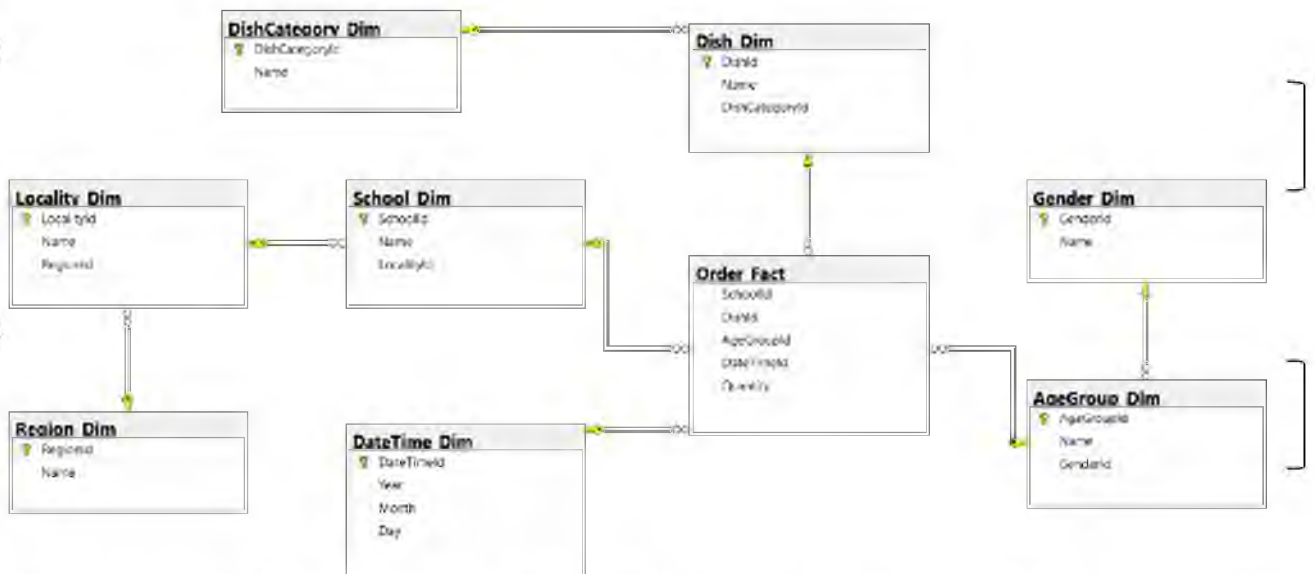


Рис. 4 Структура сховища даних

Для зберігання даних використовується схема «зірка». Як видно з рис.

3, сховище даних має 8 вимірів та 1 таблицю фактів, які потрібні для:

- DishCategory_Dim – міститься інформація про категорію страви, наприклад, друга страв, напої, тощо;
- Dish_Dim – міститься інформація про назву страви та категорія, до якої страв належить;
- Gender Dim – міститься інформація про стать дитини: дівчина чи хлопець;

• AgeGroup_Dim – міститься інформація про вікову категорію та стать, до якої належить вікова категорія;

• Region_Dim – знаходиться інформація про область України;

• Locality_Dim – знаходиться інформація про назву міста та область, яка їй належить;

• School_Dim – знаходиться інформація про школи в містах та їх областях;

• DateTime_Dim – часова таблиця вимірів, в якій є рік, місяць та день;

• Order_Fact – в таблиці фактів знаходиться вся інформація з таблиць вимірів таких як назва школи, яке місто, який регіон школи, назва страви та її категорія, вікова категорія дитини та її стать, а також дата замовлення.

2.5 Постачання даних до гіперкубу

2.5.1 Побудова розгорнутого кубу. OLAP надає зручні швидкодіючі засоби доступу, перегляду та аналізу інформації. Користувач отримує природну, інтуїтивно зрозумілу модель даних, організовуючи їх у вигляді багатовимірних кубів.

Осями багатовимірної системи координат є основні атрибути аналізованого бізнес-процесу. В якості одного з вимірів використовується час. На перетинах осей – виміру знаходяться дані, які кількісно характеризують процес – міри. Користувач, що аналізує інформацію, може "розрізати" куб за різними напрямками, отримувати зведені (по місяцях, роках) або, навпаки, детальні (по днях) відомості та здійснювати інші маніпуляції, які може вигадати користувач в процесі аналізу даних.

На основі сховища даних було створено гіперкуб для подальшого проведення аналізу даних. Для побудови гіперкубу було використано

технологію SQL Server Data Tools від Microsoft, яка інтегрується в середовище розробки Visual Studio. Дана технологія дозволяє працювати зі сховищами даних, створювати проекти SSAS та SSIS.

Першим етапом побудови гіперкубу було створення проекту типу Analysis Service (рис. 5). Microsoft Analysis Services - частина Microsoft SQL Server, системи управління базами даних (СУБД). Microsoft включила набір служб в SQL Server, пов'язаних з бізнес-аналізом і зберіганням даних.

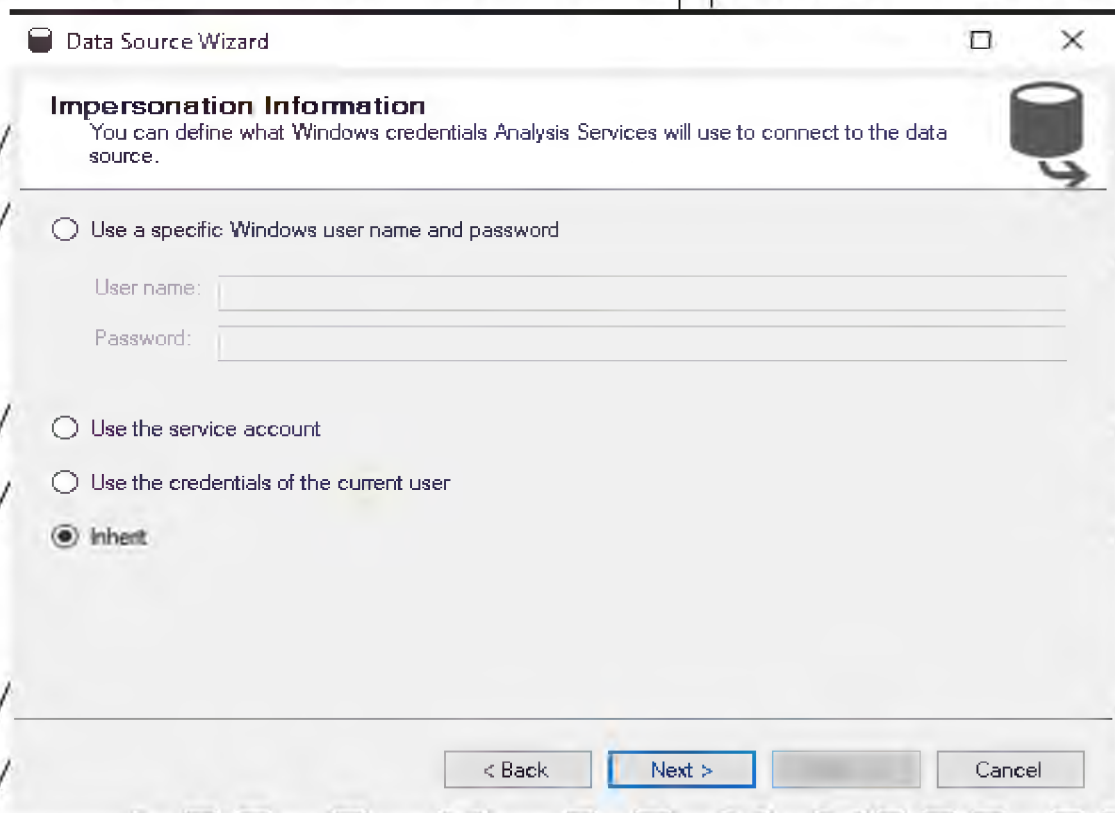


Рис. 5 Створення проекту

Наступним етапом є підключення до сховища даних, що зображено на

рис. 6

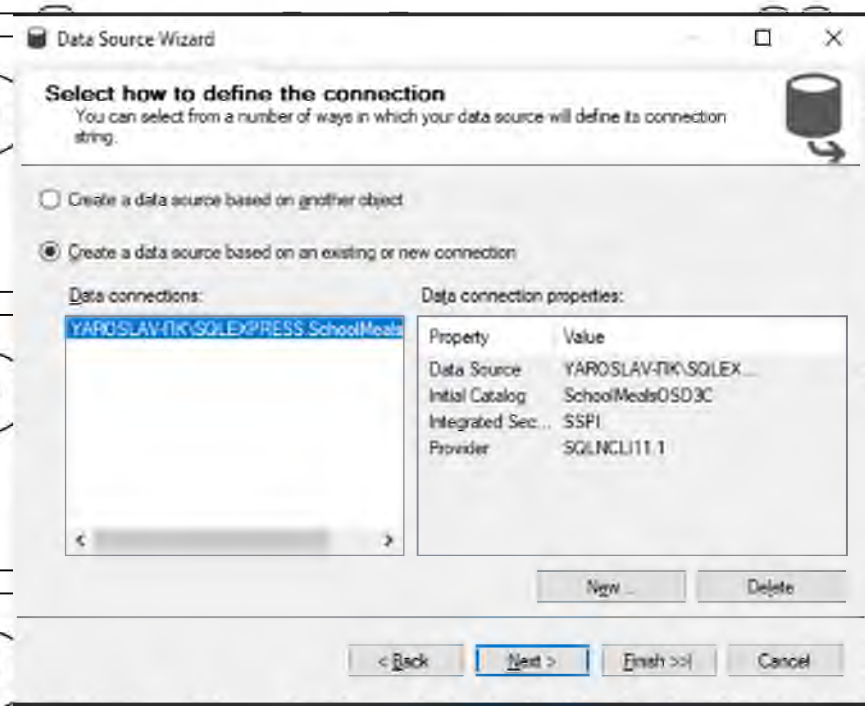


Рис. 6 Підключення до джерела даних

Тепер необхідно створити представлення джерела даних (рис. 7)

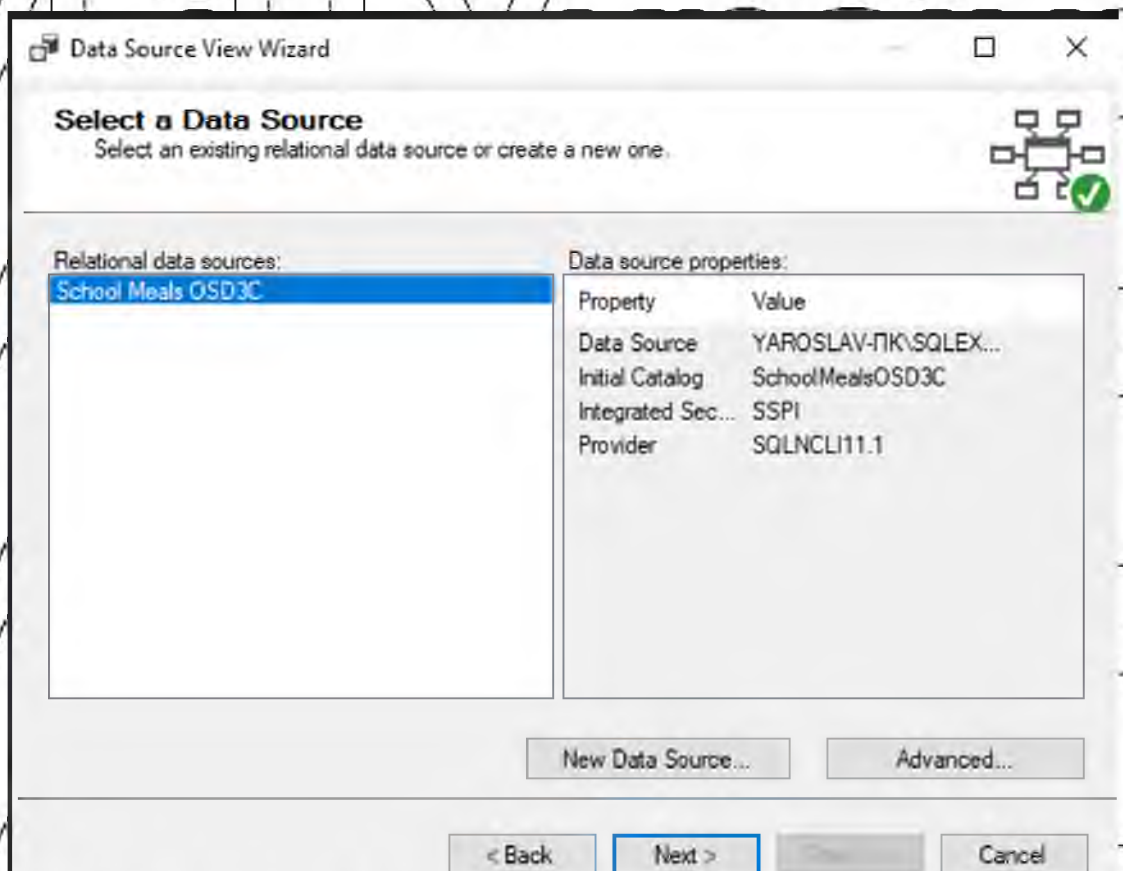


Рис. 7 Джерело даних

Наступним кроком було обирання необхідних для розгортання кубу таблиць (рис. 8-9).

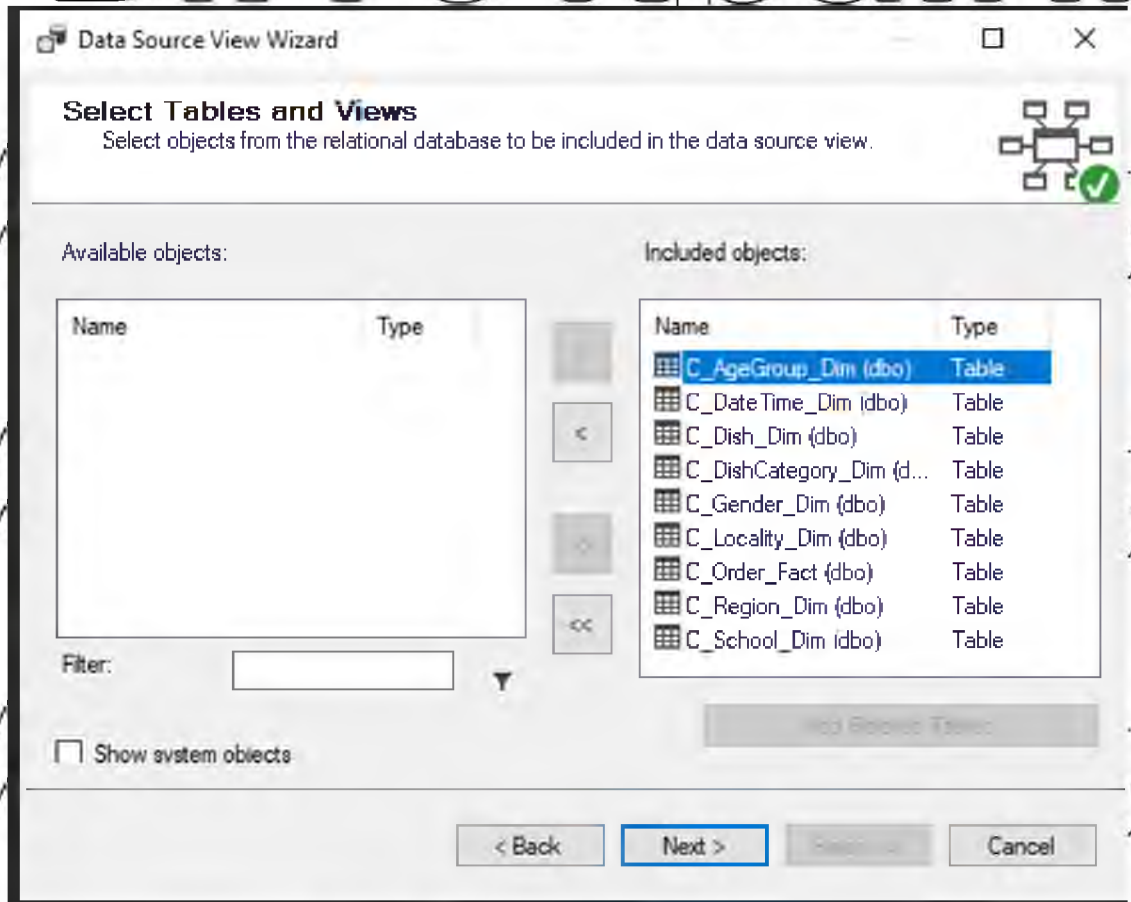


Рис. 8. Вибір таблиць для кубу

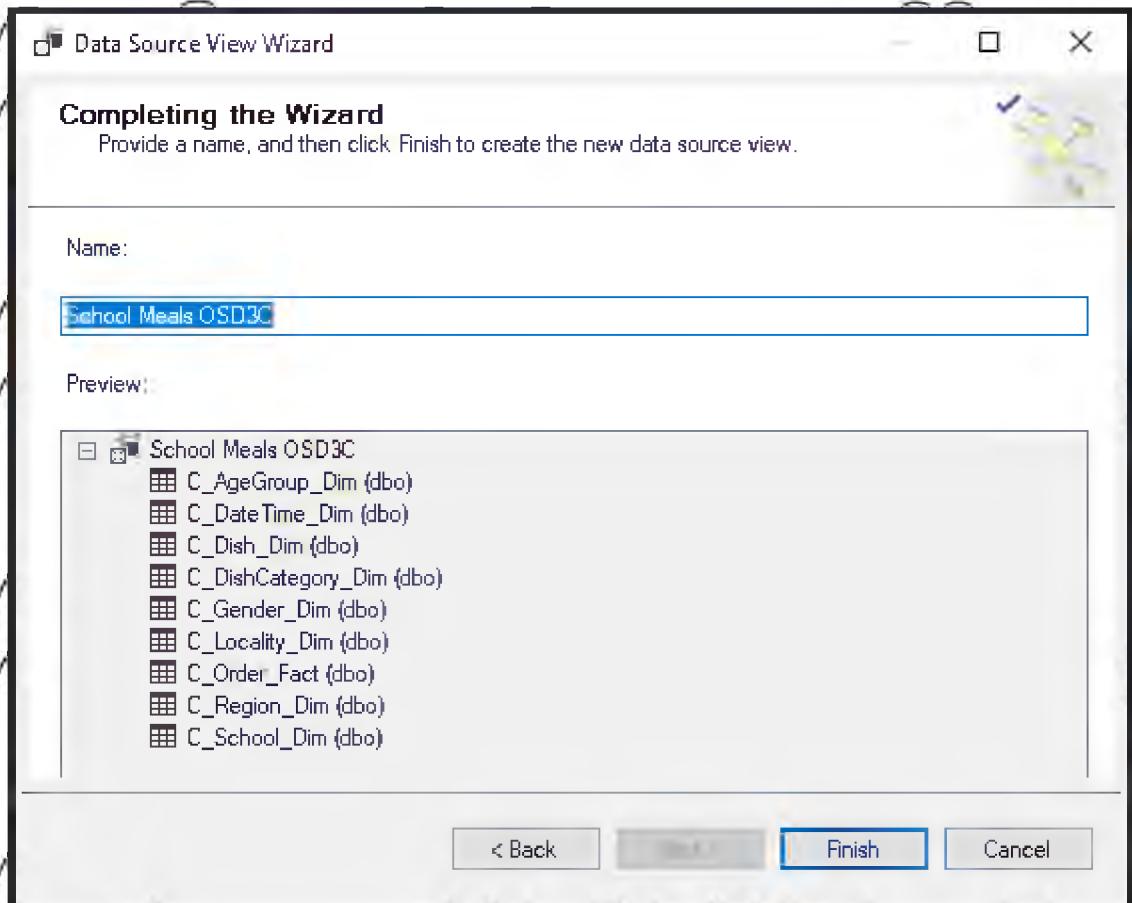


Рис. 9 Створення назви проекту для кубу

Далі ми обираємо метод, за яким розгортаємо куб (рис. 10) та обираємо таблиці (рис. 11), а потім і поля цих таблиць сховища (рис. 12) на основі яких буде створено виміри (рис. 13).

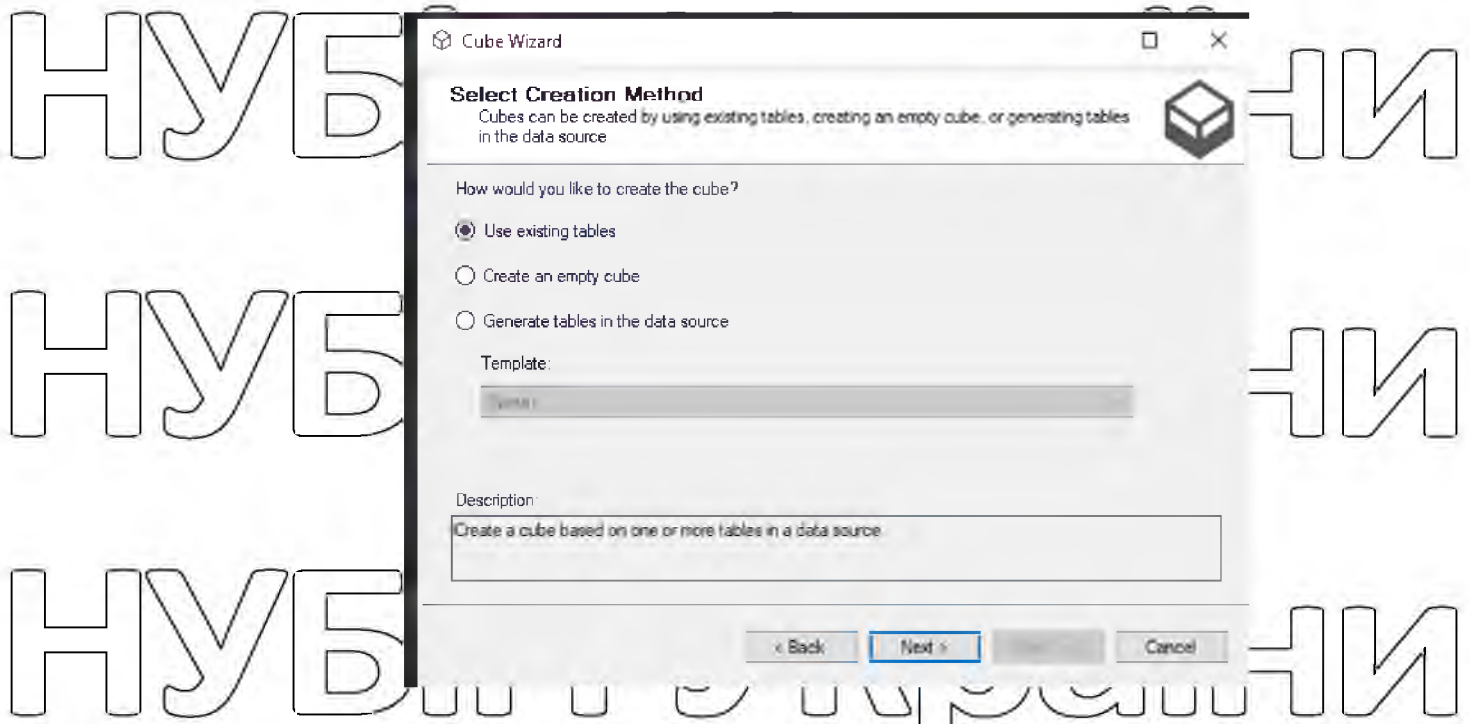


Рис. 10 Метод створення кубу

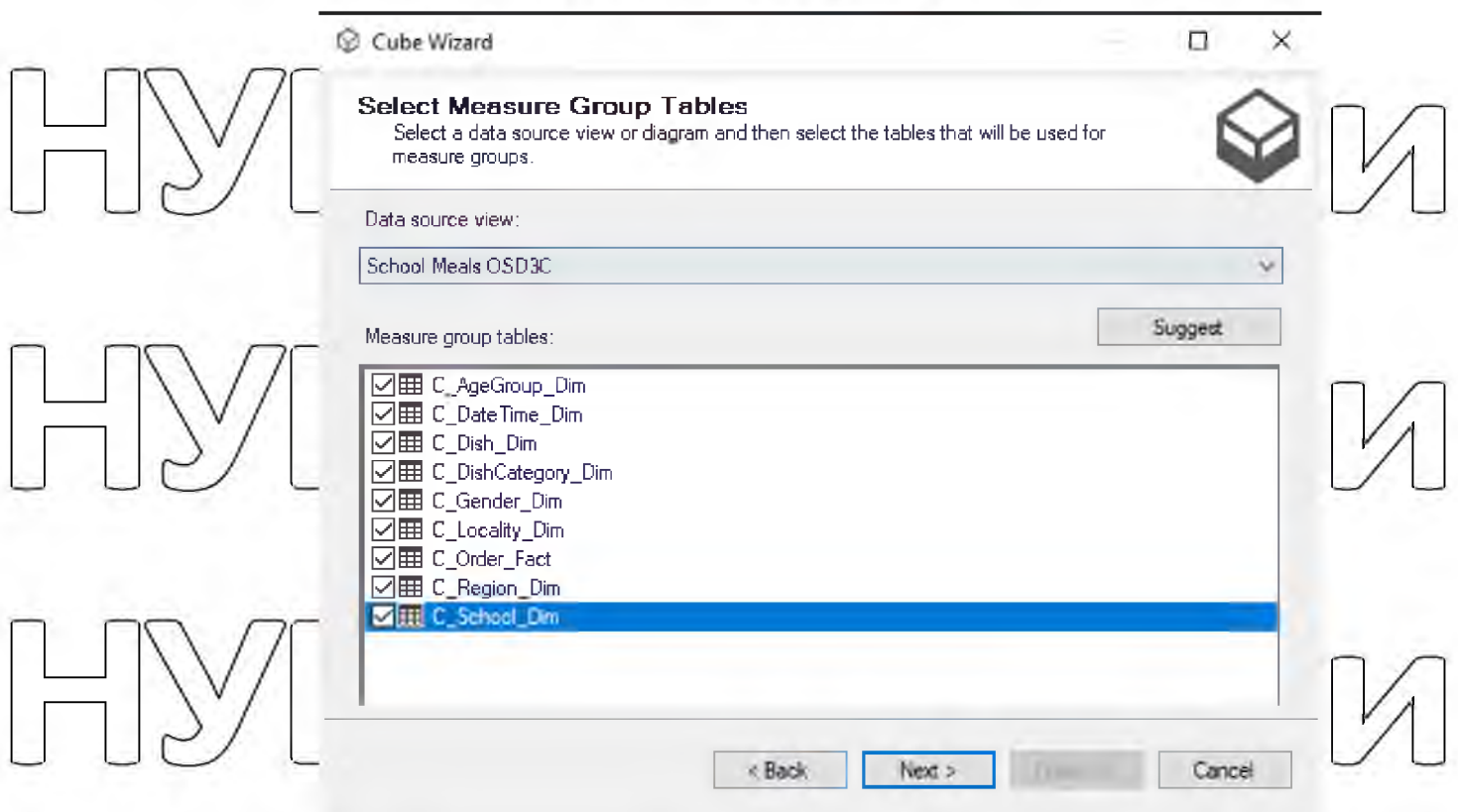


Рис. 11 Вибір таблиць для кубу

НУБІП України

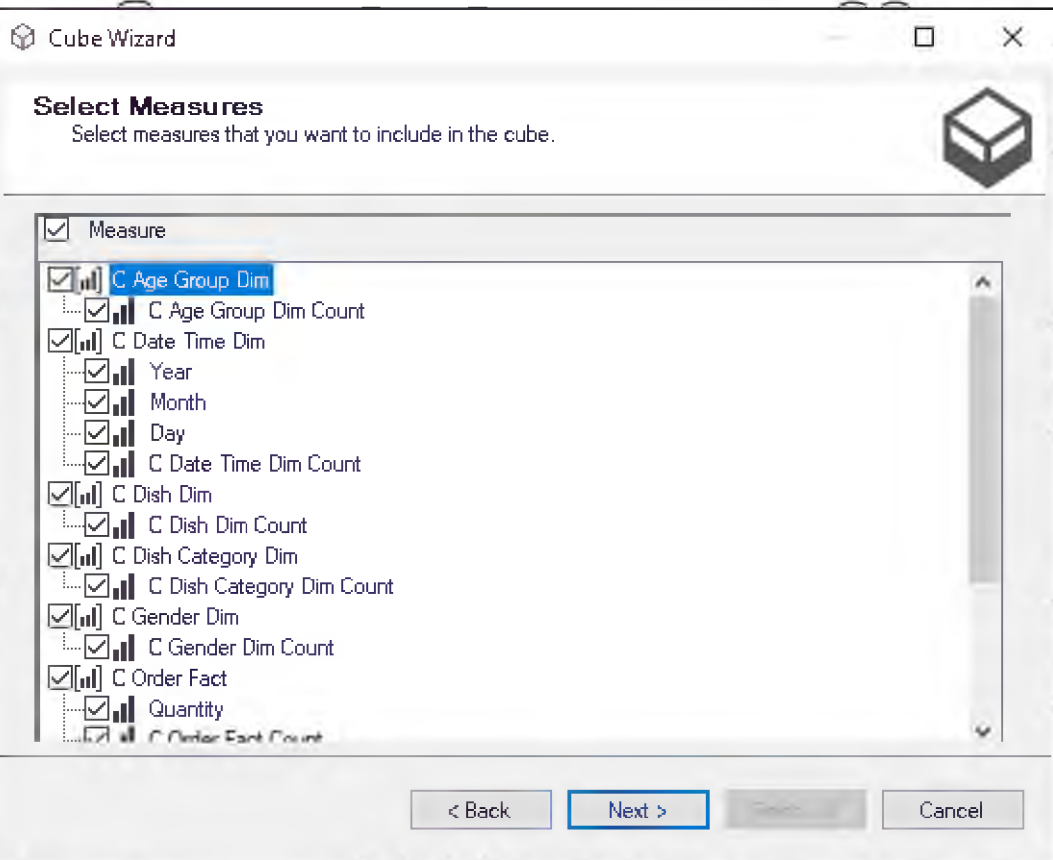


Рис. 12 Вибір даних з таблиць

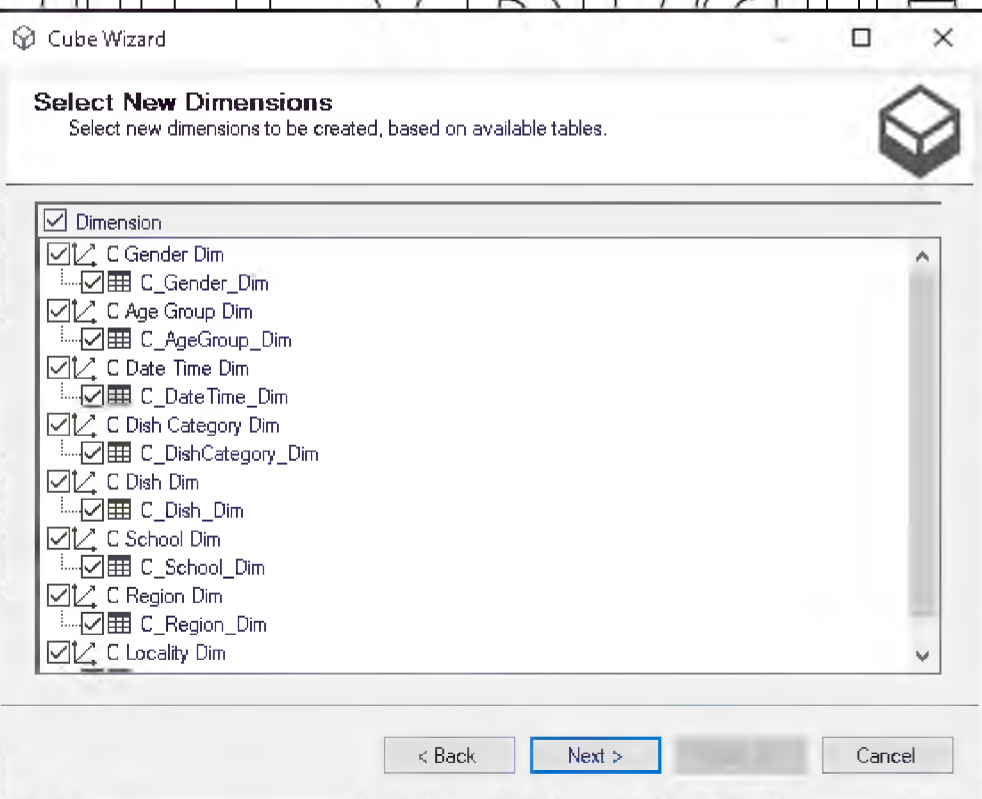


Рис. 13 Вибір вимірів для кубу

В результаті попередніх кроків ми отримали розгорнутий куб, який можна побачити на рис. 14

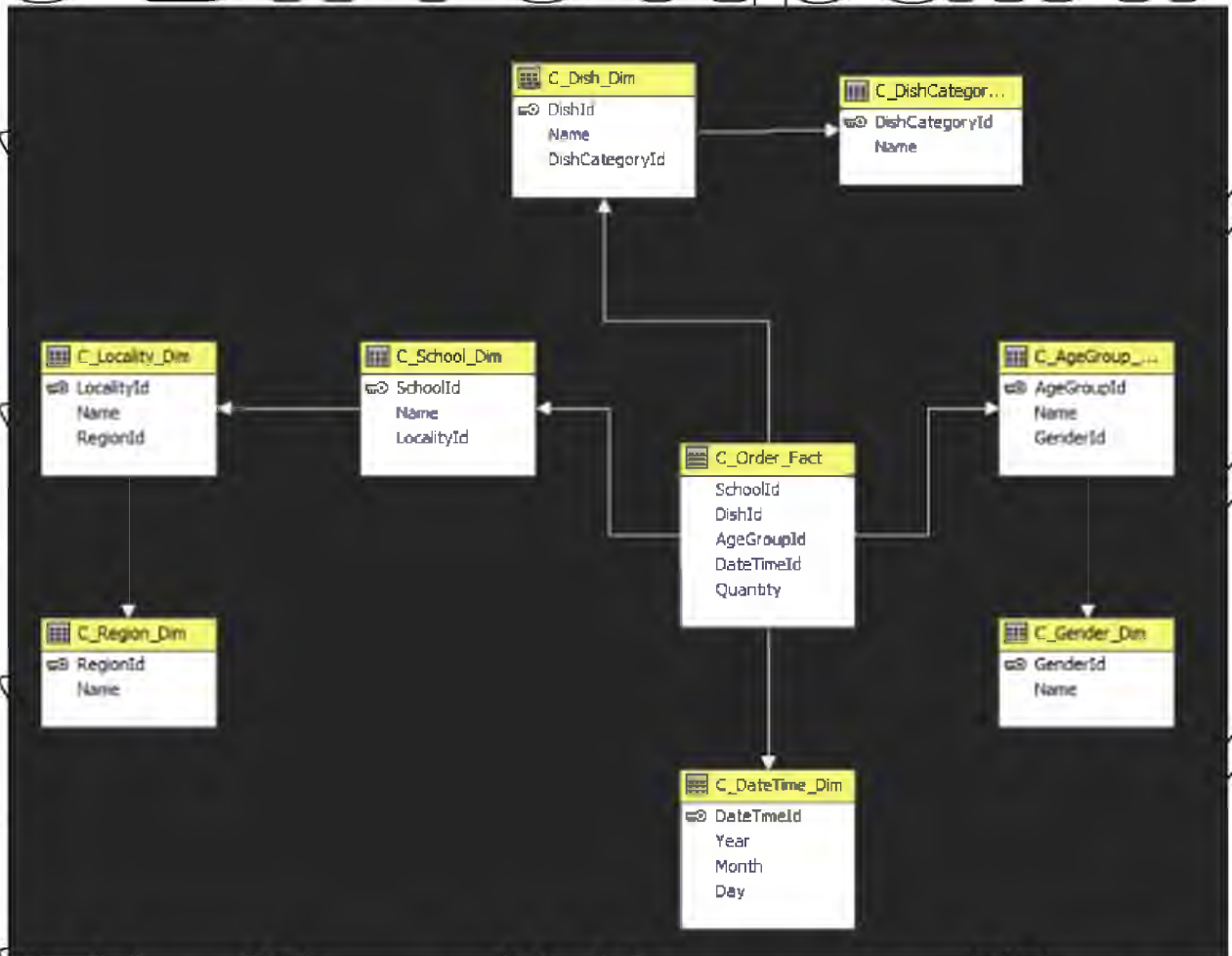


Рис. 14 Розгорнутий куб

Створення кубу на основі сховища даних було необхідним для подальшої роботи і створення аналізу даних з різних джерел. Сховище даних не може проводити аналіз, на відміну від кубу, який створений для аналізу даних за допомогою підсистем аналізу від Microsoft. Гіперкуб було створено для подальшої роботи з інструментарієм Power BI.

2.5.2 Заповнення кубу даними. SQL Server Integration Services

(скорочено SSIS) формально визначається як набір графічних, консольних утиліт та програмних об'єктів, які призначені для вилучення, перетворення та консолідації даних із різномірних джерел у різні «пункти призначення». Цей процес іноді називають ETL (від англ. Extract, Transform, Load - дослівно

«витяг, перетворення, завантаження»). Можна вважати, що SSIS – це щось подібне до потужного насоса, призначеного для перекачування даних з одного місця в інше. Крім того, служби SSIS у процесі «перекачування» можуть перетворювати та перевіряти дані.

У попередніх версіях SQL Server служби-попередники SSIS, призначені для перекачування даних, називалися Data Transformation Services (DTS). Однак зміни в SQL Server 2005 щодо SSIS/DTS можна назвати радше революційними, ніж еволюційними. Змінилося практично все: засоби створення та адміністрування пакетів, формат пакетів, середовище виконання, об'єктні моделі, консольні утиліти тощо. [19]

Найважливішим завданням у SSIS є завдання потоку даних (Data Flow Task). Завдання потоку даних SSIS можна вибрати безпосередньо з панелі інструментів SSDT, а потім джерело та призначення визначаються в рамках завдання. Завдання потоку даних – це не просто перетворення зіставлення для вхідних і вихідних стовпців. Це завдання має свою власну поверхню проектування, як-от Control Flow, де ви можете організувати подібні завдання компоненти, які називаються трансформаціями, щоб маніпулювати даними, коли вони надходять у конвеєр від джерела до місця призначення. Потік даних, як ви можете собі уявити, є серцем SSIS, оскільки він інкапсулює всі аспекти перетворення даних ETL.

Потоки даних можуть розділяти дані в конвеєрі на основі елемента даних і обробляти кожен потік окремо. У потоці даних рядок заголовка файлу можна відокремити та розглянути окремо від рядків деталей. Коли конвеєр виходить з процесу очищення даних, потоки можуть бути відправлені в окремі пункти призначення або зведені до кінцевого комбінованого призначення. Варто зауважити, що в пакеті SSIS може бути кілька різних потоків даних. Для кожної із завдань потоку даних, яку додають до поверхні керування, можна мати відповідну поверхню потоку даних. Це завдання є настільки важливим і невід'ємною частиною переміщення даних у SSIS.

У Data Flow Task передбачено чотири типи елементів:

- Data Flow Sources (Джерела потоку даних) — це джерела даних, з яких Data Flow Task завантажуватиме дані для переміщення до пункту призначення. У розпорядженні є шість видів джерел. Всі вони доступні в розділі Data Flow Sources у Toolbox;

- Data Flow Destinations (Призначення потоку даних) — пункти призначення для даних, куди це завдання поміщатиме отримані з джерела та перетворені дані. Усі 11 призначень доступні з розділу Data Flow Destinations у Toolbox;

- Data Flow Transformations (Перетворення потоку даних) — це перетворення, які будуть виконані в процесі переканування даних. Безліч перетворень доступна з розділу Data Flow Transformations у Toolbox.

Зазвичай вони розміщуються між джерелом та призначенням даних.[20]

У випадку моєї магістерської роботи джерелом є база даних. Для початку необхідно створити проект служби SSIS. Після цього створюються чотири задачі потоку управління, які будуть запускатися по черзі (таблиці вимірів та таблиця фактів), що зображені на рис. 15.



Рис. 15 Задачі потоку даних

В кожному потоці було створено окремі потоки даних декількох вимірів для кожної таблиці. На рис. 16 показано третій рівень, який знаходиться

якнайдалі від таблиці фактів відносно зв'язків між таблицями, в ньому потоці знаходиться інформація про області України.

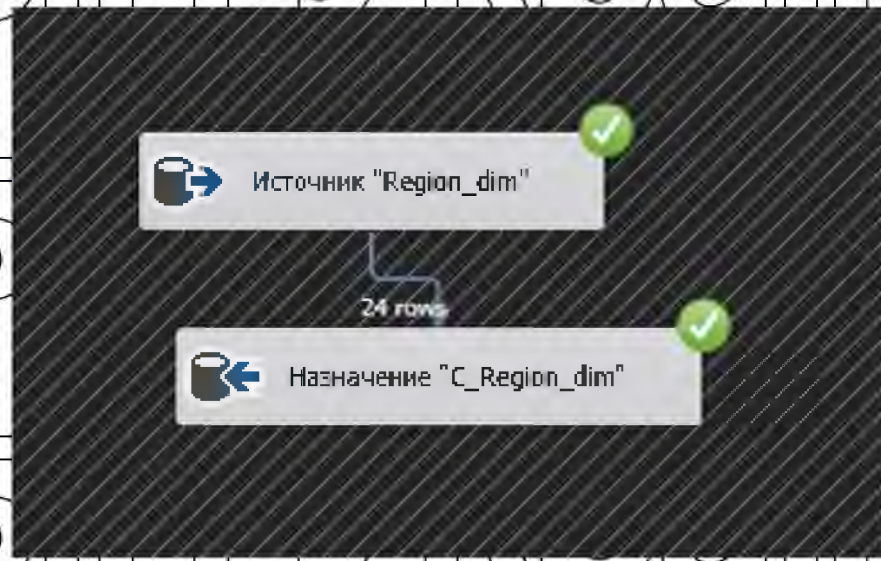


Рис. 16 Потік даних 3-го рівня

На рис. 17 зображено другий рівень, який поєднує третій та перший рівні вимірів сховища даних. В нього входять такі таблиці вимірів, які мають інформацію про стать дитини, категорію страви, а також інформація про міста.

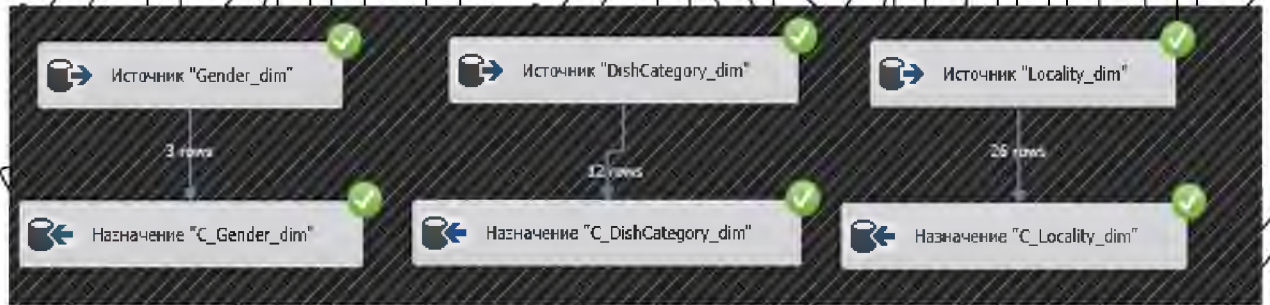


Рис. 17 Потік даних 2-го рівня

І на останок, потік першого рівня – це потік фактів, що на рис. 18.



Рис. 18 Потік фактів

На рис. 14-18 видно, що дані було передано вдало з різною кількістю рядків.

4 АНАЛІЗ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ

3.1 Технології, алгоритми та методи аналізу даних

Аналіз даних — це процес виконання послідовних логічних дій з інтерпретації відповідей респондентів та їх перетворення у статистичні форми, необхідні для ухвалення маркетингових та управлінських рішень. Зазвичай цей процес складається з трьох послідовних етапів (рис. 19).

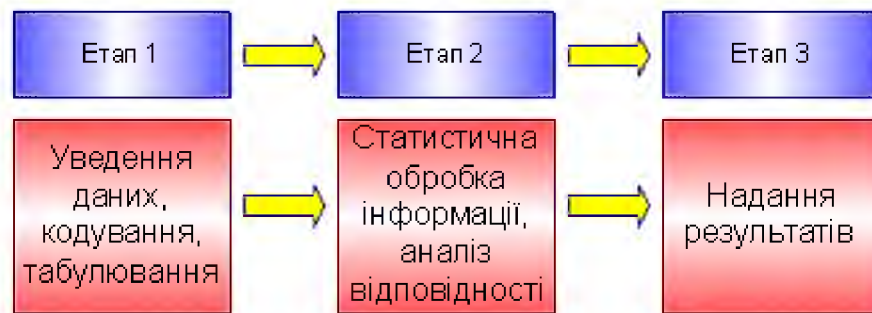


Рис. 19 Процес аналізу даних

На першому етапі дані, зібрані у процесі опитування, вводять у комп'ютер як матриці (відбувається табуляція даних). Уведення даних і їх табуляція може здійснюватися за допомогою спеціальних комп'ютерних програм, наприклад, Vortek, SPSS, Statistica, або в середовищі MS Excel чи Access. І в першому, і в другому випадку уможливується сортування, форматування, відбір та інші операції оброблення даних.

Слід зауважити, що введення даних є процесом трудомістким і часто вимагає залучення спеціально підготовлених фахівців. Тому ще на етапі планування дослідження доцільно визначитися з оптимальною (а краще мінімальною) кількістю запитань у анкеті для запобігання труднощам під час введення та аналізу отриманих даних. Проблеми загострюються, коли опрацьовувати доводиться значну кількість анкет.

На другому етапі проводять статистичний аналіз даних, пошук взаємозв'язків і відмінностей у масивах. Крім того, статистична обробка

передбачає розробку висновків та гіпотез (концептуалізацію даних), одночасно проводять перевірку репрезентативності результатів, їх здатності до перенесення на всю генеральну сукупність. [21]

Засоби аналізу даних полегшують користувачам обробку даних і маніпулювання ними, аналіз взаємозв'язків та кореляцій між наборами даних, а також допомагає виявити закономірності та тенденції інтерпретації. Існує кілька типів методів аналізу даних, які були створені на основі бізнесу та технологій. Основними видами аналізу даних є:

- Аналіз тексту;
- Статистичний аналіз;
- Діагностичний аналіз;
- Прогностичний аналіз;
- Рекомендаційний аналіз.

Аналіз тексту. Аналіз тексту також називають Data Mining. Це один із методів аналізу даних для виявлення закономірності у великих наборах даних за допомогою баз даних або інструментів аналізу даних. Він використовувався для перетворення вихідних даних у бізнес-інформацію. На ринку присутні інструменти Business Intelligence, які використовуються для прийняття стратегічних бізнес-рішень. Загалом він пропонує спосіб вилучення та вивчення даних, виведення шаблонів і, нарешті, інтерпретації даних.

Статистичний аналіз. Статистичний аналіз показує «Що станеться?» використовуючи минулі дані у вигляді інформаційних панелей. Статистичний аналіз включає збір, аналіз, інтерпретацію, представлення та моделювання даних. Він аналізує набір даних або вибірку даних. Існує дві категорії цього типу аналізу – описовий аналіз і аналіз висновків.

Описовий аналіз аналізує повні дані або вибірку зведених числових даних. Він показує середнє значення та відхилення для безперервних даних, тоді як відсоток і частоту для категорійних даних.

Висновок аналізує зразок з повних даних. У цьому типі аналізу ви можете знайти різні висновки з одних і тих же даних, вибравши різні зразки.

Діагностичний аналіз. Діагностичний аналіз показує «Чому це сталося?» знайшовши причину з огляду, знайденого в статистичному аналізі.

Цей аналіз корисний для визначення моделей поведінки даних. Якщо у вашому бізнес-процесі з'являється нова проблема, ви можете переглянути цей аналіз, щоб знайти подібні моделі цієї проблеми. І він може мати шанси використовувати подібні рецепти для нових проблем.

Прогнозний аналіз. Прогнозний аналіз показує, «що може статися», використовуючи попередні дані. Найпростіший приклад аналізу даних: якщо минулого року я купив дві гри на основі своїх заощаджень, а цього року моя зарплата збільшиться вдвічі, то я можу купити чотири гри. Але, звичайно, це не просто, тому що потрібно думати про інші обставини, як-от ймовірність зростання цін на ігри цього року або, можливо, замість ігри ви хочете купити новий телефон, або вам потрібно купити будинок.

Отже, цей Аналіз робить прогнози щодо майбутніх результатів на основі поточних або минулих даних. Прогноз – це лише оцінка. Його точність залежить від того, скільки детальної інформації у вас є і скільки ви в ній комаєтесь.

Рекомендаційний аналіз. Рекомендаційний аналіз поєднує уявлення з усіх попередніх аналізів, щоб визначити, які дії слід вжити у поточній проблемі чи рішенні. Більшість компаній, що керуються даними, використовують директивний аналіз, оскільки прогнозного та описового аналізу недостатньо для підвищення продуктивності даних. Виходячи з поточних ситуацій і проблем, вони аналізують дані та приймають рішення.

Процес аналізу даних — це ніщо інше, як збір інформації за допомогою відповідної програми або інструменту, який дозволяє досліджувати дані та знаходити в них закономірність. На основі цієї інформації та даних ви можете приймати рішення або робити остаточні висновки.

НУВІП УКРАЇНИ

Аналіз даних складається з наступних етапів:

- збір вимог до даних;
- збір даних;
- очищення даних;

НУВІП УКРАЇНИ

- аналіз даних;
- інтерпретація даних;
- візуалізація даних;
- збір вимог до даних;

НУВІП УКРАЇНИ

Збір вимог до даних. Перш за все, необхідно подумати навіщо робити цей аналіз даних. Необхідно вирішити, який тип аналізу даних потрібно зробити! На цьому етапі потрібно вирішити, що аналізувати і як це виміряти, зрозуміти, чому проводиться дослідження і які заходи повинні використовуватись для цього аналізу.

НУВІП УКРАЇНИ

Збір даних. Після збору вимог ви отримаєте чітке уявлення про те, які речі ви повинні виміряти і якими повинні бути ваші висновки. Тепер потрібно збирати дані на основі вимог. Коли збираються дані, слід пам'ятати, що зібрані дані мають бути оброблені або організовані для аналізу. Оскільки

НУВІП УКРАЇНИ

зібрані дані з різних джерел, необхідно вести журнал із датою збору та джерелом даних.

НУВІП УКРАЇНИ

Очищення даних. Тепер усі зібрані дані можуть бути некорисними або невідповідними для мети аналізу, тому їх слід очистити. Зібрані дані можуть містити повторювані записи, пробіли або помилки. Дані повинні бути очищені та без помилок. Цей етап необхідно виконати перед аналізом, оскільки на основі очищення даних результат аналізу буде ближче до очікуваного.

НУВІП УКРАЇНИ

Аналіз даних. Коли дані зібрано, очищено й оброблено, вони готові до аналізу. Коли маніпулюють даними, можна виявити, що є точна інформація, яка потрібна, або може знадобитися зібрати більше даних. На цьому етапі можна використовувати інструменти та програмне забезпечення аналізу

даних, які допоможуть зрозуміти, інтерпретувати та робити висновки на основі вимог.

Інтерпретація даних. Після аналізу даних настав час інтерпретувати результати. Можна вибрати спосіб вираження або передачі аналізу даних або просто словами або, можливо, таблицею чи діаграмою. Потім скористатися результатами процесу аналізу даних, щоб обрати найкращий спосіб дій.

Візуалізація даних. Візуалізація даних дуже поширена у нашому повсякденному житті; вони часто представлені у вигляді діаграм і графіків.

Іншими словами, дані відображаються графічно, щоб людському мозку було легше їх зрозуміти та обробити. Візуалізація даних часто використовується для виявлення невідомих фактів і тенденцій. Спостерігаючи за взаємозв'язками та порівнюючи набори даних, можна значущу інформацію.

Висновок:

- Аналіз даних означає процес очищення, перетворення та моделювання даних для виявлення корисної інформації для прийняття бізнес-рішень

- Типи аналізу даних: текстовий, статистичний, діагностичний, прогнозний, наказовий аналіз

- Аналіз даних складається зі збору вимог до даних, збору даних, очищення даних, аналізу даних, інтерпретації даних, візуалізації даних. [22]

Існують різноманітні методи аналізу даних, в залежності від питання, яке розглядається, типу даних та кількості даних, що зібрана. Кожен з цих методів сфокусований на стратегії вивчення нових даних, деталізації інформації для перетворення фактів та цифр у параметри прийняття рішень.

Відповідно, різні методи аналізу даних можна класифікувати наступним чином:

1. Методи, засновані на математиці та статистиці:

- описовий аналіз;
- регресійний аналіз;

НУВІП УКРАЇНИ

- факторний аналіз;
- дискримінантний аналіз;
- аналіз часових рядів.

2. Методи, засновані на штучному інтелекті та машинному навчанні:

НУВІП УКРАЇНИ

- штучні нейронні мережі;
- дерева рішень;
- еволюційне програмування;
- нечітка логіка.

3. Прийоми, засновані на візуалізації та графіках:

НУВІП УКРАЇНИ

- Стовпчаста діаграма, гістограма;
- Лінійна діаграма;
- Діаграма району;
- Кругова діаграма;

НУВІП УКРАЇНИ

- Діаграма воронки;
- Слово хмара діаграми;
- Діаграма Ганта;
- Радарна діаграма;

НУВІП УКРАЇНИ

- Діаграма розкиду;
- Діаграма міхурів;
- Датчик;
- Кадрова діаграма;

НУВІП УКРАЇНИ

- Діаграма прямокутного дерева;
- Карти.

На початку двохтисячних років, через значне збільшення даних, які потребують аналізу, було введено таке поняття як «Big Data».

НУВІП УКРАЇНИ

Big Data (великі дані) – це поєднання структурованих, напівструктурованих та неструктурованих даних, які можуть бути видобуті для отримання інформації та використані в проєктах машинного навчання, прогнозного моделювання та інших передових програм аналітики.

Системи, які обробляють і зберігають Big Data, стали загальним компонентом архітектур управління даними в великих організаціях. Big Data часто описуються такими характеристиками (англійською –

3V):

- великий обсяг даних (Volume).
- широкий вибір типів даних, що зберігаються в системах великих даних (Variety).

- швидкість, з якою дані генеруються, збираються та обробляються (Velocity)

Нещодавно до опису того, що таке Big Data, було додано кілька нових V:

- достовірність (Veracity)
- цінність (Value)

- мінливість (Variability)

Обсяг є найбільш часто цитованою характеристикою великих даних. Говорячи про те, що таке Big Data, слід пам'ятати, що великі дані – це

сукупність даних з різних джерел, починаючи від чітко визначених і закінчуючи слабо визначеними. Вони походять від людських або машинних джерел.

Різноманітність даних. Big Data також охоплюють широкий спектр типів даних, включаючи наступні:

- структуровані дані в базах даних та сховищах даних на основі мови структурованих запитів (SQL);
- неструктуровані дані, такі як текстові та файли документів, що зберігаються в кластерах Hadoop або системах баз даних NoSQL
- напівструктуровані дані, такі як журнали веб-сервера або потокові дані з датчиків.

з датчиків.

Всі різні типи даних можна зберігати разом в озері даних, яке, як правило, базується на Hadoop або службі зберігання хмарних об'єктів (що таке хмарні технології).

Крім того, програми для Big Data часто містять кілька джерел даних, які в іншому випадку не можуть бути інтегровані.

Наприклад, проєкт аналітики великих даних може спробувати оцінити успіх товару та майбутні продажі, співвідносячи дані про минулі продажі, дані про повернення та дані огляду онлайн-покупців цього товару.

Швидкість відноситься до швидкості, з якою генеруються великі дані і вони повинні бути оброблені та проаналізовані.

У багатьох випадках набори великих даних оновлюються в режимі майже реального часу, замість щоденних, щотижневих або щомісячних оновлень, характерних багатьом традиційним сховищам даних.

Програми аналітики великих даних співвідносять та аналізують вхідні дані, а потім надають відповідь або результат на основі запиту. Це означає, що аналітики даних повинні детально розуміти наявні дані та мати певне розуміння того, які відповіді вони шукають, щоб переконатися, що отримана інформація є дійсною та актуальною.

Управління швидкістю передачі даних також має важливе значення, оскільки аналіз великих даних поширюється на такі сфери, як машинне навчання та штучний інтелект, де аналітичні процеси автоматично знаходять закономірності у зібраних даних та використовують їх для отримання знань.

Хоча великі дані кількісно не порівнюються до будь-якого конкретного обсягу даних, розгортання великих даних часто включає в себе терабайти (TB), петабайти (PB) і навіть екзабайти (EB) даних.

Важливість великих даних. Компанії використовують накопичені в їх системах Big Data для поліпшення операцій, забезпечення кращого обслуговування споживачів, створення персоналізованих маркетингових

кампаній на основі конкретних уподобань клієнтів і, зрештою, підвищення прибутковості.

Підприємства, які використовують великі дані, мають потенційну конкурентну перевагу перед тими, хто цього не робить. Вони можуть приймати швидші та більш обгрунтовані ділові рішення, за умови, що вони ефективно використовують дані.

Наприклад, Big Data можуть надати компаніям цінну інформацію про своїх клієнтів. Вона може бути використана для вдосконалення маркетингових кампаній з метою збільшення залучення клієнтів та коефіцієнтів конверсії.

Крім того, використання великих даних дозволяє компаніям дедалі краще орієнтуватися на споживача.

Історичні дані та дані в реальному часі можуть бути використані для оцінки мінливих уподобань споживачів. Це дозволить підприємствам оновлювати та вдосконалювати свої маркетингові стратегії та ставати більш чутливими до бажань та потреб клієнтів.

Великі дані також використовуються медичними дослідниками для виявлення факторів ризику захворювання та лікарями для діагностики захворювань та станів у окремих пацієнтів.

Крім того, дані, отримані з електронних медичних записів, соціальних мереж, Інтернету та інших джерел, надають організаціям охорони здоров'я та державним установам найсвіжішу інформацію про загрози інфекційних захворювань чи спалахи захворювання.

В енергетичній галузі Big Data допомагають нафтогазовим компаніям визначати потенційні місця буріння та контролювати експлуатацію трубопроводів. Так само комунальні служби використовують їх для спостереження за електричними мережами.

Фірми фінансових послуг використовують системи Big Data для управління ризиками та аналізу ринкових даних у реальному часі.

Виробники та транспортні компанії покладаються на великі дані для управління своїми ланцюгами поставок та оптимізації шляхів доставки.

Інші сфери використання включають – реагування на надзвичайні ситуації, запобігання злочинності та побудова розумних міст.[23]

Методи Data Mining активно використовуються для пошуку раніше не відомих залежностей у великих масивах даних.

Data mining (інтелектуальний аналіз даних) – це процес, який використовується компаніями для перетворення необроблених даних в корисну інформацію. Використовуючи програмне забезпечення для пошуку закономірностей у великих пакетах даних, підприємства можуть дізнатися більше про своїх клієнтів, щоб розробити ефективніші маркетингові стратегії, збільшити продажі та зменшити витрати. Інтелектуальний аналіз даних залежить від ефективного збору, зберігання та комп'ютерної обробки даних [24]

Існує багато методів Data Mining, які можна використовувати, задля того, щоб перетворити необроблені дані на практичну інформацію, наприклад:

- очищення та підготовка даних;
- відстеження шаблонів;
- класифікація;
- асоціація;
- кластеризація;
- регресія;
- прогнозування;
- послідовні шаблони;
- дерева рішень;
- статистичні методи;
- візуалізація;
- нейронні мережі;

- машинне навчання та штучний інтелект.

В цій магістерській роботі для аналізу даних було обрано Data Mining, візуалізацію даних та технології OLAP.

3.2 Використання OLAP-технологій для аналізу даних

3.2.1 Вибір інструментарію для аналізу. Для побудови звітів з використання OLAP технології було обрано програмне середовище Power BI.

Power BI Desktop — це безкоштовна програма, яка встановлюється на локальному комп'ютері та дозволяє підключатися до джерел даних, перетворювати та візуалізувати дані. За допомогою Power BI Desktop можна підключатися до кількох різних джерел даних, а також поєднувати їх у моделі даних (цей процес називається моделюванням). Ця модель даних дозволяє створювати візуальні елементи та колекції візуальних елементів, якими можна ділитися з іншими людьми у вигляді звітів. Більшість користувачів, які працюють над проектами бізнес-аналітики, використовують Power BI Desktop для створення звітів, а службу Power BI — для надання до звітів загального доступу іншим користувачам.

Нижче наведені найпоширеніші варіанти застосування Power BI Desktop:

- підключення до даних;
- перетворення та очищення даних для створення моделі даних;
- створення візуальних елементів, таких як діаграми та графіки, які наочно відображають дані;
- створення звітів, які є колекціями візуальних елементів на одній або декількох сторінках звіту;
- спільне використання за допомогою служби Power BI.

Співробітники, які зазвичай виконують такі завдання, найчастіше називаються фахівцями з аналізу даних (або аналітиками) або фахівцями з

бізнес-аналітики (іноді їх ще називають творцями звітів). Але багато користувачів, які не вважають себе аналітиками або творцями звітів, успішно застосовують Power BI Desktop для створення привабливих звітів або вилучення даних з різних джерел і побудови моделей даних, які можна спільно використовувати з іншими співробітниками та організаціями. [25]

3.2.2 Результати аналізу даних на основі звітів. Power BI Desktop дозволяє створювати складні та візуально ефектні звіти, поєднуючи дані з кількох джерел в один звіт, до якого можна надати доступ багатьом користувачам в організації.

На основі даних, які було внесено до сховища, було створено різні звіти, які допоможуть проаналізувати та зробити висновки стосовно дитячого харчування.

Дані звіти мають відповідати на основне питання: які страви найчастіше замовляють діти певної вікової категорії за певний період часу в певному регіоні.

На рис. 20 зображено звіт, який дає зрозуміти в яких регіонах замовляють більше страв за 2020 та 2021 роки.

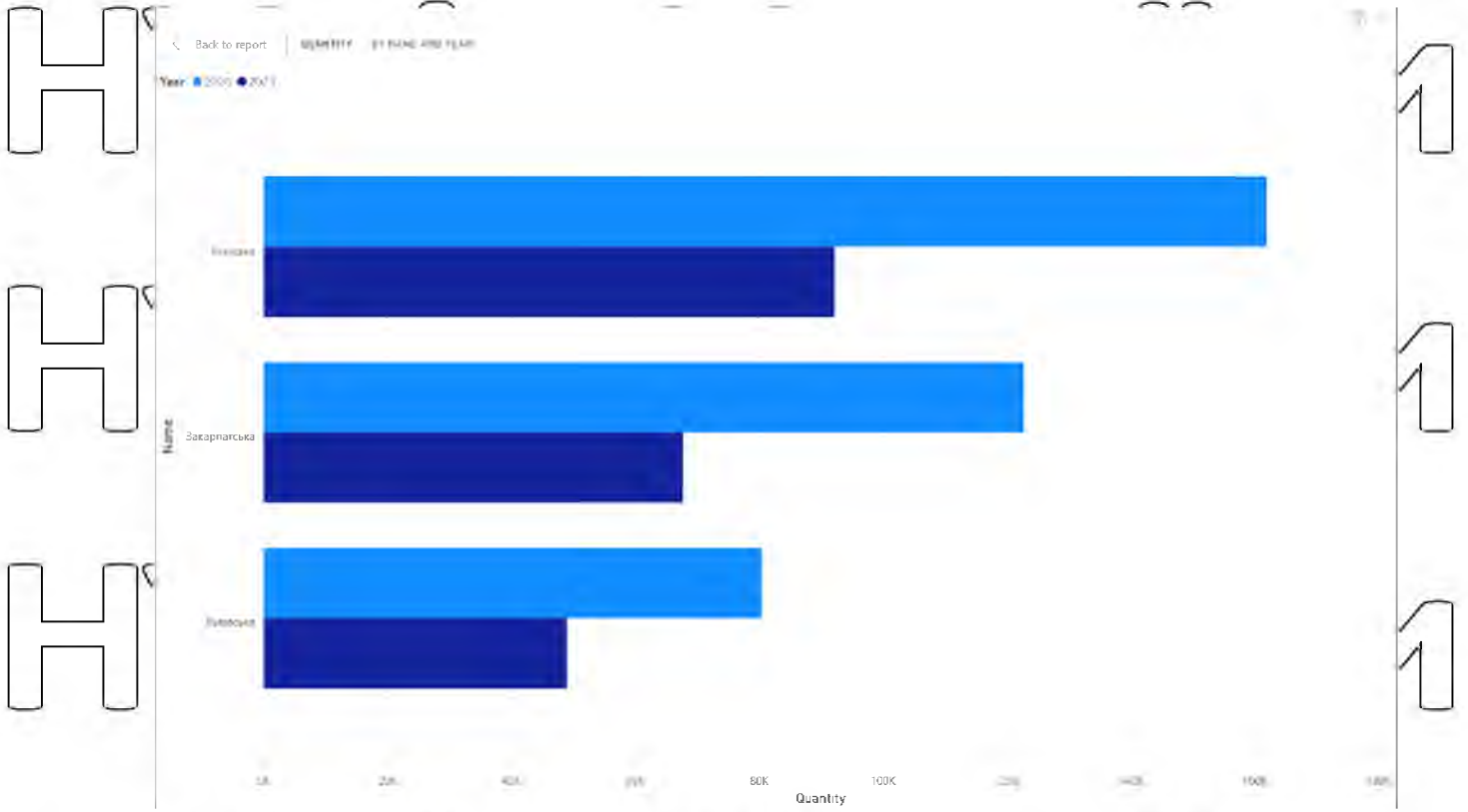


Рис. 20 Залежність замовлень страв від регіону

На рис. 21 звіт відображає залежність кількості замовлень страв від гендеру дитини по роках.

НУБІП України

НУБІП України

НУБІП України

НУБІП України

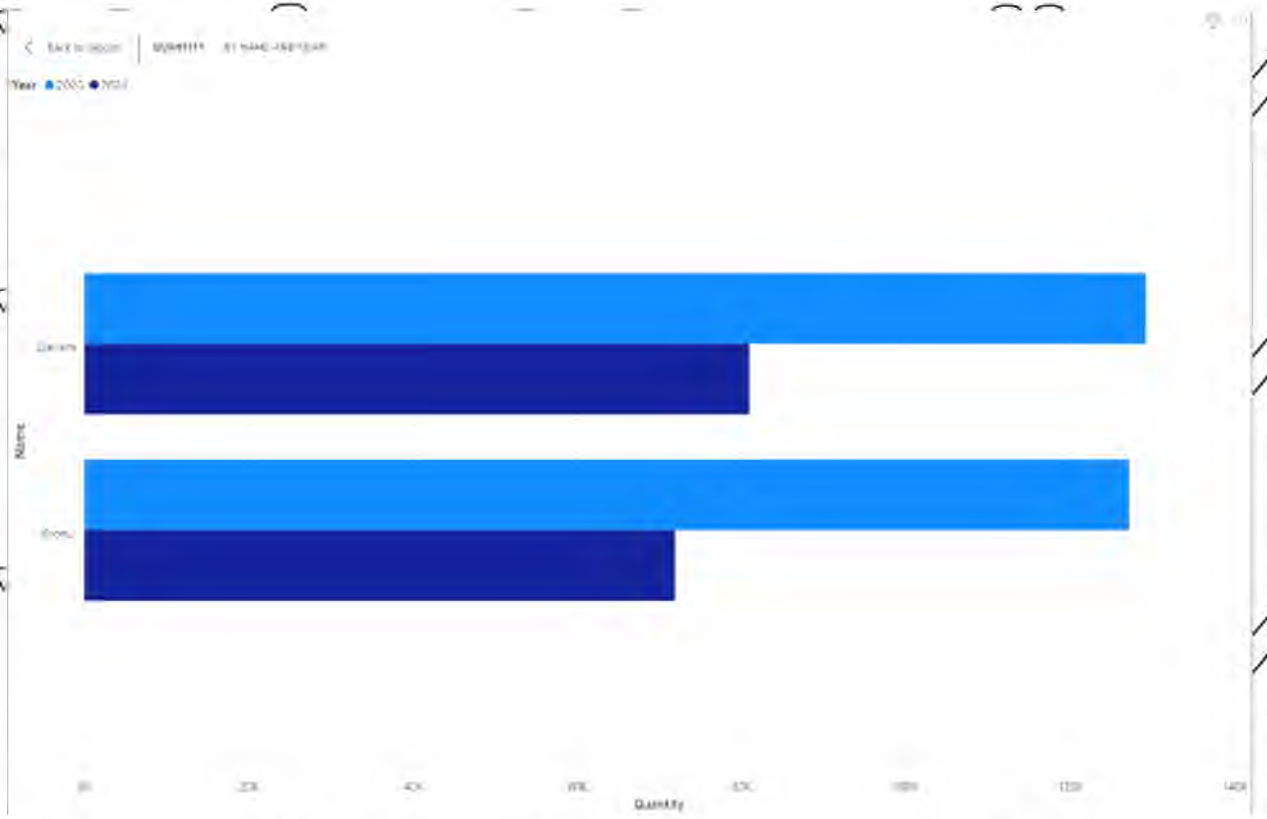


Рис. 21 Залежність замовлення страв від гендеру
Залежність вибору страв за роки від її категорії зображено на рис. 22.

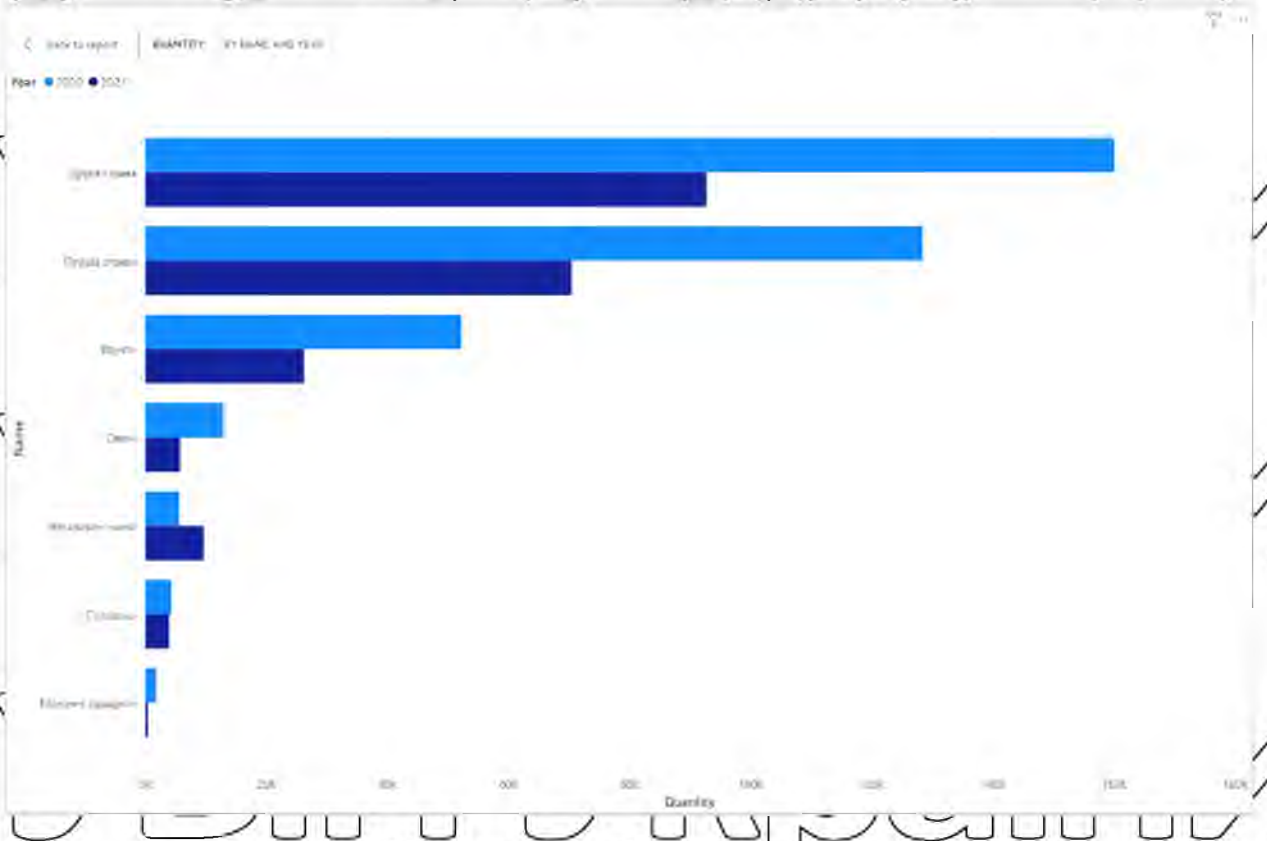


Рис. 22 Залежність вибору страв від категорії

На рис. 23 зображено вибір кількості страв залежно від вікової категорії.

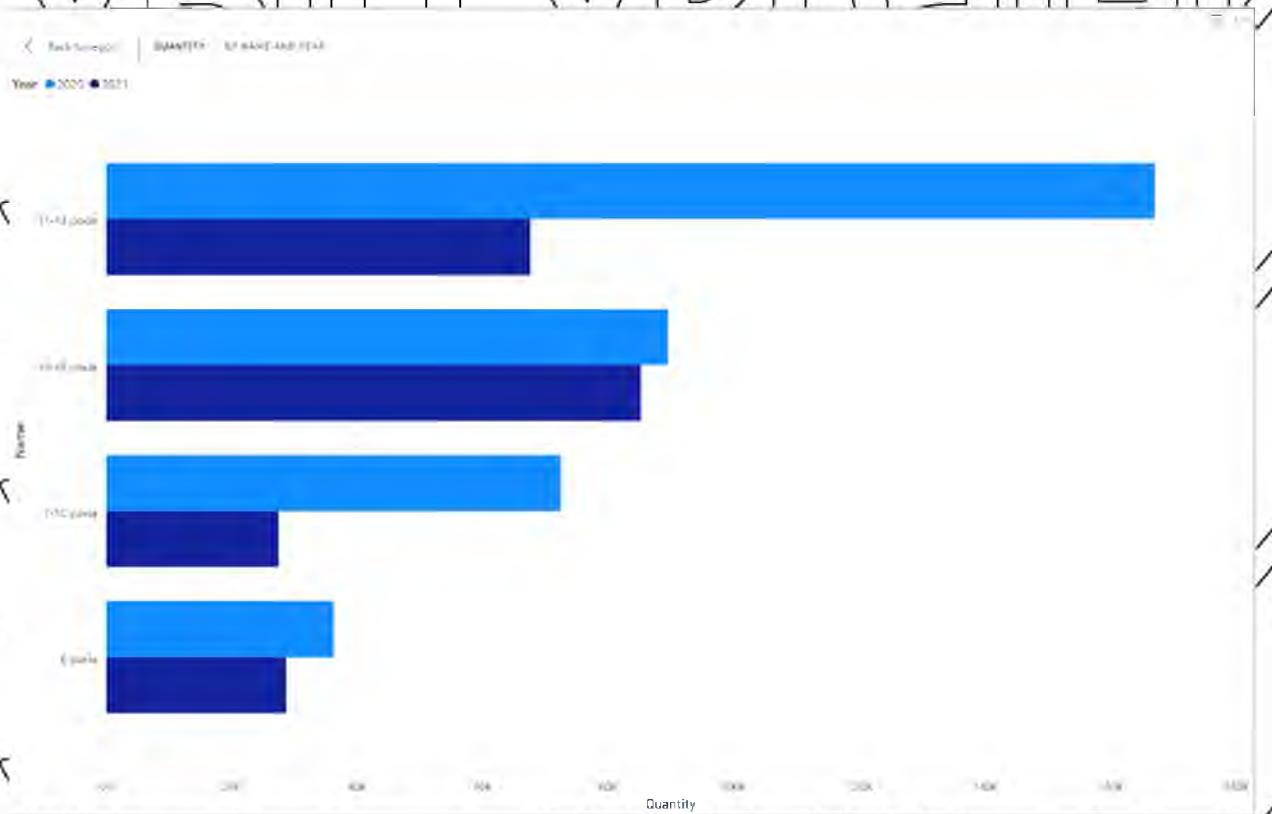


Рис. 23. Залежність замовлених страв від вікової категорії

На рис. 24 виведено залежність замовлених страв певних категорій від регіону країни.

НУБІП України

НУБІП України

НУБІП України

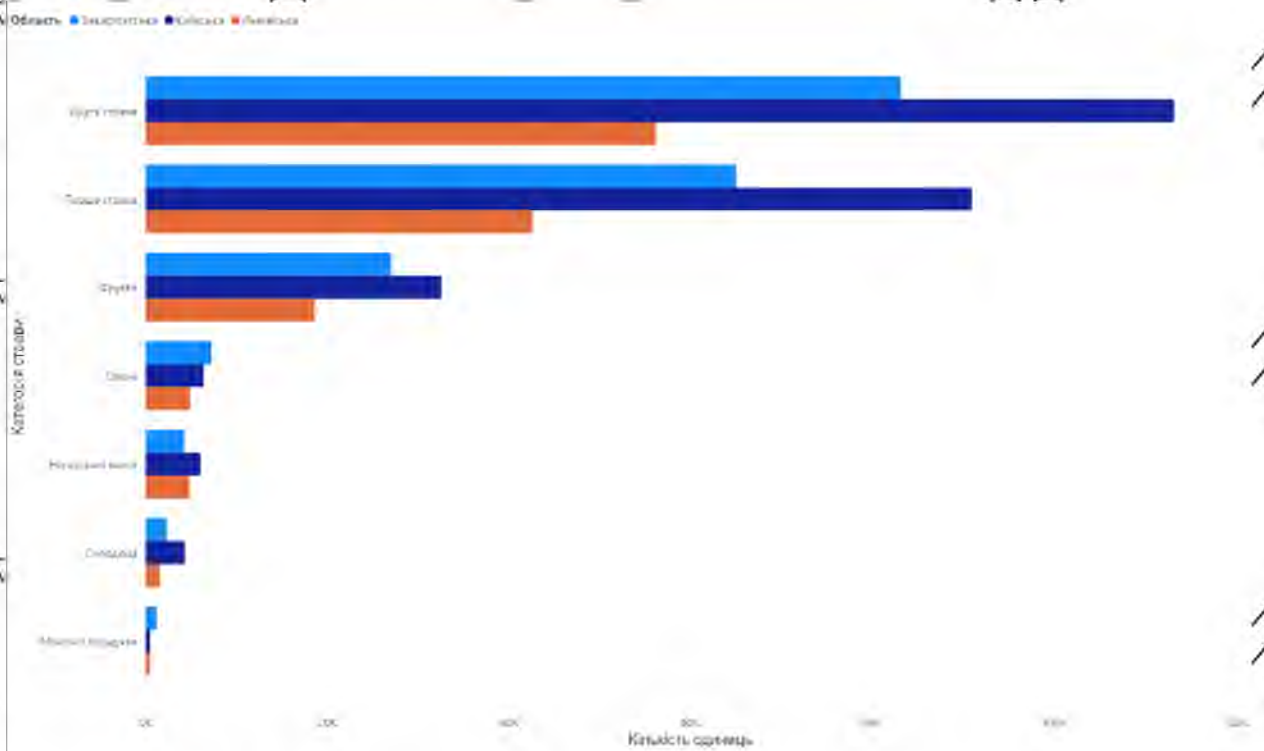


Рис. 24 Залежність замовлених страв певної категорії від регіону

Після попереднього було розроблено звіт, який показує, а які саме страви замовляються у певних регіонах найбільше (рис. 25).

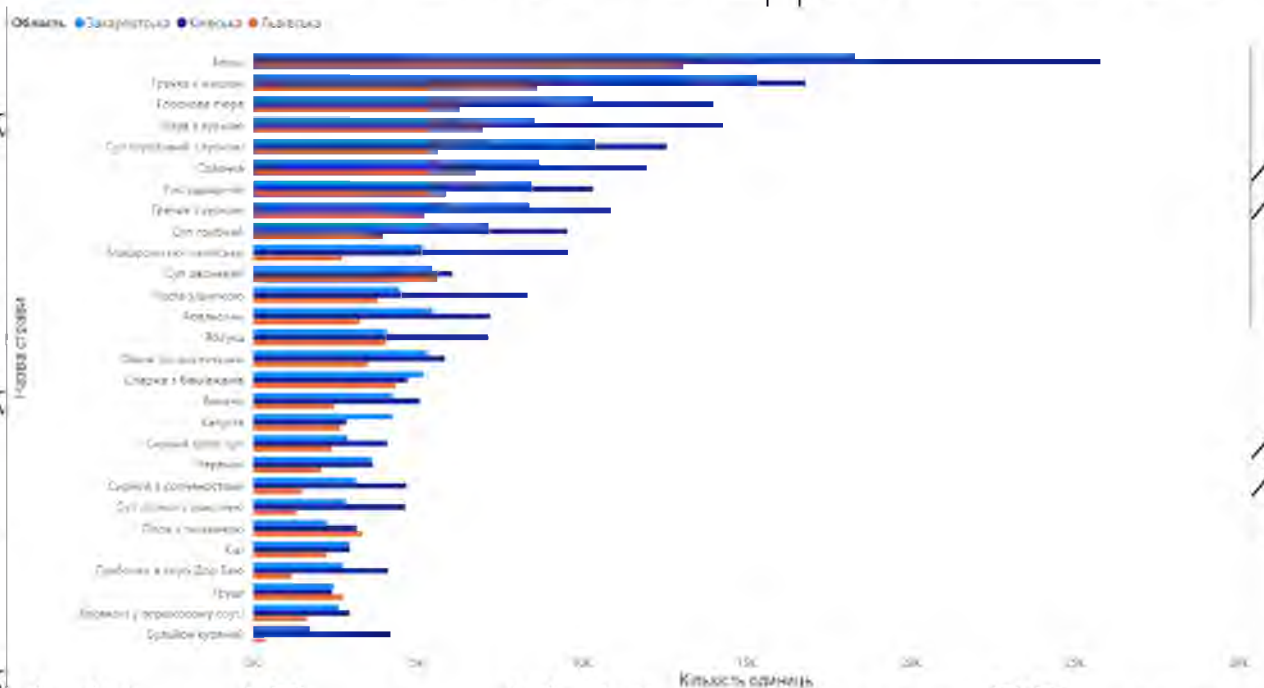


Рис. 25 Залежність вибору страв від регіону

Також було створено звіт, який показує які страви найчастіше замовлялись дітьми за роки (рис. 26)

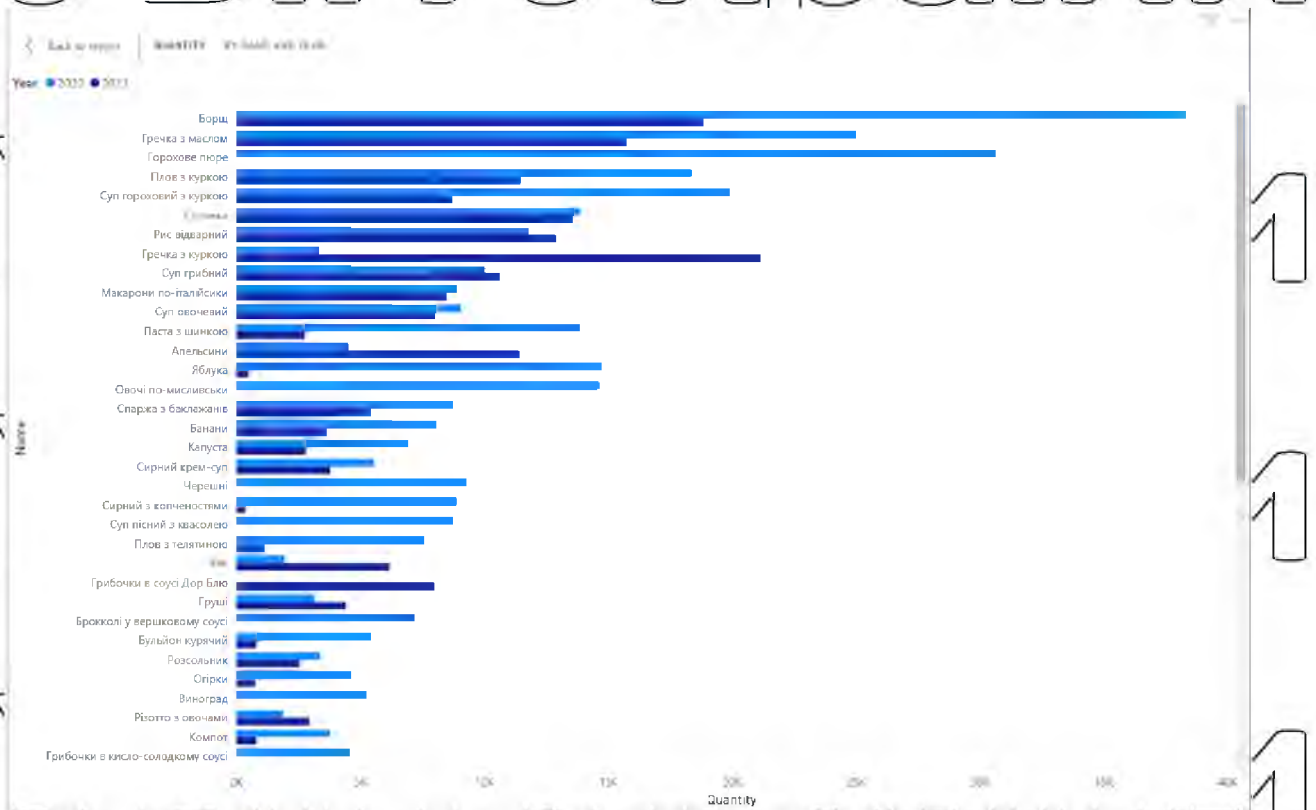


Рис. 26 Залежність вибору страв за весь період

3.2.3 Розрахунок KPI. KPI (key performance indicators, ключові

показники ефективності) - показники діяльності підрозділу (підприємства), які допомагають організації у досягненні стратегічних та тактичних (операційних) цілей. Використання ключових показників ефективності дає

підприємству можливість оцінити свій поточний стан та допомогти оцінити успішність реалізації обраної стратегії. KPI дозволяють проводити контроль лідової активності співробітників, підрозділів та компанії в цілому.

Самі по собі KPI не є системою мотивації, це інструмент для побудови системи управління - рахункові показники. Вони можуть бути кількісними (обсяг товару, витрачений час, отриманий прибуток) або якісними (рейтинг співробітників, нарахування балів).

Що дають показники KPI:

- Контроль над показниками ефективності;

НУБІП України

- Об'єктивна оцінка показників кожного фахівця;
- Мотивація персоналу на досягнення необхідних результатів;
- Відповідальність кожного співробітника за показники;
- Управління бюджетом, виділеним на оплату праці.

НУБІП України

Види ключових показників ефективності:

- KPI результату – кількісні та якісні показники результату;
- KPI витрат - кількість витрат ресурсів;
- KPI функціонування – наскільки процес виконання відповідає

НУБІП України

встановленому алгоритму;

- KPI продуктивності – похідні показники, що характеризують співвідношення отриманого результату та часу, витраченого на його отримання;

НУБІП України

- KPI ефективності (показники ефективності) – це похідні показники, які характеризують співвідношення отриманого результату витрат ресурсів. [26]

Для цієї магістерської роботи оптимальним є розрахунок ключового показника ефективності результату. Показником по якому розраховується

НУБІП України

KPI є кількість замовлень.

Для розрахунку KPI результату магістерської роботи були використані дані наступних таблиць:

- DateTime_Dim;
- Order_Fact.

НУБІП України

Було проаналізовано всі замовлення по рокам. Для цілі KPI було обрано найбільшу кількість замовлень страв за 2 роки, а задля розуміння чи був досягнутий KPI було вираховане середнє значення замовлень страв за 2 роки

(рис. 27).

НУБІП України

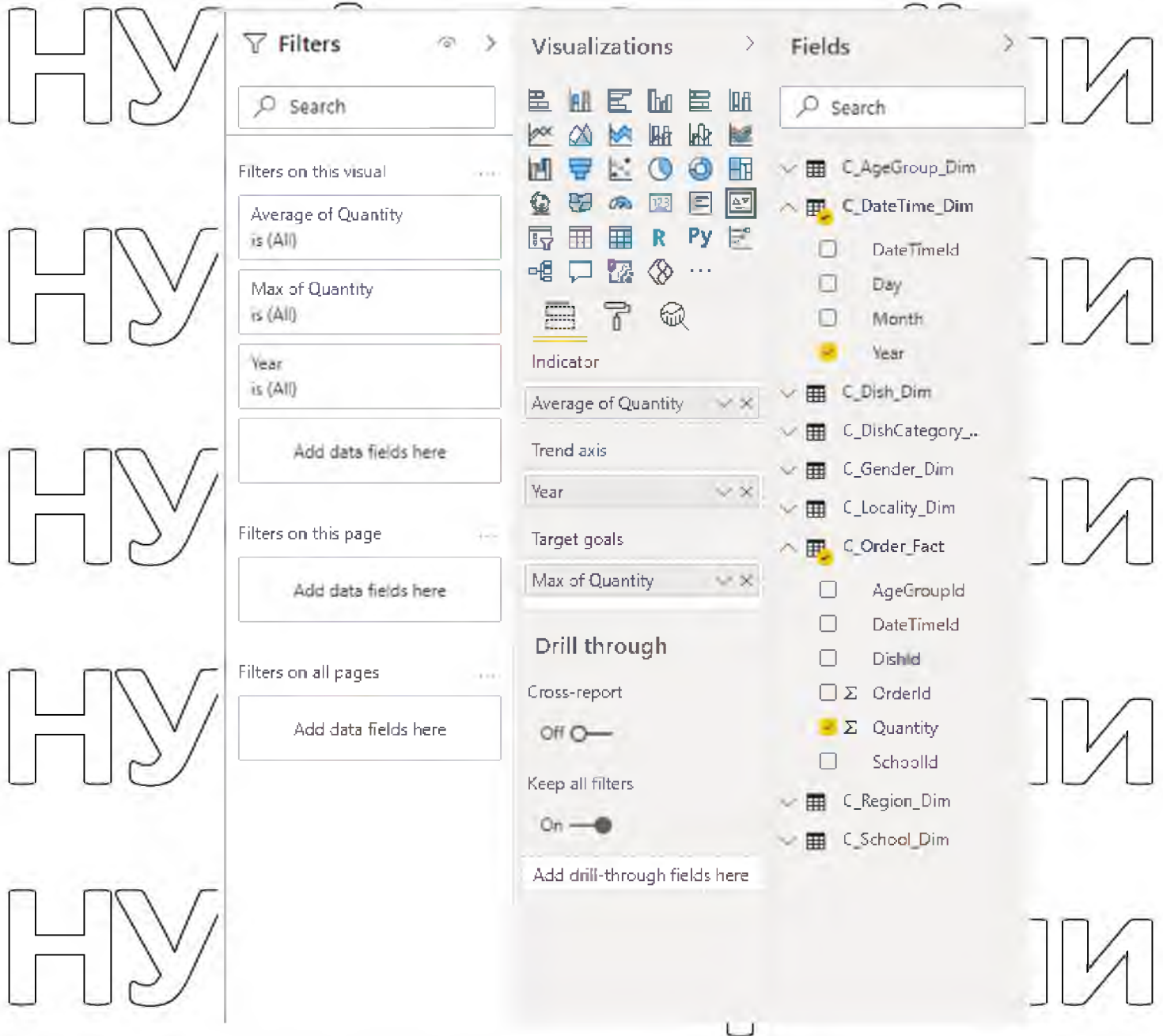


Рис. 27 Задані параметри для розрахунку KPI

Результат розрахунку KPI для кількості замовлених страв за 2020-2021 роки зображено на рис. 28. З даних результатів можна зробити висновок, що ціль в 650 одиниць страв не досягнута так як у середньому значення досягає 388 одиниць, що становить 59,75 відсотків від максимуму. Тобто отримане значення на 40,25 відсотків менше від максимуму і тренд показує спад.



Рис. 28 Розрахунок КРІ

Також розглянемо розрахунок КРІ для окремої області – Закарпатської (рис. 29). Судячи з розрахунку, можна дійти висновку, що ціль в 650 одиниць страв не досягнута так як у середньому значення досягає 387 одиниць що становить 59,56 відсотків від максимуму, тобто отримане значення на 40,44 відсотк в менше від максимуму. Тобто значення в порівнянні з цллю є менше і тренд має спадаючий характер.

НУБІП України

НУБІП України

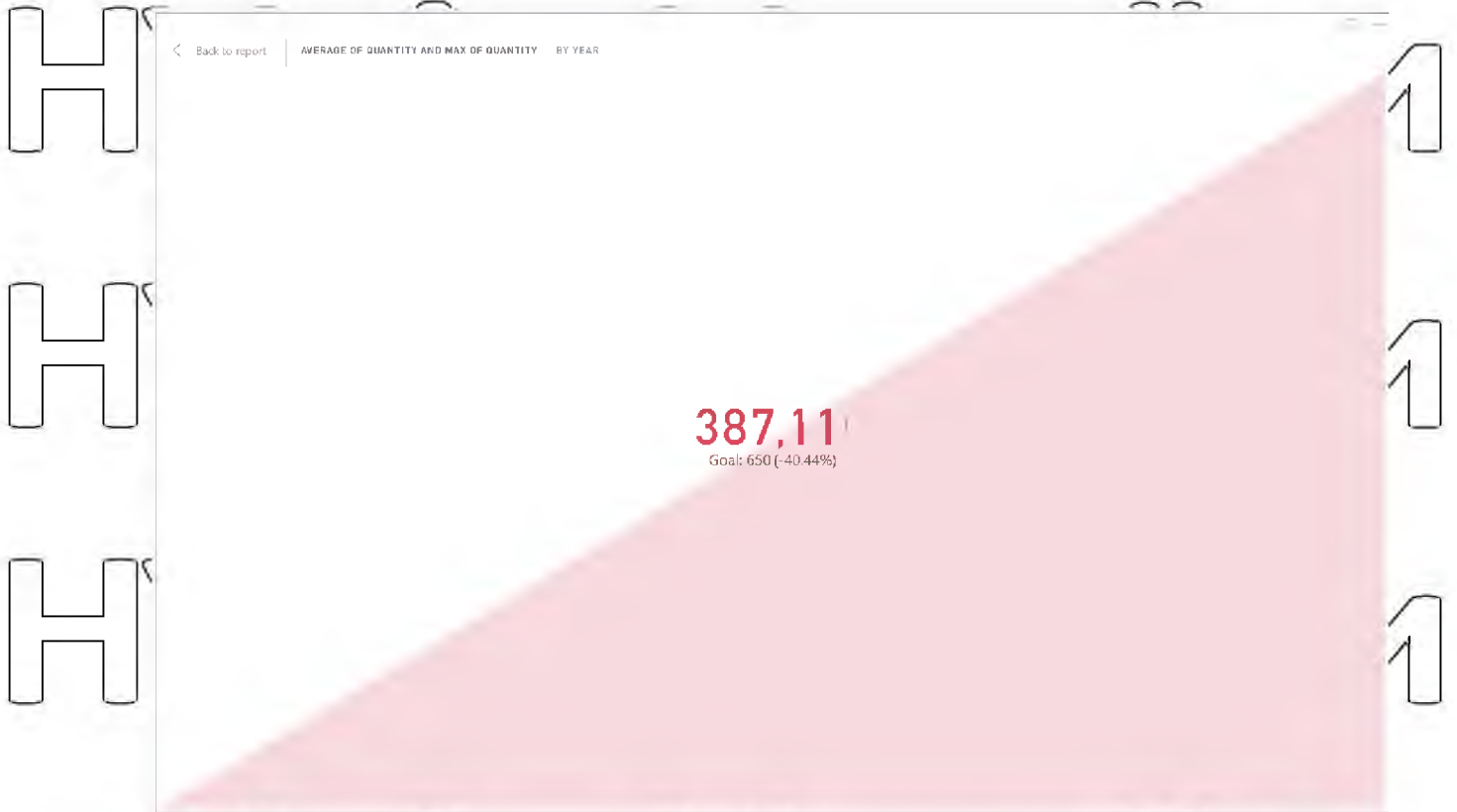


Рис. 29 Розрахунок КРІ для Закарпатської області

НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП України

• ВИСНОВКИ

Метою даної магістерської роботи є аналіз замовлення страв шкільного харчування за певними критеріями для подальшої оптимізації шкільного меню.

Ця підсистема підтримки була розроблена на основі нового проекту про дитяче харчування під патронатом Олени Зеленської та при комунікації зі студентами університету НУБіП України факультету Харчових технологій, які надали інформацію для сховища даних.

У результаті виконання магістерської роботи було проведено:

- Аналіз області дитячого шкільного харчування;
- Розроблено набір діаграм для моделювання предметної області;
- Спроектовано логічну модель, а згодом і саму базу даних у середовищах Erwin Data Modeler та MS SQL Server;
- Створено сховище даних в тому ж MS SQL Server;
- За допомогою середовища MS Power BI створено різні звіти, які відповідають на головні питання до підсистеми та зображають основну інформацію, що знаходиться у сховищі даних;
- Було зображено ключові показники ефективності, які показують чи досягнена ціль по кількості страв замовлень за роки.

Результати проведеної роботи дають змогу будь-якому користувачеві підсистеми використовувати дану систему для аналізу та оптимізації дитячого шкільного харчування в Україні. Система має змогу продумати та розробити меню, ланчі та раціони на наступний період шкільного навчання.

Розроблена підсистема підтримки може підвищити якість системи дитячого харчування. В результаті дослідження було виявлено, що найбільше страв вживаються дівчатами віком 11-13 років. Найчастіше замовляють категорію страв «Друга страва» та «Перша страва». Якщо

НУБІП УКРАЇНИ

говорити про конкретну страву, то найчастіше школярі замовляли борщ серед усіх областей. На другому місці була гречка з маслом.

На мою думку, мета та задача, що була поставлена в даній магістерській роботі була виконана успішно.

НУБІП УКРАЇНИ

НУБІП УКРАЇНИ

НУБІП УКРАЇНИ

НУБІП УКРАЇНИ

НУБІП УКРАЇНИ

НУБІП УКРАЇНИ

СПИСОК ВИКОНАНИХ ДЖЕРЕЛ

1. Про затвердження норм та Порядку організації харчування у закладах освіти та дитячих закладах оздоровлення та відпочинку [Електронний ресурс]. URL: <https://zakon.rada.gov.ua/rada/show/305-2021-%D0%BF#LcN>
2. Про затвердження Норм фізіологічних потреб населення України в основних харчових речовинах і енергії [Електронний ресурс]. URL: <https://zakon.rada.gov.ua/rada/show/z1206-17#n14>
3. Принципи і методи системного аналізу [Електронний ресурс]. URL: <https://stud.com.ua/45001/investuvannya/printsipi-metodi-sistemnogo-analizu>
4. Тема 8. Унифіцированный язык моделирования UML [Електронний ресурс]. URL: <https://samara.mgpu.ru/~dzhadza/dis/15/200.html>
5. 1.3. Диаграмма прецедентов (use case diagram) [Електронний ресурс]. URL: <https://studfile.net/preview/7244123/page:4/>
6. Диаграма діяльності [Електронний ресурс]. URL: <http://5fan.ru/wie/job.php?id=21291>.
7. Проектування програмного забезпечення засобами UML [Електронний ресурс]. URL: http://mmsa.kpi.ua/sites/default/files/disciplines/didkovska_m_v_testing_lecture_3.pdf
8. Моделі і методи проектування інформаційних систем [Електронний ресурс]. URL: <https://elearning.sumdu.edu.ua/free-content/lectured:de1c9452f2a161439391120eef364dd8ce4d8e5e/20160217112601/170352/index.html>
9. Диаграма розгортання [Електронний ресурс]. URL: <https://studfiles.net/preview/5010027/page:6/>

10. Методи створення оптимальної моделі баз даних [Електронний ресурс]. URL: <https://studfile.net/preview/7144845/page:28/>

11. Теорія нормалізації відношень [Електронний ресурс]. URL: <https://studfile.net/preview/7144845/page:28/>

12. Побудова логічної моделі даних [Електронний ресурс]. URL: <https://studfile.net/preview/7144845/page:29/>

13. Erwin Data Modeler [Електронний ресурс]. URL: https://ru.bmstu.wiki/Erwin_Data_Modeler

14. Microsoft SQL Server – реалізація мови програмування SQL [Електронний ресурс]. URL: <http://progopedia.ru/implementation/microsoft-sql-server/>

15. Реляційні СУБД - порівняння MySQL і SQL сервер [Електронний ресурс]. URL: <https://www.hostinger.com.ua/rukovodstva/rejacionnye-subd-sravnenie-mysql-i-sql-server/>

16. OLAP-технології [Електронний ресурс]. URL: <https://kaidev.ru/Pages/Article.aspx?p=OlapAbout>

17. Поняття сховищ даних та основи їх створення [Електронний ресурс]. URL: <https://studfile.net/preview/7144845/page:29/>

18. Технологія сховищ даних Data Warehousing [Електронний ресурс]. URL: <https://studfile.net/preview/5118185/page:31/>

19. Применение SQL Server Integration Services [Електронний ресурс]. URL: http://www.askit.ru/custom/sql2005_admin/m10/10_01_ssis_basics.htm

20. Элементы Data Flow Task [Електронний ресурс]. URL: http://www.askit.ru/custom/sql2005_admin/m10/10_06_02_data_flow_task_elements.htm

21. Аналіз даних і результати дослідження [Електронний ресурс]. URL: <https://sites.google.com/site/marketingdistance/rema-2/2-6-analiz-danhi-i-rezultati-doslidzenna>

22. What is Data Analysis? Research | Types | Methods | Techniques
[Електронний ресурс]. URL: <https://www.guru99.com/what-is-data-analysis.html>

23. Що таке Big Data? [Електронний ресурс]. URL:

<https://futurenow.com.ua/shho-take-big-data-velyki-dani/>

24. Data Mining [Електронний ресурс]. URL:
<https://www.investopedia.com/terms/d/datamining.asp>

25. Что такое Power BI Desktop? [Електронний ресурс]. URL:

<https://docs.microsoft.com/ru-ru/power-bi/fundamentals/desktop-what-is-desktop>

26. Що таке KPI (ключові показники ефективності): приклади показників, мотивація KPI [Електронний ресурс]. URL: <https://bohemia-club.com.ua/posts/uk/buhgalteria-so-take-kpi-klucovi-pokazniki-efektivnosti-prikladi-pokaznikiv-motivacia-kpi.shtml>

НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП України • ДОДАТОК А

НУБІП України

НУБІП України

ЛОГІЧНА МОДЕЛЬ БД

НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП України Сторінка 1

НУБІП України

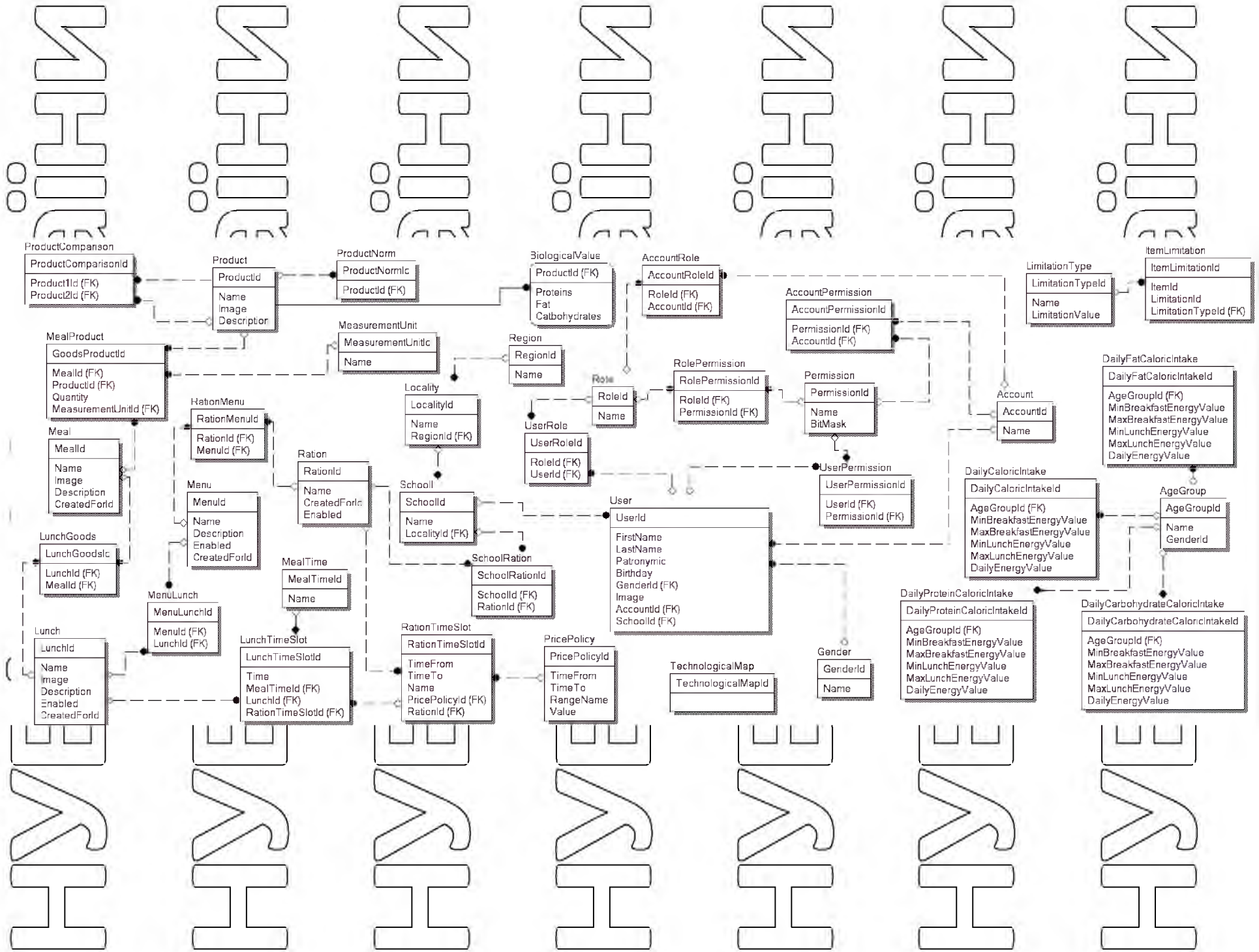
НУБІП України

НУБІП України

НУБІП України

НУБІП України

НУБІП України



НУБІП України • ДОДАТОК Б

НУБІП України

НУБІП України ПУБЛІКАЦІЇ

НУБІП України

НУБІП України

НУБІП України

НУБІП України

УДК 004.42

Підсистема адміністрування системи підтримки прийняття рішень з управління шкільним харчуванням

Понзель Я.Ю., науковий керівник Голуб В.П.

Харчування дітей, які навчають в школі, окрім того, що є дуже серйозна темою, це також складний процес, який потребує ретельного планування, детального розрахунку та клопіткої праці над ним. Незалежно від типу і форми власності закладу освіти планування та організація харчування дітей повинно здійснюватися згідно до Конституції України та законів України «Про освіту», «Про повну загальну середню освіту» та «Про основні принципи та вимоги до безпечності та якості харчових продуктів». Процес цей організовується заради забезпечення учнів енергією та поживними речовинами для різностороннього розвитку, зміцнення здоров'я та покращення успіхів у навчанні. Необхідно враховувати калорійність їжі, яка потрібна для певної вікової групи хлопчиків або дівчат. При плануванні меню враховувати раціон поживних речовин та вітамінів, які необхідні для певної пори року, особливості національної кухні, можливі релігійні обмеження, алергійні протипоказання та індивідуальні смакові переваги школярів.

Вирішення цих, а також не менш важливих інших, питань бере на себе «Система підтримки прийняття рішень з управління шкільним харчуванням». Програмний проєкт дає змогу чітко розпланувати шкільне меню для дітей, щоб забезпечити їх необхідними поживними речовинами протягом всього дня. Система підходить для функціонування на національному рівні, так як дає можливість скласти індивідуальне шкільне меню для кожного окремого освітнього закладу. В системі присутня функція замовлення, яка реалізує змогу вибору індивідуального замовлення меню для дитини, яке буде враховувати його індивідуальні вподобання, нівелювати вміст в продуктах алергічних речовин та добавляти додаткові продукти до меню школяра.

Таким чином, метою даного проєкту являється підвищення ефективності планування та організації шкільного харчування шляхом створення системи підтримки прийняття рішень з управління шкільним харчуванням.

Об'єктом дослідження являється процес планування та організації харчування для учнів різного типу закладів освіти.

Предмет дослідження – це система підтримки прийняття рішень з управління шкільним харчуванням.

Ця велика система потребує адміністрування та постійного контролю даних. Щоб не допустити несанкціонованого доступу до даних, а також обмеження функціональних прав користувачів, в проєкті впроваджено системні ролі, що дозволяють програмі поділяти користувачів на певні підгрупи із індивідуальними правами.

Для реалізації даного проєкту було використано такі засоби: СУБД MS SQL Server Management Studio 2017, SQL Server 2017 Express, мова Transact-SQL.

Для реалізації всього проєкту було використано: середовище розробки Eclipse, мова програмування Java, мова програмування TypeScript та фреймворк Angular.

У майбутньому планується створення окремих програмних підсистем, які б змогли реалізувати функції для наступних користувачів:

- Дієтологи – підготовка норм харчування, формування списку харчових продуктів по групам, розробка принципів та зразків шкільного меню
- Технологи – формування переліку страв та напоїв для різної кратності харчування, відбір продуктів для асортименту шкільних буфетів, розробка технологічних карт разом із кухарями
- Кухарі – підготовка рецептів на відібрані страви та напої, розробка технологічних карт разом із технологами.
- Адміністратори – контролювання даних в системі.

Економісти обрахунок орієнтованої вартості сезонного меню, розробка системи звітності за витрачені кошти.

На рис. 1 представлено діаграму бази даних проекту «Підсистема адміністрування системи підтримки прийняття рішень з управління шкільним харчуванням». Для побудови даної схеми було використано функціональні можливості побудови табличних діаграм на основі програмного коду середовища управління базами даних MS SQL Server Management Studio 2017.

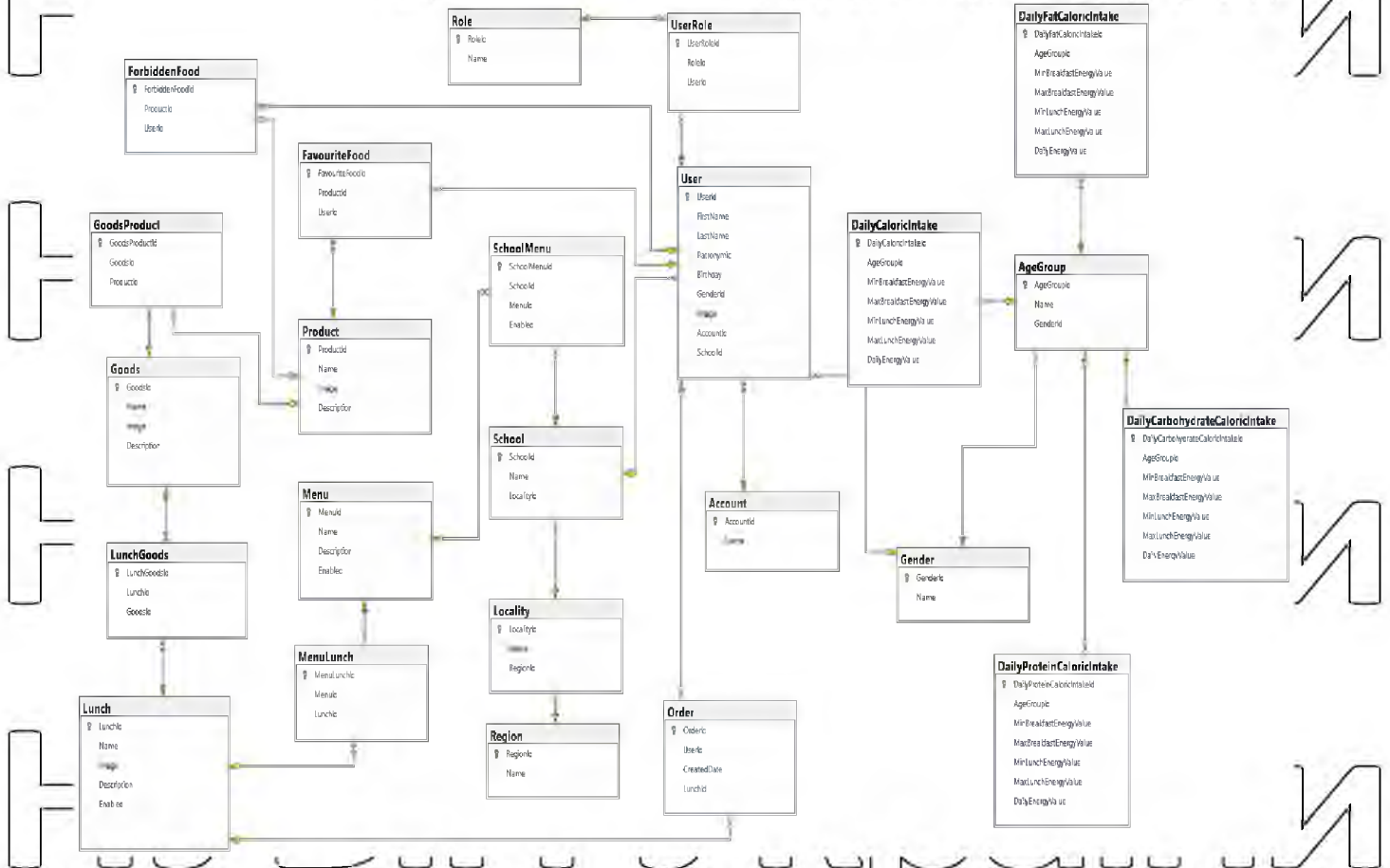


Рис. 1 діаграма бази даних проекту «Підсистема адміністрування системи підтримки прийняття рішень з управління шкільним харчуванням»

СНИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Джоел Сполські. Крайні приклади розробки програмного забезпечення. Видавництво «Пітер». 2007 р. 210ст.
2. Лейн Кемпбелл, Черіл Мейджорс. Бази даних. Інженеринг надійності. Видавництво «Пітер». 2020 р. 304ст.
3. Сергій Тарасов. Системи управління базами даних для програміста. Бази даних зсередини. Видавництво «СОЛОН-Пресс». 2015 р. 320ст

Звітність для підсистеми адміністрування системи підтримки прийняття рішень з управління шкільним харчуванням

Понзель Я.Ю., науковий керівник Голуб Б.Л.

На сьогодні гостро постає питання стосовно дитячого харчування в освітніх закладах. Саме від нього залежить здоров'я дитини. Багато батьків незадоволені харчуванням дітей у школі, а все через погано організовані процеси. Тому з'являється потреба у створенні статистичної звітності, щоб потім на її основі будувати подальші стратегічні дії щодо оптимізації системи шкільного харчування в Україні.

Мета/оптимізація системи шкільного харчування в Україні.

Актуальність: складний процес, який потребує детального планування; планування харчування регулюється певними нормами, яких важливо дотримуватись; важливо розуміти, які продукти найчастіше замовляються дітьми

Для побудови звітності та розрахунку KPI було обрано середовище MS Power BI.

Ключові показники ефективності (KPI) – це візуальні міри ефективності. KPI базується на певному обчислюваному полі й використовується, щоб швидко оцінити поточне значення та стан певного показника порівняно з визначеним цільовим значенням. KPI порівнює ефективність значення, визначеного базовою мірою, з цільовим значенням, яке також визначається мірою або абсолютним значенням.

Так було виконано обчислення KPI для порівняння середньої та максимальної оцінки за певний період часу. Результати обрахування KPI показано нижче (Рис. 1).

Back to report AVERAGE OF QUANTITY AND MAX OF QUANTITY BY YEAR

388,40

Goal: 650 (-40.25%)

Рис. 1 Обчислення KPI для порівняння середньої та максимальної кількості замовлень страв за весь час

Power BI - це аналітичне середовище (комплекс програм і онлайн сервісів), яке дає можливість легкого підключення до будь-якого вивантаження інформації з різних джерел, об'єднання і приведення цієї інформації в єдину стандартизовану модель даних (єдиний інформаційний косяк), обчислення необхідних параметрів і KPI на основі цих об'єднаних даних, побудови візуальних графіків, звітів і дашборда (dashboard)

Сторінка 4

Було створено звіт, де показано залежність вибору категорії страв від регіону (Рис. 2).

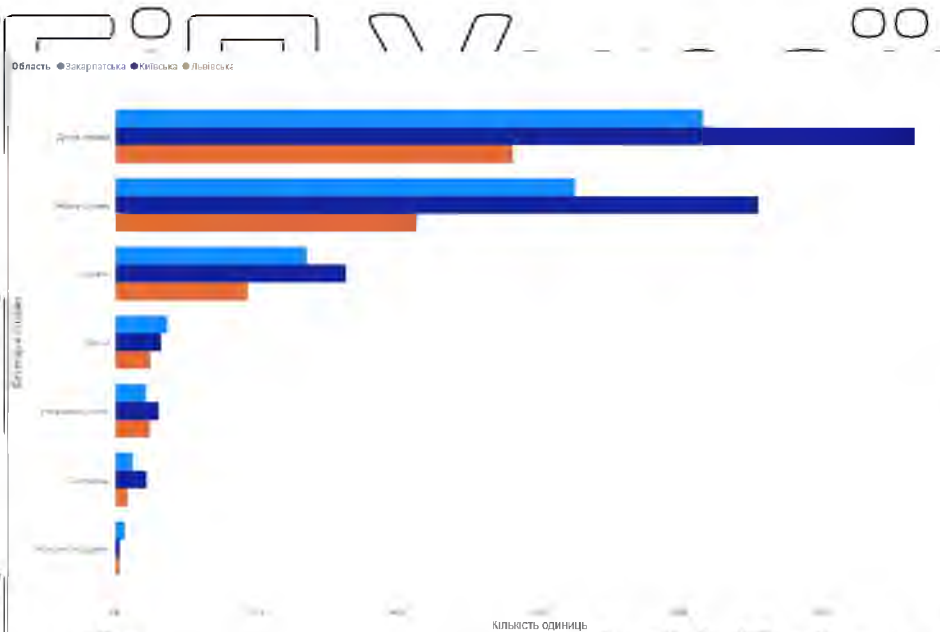


Рис. 2 Звіт "Залежність вибору категорії страв від регіону"

По діаграмі видно, що найпопулярніша страва першої категорії - борщ, а за нею йдуть страви другої категорії гречка з маслом та горохове пюре.

Було також проаналізовано, які страви найчастіше замовляються в залежності від вікової категорії учнів. (Рис. 3).

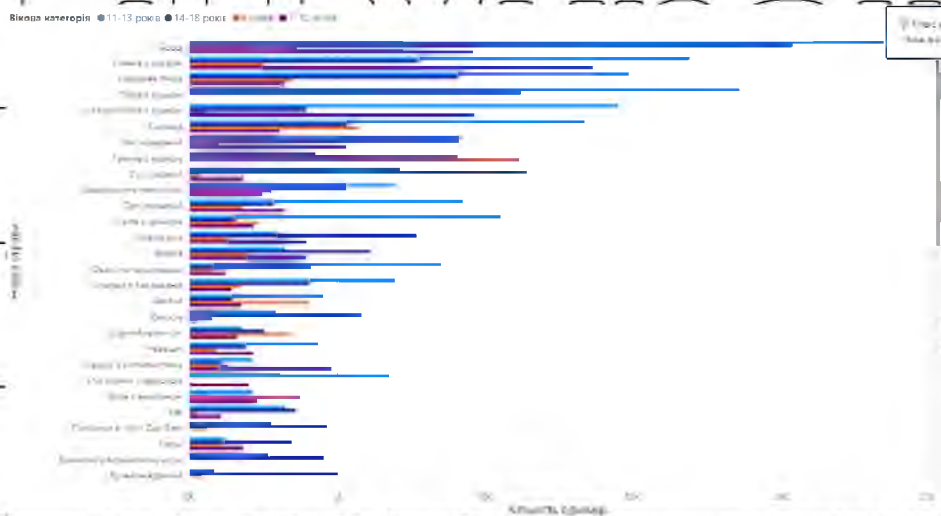


Рис. 3 Звіт "Залежність вибору страв від вікової категорії"

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Бретт Павелл. Mastering Microsoft Power BI: Expert techniques for effective data analytics and business intelligence. Видавництво «Packt Publishing». 2018 р. 638ст.
2. Альберто Феррарі, Марко Руссо. Introducing Microsoft Power BI. Видавництво «Microsoft Press». 2016 р. 306ст.
3. Альберто Феррарі, Марко Руссо. Analyzing Data with Power BI and Power Pivot for Excel. Видавництво «Microsoft Press». 2017 р. 256ст.