

**Київський національний торговельно-економічний університет
Кафедра інженерії програмного забезпечення та кібербезпеки**

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

**«Розробка програмного забезпечення управління рестораним
бізнесом»**

Студента 4 курсу, 7 групи,
спеціальності 121

«Інженерія програмного
забезпечення»

(підпис студента)

Тараканова Євгенія
Олександровича

Науковий керівник
кандидат педагогічних
наук,
ст.викладач

(підпис керівника)

Жирова Тетяна
Олександрівна

Гарант освітньої програми
кандидат технічних наук,
доцент

(підпис керівника)

Цензура Микола
Олександрович

КИЇВ – 2021

ЗМІСТ

ВСТУП.....	3
РОЗДІЛ 1.	5
ТЕОРЕТИЧНІ ВІДОМОСТІ ПРО УПРАВЛІННЯ РЕСТОРАННИМ БІЗНЕСОМ.....	5
1.1. Загальні відомості.....	5
1.2. Технічне завдання.....	6
1.3. Висновки до розділу 1.....	8
РОЗДІЛ 2.	9
ВИМОГИ ДО БАЗИ ДАНИХ УПРАВЛІННЯ РЕСТОРАННИМ БІЗНЕСОМ. РОЗРОБКА БАЗИ ДАНИХ.....	9
2.1. Основні особливості розробки бази даних управління ресторанним бізнесом	9
2.2. Проектування бази даних. Концептуальна, логічна та фізична моделі	10
2.3. Створення бази даних управління ресторанним бізнесом	14
2.4. Створення представлення	18
2.5. Висновки до розділу 2.....	20
РОЗДІЛ 3.	22
РОЗРОБКА СИСТЕМИ УПРАВЛІННЯ РЕСТОРАННИМ БІЗНЕСОМ	22
3.1. Середовище розробки та його інструменти	22
3.2. Вхід у додаток та головна форма.....	22
3.3. Висновки до розділу 3.....	32
ВИСНОВКИ І ПРОПОЗИЦІЇ.....	33
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	35
ДОДАТКИ.....	37

					<i>КНТЕУ 121 07-21.БР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Зав. кафедри</i>	<i>Криворучко О.В.</i>				<i>Розробка програмного забезпечення управління ресторанним бізнесом</i>	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
<i>Керівник</i>	<i>Жирова Т.О.</i>					<i>Зміст</i>	<i>2</i>	<i>36</i>
<i>Гарант</i>	<i>Цензура М. О.</i>					Факультет інформаційних технологій, 4 курс, 7 група		
<i>Розробник.</i>	<i>Тараканов Є.О.</i>							
					Зміст			

ВСТУП

База даних управління ресторанним бізнесом є прототипом сучасної бази даних реалізації тих чи інших процесів будь-якого підприємства.

Ресторани завжди відігравали важливу роль у бізнесі, соціальному, інтелектуальному та мистецькому житті процвітаючого суспільства. Якщо пригадати кафе Парижу в 20-х: три обіди мартіні 50-х та 60-х, замальовувалися ідеї, що змінюються у світі, і плановані революції в ресторанах. Перший контракт Ельвіса в Лас-Вегасі був написаний на скатертині ресторану.

Призначенням випускної кваліфікаційної роботи є створення бази даних управління ресторанним бізнесом, яка надаватиме можливість швидкого доступу до інформації бази даних, отримання потрібних вибірок по таблицях, а також додавання нової і редагування наявної інформації в базі даних (наприклад, інформації про нового працівника, нове замовлення чи нову страву в меню).

Мета дослідження: розробити базу даних реалізації управління ресторанним бізнесом та клієнтський додаток для зручного керування цією базою даних.

Об'єкт: управління діяльністю ресторану.

Предметом дослідження є проектування та розробка бази даних управління ресторанним бізнесом.

Завдання випускної кваліфікаційної роботи містить такі пункти:

1. З'ясування основних положень діяльності ресторанного бізнесу та його управління.
2. Побудувати концептуальну модель бази даних управління ресторанним бізнесом.
3. Спроекувати логічну та фізичну модель бази даних.

					<i>КНТЕУ 121 07-21.БР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Зав. кафедри</i>	Криворучко О.В.				Розробка програмного забезпечення управління ресторанним бізнесом	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
<i>Керівник</i>	Жирова Т.О.					<i>Вступ</i>	3	36
<i>Гарант</i>	Цензура М. О.					Факультет інформаційних технологій, 4 курс, 7 група		
<i>Розробник.</i>	Тараканов Є.О.							
					<i>Вступ</i>			

4. Побудувати базу даних в Microsoft SQL Management Studio шляхом написання відповідних команд.
5. Створити декілька форм представлення інформації бази даних.
6. Написати програму, яка надаватиме доступ до роботи з базою даних через зручний інтерфейс програми.

						Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата	КНТЕУ 121 07-21.БР	
						4

РОЗДІЛ 1.

ТЕОРЕТИЧНІ ВІДОМОСТІ ПРО УПРАВЛІННЯ РЕСТОРАННИМ БІЗНЕСОМ

1.1. Загальні відомості

Більшість закладів ресторанного господарства є невеликими за розмірами, у своєму штаті мають не більше 10 осіб постійних працівників. Управляють такими закладами, як правило, їх власники.

На початку 80-х років ХХ ст. склалася тенденція, що спостерігається й нині. Вона характеризується об'єднанням невеликих закладів ресторанного господарства у мережі та створенням великих конгломератів.

Незалежні заклади ресторанного господарства своєю більшістю представлені "сімейним" невеликим і середнім бізнесом, характерною рисою якого є сезонність операцій.

Звичайним явищем для великих закладів є наявність у штаті, крім посади генерального директора, ще й директора з харчування, директора ресторану, директора з маркетингу, директора з організації прийомів, фінансового директора і т.п. Такий тип управлінської підтримки, якщо навіть він і наявний не в повному обсязі, для закладу, що входить до мережі, може здійснюватися зі штаб-квартири корпорації. Генеральний директор закладу, таким чином може не включати усіх або більшість названих вище фахівців до свого штату.

Зазвичай, найбільш складні проблеми у незалежних закладах виникають з питань бухгалтерського обліку, підбору та розстановки кадрів і контролю за виходом готової продукції. Тому для ефективного управління тут дуже необхідні знання та практичний досвід з питань бухгалтерського обліку та менеджменту.

					<i>КНТЕУ 121 07-21.БР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум</i>	<i>Підпис</i>	<i>Дата</i>	<i>Розробка програмного забезпечення управління ресторанним бізнесом</i>	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
<i>Зав. кафедри</i>		<i>Криворучко О.В.</i>				<i>Р1</i>	<i>5</i>	<i>36</i>
<i>Керівник</i>		<i>Жирова Т.О.</i>				Факультет інформаційних технологій, 4 курс, 7 група		
<i>Гарант</i>		<i>Цензура М. О.</i>						
<i>Розробник</i>		<i>Тараканов Є.О.</i>			Теоретичні відомості про управління ресторанним бізнесом			

Крім зарахування до штату технічного персоналу, який володіє знаннями у цих сферах, приділяється увага специфічній системі оперативного контролю і звітності. Незалежні заклади не мають такої можливості, тому для досягнення великого прибутку закладу генеральний директор повинен сам знайти системний підхід щодо прийняття управлінських рішень.

1.2. Технічне завдання

1. Загальні відомості

1.1.1. Програмне забезпечення для управління ресторанним бізнесом.

Вона допомагатиме вести облік страв меню, замовлень та працівників ресторану.

1.2. Найменування системи

1.2.1. Повне найменування системи «Програмний додаток для управління ресторанним бізнесом»

1.2.2. Скорочене найменування системи

“Conduct Business”

1.3. Планові терміни початку та закінчення робіт

15.01.2020 – квітень 2021.

2. Мета та призначення створення системи

2.1. Призначення системи

Система призначена для полегшення ведення ресторанної справи, а саме: вона вестиме облік замовлень, на основі який надаватиметься аналітична інформація.

2.2. Мета створення системи

Метою створення системи є отримати програмне забезпечення, яке буде зберігати інформацію про страви меню, надавати можливість робити замовлення та зберігати його в базі даних з можливістю переглянути.

						Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата	КНТЕУ 121 07-21.БР	6

3. Вимоги до системи

3.1. Вимоги до системи в цілому

3.1.1. Вимоги до структури та функціонування системи, перелік підсистем

3.1.1.1. Вимоги до способів і засобів інформаційного обміну між компонентами системи

Програмне забезпечення тісно пов'язане з базою даних, оскільки саме в ній зберігається вся інформація про меню, замовлення та працівників.

Комунікація бази даних та програмного додатку відбувається через збережені процедури, функції та запити.

3.1.1.2. Вимоги до режимів функціонування системи

Система має 1 режим функціонування.

3.1.2. Параметри, що характеризують ступінь відповідності системи призначенням

Основним параметром, по якому систему можна вважати відповідною призначенню, є виконання всіх основних функцій програмного забезпечення

3.1.3. Вимоги до експлуатації, технічного обслуговування, ремонту і зберігання компонентів системи

Програмний додаток потребує технічного обслуговування чи ремонту в тому випадку, якщо виникає потреба в доробці нового функціоналу.

3.1.4. Вимоги до складу, структури і способів організації даних в системі

Система складається з вітального вікна, головної форми та форми підрахунку вартості замовлення.

3.1.5. Вимоги до інформаційного обміну між компонентами системи

						Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата	КНТЕУ 121 07-21.БР	
						7

Кожна форма працює з базою даних індивідуально, підключення до бази даних знаходиться в головному класі, а всі інші класи містять лише посилання на нього(для полегшення зміни підключення в майбутньому).

3.1.6. Вимоги до структури процесу збору, обробки, передачі даних в системі представлення даних

Збір, обробка та передача даних в базу даних здійснюється через запити та збережені процедури.

4. Вимоги до програмного забезпечення включають всі вищенаведені вимоги.

5. Вимоги до технічного забезпечення

Програмний застосунок повинен функціонувати на операційній системі Windows.

1.3. Висновки до розділу 1

Ресторанний бізнес розвивається досить стрімко. Це обумовлено тим, що потреба в їжі залишається актуальною, незважаючи на будь-які обставини, будь то науково-технічний прогрес, революція чи щось інше. Саме це і забезпечує актуальність і популярність ведення ресторанного бізнесу. Діяльність ресторану містить дуже багато інформації, яка постійно додається, модифікується чи видаляється. Для забезпечення зручності цих дій постає потреба в створенні бази даних керування ресторанним бізнесом, яка забезпечить швидкий доступ до бази даних.

Було створено технічне завдання, де розписано основні вимоги до програмного забезпечення.

						Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата	КНТЕУ 121 07-21.БР	8

РОЗДІЛ 2.

ВИМОГИ ДО БАЗИ ДАНИХ УПРАВЛІННЯ РЕСТОРАННИМ БІЗНЕСОМ.

РОЗРОБКА БАЗИ ДАНИХ

2.1. Основні особливості розробки бази даних управління ресторанним бізнесом

База даних управління ресторанним бізнесом повинна містити досить багато інформації. Для забезпечення зручної роботи з базою даних база даних повинна бути правильно організованою. Для цього потрібно розділити інформацію про управління ресторанним бізнесом на невеликі складові, наприклад, інформація про працівників, відомості про замовлення та інформація про меню ресторану.

Інформація про працівників повинна включати прізвище, ім'я та по батькові працівника, його мобільний номер телефону, назву посади, досвід роботи та банківський рахунок. Також створюється таблиця посад, на яку і буде посилатися таблиця працівників. База даних повинна забезпечувати зручний доступ до цієї інформації, а також надавати можливість її модифікації та видалення.

Відомості про замовлення мають містити інформацію про страви, які були замовлені, а також час замовлення, час сплати та ім'я офіціанта, який обслуговував клієнта. Для реалізації цієї таблиці необхідно створити ще додаткову, наприклад «Деталі замовлення». Ця таблиця міститиме деталі замовлення, такі як назва страви та кількість. Це забезпечує можливість замовлення більше одного виду їжі.

Меню повинно містити всю необхідну інформацію про їжу, а саме тип їжі, назва, ціна однієї порції, час приготування. Додатково до цієї таблиці створюється таблиця типів їжі(перші страви, другі, напої і т.д.).

Вся вищеперерахована інформація повинна бути реалізована в базі даних та подана через клієнтську програму в зручній формі.

					<i>КНТЕУ 121 07-21.БР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Зав. кафедри</i>	Криворучко О.В.				<i>Розробка програмного забезпечення управління ресторанним бізнесом</i>	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
<i>Керівник</i>	Жирова Т.О.					<i>Р 2</i>	<i>9</i>	<i>36</i>
<i>Гарант</i>	Цензура М. О.					Факультет інформаційних технологій, 4 курс, 7 група		
<i>Розробник.</i>	Тараканов Є.О.							
					<i>Вимоги до бази даних управління ресторанним бізнесом. Розробка бази даних</i>			

2.2 Проектування бази даних. Концептуальна, логічна та фізична моделі

Щоб створити базу даних управління ресторанним бізнесом необхідно спочатку побудувати концептуальну, логічну та фізичну моделі цієї бази даних.



Рис. 2.1. Концептуальна модель БД

На рисунку 2.1 зображено концептуальну модель бази даних управління ресторанним бізнесом. Ця модель відображає основні таблиці бази даних і дає лише загальне уявлення про структуру бази даних. Більш конкретні і повні уявлення про БД дає логічна та фізична моделі.

						Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата	КНТЕУ 121 07-21.БР	10

Фізична модель є удосконаленою версією логічної і створюється шляхом додавання типів даних до атрибутів.

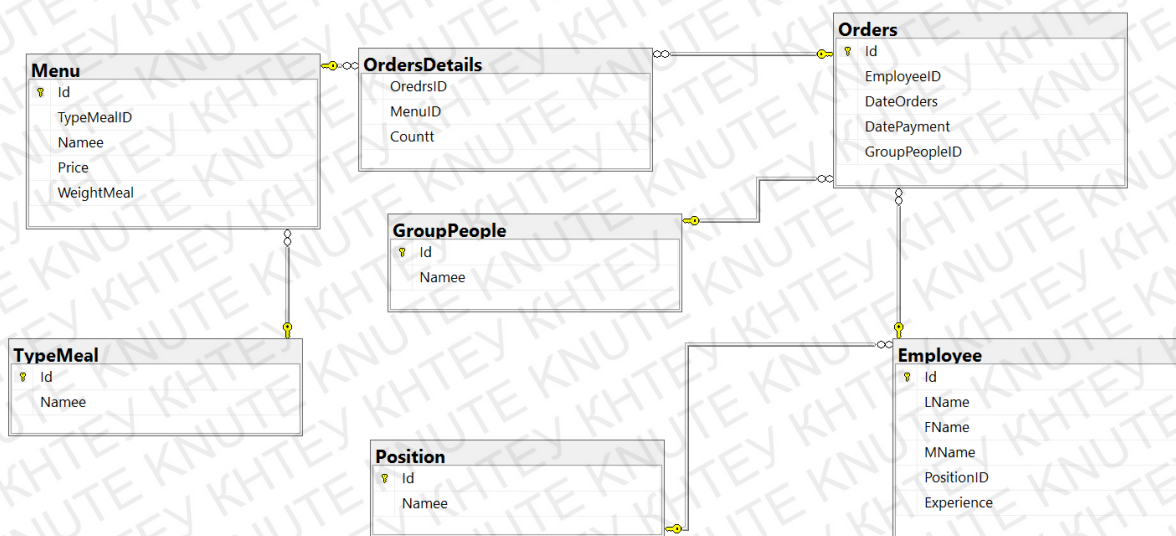


Рис. 2.2. Логічна модель БД

Зображена на рисунку 2.2 логічна модель показує зв'язки між таблицями, а також назви стовпців таблиць. А фізична модель вказує ще й типи даних.

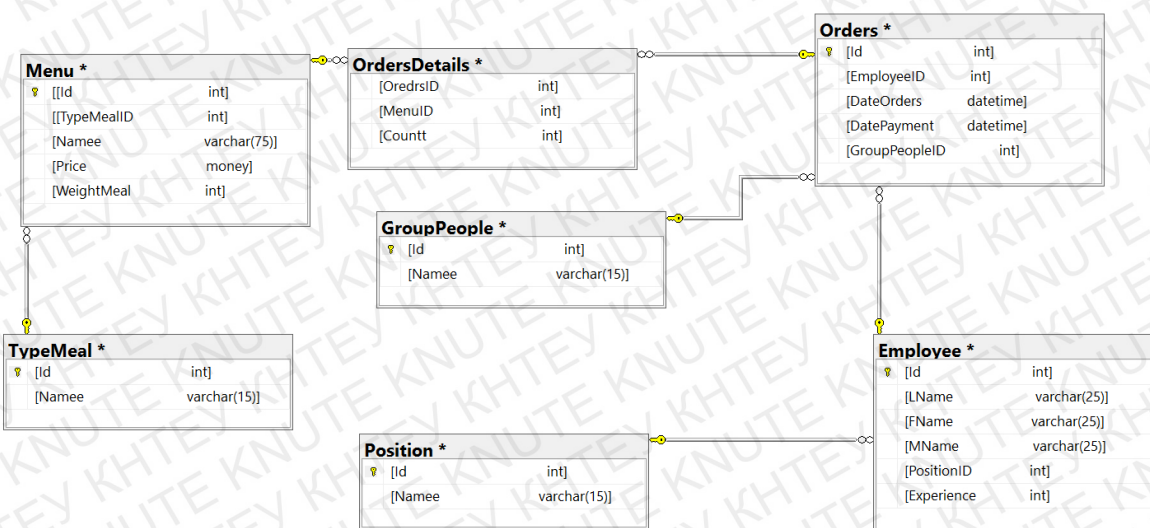


Рис. 2.3. Фізична модель БД

Фізична модель бази даних управління ресторанним бізнесом складається з таких таблиць:

- «TypeMeal» - містить інформацію про типи страв та має такі атрибути:
 - «Id» типу int,
 - «Name» типу varchar(15).
- «Menu» - таблиця меню, яка містить інформацію про всі страви та має такі атрибути:
 - «Id» типу int,
 - TypeMealID» типу int, який має обмеження FOREIGN KEY і посилається на таблицю «TypeMenu» на поле ID,
 - «Name» типу varchar(75),
 - «Price» типу money,
 - «WeightMeal»(вага) типу int.
- «Position» - таблиця, яка містить назви посад ресторану. Атрибутами цієї таблиці є:
 - «Id» типу int,
 - «Name» типу varchar(15).
- «Employee» - таблиця працівників ресторану, атрибутами якої є:
 - «Id» типу int,
 - «LName»(прізвище) типу varchar(25),
 - «FName»(ім'я) типу varchar(25),
 - «MName»(по батькові) типу varchar(25),
 - «PositionID» типу int, який має обмеження FOREIGN KEY і посилається на таблицю «Position».
 - «Experience»(років) типу int.
- «GroupPeople» - таблиця, в якій розміщується інформація про кількість групи людей, які є гостями ресторану. Ця таблиця має такі атрибути:

						Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		12

КНТЕУ 121 07-21.БР

- “Id” типу int,
- “Namee” типу varchar(15).
- «Orders» - таблиця, що містить інформацію про замовлення, а саме такі атрибути:
 - “Id” типу int,
 - “EmployeeID” типу int, який має обмеження FOREIGN KEY і посилається на таблицю “Employee” на поле ID,
 - “DateOrders” типу datetime(час замовлення),
 - “DatePayment” типу datetime(час сплати),
 - “GroupPeopleID” типу int, який має обмеження FOREIGN KEY і посилається на таблицю “GroupPeople” на поле ID.
- «OrdersDetails» - це таблиця, яка містить інформацію про деталі замовлення і містить такі атрибути:
 - “OrderID” типу int, який має обмеження FOREIGN KEY і посилається на таблицю “Orders” на поле ID.
 - “MenuID” типу int, який має обмеження FOREIGN KEY і посилається на таблицю “Menu” на поле ID.
 - “CountMeal” (кількість порцій) типу int.

						Архиви
						КНТЕУ 121 07-21.БР
Зм.	Архиви	№ докум	Підпис	Дата		13

2.3. Створення бази даних управління ресторанним бізнесом

Коли фізична модель є побудованою, починається процес створення бази даних управління ресторанним бізнесом. Щоб створити базу даних, необхідно в SQL Management Studio написати наступний код(рис. 2.4).

```
CREATE DATABASE RestaurantDB
GO
```

Рис. 2.4. Створення бази даних

Після бази даних наступними створюються таблиці цієї бази даних. Потрібно починати створювати ті таблиці, які не мають посилань на інші таблиці. Такими таблицями є «TypeMeal», «Position», «GroupPeople».

Перші три таблиці є простішими за структурою, тому з них і починається створення таблиць.

```
CREATE TABLE TypeMeal
(
    Id int PRIMARY KEY IDENTITY NOT NULL,
    Namee varchar(15) NOT NULL
)

CREATE TABLE GroupPeople
(
    Id int PRIMARY KEY IDENTITY NOT NULL,
    Namee varchar(15) NOT NULL
)

CREATE TABLE Position
(
    Id int PRIMARY KEY IDENTITY NOT NULL,
    Namee varchar(15) NOT NULL
)
-----
```

Рис. 2.5. Створення перших 3 таблиць

Таблиця «TypeMeal» містять поле, на яке посилається таблиця «Menu». Посилання відбувається за допомогою “FOREIGN KEY REFERENCES TypeMeal(column)”. А таблиця «Position» містить поле, на яке посилається таблиця «Employee». Створення цих таблиць зображено на рис. 2.6.

						Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		14

```

CREATE TABLE Menu
(
    Id int PRIMARY KEY IDENTITY NOT NULL,
    TypeMealID int FOREIGN KEY REFERENCES TypeMeal(Id),
    Namee varchar(75) NOT NULL,
    Price money NOT NULL,
    WeightMeal int NOT NULL
)
CREATE TABLE Employee
(
    Id int PRIMARY KEY IDENTITY NOT NULL,
    LName varchar(25) NOT NULL,
    FName varchar(25) NOT NULL,
    MName varchar(25) NOT NULL,
    PositionID int FOREIGN KEY REFERENCES Position(Id) NOT NULL,
    Experience int NOT NULL
)

```

Рис. 2.6. Створення таблиці "Menu" і "Employee"

Таблиця «Orders» має кілька атрибутів, які характеризують замовлення(рис 2.7).

```

CREATE TABLE Orders
(
    Id int PRIMARY KEY IDENTITY NOT NULL,
    EmployeeID int FOREIGN KEY REFERENCES Employee(Id) NOT NULL,
    DateOrders datetime NOT NULL,
    DatePayment datetime NOT NULL,
    GroupPeopleID int FOREIGN KEY REFERENCES GroupPeople(Id) NOT NULL
)

```

Рис. 2.7. Створення таблиці "Orders"

І останньою таблицею бази даних управління ресторанним бізнесом є таблиця «OrdersDetails».

```

CREATE TABLE OrdersDetails
(
    OredrsID int FOREIGN KEY REFERENCES Orders(Id) NOT NULL,
    MenuID int FOREIGN KEY REFERENCES Menu(Id) NOT NULL,
    Countt int NOT NULL
)

```

Рисунок2. 8. Створення таблиці "OrdersDetails"

						Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		15

КНТЕУ 121 07-21.БР

Після закінчення процесу створення таблиць було почато процес наповнення таблиць даними. Для уникнення небажаних помилок варто починати наповнювати ті таблиці, які не мають обмежень типу «Foreign Key»(посилань на інші таблиці). Отже, таблиці наповнюються в такому порядку як і створювалися.

```

INSERT INTO TypeMeal VALUES
('Закуси'),
('Салати'),
('Супи'),
('Сніданки'),
('Піца'),
('Десерти')

INSERT INTO GroupPeople VALUES
('1-2 людини'),
('від 3 до 6'),
('від 7 до 10'),
('від 11 до 20'),
('від 21')

INSERT INTO Position VALUES
('Офіціант'),
('Касир'),
('Хостес'),
('Кухар'),
('Кондитер'),
('Шеф кухар')

```

Рис. 2.9. Наповнення таблиць "TypeMeal", "GroupPeople", "Position"

Після них наповнюється таблиця «Menu».

```

INSERT INTO Menu VALUES--тип, назва, грам, час
(1, 'Тар тар з тунця з кремом із броколі та деревесними грибами', 180, 190),
(1, 'Карпачо з гребінця в соусі Амарило', 150, 190),
(1, 'Крудо з лосося', 200, 190),
(1, 'Смажена хамса з грузинською аджикою', 150, 135),
(1, 'Хумус з баклажанів', 100, 105),
(1, 'Ікра овочева', 180, 145),
(1, 'Паштет смачніше мандарина', 160, 145),
(2, 'Олів-е з хрусткими опеньками та телячим язиком', 280, 145),
(2, 'Олів-е з лососем, гребінцем та креветкою', 280, 195),
(2, 'Цезар з обсмаженим до хрусткої скоринки курчачим', 150, 245),
(2, 'Салат з креветками, з руколюю, авокадо та трюфельною олією', 180, 425),
(2, 'Салат з вугрем, зеленим авокадо та горіховим соусом', 380, 345),
(2, 'Салат грецький з особливими оливками', 200, 145),
(3, 'Суп з морепродуктами', 300, 165),
(3, 'Карпатський суп з білими грибами', 350, 142),
(3, 'Курячий бульйон з потрохами', 350, 135),
(4, 'Омлет з томатами і авокадо', 200, 128),
(4, 'Сирники', 250, 108),
(5, 'Піца з куркою і ананасами', 460, 148),
(5, 'Піца з куркою і ананасами, велика', 1000, 298),
(5, 'Піца Діабло', 460, 168),
(5, 'Піца з грибами', 400, 143),
(5, 'Піца з саламі', 490, 148),
(5, 'Піца Карбонара', 300, 138),

```

Рис. 2.10. Наповнення таблиці «Menu»

					Аркуш
					КНТЕУ 121 07-21.БР
Зм.	Аркуш	№ докум	Підпис	Дата	16

Лишилися наповнити три таблиці, дві з яких є найбільшими в БД. Першою буде створено таблицю «Employee», наступною «Orders».

```

INSERT INTO Employee VALUES ---ПІБ позиція досвід
('Більник', 'Ігор', 'Олександрович', 1, 1),
('Баторик', 'Анатолій', 'Семенович', 1, 3),
('Курілець', 'Назарій', 'Сергійович', 1, 1),
('Житник', 'Олег', 'Олександрович', 1, 3),
('Косило', 'Сергій', 'Євгенович', 1, 2),
('Баторик', 'Анатолій', 'Семенович', 1, 3),
('Колопельник', 'Ольга', 'Віталіївна', 2, 2),
('Воробець', 'Євгеній', 'Ігорович', 5, 2),
('Перегінець', 'Олег', 'Іванович', 4, 3),
('Бідула', 'Анастасія', 'Валеріївна', 3, 2),
('Самусь', 'Олександр', 'Вікторович', 6, 6)
    
```

Рис. 2.11. Додавання записів у таблиці «Employee»

В таблицю «Employee» було додано 11 працівників, а в таблицю «Orders» - 50 ЗАМОВЛЕНЬ.

```

INSERT INTO Orders VALUES
(3, DATEADD(DAY, -4, GETDATE()), DATEADD(DAY, -5, GETDATE()), 5),
(1, DATEADD(DAY, -4, GETDATE()), DATEADD(DAY, -5, GETDATE()), 5),
(1, DATEADD(DAY, -12, GETDATE()), DATEADD(DAY, -13, GETDATE()), 4),
(6, DATEADD(DAY, -17, GETDATE()), DATEADD(DAY, -18, GETDATE()), 5),
(4, DATEADD(DAY, -11, GETDATE()), DATEADD(DAY, -12, GETDATE()), 2),
(4, DATEADD(DAY, -8, GETDATE()), DATEADD(DAY, -8, GETDATE()), 3),
(4, DATEADD(DAY, -16, GETDATE()), DATEADD(DAY, -16, GETDATE()), 5),
(2, DATEADD(DAY, -17, GETDATE()), DATEADD(DAY, -18, GETDATE()), 5),
(4, DATEADD(DAY, -1, GETDATE()), DATEADD(DAY, -2, GETDATE()), 4),
(2, DATEADD(DAY, -14, GETDATE()), DATEADD(DAY, -14, GETDATE()), 3),
(5, DATEADD(DAY, -6, GETDATE()), DATEADD(DAY, -6, GETDATE()), 1),
(4, DATEADD(DAY, -15, GETDATE()), DATEADD(DAY, -15, GETDATE()), 5),
(1, DATEADD(DAY, -2, GETDATE()), DATEADD(DAY, -2, GETDATE()), 5),
(4, DATEADD(DAY, -15, GETDATE()), DATEADD(DAY, -15, GETDATE()), 5)
    
```

Рис. 2.12. Додавання записів у таблиці «Orders»

Наступна таблиця – «OrdersDetails» - є найбільшою за кількістю записів(100). Вона наповнюється аналогічним шляхом.

```

INSERT INTO OrdersDetails VALUES
(7, 6, 4),
(42, 23, 4),
(33, 6, 5),
(32, 22, 5),
(17, 16, 4),
(23, 14, 5),
(50, 13, 3),
(15, 27, 4),
(16, 17, 1),
(47, 13, 5),
    
```

Рис. 2.13. Додавання запитів у таблицю "OrdersDetails"

						Аркуш
						17
Зм.	Аркуш	№ докум	Підпис	Дата	КНТЕУ 121 07-21.БР	

2.4 Створення представлення

Представлення – це спосіб отримання потрібної інформації в деякому вигляді.

```
SELECT Orders.Id, Employee.LName, Orders.DateOrders, Orders.DatePayment
FROM Orders JOIN OrdersDetails ON Orders.Id = OrdersDetails.OredrsID
JOIN Menu ON Menu.Id = OrdersDetails.MenuID
JOIN Employee ON Employee.Id = Orders.EmployeeID
WHERE Menu.Namee = 'Піца з грибами'
```

Рис. 2.14. Створення запиту

Найпростішим типом представлення є вибірка. Наприклад, якщо потрібно отримати інформацію про всі замовлення, які містили конкретну страву, то пишеться такий код(рис.2.14).

В результаті компіляції виводиться такий результат(рис. 2.15).

	Id	LName	DateOrders	DatePayment
1	32	Житник	2020-04-05 10:33:05.567	2020-04-04 10:33:05.567
2	39	Курілець	2020-04-20 10:33:05.567	2020-04-19 10:33:05.567
3	7	Житник	2020-04-08 10:33:05.567	2020-04-08 10:33:05.567
4	39	Курілець	2020-04-20 10:33:05.567	2020-04-19 10:33:05.567

Рис. 2.15. Результат виконання запиту

Наступний тип представлення – процедура. Такий тип представлення до

```
CREATE PROC SearchByMeal
@MealName varchar(70) = NULL
AS
SELECT Orders.Id, Employee.LName, Orders.DateOrders, Orders.DatePayment
FROM Orders JOIN OrdersDetails ON Orders.Id = OrdersDetails.OredrsID
JOIN Menu ON Menu.Id = OrdersDetails.MenuID
JOIN Employee ON Employee.Id = Orders.EmployeeID
WHERE Menu.Namee = @MealName
GO
EXEC SearchByMeal 'Хумус з баклажанів'
```

Рис.2.16. Збережена процедура

					Аркуш
					КНТЕУ 121 07-21.БР
Зм.	Аркуш	№ докум	Підпис	Дата	18

попередньої задачі буде виглядати наступним чином(рис. 2.16).

Результатом компіляції останнього рядка коду, зображеного на рис.2.16 буде такий результат.

	Results	Messages																																								
	<table border="1"><thead><tr><th></th><th>Id</th><th>LName</th><th>DateOrders</th><th>DatePayment</th></tr></thead><tbody><tr><td>1</td><td>45</td><td>Косило</td><td>2020-04-15 13:04:18.913</td><td>2020-04-14 13:04:18.913</td></tr><tr><td>2</td><td>29</td><td>Баторик</td><td>2020-04-14 13:04:18.913</td><td>2020-04-13 13:04:18.913</td></tr><tr><td>3</td><td>5</td><td>Житник</td><td>2020-04-13 13:04:18.913</td><td>2020-04-12 13:04:18.913</td></tr><tr><td>4</td><td>6</td><td>Житник</td><td>2020-04-16 13:04:18.913</td><td>2020-04-16 13:04:18.913</td></tr><tr><td>5</td><td>35</td><td>Житник</td><td>2020-04-06 13:04:18.913</td><td>2020-04-05 13:04:18.913</td></tr><tr><td>6</td><td>27</td><td>Баторик</td><td>2020-04-16 13:04:18.913</td><td>2020-04-15 13:04:18.913</td></tr><tr><td>7</td><td>46</td><td>Курілець</td><td>2020-04-10 13:04:18.913</td><td>2020-04-10 13:04:18.913</td></tr></tbody></table>		Id	LName	DateOrders	DatePayment	1	45	Косило	2020-04-15 13:04:18.913	2020-04-14 13:04:18.913	2	29	Баторик	2020-04-14 13:04:18.913	2020-04-13 13:04:18.913	3	5	Житник	2020-04-13 13:04:18.913	2020-04-12 13:04:18.913	4	6	Житник	2020-04-16 13:04:18.913	2020-04-16 13:04:18.913	5	35	Житник	2020-04-06 13:04:18.913	2020-04-05 13:04:18.913	6	27	Баторик	2020-04-16 13:04:18.913	2020-04-15 13:04:18.913	7	46	Курілець	2020-04-10 13:04:18.913	2020-04-10 13:04:18.913	
	Id	LName	DateOrders	DatePayment																																						
1	45	Косило	2020-04-15 13:04:18.913	2020-04-14 13:04:18.913																																						
2	29	Баторик	2020-04-14 13:04:18.913	2020-04-13 13:04:18.913																																						
3	5	Житник	2020-04-13 13:04:18.913	2020-04-12 13:04:18.913																																						
4	6	Житник	2020-04-16 13:04:18.913	2020-04-16 13:04:18.913																																						
5	35	Житник	2020-04-06 13:04:18.913	2020-04-05 13:04:18.913																																						
6	27	Баторик	2020-04-16 13:04:18.913	2020-04-15 13:04:18.913																																						
7	46	Курілець	2020-04-10 13:04:18.913	2020-04-10 13:04:18.913																																						

Рис. 2.17. Результат виконання процедури

І останній вид представлення – це функції. Наприклад, треба знайти скільки десертів(будь-яких) було замовлено в ресторані за весь час. Потрібна функція виглядає наступним чином:

```
CREATE FUNCTION SumByPeriod(@typeMeal varchar(20))
RETURNS int
AS
BEGIN
RETURN (SELECT COUNT(OrdersDetails.MenuID)
FROM OrdersDetails JOIN Menu ON Menu.Id=OrdersDetails.MenuID
WHERE OrdersDetails.MenuID in (SELECT Menu.Id
FROM Menu JOIN TypeMeal ON TypeMeal.Id=Menu.TypeMealID
WHERE TypeMeal.Namee = @typeMeal))
END
GO

Print dbo.SumByPeriod('Десерти')
```

Рис. 2.18. Створення функції

Після компіляції виводиться результат.

```
23
Completion time: 2
```

Рис. 2.19. Результат виконання функції, кількість замовлених десертів

					Аркуш
					КНТЕУ 121 07-21.БР
Зм.	Аркуш	№ докум	Підпис	Дата	19

Отже було продано 23 десерти. А зараз треба вивести суму грошей, скільки було отримано внаслідок замовлень десертів.

```
EXEC SearchByMeal 'Хумус з баклажанів'  
  
DROP FUNCTION SumByTypeMeal  
CREATE FUNCTION SumByTypeMeal(@typeMeal varchar(20))  
RETURNS int  
AS  
BEGIN  
RETURN (SELECT SUM(Menu.Price)  
FROM OrdersDetails JOIN Menu ON Menu.Id=OrdersDetails.MenuID  
WHERE Menu.ID in (SELECT OrdersDetails.MenuID  
FROM Menu JOIN TypeMeal ON TypeMeal.Id=Menu.TypeMealID  
JOIN OrdersDetails ON Menu.Id=OrdersDetails.MenuID  
WHERE TypeMeal.Name = @typeMeal))  
  
END  
GO  
  
Print dbo.SumByTypeMeal('Десерти')
```

Рис. 2.20. Знаходження суми, на яку було замовлено десертів, за допомогою

Після компіляції виводиться такий результат.

```
-----  
3770  
  
Completion time: 2
```

Рис. 2.21. Результат виконання функції

Отже, було продано десертів на 3770 грн.

2.5 Висновки до розділу 2

Спочатку було сформовано вимоги до бази даних. Вони описують загальну інформацію про базу даних.

Створення бази даних управління ресторанним бізнесом починається з побудови концептуальної, логічної та фізичної моделей. Спочатку створювалася концептуальна модель, на основі якої була побудована логічна. Логічна модель зображує всі таблиці, їх стовпці та зв'язки між таблицями. Фізична модель крім всього раніше перерахованого показує ще й типи даних стовпців таблиць.

					Аркуш
					КНТЕУ 121 07-21.БР
Зм.	Аркуш	№ докум	Підпис	Дата	20

Після побудови моделей було створено базу даних, створено та наповнено таблиці. Потім постала потреба в зручному представленні інформації бази даних. Тому було створено функції і процедури, які приймають в якості параметрів якусь інформацію і виводять результат, який відповідає фільтрам-параметрам.

Деякі інші представлення будуть описані в 3 розділі, тому що саме цей розділ описує розробку програмного забезпечення.

						Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата	КНТЕУ 121 07-21.БР	
						21

РОЗДІЛ 3.

РОЗРОБКА СИСТЕМИ УПРАВЛІННЯ РЕСТОРАННИМ БІЗНЕСОМ

3.1. Середовище розробки та його інструменти

Програмне забезпечення створюватиметься на платформі .NET на мові програмування C#. Середовище розробки Visual Studio надає безліч рішень, серед яких можна знайти Windows Form. Це десктопні додатки, які працюють на операційній системі Windows.

Панель елементів Visual Studio містить безліч інструментів, які можна помістити на форму, для того щоб користувач взаємодіяв з функціоналом через інтерфейс додатку – елементи форми.

Всі елементи мають цілий ряд властивостей і подій. Для прикладу колір, шрифт тексту, ширина, висота, розташування, а також безліч подій: натискання, подвійне натискання, змінення значення, позначення відміток, переміщення та інше.

Все це дозволяє створювати потужні системи на базі рішення Windows Form.

3.2. Вхід у додаток та головна форма

Після відкриття програми користувача зустрічає вітальне вікно, яке відображається 3 секунди та виглядає так як на рис 3.1.

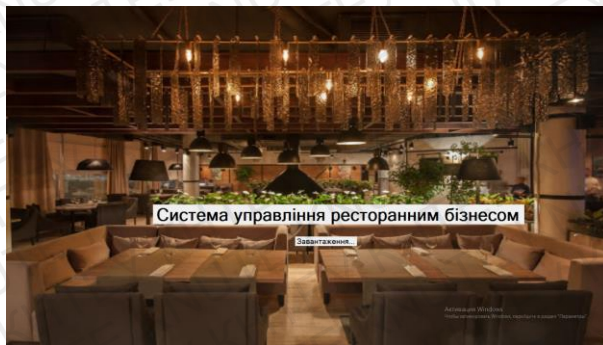


Рис.3.1. Відкриття додатку

					<i>КНТЕУ 121 07-21.БР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум</i>	<i>Підпис</i>	<i>Дата</i>				
<i>Зав. кафедри</i>	Криворучко О.В.				<i>Розробка програмного забезпечення управління ресторанним бізнесом</i>	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
<i>Керівник</i>	Жирова Т.О.					<i>Р 3</i>	<i>22</i>	<i>36</i>
<i>Гарант</i>	Цензура М. О.					Факультет інформаційних технологій, 4 курс, 7 група		
<i>Розробник.</i>	Тараканов Є.О.							
					<i>Розробка системи управління ресторанним бізнесом</i>			

Після трьох секунд відображення вітального вікна, поточне вікно закривається і відкривається вікно «Головна форма»(рис.3.2).

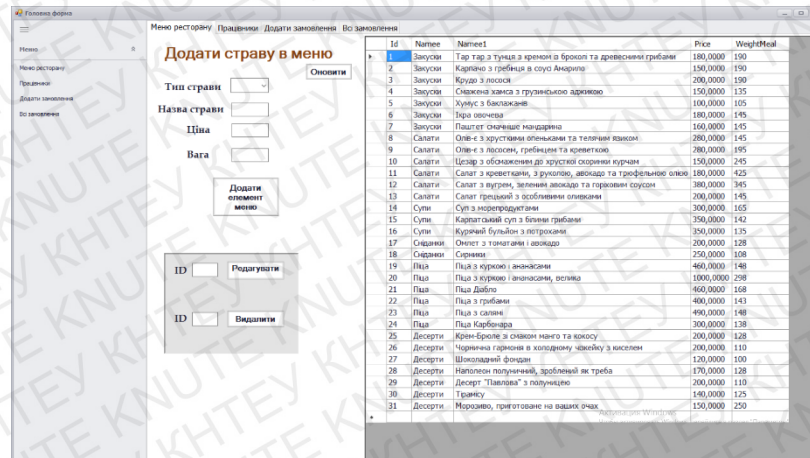


Рис.3.2. Головна форма

На цій формі зліва є меню форми, на якому є клавіші для переходу до відповідних вкладок:

- Меню ресторану
- Працівники
- Додати замовлення
- Всі замовлення

Зверху форми дублюються ці вкладки, тому користувачі можуть обрати більш зручний спосіб для себе.

Форма відображається на весь екран, щоб зменшити/закрити/згорнути варто скористатися клавішами у верхньому правому кутку(рис.3.3).



Рис.3.3. Клавіші закриття/згортання/зменшення розміру форми

Перша вкладка – меню ресторану. Справа відображається таблиця всіх страв. Ця таблиця відображається завдяки процедурі(рис. 3.4).

						Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		
КНТЕУ 121 07-21.БР						23

```

CREATE PROC ShowMenu
AS
SELECT Menu.Id, TypeMeal.Nameee, Menu.Nameee, Menu.Price, Menu.WeightMeal
FROM Menu JOIN TypeMeal ON TypeMeal.Id=Menu.TypeMealID
GO

EXEC ShowMenu

```

Рис.3.4. Процедура відображення даних у вигляді таблиці

Цю процедуру було підключено до програми за допомогою наступного коду(рис. 3.5).

```

public void ShowMenu()
{
    using (SqlConnection sqlConnection = new SqlConnection(connectionString))
    {
        sqlConnection.Open();
        SqlCommand command = new SqlCommand("ShowMenu", sqlConnection);
        command.CommandType = CommandType.StoredProcedure;
        DataTable dt = new DataTable();
        dt.Load(command.ExecuteReader());
        dataGridView1.DataSource = dt;
        sqlConnection.Close();
    }
}

```

Рис. 3.5. Код, який реалізовує процедуру на рис. 3.4

Додавання страви в меню відбувається шляхом заповнення полів, розміщених на цій панелі та натиснення клавіші «Додати». Код, який забезпечує додавання записів у таблицю «Меню» зображений на рис. 3.6.

```

using (SqlConnection sqlConnection = new SqlConnection(connectionString))
{
    sqlConnection.Open();
    SqlCommand command = new SqlCommand("INSERT INTO [Menu] VALUES (@typeMeal, @name, @price, @weight)", sqlConnection);

    command.Parameters.AddWithValue("typeMeal", typeMeal);
    command.Parameters.AddWithValue("name", name);
    command.Parameters.AddWithValue("price", price);
    command.Parameters.AddWithValue("weight", weight);

    await command.ExecuteNonQueryAsync();
}

```

Рис. 3.6. Процедура додавання нових записів у таблицю

						Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		24

КНТЕУ 121 07-21.БР

Редагування та видалення відбуваються аналогічним шляхом за винятком тієї відмінності, що при редагуванні необхідно заповнювати крім полів для додавання, ще й поле ID, а для видалення лише поле ID й треба.

Зверху поточного вікна розміщується клавіша «Оновити»(рис. 3.7), по натисненні на яку, таблиця меню оновлюється і відображає всю актуальну

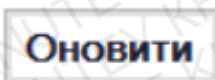


Рис.3.7. Клавіша оновлення меню

інформацію про страви.

Наступна вкладка – працівники(рис.3.8). На цій вкладці немає можливості змінювати інформацію про працівників з міркувань безпеки. Ці дії може виконати адміністратор системи ресторану безпосередньо в SQL Server Management Studio.

Наступна вкладка – додавання замовлення(рис.3.9).

Для забезпечення замовлення більше однієї страви на одне замовлення було

Id	LName	FName	MName	PositionID	Experience
1	Більник	Ігор	Олександр...	1	1
2	Баторик	Анатолій	Семенович	1	3
3	Курілець	Назарій	Сергійович	1	1
4	Житник	Олег	Олександр...	1	3
5	Косило	Сергій	Євгенович	1	2
6	Баторик	Анатолій	Семенович	1	3
7	Колопельник	Ольга	Віталівна	2	2
8	Воробець	Євгеній	Ігорович	5	2
9	Перегінець	Олег	Іванович	4	3
10	Бідула	Анастасія	Валерівна	3	2
11	Самусь	Олександр	Вікторович	6	6

Рис.3.8. Працівники ресторану

створено дві таблиці, які організували цей процес. Перша таблиця «Orders» містить загальні дані про замовлення, а саме ID замовлення, офіціанта, який обслуговував гостей, дату замовлення та дату сплати, а також кількість гостей.(рис 3.9 зліва).

						Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		25

Меню ресторану Працівники Додати замовлення Всі замовлення

Додати замовлення

Дата замовлення: 10 марта 2021

ID групи людей:

ID працівника:

Додати деталі замовлення

ID замовлення:

ID страви:

Кількість:

Id	Працівник	Дата	Група_людей	Замовлення	ID_Меню	Страва
62	Баторик	10.03.2021 11:51	від 3 до 6	62	2	Карпачо з гребінця в соусі Амарило
63	Баторик	10.03.2021 11:52	1-2 людини	62	5	Хумус з баклажанів
64	Курілець	10.03.2021 18:31	від 7 до 10	62	1	Тар тар з тунця з кремом із броколі та деревесними гриба
				63	2	Карпачо з гребінця в соусі Амарило
				63	4	Смажена хамса з грузинською аджікою
				63	30	Тірамісу
				64	3	Крудо з лосося

Рис.3.9. Працівники ресторану

Друга таблиця містить детальні відомості про замовлення, а саме страви, які були замовлені, та їх кількість.(рис 3.9 справа).

Для додавання замовлення чи деталей замовлення треба заповнити відповідні поля і натиснути клавішу «Додати».

Ця вкладка відображає всі замовлення за попередні 24 години. Це досить зручно, оскільки відображаються лише потрібні замовлення, а всі замовлення можна переглянути в наступній вкладці.

Це забезпечують процедури ShowOrders(рис.3.10) і ShowOrdersDetails.

```

1 USE [RestaurantBusinessDB]
2 GO
3 /***** Object: StoredProcedure [dbo].[ShowOrders]    Script Date: 11.03.2021 10:35:23 *****/
4 SET ANSI_NULLS ON
5 GO
6 SET QUOTED_IDENTIFIER ON
7 GO
8 ALTER PROCEDURE [dbo].[ShowOrders]
9 AS
10 SELECT Orders.Id, Employee.LName as Працівник, Orders.DateOrders as Дата, GroupPeople.Name as Група_людей
11 FROM Orders JOIN Employee ON Employee.Id=Orders.EmployeeID
12 JOIN GroupPeople ON GroupPeople.Id=Orders.GroupPeopleID
13 where Orders.DateOrders between DATEADD(DAY, -1, GETDATE()) and SYSDATETIME()
14

```

Рис.3.10. Процедура відображення замовлень

Код підключення цих процедур знаходиться в додатку А.

Ця процедура має фільтр “Where”, де відбувається відбір по даті за мовлення. Аналогічно і процедура ShowOrdersDetails також має фільтр по даті(рис.3.11).

```

1 USE [RestaurantBusinessDB]
2 GO
3 /***** Object: StoredProcedure [dbo].[ShowOrdersDetails] Script Date: 11.03.2021 10:39:09 *****/
4 SET ANSI_NULLS ON
5 GO
6 SET QUOTED_IDENTIFIER ON
7 GO
8 ALTER PROCEDURE [dbo].[ShowOrdersDetails]
9 AS
10 SELECT Orders.Id AS Замовлення, Menu.Id AS ID Меню, Menu.Namee AS Страва, OrdersDetails.Countt AS Кількість
11 FROM Orders JOIN OrdersDetails ON OrdersDetails.OredrsID = Orders.Id
12 JOIN Menu ON Menu.Id = OrdersDetails.MenuID
13 where Orders.DateOrders between DATEADD(DAY, -1, GETDATE()) and SYSDATETIME()
14 ORDER BY Orders.Id

```

Рис.3.11. Процедура відображення деталей замовлень

Щоб отримати весь рахунок по замовленню необхідно натиснути клавішу «Рахунок». Після натиснення відкривається наступна форма(рис.3.12).

Рис.3.12. Рахунок

Щоб отримати всі страви по замовленню, треба ввести айді та натиснути клавішу «ОК»(рис.3.13).

						Аркуш
						27
Зм.	Аркуш	№ докум	Підпис	Дата	КНТЕУ 121 07-21.БР	

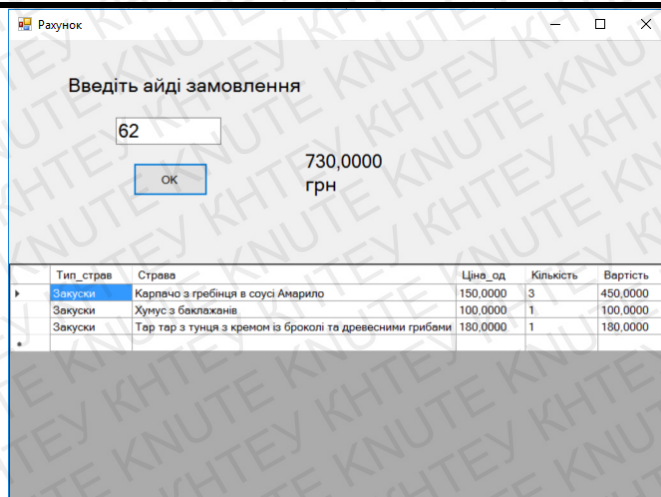


Рис.3.13. Рахунок на конкретне замовлення

```
ALTER PROCEDURE [dbo].[GetAllMealsByOrders]
    @Id int
AS
BEGIN
    select TypeMeal.Nameee as Тип_страви, Menu.Nameee as Страва, Menu.Price as Ціна_од,
        OrdersDetails.Countt as Кількість, Menu.Price*OrdersDetails.Countt as Вартість
    from Orders join OrdersDetails on Orders.Id=OrdersDetails.OredrsID
        join Menu on Menu.Id = OrdersDetails.MenuID
        join TypeMeal on TypeMeal.Id=Menu.TypeMealID
    where Orders.Id = @Id
END
```

Рис.3.14. Процедура, що відображає список страв замовлення

В таблиці відображаються такі стовпці як: тип страви, назва страви, ціна за одиницю, кількість та вартість, яка дорівнює добутку двох попередніх стовпців.

Тут для відображення списку страв використовується процедура, код якої зображено на рис.3.14.

Ця процедура приймає один параметр – це айді замовлення, по якому, власне, і проводиться пошук в таблиці замовлень всіх страв.

						Аркуш
						28
Зм.	Аркуш	№ докум	Підпис	Дата		

КНТЕУ 121 07-21.БР

Для відображення загальної вартості замовлення використовується інша процедура(рис.3.15).

```
ALTER PROCEDURE [dbo].[GetPriceByOrders]
    @Id int
AS
BEGIN
    select sum(Menu.Price*OrdersDetails.Counttt) as Загальна_вартість
    from Orders join OrdersDetails on Orders.Id=OrdersDetails.OredrsID
    join Menu on Menu.Id = OrdersDetails.MenuID
    join TypeMeal on TypeMeal.Id=Menu.TypeMealID
    where Orders.Id = @Id
END
```

Рис.3.15. Процедура, що відображає загальну вартість замовлення

Ця процедура як і попередня приймає параметр айді замовлення, але виводить не перелік страв за цим айді, а суму їх вартостей.

Код цієї форми знаходиться в додатку Б.

Остання вкладка відображає всі замовлення(рис.3.16).

Рис.3.16. Процедура, що відображає реєстр замовлень

Для прикладу, якщо потрібно отримати всі замовлення за 2020 і 2021 рік, то треба встановити відповідні дати і натиснути клавішу «Фільтрувати»(рис.3.17).

						Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата		29

Меню ресторану Працівники Додати замовлення Всі замовлення

Відфільтрувати по

даті замовлення з 1 января 2020 по 11 марта 2021

типу страви

назві страви

ПІБ працівника

Фільтрувати

Id	Дата	Тип_страви	Страва
25	22.04.2020 13:04	Салати	Салат грецький з особливими
7	08.04.2020 13:04	Закуси	Паштет смачніше мандарина
24	05.04.2020 13:04	Десерти	Шоколадний фондан
7	08.04.2020 13:04	Салати	Олів-е з хрусткими опеньками
21	19.04.2020 13:04	Піца	Піца з куркою і ананасами
6	16.04.2020 13:04	Закуси	Тар тар з тунця з кремом із б
43	05.04.2020 13:04	Десерти	Морозиво, приготоване на ва
46	10.04.2020 13:04	Закуси	Хумус з баклажанів
36	05.04.2020 13:04	Десерти	Тірамісу
19	19.04.2020 13:04	Закуси	Смажена хамса з грузинськок
50	25.04.2020 15:50	Закуси	Карпачо з гребінця в соусі Ам
55	19.05.2020 13:47	Закуси	Тар тар з тунця з кремом із б
58	18.05.2020 14:00	Закуси	Тар тар з тунця з кремом із б
60	09.03.2021 18:08	Закуси	Смажена хамса з грузинськок
60	09.03.2021 18:08	Закуси	Ікра овочева
61	09.03.2021 18:21	Салати	Салат з креветками, з руколо
64	10.03.2021 18:31	Закуси	Крудо з лосося
62	10.03.2021 11:51	Закуси	Карпачо з гребінця в соусі Ам
62	10.03.2021 11:51	Закуси	Хумус з баклажанів
63	10.03.2021 11:52	Закуси	Карпачо з гребінця в соусі Ам
63	10.03.2021 11:52	Закуси	Смажена хамса з грузинськок

Рис.3.17. Замовлення від 01.01.2020 до 11.03.2021

Також можна встановити фільтр по типу страви, назві страви та ПІБ

Меню ресторану Працівники Додати замовлення Всі замовлення

Відфільтрувати по

даті замовлення з 1 января 2020 по 11 марта 2021

типу страви Салати

назві страви цезар

ПІБ працівника

Фільтрувати

Id	Дата	Тип_страви	Страва	Ціна	Кіль
39	20.04.2020 13:04	Салати	Цезар з обсмаженим до хрусткої скоринки курчам	150,0000	2
41	16.04.2020 13:04	Салати	Цезар з обсмаженим до хрусткої скоринки курчам	150,0000	4
37	12.04.2020 13:04	Салати	Цезар з обсмаженим до хрусткої скоринки курчам	150,0000	3
17	21.04.2020 13:04	Салати	Цезар з обсмаженим до хрусткої скоринки курчам	150,0000	1

Рис.3.18. Замовлення від 01.01.2020 до 11.03.2021, тип страви салат, назва містить слово «цезар»

працівника, який обслуговував гостей(рис.3.18).

Якщо посунути бігунець, то можна побачити стовпці, які зараз не відображаються. Це кількість і ПІБ працівника.

Цей функціонал забезпечує процедура групування(рис.3.19).

					Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата	30

КНТЕУ 121 07-21.БР

```

ALTER PROCEDURE [dbo].[Groupingg]
    @dateStart datetime,
    @dateFinish datetime,
    @typeMeal varchar(30) = '',
    @Employee varchar(100)='',
    @meal varchar(100)=''
AS
BEGIN
    select Orders.Id, Orders.DateOrders as Дата, TypeMeal.Namee as Тип_страви,
        Menu.Namee as Страва, Menu.Price as Ціна, OrdersDetails.Countt as Кількість,
        Employee.LName+' '+Employee.FName as Працівник
    from Orders join OrdersDetails on Orders.Id=OrdersDetails.OredrsID
    join Menu on Menu.Id = OrdersDetails.MenuID
    join GroupPeople on GroupPeople.Id=Orders.GroupPeopleID
    join TypeMeal on TypeMeal.Id=Menu.TypeMealID
    join Employee on Employee.Id=Orders.EmployeeID
    where Orders.DateOrders between @dateStart and @dateFinish
    and TypeMeal.Namee like '%'+@typeMeal+'%'
    and ((Employee.LName like '%'+@Employee+'%' or (Employee.FName like '%'+@Employee+'%'))
    or (Employee.MName like '%'+@Employee+'%'))
    and Menu.Namee like '%'+@meal+'%'
END

```

Рис.3.19. Процедура фільтрації замовлень

Ця процедура приймає параметри: дата замовлення від і до, тип страви, ПІБ працівника та назва страви. Повертає процедура основну інформацію про замовлення: айді замовлення, дата, тип страви, назва страви, кількість та ПІБ працівника.

Код форми реєстру всіх замовлень знаходиться в додатку В.

						Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата	КНТЕУ 121 07-21.БР	
					31	

3.3. Висновки до розділу 3

Отже, першим кроком до створення програми було обрання необхідних засобів для розробки програмного забезпечення. Програма була написана на платформі .NET мовою програмування C#.

Головне вікно відкривається одразу після запуску програми і відображається 3 секунди, потім його змінює наступне вікно, яке є головною формою. З її назви стає зрозумілим ця форма містить основний функціонал програми.

Перша вкладка головної форми – меню. На цій вкладці відображається все меню ресторану, а також можна здійснювати різні операції з меню, такі як: додавання, редагування та видалення записів з меню.

За цієї вкладкою розташовується реєстр працівників. Він поданий лише на читання.

Наступна вкладка – створення замовлень. Ця вкладка містить відомості про замовлення. Інформація про замовлення була розподілена на дві таблиці, що забезпечило програму можливістю додавання більше однієї страви на одне замовлення(на одне ID).

Остання вкладка – реєстр всіх замовлень з можливістю відбору потрібних замовлень.

В результаті – програма з хорошим функціоналом та приємним інтерфейсом.

						Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата	КНТЕУ 121 07-21.БР	32

ВИСНОВКИ І ПРОПОЗИЦІЇ

Початком написання випускної кваліфікаційної роботи було визначення та формулювання актуальності предметної області. Ресторанний бізнес дійсно користується великим попитом, оскільки потреба в їжі була, є і буде завжди.

Наступним етапом було моделювання бази даних. Спочатку було побудовано концептуальну модель, яка дає загальне уявлення про базу даних керування ресторанним бізнесом. Наступною моделлю була логічна модель, яка показує всі стовпці таблиць, а також зв'язки між таблицями. Завершальною моделлю була фізична модель, яка вказує типи даних атрибутів таблиць.

Після того коли остання, фізична, модель була побудована, було створено базу даних, її таблиці, а також наповнення цих таблиць. Для зручного представлення інформації бази даних було створено кілька представлень інформації БД реалізації керування ресторанним бізнесом. Одним з таких представлень є підрахунок вартості замовлення за його айді. Ця та інші функції були реалізовані в програмі, що зробило її достатньо зручною для користування.

В майбутньому передбачається доповнення бази даних схожими функціями, а саме:

1. Функція, яка підраховуватиме кількість замовлень, який здійснив певний офіціант.
2. Функція, яка знаходить проміжки часу, коли замовлень найбільше.
3. Функція, яка показує замовлення ресторану за певний період(частково реалізовано).
4. Функція, яка підраховує суму замовлень за певний період і тд.

					<i>КНТЕУ 121 07-21.БР</i>			
<i>Зм.</i>	<i>Аркуш</i>	<i>№ докум</i>	<i>Підпис</i>	<i>Дата</i>	<i>Розробка програмного забезпечення управління ресторанним бізнесом</i>	<i>Стадія</i>	<i>Аркуш</i>	<i>Аркушів</i>
<i>Зав. кафедри</i>	<i>Криворучко О.В.</i>					<i>ВП</i>	<i>33</i>	<i>36</i>
<i>Керівник</i>	<i>Жирова Т.О.</i>					<i>Факультет інформаційних технологій, 4 курс, 7 група</i>		
<i>Гарант</i>	<i>Цензура М. О.</i>							
<i>Розробник.</i>	<i>Тараканов Є.О.</i>				<i>Висновки та пропозиції</i>			

Справді, база даних керування діяльністю ресторанного бізнесу може розширювати свій функціонал далі і далі, в залежності від потреб певного ресторану.

						Аркуш
Зм.	Аркуш	№ докум	Підпис	Дата	КНТЕУ 121 07-21.БР	
						34

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. А. Д. Хомоненко – Бази даних/ Посібник для вищих навчальних закладів. 2009 – 734ст.
2. Andrew Troelsen, Philip Japikse - PRO C# //With .NET and .NET Core //Eighth Edition. – 1410ст.
3. w3schools.com – SQL – Електронний доступ:
https://www.w3schools.com/sql/sql_intro.asp
4. METANIT.COM//Сайт про програмування. Електронний доступ:
<https://metanit.com/sql/>.
5. Бен Форта - Microsoft SQL Server T-SQL in 10 Minutes, 2nd Edition. 2017 – 384ст.
6. Новик Володимир Довідник. SQL, PL / SQL, SQL * Plus – Ізраїль: 2001. – 126 с.
- 7.
8. Бондар А.Г. - Microsoft SQL Server 2012. 2013 – 308ст.
9. Microsoft Corporation - Microsoft SQL Server 2005. Реалізація і обслуговування 2005р.
10. Пол Нільсон - Microsoft SQL Server 2005. Біблія користувача. 2008р – 1232ст.
11. Пасічник В. В., Резниченко В. А. Організація баз даних та знань. – К.: Видавнича група ВНУ, 2006. –384 с.
12. Неня А. В. Організація баз даних та знань: конспект лекцій для студентів заочної форми навчання. – Суми: Вид-во СумДУ, 2010. – 109 с.
13. Тімоті Бадд, Об'єктно-орієнтоване програмування у дії, 3.-Пб., 1997

					<i>КНТЕУ 121 07-21.БР</i>			
Зм.	Аркуш	№ докум	Підпис	Дата				
Зав. кафедри	Криворучко О.В.				Розробка програмного забезпечення управління ресторанним бізнесом	Стадія	Аркуш	Аркушів
Керівник	Жирова Т.О.					СД	35	36
Гарант	Цензура М. О.					Факультет інформаційних технологій, 4 курс, 7 група		
Розробник.	Тараканов Є.О.							
					Список використаних джерел			

14. Dmitri Korotkevitch - Expert SQL Server Transactions and Locking: Concurrency Internals for SQL Server Practitioners. 2018 – 340ст.
15. Grant Fritchey - SQL Server 2017 Query Performance Tuning: Troubleshoot and Optimize Query Performance. 2018р. – 932ст.
16. Kalen Delaney Craig Freedman - Microsoft SQL Server 2012 Internals. 2013 – 982ст.
17. METANIT.COM//Сайт про програмування. Електронний доступ: <https://metanit.com/sql/>.
18. Мартин Грубер – Розуміння SQL. Москва 1993. – 291.
19. Host IQ/DATABASE. Електронний доступ: <https://hostiq.ua/wiki/ukr/database/>

						Аркуш
						36
Зм.	Аркуш	№ докум	Підпис	Дата	КНТЕУ 121 07-21.БР	

ДОДАТКИ

Додаток А

```
public void ShowOrders()
{
    using (SqlConnection sqlConnection = new
SqlConnection(connectionString))
    {
        sqlConnection.Open();
        SqlCommand command = new
SqlCommand("ShowOrders", sqlConnection);
        SqlCommand command2 = new
SqlCommand("ShowOrdersDetails", sqlConnection);
        command.CommandType =
CommandType.StoredProcedure;
        command2.CommandType =
CommandType.StoredProcedure;
        DataTable dt = new DataTable();
        DataTable dt2 = new DataTable();
        dt.Load(command.ExecuteReader());
        dt2.Load(command2.ExecuteReader());
        dataGridView3.DataSource = dt;
        dataGridView4.DataSource = dt2;

        sqlConnection.Close();
    }
}
```

```
private async void button1_Click(object sender, EventArgs e)
{
    using (SqlConnection sqlConnection = new
SqlConnection(connectionString))
    {
        sqlConnection.Open();
        SqlCommand command = new
SqlCommand("[GetAllMealsByOrders]", sqlConnection);
        command.CommandType =
CommandType.StoredProcedure;
        command.Parameters.AddWithValue("@Id",
SqlDbType.Text).Value = textBox1.Text.Trim();

        command.ExecuteNonQuery();
        DataTable dt = new DataTable();
        dt.Load(command.ExecuteReader());
        dataGridView2.DataSource = dt;

        SqlDataReader sqlReader = null;

        sqlConnection.Close();
    }
    using (SqlConnection sqlConnection = new
SqlConnection(connectionString))
    {
        sqlConnection.Open();
        SqlCommand command = new
SqlCommand("[GetPriceByOrders]", sqlConnection);
        command.CommandType =
CommandType.StoredProcedure;
        command.Parameters.AddWithValue("@Id",
SqlDbType.Text).Value = textBox1.Text.Trim();
        SqlDataReader sqlReader = null;

        try
        {
            sqlReader = await
command.ExecuteReaderAsync();
            while (await sqlReader.ReadAsync())
            {
```

```
        label2.Text =  
        Convert.ToString(sqlReader["Загальна_вартість"]);  
        //MessageBox.Show(Convert.ToString(sqlReader["Загальна_вартість"]));  
    }  
    }  
    catch (Exception ex)  
    {  
        MessageBox.Show(ex.Message);  
    }  
    }  
}
```

```
private void button6_Click(object sender, EventArgs e)
{
    using (SqlConnection sqlConnection = new
SqlConnection(connectionString))
    {
        sqlConnection.Open();
        SqlCommand command = new
SqlCommand("[Groupingg]", sqlConnection);
        command.CommandType =
CommandType.StoredProcedure;
        command.Parameters.AddWithValue("@dateStart",
SqlDbType.NVarChar).Value = dateTimePicker3.Value;
        command.Parameters.AddWithValue("@dateFinish",
SqlDbType.NVarChar).Value = dateTimePicker2.Value;
        command.Parameters.AddWithValue("@typeMeal",
SqlDbType.NVarChar).Value = comboBox3.Text.Trim();
        command.Parameters.AddWithValue("@Employee",
SqlDbType.Text).Value = textBox7.Text.Trim();
        command.Parameters.AddWithValue("@meal",
SqlDbType.Text).Value = textBox4.Text.Trim();

        command.ExecuteNonQuery();
        DataTable dt = new DataTable();
        dt.Load(command.ExecuteReader());
        dataGridView5.DataSource = dt;
        sqlConnection.Close();
    }
}
```